



CK 802 紧耦合 IP 用户手册

USERGUIDE

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).



声明:

杭州中天微系统有限公司(C-SKY Microsystems Co.,Ltd)保留本文档的所有权利。
本文档的内容有可能发生更改、更新、删除、变动,恕不另行通知。

版权所有 © 2001-2017 杭州中天微系统有限公司

公司地址: 杭州市西湖区西斗门路 3 号天堂软件园 A 座 15 层

邮政编码: 310012

电话: 0571-88157059

传真: 0571-88157059-8888

主页: www.c-sky.com

E-mail: info@c-sky.com



C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

版本历史:

版本	日期	描述	作者
1.0	11/21/2011	第一版	杭州中天微系统有限公司
1.1	09/02/2014	1、删除加解密模块 Scrambler 2、加入片内高速缓存控制寄存器单元 CRU 3、修正系统计时器 CoreTim 4、修正矢量中断控制器 VIC	杭州中天微系统有限公司
1.1.1	02/04/2015	1、修改 cache 控制寄存器内容 2、修改 PWM 复位值	杭州中天微系统有限公司
1.1.2	05/12/2015	1、加入总线峰值功耗管理	杭州中天微系统有限公司
1.1.3	08/12/2016	1、部分勘误	杭州中天微系统有限公司
1.14	04/14/2017	1、增加 Tspend 中断寄存器介绍 2、增加第五章高速缓存部分 3、增加第六章调试寄存器映射部分	杭州中天微系统有限公司
1.5	4/24/2017	1、增加中断设置示例	杭州中天微系统有限公司
1.6	01/12/2018	1、增加中断源数量配置	杭州中天微系统有限公司

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

目录:

1. 紧耦合 IP 简介	7
2. 系统计时器	9
2.1. 简介	9
2.2. 寄存器定义	9
2.2.1. 控制和状态寄存器 (CORET_CSR)	9
2.2.2. 回填值寄存器 (CORET_RVR)	10
2.2.3. 当前值寄存器 (CORET_CVR)	11
2.2.4. 校准寄存器 (CORET_CALIB)	12
2.3. 操作步骤	12
2.4. 接口信号	13
3. 矢量中断控制器	15
3.1. 简介	15
3.2. 寄存器定义	16
3.2.1. 中断使能设置寄存器 (VIC_ISER)	18
3.2.2. 中断低功耗唤醒设置寄存器 (VIC_IWER)	19
3.2.3. 中断使能清除寄存器 (VIC_ICER)	20
3.2.4. 中断低功耗唤醒清除寄存器 (VIC_IWDR)	20
3.2.5. 中断等待设置寄存器 (VIC_ISPR)	21
3.2.6. 中断等待清除寄存器 (VIC_ICPR)	21
3.2.7. 中断响应状态寄存器 (VIC_IABR)	22
3.2.8. 中断优先级设置寄存器 (VIC_IPR0 - VIC_IPR7)	22
3.2.9. 中断状态寄存器 (VIC_ISR)	24
3.2.10. 中断优先级阈值寄存器 (VIC_IPTR)	24
3.2.11. Tspend 中断使能设置寄存器 (VIC_TSPEND)	25
3.2.12. Tspend 中断响应状态寄存器 (VIC_TSABR)	25
3.2.13. Tspend 中断优先级设置寄存器 (VIC_TSPR)	26
3.3. 中断处理机制	26
3.3.1. 中断状态位	27
3.3.2. 中断优先级	27
3.3.3. 中断向量号	28
3.3.4. 中断处理过程	29
3.3.5. 中断嵌套	29
3.4. TSPEND 中断	31
3.5. 操作步骤	32
3.6. 接口信号	32
3.7. 中断使能初始化	35
4. 功耗管理模块	37
4.1. 简介	37

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

No:

4.2. 寄存器定义	37
4.2.1. 控制寄存器 (PCR)	37
4.3. 操作步骤	38
4.4. 接口信号	38
5. 高速缓存控制寄存器单元	40
5.1. 简介	40
5.2. 寄存器定义	40
5.2.1. 高速缓存使能寄存器(CER).....	40
5.2.2. 高速缓存无效寄存器(CIR).....	41
5.2.3. 可高缓区配置寄存器 0~3 (CRCR)	41
5.2.4. 寄存器使用说明.....	42
5.3. 操作步骤	43
5.4. 接口信号	43
6. 调试寄存器映射	46
6.1. 简介	46
6.2. 寄存器定义	46
6.3. 地址观测异常	47
6.3.1. 简介.....	47
6.3.2. 地址观测异常相关寄存器.....	47
6.4. HAD 直接访问内存功能 (DDMA)	48
6.4.1. 简介.....	48
6.4.2. DDMA 寄存器定义.....	49
6.4.3. DDMA 操作示例.....	50

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

图表目录:

图表 1-1 紧耦合 IP 主要功能.....	7
图表 1-2 紧耦合 IP 的系统结构图.....	7
图表 1-3 紧耦合 IP 的内存地址分配.....	8
图表 2-1 系统计时器结构图.....	9
图表 2-2 系统计时器寄存器定义.....	9
图表 2-3 系统计时器控制与状态寄存器.....	10
图表 2-4 系统计时器控制与状态寄存器域描述.....	10
图表 2-5 系统计时器回填值寄存器.....	11
图表 2-6 系统计时器回填值寄存器域描述.....	11
图表 2-7 系统计时器当前值寄存器.....	11
图表 2-8 系统计时器当前值寄存器域描述.....	11
图表 2-9 系统计时器校准寄存器.....	12
图表 2-10 系统计时器校准寄存器域描述.....	12
图表 2-11 系统计时器接口信号.....	14
图表 3-1 矢量中断控制器系统结构图.....	15
图表 3-2 矢量中断控制器寄存器定义.....	18
图表 3-3 中断使能设置寄存器.....	19
图表 3-4 中断使能设置寄存器域定义.....	19
图表 3-5 中断低功耗唤醒设置寄存器.....	19
图表 3-6 低功耗唤醒使能设置寄存器域定义.....	19
图表 3-7 中断使能清除寄存器.....	20
图表 3-8 中断使能清除寄存器域描述.....	20
图表 3-9 中断低功耗唤醒清除寄存器.....	20
图表 3-10 中断低功耗唤醒清除寄存器域描述.....	21
图表 3-11 中断等待设置寄存器.....	21
图表 3-12 中断等待设置寄存器域描述.....	21
图表 3-13 中断等待清除寄存器.....	21
图表 3-14 中断等待清除寄存器.....	21
图表 3-15 中断响应状态寄存器.....	22
图表 3-16 中断响应状态寄存器域描述.....	22
图表 3-17 中断优先级设置寄存器整体分布.....	22
图表 3-18 中断优先级设置寄存器.....	23
图表 3-19 中断优先级设置寄存器域描述.....	23
图表 3-20 中断状态寄存器.....	24
图表 3-21 中断状态寄存器域描述.....	24
图表 3-22 中断优先级阈值寄存器.....	24
图表 3-23 中断优先级阈值寄存器域描述.....	25
图表 3-24 TSPEND 中断使能设置寄存器.....	25
图表 3-25 TSPEND 中断使能设置寄存器域定义.....	25
图表 3-26 TSPEND 中断响应状态寄存器.....	25
图表 3-27 TSPEND 中断响应状态寄存器域描述.....	26

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

No:

图表 3-28 TSPEND 中断优先级设置寄存器	26
图表 3-29 TSPEND 中断优先级设置寄存器域描述	26
图表 3-30 中断向量号描述	28
图表 3-31 中断源请求同步电路示例	29
图表 3-32 中断嵌套优先级示例	30
图表 3-33 中断嵌套时机条件示例	31
图表 3-34 矢量中断控制器接口信号	34
图表 3-35 没有配置矢量中断控制器的接口信号	34
图表 4-1 功耗管理模块系统结构图	37
图表 4-2 功耗管理模块寄存器定义	37
图表 4-3 功耗管理模块控制寄存器	38
图表 4-4 功耗管理模块控制寄存器域描述	38
图表 4-5 功耗管理模块接口信号描述	39
图表 5-1 CACHE 控制寄存器单元结构图	40
图表 5-2 CACHE 控制寄存器定义	40
图表 5-3 高速缓存使能寄存器	41
图表 5-4 高速缓存无效寄存器	41
图表 5-5 可高缓区配置寄存器	41
图表 5-6 可高缓区大小配置和其对基址要求	41
图表 5-7 CACHE 控制寄存器单元接口信号描述	45
图表 6-1 HAD 寄存器 TCIP 映射地址说明	46
图表 6-2 HAD 寄存器表	46
图表 6-3 CPU 和 HAD 交互	47
图表 6-4 MBIR 寄存器	48
图表 6-5 DDMA 功能示意图	48
图表 6-6 DACSR 寄存器	49
图表 6-7 DACSR 表项说明	49
图表 6-8 DATR 寄存器	49
图表 6-9 DARWR 寄存器	50
图表 6-10 DARWR 驱动总线传输	50

C-Sky Confidential

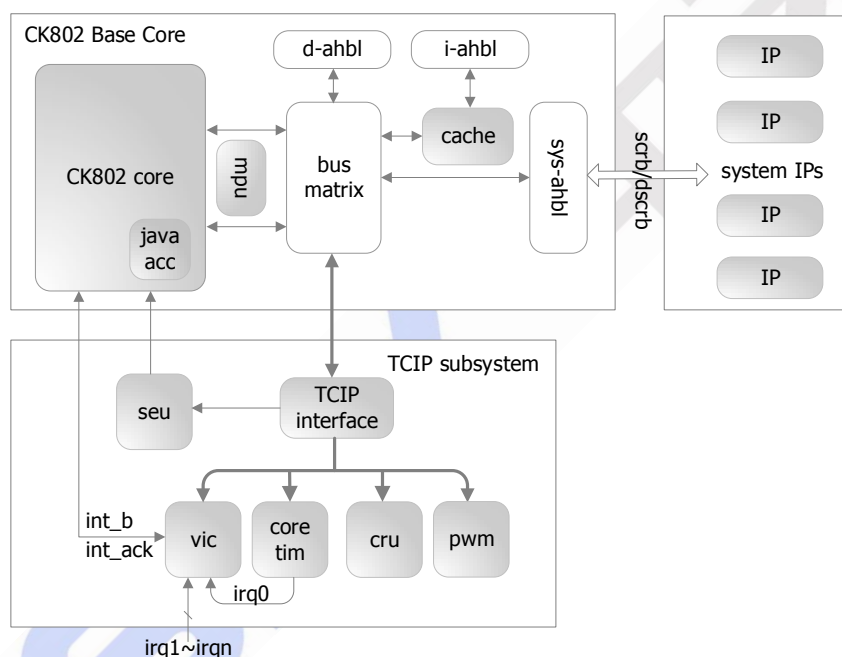
The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

1. 紧耦合 IP 简介

为了提高 CK802 的系统集成度，方便用户集成与使用，CK802 实现了一系列与处理器关系密切的系统关键 IP，这些 IP 统称为紧耦合 IP（Tightly Coupled IP，TCIP）。CK802 的紧耦合 IP 包括系统计时器 CoreTim、矢量中断控制器 VIC、功耗管理模块 PWRM、片内高速缓存控制寄存器单元 CRU。这些紧耦合 IP 配合 CK802，外加存储器等少量资源，便可以组成一款最小功能的 SoC 系统，提高了用户使用 CK802 的便捷性，减少了 CK802 的开发与应用成本。CK802 的紧耦合 IP 主要功能如图表 1-1，系统结构图如图表 1-2。

IP名	主要功能
系统计时器	完成系统的计时功能，可在低功耗时唤醒 CPU
矢量中断控制器	完成中断的收集、仲裁、硬件嵌套以及与处理器的交互
功耗管理模块	控制 CK802 的峰值功耗和平均功耗
片内高速缓存控制寄存器单元	设置 CK802 片内高速缓存，如开关 Cache，配置可高缓的区域等
调试寄存器映射区	能通过 TCIP 来控制调试寄存器

图表 1-1 紧耦合 IP 主要功能



图表 1-2 紧耦合 IP 的系统结构图

与传统 IP 不同，紧耦合 IP 通过专用的紧耦合 IP 总线接口与处理器相连，无需通过系统总线访问。其中，紧耦合 IP 总线接口直接与内核的总线互联单元（Bus Matrix Unit, BMU）相连，支持单个 CPU 时钟周期的紧耦合 IP 访问传输，不仅提高了紧耦合 IP 的访问效率，而且提高了系统集成效率。紧耦合 IP 与其它系统 IP 共享统一的内存地址空间，通过传输指

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

No:

令（Load）和存储指令（Store）进行寄存器访问和功能控制。紧耦合 IP 的内存地址分配如图表 1-3 所示。

IP名	内存地址空间
系统计时器	0xE000E010-0xE000E0FF
矢量中断控制器	0xE000E100-0xE000ECFF
功耗管理模块	0xE000EF90-0xE000EF9F
片内高速缓存控制寄存器单元	0xE000F000-0xE000FFFF
调试寄存器映射	0xE0011000 -0xE001117C

