

# 应对多样化数字接口测试挑战的解决方案

## ——基于可编程 FPGA 的测试仪器

摘 要 .....	2
背 景 .....	3
可编程 FPGA 的灵活性 .....	5
测试应用举例 .....	7
总 结 .....	11
参 考 .....	12

## 摘要

现今，电子测试工程师面临着大量多样化的数字接口带来的挑战。军事、航空、通信和汽车电子都同时使用工业标准和定制数字接口。专用特定接口的测试仪器和用于通用数字 I/O 接口的测试仪器可以支持以上大部分接口。专用测试仪器是非常昂贵的，通用数字 I/O 接口的测试仪器仅试用部分方案并且需要大量编程/调试工作，最终也会增加成本。然而，基于可编程 FPGA 的测试仪器可以提供成本合理的解决方案，并可选通用数字 I/O 接口还是专用特定接口。

本文将概述现代数字测试接口的需求，重点讲解在电子模块和系统的监视和控制中应用越来越普遍的串行数字接口。这些接口包括定制和工业标准版本，例如 SPI、I<sup>2</sup>C、RapidIO、Ethernet 和 SpaceWire 等。对于高速串行接口，带有特定接口的 SERDES 的使用和功能逻辑可以提供创建满足需求的测试仪器的基础。本文将会讲述当结合可用的商用 [IP \( Intellectual Property \)](#) 或定制的逻辑单元时，基于可编程 FPGA 的测试仪器如何提供低成本、高灵活性的测试方案，此种测试仪器解决方案将实现单台仪器支持多种应用和多种接口。

关键字：FPGA 数字测试 接口

## 背景

当测试工程师考虑如何测试数字接口时，就必须在技术指标需求和方案成本之间进行权衡。有些公司会选择为特定通信协议测试而设计的专用仪器。或者，选择通用数字 I/O 测试设备，只要测试工程师付出一些努力就可以用来测试大量协议。除了选择测试仪器，测试工程师经常通过自行设计电路来实现与被测部件（UUT）的逻辑系列接口吻合。随着可编程硬件的到来，尤其是 FPGAs，测试工程师在测试设备的成本选择方面有了新工具。基于 FPGA 的测试仪器使测试工程师可以通过使用商用 IP 内核、自主开发或定制 IP 内核构建测试方案。通过对这些 IP 内核与 FPGA 中额外的寄存器/逻辑的集成，并提供必要的逻辑系列接口（TTL、LVDS 等），测试工程师可以实现使用单台仪器测试多种接口。

很多公司支持军方客户发展可以测试多种产品的通用测试站。测试站被设计用于测试“大部分” UUTs。对于测试“大部分”以外的 UUTs，随着“大部分”以外的 UUTs 的数量和种类的增加，为了完成全部测试，需要使用的测试仪器的数量也会成直线飙升。在通用测试站中使用可编程仪器，可减少对测试“大部分”以外的 UUTs 的仪器数量甚至不再需要额外的测试仪器。基于 PXI<sup>[1]</sup> FPGA 板卡为测试站提供此种解决方案，如图 1。



图 1 一个 FPGA 板卡可以替代多台仪器

每一代新产品的电路复杂性、I/O 密度和信号速率都在持续增高。工厂已经采纳将串行总线应用在设计中作为一种减少需求最小化 I/O 密度和提高数据传输速率的方法。这些串行总线应用在 PCBs 布线中，用于连接微处理器和外设、传感器和储存器。在系统中，串行总线被用来控制设备和模块。有很多串行总线可以应用在嵌入式系统设计中，例如 I<sup>2</sup>C、SPI、Rapid IO 和 PCI Express，PCI Express 可以通过背板/线缆在板卡或模块之间通信。其他串行总线，例如 RS-232、RS-485、Ethernet 和 USB，支持远距离通信。

## 可编程 FPGA 的灵活性

各式各样的总线应用于电子产品中，在特殊行业会应用特殊的总线接口。例如，汽车产业使用 CAN 总线，军用飞机使用 MIL-STD-1553 总线，商用飞机使用 ARINC-429 总线，空间电子使用 MIL-STD-1553 总线、SpaceWire 总线和 RapidIO 总线。测试系统为了支持这些总线接口，需要专用的测试仪器。但是，测试系统可能会需要使用一台测试设备支持多种总线结构，因为使用专用测试仪器会增加额外的系统空间、成本和测试系统复杂性。

