Digital Avionics Interface Selection: Simplifying a Not So Simple Choice

Wednesday, May 25th – 9:00am

Brandon Treece, NI

Nathan Johnson-Williams, New Wave DV

Digital Avionics Interface Selection: Simplifying a Not So Simple Choice

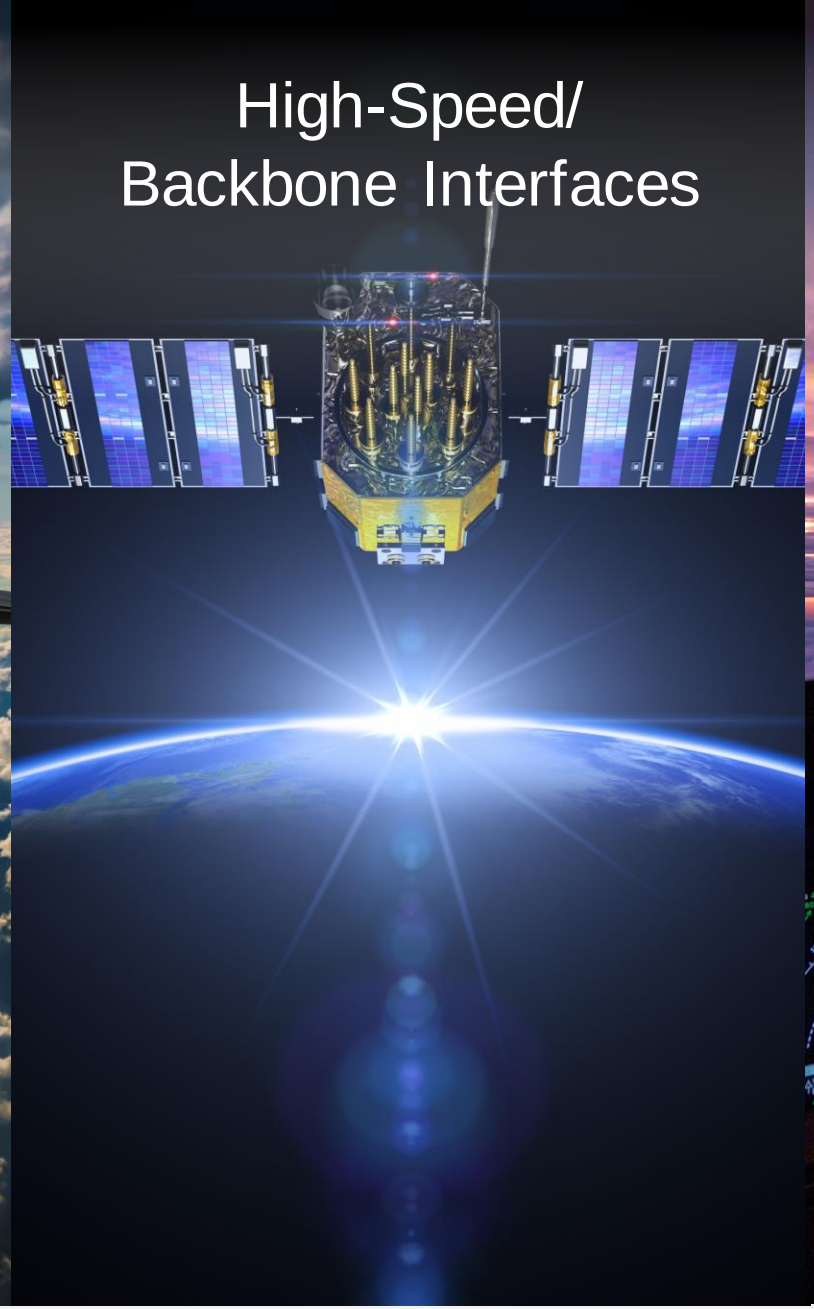
Brandon Treece
Chief Solutions Marketing Manager
Electronics Test, Defense Aircraft
Aerospace, Defense, and Government
NI