图表 1-3 紧耦合 IP 的内存地址分配

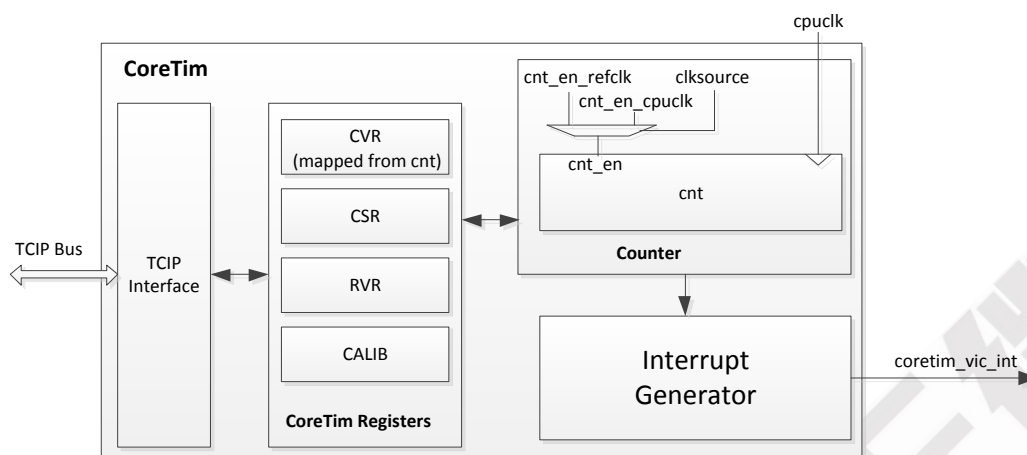
紧耦合 IP 除了通过专用总线接口与 CK802 发生通信，紧耦合 IP 还以直接相连的方式与处理器进行功能交互。其中，矢量中断控制器将仲裁之后的中断信息传送给处理器并接受处理器返回的中断响应信号；片内高速缓存位于 CK802 总线互联单元（BMU）和指令总线接口单元（I-AHBL）之间；功耗管理单元直接对 CK802 两级流水线进行控制以调整处理器功耗。

紧耦合 IP 的所有控制寄存器都是 32-bit 的寄存器，因此只能通过以 word 为单位进行访问传输，任何以 half-word 或者 byte 为单位的访问都将造成不可预期的错误，并且紧耦合 IP 寄存器只支持小端格式。所有对紧耦合 IP 寄存器的访问均需在超级用户模式下才可进行，在普通用户模式下访问会产生访问错误异常。

2. 系统计时器

2.1. 简介

系统计时器是 CK802 的一个可选模块，它主要用于计时。系统计时器提供了一个简单易用的 24 位循环递减的计数器，当系统计时器使能时，计数器开始工作。当计数器递减到 0 时，系统计时器会向矢量中断控制器发起中断请求，申请获得处理器响应并处理系统计时器的事务。系统计时器的结构框图如图表 2-1 所示。



图表 2-1 系统计时器结构图

2.2. 寄存器定义

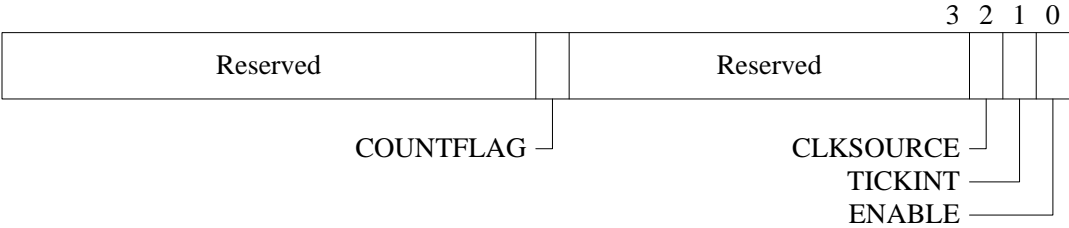
系统计时器每一个寄存器宽度是 32 位，寄存器地址空间为：

地址	名称	类型	初始值	描述
0xE000E010	CORET_CSR	读/写	0x00000004	控制状态寄存器
0xE000E014	CORET_RVR	读/写	-	回填值寄存器
0xE000E018	CORET_CVR	读/写	-	当前值寄存器
0xE000E01C	CORET_CALIB	只读	-	校准寄存器
0xE000E020-0xE000E0FF	-	-	-	保留

图表 2-2 系统计时器寄存器定义

2.2.1. 控制和状态寄存器（CORET_CSR）

CORET_CSR 是系统计时器的控制和状态寄存器如图表 2-3 所示，CORET_CSR 寄存器的位说明如图表 2-4 所示。



图表 2-3 系统计时器控制与状态寄存器

位	类型	名称	描述
31:17	-	-	保留
16	只读	COUNTFLAG	表示在上一次读此寄存器后计数器是否计数到 0： 0 计数器还没有计数到 0 1 计数器已经计数到 0 在计数器的值由 1 变到 0 时，COUNTFLAG 会被置位。 读 CSR 寄存器以及任何写 CVR 寄存器会使 COUNTFLAG 清零。
15:3	-	-	保留
2	读/写	CLKSOURCE	表示系统计时器的时钟源： 0 用可选的外部参考时钟作为计数器的时钟源 1 用内部时钟作为计数器的时钟源 如果没有外部时钟，读此位将返回 1，写此位没有任何作用。外部参考时钟频率必须小于或等于内部时钟频率的一半。
1	读/写	TICKINT	表示计数到 0 时系统计时器是否触发中断： 0 计数到 0 时系统计时器不会触发中断 1 计数到 0 时系统计时器将触发中断 写 CVR 寄存器会使计数器清零，但该方法不会导致系统计时器触发中断。
0	读/写	ENABLE	表示系统计时器的使能状态位： 0 计数器没有使能 1 计数器使能

图表 2-4 系统计时器控制与状态寄存器域描述

2.2.2. 回填值寄存器（CORET_RVR）

CORET_RVR 寄存器用于在每一次计数循环开始时给 CORET_CVR 寄存器赋值，CORET_RVR 寄存器及其位说明如图表 2-5，图表 2-6 所示。



图表 2-5 系统计时器回填值寄存器

位	名称	描述
31:24	-	保留
23:0	RELOAD	<p>在计数器计数到 0 时, RELOAD 值会被赋给 CORET_CVR 寄存器。</p> <p>向 CORET_RVR 寄存器写 0 会使计数器在完成当前次计数后停止工作, 此后计数器的值将一直保持为 0。</p> <p>当使用外部参考时钟使能计数器后, RELOAD 值通过外部时钟赋值给 CORET_CVR, 因此必须在确保 RELOAD 赋值成功, 计数器正常计数开始后 (即 CORET_CVR 变为非 0 值时), 才可以将 CORET_RVR 置为 0, 从而让计数器在完成当前次计数后停止工作, 否则计数器将无法开始计数。</p>

图表 2-6 系统计时器回填值寄存器域描述

2.2.2.1. 如何计算 RELOAD 值

RELOAD 的正常取值范围在 0x1-0x00FFFFFF 之间。要产生一个周期为 N 个计数时钟周期的计时器, RELOAD 的值需要被赋为 N-1。比如要在每 100 计数时钟周期时产生一个 CoreTim 中断, 需要给 RELOAD 赋值 99。

2.2.3. 当前值寄存器 (CORET_CVR)

CORET_CVR 包含了系统计时器的当前值, CORET_CVR 寄存器及其位说明如图表 2-7, 图表 2-8 所示。

31	24	23	0
Reserved	CURRENT		

图表 2-7 系统计时器当前值寄存器

位	名称	描述
31:24	-	保留
23:0	CURRENT	<p>它指示了计数器在被读取时的值。</p> <p>写 CORET_CVR 寄存器会同时使此寄存器和 CORET_CSR 寄存器中的 COUNTFLAG 状态位清零, 进一步, 它会导致系统计时器在下一个时钟周期取出寄存器 CORET_RVR 的值并赋给 CORET_CVR。</p> <p>注意: 写 CORET_CVR 不会导致系统计时器触发中断, 读 CORET_CVR 会返回当前计数器的值。</p>

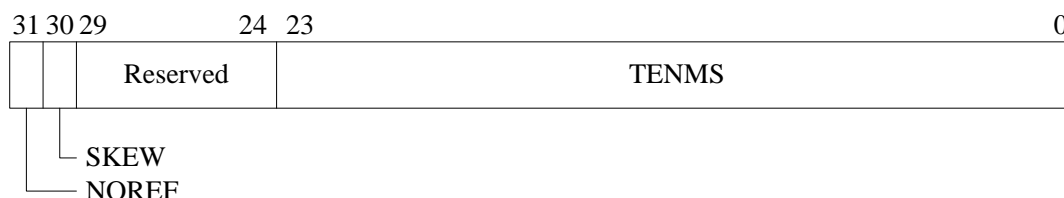
图表 2-8 系统计时器当前值寄存器域描述

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

2.2.4. 校准寄存器（CORET_CALIB）

CORET_CALIB 寄存器描述了系统计时器的校准功能。它的复位值跟具体实现相关：需要从设备提供商提供的文档里得到关于 CORET_CALIB 位信息的含义，以及 CORET_CALIB 寄存器中的计时器校验值 TENMS。根据 TENMS 这个校准值，软件可以通过将这个值乘上一定的比例，从而得到其他不同的计数周期，此计数周期必须在计数器的取值范围之内；CORET_CALIB 寄存器及其位说明如图表 2-9，图表 2-10 所示。



图表 2-9 系统计时器校准寄存器

位	名称	描述
31	NOREF	表示设备是否实现了外部参考时钟： 0 设备有外部参考时钟 1 设备没有外部参考时钟 当这位是 1 时，CORET_CSR 寄存器的 CLKSOURCE 位固定为 1，不能被改写。
30	SKEW	表示 10ms 校准值是否准确无误： 0 10ms 校准值准确无误 1 10ms 校准值由于时钟频率的问题而有误差
29:24	-	保留
23:0	TENMS	用以表示 10ms 时间对应的回填值。根据 SKEW 具体值的不同，它可能表示完全准确的 10ms 值或者最接近 10ms 的值。 如果这个域的值是 0，表示校准值未知。这可能是因为参考时钟是一个未知的输入或者是动态变化的。

图表 2-10 系统计时器校准寄存器域描述

注意：如果将 CORET_RVR 设置为 0，那么 CoreTim 计数器将在下一回合停止工作，而不管计数器的使能位状态。

SYST_CVR 寄存器的值在复位时是未知的。在使能 CoreTim 计数器之前，软件必须先将需要的计数值写入 CORET_RVR 寄存器，然后再向 CORET_CVR 写入任意值，后一操作会使 CORET_CVR 的值清零。这样在使能计数器后，计数器就可以读取 CORET_RVR 并从这个值开始向下计数，从而避免了从一个任意的值开始计数。

2.3. 操作步骤

由于系统计时器中 CORET_RVR 和 CORET_CVR 两个寄存器没有复位值，在系统计时器工作之前，必须按照下列步骤进行操作：

C-Sky Confidential

No:

- 1) 向 CORET_RVR 寄存器里写入需要的回填值;
- 2) 向 CORET_CVR 寄存器里写入任意值从而使它清零;
- 3) 操作 CORET_CSR 寄存器, 使能系统计时器。

2.4. 接口信号

信号名	I/O	Reset	定义
紧耦合IP总线接口信号:			
tcipif_coretim_sel	I	0	选中信号: 指示选中系统计时器并进行数据传输。 1: 指示选中 0: 指示未选中
tcipif_coretim_addr[15:0]	I	-	地址总线: 16 位地址总线 (截取 32 位地址总线的低 16 位), 指示访问地址。
tcipif_coretim_write	I	0	读写表示信号: 指示当前 TCIP 访问是读取数据还是写数据: 1: 指示是写访问; 0: 指示是读访问。
tcipif_coretim_wdata[31:0]	I	-	写数据总线: 32 位写数据总线。
coretim_tcipif_rdata[31:0]	O	-	读数据总线: 32 位读数据总线。
coretim_tcipif_cmplt	O	0	传输完成指示信号: 有效时指示当前传输已完成。
处理器相关信号:			
core_dbgcon	I	0	处理器调试模式: 指示处理器处于调试模式。此时, 系统计时器停止计时。 1: 指示处理器处于调试模式; 0: 指示处理器未处于调试模式。
中断信号:			
ctim_pad_int_vld	O	0	中断有效指示信号: 指示系统计时器产生中断, 该信号高电平有效。
时钟信号:			

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

No:

信号名	I/O	Reset	定义
forever_cpucclk	I	-	提供 CPU 内核工作时钟： CoreTim 中与 CPU 低功耗状态唤醒无关的寄存器均工作于该时钟。
forever_cpucclk_nogated	I	-	提供 CPU 内核工作时钟： CoreTim 中与 CPU 低功耗状态唤醒相关的寄存器均工作于该时钟。
pad_ctim_refclk	I	-	CoreTim 的外部参考时钟。 CoreTim 先将 pad_ctim_refclk 同步到 forever_cpucclk_nogated 域，然后再采样该信号的上升沿进行计数。 pad_ctim_refclk 的频率必须小于 forever_cpucclk_nogated 频率的一半。
pad_ctim_calib[25:0]	I	-	[25]表示是否提供参考时钟 0: 提供参考时钟 1: 没有提供参考时钟 [24]表示校准值是否精准 0: 10ms 校准值是精准的 1: 10ms 校准值是不精准的 [23:0]表示 10ms 时间校准值
复位信号:			
cpurst_b	I	-	系统计时器复位信号： 低电平时，初始化系统计时器内部。系统计时器采用异步复位方式。
其它信号:			
pad_yy_gate_clk_en_b	I	-	门控时钟使能信号： 只有当这个信号有效时，CoreTim 的内部模块的门控时钟才能有效。 不用该信号时，需要接 1。
pad_yy_test_mode	I	-	进入测试模式： 使 CoreTim 进入测试模式，此时 CoreTim 时钟为测试时钟（pad_had_jtg_tclk）。只有 CoreTim 进入测试模式，并且处理器输入信号 pad_yy_scan_enable 有效时，才可以通过扫描链进行测试。不用该信号时，需要接 0。

