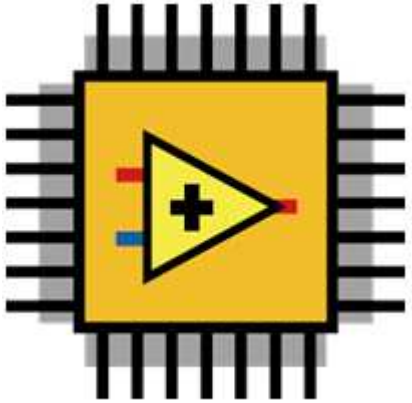
LabVIEW Real-Time and LabVIEW FPGA Programming Tips & Tricks how to get started

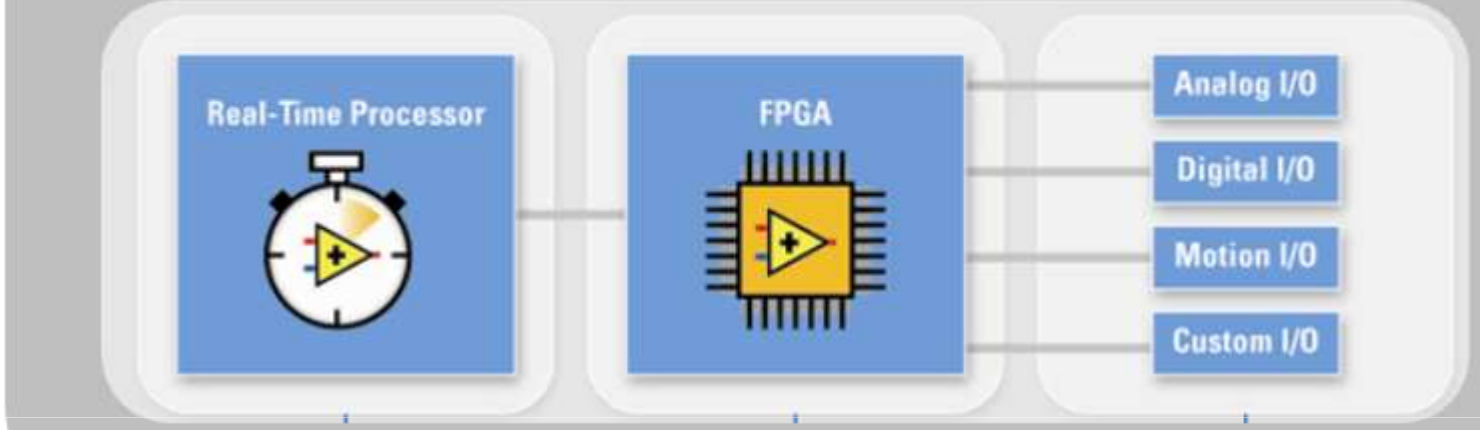
[Presenter]

<title>



NATIONAL INSTRUMENTS

Reconfigurable I/O (RIO) Architecture

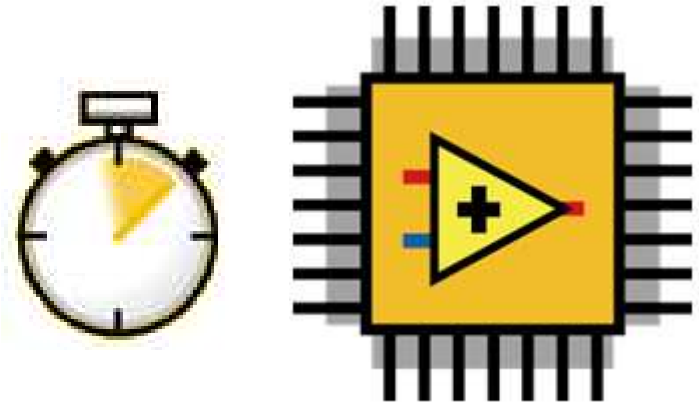


NATIONAL INSTRUMENTS

Two programming modes available

What is the Scan Mode and When to use it?

What about FPGA mode?

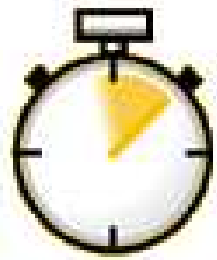


NATIONAL INSTRUMENTS

What is the Scan Mode and When to use it?

We have added a Scan Engine to LabVIEW and CompactRIO to make it easier to program I/O while retaining the flexibility of FPGAs.

- ✓ Fast time to first measurement
- ✓ Synchronous scanned I/O updates
- ✓ Automatic network published I/O values
- ✓ Ideal for single-point I/O updates < 1kHz

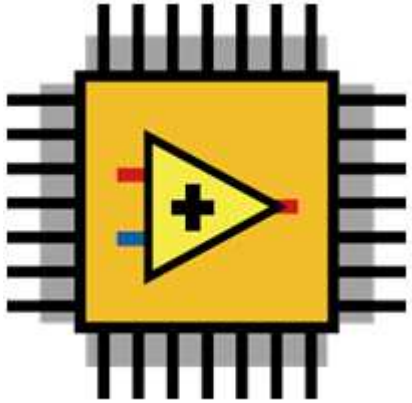


NATIONAL INSTRUMENTS

What is the FPGA Mode and When to use it?

