LabVIEW RIO Architecture – Overview and Technology Advancements

Russ Greenwood – NI Michigan
West Michigan Consultant’s Network
5/19/2014
Project Explorer
Manage and organize all system resources, including I/O and deployment targets

Deployment Targets
Deploy LabVIEW code to the leading desktop, real-time, and FPGA hardware targets

Instant Compilation
See the state of your application at all times, instantly

Front Panel
Create event-driven user interfaces to control systems and display measurements

Models of Computation
Combine and reuse .m files, C code, and HDL with graphical code

Hardware Connectivity
Bring real-world signals into LabVIEW from any I/O on any instrument

Parallel Programming
Create independent loops that automatically execute in parallel

Block Diagram
Define and customize the behavior of your system using graphical programming

Analysis Libraries
Use high-performance analysis libraries designed for engineering and science

Timing
Define explicit execution order and timing with sequential data flow

Accelerates Your Success
By abstracting low-level complexity and integrating all of the tools you need to build any measurement or control system
Field-Programmable Gate Array (FPGA)

- **Memory Blocks**: Store data sets or values in user defined RAM
- **Configurable Logic Blocks (CLBs)**: Implement logic using flip-flops and LUTs
- **Multipliers and DSPs**: Implement signal processing using multiplier and multiplier-accumulate circuitry
- **Programmable Interconnects**: Route signals through the FPGA matrix
- **I/O Blocks**: Directly access digital and analog I/O
The NI Approach to Flexible Hardware

We call this the LabVIEW RIO Architecture.

- Processor
  - Real-time OS
  - Application software
  - Networking and peripheral I/O drivers
  - DMA, interrupt, and bus control drivers

- FPGA
  - Application IP
  - Control IP
  - DSP IP
  - Specialized I/O drivers and interface
  - DMA controller

- Analog I/O
- Digital I/O
- Specialized I/O
- Custom I/O
- Bus Protocols

Highly Productive LabVIEW Graphical Programming Environment for Programming Host, FPGA, I/O, and Bus Interfaces
LabVIEW Reconfigurable I/O (RIO) Architecture

- Processor
- FPGA
- I/O
- I/O
- I/O
- Custom I/O

Compartmentalization:

CompactRIO & Single-Board RIO

PXI, PC RIO (R Series, FlexRIO)

ni.com
Use of LabVIEW RIO Architecture

- Control / Simulation / Data Acquisition
Use of LabVIEW RIO Architecture

- Drivven – Data Logging / Control Prototyping
Use of LabVIEW RIO Architecture
- Subaru – HIL / Simulation

"By adopting FPGA-based simulation using the NI hardware and software platforms, we achieved the simulation speed and model fidelity required for verification of an electric motor ECU. We reduced test time to 1/20 of the estimated time for equivalent testing on a dynamometer."
- Mr. Tomohiro Morita, FUJI Heavy Industries, Ltd.
Use of LabVIEW RIO Architecture

- High Speed Control - CERN
Using the software-designed NI PXI vector signal transceiver and the NI WLAN Measurement Suite, we improved test speeds by more than 200 times compared to traditional rack-and-stack instruments while significantly improving test coverage."

- Doug Johnson, Qualcomm Atheros

Use of LabVIEW RIO Architecture
- Qualcomm Atheros – WLAN Device Characterization

With traditional instrumentation, approximately 40 points of meaningful WLAN transceiver data were collected per iteration. The speed increase of the NI PXI vector signal transceiver triggered full gain table sweeps to acquire all 300,000 points.
Use of LabVIEW RIO Architecture

- Optimedica – Custom Timing and Triggering

“By programming in LabVIEW FPGA, we were able to vary the timing and power of each pulse to optimize for speed and precision.”

– Michael Wiltberger, OptiMedica Corporation
NI CompactRIO

- **Extreme Ruggedness:** -40 to 70 °C temperature range; 50 g shock, 5 g vibration
- **High Performance:** Up to 1.33 GHz, dual-core i7 processor
- **Comprehensive I/O:** Analog, digital, custom, specialty, bus communication

ni.com
NI CompactRIO

Highly Productive LabVIEW Graphical Programming Environment for Programming Host, FPGA, I/O, and Bus Interfaces
Connect to Any Sensor on Any Bus

100+ Industrial I/O Modules

- Accelerometer
- Strain gage
- Resistance
- Load cells
- Digital I/O and protocols
- Microphone
- Bus communications

- Thermocouples
- 4 to 20 mA
- Storage media
- RTD
- Engine control
- Industrial vision
- Motion control
The Redesigned CompactRIO

NI LabVIEW System Design
Program with LabVIEW Real-Time and LabVIEW FPGA modules
Quickly port existing LabVIEW applications

High Throughput and Performance
Dual-Core ARM 667 MHz processor
Xilinx 7 Series FPGA fabric with 85k logic cells
16 DMA FIFO channels for data streaming

Ultra Rugged
-40 to 70° C operating temperature range
50 g shock and 5 g vibration tolerance

Community and Code Reuse
NI Linux Real-Time Operating System
Integrate existing applications and libraries
Develop, debug, and deploy C/C++ code

ni.com
The New cRIO-9068

- 2x Gigabit Ethernet
- 667 MHz Dual-Core ARM Processor
- Artix-7 FPGA
- 512MB DDR3 RAM
- 1GB Storage
- 2x Gigabit Ethernet
- 667 MHz Dual-Core ARM Processor
- Artix-7 FPGA
- 512MB DDR3 RAM
- 1GB Storage
- 9-30 VDC Dual Input
- 31.3cm* x 10.1cm x 3.4cm
- -40 to 70° C Operating Temperature
- 5g Vibration, 50g Shock Tolerance
- RS-232
- USB 2.0
- RS-485
- 8x C Series IO

1 FPGA Specifications: 85K Logic Cells, 560KB Block Ram, 220 Multipliers
* 31.3cm x 10.1cm x 3.4cm
Performance Benchmarks

The cRIO-9068 has 4 times the performance of a similar 8 slot value CompactRIO system.
Linux Operating System Background

- Operating system created under the model of free and open source software development and distribution

- First released in 1991 for x86 but has been ported to other computer hardware platforms including ARM

- Common OS for servers and big mainframes

- Linux is used in a wide breath of embedded systems including:
  - Mobile phones (Android)
  - Tablets
  - Network routers
  - Televisions
  - Video game consoles
Linux Ecosystem

**Database**
- Raima
- MySQL
- SQLite
- MongoDB
- CouchDB

**Security**
- OpenVPN
- IP Tables
- System Logging
- fail2ban
- denyhost

**Code Re-use**
- C/C++
- Shell scripting
- Python
- Ruby
- Perl

**Connectivity**
- Isshd
- IPv6
- SNMP
- NTP
- netstat
“By leveraging the open Linux-based real-time operating system on a commercial off-the-shelf controller, we could port our existing Linux software components in a very short time. That saved us at least four months of development time.”

- Wolfram Koerver, executive director of S.E.A.
LabVIEW Programmed NI CompactRIO

cRIO-9002
cRIO-9004
cRIO-9072
cRIO-9074
cRIO-9076
cRIO-9012
cRIO-9014
cRIO-9024
cRIO-9025
cRIO-9081
cRIO-9082
cRIO-9068
Questions?