图表 2-11 系统计时器接口信号

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

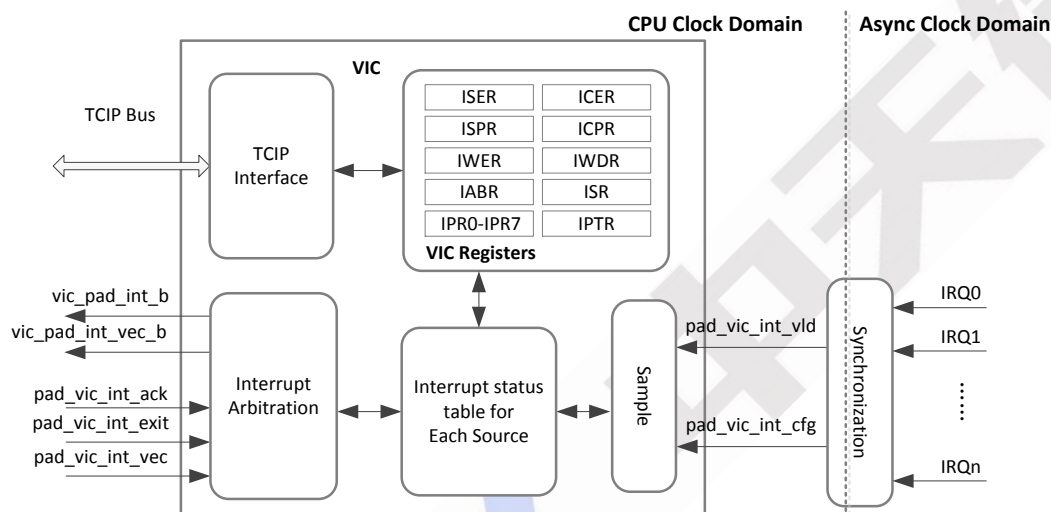
3. 矢量中断控制器

3.1. 简介

矢量中断控制器（VIC）是一个与 CK802 紧耦合的 IP 单元，用于中断的高效处理。矢量中断控制器最大可支持 128 个中断源（IRQ[127:0]），每个中断源拥有独立软件可编程的中断优先级。矢量中断控制器收集来自不同中断源的中断请求，依据中断优先级对中断请求进行仲裁。最高优先级的中断将获得中断控制权并向处理器发出中断请求。当处理器响应了中断请求，处理器返回中断请求响应信号给 VIC；当处理器退出中断服务程序（ISR），处理器返回中断退出信号给 VIC。

矢量中断控制器支持中断嵌套。当处理器正在处理一个中断请求的同时来了一个更高优先级的中断请求，处理器将中断当前中断服务程序的处理，响应该更高优先级的中断请求。在更高优先级的中断请求处理结束时，CPU 返回被打断的中断服务程序继续执行。矢量中断控制器允许高优先级的中断请求抢占低优先级的中断请求，但不允许同级别或者低优先级的中断抢占，保证了中断响应的实时性。

矢量中断控制器的系统结构图如图表 3-1 所示。



图表 3-1 矢量中断控制器系统结构图

矢量中断控制器支持以下功能：

- 中断数量硬件可配，支持 4、8、16、24、32、64、96 和 128；
- 软件通过 8 比特寄存器为每个中断配置优先级，在 CK802 中若中断源数量小于或等于 32 个则仅使用 8 位中的最高 2 位有效位表征优先级高低，其余位始终保持为 0；若中断源数量为 64 个则可以使用 8 比特中的最高 3 位来表征 8 个优先级；若中断源数量为 96 或 128 个则可以使用 8 比特中的最高 4 位来表征 16 个优先级。优先级从低到高排列，0 级优先级为最高优先级，P（P=4，8，16）级优先级为最低优先级；
- 支持电平和脉冲两种中断源信号；
- 中断在处理的过程中支持优先级的动态调整，通过设置优先级阈值寄存器以较小的硬件代价实现中断优先级的反转；

C-Sky Confidential

- 支持中断嵌套, CPU 在执行中断服务程序过程中, 允许更高优先级的中断抢占。

3.2. 寄存器定义

VIC 提供一组 32-bit 的寄存器, 各个寄存器的地址空间如图表 3-2 所示。

地址	名称	类型	初始值	描述
0xE000E100	VIC_ISER0	读/写	0x00000000	中断使能设置寄存器 (0-31 号中断)
0xE000E104	VIC_ISER1	读/写	0x00000000	中断使能设置寄存器 (32-63 号中断)
0xE000E108	VIC_ISER2	读/写	0x00000000	中断使能设置寄存器 (64-95 号中断)
0xE000E10C	VIC_ISER3	读/写	0x00000000	中断使能设置寄存器 (96-127 号中断)
0xE000E110- 0xE000E13F	-	-	-	保留
0xE000E140	VIC_IWER0	读/写	0x00000000	低功耗唤醒设置寄存器 (0-31 号中断)
0xE000E144	VIC_IWER1	读/写	0x00000000	低功耗唤醒设置寄存器 (32-63 号中断)
0xE000E148	VIC_IWER2	读/写	0x00000000	低功耗唤醒设置寄存器 (64-95 号中断)
0xE000E14C	VIC_IWER3	读/写	0x00000000	低功耗唤醒设置寄存器 (96-127 号中断)
0xE000E150- 0xE000E17F	-	-	-	保留
0xE000E180	VIC_ICER0	读/写	0x00000000	中断使能清除寄存器 (0-31 号中断)
0xE000E184	VIC_ICER1	读/写	0x00000000	中断使能清除寄存器 (32-63 号中断)
0xE000E188	VIC_ICER2	读/写	0x00000000	中断使能清除寄存器 (64-95 号中断)
0xE000E18C	VIC_ICER3	读/写	0x00000000	中断使能清除寄存器 (96-127 号中断)
0xE000E190-	-	-	-	保留

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

No:

地址	名称	类型	初始值	描述
0xE000E1BF				
0xE000E1C0	VIC_IWDR0	读/写	0x00000000	低功耗唤醒清除寄存器 (0-31 号中断)
0xE000E1C4	VIC_IWDR1	读/写	0x00000000	低功耗唤醒清除寄存器 (32-63 号中断)
0xE000E1C8	VIC_IWDR2	读/写	0x00000000	低功耗唤醒清除寄存器 (64-95 号中断)
0xE000E1CC	VIC_IWDR3	读/写	0x00000000	低功耗唤醒清除寄存器 (96-127 号中断)
0xE000E1D0- 0xE000E1FF	-	-	-	保留
0xE000E200	VIC_ISPR0	读/写	0x00000000	中断等待设置寄存器 (0-31 号中断)
0xE000E204	VIC_ISPR1	读/写	0x00000000	中断等待设置寄存器 (32-63 号中断)
0xE000E208	VIC_ISPR2	读/写	0x00000000	中断等待设置寄存器 (64-95 号中断)
0xE000E20C	VIC_ISPR3	读/写	0x00000000	中断等待设置寄存器 (96-127 号中断)
0xE000E210- 0xE000E27F	-	-	-	保留
0xE000E280	VIC_ICPR0	读/写	0x00000000	中断等待清除寄存器 (0-31 号中断)
0xE000E284	VIC_ICPR1	读/写	0x00000000	中断等待清除寄存器 (32-63 号中断)
0xE000E288	VIC_ICPR2	读/写	0x00000000	中断等待清除寄存器 (64-95 号中断)
0xE000E28C	VIC_ICPR3	读/写	0x00000000	中断等待清除寄存器 (96-127 号中断)
0xE000E290- 0xE000E2FF	-	-	-	保留

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).



地址	名称	类型	初始值	描述
0xE000E300	VIC_IABR0	读/写	0x00000000	中断响应状态寄存器 (0-31 号中断)
0xE000E304	VIC_IABR1	读/写	0x00000000	中断响应状态寄存器 (32-63 号中断)
0xE000E308	VIC_IABR2	读/写	0x00000000	中断响应状态寄存器 (64-95 号中断)
0xE000E30C	VIC_IABR3	读/写	0x00000000	中断响应状态寄存器 (96-127 号中断)
0xE000E310- 0xE000E3FF				
0xE000E400- 0xE000E47C	VIC_IPR0- VIC_IPR31	读/写	0x00000000	中断优先级设置寄存器
0xE000E480- 0xE000EBFF	-	-	-	保留
0xE000EC00	VIC_ISR	只读	0x00000000	中断状态寄存器
0xE000EC04	VIC_IPTR	读/写	0x00000000	中断优先级阈值寄存器
0xE000EC08-	VIC_TSPEND	读/写	0x00000000	Tspending 使能设置寄 存器
0xE000EC0C	VIC_TSABR	读/写	0x00000000	Tspending 响应状态寄 存器
0xE000EC10	VIC_TSPR	读/写	0x00000000	Tspending 等待设置寄 存器
0xE000EC14- 0xE000ECFF	-	-	-	保留

图表 3-2 矢量中断控制器寄存器定义

3.2.1. 中断使能设置寄存器（VIC_ISER）

VIC_ISER 用于使能各个中断，并且反馈各个中断的使能状态。图表 3-3 描述了 VIC_ISER 的位分布，图表 3-4 描述了 VIC_ISER 的位定义。

31	0
SETENA	



图表 3-3 中断使能设置寄存器

位	名称	描述
31:0	SETENA	设置使用，读取一个或者多个中断的使能状态。每一个位对应相同编号的中断源： 读操作 0 对应中断未使能。 1 对应中断已使能。 写操作 0 无效。 1 使能对应中断。

图表 3-4 中断使能设置寄存器域定义

如果一个处于等待状态的中断已使能，矢量中断控制器会根据其优先级激活该中断。如果一个中断未使能，该中断即使处于等待状态，矢量中断控制器也不会激活该中断。

3.2.2. 中断低功耗唤醒设置寄存器（VIC_IWER）

VIC_IWER 用于使能各个中断的低功耗唤醒功能，并且反馈各个中断低功耗唤醒的使能状态。图表 3-5 描述了 VIC_IWER 的位分布，图表 3-6 描述了 VIC_IWER 的位定义。



图表 3-5 中断低功耗唤醒设置寄存器

位	名称	描述
31:0	SETENA	设置使用，读取一个或者多个中断低功耗唤醒的使能状态。每一个位对应相同编号的中断源： 读操作 0 对应中断的低功耗唤醒功能未使能。 1 对应中断的低功耗唤醒功能已使能。 写操作 0 无效。 1 使能对应中断的低功耗唤醒功能。

图表 3-6 低功耗唤醒使能设置寄存器域定义

如果一个中断的低功耗唤醒功能已使能且该中断处于等待状态，VIC 产生低功耗唤醒请求。如果一个中断的低功耗唤醒功能未使能，即使该中断处于等待状态，VIC 也不产生低功耗唤醒请求。

注意：中断使能和中断唤醒使能分别控制中断事务和中断唤醒功能。当两者都设置时，一个处于等待状态的中断既产生中断请求又产生低功耗唤醒请求；当只有其中一个使能时，

只激活对应的功能；当两者都没有使能时，即使该中断处于等待状态，也不会产生中断请求或低功耗唤醒请求。

3.2.3. 中断使能清除寄存器（VIC_ICER）

VIC_ICER 用于清除各个中断的使能，并且反馈各个中断的使能状态。图表 3-7 描述了 VIC_ICER 的位分布，图表 3-8 描述了 VIC_ICER 的位定义。



图表 3-7 中断使能清除寄存器

位	名称	描述
31:0	CLRENA	清除使用，读取一个或者多个中断的使能状态。每一个位对应相同编号的中断源： 读操作 0 对应中断未使能。 1 对应中断已使能。 写操作 0 无效。 1 清除对应中断的使能。

图表 3-8 中断使能清除寄存器域描述

3.2.4. 中断低功耗唤醒清除寄存器（VIC_IWDR）

VIC_IWDR 用于清除各个中断的低功耗唤醒使能，并且反馈各个中断低功耗唤醒的使能状态。图表 3-9 描述了 VIC_IWDR 的位分布，图表 3-10 描述了 VIC_IWDR 的位定义。



图表 3-9 中断低功耗唤醒清除寄存器

位	名称	描述
31:0	CLRENA	清除使用，读取一个或者多个中断低功耗唤醒的使能状态。每一个位对应相同编号的中断源： 读操作 0 对应中断的低功耗唤醒功能未使能。 1 对应中断的低功耗唤醒功能已使能。 写操作 0 无效。 1 清除使能对应中断的低功耗唤醒功能。



图表 3-10 中断低功耗唤醒清除寄存器域描述

3.2.5. 中断等待设置寄存器（VIC_ISPR）

VIC_ISPR 表征设置各个中断到等待状态。图表 3-11 描述了 VIC_ISPR 的位分布，图表 3-12 描述了 VIC_ISPR 的位定义。



图表 3-11 中断等待设置寄存器

位	名称	描述
31:0	SETPEND	<p>更改一个或多个中断到等待状态。每一个位对应相同编号的中断源：</p> <p>读操作 0 对应中断未处于等待状态。</p> <p> 1 对应中断处于等待状态。</p> <p>写操作 0 无效。</p> <p> 1 改变对应中断到等待状态。</p>

图表 3-12 中断等待设置寄存器域描述

3.2.6. 中断等待清除寄存器（VIC_ICPR）

VIC_ICPR 表征清除各个中断的等待状态。图表 3-13 描述了 VIC_ICPR 的位分布，图表 3-14 描述了 VIC_ICPR 的位定义。



图表 3-13 中断等待清除寄存器

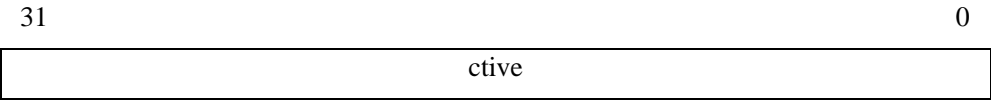
位	名称	描述
31:0	CLRPEND	<p>清除一个或多个中断的等待状态。每一个位对应相同编号的中断源：</p> <p>读操作 0 对应中断处于未等待状态。</p> <p> 1 对应中断处于等待状态。</p> <p>写操作 0 无效。</p> <p> 1 清除对应中断的等待状态。</p>

图表 3-14 中断等待清除寄存器

C-Sky Confidential

3.2.7. 中断响应状态寄存器（VIC_IABR）

VIC_IABR 用于指示各个中断当前的 Active 状态，是一个供软件查询的寄存器,另外，软件可在初始化 VIC 时，将所有中断的 Active 状态清 0。图表 3-15 描述了 VIC_IABR 的位分布，图表 3-16 描述了 VIC_IABR 的位定义。