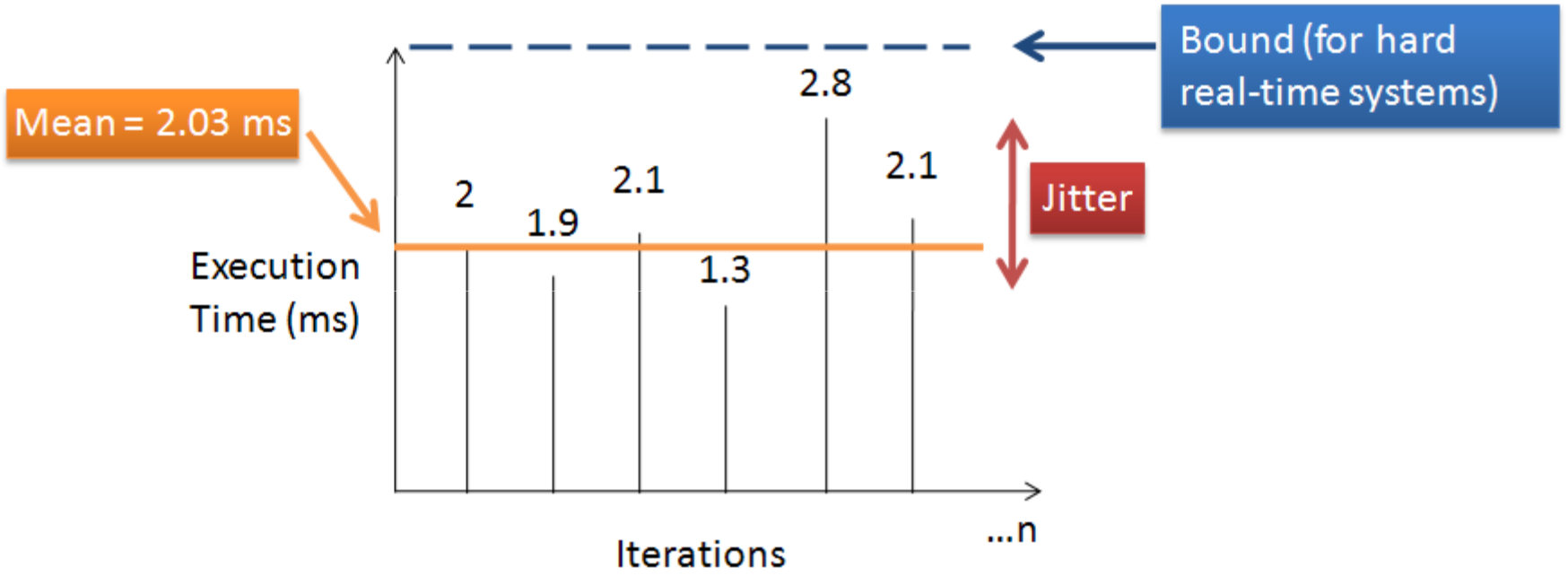
In FPGA mode you need to program I/O. More complex but you will gain much more performance.

- ✗ Slower time to first measurement
- ✓ Fast I/O updates
- ✓ Tight Determinism
- ✓ Parallelism
- ✓ Ideal for I/O updates > 1kHz



