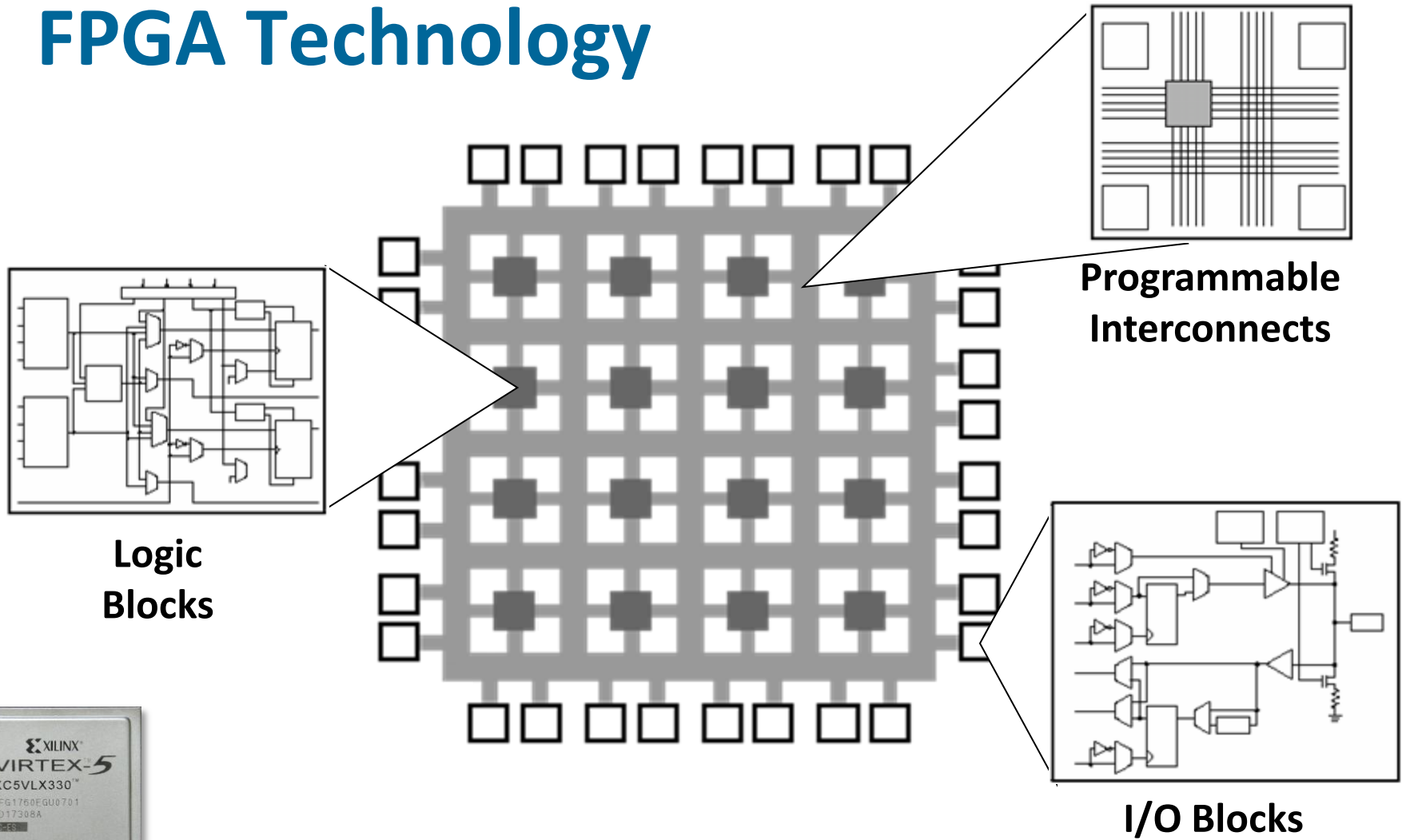




# Agenda

- **Programming FPGAs**
- Why Are They Useful?
- NI FPGA Hardware
- Common Applications for FPGAs
- How to Learn More

