



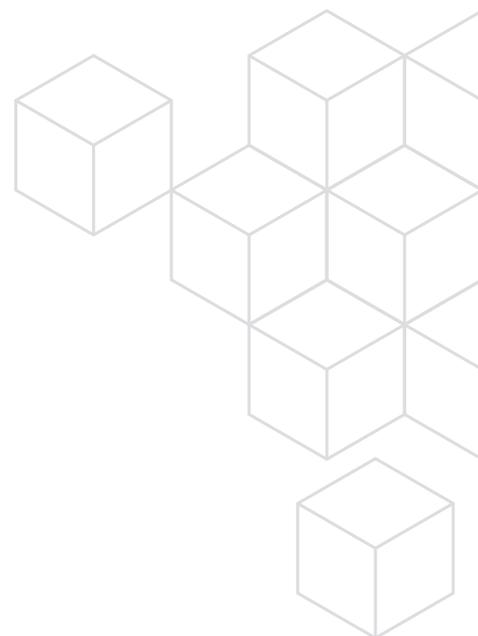
解决系统级一致性验证挑战

来源: Cadence 公司

本文简要介绍现代系统级芯片的一致性, 以及系统级芯片集成商为确保可靠的系统功能而面临的挑战。此外说明 Cadence® 系统级验证 IP (System VIP) 如何为加速系统级芯片验证提供重要帮助。

目录

引言	2
挑战	3
System VIP	5
总结	8



简介

经过摩尔定律数十年的发展，定期提升性能、功耗和功能的理念已经在技术领域根深蒂固。移动设备在 20 世纪 90 年代初至 21 世纪初的 10 年间引领了技术发展，但在 21 世纪初期，由于频率扩展无法提供相应的功耗或性能，单处理器架构显然失去了发展势头。

在移动领域，系统通常包括一个专用的图形处理器，从而实现图形相关任务的加速。向多处理器设备的转变是由通用处理器供应商发起的，但很显然，为了满足对于 AI 能力与日俱增的需求，需要将很多特定领域的加速器纳入到系统级芯片 (SoC) 架构中。

除了移动领域，其他领域也同样推动更多创新架构的开发。基础架构领域需要可编程加速器、高性能网络引擎等，而汽车系统和边缘物联网系统则需要大量专用引擎。

计算创新的爆炸式增长表明我们正处于系统级芯片架构的“黄金时代”，这的确是令人激动的时代。这些异构引擎需要共享数据，同时避免不必要的数据移动，从而实现高效和有效的运行。为了实现这一目标，整个系统级芯片都采用了本地化高速缓存 (cache)，而在共享数据的位置，需要保持缓存一致性。

为了说明现代系统级芯片中高速缓存的爆炸式增长，图 1 展示了一个典型的双集群示例。架构上的所有创新都需要耗费巨大的成本，即需要在流片之前验证缓存系统。在完成所有一致性组件的集成后，需要在系统级芯片级别上进行全面的验证，以避免一致性失败的风险。如果系统级芯片包含 CCIX 或 CXL 等任何包含一致性的片外协议，而这些协议旨在支持加速器或多个处理集群等一致性的配套芯片，那么会带来超出 CPU 子系统的挑战。