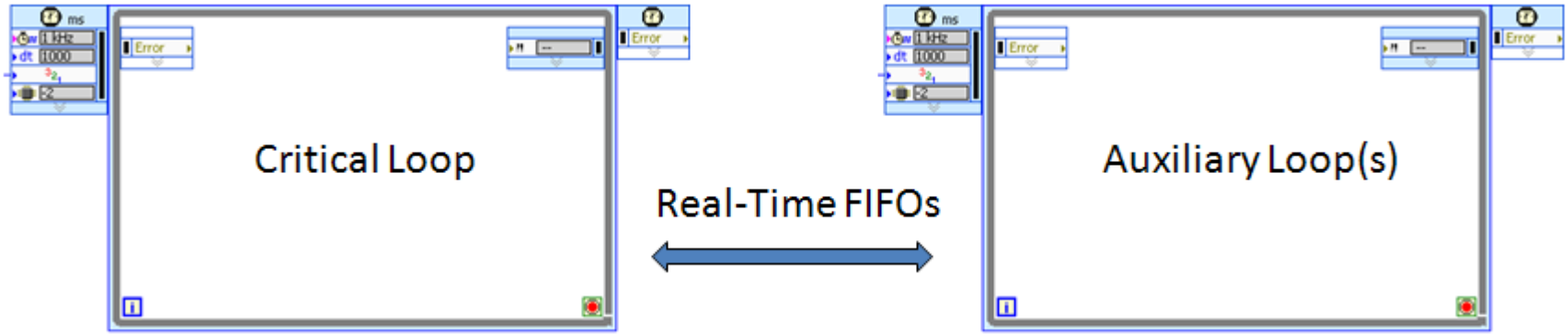
NATIONAL INSTRUMENTS

Programming With Jitter in Mind



NATIONAL INSTRUMENTS

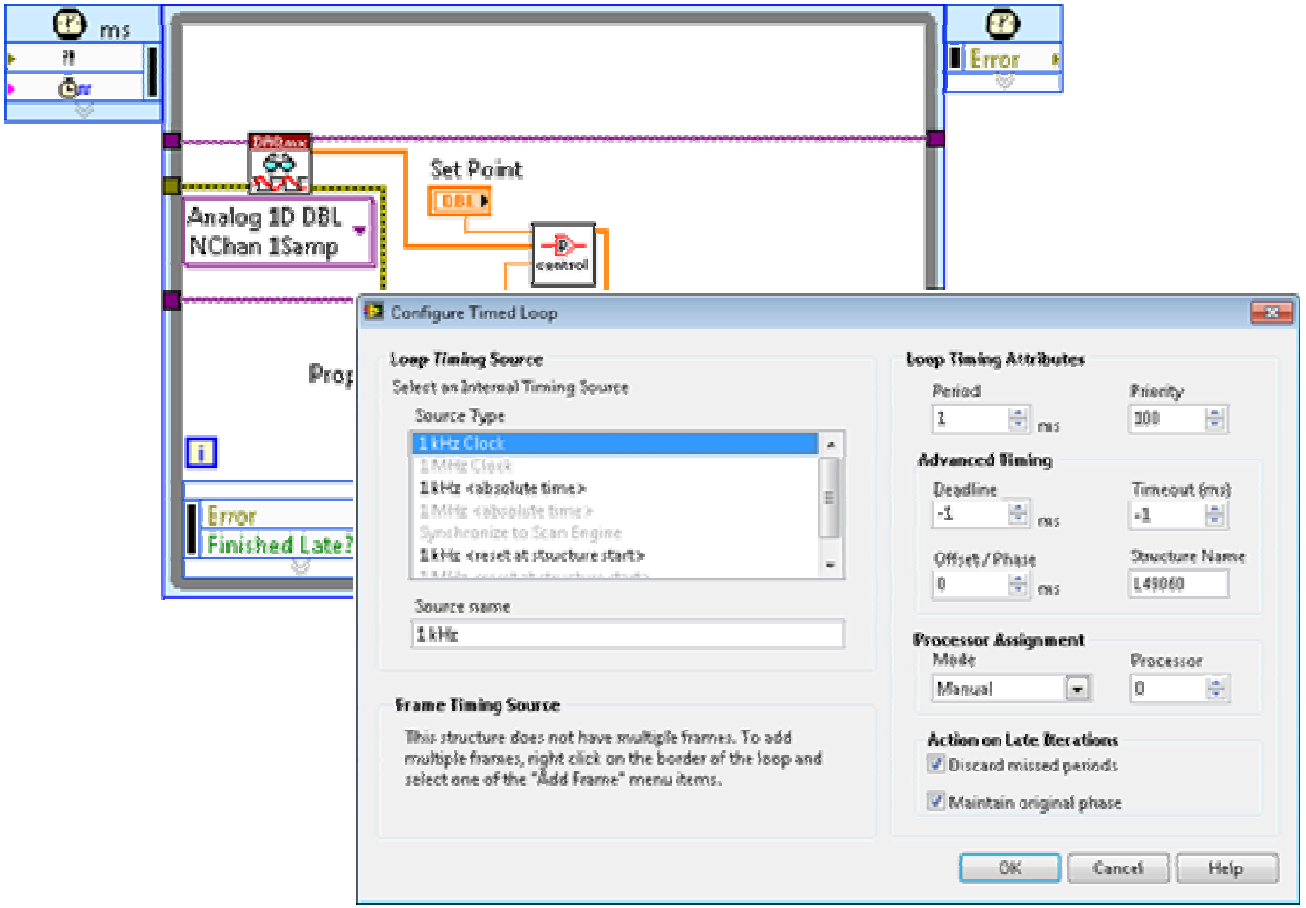
Programming With Jitter in Mind



Deterministic Operations	Nondeterministic Operations
• PID control • Real-time FIFO communication • Safety logic • Calls to deterministic drivers or libraries	• File I/O • Network or serial communication • Memory allocation o Dynamically sized data types • Calls to nondeterministic drivers or libraries

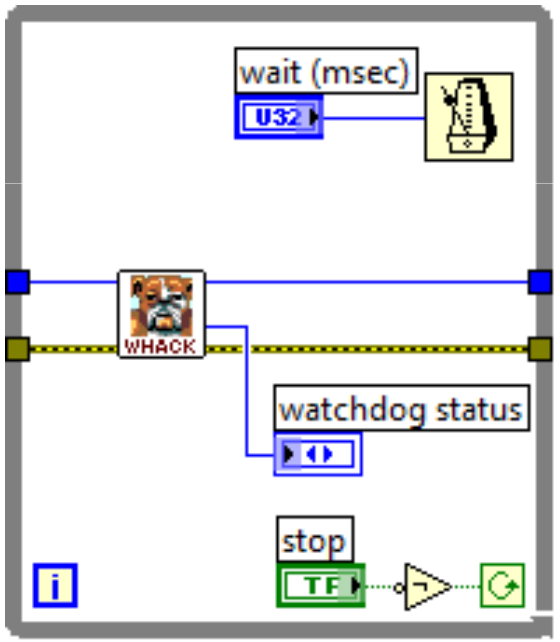
NATIONAL INSTRUMENTS

Programming With Jitter in Mind



Planning for the Worst-Case Scenario

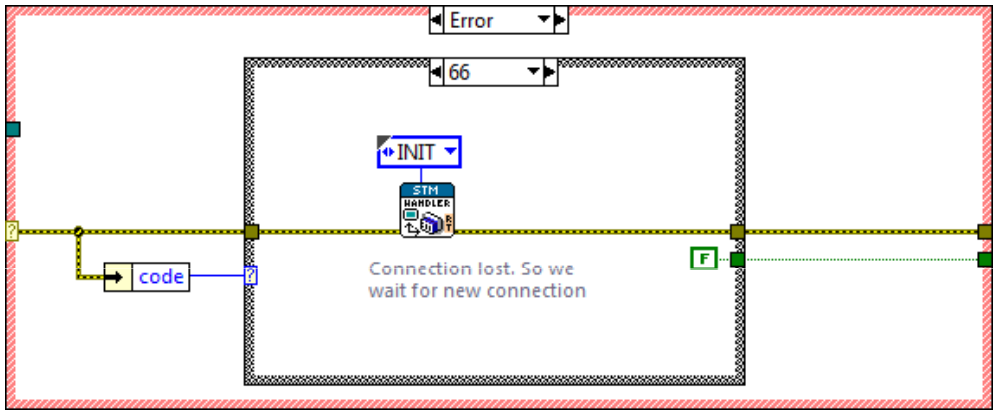
- Use watchdog timer API to auto-restart hardware on software hang
- Can also be programmed to assert a PXI trigger line or generate an occurrence
- Think carefully about FPGA safe states and startup conditions



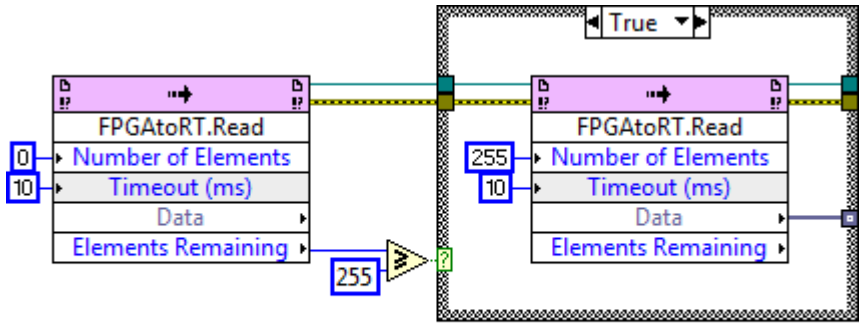
NATIONAL INSTRUMENTS

Planning for the Worst-Case Scenario

Handle communication Errors



Handle Data Buffer Underflow/Overflow



Read Zero Elements
Just to get Remaining Elements

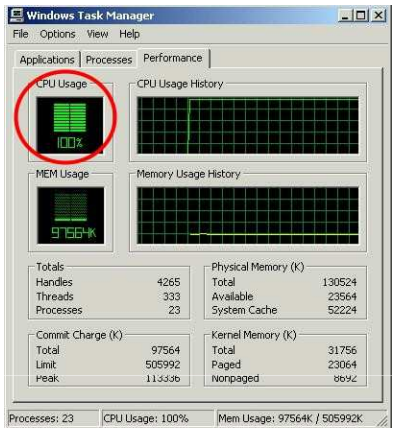
Read data only if there is enough
data available

NATIONAL INSTRUMENTS

Planning for the Worst-Case Scenario

Use CompactRIO User Led to indicate system status:

