

攀划新一代 SoC 验证平台

预制与定制 FPGA 式原型板加入协同仿真(co-emulation and co-simulation)功能, 能够提供高速、高能见度平台, 实现 SoC 的快速、早期验证。

系统芯片(SoC)设计的规模与复杂度不断地攀升。同时, 产品在市场上的存活时间不断地紧缩, 当今的电子市场也对于上市所需的前置时间非常敏感。这些全都加深了 SoC 设计与验证团队的压力。事实上, 现在广为接受的说法就是, 验证占了整体 SoC 开发时间的 70%。所以, 能够降低验证成本、加速验证执行以及在开发初期尽早进行验证的作法, 都是众人瞩目的焦点。

这篇文章首先介绍构成典型 SoC 设计与验证环境的主要因素; 也会说明传统验证解决方案的优缺点, 包括软件仿真、硬件辅助加速与仿真, 还有 FPGA 式原型板的使用。接着会说明创新而且价格合理的做法, 让标准 FPGA 式原型板能够转变成为完善的桌上型硬件仿真器。我们推荐的这种做法是一种典范转移(paradigm shift), 能使既有的硬件仿真(in-circuit emulation)功能自动化, 并且增添全新协同仿真功能, 大幅提高预制与定制设计 FPGA 式原型板的验证效率。

典型的 SoC 设计与验证环境

首先, 让我们来看看典型 SoC 设计与验证环境的前端部分。至少, 这包含某些形式的设计输入、软件仿真方式的功能验证, 还有逻辑综合, 如图 1 所示。甚至, 这种设计环境现在大多也包括 SpringSoft 的 Verdi™ 自动化侦错系统。Verdi 系统让用户能够分析自己软件仿真器¹的结果并执行侦错, 还可以自动使任何门级结果与对应的 RTL 源代码发生关连。

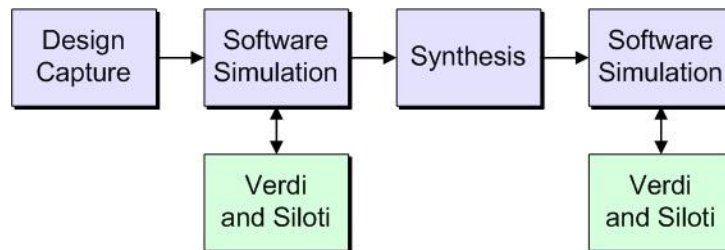


图 1. 简化的 SoC 前端设计与验证环境。

各式各样功能验证的问题之一, 就是要收集与储存的数据量。以软件仿真器为例, 监视大量信号会大幅拖慢仿真速度, 而且冗长的仿真进度又会导致更大量的数据; 所以, 许多设计与验证环境都使用了 SpringSoft 的 Siloti™ 能见度自动增强系统, 来减少记录设计中许多信号数据的负担。Siloti 系统用来找出仿

¹ The Verdi system also allows users to analyze and debug results from hardware accelerators, emulators, and FPGA-based prototype boards as discussed later in this paper.

¹ Verdi 系统也让用户能够分析硬件加速器、仿真器与 FPGA 式原型板的结果并执行侦错, 如本文之稍后讨论。

真进行中最低限度必须记录的信号；然后 Siloti 系统会运用这些信号，自动且迅速的产生所有需要但未记录的信号数据。

软件仿真的主要优势就是对设计有完全的能见度；而主要的缺点就是慢，即使在威力强大、高阶工作站上执行，并采用 Siloti 能见度自动增强技术 – 当今大型 SoC 设计的软件仿真只能勉强达到几 Hz 的仿真速度 (也就是说，每秒钟只能有几次设计的实时主系统频率周期)。这表示，软件仿真通常仅适用于设计的一小部分，或者适用于全芯片设计的数十个频率周期而已。但是，完全验证现代化 SoC 需要成千上万甚至数百万频率周期，这时就需要某种形式的硬件辅助验证，如图 2 所示。