若想采购支持每种总线结构的独立仪器可以考虑可编程 FPGA 测试仪器，这种仪器可以通过合适的固件逻辑配置模拟指定的总线标准。有很多供应商针对这些总线结构提供 IP 内核逻辑，甚至是开源的，例如 [opencores.org](http://opencores.org)<sup>[2]</sup>。一旦开始运行，用于 FPGA 仪器的文件会被储存在测试系统控制器的硬盘中。然后，当技术人员使用测试程序欲对 UUT 进行测试时，与接口 IP 有关的文件就会被加载到 FPGA 仪器中。最后，完成配置的 FPGA 板卡就可以用来测试 UUT 了。在测试完成后，FPGA 板卡可以重置并用于不同的 UUT 测试，如图 2。

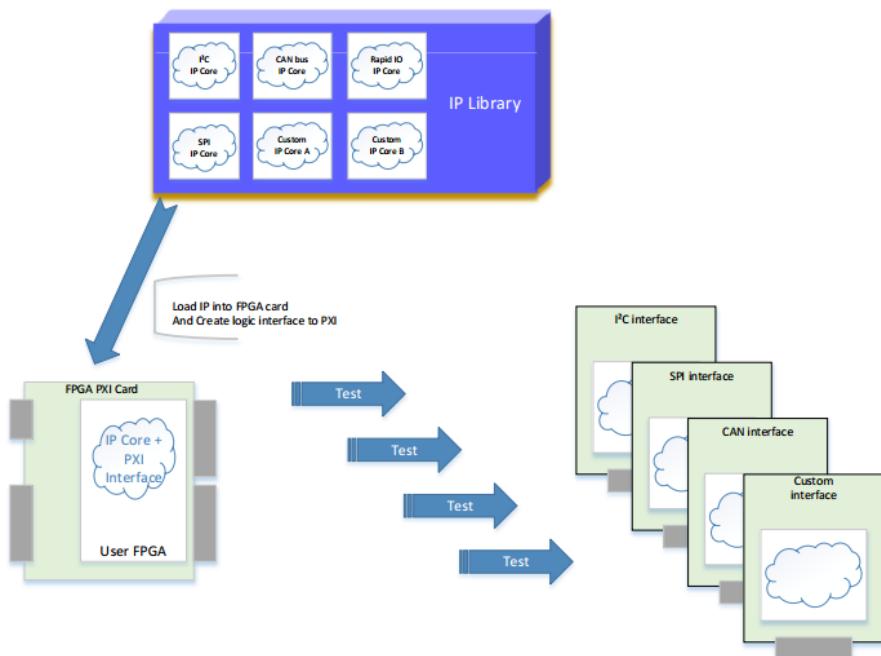


图 2 FPGA 仪器功能 – 支持多种 UUTs & 接口

除了使用指定的 IP 或 IPs，测试工程师必须考虑不同总线对物理层（PHY）/电气接口的要求，因为 FPGA 本身不具备接口电路，一般不支持总线 PHY 的要求。下表列出了一些总线和对应的 PHY 接口。

表 1 总线和对应的物理层接口

<i>Bus Interface</i>	<i>Physical Interface / Characteristics</i>
MIL-STD 1553	Tri-axial, transformer coupled, 18 -26 Vpp output
ARINC 429	Differential, high voltage
RS-232	Unbalanced, bi-polar
RS-422 / RS-485	Balanced, bi-polar
CAN Bus	Differential, CAN bus specific XCVR
Ethernet	Differential, transformer coupled
USB	Differential, logic levels
I2C, SPI	Buffered, logic levels
SpaceWire	LVDS
SerDes	LVDS

如表中所列，每种总线都要求专用或缓冲物理层接口，这些要求都可以通过在 FPGA 模块的子板卡上集成必须的接口实现。因为 FPGA 提供成百上千路 I/O 信号，所以构建可支持多种接口类型的子板卡是可能的。使用特定的接口子板卡时，基于 FPGA 的板卡是如何编程的，如图 3。