Working With Constrained Resources



Disk Space | RAM | CPU Bandwidth | FPGA Gates

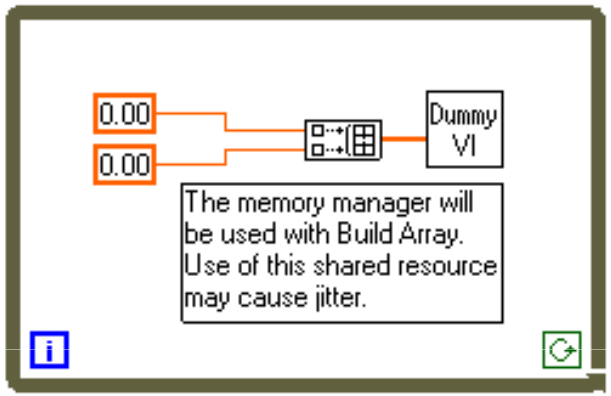
EFFECT

Lost Data	Crash	Starvation	Failure to Create Bitfile

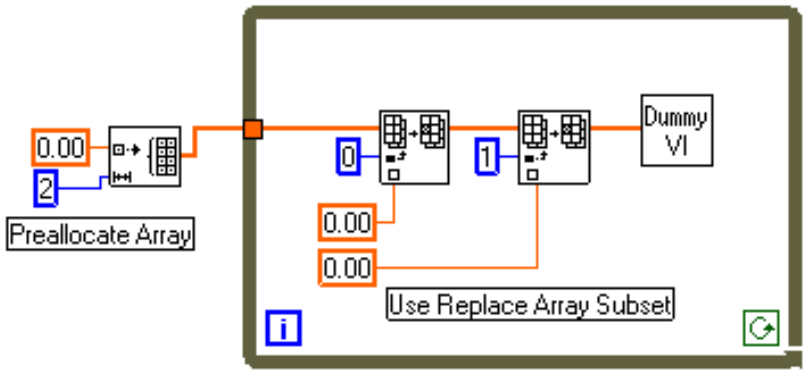
NATIONAL INSTRUMENTS

Working With Fixed-Size Data

- LabVIEW Real-Time:
  - Reduces jitter due to allocations
  - Reduces CPU load
  - Improves reliability



- LabVIEW FPGA
  - Mandated; no dynamically sized data

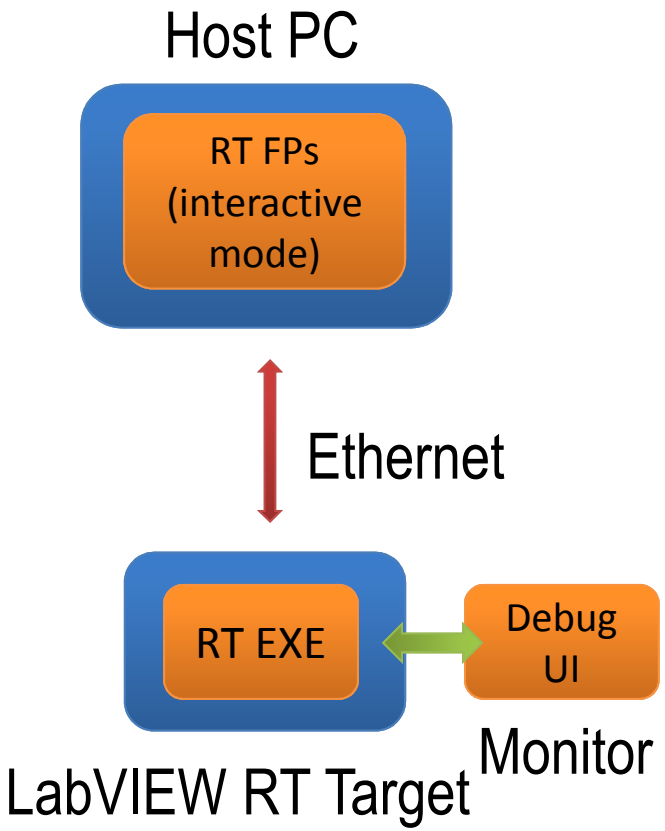


NATIONAL INSTRUMENTS

Remote Debugging

LabVIEW Real-Time

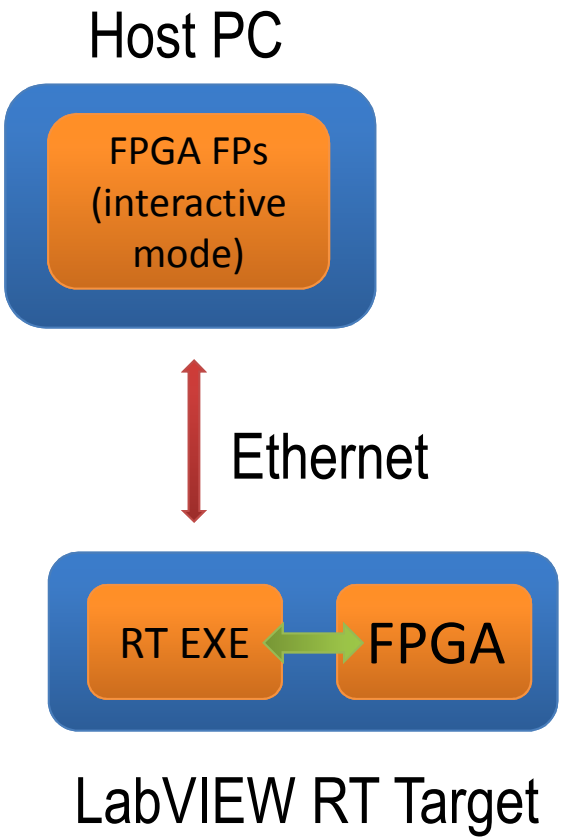
- Can use standard LabVIEW debugging tools on host
- Front panel of LabVIEW Real-Time VI automatically updates via network
- Possible to programmatically print messages to RT console



Remote Debugging

LabVIEW FPGA

- **Cannot** use standard LabVIEW debugging tools (HW vs. SW)
- Front panel of LabVIEW FPGA VI automatically updates via network
  - Instrument VI for debugging
- For functional debugging, use FPGA simulation