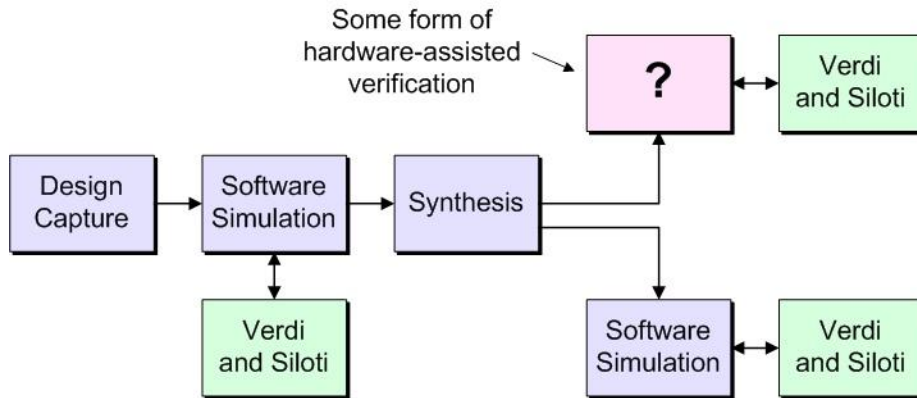


图 2. 简化的 SoC 前端设计与验证环境，增添某种形式的硬件辅助验证。

传统的硬件辅助验证解决方案

各种硬件辅助验证解决方案具备不同的功能、优点与缺点。也有许多方法可以运用不同的系统，来解决不同类型的问题；包括硬件仿真(in-circuit emulation)、交易级协同仿真以及 HDL 协同仿真。

一般而言，传统的硬件辅助验证解决方案仅包括硬件加速器与/或仿真器。FPGA 式原型板通常不被视为可行的替代方案，因为缺乏与工作站连结的能力，而且无法提供足够的设计能见度以供侦错使用。

传统的硬件加速与/或仿真系统都是专属系统，利用定制设计的芯片、或标准 FPGA，来建构特殊目的的系统。这些系统的目的是要尽可能如同软件仿真器一般运作，包括能见度与侦错功能等因素。这些系统运用能够驾驭其定制芯片或架构的特殊软件，提供大型容量与相当快速的编译时间，让设计能够映像到硬件上；他们也提供相当优良的设计能见度(能见度与可控制度)。但是，这些系统非常昂贵，难以让许多使用者、项目与团队广泛的运用。再者，一旦采用这种系统之后，就很难升级到新一代的系统；除了新版定制设计芯片与系统需要耗费时间来开发之外，还必须考虑高昂的过渡成本等其他因素。

转移到 FPGA 式原型板

作为硬件加速器与仿真器的替代方案，许多设计业者运用可以现成购买的，或者由 SoC 验证团队定制设计 FPGA 式原型板。如图 3 的例子，设计在工作站上编译(综合)后；映射、布局与绕线；然后将结果的 FPGA 配置文件(或者是系统包含许多 FPGA 时的文件)下载到原型板上。典型的运用模型适合于 SoC 设计(或部分设计模块)验证时使用硬件仿真(in-circuit emulation)的方式；也就是说，如图 3 所示，运用实际输入