Digital Avionics Interfaces Span Platform Assets

Generic Interfaces

MIL-STD-1553

F-35 ON-BOARD DATA HANDLING



High-Speed/ Backbone Interfaces

Serial RapidIO®

FOCAL PLAN ARRAY IMAGE DATA

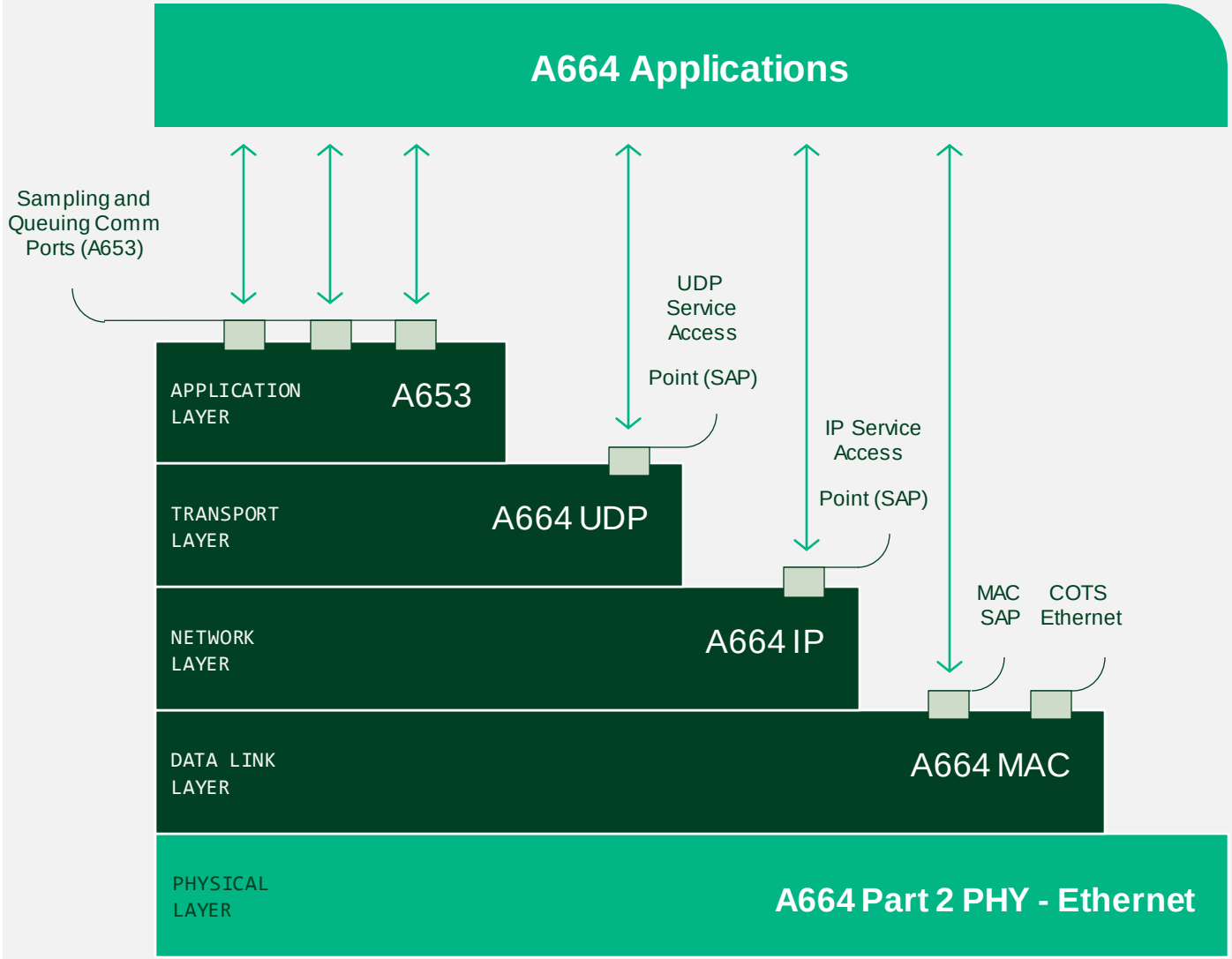
Application-Specific Interfaces

ARINC 818

COCKPIT VIDEO DISPLAY

OSI LAYERS OF ARINC-664P7/AFDX

Generic Digital Avionics Bus Interfaces



Commoditization has led to little differentiation in vendor hardware offerings

Test equipment vendors differentiate at the higher layers of the OSI stack

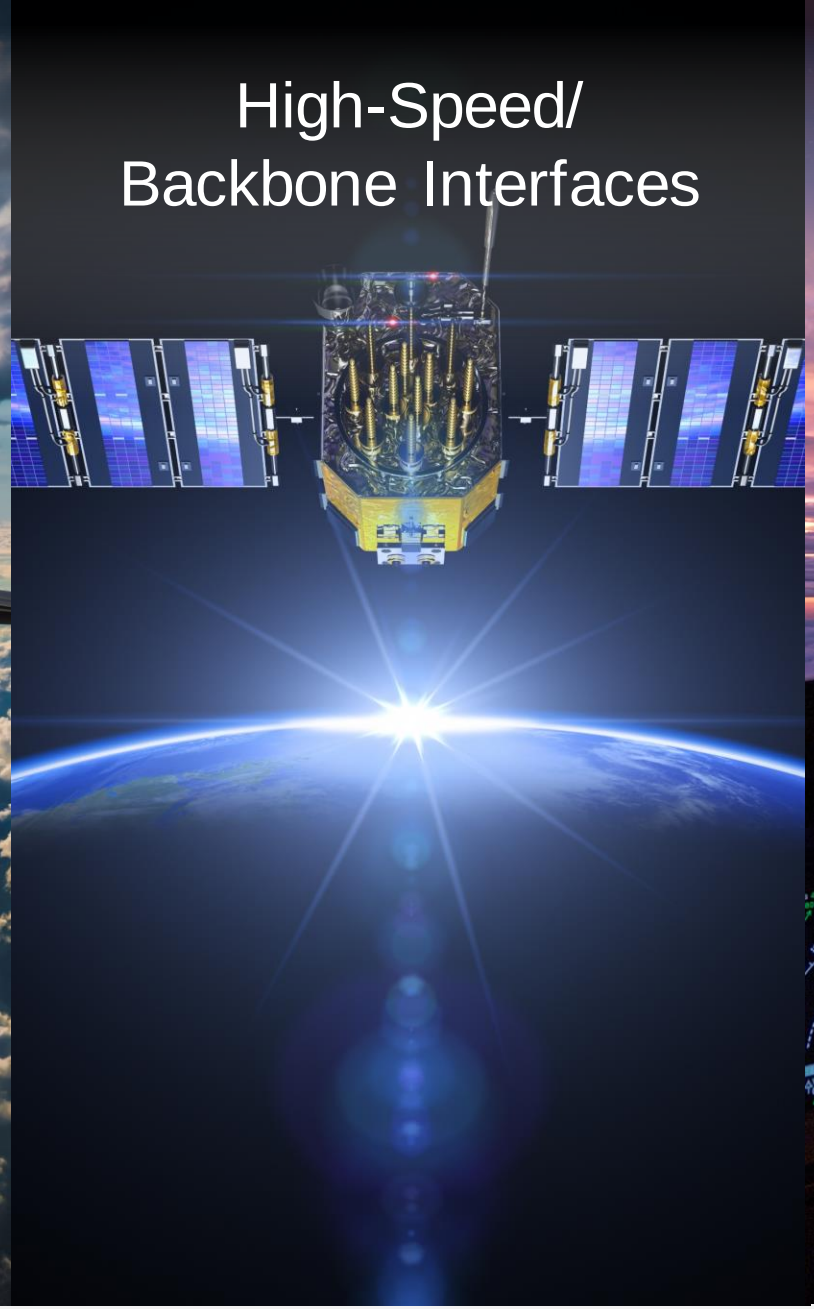
Manufacturing and Depot test requirements typically don't require advanced software features