图表 3-15 中断响应状态寄存器

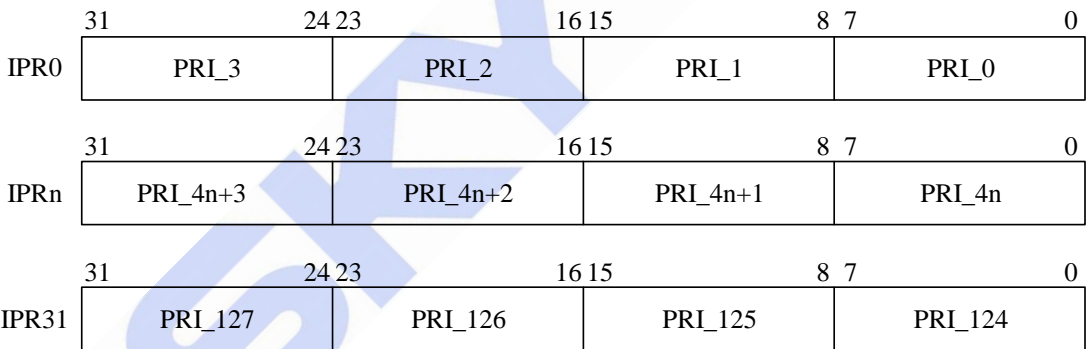
位	名称	描述
31:0	Active	<p>查询位，指示该中断源是否已经被 CPU 响应但还没处理完。每一位对应相同编号的中断源。</p> <p>读操作 0 没有被 CPU 响应</p> <p> 1 已经被 CPU 响应但还没处理完</p> <p>写操作 0 清除中断的 Active 状态（软件不可对该寄存器写 1，否则会导致不可预期的错误）</p>

图表 3-16 中断响应状态寄存器域描述

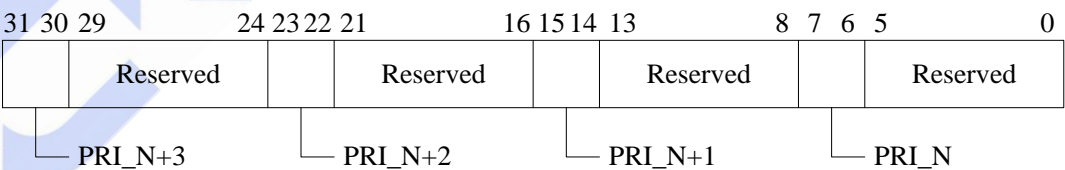
3.2.8. 中断优先级设置寄存器（VIC_IPR0 - VIC_IPR31）

每个中断优先级设置寄存器 提供 4 个中断源的优先级设置。根据应用的定义，每个中断源设置区域对应相应的中断源。对于硬件支持 128 个中断源的实现，寄存器从 IPR0 到 IPR31 如图表 3-17 所示，图表 3-18 描述了描述了中断优先级设置寄存器。

矢量中断控制器根据优先级号选择中断优先级，优先级号越小，优先级越高。如果优先级号相同，根据中断源号决定优先级顺序，号码越小，优先级越高。



图表 3-17 中断优先级设置寄存器整体分布



图表 3-18 中断优先级设置寄存器

图表 3-19 描述了 VIC_IPRn 的位定义。在这张表中， $N = 4n$ ， n 为 VIC_IPRn 寄存器编号。如 VIC_IPR2, n 为 2 则 N 为 8。

优先级阶数	位	名称	描述
2 比特 4 个 优先级（中 断数小于等 于 32 个）	31:30	PRI_N+3	中断号为 N+3 的优先级，值越小优先级越高。
	29:24	-	保留
	23:22	PRI_N+2	中断号为 N+2 的优先级，值越小优先级越高。
	21:16	-	保留
	15:14	PRI_N+1	中断号为 N+1 的优先级，值越小优先级越高。
	13:8	-	保留
	7:6	PRI_N	中断号为 N 的优先级，值越小优先级越高。
	5:0	-	保留
3 比特 8 个 优先级（中 断数量为 64 个）	31:29	PRI_N+3	中断号为 N+3 的优先级，值越小优先级越高。
	28:24	-	保留
	23:21	PRI_N+2	中断号为 N+2 的优先级，值越小优先级越高。
	20:16	-	保留
	15:13	PRI_N+1	中断号为 N+1 的优先级，值越小优先级越高。
	12:8	-	保留
	7:5	PRI_N	中断号为 N 的优先级，值越小优先级越高。
	4:0	-	保留
4 比特 16 个 优先级（中 断数量为 96 或 128 个）	31:28	PRI_N+3	中断号为 N+3 的优先级，值越小优先级越高。
	27:24	-	保留
	23:20	PRI_N+2	中断号为 N+2 的优先级，值越小优先级越高。
	19:16	-	保留
	15:12	PRI_N+1	中断号为 N+1 的优先级，值越小优先级越高。
	11:8	-	保留
	7:4	PRI_N	中断号为 N 的优先级，值越小优先级越高。
	3:0	-	保留

图表 3-19 中断优先级设置寄存器域描述

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

3.2.9. 中断状态寄存器（VIC_ISR）

VIC_ISR 指示了当前 CPU 正在处理的中断向量号和处于等待的优先级最高的中断向量号，该寄存器是一个供软件查询的只读寄存器。图表 3-20 描述了 VIC_ISR 的位分布，图表 3-21 描述了 VIC_ISR 的位定义。

31	21	20	12	11	9	8	0
Reserved				VECTPENDING		Reserved	VECTACTIVE

图表 3-20 中断状态寄存器

位	名称	描述
31:21	Reserved	保留
20:12	VECTPENDING	指示当前处于等待状态的优先级最高的中断向量号；
11:9	Reserved	保留
8:0	VECTACTIVE	指示 CPU 当前正在处理的中断向量号

图表 3-21 中断状态寄存器域描述

3.2.10. 中断优先级阈值寄存器（VIC_IPTR）

VIC_IPTR 定义了当前处于等待状态的中断请求能够发起中断抢占的优先级临界值。处于等待状态的中断请求的优先级必须高于 VIC_IPTR 定义的优先级阈值，才能发起中断抢占请求。图表 3-22 描述了 VIC_IPTR 的位分布，图表 3-23 描述了 VIC_IPTR 的位定义。

31	30	17	16	8	7	0
EN	Reserved				VECTTHRESHOLD	PRIOTHRESHOLD

图表 3-22 中断优先级阈值寄存器

位	名称	描述
31	EN	中断优先级阈值有效位： 0 中断抢占不需要优先级高于阈值 1 中断抢占需要优先级高于阈值
30:17	Reserved	保留
16:8	VECTTHRESHOLD	指示优先级阈值对应的中断向量号。当 VIC 发现 CPU 从 VECTTHRESHOLD 对应的中断服务程序退出时，硬件自动清除中断优先级阈值有效位。
7:0	PRIOTHRESHOLD	指示中断抢占的优先级阈值。 注：CK802 中根据配置的中断数量决定的优先级个数可

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

No:

位	名称	描述
		以设[7:6],[7:5]或[7:4]位来表征优先级阈值。

图表 3-23 中断优先级阈值寄存器域描述

3.2.11. Tspend 中断使能设置寄存器 (VIC_TSPEND)

VIC_TSPEND 用于使能 tspend 中断，并且反馈 tspend 中断的使能状态。图表 3-24 描述了 VIC_TSPEND 的位分布，图表 3-25 描述了 VIC_TSPEND 的位定义。

31	1	0
Reserved		SETENA

图表 3-24 tspend 中断使能设置寄存器

位	名称	描述
0:0	SETENA	<p>设置使能，读取 tspend 中断的使能状态：</p> <p>读操作 0 对应 tspend 中断未使能。</p> <p> 1 对应 tspend 中断已使能。</p> <p>写操作 0 无效。</p> <p> 1 使能 tspend 中断。</p>

图表 3-25 tspend 中断使能设置寄存器域定义

3.2.12. Tspend 中断响应状态寄存器 (VIC_TSABR)

VIC_TSABR 用于指示 tspend 中断当前的 Active 状态，是一个供软件查询的寄存器，另外，软件可在初始化 VIC 时，将所有 tspend 中断的 Active 状态清 0。图表 3-26 描述了 VIC_TSABR 的位分布，图表 3-16 描述了 VIC_IABR 的位定义。

31	1	0
Reserved		Active

图表 3-26 tspend 中断响应状态寄存器

位	名称	描述
0	Active	<p>查询位，指示 tspend 中断是否已经被 CPU 响应但还没处理完。该寄存器只有最低位有效。</p> <p>读操作 0 没有被 CPU 响应</p> <p> 1 已经被 CPU 响应但还没处理完</p>

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).



位	名称	描述
		写操作 0 清除 tsend 中断的 Active 状态 (软件不可对该寄存器写 1, 否则会导致不可预期的错误)

图表 3-27 tsend 中断响应状态寄存器域描述

3.2.13. Tsend 中断优先级设置寄存器 (VIC_TSPR)

VIC_TSPR 提供 tsend 中断的优先级设置，tsend 中断的优先级需要设置为最低。
图表 3-28 描述了 VIC_TSPR 的位分布，图表 3-29 描述了 VIC_TSPR 的位定义。

31	8	7	0
Reserved			PRI

图表 3-28 tsend 中断优先级设置寄存器

	位	名称	描述
2 比特 4 个优先级 (中断数小于等于 32 个)	31:8	-	保留
	7:6	PRI	建议设置成最低优先级 2'b11
	5:0	-	保留
3 比特 8 个优先级 (中断数量为 64 个)	31:8	-	保留
	7:5	PRI	建议设置成最低优先级 3'b111
	5:0	-	保留
4 比特 16 个优先级 (中断数量为 96 或 128 个)	31:8	-	保留
	7:4	PRI	建议设置成最低优先级 4'b1111
	5:0	-	保留

图表 3-29 tsend 中断优先级设置寄存器域描述

3.3. 中断处理机制

矢量中断控制器支持电平中断和脉冲中断。

对于电平中断，矢量中断控制器采样到中断有效信号的高电平后设置对应中断进入等待状态，然后请求 CPU 响应。电平中断要求中断服务程序中清除外设的中断源有效信号，否则当中断退出时中断控制器会重新向 CPU 发起中断请求。外设可以根据这一特点，常置中断信号直到不再需要中断处理程序处理。

对于脉冲中断，又称为边沿中断，矢量中断控制器采样中断有效信号的上升沿，然后设

No:

置对应中断进入等待状态，随后向 CPU 发起中断请求。为了确保矢量中断控制器检测到脉冲中断，外围需要将中断信号至少保持一个 CPU 时钟周期。在 CPU 响应该脉冲中断请求前，若脉冲中断源向矢量中断控制器发起多次中断请求，矢量中断控制器只会记录一次中断请求；在 CPU 响应该脉冲中断请求后，若脉冲中断源再次向中断控制器发起请求，矢量中断控制器会再次触发对应中断进入等待状态，该处于等待状态的中断请求在上次中断退出后才能够再次被 CPU 响应。

另外，矢量中断控制器支持软件中断。软件可通过设置中断设置等待寄存器(VIC_ISPR)置高相应的中断等待状态位，触发该中断进入等待状态，向 CPU 发送中断请求。

当处理器响应中断请求后，矢量中断控制器会自动清除相应中断的等待状态位。也可以通过设置中断清除等待寄存器(VIC_ICPR)清除相应中断的等待状态位。对于电平中断，如果中断有效信号持续为高，则无法通过设置 VIC_ICPR 寄存器清除等待状态位。

3.3.1. 中断状态位

VIC 为每个中断源提供 2 比特状态位，分别为：

- Pending: 表征该中断处于等待状态，即该中断请求等待 CPU 进行响应。
 - 0: 表征中断尚未处于等待状态；
 - 1: 表征中断已经处于等待状态。
- Active: 表征该中断请求已经被 CPU 响应，但尚未处理完成。
 - 0: 表征该中断请求尚未被 CPU 响应；
 - 1: 表征该中断请求已经被 CPU 响应，但尚未处理完成。

Pending 位的设置条件：

- 1) 电平中断源，中断源有效信号置高，且 Active 为低或者 CPU 正在退出该中断服务程序；
- 2) 脉冲中断源，上升沿有效；
- 3) 软件设置 ISPR。

Pending 位的清除条件：

- 1) CPU 响应该中断请求；
- 2) 软件清除 ICPR。

Active 位的设置条件：

- 1) CPU 响应该中断请求。

Active 位清除条件：

- 1) CPU 退出该中断服务程序。

注意：对于电平中断源，由于需要中断服务程序中将电平中断源请求拉低，因此只需要在 Active 为低或者退出中断服务程序时采样中断源有效信号并设置 Pending 位；对于脉冲中断源，由于请求信号会自动拉低，因此需要实时采样脉冲中断信号的上升沿并设置 Pending 位。

3.3.2. 中断优先级

VIC 通过中断优先级设置寄存器(IPR0~IPR31)为每个中断源提供优先级设置，中断个数小于等于 32 个时，硬件可以提供 4 个优先级；中断个数小于等于 64 个，硬件可以提供 8 个优先级；中断个数为 96 个或 128 个时，硬件可以提供 16 个优先级。优先级号越低，优先级越高，具体参考 3.2.8 节。当软件没有设置优先级寄存器时，所有中断源优先级默认为最高优先级——0。

当多个中断处于 Pending 状态时，VIC 根据各个中断的优先级仲裁出优先级最高的中

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

No:

断请求提交给 CPU 处理。例如，两个中断请求 IRQ0 和 IRQ1 同时处于 Pending 状态，如果 IRQ1 的优先级号小于 IRQ0，即 IRQ1 的优先级高于 IRQ0，因此 IRQ0 先提交给 CPU 处理。

当多个 Pending 的中断拥有相同的优先级号时，根据中断号决定中断提交的顺序，中断号小的优先提交给 CPU 处理。例如，两个中断请求 IRQ0 和 IRQ1 的中断优先级相同，IRQ0 的中断源号小于 IRQ1，因此 IRQ0 优先提交给 CPU 处理。