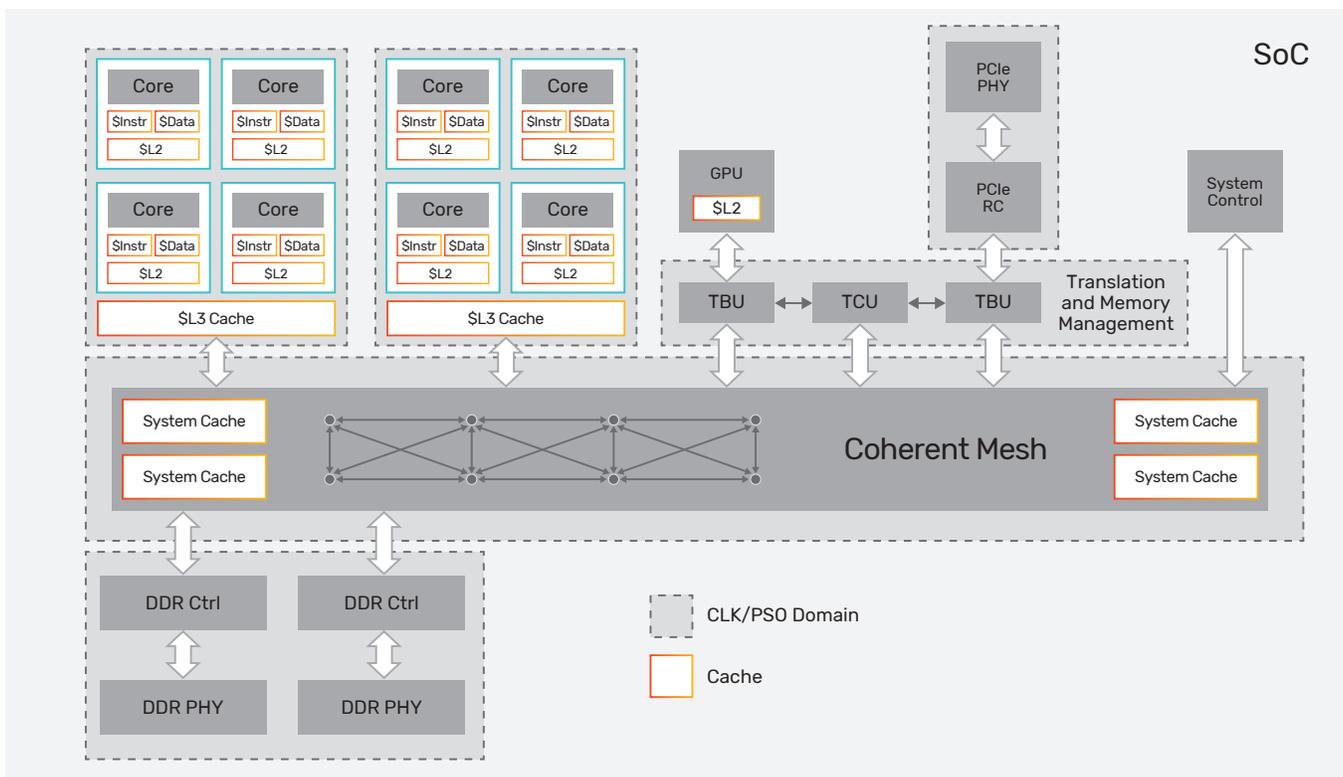


图 1: 典型的 CPU 子系统缓存架构

挑战

所有这些异构处理单元之间的一致性必须完美无缺。系统级芯片开发人员应该确信，所有的部件在集成后能够在任何场景下共同工作。而在大多数情况下，这些部件来自内部或第三方供应商等不同的团队。为确保一致性系统的保真度，验证工程师必须创建三部分内容。首先创建测试平台，从而提供框架来执行第二部分测试内容。第三部分内容是检查程序 (checker)，以确保始终保持一致性。

基于执行测试所选择的引擎进一步增加了复杂性。在较小部件和子系统验证时，Cadence Xcelium™ Logic Simulator 等 RTL 仿真器成为验证工程师的首选。然而，当子系统完成组装，完全集成的系统级芯片成为待测设计 (DUT) 时，RTL 仿真不再有效，这时需要选择硬件加速器。为轻松完成整个系统的一致性验证，同时使用仿真器和硬件加速器的能力至关重要。测试平台需要为一系列一致性的片上总线协议提供支持，平台通常由 VIP 构建，以便进行仿真或硬件加速。

搭建好测试平台后，创建测试内容既耗时且容易出错，如果手动执行更是如此。用于仿真的测试平台通常使用通用验证方法 (UVM) 组件，而这些组件需要使用 SystemVerilog 编写测试序列 (sequence)。虽然 VIP 可为一致的片上总线提供基本的序列支持，但创建内容十分艰巨，并且这些序列通常无法移植到硬件加速测试平台上运行，因为硬件加速平台倾向使用 C 语言编写用例。

我们可以快速浏览一下验证系统级一致性所需的内容类型，从而更加了解这一挑战。图 2 展示了在典型集群内引发核心间的一致性操作所需生成的一致性数据流。配置系统内存映射后，CPU 核的数据流生成相对简单，但使用 UVM VIP 模仿这种行为会较为棘手，因为需要考虑所有可共享或不可共享的边带信号。从底层接口使测试抽象化的能力是实现高效测试开发的关键。