NI offers interfaces for MIL-STD-1553, ARINC 429, RS232/422/485, and CANbus

Generic Interfaces

MIL-STD-1553

F-35 ON-BOARD DATA HANDLING



High-Speed/ Backbone Interfaces

Serial RapidIO®

FOCAL PLAN ARRAY IMAGE DATA

Application-Specific Interfaces

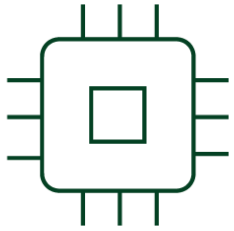
ARINC 818

COCKPIT VIDEO DISPLAY

Numerous Approaches to Incorporating FPGAs

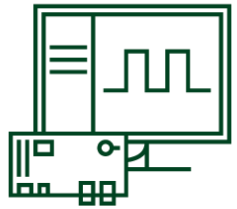
The wrong approach leads to risk of increases support burden over time

Develop your own hardware and IP from scratch

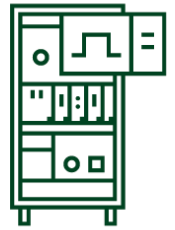


Use a COTS FPGA development board, and write your own IP

Use a COTS FPGA development board, and off-the-shelf IP core



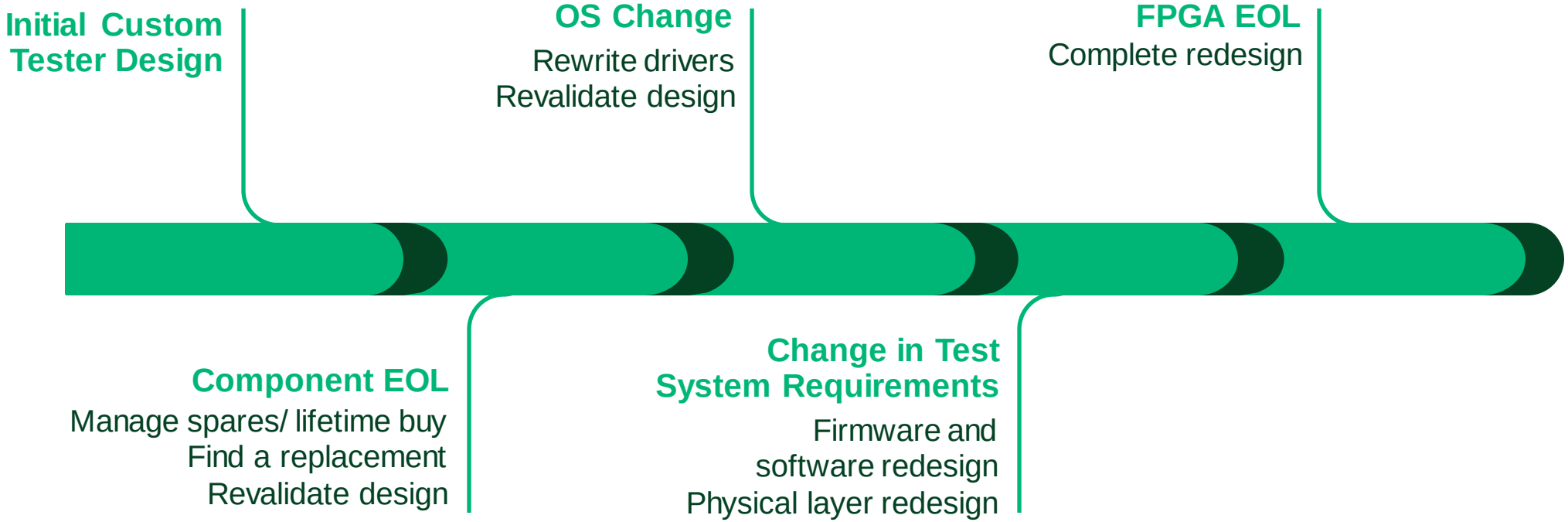
Use a COTS FPGA-enabled PXI module, and write your own IP



Use a COTS FPGA-enabled PXI module, and off-the-shelf IP core

DECREASING RISK

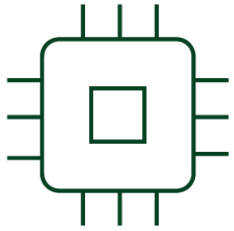
Life-Cycle Management of a “Homegrown” Test System



Numerous Approaches to Incorporating FPGAs

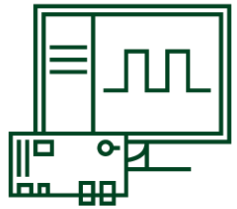
The wrong approach leads to risk of increases support burden over time

