

深入了解 LabVIEW FPGA

NI 通过 LabVIEW FPGA 模块和可重复配置 I/O(RIO)硬件设备，为测量和控制系统中整合 FPGA 技术的灵活性提供了直观且现成可用的解决方案。您可以使用 LabVIEW 图形化编程定义 FPGA 芯片上的逻辑功能，您不需要任何的有关底层硬件描述语言(HDLs)的知识，如 VHDL 或是 Verilog，也不需要了解板卡级硬件设计，就可以将 FPGA 芯片嵌入到 NI 可重复配置 I/O 系列硬件目标当中。另外，LabVIEW 还可以让您轻松地集成图象采集/分析、运动控制，以及 CAN 和 RS232 等工业通信功能。

LabVIEW FPGA 最适合于那些经常需要定制硬件的应用，包括：灵活的编码器输入、PWM 信号 I/O、超高速控制、定制计数器、数字协议仿真、离散控制，以及在单个设备中定制模拟和数字混合 I/O。

目录

- [简介](#)
- [NI RIO 硬件设备](#)
- [使用 LabVIEW FPGA 创建定制功能](#)
- [FPGA 开发流程](#)
- [图形化编程的性能优势](#)
- [典型应用](#)
- [总结](#)

简介

您可以使用 LabVIEW FPGA 模块，通过图形化编程对 NI RIO 设备上的现场可编程逻辑阵列(FPGA)进行配置。LabVIEW FPGA 模块和 RIO 设备共同提供了一个灵活的平台，能够创建在从前只能使用定制设计的硬件设备才能够完成的复杂测量和控制系统。

FPGA 是一种由许多待配置的逻辑门所组成的芯片。与功能由厂商确定的特定应用集成电路芯片(ASIC)不同，您可以根据不同应用的需要对 FPGA 进行反复配置。在一些开发和制造 ASIC 芯片的成本不可接受，或是在投入应用之后需要重新对硬件进行配置的应用场合，FPGA 被广泛地采用，这是由于 FPGA 可以实现硬件上执行定制算法，而且还可以提供精确的定时和同步，快速的决策以及并行任务的同时执行。现在，FPGA 在仪器、消费电子、汽车、飞机、复印机以及特殊应用计算机硬件等各类设备中都有应用。FPGA 也经常用于测量和控制类的产品，但是这些系统的终端用户往往不愿意开发自己基于 FPGA 的系统。

之前很长一段时间，对 FPGA 编程的工作只有对 VHDL 或其他底层设计工具有着深入了解的工程师才可以胜任，但掌握这些工具需要很长时间的学习和积累。而使用 LabVIEW FPGA 模块，更多的工程师可以使用 LabVIEW 图形化开发环境对 FPGA 的逻辑功能进行定义。而不再需要有关其他设计工具的知识，就可以对 RIO 设备上的 FPGA 逻辑功能进行配置。测量和控制工程师可以专注于他们所擅长的测试和控制应用，而不必考虑如何在芯片的各个单元上实现逻辑功能。

另外，LabVIEW 图形化编程的并行性非常适合于 FPGA 的并行架构，可以实现同步或异步模式的并行任务。最后，由于 LabVIEW FPGA 模块在硬件中执行逻辑功能，因此您的系统可以快速而确定地生成

同步的模拟和数字信号，并进行处理。图 1 向您展示通过 LabVIEW FPGA 模块可以进行配置的一些 NI RIO 设备。

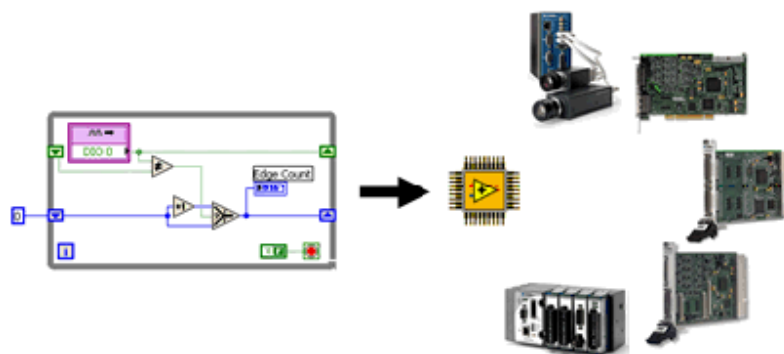


图 1 LabVIEW FPGA VI 程序框图和 RIO 硬件平台

NI RIO 硬件设备

LabVIEW FPGA 模块可以提供 LabVIEW 软件、商业化现成可用的(COTS) FPGA 硬件架构及其周边 I/O 组件之间的紧密整合，从而为测量和控制工程师提供更多的价值。LabVIEW FPGA 正在继续扩展到 NI 的许多平台上。