LabVIEW FPGA Compile Farm Toolkit

Development PC

Compile Server and Workers

High-Performance Cloud

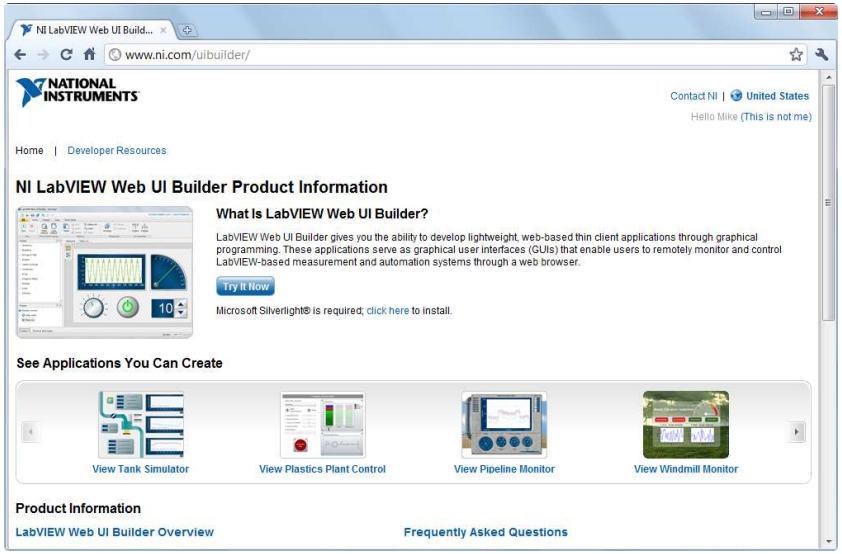
Single CPU Compile → On-Site Compile Farm → Cloud Compilation

NATIONAL INSTRUMENTS

TIP: Use UI Builder for thin clients

✓ No Software installation needed

✓ Works with Web Browser that supports MS Silverlight



NATIONAL INSTRUMENTS

ni.com/uibuilder

Synthesizing vs. Compiling

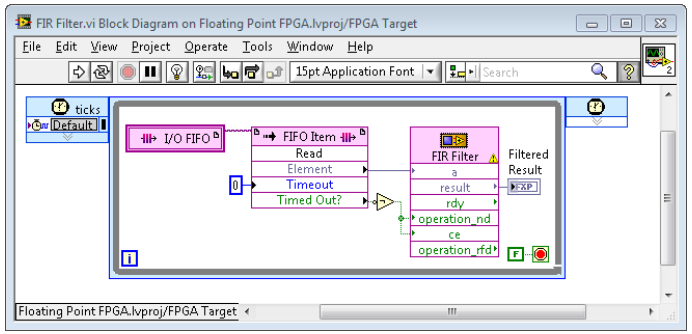
Reusing IP

- IPNet sharing
- IP Integration Node
  - Reuse HDL
  - Import Xilinx CORE Generator libraries
- CLIP Node
  - Asynchronous IP

IPNet Browse, Download, and Share LabVIEW FPGA IP

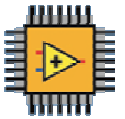
Signal Processing and Measurements

Name	LabVIEW Version	IP or Example	Source	Code Maturity
Butterworth Filter	8.2-2009	IP	LabVIEW FPGA	5
Notch Filter	8.5-2009	IP	LabVIEW FPGA	5
Dolph-Chebyshev Filter (FIR)	8.2-2009	IP	DFD Toolkit	5
Kaiser Window Method Filter (FIR)	8.2-2009	IP	DFD Toolkit	5
Equi-Ripple Filter (Remez) (FIR)	8.2-2009	IP	DFD Toolkit	5
Time-Domain Window Filter (FIR)	8.2-2009	IP	DFD Toolkit	5
Bessel Filter (IIR)	8.2-2009	IP	DFD Toolkit	5
Chebyshev Filter (IIR)	8.2-2009	IP	DFD Toolkit	5



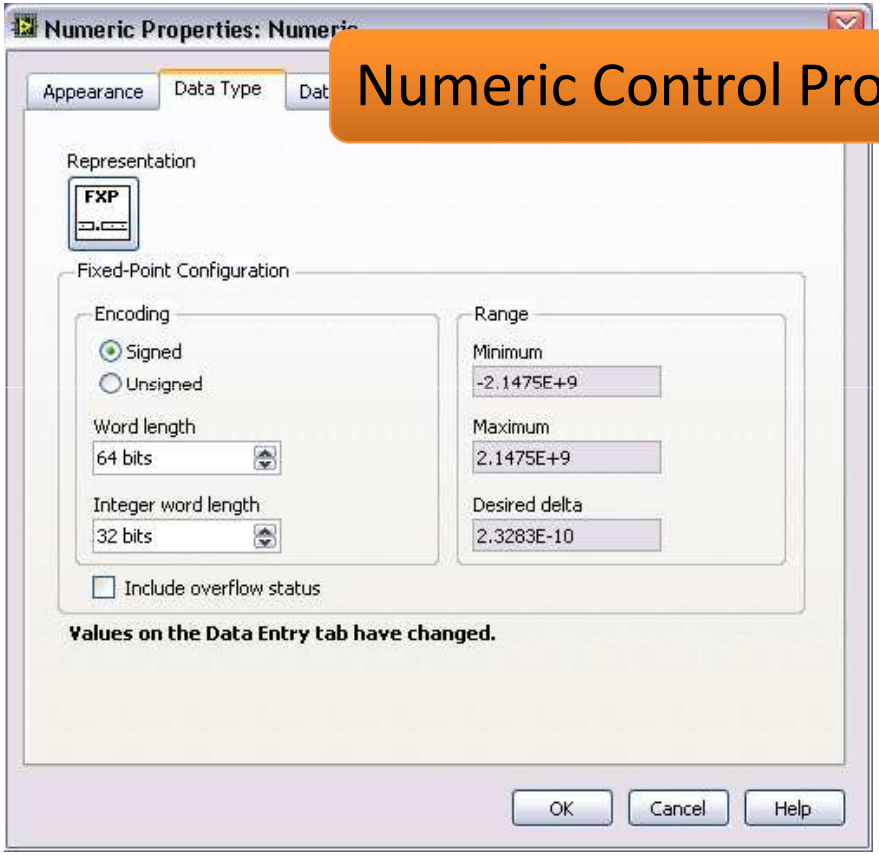
NATIONAL INSTRUMENTS

Working With Fixed-Point Math



	Floating Point	Fixed Point
Resource Requirements for FPGA Computations	High	Low
Radix Point	“Slides” around	Fixed; a set number of bit positions after radix point
Precision	Changes as radix point moves	Fixed
Range	Very large (~3.4 x 10^38 for single)	Small (less than int - no exponent)
Potential for Overflow	Low due to large range	Higher due to small range