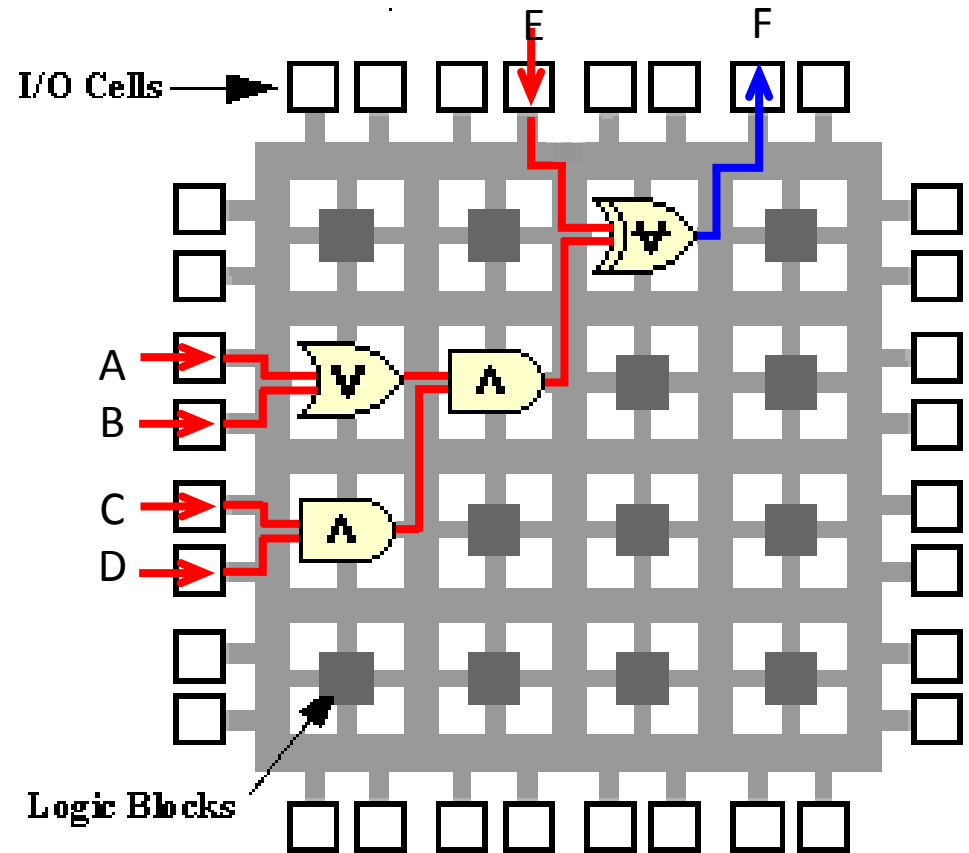
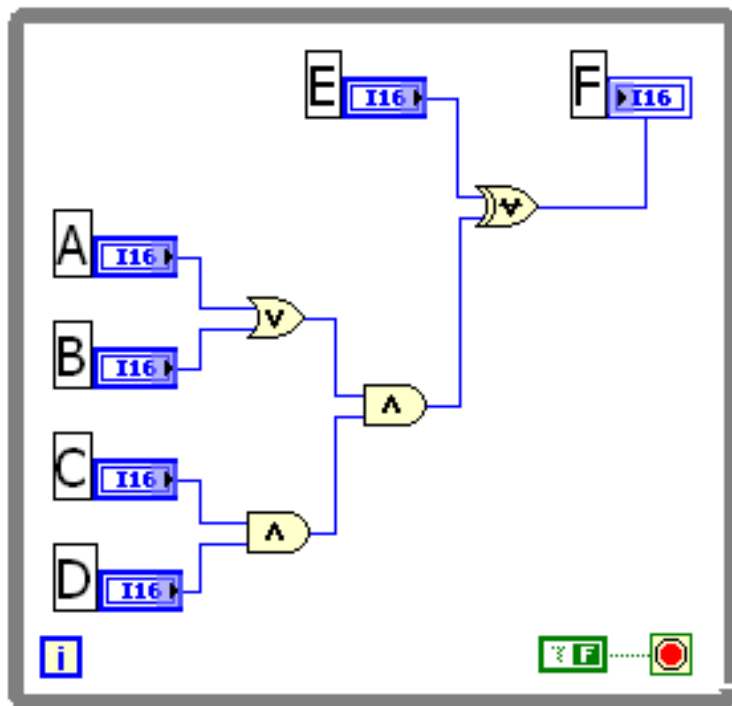
# FPGA Technology



# FPGA Logic Implementation

Implementing Logic on FPGA:  $F = \{(A+B)CD\} \oplus E$

LabVIEW FPGA Code

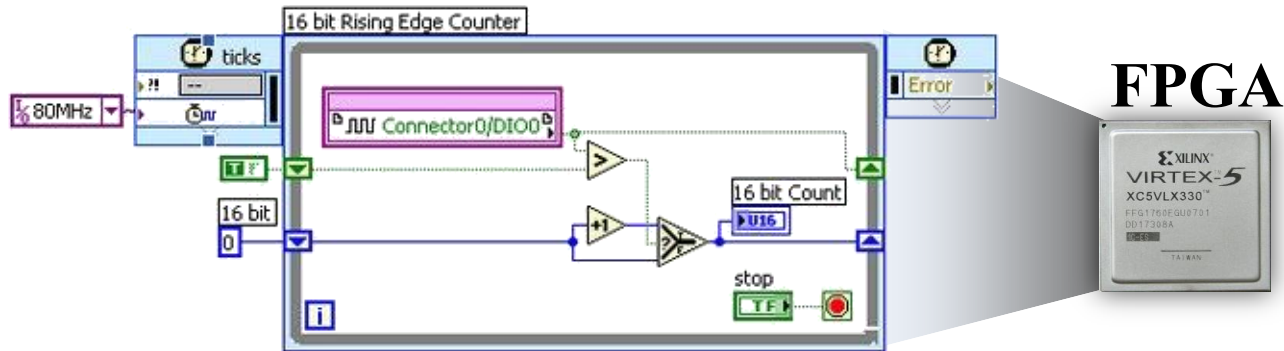


# Demo

- Filter



# LabVIEW FPGA Module



- LabVIEW code is translated to hardware circuitry implemented on the FPGA
- Natural representation of FPGA logic

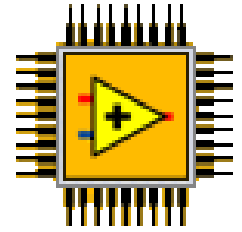
# Demo

- Simple I/O

# Agenda

- Programming FPGAs
- **Why Are They Useful?**
- NI FPGA Hardware
- Common Applications for FPGAs
- How to Learn More