随着 RIO 技术在 PCI、PXI、便携式视觉系统以及 CompactRIO 平台上的实现，工程师们在构建测量和控制系统的時候，可以利用 COTS 平台，并享受 FPGA 所带来的高性能，灵活性和可定制化等优势。

NI PCI 和 PXI 的 R 系列插入式数据采集设备可以在一台设备上为高性能、用户可配置定时和同步的应用提供模拟和数字数据采集和控制，并进行板上决策。使用这些设备，您可以对您的 NI PXI 或 PCI 系统进行扩展，包括高速离散和模拟控制，定制通信协议，与定制传感器之间的接口，定制数据采集设备，以及精确定时和同步等功能。

NI CompactRIO 是一个以 RIO 技术为核心的平台，它可以提供一个小型的，工业级坚固的模块化平台，为您提供高性能的 I/O，以及对于现成可用的嵌入式系统来说前所未有的灵活性。您可以使用 NI CompactRIO 构建一个嵌入式系统，如车载数据采集，移动 NVH 测试和嵌入式机械控制系统等。另外，CompactRIO 提供了一个灵活的架构，可以搭载各类现成可用的处理器(如 Freescale PowerPCs)，实时操作系统(如 Wind River VxWorks)，使用 Xilinx FPGA 进行定制的数字逻辑，以及 NI 和其他厂商生产的 I/O 设备等组件。您还可以使用 CompactRIO 模块开发组件，通过这样一个框架结构开发您自己的定制 CompactRIO 模块。NI CompactRIO 系统的坚固性是经过工业级认证的，可以在-40 至 70 °C 的环境中进行工作，并可以承受强度超过 50 g 的震动。

NI Compact Vision System(CVS)是坚固的机器视觉系统工具，可以耐受在机器人研究，自动化测试，和工业监测系统中经常出现的严酷环境。NI CVS-145x 设备为分布式机器视觉系统提供了前所未有的 I/O 能力和网络连接功能。NI CVS-145x 系统使用了 IEEE 1394(火线)技术，可以兼容四十多种功能、性能和价格各异的摄像机。NI CVS-1455 和 NI CVS-1456 设备包含了可重复配置的 FPGA，您可以通过 FPGA 在您的机器视觉应用中实现定制计数器、定时或马达控制功能。

NI PXI-6653 和 NI PXI-6652 定时同步控制模块使用了 PXI 的触发总线，星形触发，以及系统参考时钟等特性，实现了多设备和多机箱间的同步。LabVIEW FPGA 模块以 PXI-6653/2 模块搭载的 FPGA 为目标对象，可以开发用户自定义的应用，这在之前使用现成可用的硬件设备是不可能实现的。这些定制应用包括定时和同步技术，定制触发以及高速模拟计数器。

使用 LabVIEW FPGA 创建定制功能

您可以使用 LabVIEW FPGA 模块，在 LabVIEW 中编程对 FPGA 设备的功能进行配置。由于许多测量和控制系统的程序都是使用 LabVIEW 进行编程的，因此使用同一个 LabVIEW 图形化开发环境对 FPGA 进行编程就显得更加简单。很多情况下，在 LabVIEW 中对 FPGA 进行编程比编制 Windows 或实时系统程序更加简便，因为您不需要了解 LabVIEW 的调度或线程切换。

一旦您选择 NI RIO 设备上的 FPGA 作为目标对象，LabVIEW 将只显示 FPGA 可以实现的功能 (图 2)，更加方便您使用 LabVIEW 对 FPGA 进行编程。LabVIEW FPGA 模块函数面板包括典型的 LabVIEW 结构和函数，如 While 循环，For 循环，条件结构，顺序结构以及 LabVIEW FPGA 针对数学、信号生成和分析、线性与非线性控制、比较逻辑、数组和簇操作、并发、模拟和数字 I/O，以及定时等方面的专用函数。您可以使用这些函数进行组合，定义逻辑功能，并将其嵌入到您的 NI RIO 设备当中。