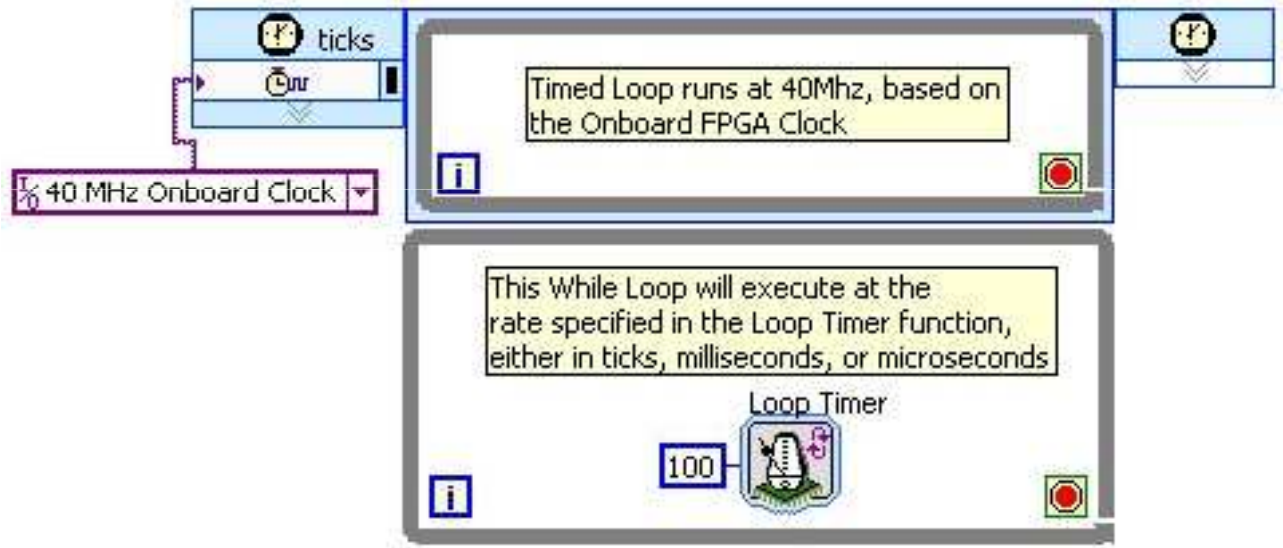
Working With Fixed-Point Math



Numeric Control Property Page

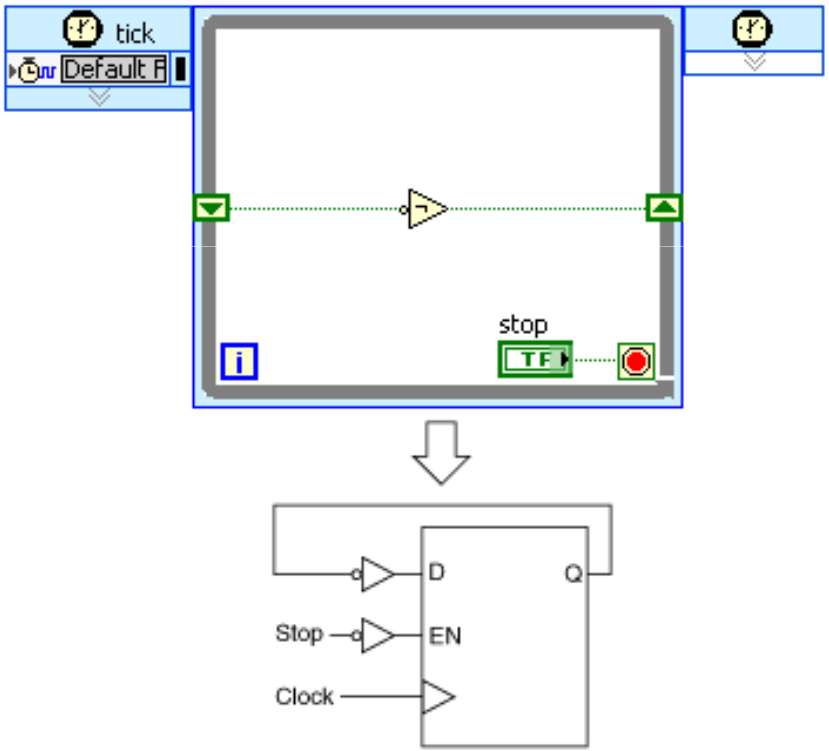
NATIONAL INSTRUMENTS

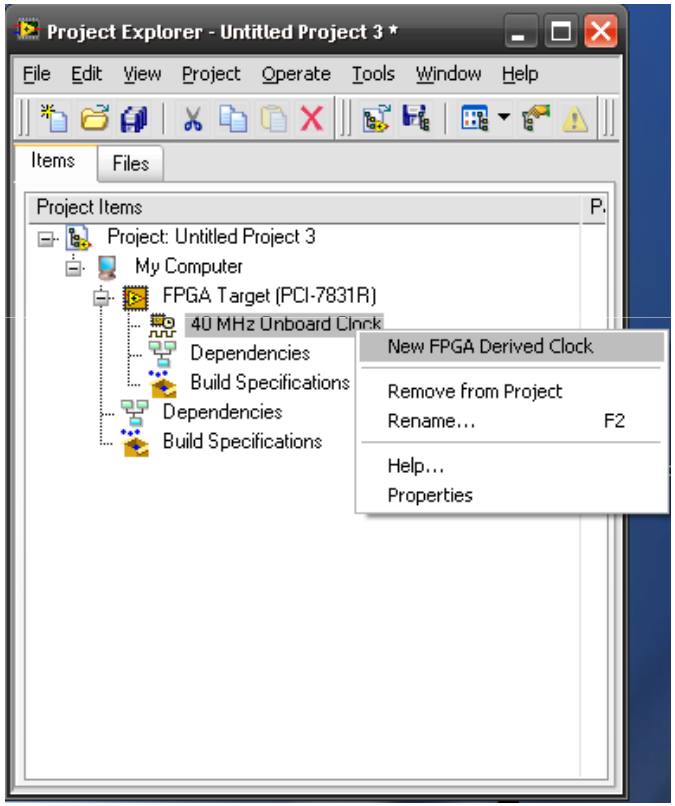
Understanding Clocks and Hardware Concurrency