3.3.3. 中断向量号

中断向量号，是中断请求在异常向量表的位置编号。下图给出了 CK802 的异常向量表，开始的 0~30 号向量是用作处理器内部识别的向量；31 号向量保留；从 32 开始的向量号预留给外部中断请求，且每个中断源对应一个中断向量号。

向量号	向量偏移（十六进制）	向量分配
0	000	重启异常。
1	004	未对齐访问异常。
2	008	访问错误异常。
3	00C	保留。
4	010	非法指令异常。
5	014	特权违反异常。
6	018	保留。
7	01C	断点异常。
8	020	不可恢复错误异常。
9-15	024-03C	保留。
16-19	040-04C	陷阱指令异常（TRAP #0-3）。
20-21	050-054	保留。
22	058	Tspend 中断
23-30	05C-078	保留。
31	07C	保留
32	080	IRQ0。
33	084	IRQ1。
.....
32+n	0x80+4n	IRQn

图表 3-30 中断向量号描述

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

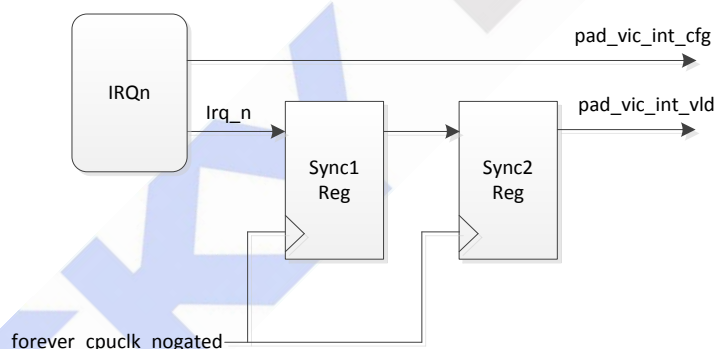
3.3.4. 中断处理过程

中断处理过程可分以下几个步骤进行：

- 中断源请求的同步：外部设备产生中断源请求，系统完成异步中断源请求到 CPUCLK 时钟域的同步操作，置高 pad_vic_int_vld；
- 中断源请求的采样：VIC 根据中断源的类型对中断源请求进行采样；当采样到有效的中断请求时，设置 Pending 状态位，触发对应中断进入等待状态；
- 中断请求发起：在所有处于等待状态的中断中，经过优先级仲裁向 CPU 发起中断请求；
- 中断响应：CPU 在指令退休时响应中断，返回中断响应信号给 VIC，同时将 PSR 和 PC 更新到 EPSR 和 EPC，并将被响应中断的中断向量号更新 PSR.VEC，清除 PSR.EE，最后取异常入口地址；VIC 根据中断响应信号清除相应中断的 Pending 状态位，并设置其 Active 状态位；
- 中断现场保存：首先保存中断控制寄存器现场{EPSR, EPC}，打开 PSR.EE 和 PSR.IE，以使能中断嵌套；随后保存通用寄存器现场；
- 中断事务：CPU 开始处理中断事务，对于电平中断，需要将中断源信号清除，否则在中断退出时会重入该中断；
- 中断现场恢复和中断退出：首先恢复通用寄存器现场；随后恢复中断控制寄存器现场{EPSR, EPC}，将 EPC 和 EPSR 恢复到 PC 和 PSR，退出中断服务程序；VIC 接收中断退出信号，清除 Active 状态位。

中断现场的保存可通过在中断服务程序的起始处执行 NIE 和 IPUSH 指令完成，中断现场的恢复和退出可通过在中断服务程序的结尾处执行 IPOP 和 NIR 指令完成。

中断源请求的同步由系统完成，VIC 内部不实现。下图给出了 pad_vic_int_vld 信号同步的一个示例，中断源请求信号 irq_n 经过两级 CPUCLK 的寄存器同步到 CPU 时钟域上。另外，中断源的类型配置信号 pad_vic_int_cfg 对于给定的中断源是一个固定值，0 表示电平中断源，1 表示脉冲中断源，因此不需要同步。



图表 3-31 中断源请求同步电路示例

3.3.5. 中断嵌套

VIC 支持中断嵌套功能，在中断处理的过程中允许更高优先级的中断抢占，从而提高中断响应实时性。当存在多级中断嵌套时，特定场景下可能需要改变被抢占中断的优先级，例如低优先级中断响应时间达到最大限制。此时软件可通过设置中断优先级阈值寄存器，提高中断抢占的优先级条件，使被抢占的低优先级中断能得到及时响应。

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

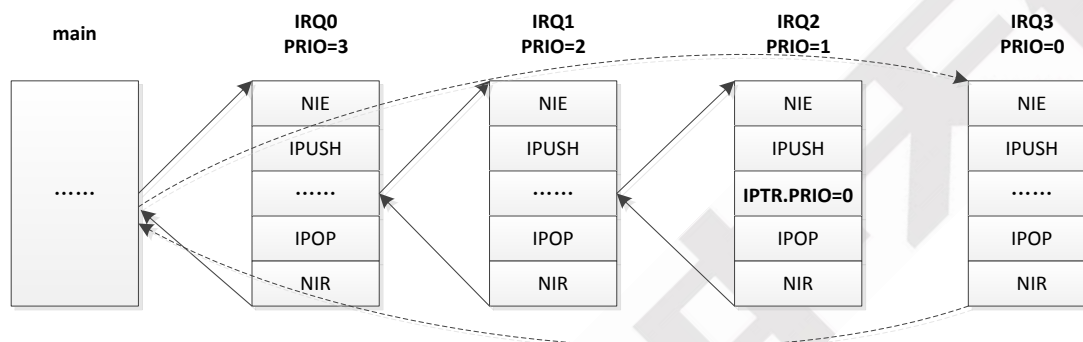
3.3.5.1. 中断嵌套优先级条件

中断抢占的优先级条件可分为两种，如下所示：

- 当中断优先级阈值未使能时，中断抢占的优先级必须高于当前 CPU 正在处理的中断的优先级；同级优先级不能进行抢占；
- 当中断优先级阈值使能时，中断抢占的优先级不仅要高于当前 CPU 正在处理的中断优先级，而且要高于中断优先级阈值寄存器设置的阈值。

VIC 支持中断优先级的动态调整，当被抢占中断的优先级需要调高或者调低时，在设置中断优先级设置寄存器的同时设置中断优先级阈值寄存器。

下图给出了中断抢占的示例。中断优先级设置为：IRQ0<IRQ1<IRQ2<IRQ3；中断源请求产生的顺序为：IRQ0>IRQ1>IRQ2>IRQ3。CPU 首先响应了 IRQ0，在 IRQ0 中断服务程序执行的过程中，来了更高优先级的 IRQ1，因此 IRQ0 被抢占，CPU 开始执行 IRQ1 的中断服务程序。同样，IRQ2 对 IRQ1 进行了抢占，并设置了中断优先级阈值寄存器（IPTR.VECTTHRESHOLD=IRQ0，IPTR.PRIOTHRESHOLD=0，IPTR.EN=1）。当 IRQ3 到来时，尽管优先级高于 IRQ2，但 IRQ3 的优先级没有高于 IPTR 中设置的优先级阈值所表征的中断，因此 IRQ3 无法抢占 IRQ2。IRQ3 等到 IRQ0 的中断服务程序执行结束硬件自动清除 IPTR.EN 后，才得到 CPU 响应。



图表 3-32 中断嵌套优先级示例

3.3.5.2. 中断嵌套时机条件

中断抢占在满足优先级的条件下，还需要判断当前中断响应的阶段。中断响应的过程如 3.3.4 所示，和嵌套相关的主要分为以下几个阶段：1) EPSR、EPC、PSR 的更新和异常入口地址的读取、2) NIE 指令、3) IPUSH 指令、4) 中断事务、5) IPOP 指令、6) NIR 指令。

为了保证中断嵌套现场的保存和恢复，CPU 在以下阶段不能被中断打断：

- 中断响应之后更新 EPSR、EPC、PSR 和获取异常入口地址的过程中；
- NIE 指令执行过程中，且包括指令退休；
- PSR.IC 位关闭，IPUSH 和 IPOP 指令执行过程中，不包括指令退休；
- NIR 指令执行过程中，不包括指令退休。

CPU 在以下阶段可安全可靠地响应新的中断：

- 正常程序的执行过程中，在中断响应之前；
- IPUSH、IPOP 指令退休时；
- PSR.IC 位打开，IPUSH、IPOP 指令执行过程中；

C-Sky Confidential

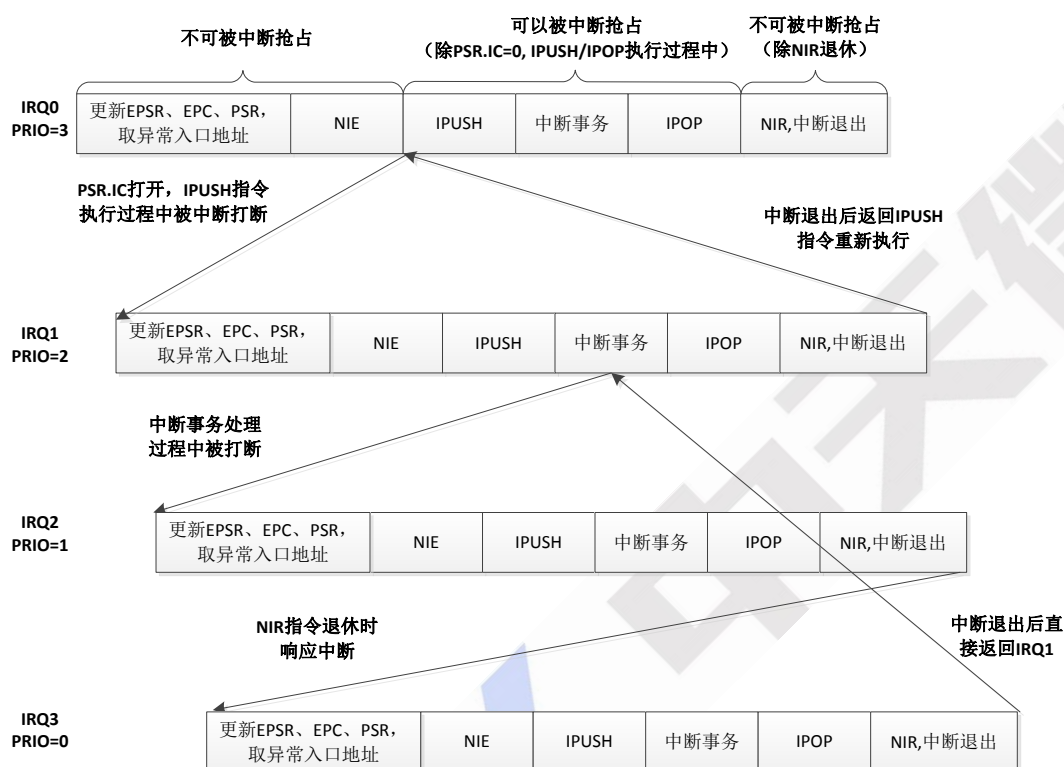
The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

No:

- NIR 指令退休时；
- 中断事务处理的过程中。

在 PSR.IC 位打开时，IPUSH 和 IPOP 指令在执行过程中可响应中断，因此中断返回时需要重新执行该指令。对于 IPUSH 或 IPOP 指令在退休时响应中断的情况，在中断退出后直接执行 IPUSH/IPOP 的下一条指令，不需要重新执行该 IPUSH/IPOP 指令。NIR 指令执行过程中不可被中断打断，但在退休时可响应中断，如果中断打在 NIR 指令上，CPU 在 NIR 指令退休时，直接将 NIR 的返回地址压入堆栈，在中断退出时返回 NIR 的目标地址。

下图给出了 IRQ0/IRQ1/IRQ2/IRQ3 中断嵌套的过程。在 IRQ0 被 CPU 响应后，产生了更高优先级的 IRQ1，当 PSR.IC 打开时，在执行到 IPUSH 指令时响应 IRQ1。IRQ2 在 IRQ1 处理中断事务时产生，因此可立即被 CPU 响应。IRQ3 在 IRQ2 执行 NIR 指令时产生，在 NIR 指令退休时响应 IRQ3。当 IRQ3 处理完退出中断服务程序时，直接返回到 IRQ1 被 IRQ2 打断的点。当 IRQ1 返回 IRQ0 时，需要重新执行 IPUSH 指令。



图表 3-33 中断嵌套时机条件示例

3.4. Tspend 中断

Tspend 中断控制器提供的,一个不占用外部中断向量号(32~159)的中断,Tspend 中断向量号为 22, tspend 中断功能和设置同其他中断基本相同, tspend 中断和其他中断的区别在于:

- (1) Tspend 中断,不需要外部设别产生中断源请求,无 pending 状态位,软件设置 VIC_TSPEND 使能时进入等待状态。
- (2) Tspend 中断和其他中断优先级相同时,不以中断号大小判断中断优先级,相同中断优先级时 Tspend 中断优先级为最低;等到没有其他同级别中断处于 pending 状态时,处理 tspend 中断。

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

3.5. 操作步骤

为保证 VIC 能够生成预期优先级的中断请求，用户需要提前设置好中断优先级并使能响应中断，步骤如下：

- 首先设置 VIC_IPR0~31，给每个中断配置合适的优先级；
- 然后设置 VIC_ISER，使能相应的中断；

另外，VIC 支持软件设置 VIC_ISPR 产生中断请求，同样在 VIC_ISPR 设置之前必须按上述步骤配置中断优先级和中断使能位。

注意：CPU 响应 VIC 触发的中断请求之前需要使能 PSR.IE 和 PSR.EE，以开启 CPU 的中断响应使能，否则 CPU 无法响应中断。

VIC 采样外部中断源产生中断低功耗唤醒之前，必须先设置 VIC_IWER，置高相应的低功耗唤醒使能位，否则无法产生低功耗唤醒请求。

注意：对于某个外部中断源请求，只需要相应中断的低功耗唤醒功能被使能（对应于 VIC_IWER），就能产生低功耗唤醒请求，而不依赖于该中断自身的中断使能（对应于 VIC_ISER）。