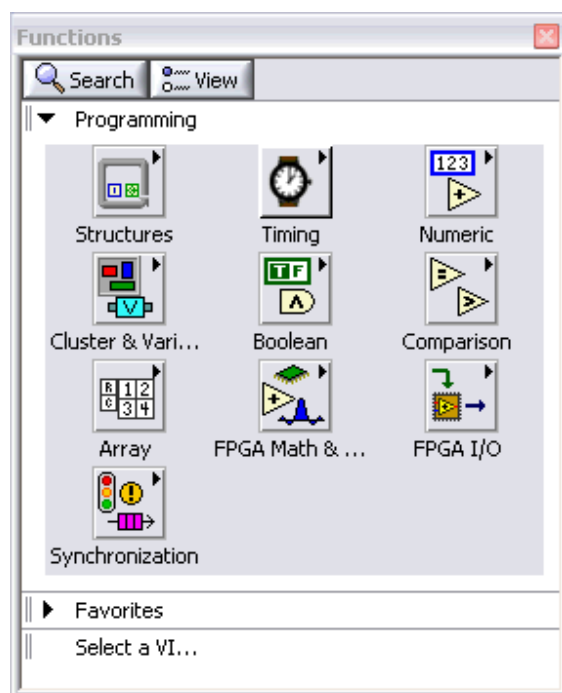


图 2. LabVIEW FPGA 函数面板

您的 LabVIEW 程序框图是在硬件中实现的，这使您能够直接通过 RIO 硬件 I/O 对设备进行控制。您可以通过多种方式对 I/O 信号进行分析和操作，这在固定 I/O 硬件平台上是不可能实现的。图 3 是一个简单的上升沿计数器的程序框图 (这是 LabVIEW FPGA 模块所附带的众多范例之一)。您只需要简单地依照数据流进行分析，就能够理解计数器的功能。我们连续地从一根数字线上读取信号，并比较两个循环间数字线状态的差异。当数字线状态有所改变时，就使计数器加一。计数器的大小是由计数器显示控件的 I32 数据类型决定的，因此这是一个 32 位的计数器。