Understanding Clocks and Hardware Concurrency

Single-Cycle Timed Loop (SCTL)

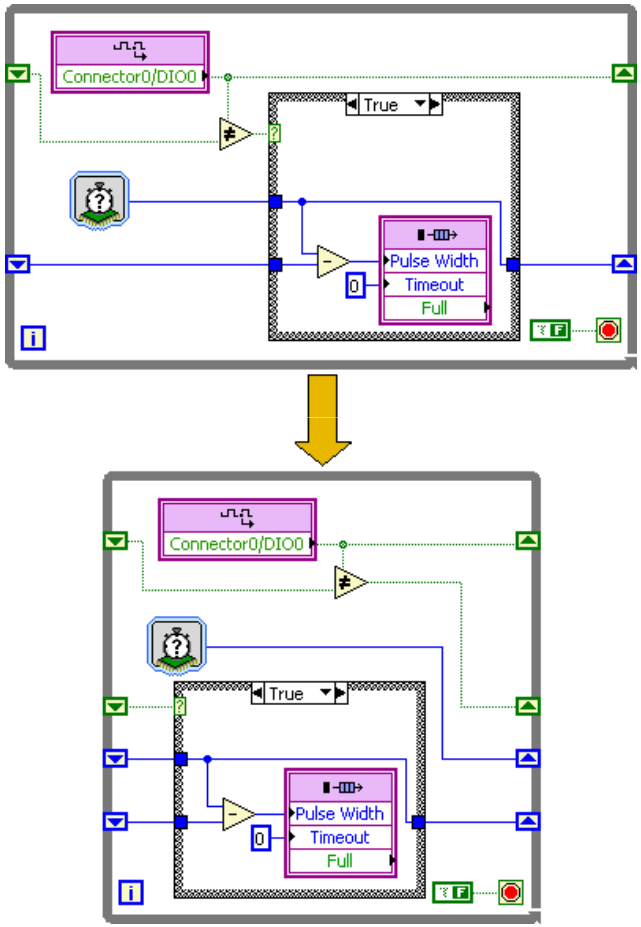




NATIONAL INSTRUMENTS

Understanding Clocks and Hardware Concurrency

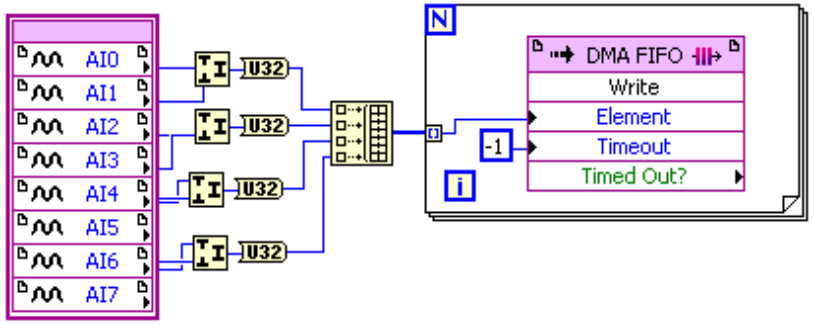
- Potentially unlimited parallelism in FPGA hardware (except for space constraints)
- Use techniques such as pipelining to maximize clock rate