Develop your own hardware and IP from scratch

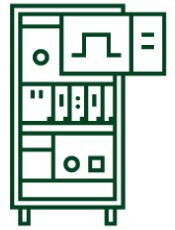


Use a COTS FPGA development board, and write your own IP

Use a COTS FPGA development board, and off-the-shelf IP core



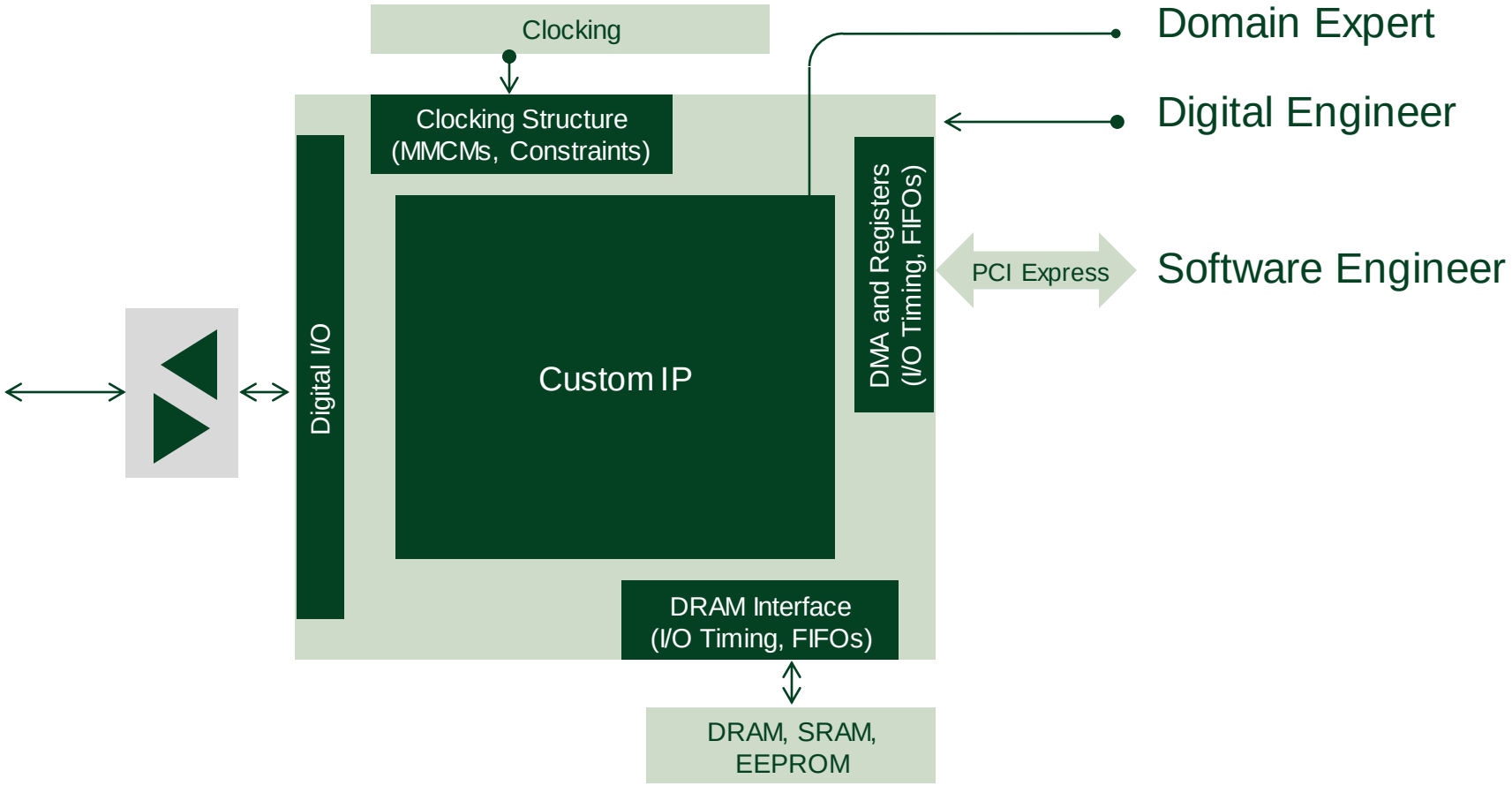
Use a COTS FPGA-enabled PXI module, and write your own IP