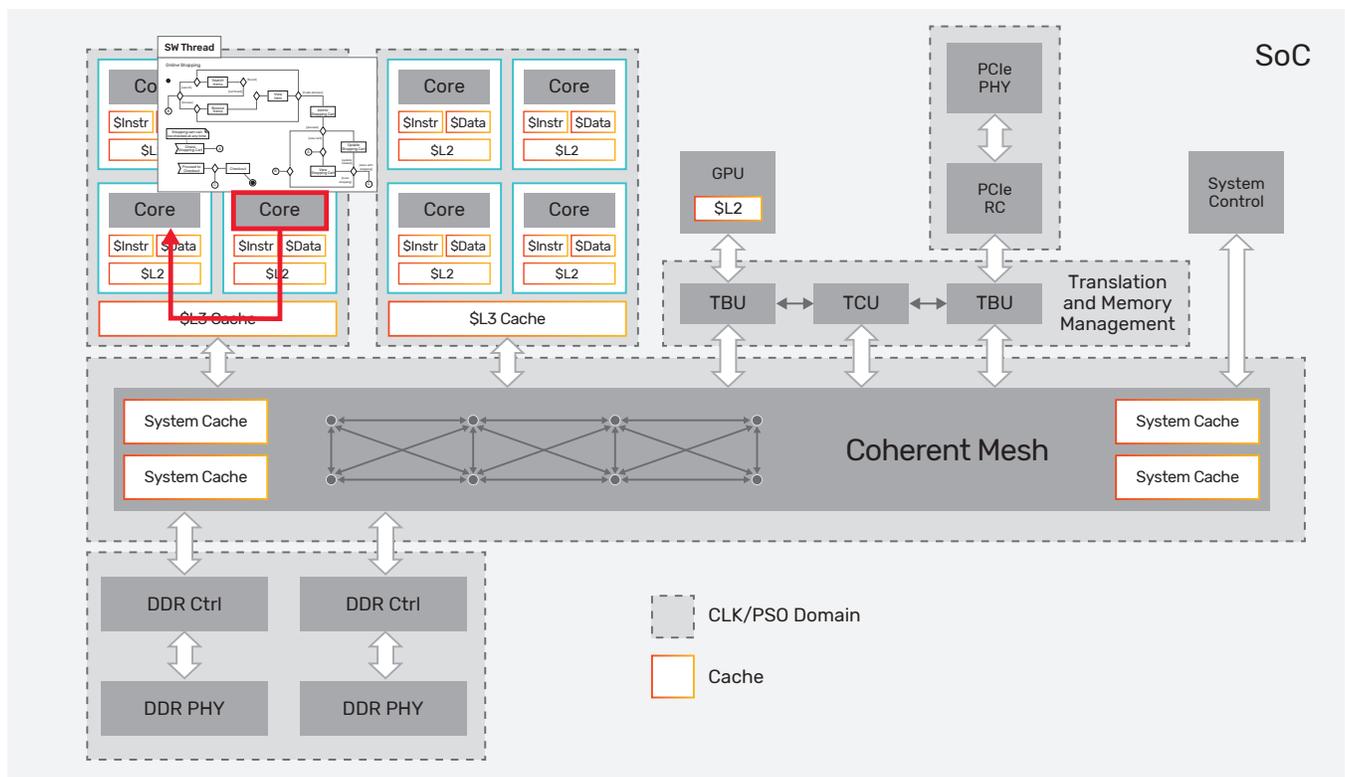


图 2: 集群内一致性

图 3 展示了如何扩展测试，以提供集群至集群的一致性。到目前为止，我们看到的很可能是同质一致性的扩展，即测试可能都在相似的执行核心上运行，可能是在相似的实现架构上运行相同的指令。但是，正如引言中所述，大多数现代系统级芯片包含一系列异构处理引擎。在我们的示例中，GPU 就具有完全不同的处理指令集和工具链。

