

LabVIEW 系统设计软件 对数字开发的影响



概览

长期以来的预测趋势揭示了整个系统设计流程中的设计和测试趋于统一，这两个先前独立的功能将被集成在一起。集成功能的明显优势在于缩短了投入市场的时间并获得更好的整体质量，而这些优势都归功于在创建设计的同时集成了测试定义和实现。在系统设计过程中，从仿真到实现以及最终系统部署，都可以对这些早期测试平台进行重用。

为了真正实现设计和测试统一，尤其对于类似 RF 通信等复杂功能，需要在测试和实现的设计生命周期的所有阶段都能够有效执行所选的系统设计软件和语言。在这之前，用于系统仿真或设计与用于系统实现的工具及技术存在很大不同。此外用于设计和实现的工具和语言通常不同于测试中使用的工具。这将导致不同功能团队使用不同的工具，将增加交流的复杂程度并降低设计和测试中可重用代码的流动性。这些因素都是实现设计和测试统一的主要障碍，因此理想的系统设计软件需要提供单一语言用于仿真、实现和测试，并在所有设计阶段和功能间最大化代码重用。

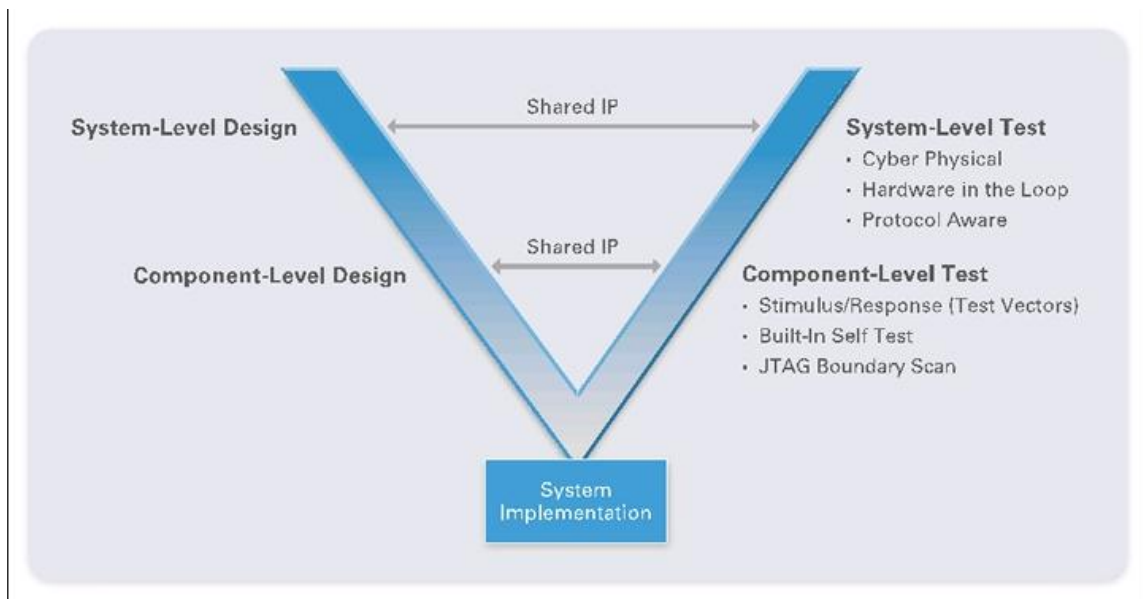
目录

1. [减少限制的传统方法](#)
2. [LabVIEW 所提供的单工具方法](#)
3. [LabVIEW 系统设计和 RF 通信设计](#)
4. [集成设计、开发和部署](#)

减少限制的传统方法

通常能够跨越设计过程各个阶段和功能的工具都尝试着减少每个阶段和功能之间的限制，而不是创建可跨越所有阶段和功能的单一环境和语言。例如，在新兴的 RF 通信标准开发中，通信系统设计专家可能会使用纯数学算法对通信流进行建模和仿真。为了测试模型，设计人员可能会创建自定义的测试平台或者重用合规套件对工业标准协议进行测试。