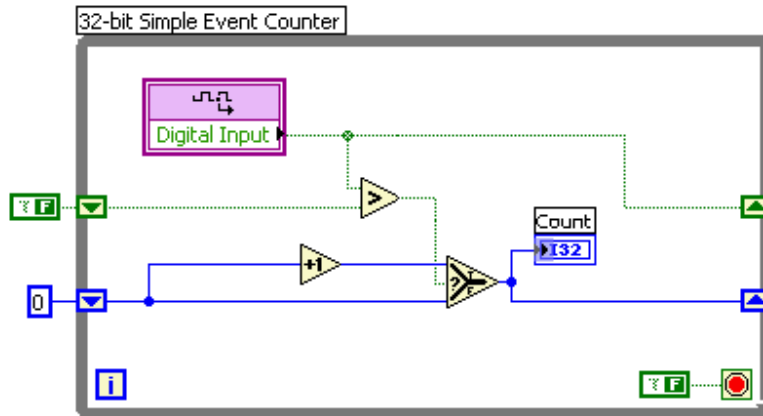


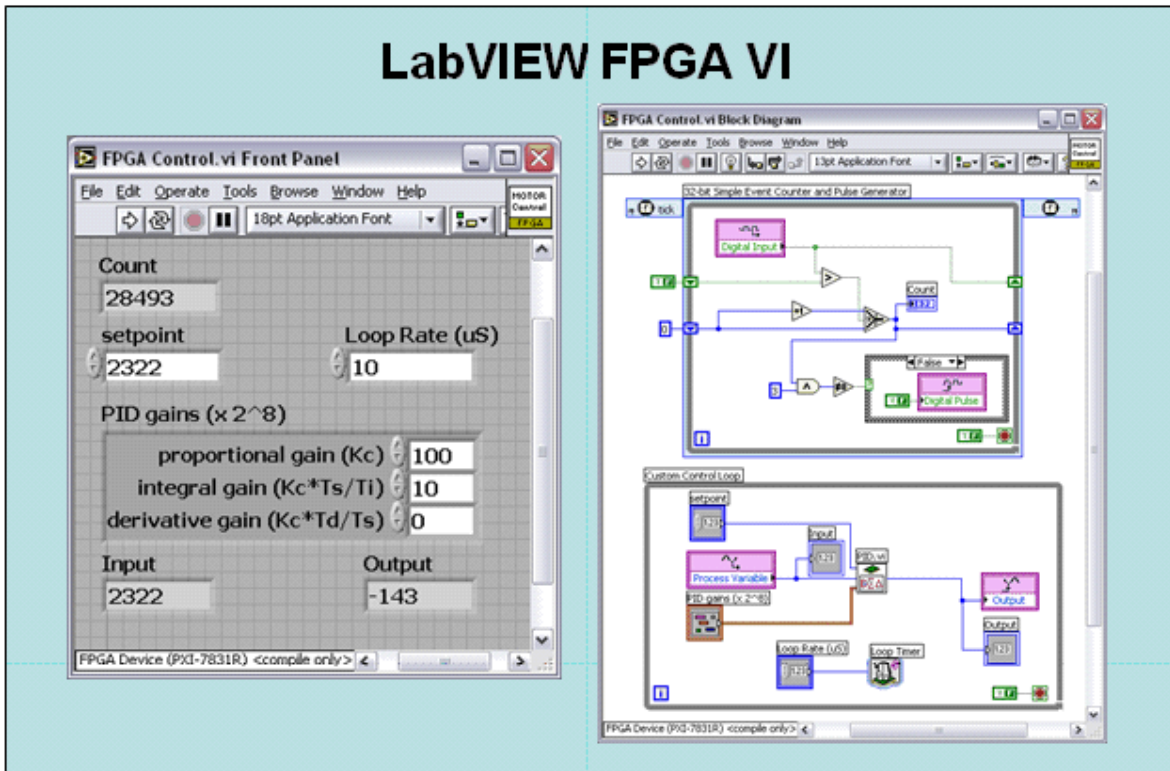
图 3. LabVIEW FPGA 范例程序—32 位计数器

在一个典型的数据采集设备上，计数器逻辑一般是在 ASIC 芯片 (如 DAQ-STC II) 上实现的，您可以使用 NI-DAQmx VI 编写您的应用程序。另外，您也可以使用 LabVIEW FPGA 模块在 LabVIEW 中实现您自己的计数器，并配置自己的 FPGA 的计数器“芯片”。而在 LabVIEW 或 LabVIEW 实时模块中实现这样的计数器就非常的困难，因为这两种解决方案都需要软件调用或上层操作系统。由于 LabVIEW FPGA 使用模块流程图可以直接将您的应用程序在硬件中实现，使得计数器的性能可以与数据采集设备上的计数器相媲美。

这个范例只介绍了一个简单的事件计数器，但是只需要修改 LabVIEW 的代码就可以对计数器进行定制，或在 NI RIO 设备上创建附加功能，如图 3 所示。与传统的 PC 处理器不同，FPGA 是并行化的处理器，向应用中添加附加的循环并不会影响其它独立循环的性能。

在图 4 中，代码已经经过了更新，可以在每四次记数之后生成一个数字脉冲，并执行一个模拟控制循环。在顶部的循环中，代码在 LabVIEW 中被封装成为定时循环中的一个特殊形式—单周期定时循环。单周期定时循环会在 FPGA 得每一个时钟周期执行一次，这意味着计数器的时钟速率在编译的时候就已经由 FPGA 的时钟速率决定了，默认值为 40MHz 或是 25 纳秒。相比较于 LabVIEW 代码定义的传统 While 循环，单周期定时循环所生成的代码更加节省空间，使用的 FPGA 门电路也更少。但由于存在一些单周期定时循环所不支持的结构，如模拟 I/O，我们还是需要传统的 While 循环。在底层循环中，我们从一个模拟输入读取数据，进行 PID 计算，并将计算结果写入一个模拟输出通道中。我们使用可以提供毫秒级，微秒级或时钟周期级分辨率的(25ns)循环定时 VI 对这个循环进行定时。这个循环执行速度可达 100kHz，即 10 微秒每次。

LabVIEW FPGA VI



LabVIEW Host VI

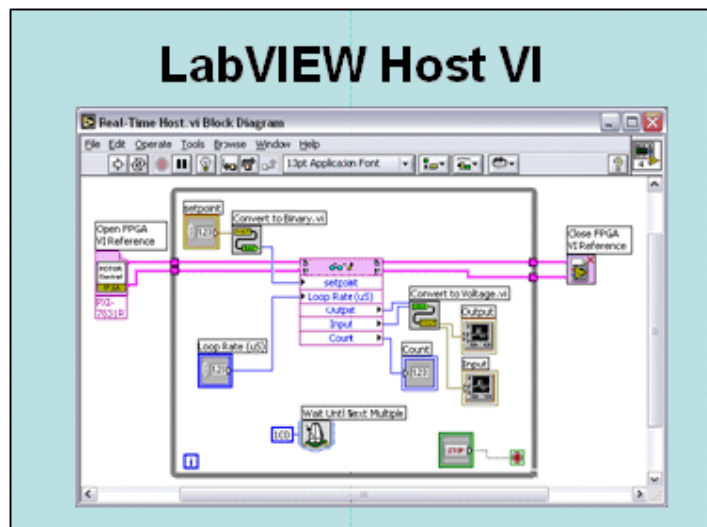


图 4. 针对定制计数器和控制应用的 FPGA 和主机应用程序

当我们使用 LabVIEW FPGA 附带的 PID 算法时，我们可以使用来自各个渠道的信号处理 IP 核心。NI SoftMotion 开发模块包含了增强的运动控制算法，以及针对许多反馈仪器的内建输入，包括正交编码器以及正弦和余弦编码器。NI 数字滤波器设计工具包含了针对设计、分析以及各种接口工具，便于您在 LabVIEW FPGA 中实现数字滤波器，提供了许多函数和交互式工具。另外，您可以使用 HDL 接口节点将定制的 VHDL 或 Verilog IP 核应用到 LabVIEW FPGA 程序框图中。