/输出 (I/O) 信号来驱动。除了由外部系统所驱动 – 和带动之外，也可以掌握实际的 I/O 信号以供逻辑分析器等来进行后续分析。

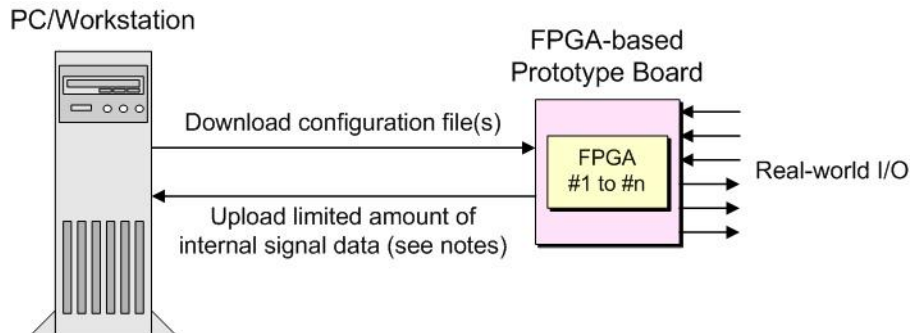


图 3. 以硬件仿真(in-circuit emulation)方式运作的传统 FPGA 式原型板环境的高阶呈现。

传统 FPGA 式原型板的主要优势就是高效能，而且相对而言比较便宜，所以能够让许多使用者、项目与团队广泛的运用。此外，这些原型板能够运用最新一代的 FPGA 技术，让使用者能够快速而轻松地转移到新一代原型板。而其主要缺点就是难以设定，而且无法与工作站连结以支持协同仿真 (co-emulation and/or co-simulation)。再加上只能提供极有限的设计能见度，所以缺乏精密的侦错功能。

强化传统 FPGA 式原型板

传统 FPGA 式原型板通常都配备 J-Connector，希望让原型板能够连接到外埠装置。本文提出的作法就是利用这个 J-Connector，建立能与其连接的特殊适配卡，以扮演主机工作站与 FPGA 式原型板之间的桥梁，如图 4 所示。

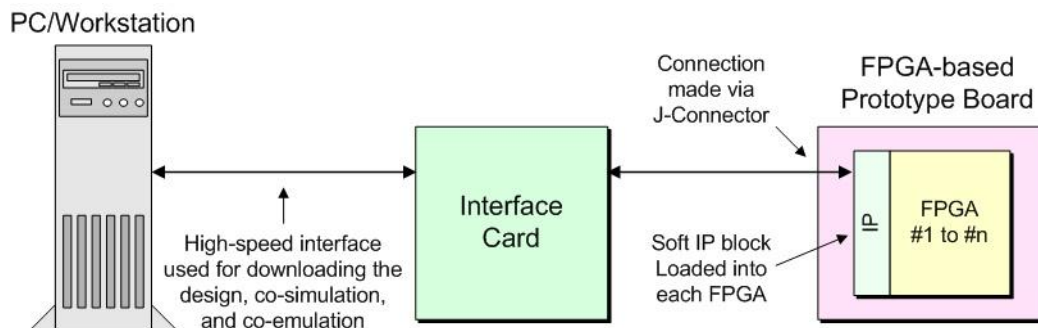


图 4. 新增适配卡与某些软 IP (soft IP)。

FPGA 式原型板与工作站之间的所有通讯都可以运用某些专属总线而建置起来，提供仿真与协同仿真所需的高性能。由于这种互连技术，使用者拥有可架构和运用不同原型板以搭配同一 J-Connector 的绝佳弹性。这种能力也表示，使用者能够快速而且轻易地转移到更快速和更庞大的 FPGA 式原型板，可以完全免除传统硬件辅助验证解决方案相关的限制。

接下来看看图 4 所示的软 IP 模块 (soft IP)。这些 IP 模块可下载至原型板上的各 FPGA，然后可用来控制和监视 FPGA 式原型板与工作站之间的数据流动。因操作模式 (硬件仿真、协同仿真) 的不同，可以自动插入适当且必要的「特色」IP 模块。这些 IP 模块可以编译到设计中，用来监控用户指定的任意信号。如此一来，就可以掌握和分析来自数千信号仿真数百万频率周期的数据。

除了互连技术之外，本文的作法也需要如图 5 所示在工作站上执行的特殊软件来执行许多功能，例如自动建立包含多颗 FPGA 的原型板。这个建立流程涵盖读取设计的 RTL 源代码 (以 VHDL、Verilog、SystemVerilog 或混合式语言)、分析 FPGA 内部与外部的内存、转换 SoC 频率已产生无 hold-time 问题的设计在 FPGA 中使用、分割完整 SoC 设计的 RTL 以便放入多颗 FPGA，以及调整 RTL 来搭配软 IP 模块与要监控的信号。