Use a COTS FPGA-enabled PXI module, and off-the-shelf IP core

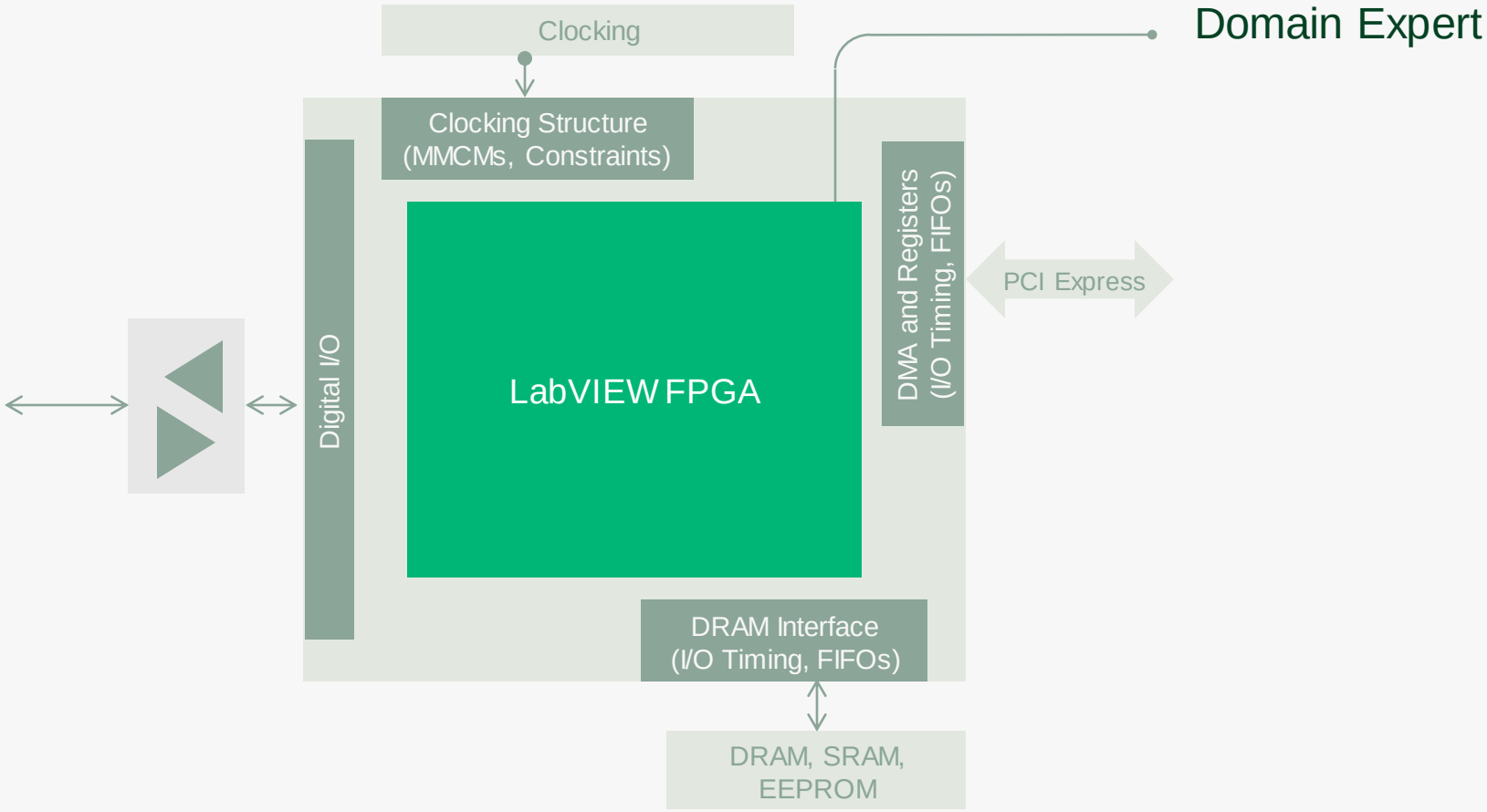
DECREASING RISK

Typical Custom Design With I/O



NI's Approach to FPGA Design

Focus on your IP with LabVIEW FPGA



NI Tools for FPGA-Based Digital Avionics Interfacing

Hardware

Software

Host Programming

C/C++
Python

LabVIEW

FPGA Development Environment

VIVADO

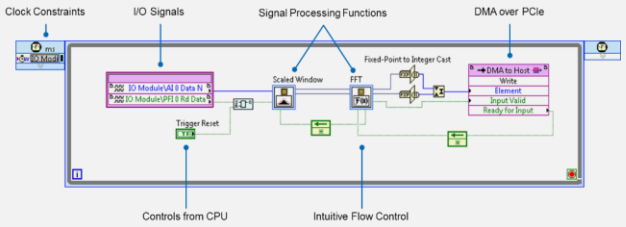
LabVIEW

VHDL Verilog Netlist — FPGA IP

LabVIEW

IP

Digital Communications IP Cores and Example Code



NEW WAVE DV Design & Verification

FPGA-Based Digital Interfacing in PXI

Reconfigurable I/O

High-Density SE Digital

High-Speed Serial

High-Speed Serial (MGT interface)

FlexRIO Custom Instruments

High-Performance SE, Differential Digital, and Custom I/O

NI Tools for FPGA-Based Digital Avionics Interfacing

Hardware

Software

Host Programming

C/C++
Python

LabVIEW

FPGA Development Environment

VIVADO

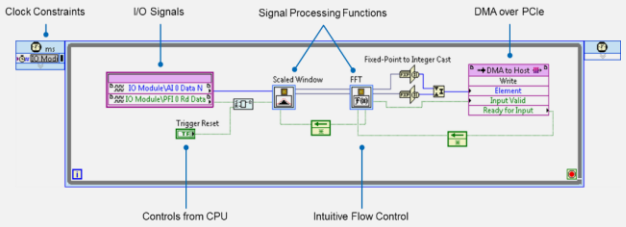
LabVIEW

VHDL Verilog Netlist — FPGA IP

LabVIEW

IP

Digital Communications IP Cores and Example Code



NEW WAVE DV Design & Verification

NEW WAVE DESIGN & VERIFICATION

Company Overview

New Wave DV is the leading expert in high-speed serial interfaces. Our expertise allows us to move data effectively and efficiently from data producer to consumer. With a focus on industry leading performance and environmental durability, our products are the perfect fit for defense and aerospace applications.

Our Mission

Enable our Partners to Change the World

About New Wave DV

Privately held, +60 employees

Headquartered in Minneapolis, Minnesota.

Second Office in Colorado Springs, Colorado

Core competency is FPGA network engineering

Local outsourced manufacturing

ITAR certification

Strong commitment to customer's support and responsiveness

Long product life cycle

New Wave DV Products

IP Cores

Ethernet

Fibre Channel

ARINC 818

Mil1394 (1394b AS5643)

SerialFPDP

HOTLink II

High Speed Data Bus (HSDB)

Serial RapidIO®

LabVIEW™

High-Speed Serial Interface Boards

High Speed Serial Cores on NI PXIe cards

Windows & Linux OS software support

Custom variants welcomed

NI PXIe cards pre-loaded with New Wave DV's IP

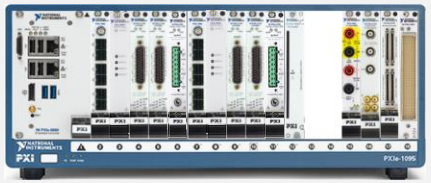
Platforms & Appliances

Interface Analysis, Test, Record, Playback

Configurable/Selectable port counts, speed, storage space, environmentals

Full IP Core Suite support

Precision-built for maintenance, sustainment, and test applications



Engineering Services

Solutions can be based on existing NWDV products or all-new. We will analyze your requirements, leverage existing hardware and IP cores where possible, create new where required, and ultimately deliver your solution.