FPGA 开发流程

在创建 LabVIEW FPGA VI 程序后，您可以将代码编译成为可以在 NI RIO 硬件设备上运行的程序。与其他的 FPGA 开发工具相似，FPGA 虚拟仪器的编译时间可能从几分钟到几个小时不等，这取决于代码的复杂度以及您所开发系统的特性。为了最大程度提高开发效率，您可以使用 R 系列 RIO 设备，在比特精度的仿真模式下验证您所设计的逻辑功能是否正确，在验证无误之后再行编译。当您的目标对象是 FPGA 设备仿真器时，LabVIEW 从设备访问 I/O，并在 Windows 开发计算机上执行 VI 的逻辑功能。在这种模式下，您可以使用与 LabVIEW Windows 相同的调试工具，如高亮执行、探针和断点等等。

LabVIEW FPGA 代码编译完毕之后，您就可以创建一个 LabVIEW 主机 VI 程序，将您的 NI RIO 硬件设备整合到您的测试和控制系统当中。图 4 是创建一个 FPGA 应用程序所经历的开发过程。主机 VI 程序在 FPGA 虚拟仪器前面板上使用了控件和指示器，以便在 RIO 设备的 FPGA 和主机处理引擎之间进行数据传输。这些前面板对象都由 FPGA 内的数据寄存器表示。主机可以是 PC、运行 Windows 的 PXI 控制器、运行实时操作系统的 PC、PXI 控制器、便携式视觉系统，或是 CompactRIO。在上面这些例子中，我们可以使用 LabVIEW 主机 VI 程序修改设定点，循环速率，模拟输入，模拟输出，以及记数数据。

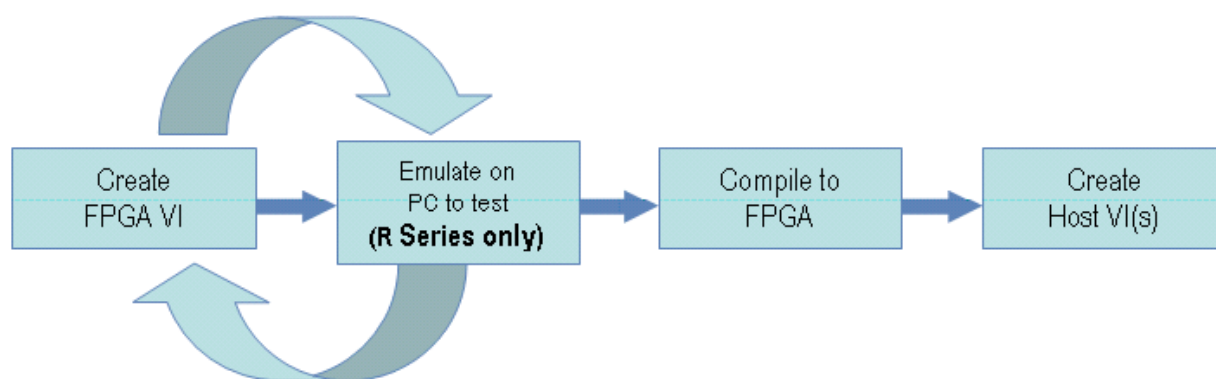


图 5. 应用开发流程

NI RIO 设备驱动包括图 5 中所示的一系列函数，您可以使用这些驱动开发与 FPGA 之间的通信接口程序。构建主机 VI 的第一步是打开一个 FPGA VI 程序和 RIO 设备的参考对象。如图 4 所示，“打开 FPGA VI 程序引用”函数，在执行时也会将编译之后的 FPGA 代码下载到 FPGA 上进行运行。打开了引用之后，您可以使用读写控制函数，对 FPGA 上的控制和指示器寄存器进行读写。将 FPGA 引用连线到该函数后，您可以简单地选择想要进行读写的控件或指示器。您还可以将 FPGA 读写函数放在一个 While 循环当中，连续对 FPGA 进行读写。图 4 中主机 VI 的最后一个函数是“关闭 FPGA VI 程序引用”函数。它将停止 FPGA VI 程序的执行，并关闭设备引用。现在，您可以将其它经过编译的 FPGA VI 程序下载到设备中，修改设备的各种功能。

LabVIEW 主机 VI 程序还可以用于执行浮点运算、数据记录、网络以及其它不适合 FPGA 实现的功能。您可以使用 LabVIEW 实时模块让您的主机应用程序在实时操作系统下运行，增加系统确定性和可靠性。LabVIEW 实时系统为 FPGA 中执行的同步或异步函数提供确定性的处理引擎。例如，浮点算术，包括快速傅立叶变换、PID 计算，以及定制的控制算法，通常都在 LabVIEW 实时环境下执行。相关的数据可以存储在 LabVIEW 实时系统中，或是传输到一台 Windows 主机上，以用于离线分析、数据记录或是用户接口显示等。图 7 说明了这种配置的架构。

