

Altera® SoC FPGA 器件中的硬核处理器系统 (HPS) 包括一个独立的、全功能的 ARM® Cortex™-A9 MPCore™ 单或双核 32 位应用处理器。Cortex-A9 MPU 子系统包括一个 Cortex-A9 MPCore、一个第 2 层 (L2) 缓存、一个加速器一致性端口 (ACP) ID 映射器和调试模块。

Cortex-A9 MPU 子系统的功能

Altera Cortex-A9 MPU 子系统提供以下功能：

- 一个或两个 Cortex-A9 处理器
- 中断控制器
- 每个处理器的专用间隔和看门狗计时器
- 全局计时器
- TrustZone® 系统安全扩展 (TrustZone® system security extensions)
- 对称多处理 (SMP) 和非对称多处理 (AMP) 模式
- 调试模块

© 2012 Altera Corporation. All rights reserved. ALTERA, ARRIA, CYCLONE, HARDCOPY, MAX, MEGACORE, NIOS, QUARTUS and STRATIX words and logos are trademarks of Altera Corporation and registered in the U.S. Patent and Trademark Office and in other countries. All other words and logos identified as trademarks or service marks are the property of their respective holders as described at www.altera.com/common/legal.html. Altera warrants performance of its semiconductor products to current specifications in accordance with Altera's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Altera assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Altera. Altera customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.

Portions © 2011 ARM Limited. Used with permission. All rights reserved. ARM, the ARM Powered logo, AMBA, Jazelle, StrongARM, Thumb, and TrustZone are registered trademarks of ARM Limited. The ARM logo, Angel, ARMulator, AHB, APB, ASB, ATB, AXI, CoreSight, Cortex, EmbeddedICE, ModelGen, MPCore, Multi-ICE, NEON, PrimeCell, ARM7TDMI, ARM7TDMI-S, ARM9TDMI, ARM9E-S, ARM966E-S, ETM7, ETM9, TDMI and STRONG are trademarks of ARM Limited. All other products or services mentioned herein may be trademarks of their respective owners. Neither the whole nor any part of the information contained in, or the product described in, this document may be adapted or reproduced in any material form except with the prior written permission of the copyright holder. The product described in this document is subject to continuous developments and improvements. All particulars of the product and its use contained in this document are given by ARM in good faith. However, all warranties implied or expressed, including but not limited to implied warranties of merchantability, or fitness for purpose, are excluded. This document is intended only to assist the reader in the use of the product. ARM Limited shall not be liable for any loss or damage arising from the use of any information in this document, or any error or omission in such information, or any incorrect use of the product. Where the term ARM is used it means "ARM or any of its subsidiaries as appropriate". This document is Non-Confidential. The right to use, copy and disclose this document may be subject to license restrictions in accordance with the terms of the agreement entered into by ARM and the party that ARM delivered this document to. The information in this document is final, that is for a developed product.



Cortex-A9 MPU 子系统结构图和系统集成

图 6-1 显示了 HPS 背景下带有 L2 缓存的双核 MPU 子系统。L2 缓存可以访问第 3 层 (L3) 互联架构或 SDRAM。

图 6-1. 具有 L3 互联的 Cortex-A9 MPU 子系统

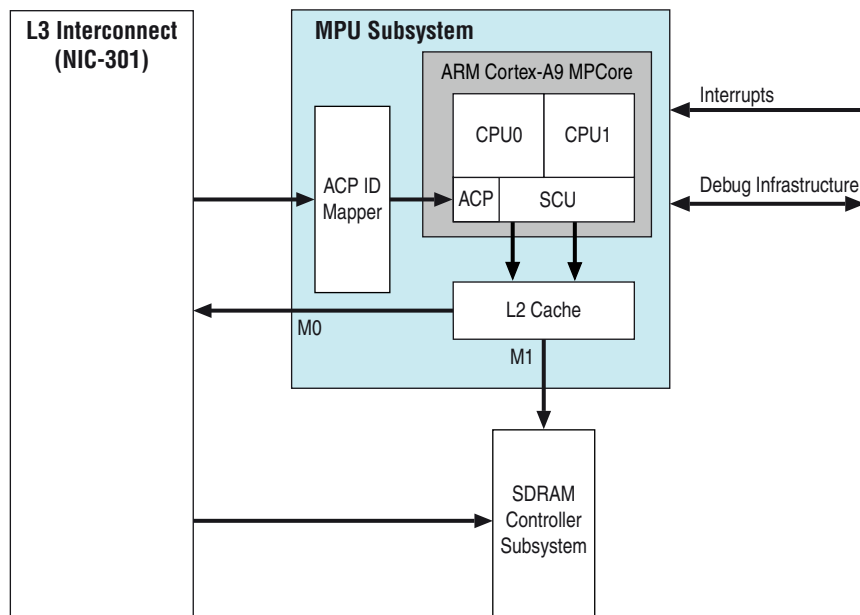
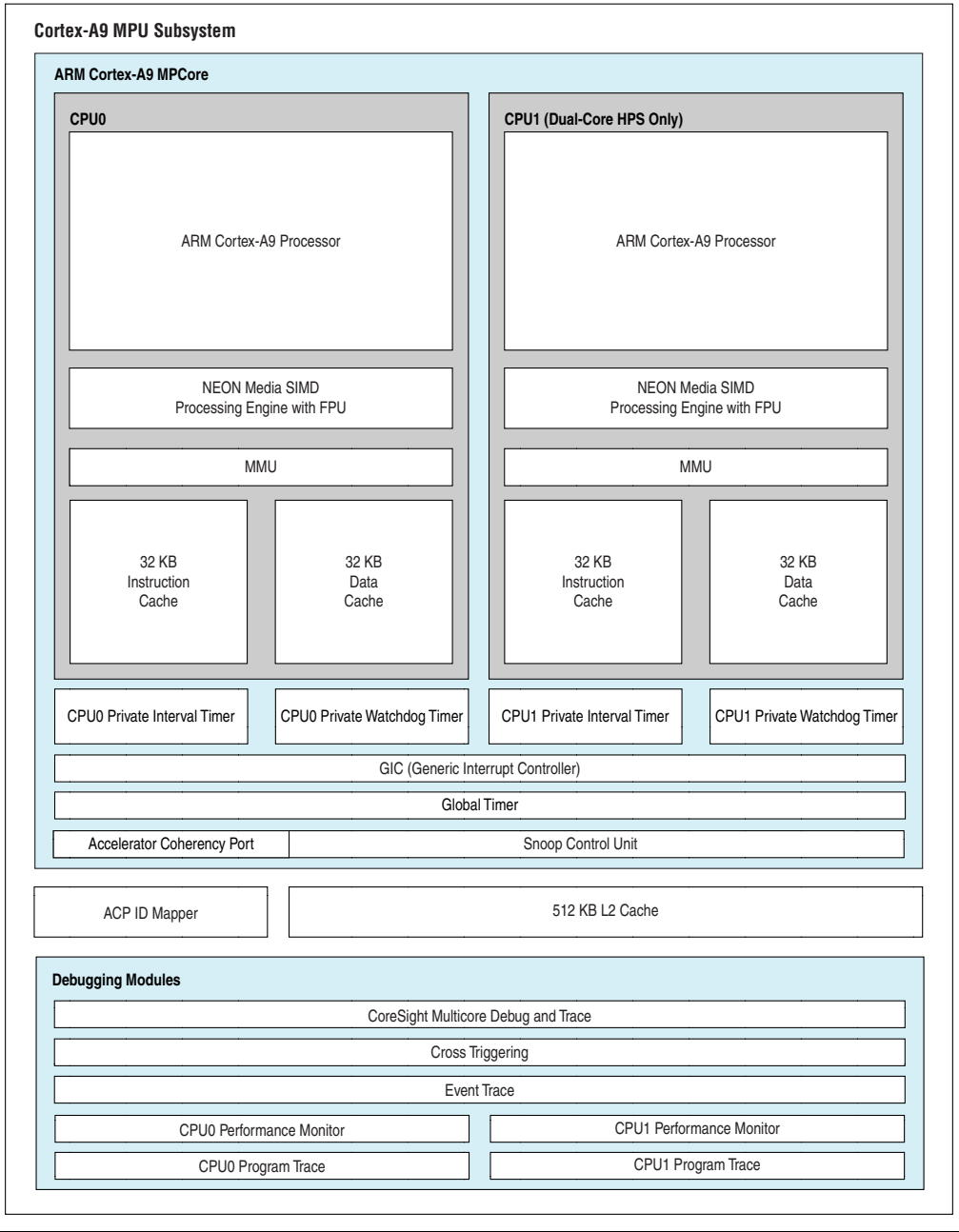