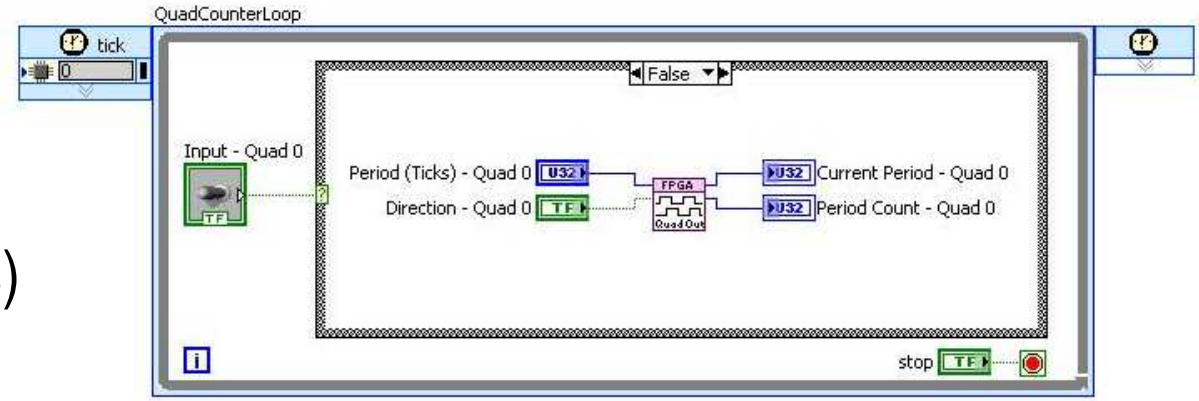
NATIONAL INSTRUMENTS

Transferring Data to the Host

DMA FIFOs

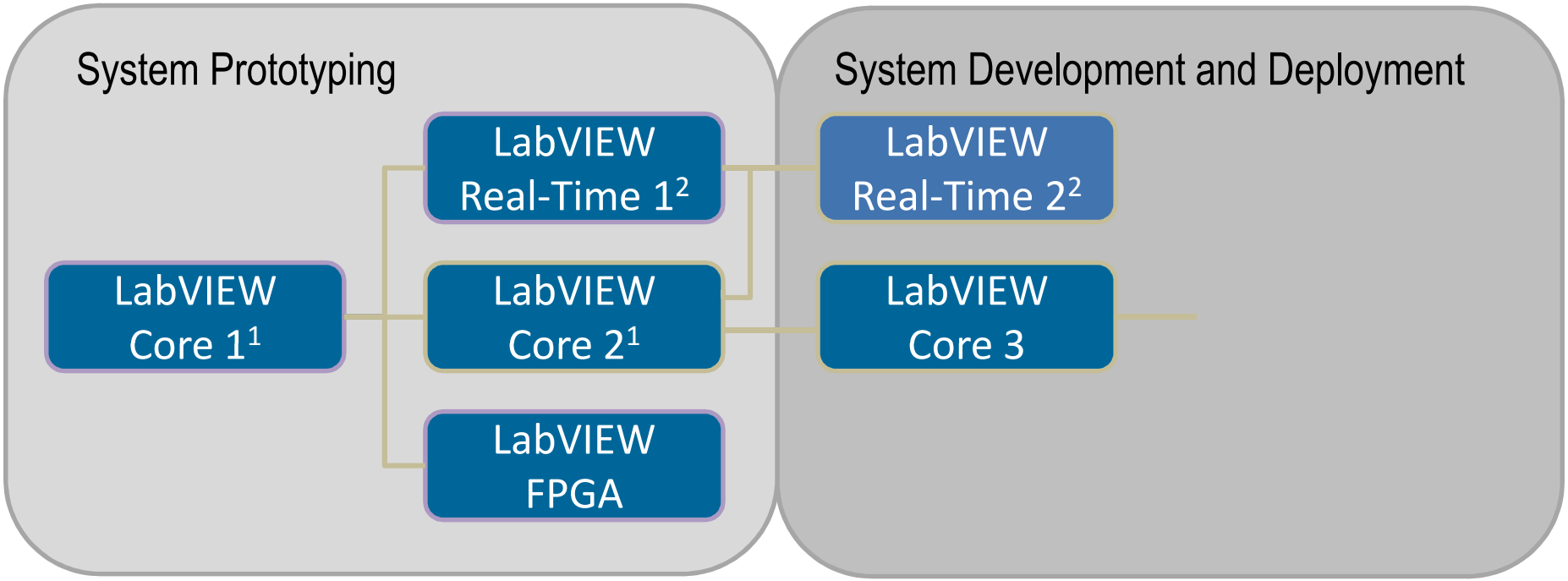


Registers
(FPGA VI Indicators)



NATIONAL INSTRUMENTS

Recommended Training Courses for LabVIEW FPGA and LabVIEW Real-Time



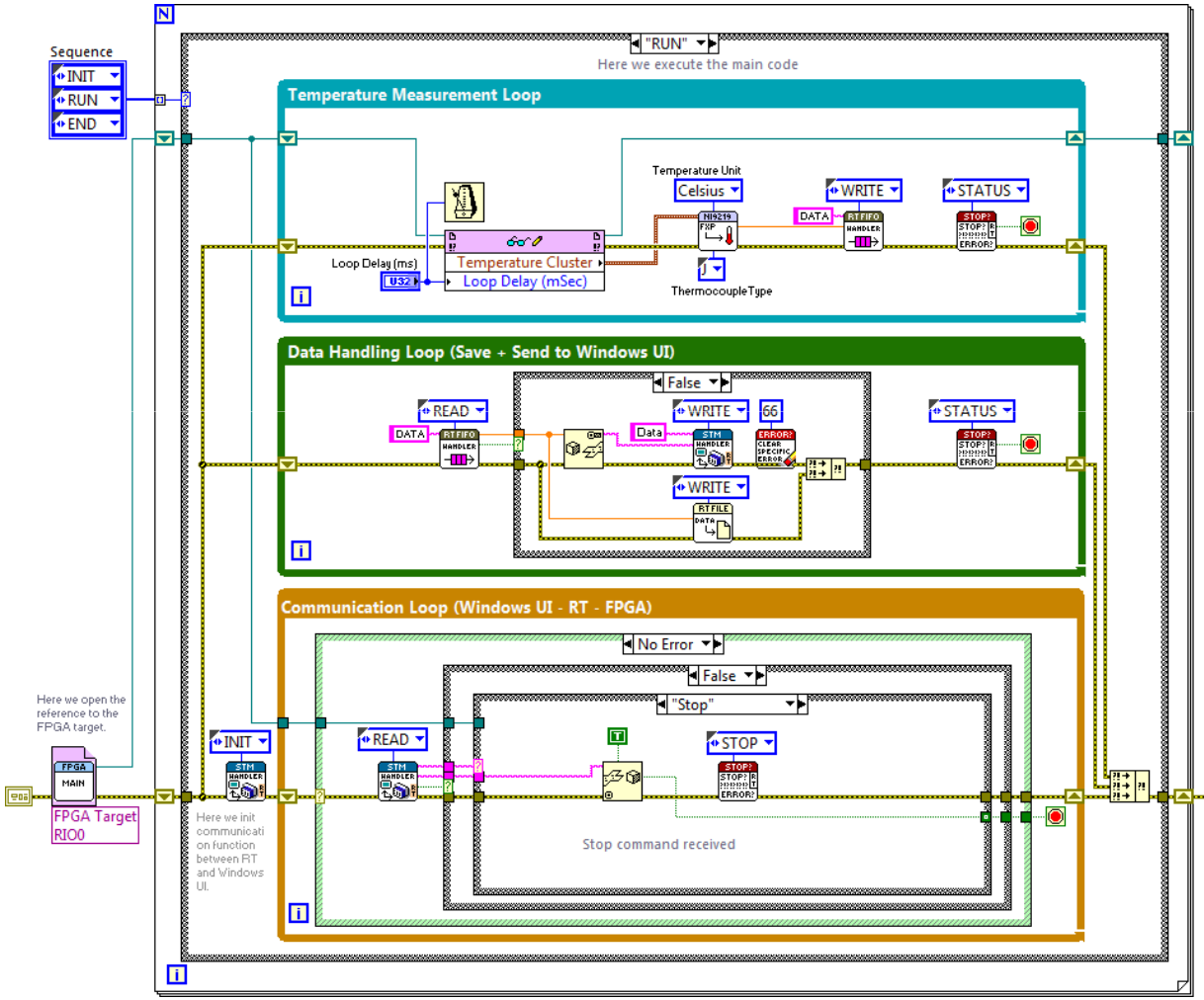
[1]LabVIEW Core 1 and 2 fit conveniently in a week, so you may choose to take both LabVIEW Core classes prior to LabVIEW FPGA and LabVIEW Real-Time 1.

[2]If you are working in Windows OS, you may choose not to take real-time training.

- Required
- Recommended
- Optional

NATIONAL INSTRUMENTS

RT Architecture Example



- Measure data
- Send data with RT FIFO

- Save data to cRIO
- Send data to Windows UI

-TCPIP Communication with Windows UI

NATIONAL INSTRUMENTS