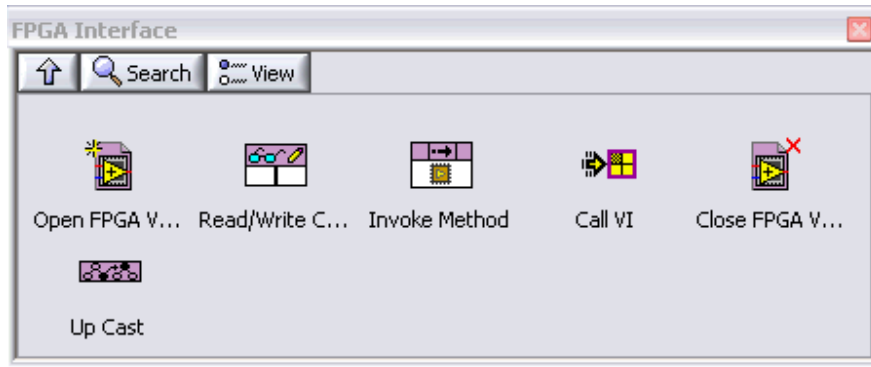


图 6. FPGA 接口面板

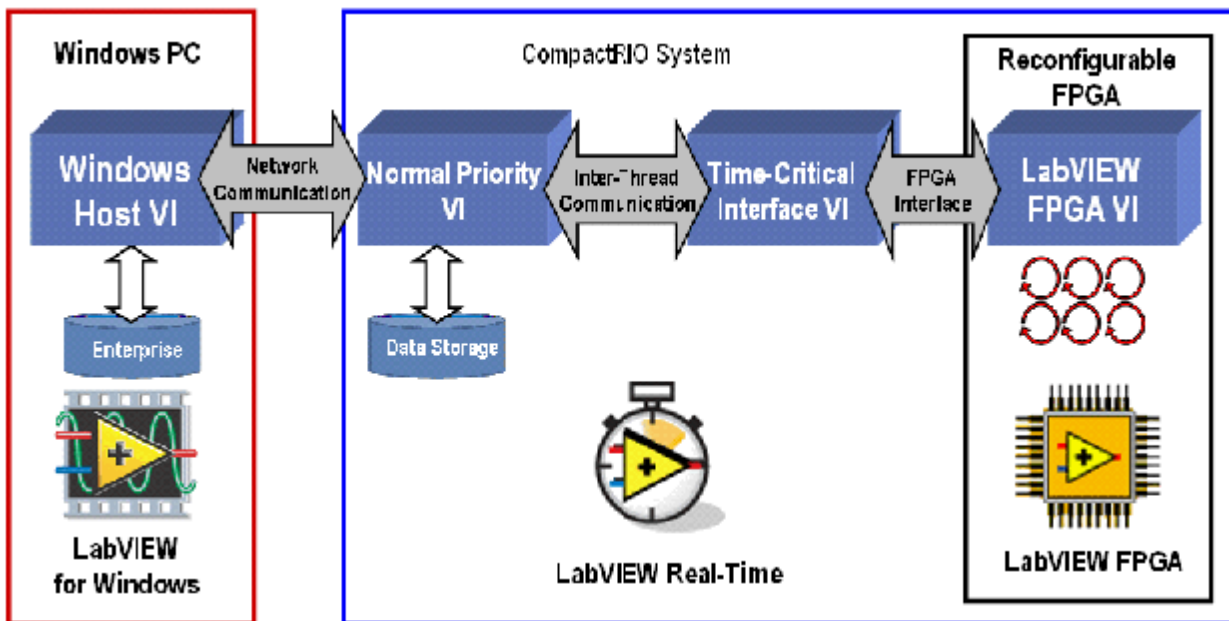


图 7. 使用 LabVIEW FPGA, LabVIEW RT 模块以及主机 PC 的完整应用架构

在 NI RIO 设备和 LabVIEW 主机应用程序之间有很多方法可以传递数据 (如图 7 所示的实时接口 Vic 程序)。在单点数据传输时, 您可以通过 FPGA 向主机发出一个物理中断, 表明有数据需要传输, 从而实现数据的同步传输。之后您可以让 FPGA 应用程序在中断处等待, 或是继续采集和处理数据。LabVIEW FPGA 还针对大量数据的传输, 配备了三个 DMA (直接内存访问) 通道。您可以使用 DMA 从 FPGA 设备流出数据。