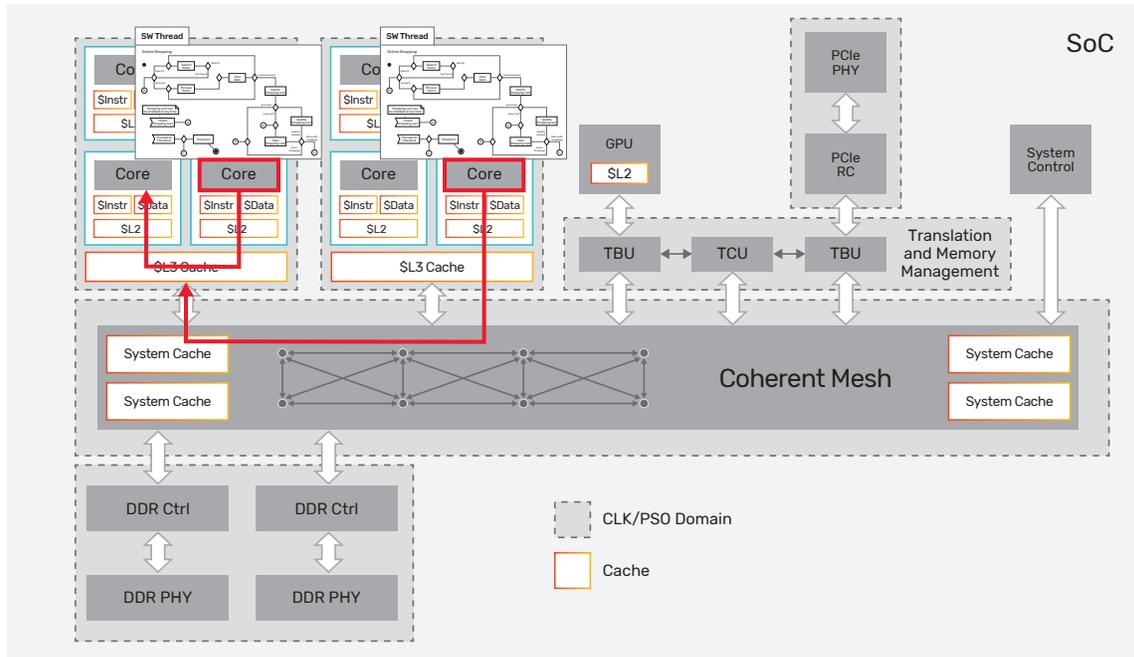


图 3: 集群间一致性

创建图 4 中所示的测试需要使用混合线程以创建一致性场景。再次强调，为实现测试扩展以提供丰富的验证场景集，使接口和指令集架构 (ISA) 抽象化的能力至关重要，这样才能确保最终的系统级芯片能在硅片处理后正常运行。