当设计功能达到要求时，从设计到实现的传统转换方法会将算法交给其它组，并通过手动方式将数学算法转换为程序实现，通常将根据实现的执行要求使用 ANSI C 或 HDL 语言。



[\[+\] 放大图片](#)

图 1. “V”框图通常用于说明设计、实现和测试的理想流程。将通用系统设计语言用于跨越整个“V”流程可以最大化技能和算法重用，同时可以最小化转换错误和查找修正周期数。

此外需要转换测试平台本身，该转换步骤需要具有不同技能的其他组实现，并且任何转换中丢失的部分都可能导致新缺陷或测试覆盖度降低。即使初始转换阶段可能不够复杂，但该问题会随着重新定义设计或发现程序实现缺陷等变得更加复杂。设计算法和程序实现的分离将导致“查找并修复”开发周期变慢。

为了缓解部分问题，一些工具可提供自动代码生成步骤以便帮助仿真至 ANSI C 或 HDL 的转换。虽然该过程有利于将算法设计迁移至处理器或现场可编程门阵列(FPGA)，但仍需要由熟悉 ANSI C 或 HDL 的开发人员进行部署和调试设计。可能不会存在完美的原始设计，而自动代码生成也会存在缺陷。

LabVIEW 所提供的单工具方法

替代方法将利用到 NI LabVIEW 系统设计软件，通信设计人员可以使用 LabVIEW 对通信流进行建模，然后实现测试平台。当满足设计和测试要求后，系统设计人员仅需将设计算法的终端直接重新设定为处理器或 FPGA 以用于具体程序实现。从最初的实验性设计到最终程序实现，系统设计人员可以使用相同的环境、算法以及调试和测试方法。该方式可以最大化技能和算法重用，同时最小化转换错误和查找修正周期数。

既然在设计、实现和测试中重用相同算法而无需独立的代码生成步骤可提供各种优势，那么为什么其他系统设计软件不直接使用重用的方法？这主要是由于

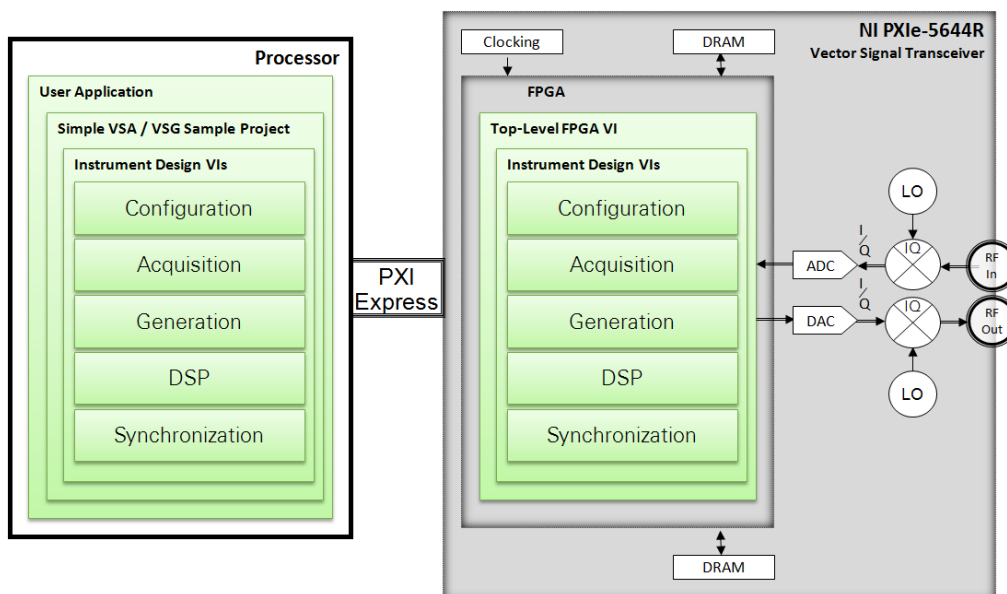
一些历史原因导致了该情况的出现。大多数系统设计软件最初用于仿真实空，并且该仿真已为系统行为的时域验证进行优化，然后向程序实现方向发展。LabVIEW 核心为系统实现，并朝设计和仿真方向发展。其编程语言、环境以及更重要的 IP/算法模块是为了在处理器和 FPGA 上以实时速率进行编译和执行而设计的。