图 6-2 显示了 Altera Cortex-A9 MPU 子系统的结构图。

图 6-2. Cortex-A9 MPU 子系统内部组件



Cortex-A9 MPU 子系统组件

Altera Cortex-A9 MPU 子系统包括下列硬件模块：

- ARM Cortex-A9 MPCore
- ARM L2C-310 L2 缓存控制器

- ACP ID 映射器
- 调试和跟踪功能

该部分介绍 Cortex-A9 MPU 子系统的组件。

Cortex-A9 MPCore

MPU 子系统包括一个独立的、全功能的 ARM Cortex-A9 MPCore 单或双核 32 位应用处理器。该处理器，如同其它的 HPS 主模块，可以通过使用 HPS-to-FPGA 桥接访问 FPGA 架构中的 IP。

功能说明

ARM Cortex-A9 MPCore 包含下列模块：

- 在 SMP 或 AMP 模式中操作的一个或两个 Cortex-A9 修订版本 r3p0 处理器
- Snoop control unit（窥测掌控单元，SCU）
- 每个处理器内核的专用间隔计时器 (Private interval timer)
- 每个处理器内核的专用看门狗计时器 (Private watchdog timer)
- 全局计时器
- 中断控制器

来自 Altera Cortex-A9 MPU 子系统的每个传输都可以被标记为安全的或不安全的。

实现详细信息

表 6-1 显示了 Altera Cortex-A9 MPCore 的参数设置。

表 6-1. Cortex-A9 MPCore 处理器配置

功能	选项
Cortex-A9 处理器	1 或 2
每 Cortex-A9 处理器的指令缓存容量	32 KB
每 Cortex-A9 处理器的数据缓存容量	32 KB
每 Cortex-A9 处理器的 TLB 容量	128 项
每 Cortex-A9 处理器的具有 NEON™ 技术的媒体处理引擎 (1)	包括
每 Cortex-A9 处理器的预加载引擎	包括
每 Cortex-A9 处理器的预加载引擎 FIFO 中的项的数量	16
每 Cortex-A9 处理器的 Jazelle DBX 扩展	完全
每 Cortex-A9 处理器的 PTM 接口	包括
支持奇偶校验错误检测 (2)	包括
ARM_BIST	包括
主端口	两个
加速器一致性端口	包括
表 6-1 注释： (1) 包括对浮点操作的支持。 (2) 要了解关于奇偶校验错误方案和奇偶校验错误信号说明的更多信息，在 ARM 网站 (infocenter.arm.com)，请参考 Cortex-A9 技术参考手册，修订版本 r3p0。	

 要了解关于 Cortex-A9 MPCore 可配置选项的更多信息，在 ARM 网站 (infocenter.arm.com)，请参考 *Cortex-A9 MPCore 技术参考手册*，修订版本 r3p0 的 *Introduction* 章节。

Cortex-A9 处理器


每个 Cortex-A9 处理器都包含以下硬件模块：

- 具有矢量浮点 (VFP)v3 双精度浮点单元的 ARM NEON™ 单指令、多数据 (SIMD) 协处理器，以实现媒体和信号处理加速
 - 单和双精度 IEEE-754 浮点数学支持
 - 整数和多项式数学支持
- 具有奇偶校验检查的第 1 层 (L1) 缓存
 - 32 KB 四路组相联 (four-way set-associative) 指令缓存
 - 32 KB 四路组相联 (four-way set-associative) 数据缓存
- CoreSight™ Program Trace Macrocell (程序跟踪宏单元, PTM) 支持指令跟踪

每个 Cortex-A9 处理器都支持以下功能：

- 带有高级分支预测的双发射超标量流水线
- 乱序 (OoO) 分配和推测指令执行
- 基于 Dhrystone 2.1 基准 (benchmark)，每 MHz 每秒 2.5 百万个指令 (MIPS)
- 128-项 translation lookaside buffer (转换查找缓冲器, TLB)
- TrustZone 安全扩展
- 可配置数据 endianness
- 字节码动态编译器支持的 Jazelle® DBX 扩展
- Cortex-A9 处理器体系结构支持以下指令集：
 - ARMv7-A 性能优化的指令集
 - 存储器优化的 Thumb®-2 混合指令集
 - 提高能量效率
 - 31% 的较小内存占用
 - 比原始 Thumb 指令集快 38%
 - Thumb 指集 — 支持传统应用
- Altera HPS 中的每个处理器内核包含一个存储器管理单元 (MMU) 来支持通用现代操作系统的存储器管理要求。

Cortex-A9 处理器是指定的 CPU0 和 CPU1。

 ARM Cortex-A9 系列处理器，修订版本 r3p0 的详细说明文档在 ARM 网站 (infocenter.arm.com) 上。

交互式调试功能

每个 Cortex-A9 处理器都有一个内置调试性能，包括以下功能：

- 6 个硬件断点，其中 2 个具有 Context ID 比较功能
- 4 个监视点

交互式调试功能可以由外部 JTAG 工具或基于处理器的监控代码控制。



要了解关于交互式调试系统的更多信息，在 ARM 网站 (infocenter.arm.com)，请参考 *Cortex-A9 技术参考手册*，修订版本 r3p0 的 *Debug* 章节。

L1 缓存

与相关处理器紧密耦合的缓冲存储器被称为第 1 层，或 L1 缓存。每个 Cortex-A9 处理器具有两个独立的 32 KB L1 缓存 — 一个用于指令，另一个用于数据 — 支持同时进行的指令预取和数据访问。

每个 L1 缓存是四路组关联，每行具有 32 个字节，并且支持奇偶校验检查。

预加载引擎

预加载引擎 (PLE) 是一个硬核模块，使 L2 缓存能够预加载选择的存储器区域。PLE 通知 L2 缓存什么时候需要一个缓存行，使处理器数据主端口开始预取数据。处理器数据主端口不完成预取或将数据返回到处理器。然而，L2 缓存可以继续加载缓存行。数据仅被加载到 L2 缓存，而不加载到 L1 缓存或处理器寄存器。

预加载功能性由软件控制。必须编程以下 PLE 控制参数：

- 包括以下编程的参数：
 - 基地址
 - 步长
 - 模块数
- 一个有效位
- 使用一个 NS(非安全)状态位，缓存存储器的 TrustZone 存储器保护
- 一个转换表基 (TTB) 地址
- 地址空间标识符 (ASID) 值



要了解关于 PLE 的更多信息，在 ARM 网站 (infocenter.arm.com)，请参考 *Cortex-A9 技术参考手册*，修订版本 r3p0 的 *Preload Engine* 章节。

浮点单元

每个 ARM Cortex-A9 处理器包含对 IEEE-754 浮点操作的完全支持。浮点单元 (FPU) 完全支持下列操作的半倍、单倍和双倍精度变化：

- 加
- 减
- 乘
- 除
- 乘法及累加计算 (MAC)

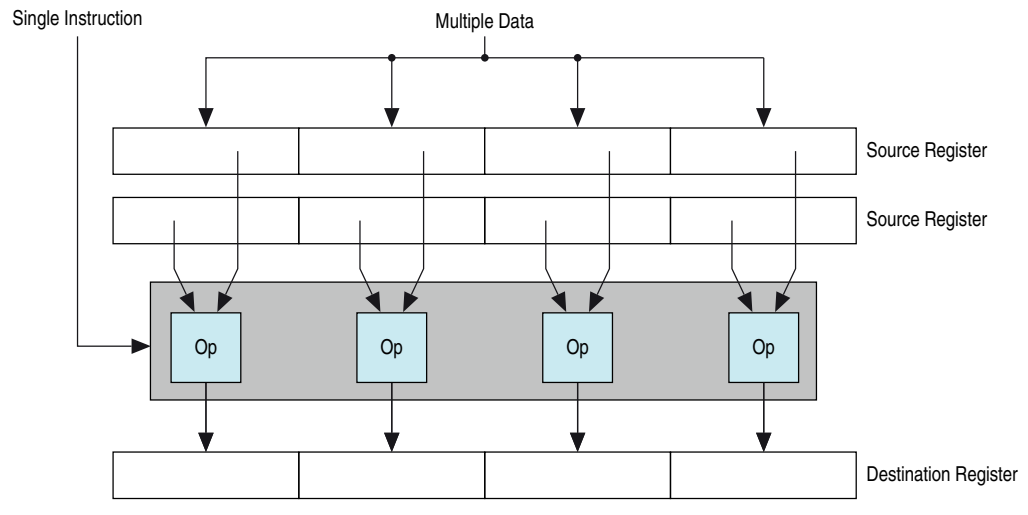
■ 平方根

FPU 也在浮点数据格式和整数之间转换，包括高层语言要求的取整到零的特殊操作。

NEON 多媒体处理引擎

NEON 多媒体处理引擎 (MPE) 对媒体和信号处理应用提供硬件加速。每个 ARM Cortex-A9 处理器都包含一个支持 SIMD 处理的 ARM NEON MPE，如图 6-3 所示。NEON 处理引擎加速多媒体和信号处理算法，例如视频编码和解码、2-D 和 3-D 图形、音频和语音处理、图像处理、电话学和声音合成。