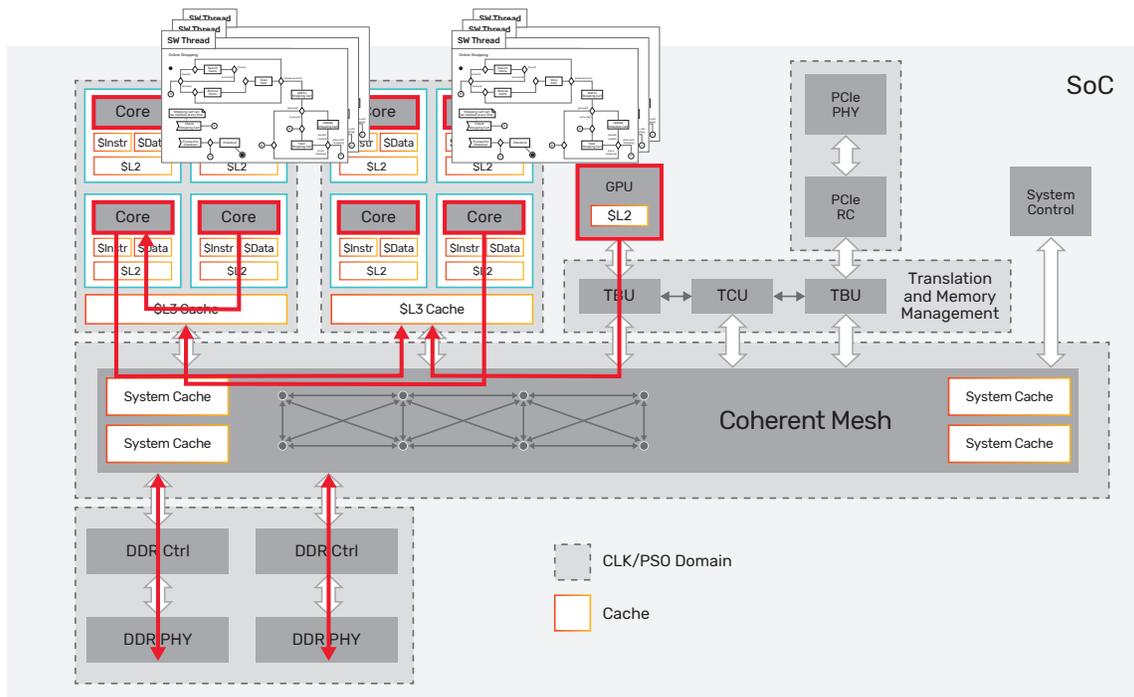


图 4: 添加 I/O 一致性缓存合作者

怎样的测试才能充分验证一致性机制是否正常运行？需要两个关键的缓存场景，“伪共享 (false sharing)”和“真共享 (true sharing)”。在伪共享场景中，多个缓存使用者共享缓存行 (cache line)，但只能访问 cache line 上的一部分，并且是独占访问 cache line 的不同部分。而在真共享时，数据将跨多个缓存使用者共享，并可产生在系统中将数据从一个使用者复制到另一个使用者的场景。图 5 展示了一种情况，其中三个缓存使用者共享同一条缓存行，但使用该行中不同的字节位置。每个使用者颜色不同，以显示其对不同缓存位置的访问。每个线程会将数据写入适当的字节地址。