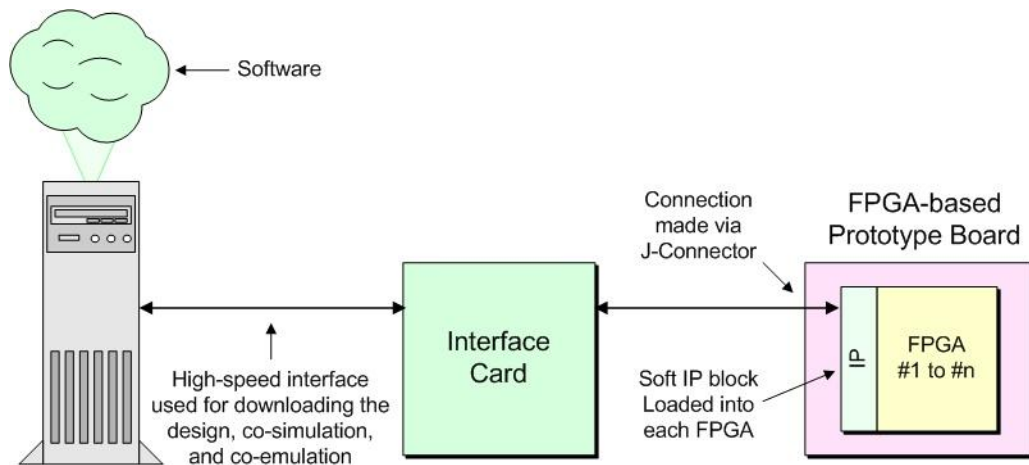


图 5. 增添在工作站上执行的特殊软件。

在验证途中，软件可以依据建立时指定的操作模式 (硬件仿真、协同仿真)，控制和管理 FPGA 式原型板与工作站之间的通讯与数据流动。另外，能快速修改所监控信号的能力也很重要，这样才能够快速且轻易地新增监控的信号，不必重复编译整个设计。最后，就是原型板上 FPGA 内部状态的能见度，这包括在特定时间或以逐一频率周期为基准所提供的寄存器 (registers) 与内存输出，在进阶侦错时非常实用。

设计原型的「桌上型」验证

为了更完全了解因本文所讨论互连与软件自动化技术而实现的各种功能与使用模式，让我们来看一些范例情境。最简单的例子就是图 6 所示的纯粹硬件仿真应用。在桌上型工作站上执行的软件会自动分割设计，并准备原型板。在建立过程中，会在设计中置入所有必要的探测点，以便在执行时掌握特定信号数据。

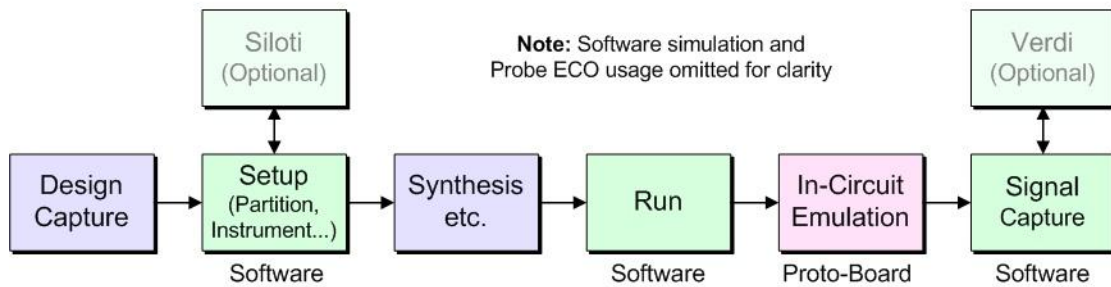
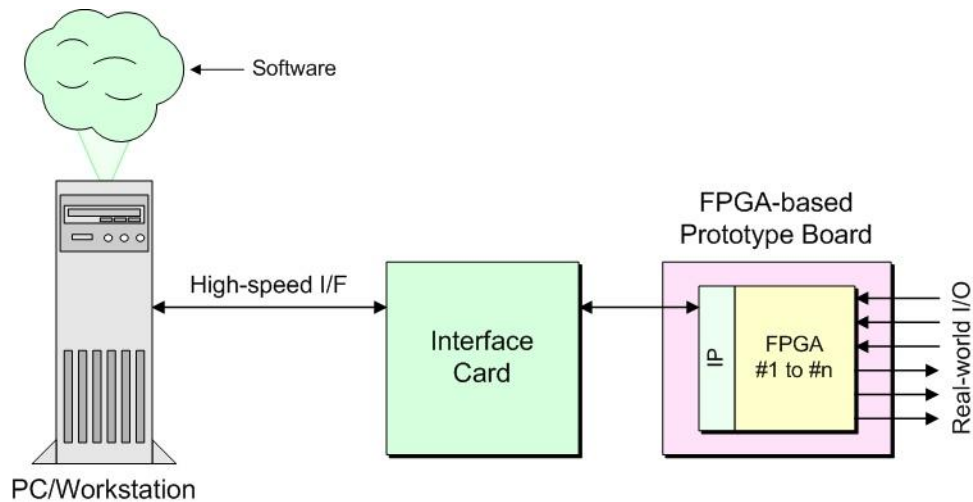