图 6-3. 单指令、多数据 (SIMD) 处理



Cortex-A9 NEON MPE 执行以下类型的操作：

- SIMD 和标量单精度浮点计算
- 标量双精度浮点计算
- SIMD 和标量半精度浮点转换
- 8 位、16 位、32 位和 64 位有符号和无符号整数 SIMD 计算
- 单位系数的 8 位或 16 位多项式计算

可进行以下操作：

- 加法和减法
- 使用任意累加器的乘法 (MAC)
- 最高或最低值驱动通道选择操作
- 平方根倒数近似值
- 全面数据结构加载指令，包括 register-bank-resident 表格查找



要了解关于 Cortex-A9 NEON MPE 的更多信息，请参考 *Cortex-A9 NEON™ 媒体处理引擎技术参考手册*，修订版本 r3p0，可以从 ARM 网站 (infocenter.arm.com) 下载。

存储器管理单元

MMU 和 L1 和 L2 缓存同时使用以将软件使用的虚拟地址转换成硬件使用的物理地址。每个处理器都有一个专用 MMU。


MMU 支持表 6 - 2 中显示的 TLB。

表 6 - 2. 支持的 TLB

TLB 类型	存储器类型	项的数量	相联性
Micro TLB	指令	32	全相联
Micro TLB	数据	32	全相联
Main TLB	指令和数据	128	两路相联

主 TLB 具有以下功能：

- 使用 lock-by-entry 模式的可锁定的项
- 支持硬件页表漫游到 L1 高速数据缓存中执行查找

 要了解关于 MMU 的更多信息，在 ARM 网站 (infocenter.arm.com)，请参考 *Cortex-A9 技术参考手册*，修订版本 r3p0 的 *存储器管理单元* 章节。

MPU 地址映射被划分为以下区域：

- 引导区域
- SDRAM 区域
- FPGA 从器件区域
- HPS 外设区域

该部分介绍每个区域的位置和内容。

引导区域

引导区域的容量是 1 MB，基于地址 0。上电后，或 L3 互联复位后，引导区域由引导 ROM 占有，从而实现 Cortex-A9 MPCore 的引导。虽然引导区域容量是 1 MB，但是超过 64 KB 的访问是不合法的，因为引导 ROM 仅为 64 KB。

如图 6 - 4 所示，该 1 MB 区域接下来可以被重新映射到 SDRAM 的底部 1 MB。要了解更多信息，请参考“[SDRAM 区域](#)”。


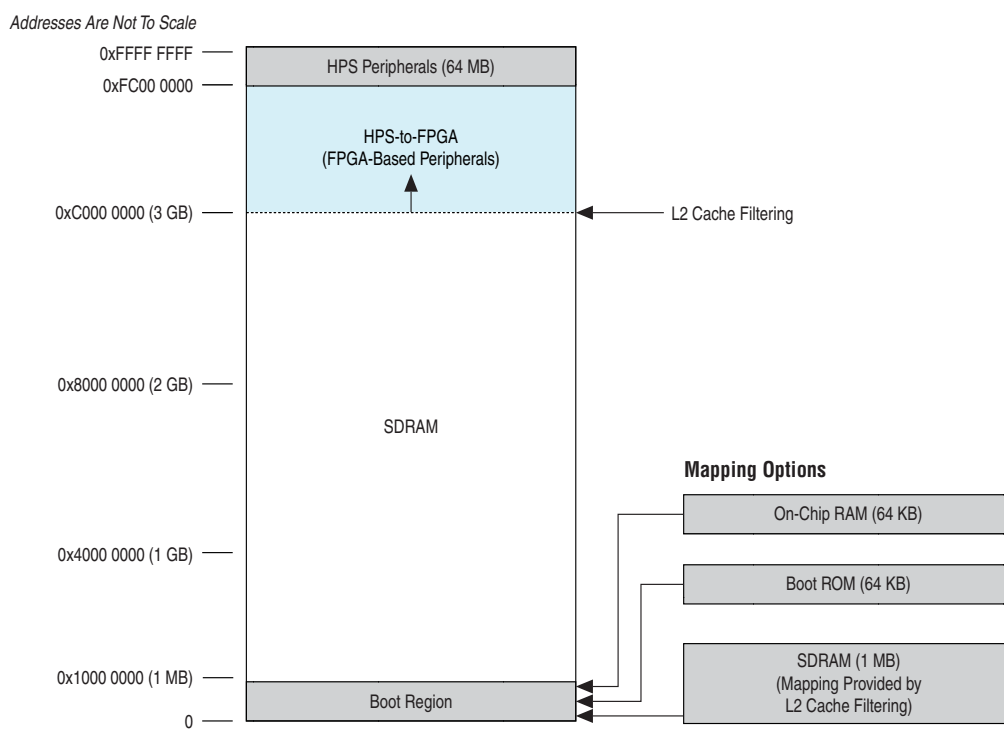
 另外，引导区域可以被映射到 64 KB 片上 RAM。要了解更多信息，请参考 *Cyclone V 器件手册* 第 3 卷的 *Interconnect* 章节。

图 6 - 4. MPCore 地址映射



SDRAM 区域

SDRAM 区域从地址 0x100000 (1 MB) 开始。区域的顶部由 L2 缓存滤波器决定。

L2 缓存包含一个过滤机制，可以将访问传导到 SDRAM 和 L3 互联。滤波器使用起始和结束地址定义滤波器范围。该滤波器范围内的所有访问都被传导到 SDRAM 子系统。该滤波器范围外的访问被传导到 L3 互联。

起始和结束地址在以下寄存器域中指定：

- regl2_addr_filtering_start.address_filtering_start
- regl2_address_filtering_end.address_filtering_end

要将 SDRAM 的较低 1MB 重新映射到引导区域，请将过滤器起始地址设置为 0x0 以确保 0x0 和 0xFFFF 之间的访问被传导到 SDRAM。单独地，可以在 0xC0000000 上将过滤器结束地址设置为以 1 MB 递增以便扩展 SDRAM 区域的较高界限。然而，通过影响 FPGA 外设地址跨度才可实现这个扩展范围。取决于 L2 缓存中的地址过滤器设置，SDRAM 区域的顶部的范围可从 0xBF000000 到 0xBF000000。

要了解关于 L2 缓存的更多信息，请参考 第 6 - 21 页的“L2 缓存”。

FPGA 从器件区域

Cortex-A9 MPU 子系统支持各种容量的 FPGA 从器件区域与基于 FPGA 的外设通信。取决于 L2 缓存过滤器设置，该区域可从低至 0xC0000000 开始。FPGA 从器件区域的顶部位于 0xBF000000。结果，FPGA 从器件区域的容量范围可以从 0 到 0x3F000000 字节。

HPS 外设区域

HPS 外设区域是在地址空间中的顶部 64 MB，从 0xFC000000 开始并且扩展到 0xFFFFFFFF。对于 Altera Cortex-A9 MPU 子系统，HPS 外设区域总是被分配到 HPS 专用外设。

性能监视单元

每个 Cortex-A9 处理器都有一个性能监视单元 (PMU)。PMU 支持 58 个事件以收集处理器和存储器系统的操作上的统计。PMU 中的 6 个计数器在实时累加事件。PMU 计数器可以使用协处理器 14 (CP14) 接口从处理器本身访问，或从外部调试器访问。可以将事件提供给 PTM 并且可用于触发或跟踪。



要了解关于 PMU 的更多信息，在 ARM 网站 (infocenter.arm.com) 上，请参考 *Cortex-A9 技术参考手册*，修订版本 r3p0 的 *性能监视单元* 章节。

MPCore 计时器

每个处理器有一个间隔计时器和一个看门狗计时器。

功能说明

每个计时器都是专用的，意味着只有它的相关的处理器可以对其访问。如果不需要看门狗计时器，那么它可以被配置为第二个间隔计时器。

每个专用间隔和看门狗计时器都有以下功能：

- 一个 32 位计数器，达到零时会选择性地生成一个中断
- 对于计数器可配置的起始值
- 限定时钟周期的 8 位预定标器值

实现详细信息

计时器可配置为 single-shot 或 auto-reload 模式。计时器模块由 mpu_periph_clk 提供时钟，以 mpu_clk 的频率的 $\frac{1}{4}$ 运行。