# Why Are They Useful?

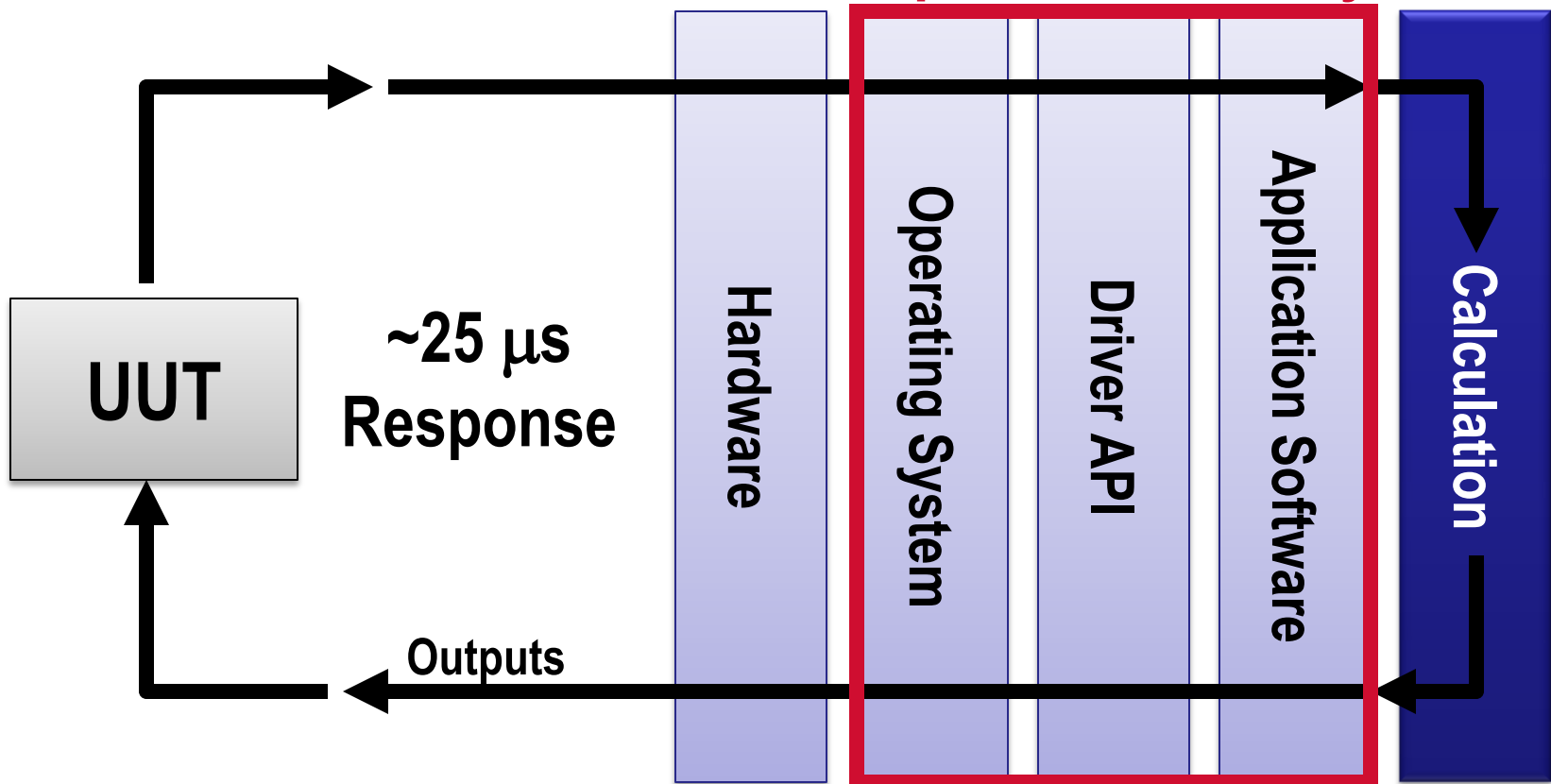


- ***True Parallelism*** – Provides parallel tasks and pipelining
- ***High Reliability*** – Designs become a custom circuit
- ***High Determinism*** – Runs algorithms at deterministic rates down to 25 ns (faster in many cases)
- ***Reconfigurable*** – Create new and alter existing task-specific personalities