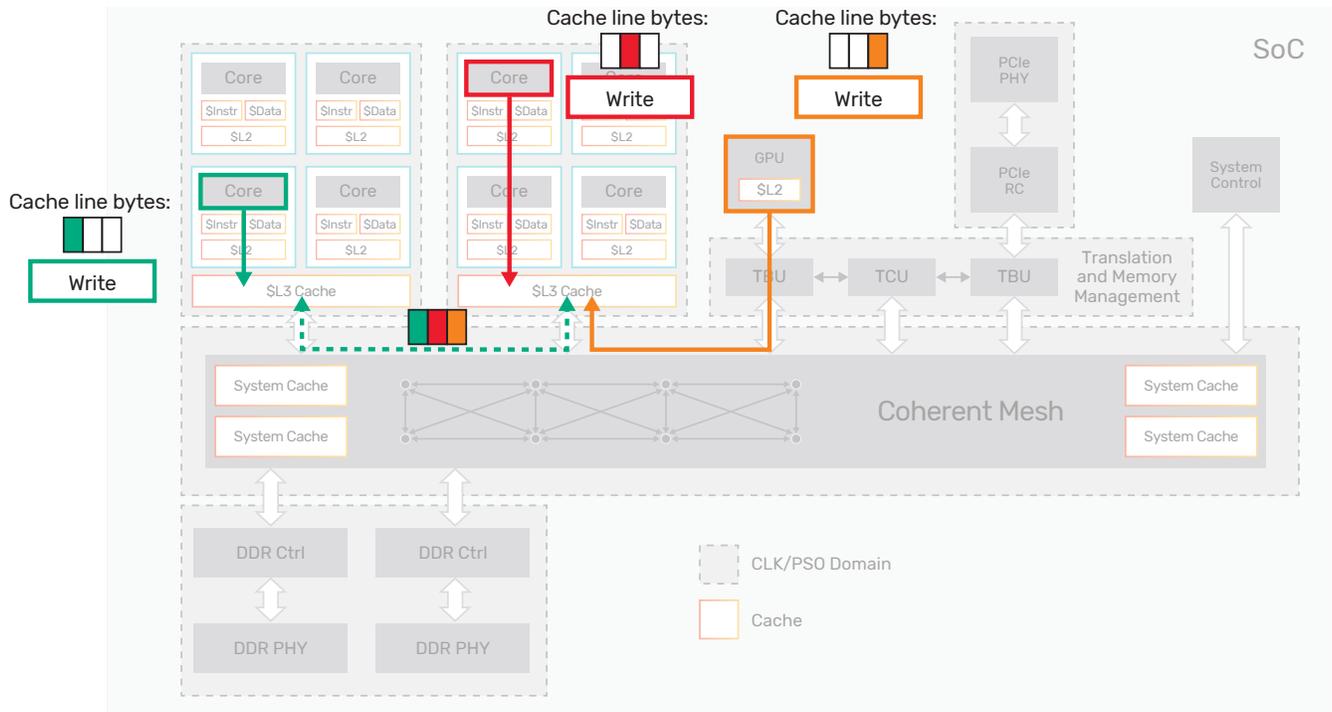


图 5: 伪共享缓存测试方案

我们已经提到，需要使用一个测试平台和一组用例来检查一致性。全面的解决方案所需的第三部分是检查程序，用于确保整个系统级芯片的数据完整性。该功能通常以记分板 (scoreboard) 的形式实现，但它必须是一个“可感知一致性的”记分板，以便建模和检查 evictions、snoops 和 cache stashing 等一致性功能。创建这样的记分板极具挑战，尤其是它需要在仿真和硬件加速环境中都能运行。

System VIP

Cadence 公司的 System VIP 解决方案是由一套工具和库组成，可为系统级芯片集成商提供重要帮助，助力集成商应对诸多挑战。该解决方案包括四大支柱，以应对现代系统级芯片验证工程师所面临的一系列挑战。利用已建立的验证引擎，即仿真、硬件加速和 FPGA 原型验证，System VIP 组件可提供测试平台自动化、开箱即用的便携式内容、性能分析以及系统级数据完整性检查器。

System Testbench Generator

System Testbench Generator 是生产力工具，通过输入的原始数据自动创建测试平台，从而实现快速测试启动，如图 6 所示。使用以逗号分隔的变量 (CSV)、IP-XACT 和拓扑详细信息，生成器基于 UVM 创建可以在 Xcelium Logic Simulator 上运行的 System Verilog 测试平台，或是可以在 Cadence Palladium® Z1 Enterprise Emulation Platform 上运行的基于 C 语言的测试平台。测试平台使用 Cadence VIP 或 Accelerated VIP (AVIP) 构建，对应目标引擎上的各种接口。

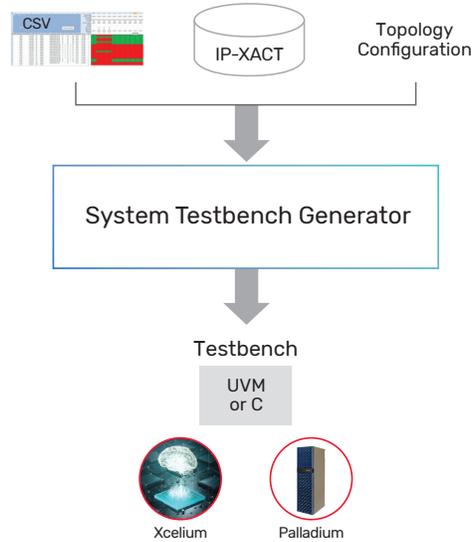


图 6: System Testbench Generator 流程

System Traffic Libraries

创建的测试平台 (无论是基于 UVM 还是 C 语言) 都能够和 System Traffic Libraries 对接, System Traffic Libraries 是一系列丰富的开箱即用内容。可以在图 7 查看生成的测试平台内容的示例。