要了解关于专用计时器的更多信息，在 ARM 网站 (infocenter.arm.com) 上，请参考 refer to *Cortex-A9 MPCore 技术参考手册*，修订版本 r3p0 的 *全局计时器、专用计时器和看门狗寄存器* 章节中的“关于专用计时器和看门狗模块”。

通用中断控制器

功能说明

通用中断控制器 (GIC) 支持高达 180 个中断源，包括 FPGA 架构中实现的专用外设和 IP。在双核系统中，GIC 由两个 Cortex-A9 处理器共享。每个处理器也有 16 个 banked 软件生成的中断和 16 个 banked 专用外设中断。

实现详细信息

GIC 的配置和控制是存储器映射的并且通过 SCU 访问。GIC 由 mpu_periph_clk 提供时钟，以 mpu_clk 的频率的 $\frac{1}{4}$ 运行。



要了解关于 GIC 的更多信息，在 ARM 网站 (infocenter.arm.com)，请参考 *Cortex-A9 MPCore 技术参考手册*，修订版本 r3p0 的 *中断控制器* 章节。

表 6-3 显示了中断映射。

表 6-3. GIC 中断映射 (1/6)

GIC 中断数 ⁽¹⁾	源模块	中断名称	合并的中断	触发
32	CortexA9_0	cpu0_parityfail	⁽²⁾	Edge
33	CortexA9_0	cpu0_parityfail BTAC		Edge
34	CortexA9_0	cpu0_parityfail GHB		Edge
35	CortexA9_0	cpu0_parityfail I Tag		Edge
36	CortexA9_0	cpu0_parityfail I Data		Edge
37	CortexA9_0	cpu0_parityfail TLB		Edge
38	CortexA9_0	cpu0_parityfail D Outer		Edge
39	CortexA9_0	cpu0_parityfail D Tag		Edge
40	CortexA9_0	cpu0_parityfail D Data		Edge
41	CortexA9_0	cpu0_deflags0		Level
42	CortexA9_0	cpu0_deflags1		Level
43	CortexA9_0	cpu0_deflags2		Level
44	CortexA9_0	cpu0_deflags3		Level
45	CortexA9_0	cpu0_deflags4		Level
46	CortexA9_0	cpu0_deflags5		Level
47	CortexA9_0	cpu0_deflags6		Level
48	CortexA9_1	cpu1_parityfail	⁽³⁾	Edge
49	CortexA9_1	cpu1_parityfail BTAC		Edge
50	CortexA9_1	cpu1_parityfail GHB		Edge
51	CortexA9_1	cpu1_parityfail I Tag		Edge
52	CortexA9_1	cpu1_parityfail I Data		Edge
53	CortexA9_1	cpu1_parityfail TLB		Edge
54	CortexA9_1	cpu1_parityfail D Outer		Edge
55	CortexA9_1	cpu1_parityfail D Tag		Edge
56	CortexA9_1	cpu1_parityfail D Data		Edge
57	CortexA9_1	cpu1_deflags0		Level
58	CortexA9_1	cpu1_deflags1		Level
59	CortexA9_1	cpu1_deflags2		Level
60	CortexA9_1	cpu1_deflags3		Level
61	CortexA9_1	cpu1_deflags4		Level
62	CortexA9_1	cpu1_deflags5		Level
63	CortexA9_1	cpu1_deflags6		Level
64	SCU	scu_parityfail0	—	Edge
65	SCU	scu_parityfail1	—	Edge
66	SCU	scu_ev_abort		Edge
67	L2-Cache	l2_ecc_byte_wr_IRQ	—	Edge
68	L2-Cache	l2_ecc_corrected_IRQ	—	Edge
69	L2-Cache	l2_ecc_uncorrected_IRQ	—	Edge
70	L2-Cache	l2_combined_IRQ	⁽⁴⁾	Level
71	DDR	ddr_ecc_error_IRQ	—	Level
72	FPGA	FPGA_IRQ0	—	Level 或 Edge
73	FPGA	FPGA_IRQ1	—	Level 或 Edge
74	FPGA	FPGA_IRQ2	—	Level 或 Edge

表 6-3. GIC 中断映射 (2/6)

GIC 中断数 ⁽¹⁾	源模块	中断名称	合并的中断	触发
75	FPGA	FPGA_IRQ3	—	Level 或 Edge
76	FPGA	FPGA_IRQ4	—	Level 或 Edge
77	FPGA	FPGA_IRQ5	—	Level 或 Edge
78	FPGA	FPGA_IRQ6	—	Level 或 Edge
79	FPGA	FPGA_IRQ7	—	Level 或 Edge
80	FPGA	FPGA_IRQ8	—	Level 或 Edge
81	FPGA	FPGA_IRQ9	—	Level 或 Edge
82	FPGA	FPGA_IRQ10	—	Level 或 Edge
83	FPGA	FPGA_IRQ11	—	Level 或 Edge
84	FPGA	FPGA_IRQ12	—	Level 或 Edge
85	FPGA	FPGA_IRQ13	—	Level 或 Edge
86	FPGA	FPGA_IRQ14	—	Level 或 Edge
87	FPGA	FPGA_IRQ15	—	Level 或 Edge
88	FPGA	FPGA_IRQ16	—	Level 或 Edge
89	FPGA	FPGA_IRQ17	—	Level 或 Edge
90	FPGA	FPGA_IRQ18	—	Level 或 Edge
91	FPGA	FPGA_IRQ19	—	Level 或 Edge
92	FPGA	FPGA_IRQ20	—	Level 或 Edge
93	FPGA	FPGA_IRQ21	—	Level 或 Edge
94	FPGA	FPGA_IRQ22	—	Level 或 Edge
95	FPGA	FPGA_IRQ23	—	Level 或 Edge
96	FPGA	FPGA_IRQ24	—	Level 或 Edge
97	FPGA	FPGA_IRQ25	—	Level 或 Edge
98	FPGA	FPGA_IRQ26	—	Level 或 Edge
99	FPGA	FPGA_IRQ27	—	Level 或 Edge
100	FPGA	FPGA_IRQ28	—	Level 或 Edge
101	FPGA	FPGA_IRQ29	—	Level 或 Edge
102	FPGA	FPGA_IRQ30	—	Level 或 Edge
103	FPGA	FPGA_IRQ31	—	Level 或 Edge
104	FPGA	FPGA_IRQ32	—	Level 或 Edge
105	FPGA	FPGA_IRQ33	—	Level 或 Edge
106	FPGA	FPGA_IRQ34	—	Level 或 Edge
107	FPGA	FPGA_IRQ35	—	Level 或 Edge
108	FPGA	FPGA_IRQ36	—	Level 或 Edge
109	FPGA	FPGA_IRQ37	—	Level 或 Edge
110	FPGA	FPGA_IRQ38	—	Level 或 Edge
111	FPGA	FPGA_IRQ39	—	Level 或 Edge
112	FPGA	FPGA_IRQ40	—	Level 或 Edge
113	FPGA	FPGA_IRQ41	—	Level 或 Edge

表 6 - 3. GIC 中断映射 (3/6)

GIC 中断数 ⁽¹⁾	源模块	中断名称	合并的中断	触发
114	FPGA	FPGA_IRQ42	—	Level 或 Edge
115	FPGA	FPGA_IRQ43	—	Level 或 Edge
116	FPGA	FPGA_IRQ44	—	Level 或 Edge
117	FPGA	FPGA_IRQ45	—	Level 或 Edge
118	FPGA	FPGA_IRQ46	—	Level 或 Edge
119	FPGA	FPGA_IRQ47	—	Level 或 Edge
120	FPGA	FPGA_IRQ48	—	Level 或 Edge
121	FPGA	FPGA_IRQ49	—	Level 或 Edge
122	FPGA	FPGA_IRQ50	—	Level 或 Edge
123	FPGA	FPGA_IRQ51	—	Level 或 Edge
124	FPGA	FPGA_IRQ52	—	Level 或 Edge
125	FPGA	FPGA_IRQ53	—	Level 或 Edge
126	FPGA	FPGA_IRQ54	—	Level 或 Edge
127	FPGA	FPGA_IRQ55	—	Level 或 Edge
128	FPGA	FPGA_IRQ56	—	Level 或 Edge
129	FPGA	FPGA_IRQ57	—	Level 或 Edge
130	FPGA	FPGA_IRQ58	—	Level 或 Edge
131	FPGA	FPGA_IRQ59	—	Level 或 Edge
132	FPGA	FPGA_IRQ60	—	Level 或 Edge
133	FPGA	FPGA_IRQ61	—	Level 或 Edge
134	FPGA	FPGA_IRQ62	—	Level 或 Edge
135	FPGA	FPGA_IRQ63	—	Level 或 Edge
136	DMA	dma_IRQ0	—	Level
137	DMA	dma_IRQ1	—	Level
138	DMA	dma_IRQ2	—	Level
139	DMA	dma_IRQ3	—	Level
140	DMA	dma_IRQ4	—	Level
141	DMA	dma_IRQ5	—	Level
142	DMA	dma_IRQ6	—	Level
143	DMA	dma_IRQ7	—	Level
144	DMA	dma_irq_abort	—	Level
145	DMA	dma_ecc_corrected_IRQ		Level
146	DMA	dma_ecc_uncorrected_IRQ		Level
147	EMAC0	emac0_IRQ	⁽⁵⁾	Level
148	EMAC0	emac0_tx_ecc_corrected_IRQ		Level
149	EMAC0	emac0_tx_ecc_uncorrected_IRQ		Level
150	EMAC0	emac0_rx_ecc_corrected_IRQ		Level