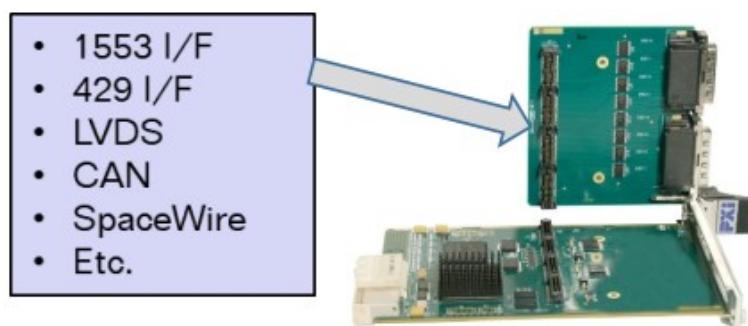


图 3 在单板卡中集成多种类型接口

## 测试应用举例

FPGA 仪器支持定制/自定义接口，促进了测试行业的发展，为验证使用定制/自定义总线接口进行内部模块通信的系统的内部的多种组件/模块提供了一种解决方案。例如，一个客户搭建了一个新测试系统，这个测试系统在一个机箱内包括多个板卡或模块。机箱采用定制/自定义的串行背板，可用于在主板卡和从板卡之间进行通讯。板卡是使用 FPGAs 进行设计的，用于实现通信/控制功能并致辞串行接口逻辑，如图 4。

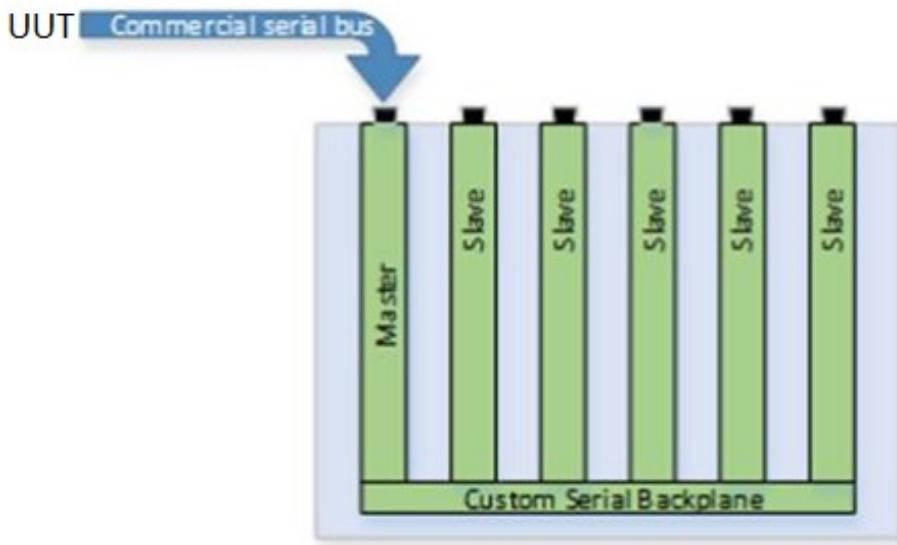


图 4 基于 FPGA 的板卡和定制/自定义串行通信总线

这种测试系统的设计需要支持定制/自定义串行总线接口并考虑使用动态数字 I/O 接口或基于可编程 FPGA 的 PXI 板卡。与 FPGA/板卡设计者一起工作的测试工程师会选择尝试基于 FPGA 的测试方案，将串行背板 VHDL<sup>[3]</sup> 内核编入 FPGA 主板卡和 FPGA 从板卡中。使用安装在子板卡中的电路限制并转换 FPGA 信号电平以匹配 UUT 逻辑系列。通过 PXI FPGA 板卡的 API 访问寄存器从而实现在 PXI FPGA 板卡和 ATE 控制器之间发送和读取数据，这种访问方法非常简单。此外，FPGA 板卡的结构分为两个独立的 PCI 总线接口，简化总体设计和