图形化编程的性能优势

由于 LabVIEW FPGA 在实际的硬件设备上执行图形化程序框图, LabVIEW FPGA 的算法性能表现远远优于基于软件的系统。我们来考察一个使用 8 个 PID 控制循环的应用。使用 LabVIEW RT(实时)模块和 PXI-7831R 进行 8 个 PID 计算, 您可以获得大约 30kHz 的循环速度。而在 R 系列或 CompactRIO 设备的 FPGA 上实现基于整数的 PID 算法, 所有 8 个循环都可以达到超过 100kHz 的循环速度。这里控制循环的速度瓶颈还在于 A/D 转换的处理时间, 如果 A/D 速度更快, 循环速度也将相应提升。如果您的 RIO 硬件允许, 数字 I/O 的控制循环, 如 PWM, 可以达到 1MHz 以上的速率。

在 FPGA 上运行 LabVIEW 代码的另一项优势在于您可以实现真正的同时并行处理。在模块上不存在必须在多个任务之间分配 CPU 时间的操作系统。我们对图 4 中讨论的应用进行考虑。这个应用在 FPGA 上创建两个完全独立的处理器，可以与 FPGA 时钟同步或在异步地运行。甚至在同一个 While 循环或是其他结构当中，您也可以通过代码的流水线化，执行并行操作，从而提升执行速度。您可以将处理过程分为多个部分，在 While 循环的各个子循环中分别执行。在下面的图 8 中，一个控制算法被分成了两个部分。

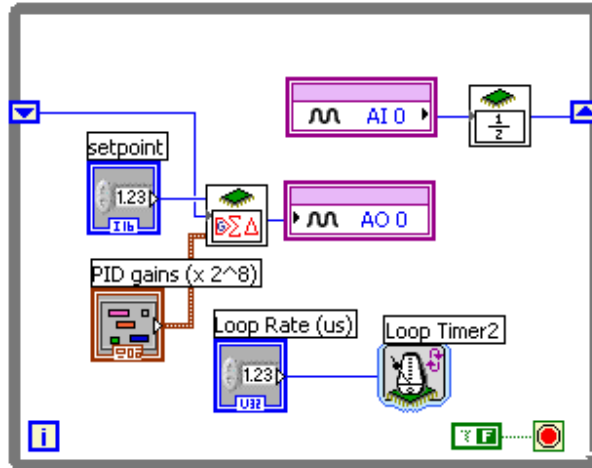


图 8. 流水线结构

在图 8 中，While 循环的每次执行都将刷新控制输出，而 AI 和 Control 1 操作与 AO 和 Control 2 操作则并行执行，最终算法的执行速度要远远高于线性处理架构下的速度。这种同时的并行处理方式在由微处理器和操作系统构成的平台上是不可能实现的。

典型应用

由于 LabVIEW FPGA 可以对 FPGA 的底层硬件设计进行配置，并在一个模块化系统当中进行使用，这对于需要定制硬件设备的应用来说是非常理想的。这些应用包括定制的模拟、数字和计数器/定时器 I/O，高达 100kHz 的模拟控制，高达 20MHz 的数字控制，以及与定制数字协议的接口连接。

定制测量应用

LabVIEW FPGA 模块为开发测量和控制系统带来全新层次的定时控制和同步能力。现在您可以通过在 LabVIEW 中绘制程序框图设计自己的 I/O 板卡。以测量汽车引擎的火花塞定时为例，即测量引擎曲轴转过以上端固定中心 (TDC) 为参考的 0 度角的时刻与某个火花塞点火的时刻之间的时间延迟。要完成测量，必须同时测量曲轴的位置和点火线圈的电压。典型的内建汽车曲柄传感器是磁性的，对曲柄上的齿轮齿数进行记数，并在您所读取的数字输入线上发出脉冲。这里忽略了 TDC 上的齿数。因此为了测量火花塞定时，必须在实时地确定传感器信号指明忽略齿数的时间。现在，您还必须实时测量这一事件的数字输入与火花塞点火这一模拟事件之间的延时。您可以通过 LabVIEW FPGA 模块实现这一逻辑功能，从而创建一个定制的 I/O 设备。大多数的 I/O 板卡会向主机返回一个电压或者脉冲宽度，而这一设备将直接返回根据混合信号输入计算得出的火花塞定时结果。

模拟控制

很多类型的模拟控制应用可以受益于 LabVIEW FPGA 和 RIO 硬件设备的硬件确定性。借助于 LabVIEW FPGA 模块的灵活性，您可以实现各种常见的控制算法，如 PID，也可以为您的应用开发定制算法。下面是一些模拟控制的实例：