表 6-3. GIC 中断映射 (4/6)

GIC 中断数 ⁽¹⁾	源模块	中断名称	合并的中断	触发
151	EMAC0	emac0_rx_ecc_uncorrected_IRQ		Level
152	EMAC1	emac1_IRQ	(5)	Level
153	EMAC1	emac1_tx_ecc_corrected_IRQ		Level
154	EMAC1	emac1_tx_ecc_uncorrected_IRQ		Level
155	EMAC1	emac1_rx_ecc_corrected_IRQ		Level
156	EMAC1	emac1_rx_ecc_uncorrected_IRQ		Level
157	USB0	usb0_IRQ		Level
158	USB0	usb0_ecc_corrected_IRQ		Level
159	USB0	usb0_ecc_uncorrected_IRQ		Level
160	USB1	usb1_IRQ		Level
161	USB1	usb1_ecc_corrected_IRQ		Level
162	USB1	usb1_ecc_uncorrected_IRQ		Level
163	CAN0	can0_sts_IRQ		Level
164	CAN0	can0_mo_IRQ		Level
165	CAN0	can0_ecc_corrected_IRQ		Level
166	CAN0	can0_ecc_uncorrected_IRQ		Level
167	CAN1	can1_sts_IRQ		Level
168	CAN1	can1_mo_IRQ		Level
169	CAN1	can1_ecc_corrected_IRQ		Level
170	CAN1	can1_ecc_uncorrected_IRQ		Level
171	SDMMC	sdmmc_IRQ		Level
172	SDMMC	sdmmc_porta_ecc_corrected_IRQ		Level
173	SDMMC	sdmmc_porta_ecc_uncorrected_IRQ		Level
174	SDMMC	sdmmc_portb_ecc_corrected_IRQ		Level
175	SDMMC	sdmmc_portb_ecc_uncorrected_IRQ		Level
176	NAND	nand_IRQ		Level
177	NAND	nandr_ecc_corrected_IRQ		Level
178	NAND	nandr_ecc_uncorrected_IRQ		Level
179	NAND	nandw_ecc_corrected_IRQ		Level
180	NAND	nandw_ecc_uncorrected_IRQ		Level
181	NAND	nande_ecc_corrected_IRQ		Level

表 6 - 3. GIC 中断映射 (5/6)

GIC 中断数 ⁽¹⁾	源模块	中断名称	合并的中断	触发
182	NAND	nande_ecc_uncorrected_IRQ		Level
183	QSPI	qspi_IRQ		Level
184	QSPI	qspi_ecc_corrected_IRQ		Level
185	QSPI	qspi_ecc_uncorrected_IRQ		Level
186	SPI0	spi0_IRQ	(6)	Level
187	SPI1	spi1_IRQ	(6)	Level
188	SPI2	spi2_IRQ	(6)	Level
189	SPI3	spi3_IRQ	(6)	Level
190	I2C0	i2c0_IRQ	(7)	Level
191	I2C1	i2c1_IRQ	(7)	Level
192	I2C2	i2c2_IRQ	(7)	Level
193	I2C3	i2c3_IRQ	(7)	Level
194	UART0	uart0_IRQ		Level
195	UART1	uart1_IRQ		Level
196	GPIO0	gpio0_IRQ	—	Level
197	GPIO1	gpio1_IRQ	—	Level
198	GPIO2	gpio2_IRQ	—	Level
199	Timer0	timer_14sp_0_IRQ	(8)	Level
200	Timer1	timer_14sp_1_IRQ	(8)	Level
201	Timer2	timer_oscl_0_IRQ	(8)	Level
202	Timer3	timer_oscl_1_IRQ	(8)	Level
203	Watchdog0	wdog0_IRQ	—	Level
204	Watchdog1	wdog1_IRQ	—	Level
205	时钟管理器	clkmgr_IRQ		Level
206	时钟管理器	mpuawakeup_IRQ		Level
207	FPGA 管理器	fpga_man_IRQ	(9)	Level
208	CoreSight	nCTIIRQ[0]		Level
209	CoreSight	nCTIIRQ[1]		Level

表 6-3. GIC 中断映射 (6/6)

GIC 中断数 ⁽¹⁾	源模块	中断名称	合并的中断	触发
210	片上 RAM	ram_ecc_corrected_IRQ		Level
211	片上 RAM	ram_ecc_uncorrected_IRQ		Level

表 6-3 注释：

(1) 要确保您使用正确的 GIC 中断数，您的代码应该参考使用符号的中断名称，如 **Interrupt Name** 列中所示。有符号的中断名称在使用操作系统的资源安装分布的头文件中定义。

(2) 该中断合并名称为 cpu0_parityfail_* 的中断。

(3) 该中断合并名称为 cpu1_parityfail_* 的中断。

(4) 该中断合并以下中断：DECERRINTR、ECNTRINTR、ERRRDINTR、ERRRTINTR、ERRWDINTR、ERRWTINTR、PARRDINTR、PARRTINTR 和 SLVERRINTR。

(5) 该中断合并 sbd_intr_o、lpi_intr_o 和 pmt_intr_o。

(6) 该中断合并以下中断：ssi_txe_intr、ssi_txo_intr、ssi_rxf_intr、ssi_rxo_intr、ssi_rxu_intr 和 ssi_mst_intr。

(7) 该中断合并以下中断：ic_rx_under_intr、ic_rx_full_intr、ic_tx_over_intr、ic_tx_empty_intr、ic_rd_req_intr、ic_tx_abrt_intr、ic_rx_done_intr、ic_activity_intr、ic_stop_det_intr、ic_start_det_intr 和 ic_gen_call_intr。

(8) 该中断合并 TIMINT1 和 TIMINT2。

(9) 该中断合并以下中断：fpga_man_irq[7..0]。

全局计时器

MPU 具有一个全局 64 位，自动递增计时器，它主要由操作系统使用。

功能说明


全局计时器由处理器通过 SCU 使用存储器映射的访问而被访问。全局计时器具有以下功能：

- 具有自动递增功能的 64 位递增计数器。它在发送中断后继续递增。
- 存储器映射在专用存储器区域。
- 仅在安全状态中复位模式时被访问。它仅可以被设置一次，但是安全代码可以随时对其进行读取。
- MPCore 中的两个 Cortex-A9 处理器都可访问。

实现详细信息

每个 Cortex-A9 处理器都有一个专用 64 位比较器，当计数器达到指定的值时会生成一个专用中断。对于该中断，每个 Cortex-A9 处理器都使用 banked ID 和 ID27。ID27 作为一个专用外设中断 (PPI) 被发送到 GIC。

全局计时器由 mpu_periph_clk 提供时钟，以 mpu_clk 的频率的 ¼ 运行。

 要了解关于全局计时器的更多信息，在 ARM 网站 (infocenter.arm.com)，请参考 *Cortex-A9 MPCore 技术参考手册*，修订版本 r3p0 中 *全局计时器*、*专用计时器*和*看门狗寄存器*章节的“关于全局计时器”。


窥测掌控单元

SCU 对 Cortex-A9 处理器和存储器系统，包括 L2 缓存管理数据业务。在多个主器件系统中，处理器和其它的主器件可以操作共享数据。SCU 确保每个处理器操作最新数据的副本，从而保持缓存的一致性。