图 6. 硬件仿真情境的范例。

此外，使用者也可以选择性的运用 Siloti 能见度自动增强系统，来帮助判断需要观察的最少信号量。以及运用 Verdi 自动化侦错系统来分析 FPGA 式硬件原型板的结果，并执行侦错。由于设立软件会自动将任何门级信号关连至对应的 RTL 信号，Verdi 系统搭配 RTL 源代码可以加速侦错的进度。

交易级协同仿真是可以大幅加速验证流程，其速度比仅使用软件仿真要快上数百或数千倍，常见的状况是 testbench (或许是设计的一部份) 常驻在工作站上，而大量 (或全部) 设计被加载到原型板。工作站上执行的软件会自动分割设计，自动建立原型板，然后插入适当的协同仿真基础架构 (例如 SCEMI 式处理装置)。在前述情境中，建立时，所有必要的探测点都会置入设计中，以便在执行时能够掌握特定的信号数据。或者，协同仿真可以暂停，也可以设置软件在特定时间或按照逐一频率周期，将 FPGA 内部状态的完全能见度输出，如图 7 所示。

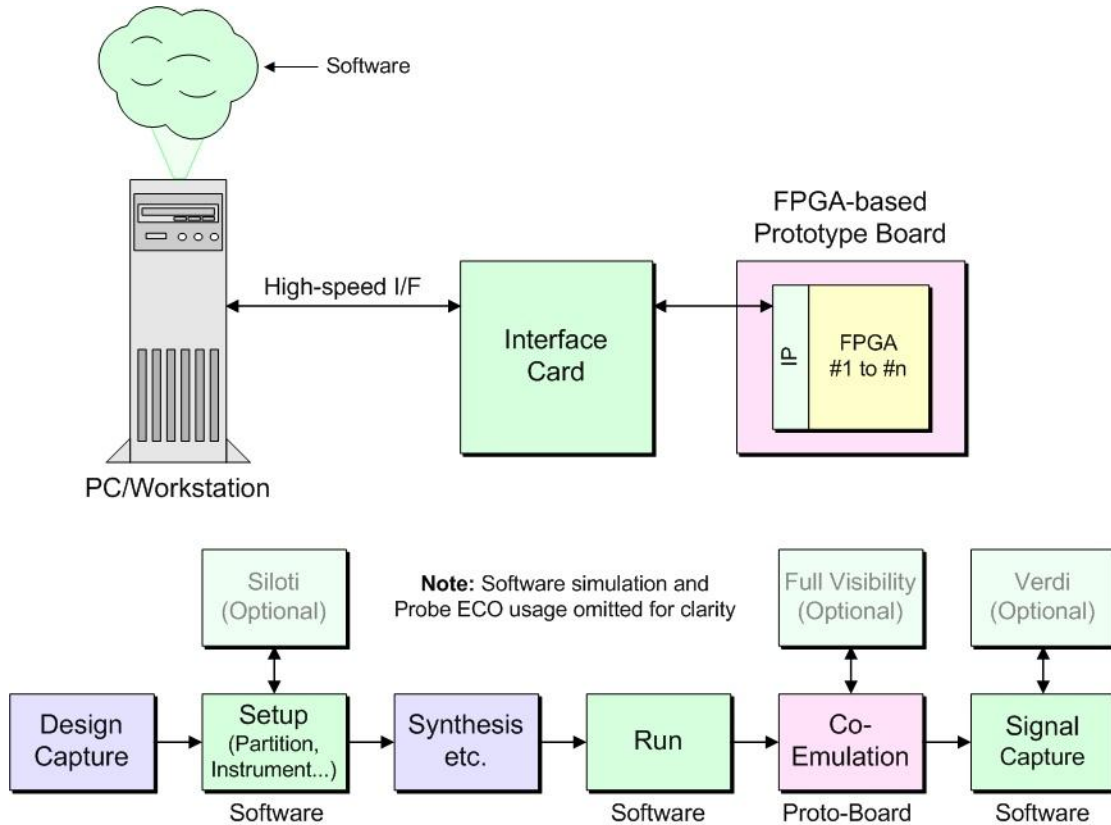


图 7. 交易级协同仿真情境的范例。

最后，让我们看看 HDL 协同仿真，这种加速验证流程，其速度比单仅使用软件仿真要快上数十倍。同样地，testbench (或许是设计的一部份) 常驻在工作站上，而大量 (或全部) 设计被加载到原型板。在这个情境中，工作站上执行的软件会自动分割设计，建立原型板，和产生可供软件仿真器连接的包装器(wrapper)。执行时，软件会控制原型板与 ModelSim、NC 或 VCS 等业界标准软件仿真器之间的协同仿真互动。如同前面几个情境中的讨论一般，透过软件与 FPGA 内部技术的组合，可在特定时间或按照逐一频率周期，将原型板上 FPGA 内部状态的能见度，包括寄存器与内存输出，如图 8 所示。