- 测力计控制（速度/负载）
- 液压伺服或电动震子控制
- 电机速度或位置控制

仿真

除了作为如此前所述的一个完整的控制系统，您还可以在系统中使用 RIO 硬件设备进行快速控制原型 (RCP)，以及对开发控制器的进行硬件在环仿真测试。电子控制器在许多产品中很常见，如汽车和家居设备。在开发的初级阶段，构建控制器之前，必须在将要受控的系统上测试控制算法。为了完成这项工作，RCP 系统可以作为控制器，使用特别设计的算法对设备的控制进行测试。当构建控制器的初级版本之后，也必须进行验证。对于扩展性测试来说，采用仿真受控系统测试控制器是相对便捷的方法。这就是通常所提到的硬件在环 (HIL) 测试。

LabVIEW FPGA 模块在 LabVIEW 和 LabVIEW 实时模块各项功能的基础上，将 LabVIEW 平台进行了扩展。通常来说，对于一个 HIL 或是 RCP 应用，您通常使用 LabVIEW 实时终端作为您的主机处理器，执行主要模型和复杂的处理算法。您现在可以使用 LabVIEW FPGA 模块对 R 系列或 CompactRIO 底板进行配置，处理 I/O，并与外部事件进行同步。对于一般固定功能的 I/O 设备来说，相比在主机上通过软件处理的速度，通过硬件处理的速度要快得多。在 FPGA 内，您可以对您的硬件处理过程进行定制，包括仿真传感器输出，生成 PWM 信号并进行解码，以及根据测量值进行板上决策。如果采用这样的架构，您就不必选择多处理器系统，而这种系统通常在同时需要仿真和 I/O 的情况下使用。

离散控制

制造业应用通常需要快速、鲁棒的控制系统。LabVIEW FPGA 模块通过使用 RIO 设备上的 FPGA 执行应用的逻辑功能，提供了一个能够开发此类控制系统的平台。R 系列和 CompactRIO 设备可以对多条数字线进行扫描和设置，并同时在不到一微秒的时间内做出控制决策。另外，由于运行 LabVIEW FPGA 的 RIO 设备没有上层的操作系统软件，如实时操作系统或是 Windows 内核的修正等，因此不会出现不可预知的延迟。

数字化协议仿真

许多应用中需要系统具备与非常用数字协议设备之间的接口。通常来说，这种硬件或是非常昂贵，或是根本没有供应。如果使用 LabVIEW FPGA 模块，您就可以利用硬件的高速数字处理能力非常灵活地进行信息编解码。这样，您可以将 FPGA 配置成为一个针对所需通信协议的接口模块。一般使用到的协议包括 SPI，I2C，SP/DIFF，RS-232 等等。

PWM 通信

脉宽调制 (PWM) 信号在汽车以及电信等各个工业领域非常普遍。PWM 脉冲序列中，频率是恒定的，信息是通过占空比——也就是脉冲电平为高的时间百分比，来进行传递。许多固定功能的数据采集设备无法处理快速变化的占空比，或只能为静态 PWM 分配固定数量的通道。而您可以使用 LabVIEW FPGA 模块，对 RIO 硬件设备的任意一根数字线进行配置，根据需要读写 PWM 信号。

灵活的编码器接口

编码器是一种用于测量速度或位置的设备。当轴柄转动时，会生成脉冲以指示转动的角度。市场上现在有很多类型的编码器，有时在同一个应用中可能使用到多种类型的编码器。您可以使用 **LabVIEW FPGA** 模块，对 **RIO** 设备进行配置，使之可以与多种类型的编码器在不同的数字线上进行连接，还可以对设备进行重新配置以适应不同应用的需要。您可以添加您自己的定制功能，例如：与其它信号同步读取计数器，而不是根据输入信号判断计数器的值。

总结

在使用 **LabVIEW FPGA** 模块之后，您的应用可以利用 **FPGA** 的高速执行以及同步能力，同时可以使用数据采集、运动控制、工业通信等传统功能。数据流模型和 **LabVIEW** 图形化编程语言所内在的并行机制可以为硬件中执行的测量和控制应用提供一个高度直观的开发环境。配合 **LabVIEW** 和 **LabVIEW** 实时模块的使用，您可以开发全新的应用，并最大限度地提升系统的性能。

LabVIEW FPGA 模块为测量和控制应用带来 **FPGA** 的灵活性、高性能，以及可定制化能力。您能够使用 **NI RIO** 设备和 **LabVIEW** 图形化编程，利用 **COTS** 平台构建灵活的定制设备，而在之前唯一的解决方案是制作定制硬件。数据流模型和 **LabVIEW** 图形化编程语言内在的并行机制可以为硬件中执行的测量和控制应用提供一个高度直观的开发环境。将 **LabVIEW FPGA** 作为您测量和控制系统的一部分将大大提升应用的灵活性，丰富系统的功能，特别是针对需要高速控制，定制数字协议接口，或是定制模拟、数字和计数器混合 **I/O** 的应用。