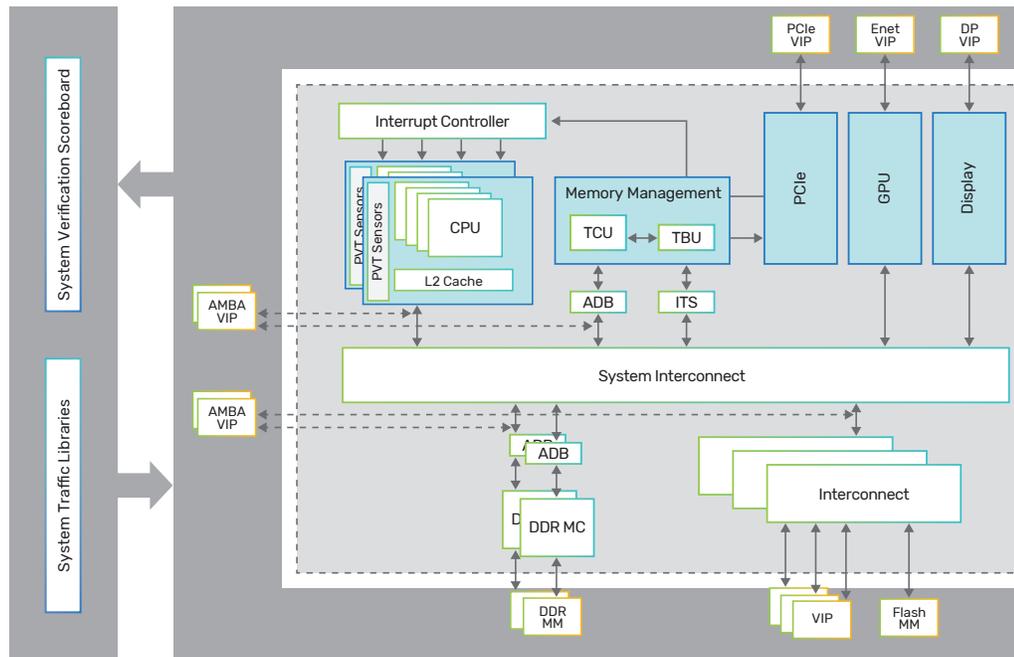


图 7: 生成的测试平台架构示例

System Traffic Libraries 包括 Coherency、Performance 和 PCIe Libraries，这些内容可以帮助客户使用 Accellera Portable Stimulus Standard (PSS) 和 Cadence Perspec™ System Verifier 快速创建测试场景。Coherency Library 包含一系列打包的用例，这些用例可以用来验证系统的一致性，例如 false sharing 场景，该场景使用可配置数量的线程来构造复杂的一致性测试用例。

图 8 显示了一个伪共享测试，可以在 16 个核心上运行。如果我们剖析细节，可以看到四个关键步骤：

1. 选择将由多个处理器共享的缓存区域
2. 在处理器之间分配缓存
3. 所有 16 个核心以不同的地址偏移量读写 cache line
4. 对照库参考模型检查缓存数据的一致性