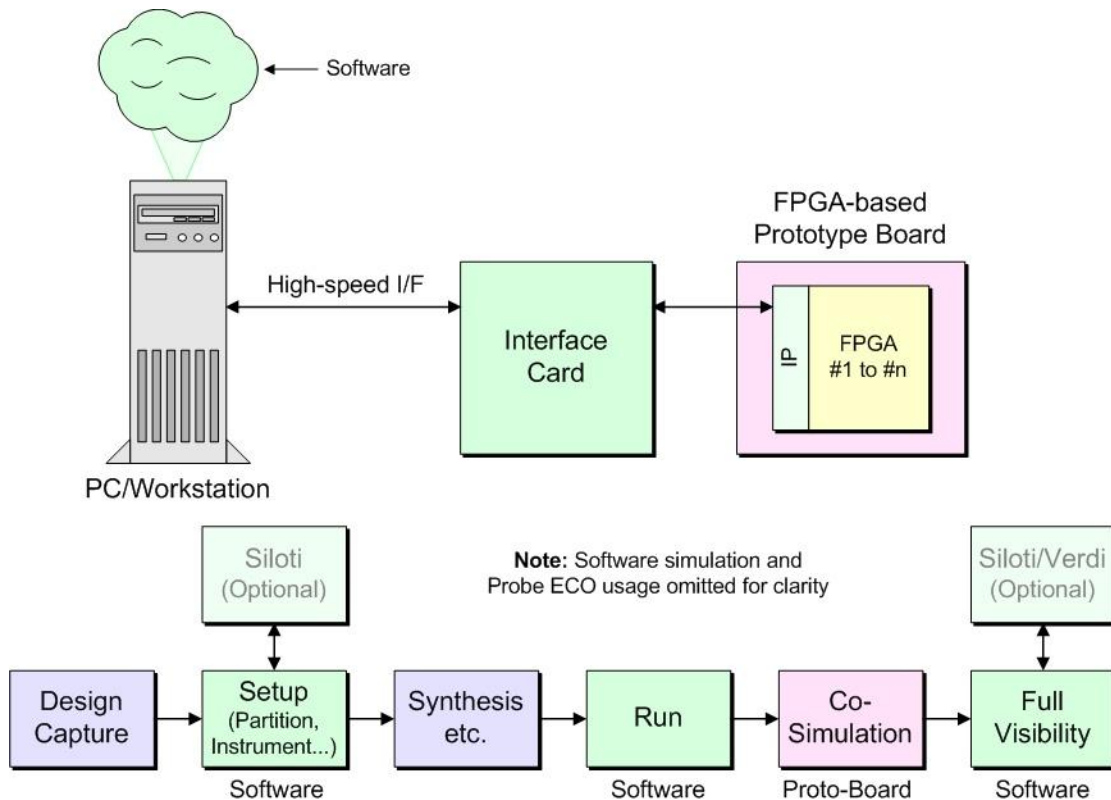


图 8. HDL 协同仿真情境的范例。

新一代原型验证平台的优点

产品在市场上的存活时间及上市前置时间不断缩短造成了当今 SoC 设计与验证团队的莫大压力。软件仿真提供对设计内部的 100%能见度，但是只适合于设计的一小部分，或者整个设计的数十个频率周期而已。然而，完全验证现代 SoC 需要耗费成千上万甚至百万频率周期，所以需要某种形式的硬件辅助验证。各式各样的硬件辅助验证解决方案具备不同的功能、优点与弱点。表 1 提供了各种硬件辅助验证技术的大略比较。

传统硬件加速器与仿真器提供大容量、相对快速的编译时间，以及相当良好的设计能见度。然而，这些系统太昂贵而难以广泛运用，而且因为转移成本高昂，也难以与时俱进地升级至新一代解决方案。比较上，传统 FPGA 式原型板提供高效能，而且比较便宜；但是，缺乏设计能见度与精密的侦错功能，无法克服当代 SoC 的复杂度，而且通常仅以硬件仿真模式来运用。

| | Accelerators and/or Emulators | FPGA-based Prototype Board (Augmented) | FPGA-based Prototype Board (Conventional) |
|--|-------------------------------|--|---|
| Cost | High | Medium | Low |
| Speed (Runtime)¹ | Slow-to-Medium | High | High |
| Speed (Compilation)² | Medium-to-High | Medium | Slow |
| Capacity³ | High | Medium | Low |
| Visibility | Good | Good | Limited |
| Emulation | Yes | Yes | Yes |
| CoEmulation | Yes | Yes | No |
| Co-Simulation | Yes | Yes | No |
| Ease of Migration⁴ | Poor | Good | Good |

¹Runtime speed varies depending on the operational mode. The comparisons in this table are based on in-circuit emulation.