3.6. 接口信号

矢量中断控制器的接口信号可分为三组，分别为：

矢量中断控制器与中断源的接口：VIC 接收并采样中断源的中断请求信号；

矢量中断控制器与 TCIP 的接口：TCIP 提供一组读写 VIC 寄存器的接口，完成 VIC 相关控制寄存器的读写，如使能中断、设置中断优先级、查询当前正在处理中断等；

矢量中断控制器与处理器核 CK802 的接口：VIC 向 CK802 发起中断请求和相应的中断向量号，CK802 返回 VIC 中断响应和中断返回指示信号以及目前正在处理的中断向量号。

信号名	I/O	Reset	定义
紧耦合IP总线接口信号：			
tcipif_vic_sel	I	0	选中信号： 指示选中矢量中断控制器并进行数据传输。 1：选中 VIC 0：未选中 VIC
tcipif_vic_addr[15:0]	I	-	地址总线： 16 位地址总线（截取 32 位地址总线的低 16 位），指示访问地址。
tcipif_vic_write	I	0	读写表示信号： 指示当前 TCIP 访问是读取数据还是写数据： 1：写访问； 0：读访问。
tcipif_vic_wdata[31:0]	I	-	写数据总线：

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

No:

信号名	I/O	Reset	定义
			32 位写数据总线。
vic_tcipif_rdata[31:0]	O	-	读数据总线： 32 位读数据总线。
vic_tcipif_cmplt	O	0	传输完成指示信号： 有效时指示当前传输已完成。
处理器中断握手信号：			
vic_pad_int_b	O	1	中断请求信号： 低电平时表示进行普通中断申请。
vic_pad_intraw_b	O	1	中断唤醒请求信号： 低电平时表示进行低功耗唤醒申请
vic_pad_int_vec_b[7:0]	O	-	中断矢量序号信号： 提供 core 进行中断处理的向量号。
pad_vic_int_ack	I	0	中断响应信号： 指示 CPU 响应了当前的中断请求
pad_vic_int_exit	I	0	中断服务程序退出信号： 指示 CPU 退出中断服务程序
pad_vic_int_vec[7:0]	I	-	指示 CPU 正在处理的中断向量号
pad_vic_ack_vec[7:0]	I	-	指示 CPU 响应的中断向量号
中断源信号：			
pad_vic_int_cfg[127:0]	I	-	中断源类型配置信号： 0： 电平中断源 1： 脉冲中断源
pad_vic_int_vld[127:0]	I	0	中断有效信号： 高电平有效。
时钟信号：			
forever_cpucclk	I	-	VIC 的工作时钟： 矢量中断控制器中和低功耗唤醒无关的寄存器均工作于该时钟，低功耗模式下可以关闭该时钟信号。
forever_cpucclk_nogated	I	-	VIC 中断采样电路的时钟： 矢量中断控制器中和低功耗唤醒相关的寄存器均工作于该时钟。在低功耗模式下不可以关闭该时钟信号。
复位信号：			

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

No:

信号名	I/O	Reset	定义
cpurst_b	I	-	矢量中断控制器复位信号： 低电平时，初始化矢量中断控制器内部。矢量中断控制器采用异步复位方式。
其它信号：			
pad_yy_gate_clk_en_b	I	-	门控时钟使能信号： 只有当这个信号有效时，VIC 的内部模块的门控时钟才能有效。 不用该信号时，需要接 1。
pad_yy_test_mode	I	-	进入测试模式： 使 VIC 进入测试模式，此时 VIC 时钟为测试时钟（pad_had_jtg_tclk）。只有 VIC 进入测试模式，并且处理器输入信号 pad_yy_scan_enable 有效时，可以通过扫描链进行测试。不用该信号时，需要接 0。

图表 3-34 矢量中断控制器接口信号

当 CPU 内部没有集成矢量中断控制器时，由外部中断控制器对中断源进行仲裁，产生中断请求发送给 CPU。CPU 对外部输入的中断请求信号进行同步并传递给核内处理。另外，CPU 也需要对外部 IP 产生低功耗唤醒信号进行同步。

信号名	I/O	Reset	定义
紧耦合IP总线接口信号：			
pad_cpu_int_b	I	1	中断请求信号： 低电平时表示进行普通中断申请。
pad_cpu_intraw_b	I	1	低功耗唤醒请求信号： 低电平时表示进行低功耗唤醒申请。
pad_cpu_int_vec_b[7:0]	I	-	中断向量号： 指示当前中断请求对应中断向量号。
时钟信号：			
forever_cpuclock_nogated	I	-	VIC 同步电路时钟： 用于中断请求和低功耗唤醒信号的同步。
clk_en	I	-	系统时钟同步使能信号

图表 3-35 没有配置矢量中断控制器的接口信号

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

3.7. 中断设置示例

```

//设置 psr 中的 ee 和 ie 位, cpu 响应中断
psrset    ee, ie
//设置中断号为 32~35 的中断使能位
lrw       r1, 0x0
bseti     r1, 0x0           //设置 32 号中断的使能位
bseti     r1, 0x1           //设置 33 号中断的使能位
bseti     r1, 0x2           //设置 34 号中断的使能位
bseti     r1, 0x3           //设置 35 号中断的使能位
lrw       r2, 0xe000e100    //中断使能寄存器 ISER 对应的地址
st.w      r1, (r2, 0x0)     //使能中断号为 32~35 的 4 个中断
//其他中断的使能同上, 通过设置 ISER 寄存器的不同位使能相应的中断

//设置中断优先级寄存器 IPR0, 设置 32~35 号中断的优先级
lrw       r1, 0x0
//设置 IPR0[7:6]设置 32 号中断的优先级为 2'b11, 最低优先级
//设置 IPR0[15:14]设置 33 号中断的优先级为 2'b00, 最高优先级
//设置 IPR0[23:22]设置 34 号中断的优先级为 2'b10
//设置 IPR0[31:30]设置 35 号中断的优先级为 2'b10
bseti     r1, 0x6           //通过低[7:6]两位设置 32 号中断优先级
bseti     r1, 0x7           //设置 32 号中断的优先级为最低, IPR0[7:6]=2'b11
bclri     r1, 0x14          //通过[15:14]两位设置 33 号中断优先级
bclri     r1, 0x15          //设置 33 号中断的优先级为最高, IPR0[15:14]=2'b00
bclri     r1, 0x22          //通过[23:22]两位设置 34 号中断优先级
bseti     r1, 0x23          //设置 34 号中断的优先级为 2, IPR0[23:22]=2'b10
bclri     r1, 0x30          //通过[31:30]两位设置 35 号中断优先级
bseti     r1, 0x31          //设置 35 号中断的优先级为 2, IPR0[31:30]=2'b10
//将设置的优先级写入 IPR0
lrw       r2, 0xe000e400    //中断使能寄存器 IPR0 对应的地址
st.w      r1, (r2, 0x0)     //完成 32~35 号的中断优先级设置
//其他中断的优先级设置同上, 通过设置对应的中断优先级寄存器 IPR1~IPR7 即可。

//设置中断优先级阈值寄存器 VIC_IPTR
lrw       r1, 0x2200        //设置中断优先级阈值寄存器的中断号, IPTR[16:8]=34
bseti     r1, 0x0           //使能中断优先级阈值寄存器
//通过 IPTR[7:6]两位设置能够抢占 34 号中断的中断优先级,
//设置能够抢占 34 号中断的优先级必须优先级高于 2'b01,
//即只有优先级为 2'b00 (最高) 的中断可以抢占 34 号中断
bseti     r1, 0x6
bclri     r1, 0x7
//将设置的 34 号中断号对应的优先级阈值,使能和中断号写入 IPTR
lrw       r2, 0xe000ec04    //中断优先级阈值寄存器地址

```

C-Sky Confidential

No:

st.w r1, (r2, 0x0) //完成 34 号中断的优先级阈值设置

C-SKY 中天微

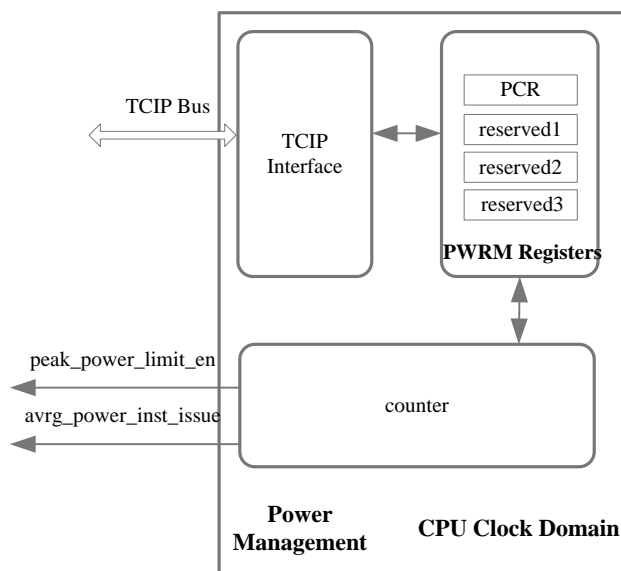
C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

4. 功耗管理模块

4.1. 简介

功耗管理模块是对处理器的峰值功耗和平均功耗进行控制的 IP。用户通过配置功耗管理模块的控制寄存器，不仅可以对处理器的峰值功耗进行限制，而且还可以约束处理器的平均功耗，用以满足 SoC 系统设计的需求。



图表 4-1 功耗管理模块系统结构图

4.2. 寄存器定义

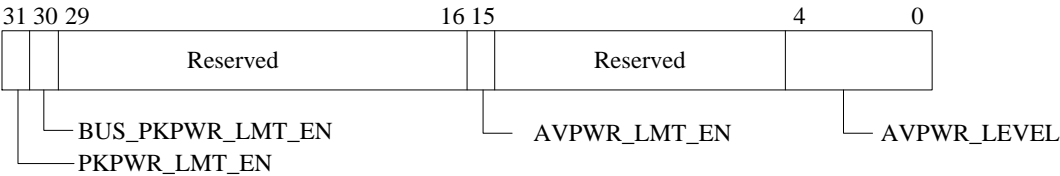
功耗管理单元每一个寄存器宽度是 32 位，寄存器地址空间为：

地址	名称	类型	初始值	描述
0xE000EF90	PCR	读/写	0x00000002	控制寄存器
0xE000EF94-0xE000EF9F	-	-	-	保留

图表 4-2 功耗管理模块寄存器定义

4.2.1. 控制寄存器（PCR）

控制寄存器由 1 比特峰值功耗控制使能位、1 比特总线峰值功耗控制使能位、1 比特平均功耗控制使能位和 5 比特的平均功耗约束强度位组成，如图表 4-3 所示。一旦控制峰值功耗或者平均功耗的使能位有效，功耗管理模块会根据控制寄存器的配置状态对处理器的峰值功耗和平均功耗进行限制管理。



图表 4-3 功耗管理模块控制寄存器

位	类型	名称	Reset值	描述
31	读/写	PKPWR_LMT_EN	0	峰值功耗限制使能位
30	读/写	BUS_PKPWR_LMT_EN	0	总线峰值功耗限制使能位
29:16	-	-	-	保留
15	读/写	AVPWR_LMT_EN	0	平均功耗限制使能位
14:5	-	-	-	保留
4:0	读/写	AVPWR_LEVEL	0x2	平均功耗限制强度标识

图表 4-4 功耗管理模块控制寄存器域描述

4.3. 操作步骤

若需要限制处理器峰值功耗，则将 PKPWR_LMT_EN 设置为 1'b1，则此时处理器的各级流水线将轮流运转以降低峰值功耗。

若需要限制总线峰值功耗，则将 BUS_PKPWR_LMT_EN 设置为 1'b1，则此时处理器任意时刻将只允许一条总线进行读写访问。

若需要限制处理器平均功耗，则将 AVPWR_LMT_EN 设置为 1'b1，并设置合适的 AVPWR_LEVEL 值，AVPWR_LEVEL 的值越大，处理器的平均功耗将越低。

4.4. 接口信号

信号名	I/O	Reset	定义
紧耦合IP总线接口信号:			
tcipif_pwrn_sel	I	0	选中信号: 指示选中功耗管理模块并进行数据传输。 1: 指示选中 0: 指示未选中
tcipif_pwrn_addr[15:0]	I	-	地址总线: 16 位地址总线 (截取 32 位地址总线的低 16 位), 指示访问地址。
tcipif_pwrn_write	I	0	读写表示信号: 指示当前 TCIP 访问是读取数据还是写数据:

C-Sky Confidential

No:

信号名	I/O	Reset	定义
			1: 指示是写访问; 0: 指示是读访问。
tcipif_pwrn_wdata[31:0]	I	-	写数据总线: 32 位写数据总线。
pwrn_tcipif_rdata[31:0]	O	-	读数据总线: 32 位读数据总线。
pwrn_tcipif_cmplt	O	0	传输完成指示信号: 有效时指示当前传输已完成。
处理器功耗控制信号:			
pwrn_cpu_avrg_power_inst_issue	O	-	允许处理器发射一条指令进入执行单元。
pwrn_cpu_peak_power_limit_en	O	-	处理器峰值功耗限制使能信号。
时钟信号:			
forever_cpucclk	I	-	提供 CPU 内核工作的时钟: 功耗管理模块的寄存器均工作于该时钟。
复位信号:			
cpurst_b	I	-	功耗管理模块复位信号: 低电平时, 初始化功耗管理模块内部。功耗管理模块采用异步复位方式。
其它信号:			
pad_yy_gate_clk_en_b	I	-	门控时钟使能信号: 只有当这个信号有效时, PWRM 的内部模块的门控时钟才能有效。 不用该信号时, 需要接 1。
pad_yy_test_mode	I	-	进入测试模式: 使 PWRM 进入测试模式, 此时 PWRM 时钟为测试时钟 (pad_had_jtg_tclk)。只有 PWRM 进入测试模式, 并且处理器输入信号 pad_yy_scan_enable 有效时, 才可以 通过扫描链进行测试。不用该信号时, 需要接 0。