Ethernet

ExpressXG

Mil1394

PHY

OHCI Link Layer Controller

GP2Lynx Link Layer

1394b AS5643 Link Layer Controller

Fibre Channel

Link Layer

Anonymous Subscriber Messaging

FC Upper Layer Protocol

Serial Front Panel Data Port

sFPDP Link Layer

sFPDP Express

ARINC 818

ARINC 818 DMA

ARINC 818 Stream

Additional Protocols

Serial RapidIO®

HOTLink II

High Speed Data Bus

NEW WAVE DESIGN & VERIFICATION

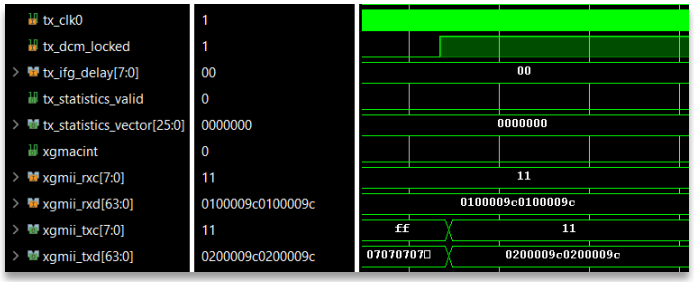
New Wave DV IP Cores

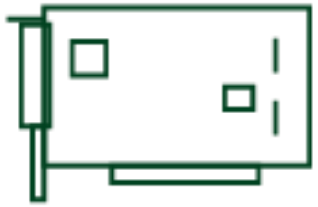
New Wave's FPGA/Interface Cores can be provided pre-loaded on NI Serial Instruments to provide a turn-key solution. They are also available stand-alone for your custom development.

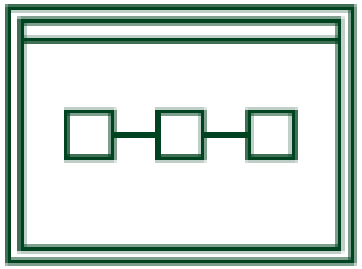
Advantages of Applications on NI FPGA

- Quickest path to HDL on Hardware
- NI FPGA provides accessory IP
- Directly interface with HDL IO

- No Driver Development
- Application Natively in LabVIEW
- No C/C++ Require







HDL Code or IP

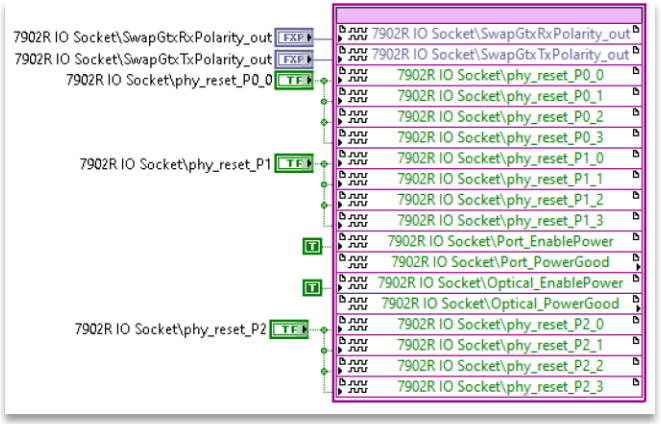
NI FPGA Hardware

LabVIEW Application

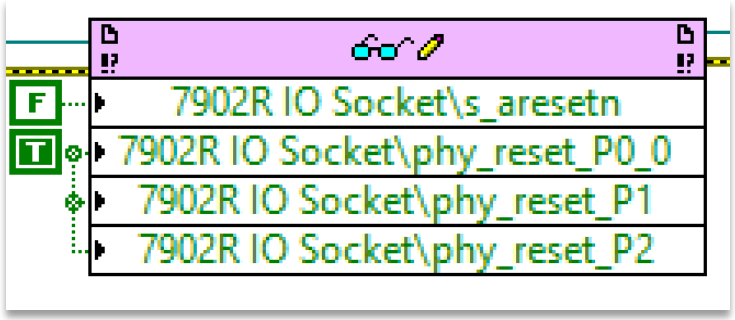
Directly Control Top-Level IO

```
phy_reset_P0_0          : in std_logic;
phy_reset_P0_1          : in  std_logic;
phy_reset_P0_2          : in  std_logic;
phy_reset_P0_3          : in  std_logic;
phy_reset_P1_0          : in  std_logic;
phy_reset_P1_1          : in  std_logic;
phy_reset_P1_2          : in  std_logic;
phy_reset_P1_3          : in  std_logic;
phy_reset_P2_0          : in  std_logic;
phy_reset_P2_1          : in  std_logic;
phy_reset_P2_2          : in  std_logic;
phy_reset_P2_3          : in  std_logic;
```

Route the signals to the top level of the Socketed CLIP



Within a single cycle loop connect controls/indicators to your top level signals

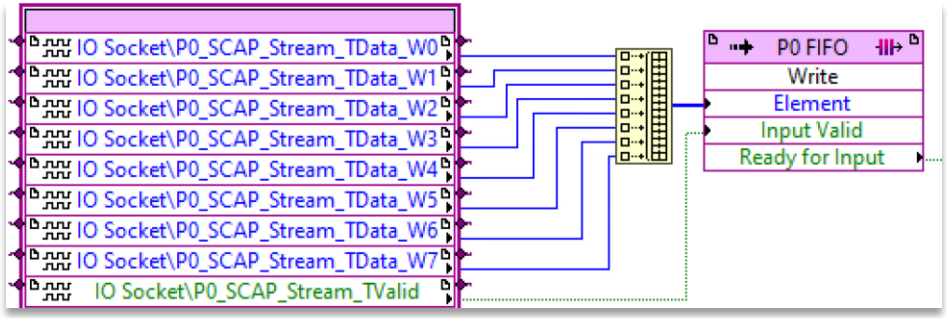


Directly read/write the signals in LabVIEW

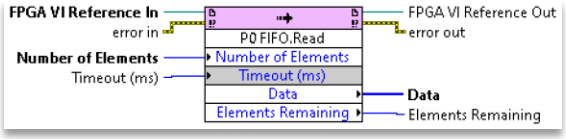
DMA to Host

```
P0_SCAP_Stream_TValid: out std_logic;
P0_SCAP_Stream_TReady: in std_logic;
P0_SCAP_Stream_TData_W0: out std_logic_vector(63 downto 0);
P0_SCAP_Stream_TData_W1: out std_logic_vector(63 downto 0);
P0_SCAP_Stream_TData_W2: out std_logic_vector(63 downto 0);
P0_SCAP_Stream_TData_W3: out std_logic_vector(63 downto 0);
P0_SCAP_Stream_TData_W4: out std_logic_vector(63 downto 0);
P0_SCAP_Stream_TData_W5: out std_logic_vector(63 downto 0);
P0_SCAP_Stream_TData_W6: out std_logic_vector(63 downto 0);
P0_SCAP_Stream_TData_W7: out std_logic_vector(63 downto 0);
```