²Compilation speed depends on whether or not special timing optimizations are used. The comparisons in this table are based on in-circuit emulation without special timing optimization.

³In this context, "High Capacity" refers to a billion or more gates, "Medium Capacity" is around 100M+ gates using multiple boards, and "Low Capacity" is around 10M+ gates on a single board. Although conventional FPGA-based prototype boards can be connected using fixed, hardwired techniques to increase capacity, this does not provide the same capabilities as the programmable methods employed by conventional accelerators/emulators and by the proposed solution.

⁴Ease of migration refers to the users' ability to migrate to the next generation of the verification solution; taking advantage of the latest generation of FPGA technology, for example.

表 1. 硬件辅助验证技术的比较。

为了解决这些问题，我们建议一种 SoC 验证的新方法，将传统 FPGA 式原型板转变成桌上型加速器/仿真器。这需要创新的互连技术，能够提供对预制与定制设计板的通用连结；还有特殊化的软件自动化，在工作站上以硬件仿真、协同仿真模式执行，并具备极高的设计能见度。

运用强化上述功能的 FPGA 式原型板，SoC 开发人员可快速验证个别模块 (包括内部与第三方 IP 模块) 和设计模块是否正常运作，甚至于整个 SoC 设计的验证。这个新一代原型验证平台可提高 FPGA 式原型版的投资报酬率及生产力，提供更高验证效率与弹性，并能快速转移到配备最新 FPGA 技术的原型板。

供稿：思源科技公司