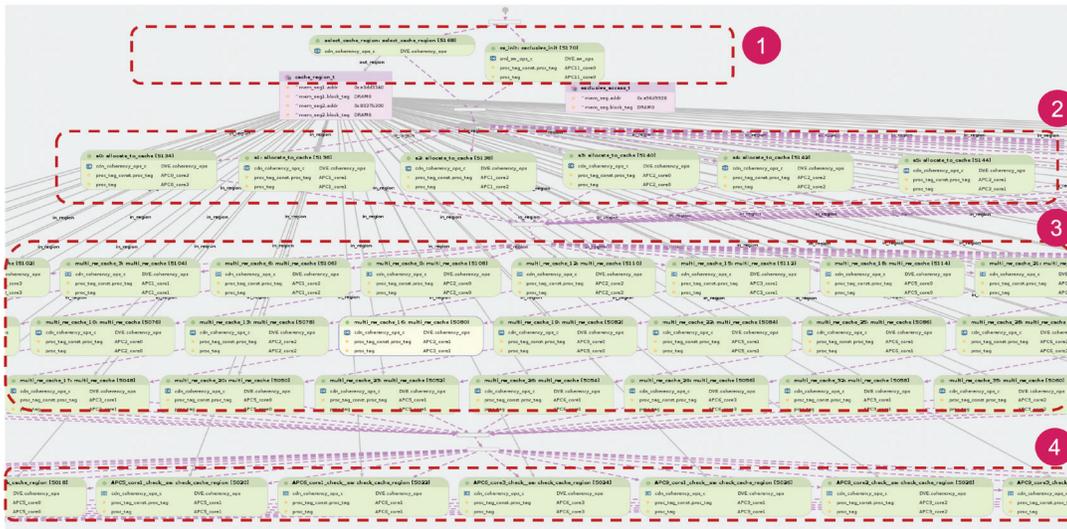


图 8: 伪共享测试

图 9 显示了运行此测试的缓存状态，其中缓存行的各个四字节部分根据访问它们的核心用不同颜色进行标记。

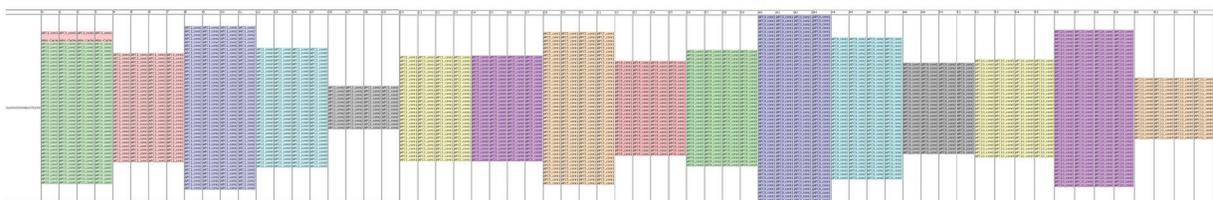


图 9: 伪共享缓存行结果

System VIP 工具和库的强大之处在于，创建这种测试只需要设置目标系统的库，选择伪共享用例，将其拖放到 Perspec 图形界面上并进行解算。手动创建此类内容将非常棘手和耗时。借助 PSS，我们可以在具有任意数量核心的任何系统级芯片上复用这种测试。该测试还可以与其他 System Traffic Libraries (例如 PCIe Library) 一起使用，以创建 I/O 一致性系统测试。

Coherency Library 包括覆盖共享 (covering sharing)、缓存状态 (cache state)、分布式虚拟内存 (DVM) 管理以及其他内容, 如图 10 的流程所示。

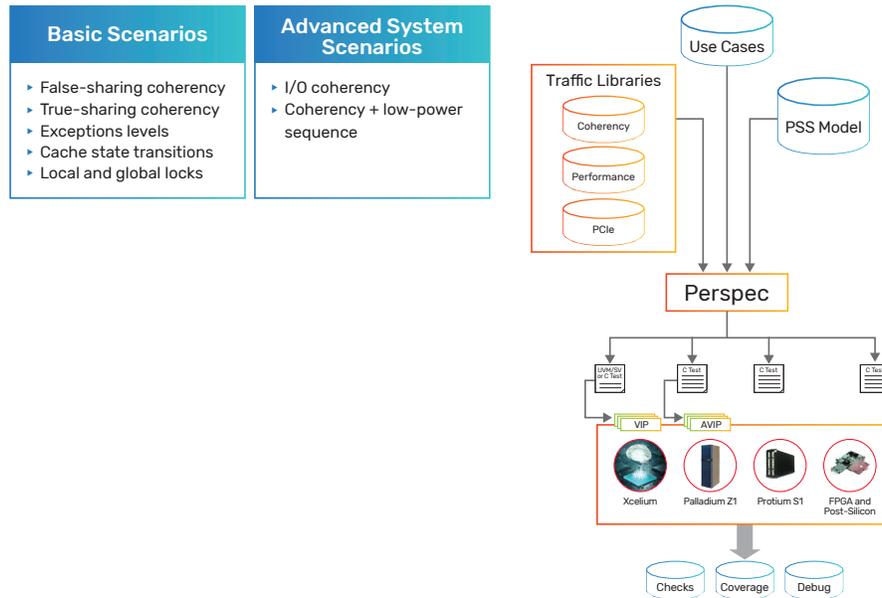


图 10: Coherency Library 内容

System Performance Analyzer

如前所述, System Performance Analyzer 提供全面的图形分析包, 用于可视化和分析跨典型系统级芯片的关键点的性能。片上总线性能为系统级芯片的运行提供了有价值的信息, 通常, 高压场景用于探索系统性能的极限, 在压力下, 存储系统的操作至关重要。

System Verification Scoreboard

除了通过 System Traffic Libraries 支持测试内容之外, System VIP 工具还包含数据完整性检查器。System Verification Scoreboard 是一种高级检查器, 它可以对一致性行为进行建模, 可以在运行时“实时”连接到仿真 VIP, 也可在后处理模式下运行, 以检测在 Palladium 平台上硬件加速运行的数据流。System Verification Scoreboard 为一系列 VIP 提供插件, 包括片上总线和 DDR 内存模型, 从而追踪整个系统级芯片的任务处理, 确保集成不会引入错误。

总结

随着数据共享引擎的数量在不同的应用领域不断扩大, 系统级一致性验证正成为越来越多的 SoC 中的一项普遍活动。Cadence System VIP 支持快速启动及复杂且可复用内容的开发, 从而为系统级芯片带来可预测性。对于此领域的完整解决方案而言, 测试平台自动化、开箱即用的内容和高级检查器至关重要。