Route the signals to the top level of the Socketed CLIP



Within a single cycle loop connect controls/indicators to your top level signals

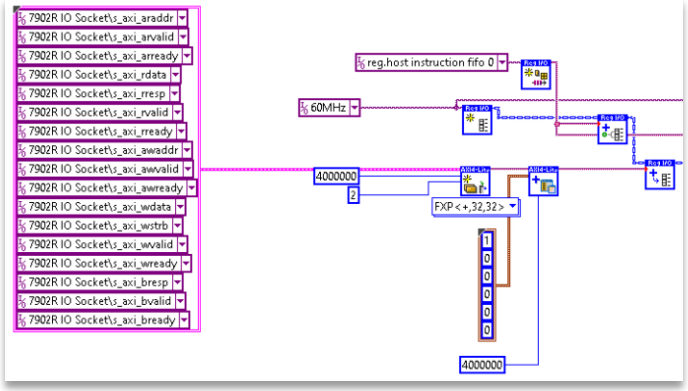


Directly read/write the signals in LabVIEW

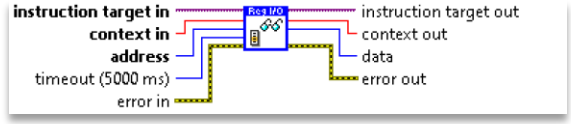
AXI-LITE Read/Write

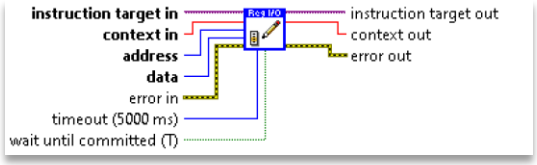
```
s_axi_awaddr            : in  std_logic_vector(31 downto 0);
s_axi_awvalid           : in  std_logic;
s_axi_awready           : out std_logic;
s_axi_wdata             : in  std_logic_vector(31 downto 0);
s_axi_wvalid            : in  std_logic;
s_axi_wready            : out std_logic;
s_axi_wstrb             : in  std_logic_vector(3 downto 0);
s_axi_bvalid            : out std_logic;
s_axi_bready            : in  std_logic;
s_axi_bresp             : out std_logic_vector(1 downto 0);
s_axi_araddr            : in  std_logic_vector(31 downto 0);
s_axi_arvalid           : in  std_logic;
s_axi_arready           : out std_logic;
s_axi_rdata             : out std_logic_vector(31 downto 0);
s_axi_rvalid            : out std_logic;
s_axi_rready            : in  std_logic;
s_axi_rresp             : out std_logic_vector(1 downto 0);
```