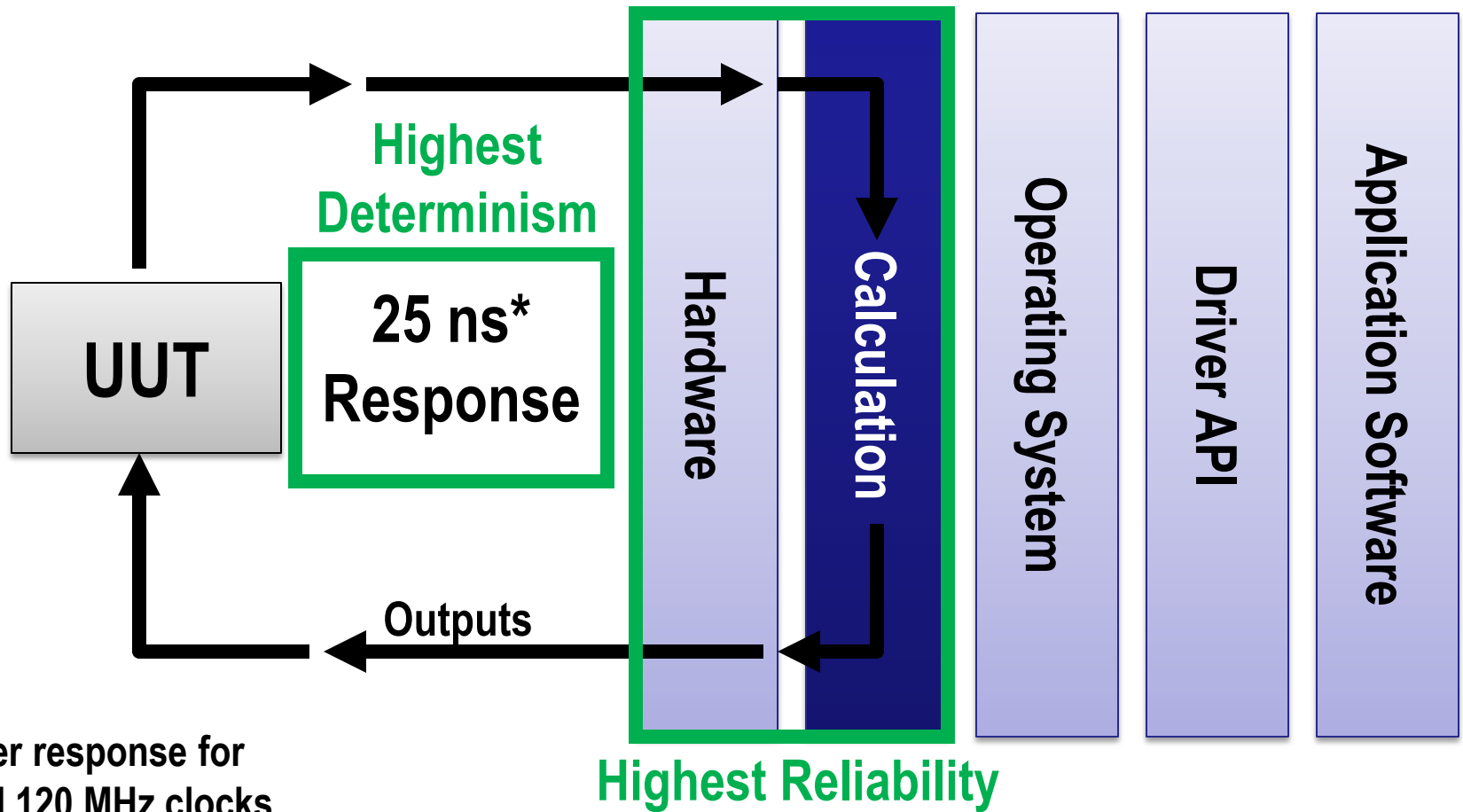
# High Reliability and Determinism

Decision Making in **Software** Multiple Software Layers



# High Reliability and Determinism

## Decision Making in Hardware

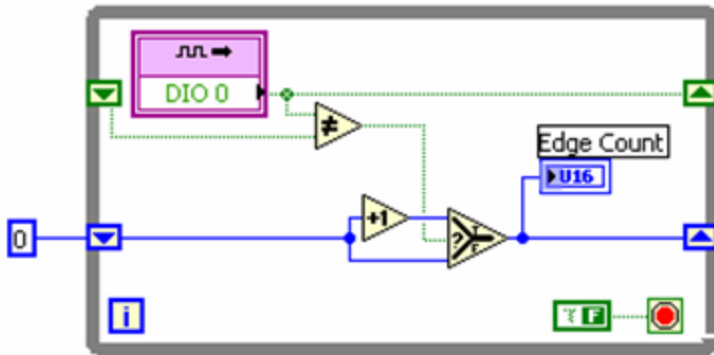
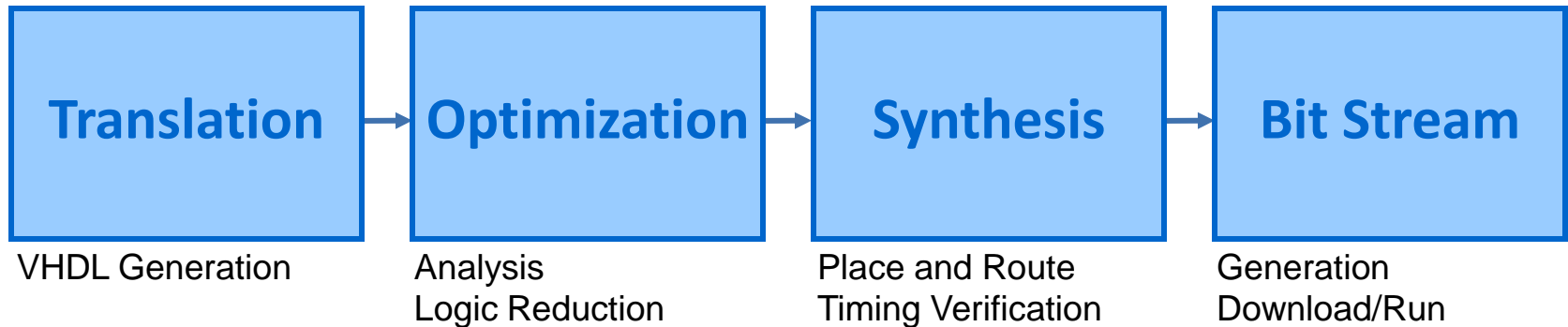


\* Faster response for  
80 and 120 MHz clocks

# Demo

- AI, AO, Custom Threshold logic

# From LabVIEW to Hardware

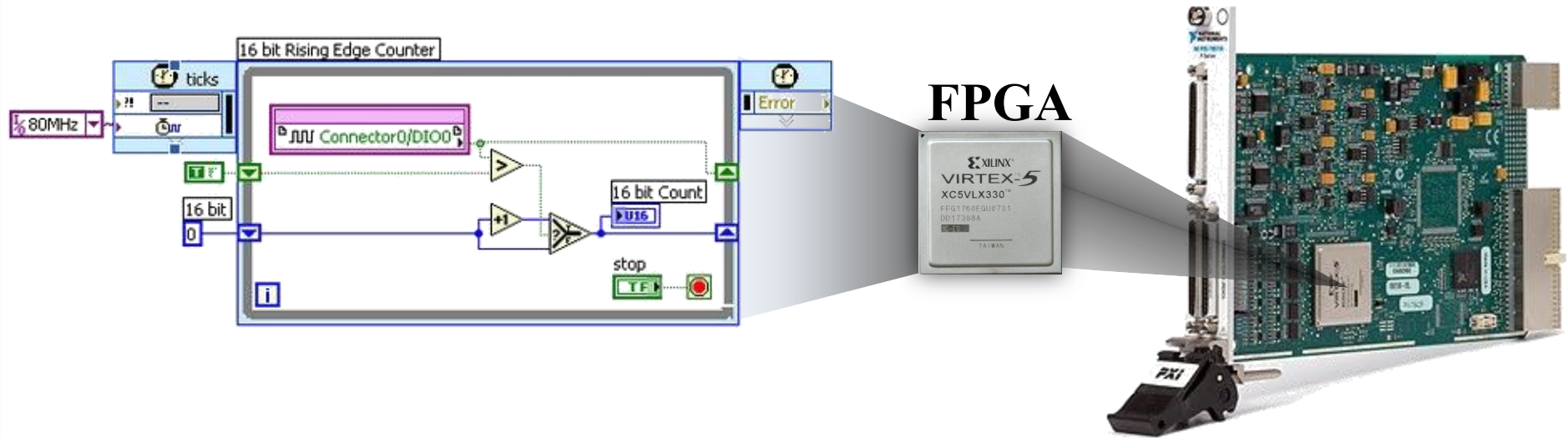


```
0010100101010001010
1001010010010100101
0010010101010111101
0010110101010001010
1010010100101001001
0100100010010010000
```

# Agenda

- Programming FPGAs
- What Are FPGAs and Why Are They Useful?
- **NI FPGA Hardware**
- Common Applications for FPGAs
- How to Learn More