功能说明

SCU 用于将 Cortex-A9 处理器和 ACP 连接到 L2 缓存控制器。SCU 执行以下功能：

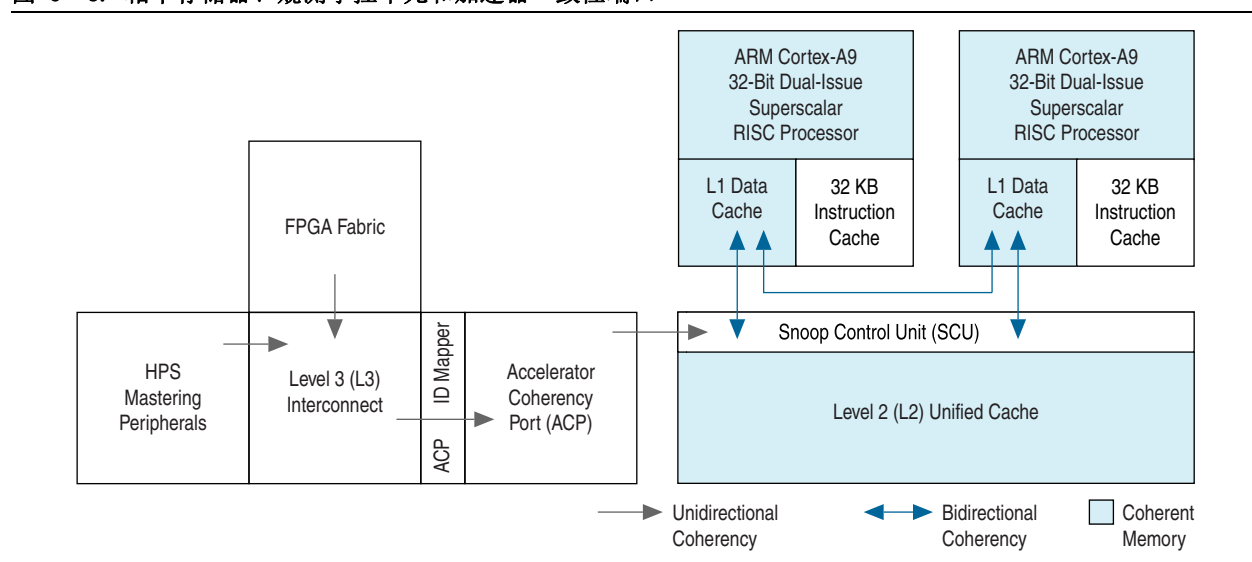
- 当处理器设置为 SMP 模式时，SCU 保持处理器之间的数据缓存一致性。


 SCU 不保持指令缓存的一致性。

- 开始 L2 缓存存储器访问
- 在请求 L2 访问的处理器之间进行仲裁
- 使用缓存一致性能管理 ACP 访问。

图 6-5 显示了双处理器系统中的 SCU，展示了 L1 数据缓存和 SCU 中的数据流程。

图 6-5. 相干存储器、窥测掌控单元和加速器一致性端口



 要了解关于 SCU 的更多信息，在 ARM 网站 (infocenter.arm.com) 上，请参考 *Cortex-A9 MPCore 技术参考手册*，修订版本 r3p0 的 *Snoop Control Unit* 章节。

实现详细信息

当处理器写入到所有相干存储器位置时，SCU 确保相关的数据是相干的（更新的、标记的或无效的）。同样，SCU 监控器从相干存储器位置读取操作。如果所需数据已经存储到其它处理器的 L1 缓存，那么数据会直接返回到请求处理器。如果数据不在 L1 缓存中，那么 SCU 发出一个读取到 L2 缓存。如果数据不在 L2 缓存存储器中，那么读取会被最终转送到主存储器。主要目的是最小化功耗和最大化整个存储器性能。

SCU 保持属于处理器的 L1 数据缓存之间的双向一致性。当一个处理器写入到它的 L1 缓存中的位置时，如果相同的位置在另一个 L1 缓存中被缓存，那么 SCU 对其进行更新。

非相干数据作为标准的读或写操作而通过。

如果两个 Cortex-A9 处理器都试图同时访问 L2 缓存，并且管理 ACP 的访问，那么 SCU 也在它们之间进行仲裁。

加速器一致性端口

ACP 支持外设 — 包括基于 FPGA 的外设 — 保持和 Cortex-A9 MPCore 处理器以及 SCU 的数据一致性。如第 6 - 17 页的图 6 - 5 所示，HPS 中的专用外设，以及内置在 FPGA 逻辑中的专用外设，通过 ACP ID 映射器和 ACP 访问相干存储器。要了解关于 ACP ID 映射器的更多信息，请参考第 6 - 18 页的“ACP ID 映射器”。


高带宽外设 (包含 FPGA 数据端口) 连接到 L3 互联。


突发容量和字节选通

ACP 提高 FPGA 架构中硬件加速器的系统性能。然而，为了实现高级性能，必须使用表 6 - 4 中列出的突发类型。其它的突发类型具有明显地较低性能。

表 6 - 4. 推荐的突发类型


突发类型	拍	宽度 (位)	地址类型	字节选通
回环	4	64	64 位对齐	置位
递增	4	64	32 位对齐	置位


 如果 FPGA- to- HPS 桥接的从端口不是 64 位宽，那么必须提供突发 (可以扩容或缩减到以上的突发类型) 到 FPGA- to- HPS 桥接。例如，如果 FPGA- to- HPS 桥接的从数据宽度是 32 位，那么 8 拍突发乘以 32 位需要有效访问 ACP。

 如果到 ACP 的传输的地址和突发容量匹配以上其中一种情况, 那么 MPU 中的逻辑假设传输的所有字节选通已经被设置。如果字节选通没有被全部设置，那么写实际上不会覆盖字中的所有字节。相反，缓存假设整个缓存行有效。如果该行较脏 (并且被写入到 SDRAM)，那么可能出现数据损坏。

专有和锁定访问

ACP 不支持相干存储器的专有访问。ACP 支持非相干存储器的专有访问；然而，专有访问传输不被 FPGA- to- HPS 桥接或 L3 互联的扩容和缩容逻辑所影响是非常重要的。如果专有访问由于容量逻辑而被拆分为多个传输，那么专有访问位由桥接或互联清零，并且专有访问失败。

 Altera 建议专有访问旁路 ACP, 通过直接连接到 L3 互联的 SDRAM 控制器的 32 位从端口或通过 FPGA- to- SDRAM 接口。

 要了解关于 SDRAM 控制器子系统的专有访问支持的更多信息，请参考 *Cyclone® V 器件手册* 第 3 卷的 *SDRAM Controller Subsystem* 章节。

ACP ID 映射器不支持锁定访问。要确保相互地专有访问共享数据，请使用内置在 SDRAM 控制器中的专有访问支持。

ACP ID 映射器

ACP ID 映射器位于第 3 层 (L3) 互联和 MPU 子系统 ACP 从器件之间。它负责将 12 位高级微控制器总线体系结构 (AMBA®) 高级可扩展接口 (AXI™) ID (输入 ID) 从 L3 互联映射到 ACP 从端口支持的 3 位 AXI ID (输出 ID)。

ACP ID 映射器将 1 GB 相干窗口实现为 4GB 地址空间。

功能说明

ACP 从器件支持高达 6 个主器件。然而，FPGA 架构中实现的定制外设可以具有需要访问 ACP 从器件的大量主器件。ACP ID 映射器使这些主器件可以访问 ACP。

ACP ID 映射器位于互联和 MPU 子系统的 ACP 从器件之间。它具有以下特性：

- 支持高达 6 个同时的 ID 映射
- 将 1 GB 相干窗口实现为 4 GB MPCore 地址空间
- 重新映射窥测掌控单元 (SCU) 和 L2 缓存使用的 5 位用户旁带信号

 要了解关于AXI用户旁带信号的更多信息, 请参考*CoreLink Level 2缓存控制器L2C-310 技术参考手册*, 可以从 ARM 网页 (infocenter.arm.com) 下载。

实现详细信息

ACP 由需要访问相干存储器的主器件访问。ACP 从端口可以由 L3 互联的主外设访问，也可以由 FPGA 架构（通过 FPGA-to-HPS 桥接）中实现的主器件访问。[第 6-2 页的图 6-1](#) 显示了 ACP ID 映射器。

ACP ID 映射器支持以下 ID 映射模式：

- 动态映射
- 固定映射

软件可以在 per-ID 基础上选择 ID 映射。对于配置为固定映射的输入 ID，有一个从输入 ID 到输出 ID 的一对一映射。当输入 ID 被配置为动态映射时，它会被自动映射到一个可用输出 ID。因为硬件处理映射，所以动态模式变得更灵活。硬件映射使您能够将一个输出 ID 用于多个输入 ID。输出 ID 在先到先得的基础上被分配到输入 ID。

在所有的 8 个输出 ID 中，只有 6 个可用于 L3 互联的主器件。前两个输出 ID (0 和 1) 专用于 MPU 子系统内的 Cortex-A9 处理器内核，后面的 6 个输出 ID (2-7) 用于 ACP ID 映射器。输出 ID 2-6 支持操作的固定和动态模式而输出 ID 7 仅支持动态模式。

通过 level 4 外设总线连接访问 ACP ID 映射器中的控制和状态寄存器，操作模式是可编程的。在复位时间，ACP ID 映射器对除了 ID 2 以外的所有输出 ID 默认为动态 ID 映射，但它对调试访问端口 (DAP) 输入 ID 复位到固定的映射。