图表 4-5 功耗管理模块接口信号描述

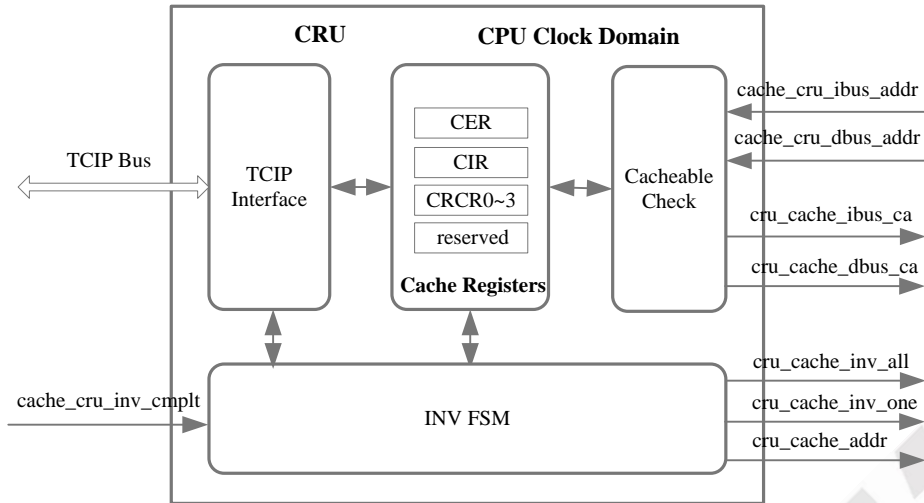
C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

5. 高速缓存控制寄存器单元

5.1. 简介

CK802 CPU 提供硬件可配置的高速缓存器 (Cache)。Cache 控制寄存器单元 (CRU) 为 Cache 提供了一组设置寄存器, 包括 Cache 的使能、缓存行的无效和高速缓存区的配置。



图表 5-1 Cache 控制寄存器单元结构图

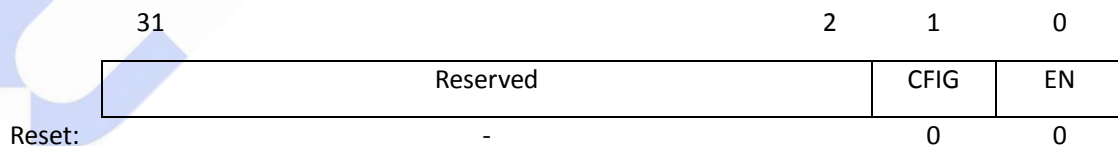
5.2. 寄存器定义

CRU 提供一组 32-bit 的寄存器, 各个寄存器地址空间如图表 5-2 所示。

地址	名称	类型	初始值	描述
0xE000F000	CER	读/写	0x00000000	高速缓存使能寄存器
0xE000F004	CIR	读/写	0x00000000	高速缓存无效寄存器
0xE000F008	CRCR0	读/写	0x00000000	0号可高缓区配置寄存器
0xE000F00C	CRCR1	读/写	0x00000000	1号可高缓区配置寄存器 (配置2个或2个以上可高缓区)
0xE000F010	CRCR2	读/写	0x00000000	2号可高缓区配置寄存器 (配置3个或3个以上可高缓区)
0xE000F014	CRCR3	读/写	0x00000000	3号可高缓区配置寄存器 (配置4个可高缓区)
0xE000F018~ 0xE000FFFF	-	-	-	保留

图表 5-2 Cache 控制寄存器定义

5.2.1. 高速缓存使能寄存器(CER)



C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

图表 5-3 高速缓存使能寄存器

CFIG-Cache 属性配置位:

- ✓ 0: 指令与数据高速缓存;
- ✓ 1: 指令高速缓存。

EN-Cache 使能位:

- ✓ 当 EN 为 0 时, 高速缓存处于关闭状态。
- ✓ 当 EN 为 1 时, 高速缓存处于工作状态。

5.2.2. 高速缓存无效寄存器(CIR)

31	4	3	2	1	0
INV_ADDR				INV_ONE	INV_ALL
Reset: 0				0	0

图表 5-4 高速缓存无效寄存器

INV_ALL-整个 Cache 无效设置位:

- ✓ 当 INV_ALL 为 1 时, 无效 Cache 中所有缓存行。

INV_ONE-单条缓存行无效设置位:

- ✓ 当 INV_ONE 为 1 时, 无效选中的缓存行。

INV_ADDR-缓存行地址:

- ✓ INV_ADDR 表征需要被无效的缓存行地址。

注意: 任何按行操作不可与任何全 cache 操作同时进行。

举例说明: 往 CIR 中写 32'b11, INV_ONE 操作将被屏蔽。

5.2.3. 可高缓区配置寄存器 0~3 (CRCR)

31	10	9	6	5	1	0
Base Address				REV	Size	EN
Reset: 0				-	0	0

图表 5-5 可高缓区配置寄存器

Base Address-可高缓区基地址:

- ✓ 可高缓区大小 4KB 到 4GB 可配置, 该域指出了可高缓区地址的高位, 例如设置页面大小为 8MB, CRCR [22:12]必须为 0, 可高速缓存区大小对基地址的要求见图表 5-6。

Size-可高缓区大小:

- ✓ 可高缓区大小可以通过公式: 可高缓区大小 = $2^{(Size+1)}$ 计算得到。因此 Size 便从 01001 到 11111, 其它一些值都会造成不可预测的结果。

图表 5-6 可高缓区大小配置和其对基址要求

C-Sky Confidential

No:

Size	高速缓存区大小	对基址的要求
00000—01010	保留	—
01011	4KB	没有要求
01100	8KB	CRCR.bit[12]=0
01101	16KB	CRCR.bit[13:12]=0
01110	32KB	CRCR.bit[14:12]=0
01111	64KB	CRCR.bit[15:12]=0
10000	128KB	CRCR.bit[16:12]=0
10001	256KB	CRCR.bit[17:12]=0
10010	512KB	CRCR.bit[18:12]=0
10011	1MB	CRCR.bit[19:12]=0
10100	2MB	CRCR.bit[20:12]=0
10101	4MB	CRCR.bit[21:12]=0
10110	8MB	CRCR.bit[22:12]=0
10111	16MB	CRCR.bit[23:12]=0
11000	32MB	CRCR.bit[24:12]=0
11001	64MB	CRCR.bit[25:12]=0
11010	128MB	CRCR.bit[26:12]=0
11011	256MB	CRCR.bit[27:12]=0
11100	512MB	CRCR.bit[28:12]=0
11101	1GB	CRCR.bit[29:12]=0
11110	2GB	CRCR.bit[30:12]=0
11111	4GB	CRCR.bit[31:12]=0

EN-可高缓区有效位:

- ✓ 当 EN 为 0 时, 可高缓区无效;
- ✓ 当 EN 为 1 时, 可高缓区有效。

5.2.4. 寄存器使用说明

5.2.4.1. CER 使用说明

CER 寄存器提供给用户以下功能:

- 使能 Cache
- 配置 Cache 属性

通过对 CER[0]置 1 可以使能 Cache, 置 0 可以关闭 Cache。通过设置 CER[1]可以配置 Cache 的属性, CER[1]置 1 为纯指令 Cache, CER[1]置 0 为指令、数据 Cache。

需要说明的是, CK802 中, Cache 只映射指令总线对应的地址空间, 即在双总线的情况下, Cache 只映射 I-AHB Lite 总线对应的内存空间。若 Cache 配置为指令 Cache, 则只有命中指令总线地址空间的取指操作才通过 CRCR 判定是否可高缓; 若 Cache 配置为数据 Cache, 则除上述取指请求外, 命中指令总线地址空间的 LOAD 操作也会通过 CRCR 判定是否可高

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

缓；另外，所有 STORE 操作均不可高缓。

5.2.4.2. CIR 使用说明

CIR 寄存器提供给用户以下功能：

- 无效单条缓存行
- 无效整个 Cache

在设置无效单条缓存行 CIR[1] 的同时需要指定该缓存行的地址 CIR[31:4]，否则将无效 CIR[31:4] 原先指向的缓存行。当 CIR[1]和 CIR[0] 同时置高时，只响应整个 Cache 的无效操作，这是因为单条缓存行的无效包含在整个 Cache 的无效里。在 Cache 无效操作执行完成后，硬件自动将 CIR[1]、CCR[0] 清除，通过软件查询这两位，读到的值恒为 0。使能 Cache 的同时或者之前需要无效整个 Cache，否则有可能导致不可预测的错误。

5.2.4.3. CRCR 使用说明

CRCR 寄存器定义了可高缓区基地址、可高缓区大小和可高缓区有效位供用户配置内存地址的可高缓属性。CRCR 寄存器的个数与用户硬件配置的可高缓区数目一致，最多可配置 4 个可高缓区。当硬件配置了多个可高缓区时，各个高缓区地址不可重叠，因此 CRCR 寄存器的配置不存在优先级，用户可根据需求灵活配置。一旦可高缓区地址配置重叠将导致不可预测的错误。用户在配置 CRCR 寄存器后，需要打开 Cache 使能位 CCR[0]，否则尽管 CRCR 寄存器指定了可高缓的内存区，它的访问属性也是不可高缓的。

5.3. 操作步骤

Cache 在 CPU reset 后默认是关闭的，因此在使用 Cache 前需要的准备工作如下：

- 首先设置 CRCR0~3，配置可高缓区；
- 然后设置 CIR[0]，无效整个 Cache；
- 最后设置 CER[0]，使能 Cache，同时设置 CER[1]，配置 Cache 的属性。

5.4. 接口信号

信号名	I/O	Reset	定义
紧耦合IP总线接口信号：			
tcipif_cru_sel	I	0	选中信号： 指示选中功耗管理模块并进行数据传输。 1：指示选中 0：指示未选中
tcipif_cru_addr[15:0]	I	-	地址总线： 16 位地址总线（截取 32 位地址总线的低 16 位），指示访问地址。

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

No:

信号名	I/O	Reset	定义
tcipif_cru_write	I	0	读写表示信号： 指示当前 TCIP 访问是读取数据还是写数据： 1：指示是写访问； 0：指示是读访问。
tcipif_cru_wdata[31:0]	I	-	写数据： 32 位写数据总线。
cru_tcipif_rdata[31:0]	O	-	读数据： 32 位读数据总线。
cru_tcipif_cmplt	O	0	传输完成指示信号： 有效时指示当前传输已完成。
Cache 控制信号：			
cache_cru_dbus_addr[31:0]	I	-	Dbus 访问地址信号： 指示当前 Cache 收到的 dbus 请求地址
cache_cru_ibus_addr[31:0]	I	-	Ibus 访问地址信号： 指示当前 Cache 收到的 ibus 请求地址
cru_cache_dbus_ca	O	-	Dbus 访问可高缓信号： 指示当前 dbus 访问地址是否可高缓： 1：可高缓 0：不可高缓
cru_cache_ibus_ca	O	-	Ibus 访问可高缓信号： 指示当前 ibus 访问地址是否可高缓： 1：可高缓 0：不可高缓
cache_cru_inv_cmplt	I	0	Cache 无效操作完成信号。
cru_cache_inv_all	O	0	无效整个 Cache 缓存行信号。
cru_cache_inv_one	O	0	无效单个 Cache 缓存行信号。
cru_cache_addr[27:0]	O	-	缓存行的地址信号： 指示需要被无效的缓存行的地址。
时钟信号：			

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

No:

信号名	I/O	Reset	定义
forever_cpucclk	I	-	提供 CPU 内核工作的时钟： Cache 的控制寄存器均工作于该时钟。
复位信号：			
cpurst_b	I	-	Cache 控制寄存器单元复位信号： 低电平时，初始化 Cache 控制寄存器单元内部；采用异步复位方式。
其它信号：			
pad_yy_gate_clk_en_b	I	-	门控时钟使能信号： 只有当这个信号有效时，CRU 的内部模块的门控时钟才能有效。 不用该信号时，需要接 1。
pad_yy_test_mode	I	-	进入测试模式： 使 CRU 进入测试模式，此时 CRU 时钟为测试时钟（pad_had_jtg_tclk）。只有 CRU 进入测试模式，并且处理器输入信号 pad_yy_scan_enable 有效时，才可以通过扫描链进行测试。不用该信号时，需要接 0。

图表 5-7 Cache 控制寄存器单元接口信号描述

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

6. 调试寄存器映射

6.1. 简介

HAD 内部的寄存器，除了通过 JTAG 接口进行读写，也可以通过 CPU 直接进行读写，以提升调试效率。为了实现 CPU 直接读写 HAD 寄存器功能，CK802 中将 HAD 内部的处理器空间，映射到了 TCIP 接口上。若处理器访问的 32 位地址空间的高 20 位为 0xE0011，则表示处理器对 HAD 内部寄存器的读写访问。

31	12	11	7	6	2	1	0
HAD IMG BASE		BANK SEL		INDEX			

图表 6-1 HAD 寄存器 TCIP 映射地址说明

如上图，处理器地址 ADDR[31:12]用于判定是否命中 TCIP 中的 HAD 寄存器映射区域，若命中，则 ADDR[11:7]用于选择 HAD 寄存器的 BANK，类似 BSEL 寄存器的作用，ADDR[6:2]用于索引制定 BANK 中的具体寄存器，类似 HACR 中 RS[4:0]的作用。

通过 TCIP 访问 HAD 寄存器，只需要一次 load 或者 store 操作即可，避免了通过 JTAG 操作时对 HACR，BSEL 等寄存器的多次读写操作，提升了调试的效率。