# What Is RIO Technology?



- Use NI LabVIEW to design custom hardware circuitry with off-the-shelf devices

# NI LabVIEW FPGA Hardware Targets



## R Series Multifunction RIO

- General Purpose I/O for Measurement and Control

## NI CompactRIO

- Industrial Control and Monitoring

## NI SingleboardRIO

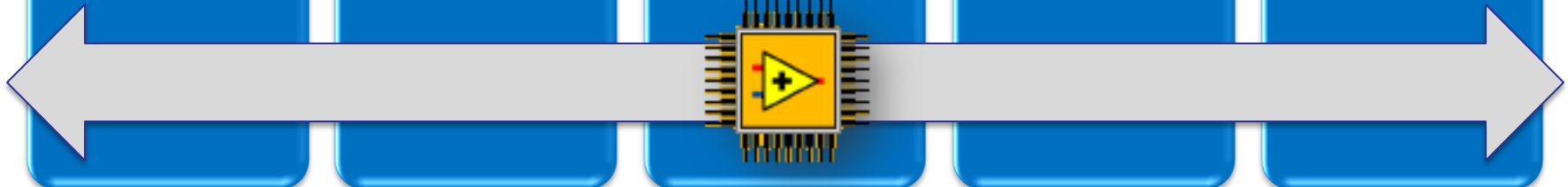
- Embedded Systems

## NI FlexRIO

- Manufacturing Test and Design Validation

## Other

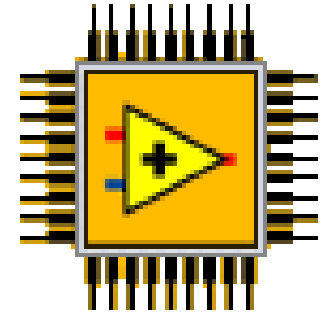
- RIO IF Transceiver
- PCIe Framegrabbers
- Compact Vision System



# Agenda

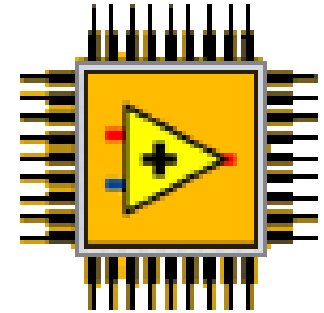
- Programming FPGAs
- What Are FPGAs and Why Are They Useful?
- NI FPGA Hardware
- **Common Applications for FPGAs**
- How to Learn More

# Common Applications



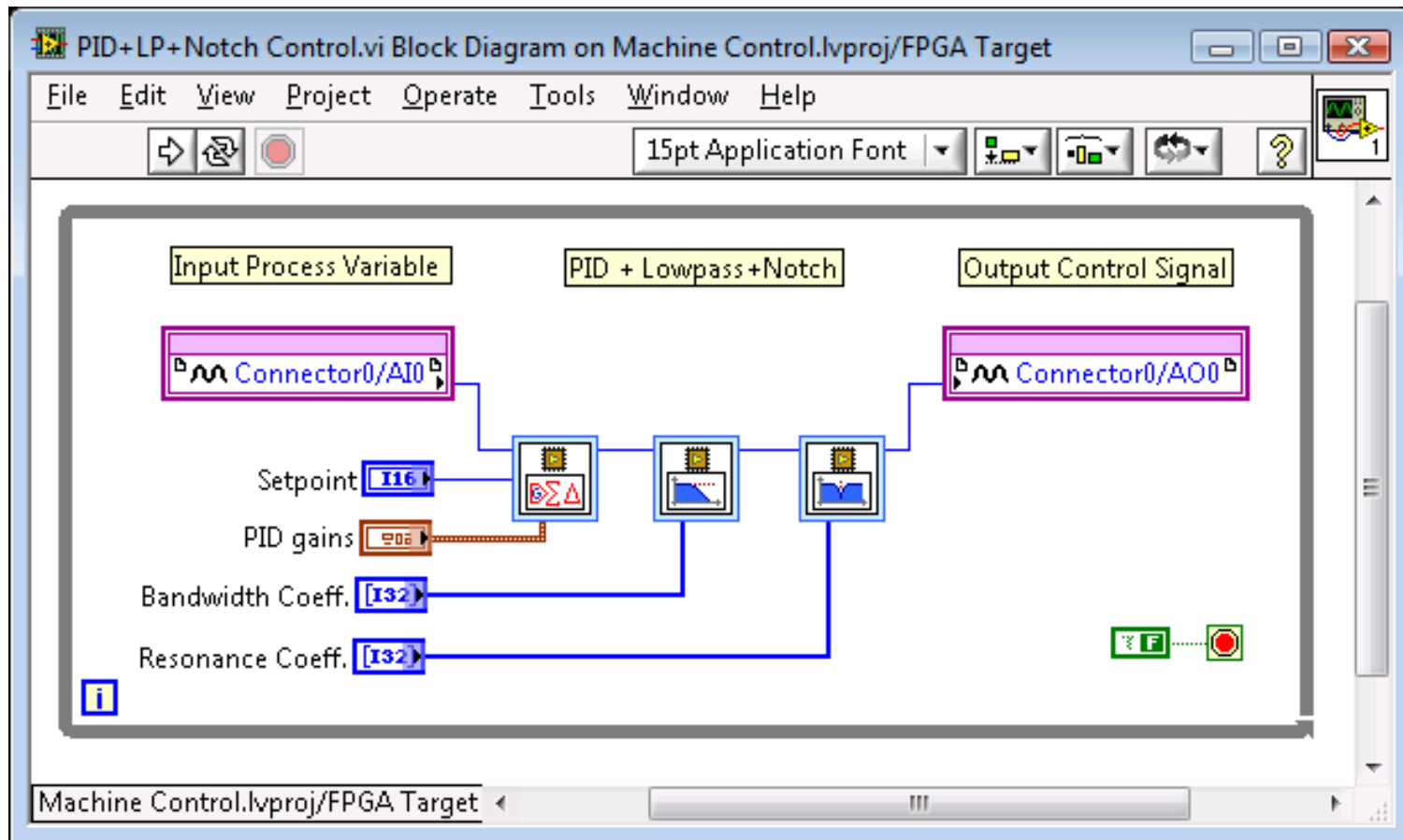
- High-speed control
- Custom DAQ
- Digital communication protocols
- Sensor simulation
- Onboard processing and data reduction