除了实现算法重用，LabVIEW 的程序实现传统很大程度上对部分仿真-时间活动进行了加速。例如，LabVIEW 可将台式机算法完整编译为专用处理器的机器代码。对于纯功能 DSP 类型算法，系统设计人员将以相对于连续时域仿真方法的更快速度对算法进行实现和调试。与用于终端 FPGA 的设计仿真相比，该优势尤其明显。LabVIEW 可支持位及周期精度仿真的预期形式。在特定情况下功能测试通常能满足需求，但 LabVIEW 可将速度提高多个数量级，主要原因为功能代码已为本地功能执行进行了完全编译和优化，而不再仅仅是“仿真”。相对于较慢的仿真算法，在仿真中重用快速的实现算法更具实际用途，因此与其他系统设计软件相比，LabVIEW 优势更为明显。

LabVIEW 系统设计和 RF 通信设计

在 RF 的设计和测试趋于统一范例中，相对于普遍统一的趋势，通信设计将导致部分特有的复杂性。最主要情况为，当测试 RF 接收器时必须创建一个发送器，而测试发送器也必须创建一个接收器。大多数时候该测试的信号和测试特性必须超出设计本身规范。此外 RF 通信方法和标准的变更频率十分迅速，这也对测试本身的灵活性和速度提出了要求。因此用于 RF 的理想测试仪器必须允许快速重用发送器和接收器数字信号处理（DSP）算法，并且具备高性能和灵活性。

该类测试的传统方法将会创建特定功能的仪器，以用于测量和测试特定的通信类型。如需完全覆盖设计和测试，不仅需要在设计流程中考虑测试流程、测试平台和附加因素，而且在理想状态下设计和测试需要直接使用实际硬件和信号而不仅是仿真。在 LabVIEW 和 NI PXIe-5644R 矢量信号收发器范例中，当在台式机中定义通信流的 DSP 基本构件并执行预期功能后，设计人员可以将算法终端从台式机设计环境更改为运行 FPGA 的矢量信号收发器。



[\[+\] 放大图片](#)

图 2. NI PXIe-5644R 矢量信号收发器的架构可允许在主机或设备固件中仿真和部署用户可编辑的 DSP 块。同时也可以对其他输入/输出以及内存接口进行修改以作为算法设计的补充。

该迁移的主要因素以及从算法设计到最终部署（无论是设计本身或测试）的主要障碍是能否正确集成实际定时，更重要的是，硬件 I/O 和信号校准。对于设计和测试功能，DSP 算法设计人员通常不同于实现包含 I/O 集成在内的设备固件的团队。通过矢量信号收发器和 LabVIEW RIO 可编程设备，NI 将提供可灵活编程的 RF 硬件平台，该平台将包含高度优化的 IP 块。该程序块可管理与信号校准集成的高速 A/D 和 D/A 的常规复杂度，以及至主机处理器和高速板载内存的 DMA 数据流。该矢量信号收发器中的 IP 代表了三个主要特性。首先可以在台式机中仿真功能以便用于算法设计。其次源代码可用于参考或修改。如之前所述，最后可支持无缝迁移至实际实时执行系统。

集成设计、开发和部署

为了真正实现从算法到部署的紧密系统设计流程，必须具备完整的软件/硬件视角。软件必须包含正确的定时、I/O 和 DSP 算法 IP。该 IP 必须能支持仿真执行，而且需要快速执行功能调试并支持快速设计周期，并且最终能将测试和设计代码从台式机环境迁移至可编程的部署硬件。

用户开发的算法和测试均属于重要知识产权(IP)，而任何系统设计工具的主要任务在于最大化开发效率，允许用户在原设计和最终实现存在较小差别时对相同 IP 进行重用。仅当系统设计软件最初为了程序实现采用了开放和可重用的构建

模块，并且该基本构建模块可完全匹配用于迭代式设计过程的可重配置硬件时，才可能实现上述方式。而结合 LabVIEW 系统设计软件和支持 LabVIEW RIO 的矢量信号收发器可以达到这一目标。