Route the AXI signals to the top level of the Socketed CLIP



Follow the example designs to connect the AXI signals

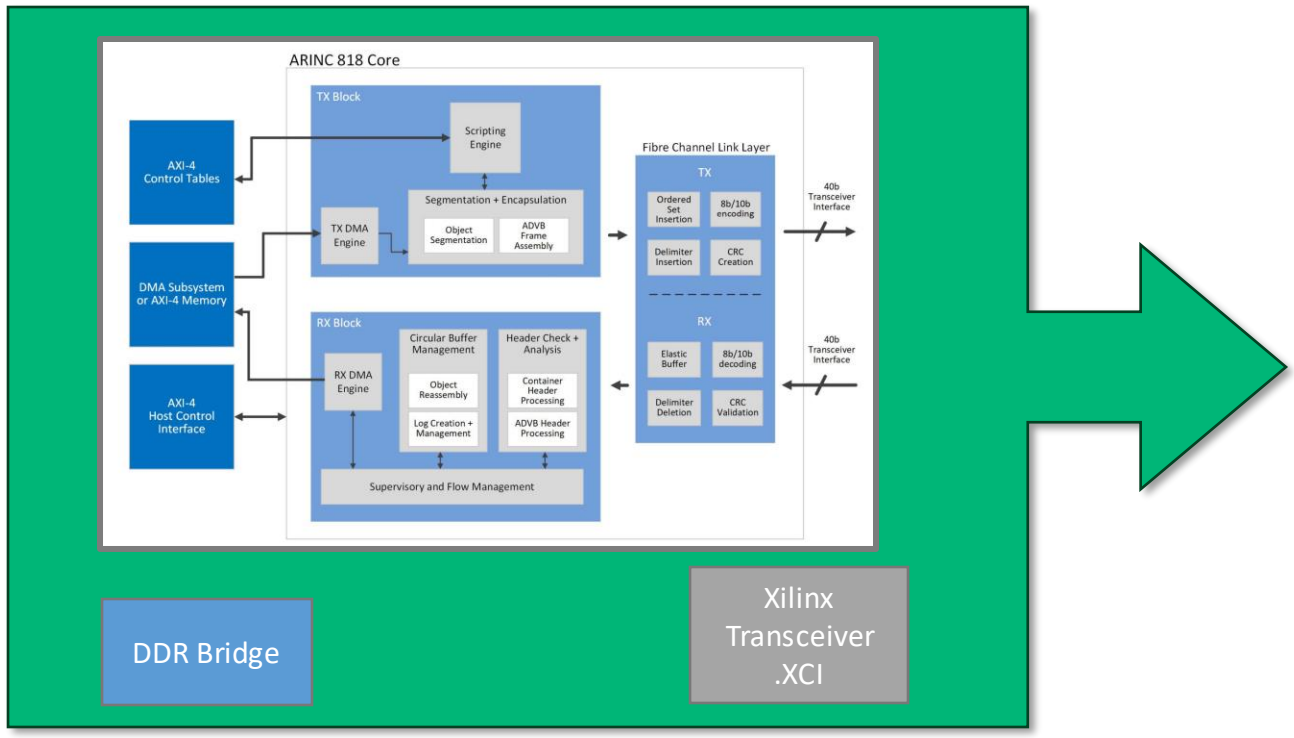


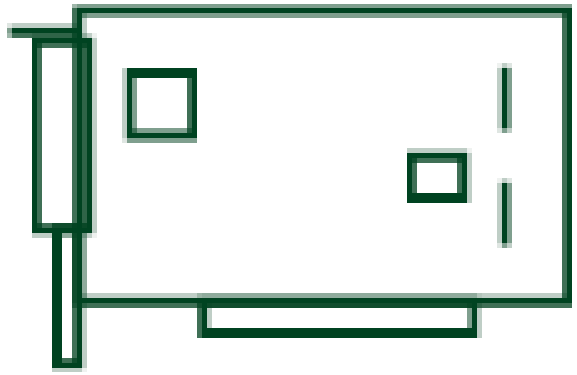


LabVIEW provides VIs to read and write registers on the AXI Bus.

Example ARINC-818

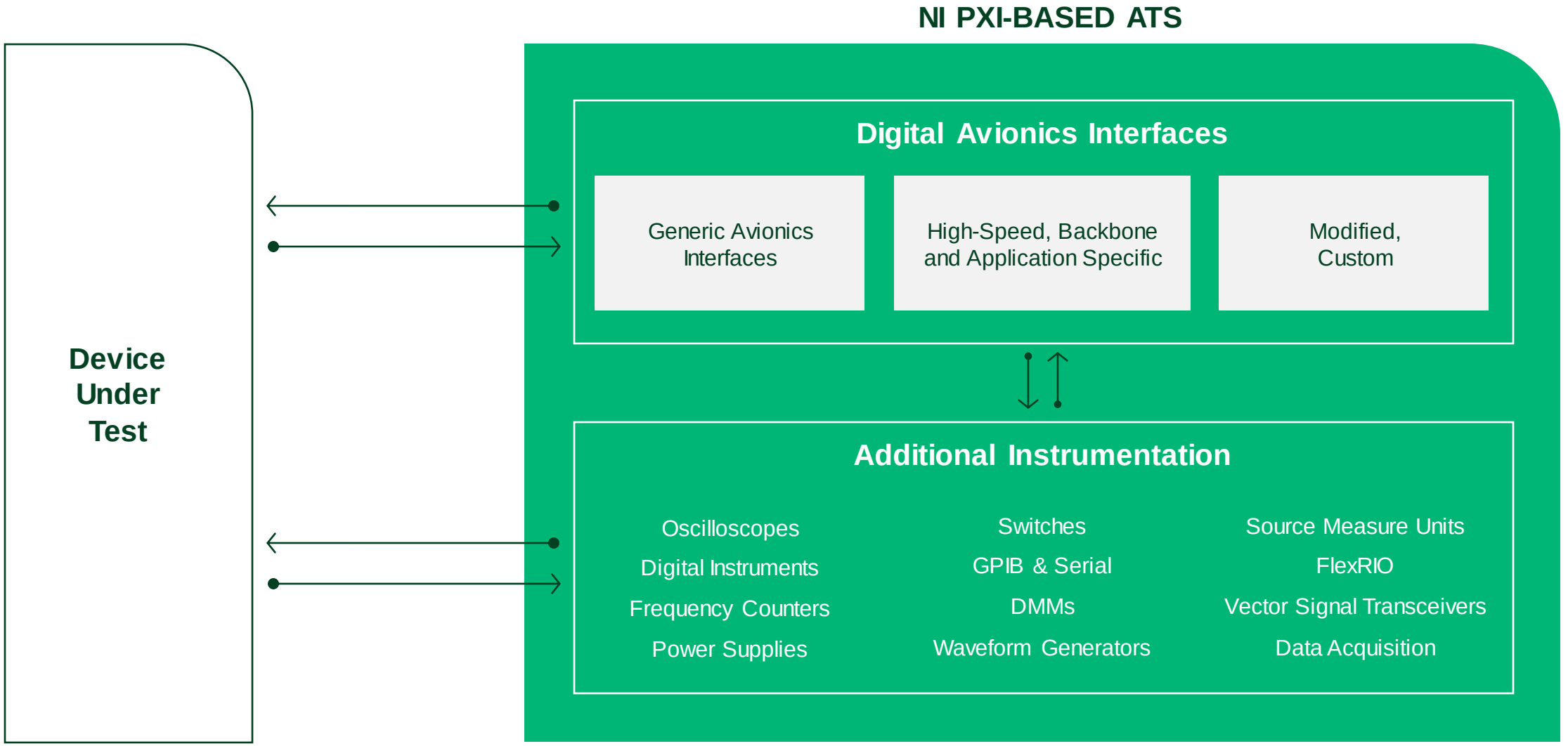
Socketed CLIP





PXIe 6592R

NI Digital Avionics Interfaces Offering



Summary

Digital avionics interfaces are routinely not considered with the same level of rigor as other instrumentation

While generic interfaces have become commoditized, high-speed and application specific protocols can introduce significant technical challenges

Whenever possible, test engineers should avoid custom solutions for manufacturing & depot test by leveraging COTS hardware, software, and IP

NI's customizable COTS hardware and software platform, along with key industry partnerships gives you customizability without additional design and maintenance cost, along with the system integration benefits of PXI