User FPGA 的实现，如图 5。这中测试方案被证明是非常成功的，结束了数字 I/O 解决方案的时代。

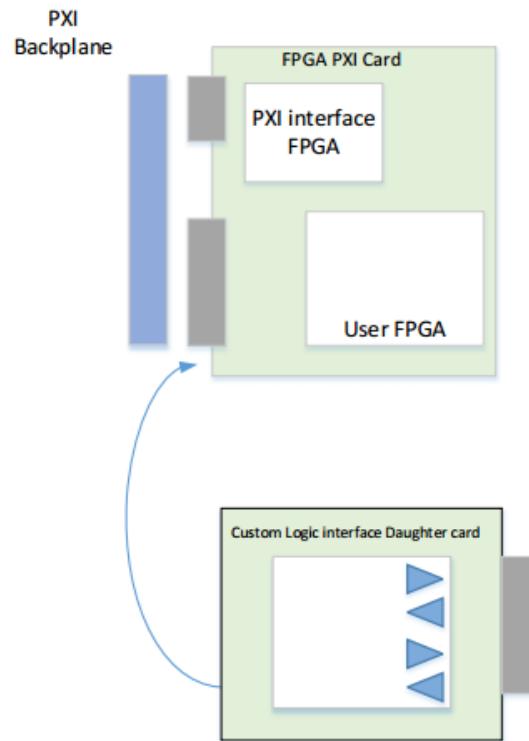


图 5 FPGA 仪器结构

若要测试从板卡，主板卡必须将 IP 内核加载到 FPGA 中，这样就可以允许测试仪器与所有从板卡进行通信和测试了，如图 6。



图 6 FPGA 仪器测试从板卡

同样地，若要测试主板卡需要执行类似的操作步骤，从板卡将 IP 内核加载到 FPGA 中后，基于 PXI 的 FPGA 主板卡仪器将作为 UUT 被从板卡进行测试，如图 7。

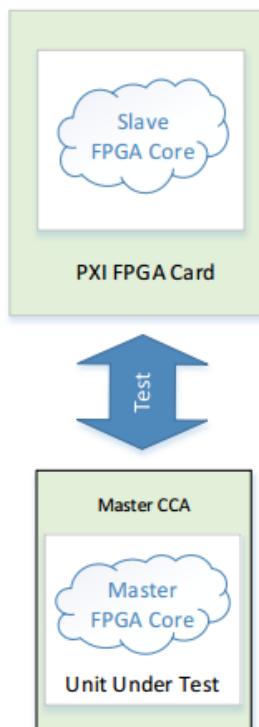


图 7 FPGA 仪器测试主板卡

以上说讲述的测试方法可以应用于机箱测试。对于这种应用，使用商用总线接口与集成的系统或机箱进行通信。通过加载商用 IP 内核到 PXI FPGA 后，FPGA 仪器就可以与机箱和机箱内相关板卡进行通信和监测了，如图 8。

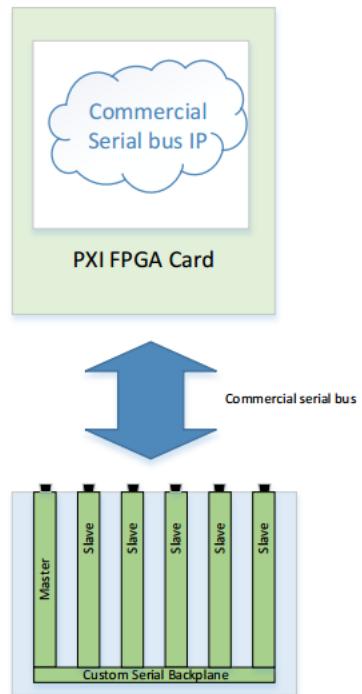


图 8 FPGA 仪器支持系统/机箱级测试

## 总 结

现今，测试工程师面对这不同领域的数字接口带来的测试挑战。对于每种接口都要使用一台相应接口的测试仪器的传统测试方案不仅成本高、空间利用率低，而且还极大地限制灵活性。然而，通过使用基于可编程 FPGA 仪器的可编程测试设备，测试工程师可以实现使用单台测试仪器应对现今和未来的多种数字接口带来的挑战。

## 参 考

[1] <http://www.pxisa.org/specifications/> - PXI Hardware specification

[2] <http://www.opencores.org>

[3] IEEE 1076-2008 VHDL Standard

若欲阅读英文原文，请点击[这里](#)

若需更多帮助，请[联系我们](#)