# Common Applications



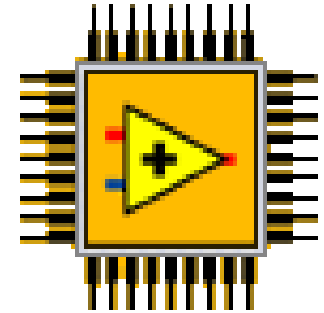
- High-speed control
- Custom DAQ
- Digital communication protocols
- Sensor simulation
- Onboard processing and data reduction
- Coprocessing

# High-Speed Control



*About 200 kHz loop rate*

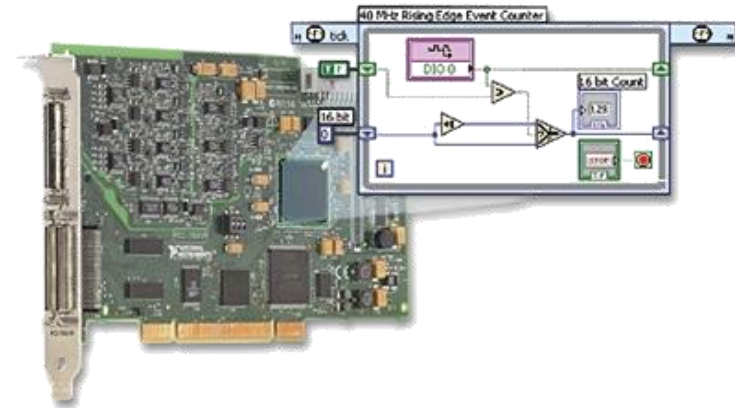
# Common Applications



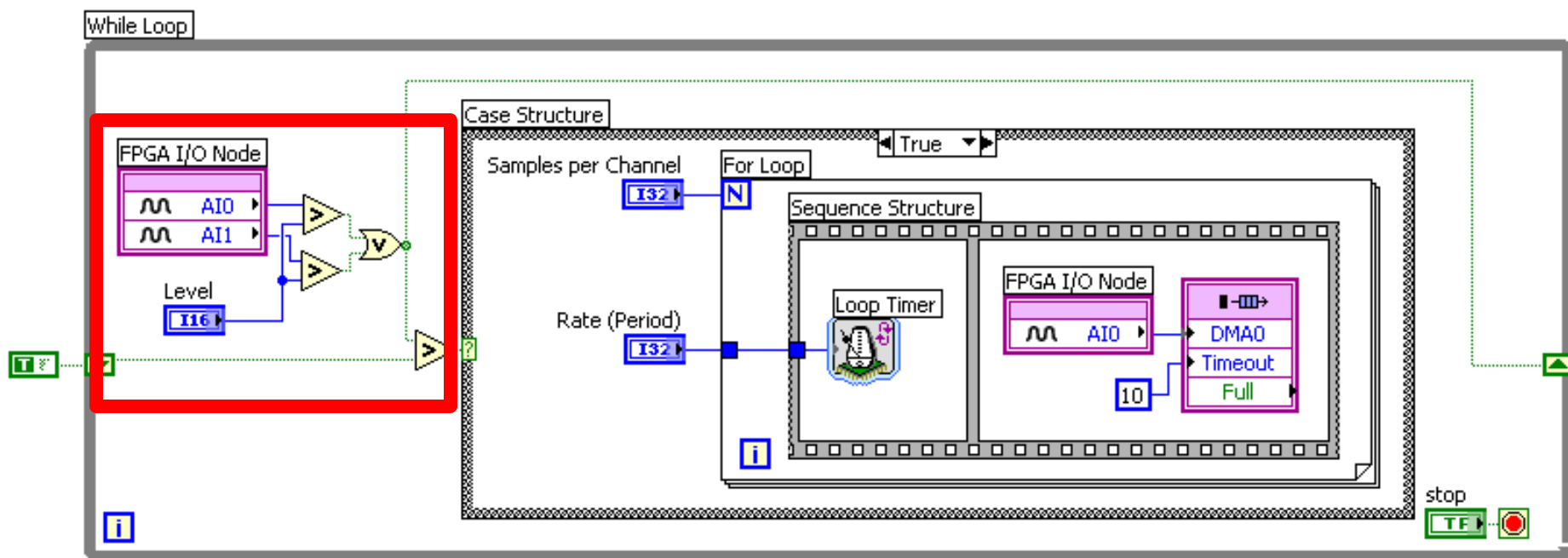
- High-speed control
- **Custom DAQ**
- Digital communication protocols
- Sensor simulation
- Onboard processing and data reduction

# Customize Your DAQ Device

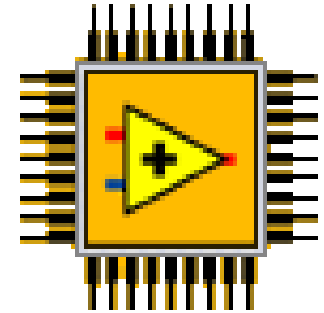
- Custom timing & synchronization
- Multi-rate sampling
- Custom triggering
- Custom counters
- Flexible PWM
- Flexible encoder interface