需要注意的是，TCIP 只有超级用户模式有访问权限。

6.2. 寄存器定义

能够通过 TCIP 访问的 HAD 寄存器地址如图表 6-2 所示。

图表 6-2 HAD 寄存器表

HAD 寄存器 TCIP 地址	信号名	类型	初始值	描述
0xE0011010	MBCA	读/写	-	内存硬断点计数寄存器 A
0xE0011014	MBCB	读/写	-	内存硬断点计数寄存器 B
0xE001101C	BABA	读/写	-	内存硬断点基地址寄存器 A
0xE0011020	BABB	读/写	-	内存硬断点基地址寄存器 B
0xE0011024	BAMA	读/写	0	内存硬断点地址掩码寄存器 A
0xE0011028	BAMB	读/写	0	内存硬断点地址掩码寄存器 B
0xE0011034	HCR	读/写	0	HAD 控制寄存器 (HAD Control Register)
0xE0011038	HSR	读	0	HAD 状态寄存器 (HAD Status Register)
0xE0011054	CSR	读/写	0	控制和状态寄存器 (Control and Status Register)
0xE0011080	BABC	读/写	-	内存硬断点基地址寄存器 C
0xE0011084	BAMC	读/写	-	内存硬断点地址掩码寄存器 C
0xE0011088	BABD	读/写	-	内存硬断点基地址寄存器 D
0xE001108C	BAMD	读/写	-	内存硬断点地址掩码寄存器 D
0xE0011090	BABE	读/写	-	内存硬断点基地址寄存器 E

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

No:

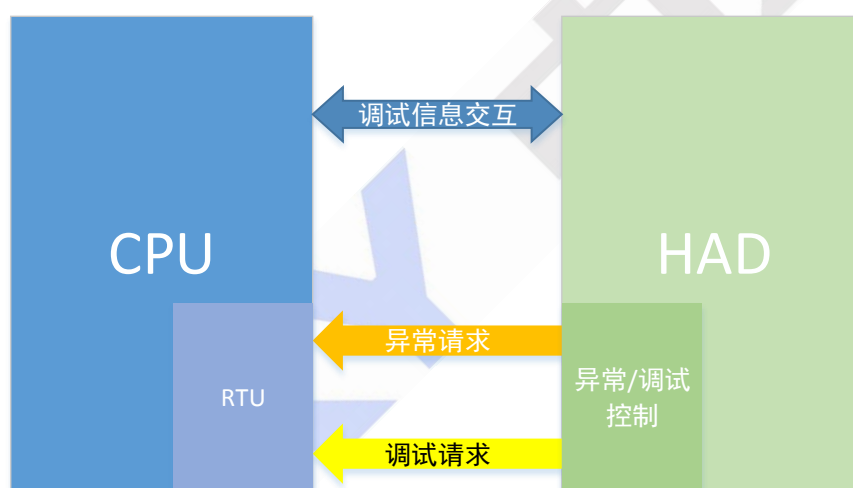
0xE0011094	BAME	读/写	-	内存硬断点地址掩码寄存器 E
0xE0011098	BABF	读/写	-	内存硬断点基地址寄存器 F
0xE001109C	BAMF	读/写	-	内存硬断点地址掩码寄存器 F
0xE00110A0	BABG	读/写	-	内存硬断点基地址寄存器 G
0xE00110A4	BAMG	读/写	-	内存硬断点地址掩码寄存器 G
0xE00110A8	BABH	读/写	-	内存硬断点基地址寄存器 H
0xE00110AC	BAMH	读/写	-	内存硬断点地址掩码寄存器 H
0xE00110B0	BABI	读/写	-	内存硬断点基地址寄存器 I
0xE00110B4	BAMI	读/写	-	内存硬断点地址掩码寄存器 I
0xE00110EC	MBIR	读	0	内存硬断点索引寄存器
0xE00110E0	DACSR	读/写	-	直接访问总线控制和状态寄存器
0xE00110E4	DATR	读/写	-	直接访问总线传输寄存器
0xE00110E8	DARWR	读/写	-	直接访问总线读写寄存器

6.3. 地址观测异常

6.3.1. 简介

为了满足操作系统层面对应用的调试需求，并且脱离 JTAG 的硬件限制，更自由的使用硬件调试功能，CK802 中 HAD 控制状态寄存器 CSR 的第 6 位 MBEE 位为 1 时，内存硬断点的发生将产生地址观测异常。

地址观测异常具有最高异常优先级，满足地址观测异常响应条件进入异常后，用户可以通过 TCIP 接口配置 HAD 寄存器，提高调试效率。



图表 6-3 CPU 和 HAD 交互

6.3.2. 地址观测异常相关寄存器

地址观测异常寄存器包括了断点相关的寄存器。在此基础上，为了方便调试人员进行定位，新增加了硬件断点触发指示寄存器 MBIR，该寄存器里记录了最近一次触发硬件断点的硬件断点号。用户读取该寄存器可以查询到最近一次触发地址观测异常的硬件断点索引号。读取方式和读取其它 HAD 寄存器一样。MBIR 的位定义如图表 6-4 所示：

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).

No:

31	4	3	2	1	0
preserved			INDEX		

图表 6-4 MBIR 寄存器

INDEX - 最近触发硬件断点索引号

该区域用于存储最近触发硬件断点的硬件断点编号，为硬件置位软件只读寄存器。

0: reset 值

1: 最近一次触发硬件断点请求时 BKPTA

2: 最近一次触发硬件断点请求时 BKPTB

如果 CPU 内配置了 BKPT9，则：

3: 最近一次触发硬件断点请求时 BKPTC

4: 最近一次触发硬件断点请求时 BKPTD

5: 最近一次触发硬件断点请求时 BKPTE

6: 最近一次触发硬件断点请求时 BKPTF

7: 最近一次触发硬件断点请求时 BKPTG

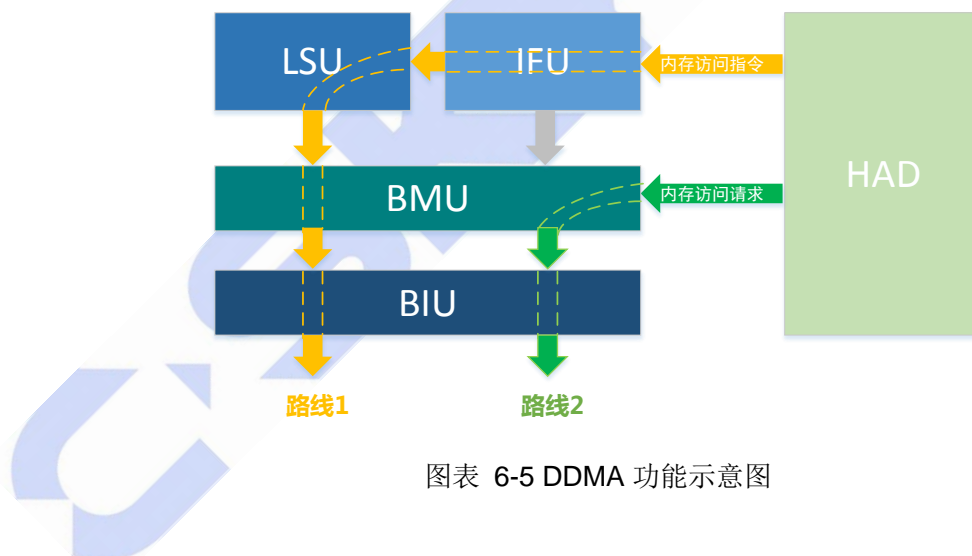
8: 最近一次触发硬件断点请求时 BKPTH

9: 最近一次触发硬件断点请求时 BKPTI

10~15: 无意义

6.4. HAD 直接访问内存功能 (DDMA)**6.4.1. 简介**

为了提高调试效率，CK802 在传统的调试单元访问内存方式的基础上，支持 HAD 直接访问内存。如下图所示，路线 1 为调试单元访问内存的传统方式，首先需要驱动一条内存访问指令，如 LD、ST 等，由指令功能决定对内存的访问方式。如路线 2 所示，HAD 直接访问内存功能将绕过 CPU 中的取指单元和执行单元，直接向总线发出传输请求，拥有更高的内存访问速度，并且降低了 CPU 状态对 HAD 内存访问操作的影响。此功能的另外一个特点是允许 HAD 在非调试模式下执行内存访问，HAD 寄存器 HCR 的第 29 位 DDAE 打开时，HAD 能发起对内存的直接访问请求。



图表 6-5 DDMA 功能示意图

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).



6.4.2. DDMA 寄存器定义

1) **DACSR**: 决定每次传输的类型、size 以及 prot。在每次传输开始前都需要将此寄存器配置完成。

31	30	29	28	12	11	8	7	6	4	3	2	0
TRANS BUSY	TRANS ERROR	DATA PHASE	0		TRANS PROT		0	TRANS SIZE		0	TRANS MODE	

图表 6-6 DACSR 寄存器

TRANS MODE[2:0]	R/W	3'b000: single, 单次传输 单次传输 3'b001: increase, 地址递增传输。 Size 为 byte: 传输完成后 DATR 加 1。 Size 为 half word: 传输完成后 DATR 加 2。 Size 为 word: 传输完成后 DATR 加 4。 3'b010~3'b111: preserved
TRANS SIZE[2:0]	R/W	3'b000: byte 3'b001: half word 3'b010: word 3'b011~3'b111: preserved
TRANS PROT[3:0]	R/W	指令/数据位: 可编辑 cacheable: 可编辑 超级用户/普通用户: 可编辑 可信/非可信: 在最高调试权限时可编辑, 在其他权限固定为 0 Prot 的内容: cacheable; secu; super; 指令数据位->[3:0]
TRANS BUSY	R	1: 前次传输未完成 0: 传输为空
TRANS ERROR	R	1: 前次传输发生异常 0: 前次传输正常完成
DATA PHASE	R/W	读传输数据阶段位: 1: 读 DARWR 将不会发起新的总线读传输。 0: 读 DARWR 将会发起新的总线读传输。 硬件置 1: 当传输类型为 single 时, 由本调试单元发起的一次总线读传输正常完成后, 此位将被硬件置为 1。 硬件清 0: 读 DARWR 后会被硬件清 0。

图表 6-7 DACSR 表项说明

2) **DATR**: 直接内存访问目的地址寄存器

31	0
TRANS ADDRESS	

图表 6-8 DATR 寄存器

该寄存器由 32 位组成, 记录了访问地址, 下一次传输地址即为此数值。如果是递增传输, 每一次传输完成地址寄存器会自动更新。递增方式参照 TRANS MODE 中的描

C-Sky Confidential

述。

3) **DARWR**: 读写寄存器

31	0
READ/WRITE DATA	

图表 6-9 DARWR 寄存器

此寄存器有两个功能:

- 1) 通过读或写这组寄存器来驱动一次总线的读或写传输, 详细情况见图表 6-10。
- 2) 通过读此寄存器, 来得到上一次读传输完成后的数据。

对 DARWR 的操作	DATA PHASE	TRANS MODE	发起的传输
读	0	Single	读
读	0	Increase	读
读	1	- (no care)	无
写	-	-	写

图表 6-10 DARWR 驱动总线传输

对于写操作来说, 此寄存器只有单一的功能: 驱动总线发起一次写传输操作。写传输地址为 DATR 的数值, 写传输的数据为写入此寄存器的数值。

对于读操作来说, 一次读有可能触发一次总线读操作, 并且把 DARWR 的数据读出。如果不想发起总线传输, 需要把 DATA PHASE 位置起。Single 的读传输正常完成后, DATA PHASE 位将自动被置 1。当 DATA PHASE 位为 1 时, 下一次读 DARWR 寄存器时, 触发总线读传输的能力将被屏蔽, 同时 DATA PHASE 位被清为 0。

6.4.3. DDMA 操作示例

1. 对地址 32'hABCD 发起单次读操作, size: word, prot: 4'b0.
 - 1) 配置 HCR 寄存器, 置 DDAE 位为 1;
 - 2) Set DATR = 32'hABCD;
 - 3) Set DACSR = 32'h20; (传输类型: single)
 - 4) Read DARWR; (总线发起读传输, 完成后 read phase 位置 1)
 - 5) Read DARWR。 (读回总线返回数据, read phase 位置 0)

在第 3 和第 4 步之间, 可以增加一步 read DACSR, 通过 busy 和 error 两位用来检查之前的传输是否已经正常完成。
2. 对地址 32'hABCD 发起连续 5 次 increase 读操作, size: word, prot: 4'b0.
 - 1) 配置 HCR 寄存器, 置 DDAE 位为 1;
 - 2) Set DATR = 32'hABCD;
 - 3) Set DACSR = 32'h21; (传输类型: increase)
 - 4) Read DARWR; (第一次传输开始)
 - 5) Read DARWR; (第二次传输开始, 第一个数据得到)
 - 6) Read DARWR; (第三次传输开始, 第二个数据得到)
 - 7) Read DARWR; (第四次传输开始, 第三个数据得到)
 - 8) Read DARWR; (第五次传输开始, 第四个数据得到)
 - 9) Read DARWR。 (第五个数据得到, 多发起一次读传输, 如要避免, 需要在第 8 步之前, 设置 DACSR 中的 DATA PHASE 位为 1)

C-Sky Confidential

No:

3. 对地址 32'hABCD 发起连续 5 次 increase 写操作, size: word, prot: 4'b0.
 - 1) 配置 HCR 寄存器, 置 DDAE 位为 1;
 - 2) Set DATR = 32'hABCD;
 - 3) Set DACSR = 32'h21; (传输类型: increase)
 - 4) write DARWR; (第一次写传输开始)
 - 5) write DARWR; (第二次写传输开始)
 - 6) write DARWR; (第三次写传输开始)
 - 7) write DARWR; (第四次写传输开始)
 - 8) write DARWR。 (第五次写传输开始)

C-SKY 中天微

C-Sky Confidential

The information contained herein is confidential and proprietary and is not to be disclosed outside of Hangzhou C-Sky Microsystems except under a Non-Disclosure Agreement (NDA).