[表 6-1](#) 总结了 3 位输出 ID 的预期使用，以及它们复位时的设置。

表 6-5. ID 的预期使用

输出 ID	复位状态	预期使用
7	动态	仅用于动态映射
6	动态	固定的或动态的，由软件编程。
5		
4		
3		
2	0x001 (DAP) 的固定状态	复位时分配到 DAP 的输入 ID。复位后，可以为固定的或动态的，由软件编程。
1	—	不被 ACP ID 映射器使用。
0		

对于不能驱动输入传输的 AXI 用户旁带信号的主器件，ACP ID 映射器可以控制覆盖该信号。ACP ID 映射器也控制 L3 互联的主器件访问映射到存储器的哪一个 1 GB 相干窗口。每个固定映射可以被分配一个不同的用户旁带信号和存储器窗口以支持不同主器件的指定设置。所有动态映射共享一个通用用户旁带信号和存储器窗口设置。

传输性能

每次，ACP ID 映射器的每 ID 映射可以接收和发出高达 15 个传输。在单独的列表中管理读和写 ID 映射，使得更多唯一输入 ID 可以在任何给定时间被重新映射。如果一个主器件发出一系列相同输入 ID 的读和写传输，那么不会存在顺序限制。

因为只有 6 个输出 ID 可用，所以每次进行的具有唯一 ID 的读和写传出各不多于 6 个。ACP 从端口的写接受是 5 个传输，读接受是 13 个传输。每次每 ID 映射只有 4 个相干读传输可以是未决的。

动态映射模式

在动态模式中，每个从 L3 主端口接收到的唯一输入 ID 都被分配到一个未使用的输入 ID。新输出 ID 被应用到传输，因为它被发送到 SCU 的 ACP 从端口。到达 ACP ID 映射器 (具有匹配已经在进行中传输的输入 ID) 的所有传输被映射到相同的输出 ID。一旦 ID 映射上的所有传输已经完成，该输出 ID 就会被释放并且可再次用于其它的输入 ID。

固定映射模式

在固定模式中，输出 ID 2 到 6 可以由软件分配到一个指定的 12 位输入 ID。这一操作将使用 L2 缓存控制器的 lock-by-master 功能变为可能，因为主器件的输入传输 ID 总是被分配到一个指定的输出 ID。与动态模式不同，ID 7 不可用于固定映射，因为它为动态模式而保留，仅用于避免系统死锁。

ACP ID 映射器具有两个 bank 的寄存器以控制映射的行为，这两个 bank 分别为请求 bank 和只读状态 bank。两个 bank 都包含相同数量的寄存器。要更改一个特定映射的设置 (一个指定的固定 ID 或所有动态映射)，软件应该写入请求 bank 中的相应寄存器。硬件检查请求，在安全的情况下，意味着当没有具有输出 ID 的未决的传输时才会应用更改。当应用更改时，状态寄存器被更新。软件通过轮询相应的状态寄存器检查更改是否已经生效。

表 6-6 显示了可以访问 ACP ID 映射器的每个 HPS 外设主器件的互联发出的输入 ID。

表 6-6. HPS 外设主器件输入 ID

互联主器件	ID ⁽¹⁾
DMA	00000xxxx011
EMAC0	10000xxxx001
EMAC1	10000xxxx010
USB0	100000000011
USB1	100000000110
NAND	1xxxxxxxx100
ETR	100000000000
DAP	000000000001
SD/MMC	100000000101
FPGA- to- HPS 桥接	0xxxxxxxx100
表 6-6 注释：	
(1) 值使用二进制表示。字母 x 表示每个主器件通过每个传输时的不同 ID 位。	

对 AXI 用户旁带信号的控制

ACP ID 映射器模块支持对 AXI 用户旁带信号值的控制。不是所有的主器件都驱动这些信号，所以 ACP ID 映射器使得驱动默认值（动态模式中）或指定值（固定模式中）的 5 位用户旁带信号成为可能。

有些寄存器可以对所有传输的用户旁带信号配置默认值，并且在固定映射模式中，对特定传输的这些信号配置固定值。在动态模式中，使用存储在寄存器中的默认值，输入传输的用户旁带信号被映射。在固定映射模式中，传输的输入 ID 被映射到 3 位输出 ID 并且传输的用户旁带信号使用与输出 ID 相一致的存储在寄存器中的值而被映射。一个重要的例外是，ACP ID 映射器总是允许 FPGA-to-HPS 桥接的用户旁带信号传递到 ACP，而与 ID 相关的用户旁带值无关。

存储器区域重新映射

ACP ID 映射器具有 1 GB 的地址空间，它被默认为到 SDRAM 的底部 1 GB 的视图。在动态和固定模式中，映射器也支持传输被传递到不同的 1 GB 容量的存储区域，称为页面。输入 32 位 AXI 地址信号的两个最高有效位被替代为 2 位用户配置的地址页面解码信息。页面解码器使用表 6-7 中显示的值。

表 6-7. 页面解码器值

页面	地址范围
0	0x00000000—0x3FFFFFFF
1	0x40000000—0x7FFFFFFF
2	0x80000000—0xBFFFFFFF
3	0xC0000000—0xFFFFFFFF

通过使用这个页面解码信息，一个主器件可以读取或写入 4 GB 存储器空间的任何 1 GB 区域，同时保持与 MPU 子系统的缓存一致性。

通过该功能，一个调试器可具有一个到主存储器的一致性的试图，而无需中断处理器。例如，复位时，DAP 输入 ID (0x001) 被映射到输出 ID 2，所以调试器可以改变 DAP 访问的 1 GB 窗口而不影响到 ACP 的任何其它传输业务。

L2 缓存

MPU 子系统包含一个辅助的 512 KB L2 共享的，统一的缓存存储器。

功能说明

L2 缓存比 L1 缓存较大。L2 缓存比外部存储器具有明显的较低延迟。L2 缓存高达八路相联，可配置为一路（直接映射）。如同 L1 缓存，L2 缓存可以按缓存线锁定、按路锁定或按总线主控锁定。

L2 缓存实现错误纠正代码 (ECC) 和 ECC 错误报告。缓存可以对处理器和操作系统报告大量的事件。


L2 缓存包含具有如下配置的 ARM L2C-310 L2 缓存控制器：

- 512 KB 的总存储
- 八路相联 (Eight-way associativity)
- 物理地址，物理标签
- 32 字节的缓存线长度

- 关键首字行填写 (Critical first word linefills)
- 支持所有 AXI 缓存模式，如表 6-8 所示。

表 6-8. AXI 缓存模式支持

缓存模式
写通 (Write-through) (1)
写回 (Write-back) (1)
分配读 (Read allocate)
分配写 (Write allocate)
分配读和写
表 6-8 注释： (1) 使用 ECC 时出现限制。要了解关于 SEU 保护的更多信息，请参考 <i>Cyclone V 器件手册</i> 的第 3 卷的 <i>System Manager</i> 章节。

- 单粒子翻转 (SEU) 保护
 - Tag RAM 上的奇偶校验
 - L2 Data RAM 上的 ECC
-  要了解关于 SEU 保护的更多信息，请参考 *Cyclone V 器件手册* 第 3 卷的 *System Manager* 章节。
- SCU 控制的 2 个从端口
- 与以下从端口连接的 2 个主端口：
 - SDRAM 控制器，64 位从端口宽度
 - L3 互联，64 位从端口宽度
- 缓存 lockdown 性能如下所示：
 - 线锁定
 - 通过路锁定
 - 通过主器件锁定（处理器和 ACP 主器件）
- TrustZone 支持
- 缓存事件监控。要了解更多信息，请参考 第 6-24 页的“L2 缓存事件监控”。

第 6-2 页的图 6-1 显示了 L2 缓存。L2 缓存可以访问 L3 互联架构或 SDRAM。L2 缓存地址过滤决定多少地址空间被分配到 HPS-to-FPGA 桥接以及多少被分配到 SDRAM，如 第 6-8 页的“存储器管理单元”中所介绍。

ECC 支持

L2 缓存具有使用 ECC 防止缓存 RAM 中 SEU 错误的选项。


使能 ECC 不影响 L2 缓存的性能。ECC 位只对数据 RAM 的写入而计算，它是 64 位宽 (8 字节或 1/4 的缓存线长度)。当计算 ECC 位时，ECC 逻辑不执行 read-modify-write。ECC 保护位在以下情况中没有效：

- 被写的数据在存储器中不是 64 位对齐
- 被写的数据在宽度上少于 64 位

在这些情况中，字节写错误 (Byte Write Error) 中断被置位。当这样的错误发生时，缓存数据仍然会被写入。然而，ECC 错误检测和纠正继续运行。因此，在接下来的读取中缓存数据可能错误。