# Custom Triggered Analog Input



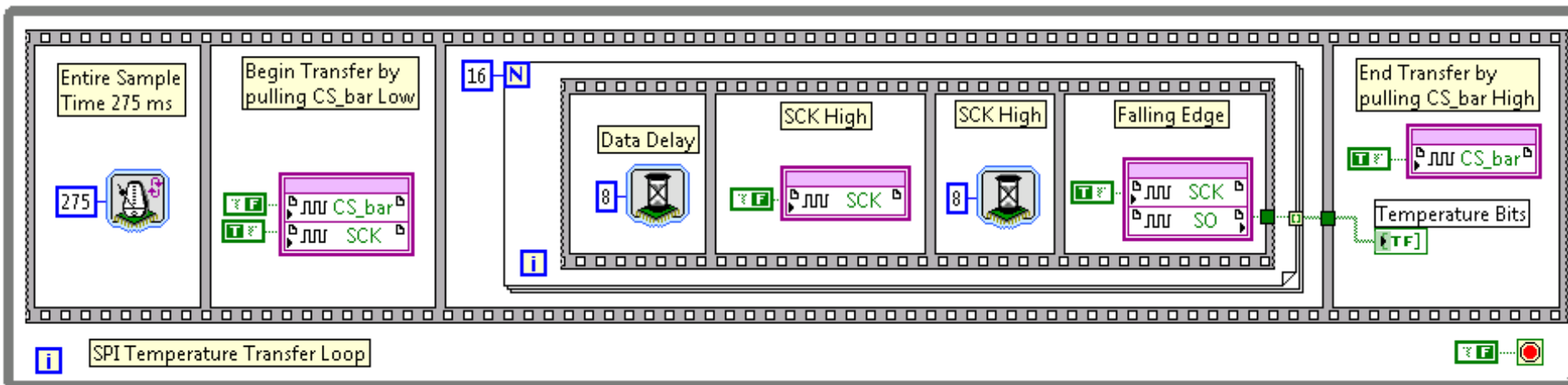
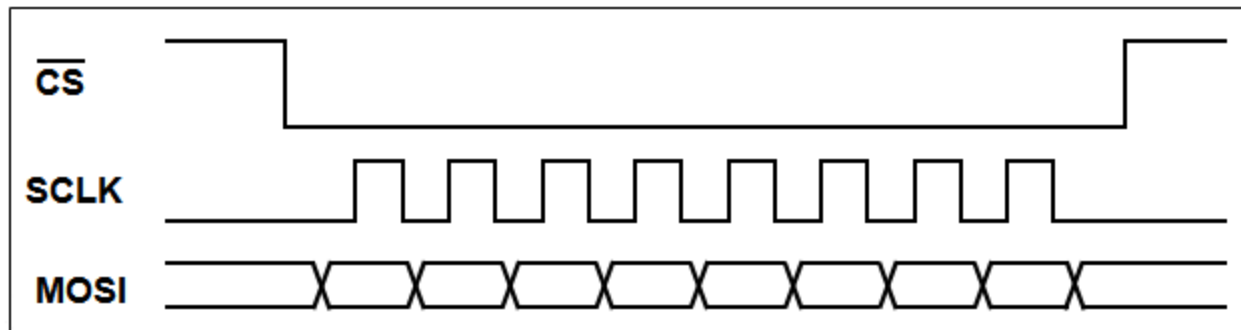
# Common Applications



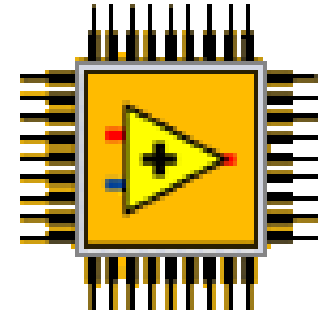
- High-speed control
- Custom DAQ
- **Digital communication protocols**
- Sensor simulation
- Onboard processing and data reduction

# Digital Communication

## Example – SPI



# Common Applications



- High-speed control
- Custom DAQ
- Digital communication protocols
- **Sensor simulation**
- Onboard processing and data reduction

# Sensor Simulation and FPGA

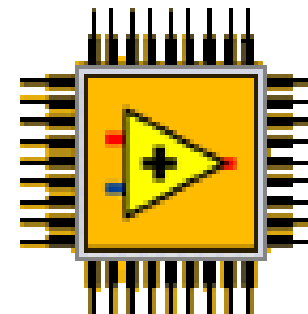
- Fully customizable hardware – Many types of sensors
- Parallelism – Many sensors on chip with no interference
- Strict timing requirements – Deterministic or highly realistic
- Onboard processing – Engineering units to sensor signals



Sensor Signals



# Common Applications



- High-speed control
- Custom DAQ
- Digital communication protocols
- Sensor simulation
- Onboard processing and data reduction

# Onboard Processing and Data Reduction

## Built-In I/O

- Analog voltages
- Digital communications
- Sensor signals

Input

## Processing

- Encoding/decoding
- Filtering/averaging
- Modulation/demodulation
- Decimation
- Stream processing

Process

## Output

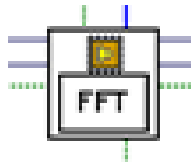
- DMA preprocessed data
- Streaming from input to output without host involvement