要使用 ECC，软件和系统必须满足以下要求：

- 对于所有可缓冲的存储器区域，L1 和 L2 缓存必须被配置为写回分配。
- 使用 ACP 的 FPGA 软核 IP 必须仅执行以下类型的数据写入：
 - 存储器中的 64 位对齐
 - 64 位宽访问

 要了解关于 SEU 错误的更多信息，请参考 *Cyclone V 器件手册* 第 3 卷的 *System Manager* 章节。

实现详细信息

表 6-9 显示了缓存控制器的参数设置。

表 6-9. 缓存控制器配置


功能	含义
快取通道 (Cache way) 容量	64 KB
快取通道数量	8 个通道
RAM 延迟	2 个周期的延迟
奇偶校验逻辑	使能奇偶校验逻辑
通过主器件锁定	使能了通过主器件锁定
通过线锁定	使能了通过线锁定
从端口上的 AXI ID 宽度	从端口上的 6 个 AXI ID 位
地址过滤	使能了地址过滤逻辑
推测性读	使能了支持推测性读的逻辑
存在 ARUSERMx 和 AWUSERMx 旁带信号	使能了旁带信号

 要了解关于缓存控制器可配置选项的更多信息，在 ARM 网站 (infocenter.arm.com)，请参考 *CoreLink Level 2 缓存控制器 L2C-310 技术参考手册*，修订版本 r3p2。

L2 缓存 Lockdown 性能

L2 缓存使用 3 个方法锁定缓存 RAM 中的数据：

- 通过线锁定 — 用于锁定缓存中的线。该方法通常用于暂时将软件的临界段加载到缓存。
- 通过路锁定 — 支持任何或所有 8 个快取通道 (cache way) 被锁定。该方法通常用于将临界数据或代码加载到缓存。
- 通过主器件锁定 — 支持快取通道专用于一个单一主端口。这一性能使得较大缓存如同 多个主端口的较小缓存。L2 缓存可以由 CPU0、CPU1 或 6 个 ACP 主器件控制，可能总共有 8 个主端口。


 要了解关于 L2 缓存 lockdown 性能的更多信息，在 ARM 网站 (infocenter.arm.com)，请参考 *CoreLink Level 2 缓存控制器 L2C-310 技术参考手册*，修订版本 r3p2 的 *功能概述* 章节中的 “缓存操作”。

L2 缓存事件监控


L2 缓存支持表 6 - 10 所示的内置缓存事件监控信号。每次，L2 缓存可以计算两个事件。

表 6 - 10. L2 缓存事件

事件	说明
C0	从 L2 缓存逐出一行。
DRHIT	L2 缓存中的数据读命中 (Data read hit)。
DRREQ	L2 缓存的数据读查看 (Data read lookup)。接下来导致命中 (hit) 或失效 (miss)。
DWHIT	L2 缓存中的数据写命中 (Data write hit)。
DWREQ	L2 缓存的数据写查找 (Data write lookup)。接下来导致一个命中或失效。
DWTREQ	具有写通属性的 L2 缓存的数据写查找 (Data write lookup)。接下来导致一个命中或失效。
EPFALLOC	分配到 L2 缓存的预取提示 (Prefetch hint)。
EPFHIT	L2 缓存中的预取提示命中 (Prefetch hint hit)。
EPFRCVDS0	从端口 S0 接收到的预取提示 (Prefetch hint)。
EPFRCVDS1	从端口 S1 接收到的预取提示 (Prefetch hint)。
IPFALLOC	分配 L2 缓存控制器生成的预取到 L2 缓存。
IRHIT	L2 缓存中的指令读命中 (Instruction read hit)。
IRREQ	L2 缓存的指令读查找 (Instruction read lookup)。接下来导致一个命中或失效。
SPNIDEN	安全特权非侵入性调试使能。
SRCONFS0	在从端口 S0 中确定的推测性读 (Speculative read)。
SRCONFS1	在从端口 S1 中确定的推测性读。
SRRCVDS0	从端口 S0 接收的推测性读。
SRRCVDS1	从端口 S1 接收的推测性读。
WA	具有写分配属性的写操作导致的 L2 缓存的分配缺失。

 要了解关于内置 L2 事件监控性能的更多信息，在 ARM 网站 (infocenter.arm.com)，请参考 *CoreLink Level 2 缓存控制器 L2C-310 技术参考手册*，修订版本 r3p2 的功能概述章节中的 “实现详细信息”。

除此之外，L2 缓存事件可以使用专用调试电路采集或盖上时间戳。

 要了解关于 L2 事件采集的更多信息，在 ARM 网站 (infocenter.arm.com)，请参考 *Cortex-A9 MPCore 技术参考手册*，修订版本 r3p0 的 *Debug* 章节。

调试模块

MPU 子系统使用 ARM CoreSight 片上调试和跟踪包括调试资源。它包括以下功能性：

- 每个处理器的单独编程跟踪
- Cortex-A9 MPCore 的事件跟踪
- 处理器和其它 HPS 调试功能之间的交叉触发

编程跟踪

每个处理器具有提供实时指令流跟踪的独立的 PTM。PTM 与大量第三方调试工具兼容。


PTM 使用高度压缩格式提供跟踪数据。在程序执行流程中，跟踪数据对于指定点包含标签，称为路标点 (waypoints)。路标点是程序流中指定的事件或更改。


PTM 识别并且标记表 6 - 11 中列出的路标点。

表 6 - 11. PTM 支持的路标点

类型	额外的路标点信息
间接分支	目标地址和条件码
直接分支	条件码
指令屏障指令 (Instruction barrier instructions)	—
例外	例外发生的位置
处理器指令集状态中的更改	—
处理器安全状态中的更改	—
Context ID 更改	—
当中止调试模式使能时从调试状态进入和返回	—

PTM 选择性地对路标点提供额外信息，包括以下信息：

- 路标点之间的处理器周期数
 - 全局时戳值
- 


要了解关于全局时戳的更多信息，请参考 *Cyclone V 器件手册*第 3 卷的 *CoreSight Debug and Trace* 章节。
- 直接分支的目标地址
- 

要了解关于 PTM 的更多信息，在 ARM 网站 (infocenter.arm.com)，请参考 *CoreSight PTM A9 技术参考手册*，修订版本 r1p0。

事件跟踪

每个处理器的事件可以用作 PTM 的输入。PTM 可以将这些事件用作跟踪和触发条件。


要了解关于事件跟踪的更多信息，请参考 第 6 - 10 页的 “性能监视单元”。

- 


要了解关于触发和跟踪性能的更多信息，在 ARM 网站 (infocenter.arm.com)，请参考 *CoreSight PTM A9 技术参考手册*，修订版本 r1p0。

交叉触发

PTM 可以导出触发事件并且在触发输入上执行操作。交叉触发信号与包含 FPGA 架构的其它 HPS 调试组件连接。同样，一个处理器中的故障可以触发另一个处理器的故障。



要了解关于交叉触发的详细信息，请参考 *Cyclone V 器件手册*第 3 卷的 *CoreSight Debug and Trace* 章节。



要了解关于 MPU 中调试硬件的更多信息，请参考 *Cyclone V 器件手册*第 3 卷的 *CoreSight Debug and Trace* 章节。

Cortex-A9 MPU 子系统寄存器实现

以下配置通过 Cortex-A9 子系统寄存器实现：

- 所有处理器相关的控制，包括 MMU 和 L1 缓存，通过使用每个单独的处理器的协处理器 15 (CP15) 寄存器而被控制。
- 所有 SCU 寄存器，包括计时器和 GIC 的控制，是存储器映射可访问的。
- 所有 L2 缓存寄存器都是存储器映射的。



要了解关于外设从端口的地址映射，包括 SCU 和 L2 缓存的更多信息，请参考 *Cyclone V* 器件手册第 3 卷的 *Introduction to the Hard Processor System* 章节。要了解关于 Altera Cortex-A9 MPU 子系统的寄存器定义的详细信息，在 ARM 网站 (infocenter.arm.com)，请参考 *Cortex-A9 MPCore 技术参考手册*，修订版本 r3p0，以及 *CoreLink Level 2 缓存控制器 L2C-310 技术参考手册*，修订版本 r3p2。

文档修订历史

表 6 - 12 显示了该文档的修订历史。

表 6 - 12. 文档修订历史

日期	版本	修订内容
2012 年 11 月	1.2	少量文本编辑。
2012 年 5 月	1.1	<ul style="list-style-type: none">■ 添加了 ACP ID 映射器的说明■ 合并冗余信息
2012 年 1 月	1.0	首次发布。