DMA to Host

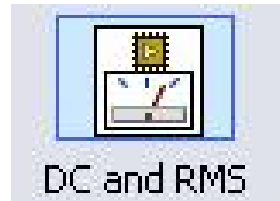
Output

# Intellectual Property (IP)

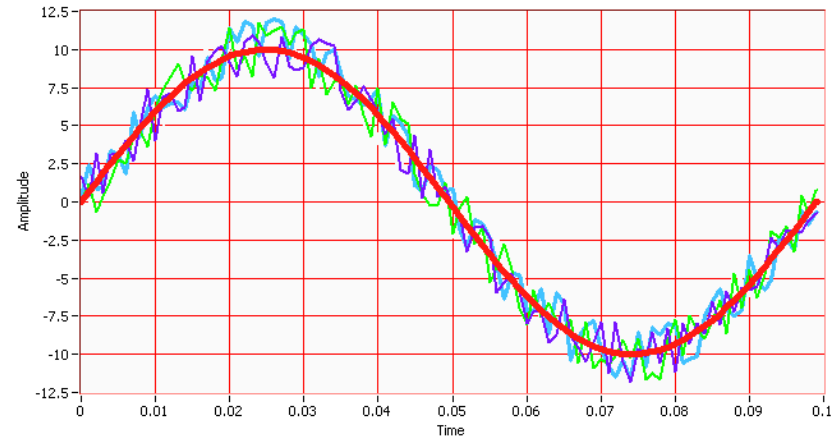
FFT



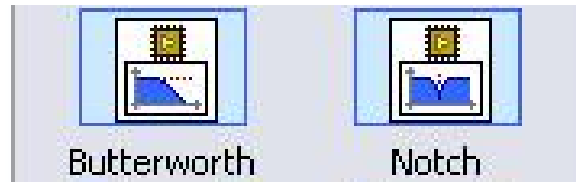
DC/RMS



Waveform Averaging

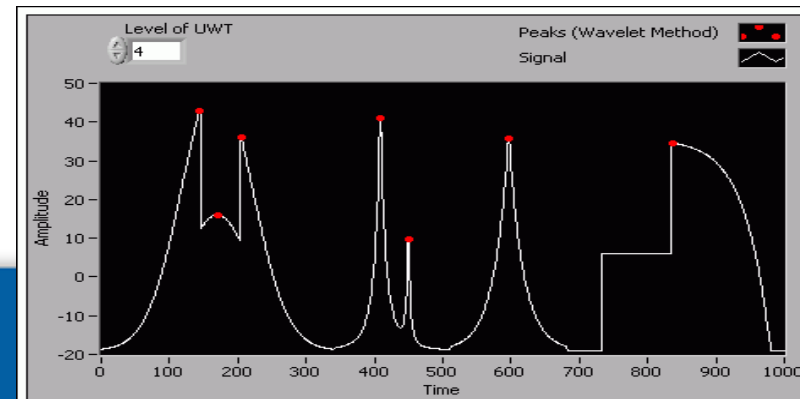


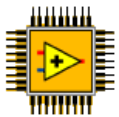
Digital filtering



Windowing

Resampling





# LabVIEW FPGA IPNet

[ni.com/ipnet](http://ni.com/ipnet)

The screenshot shows the IPNet website interface. It includes a 'Rate this document' section, a 'Share your IP' section, and a 'Table of Contents' with 11 items. Below the table of contents is a table of IP cores.

Name	LabVIEW Version	IP or Example	Source	Code Maturity
MultiIO	8.5	IP	NI Labs	5
ip1000	8.2	IP	NI Labs	2
ip1000	8.5	Example	NI Labs	5

Math

Signal Processing

Data Manipulation and Transfer

RF and Communications

Digital Protocols

Data Acquisition

Signal Generation

Control

Sensor Simulation

More than **200** IP cores and examples

# HDL-Based IP in LabVIEW FPGA

- HDL Interface Node
  - Inline HDL integration
- Component-Level IP Node
  - Parallel HDL integration

```
LIBRARY ieee;
USE ieee.std_logic_1164.ALL;

Library XilinxCoreLib;
ENTITY filt IS
    port (
        ND: IN std_logic;
        RDY: OUT std_logic;
        CLK: IN std_logic;
        RST: IN std_logic;
        RFD: OUT std_logic;
        DIN: IN std_logic_VECTOR(15 downto 0);
        DOUT: OUT std_logic_VECTOR(30 downto 0));
END filt;

ARCHITECTURE filt_a OF filt IS
    component wrapped_filt
        port (
            ND: IN std_logic;
            RDY: OUT std_logic;
            CLK: IN std_logic;
            RST: IN std_logic;
            RFD: OUT std_logic;
            DIN: IN std_logic_VECTOR(15 downto 0);
            DOUT: OUT std_logic_VECTOR(30 downto 0);
        );
    end component;
    -- Configuration specific
    for all : wrapped_filt
        generic map(
            -- Configuration specific
            use ieee.std_logic_1164.all;
            use ieee.numeric_std.all;
        )
        entity DemoClipAdder is
            port (
                clk : in std_logic;
                aReset : in std_logic;
                cPortA : in std_logic_vector(15 downto 0);
                cPortB : in std_logic_vector(15 downto 0);
                cAddout : out std_logic_vector(15 downto 0) := (others => '0')
            );
        end DemoClipAdder;

        architecture rtl of DemoClipAdder is
            begin
                process(aReset, clk) begin
                    if(aReset = '1') then
                        cAddout <= (others => '0');
                    elsif rising_edge(clk) then
                        cAddout <= std_logic_vector(signed(cPortA) + signed(cPortB));
                    end if;
                end process;
            end rtl;
        end map;
    END filt_a;
```

# How to Learn More

[ni.com/fpga](http://ni.com/fpga)

# Questions?

[ni.com/training](http://ni.com/training)

## 2 Day LabVIEW FPGA Module Course

The screenshot shows the NI website's LabVIEW FPGA Module page. At the top, there's a navigation bar with 'INSTRUMENTS' and various links like 'MyNI', 'Contact NI', 'Products & Services', 'Solutions', 'Support', 'NI Developer Zone', 'Academic', 'Events', and 'Compare'. Below this, a breadcrumb trail reads 'Home > Products & Services > Measurement & Automation Software > NI LabVIEW > LabVIEW FPGA Module'. The main content area features a large graphic with a LabVIEW block diagram and a Xilinx Virtex-5 XC5VLX50 chip. Text on the page includes 'LabVIEW on a Chip Target FPGAs with Graphical Programming' and 'Watch the overview video'. A sidebar on the left lists navigation options: 'What is LabVIEW FPGA?', 'LabVIEW FPGA Hardware Targets', 'What is FPGA Technology?', 'Video Demonstration', 'New LabVIEW FPGA Features', 'FPGA Cores and FPGA Samples', and 'FPGA Case Studies'. A main text block describes the LabVIEW FPGA Module as a way to extend graphical development to target field-programmable gate arrays (FPGAs) on NI reconfigurable I/O (RIO) hardware. Below this, there are three promotional boxes: 'NI Developer Suite' (offering a 67% discount on bundled software), 'Try NI LabVIEW on Evaluation Hardware' (offering a 90-day evaluation kit), and '2009 Virtual Automated Test Summit' (offering free training on test practices). At the bottom, there are 'Product Tutorials' and a 'Buy Now' section for the LabVIEW FPGA Module.

