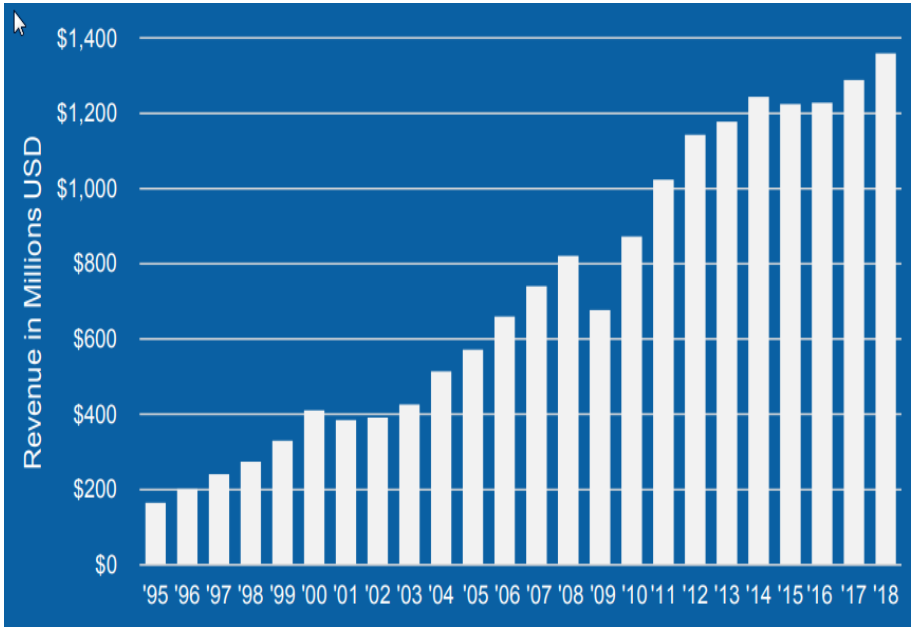
Communication Systems Design Using Dataflow Programming

Jacob Kornerup, Ph.D.
LabVIEW R&D
National Instruments

NATIONAL INSTRUMENTS

A Word About National Instruments

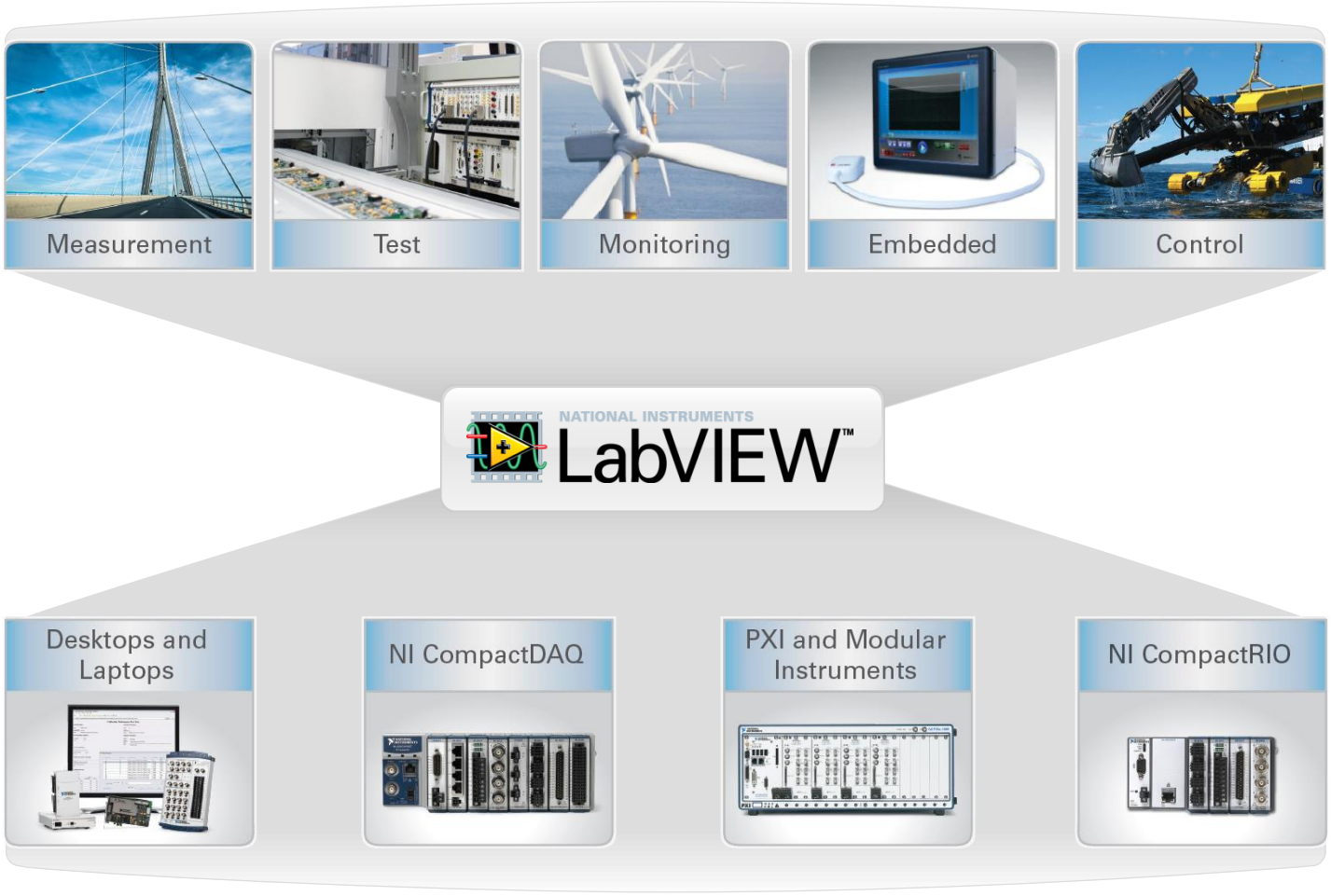
- **Annual Revenue:** $1.36 billion
- **Global Operations:** Approximately 6,870 employees; operations in more than 40 countries
- **Broad Customer Base**: More than 35,000 companies served annually



NATIONAL INSTRUMENTS

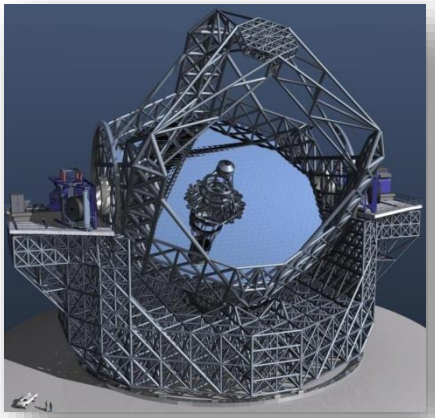
Graphical System Design

A Platform-Based Approach for Measurement and Control

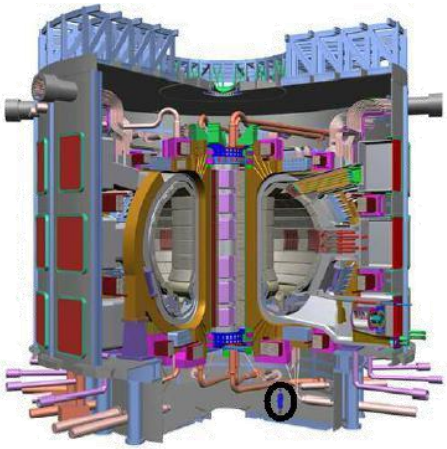


NATIONAL INSTRUMENTS

Tough Real-Time Challenges

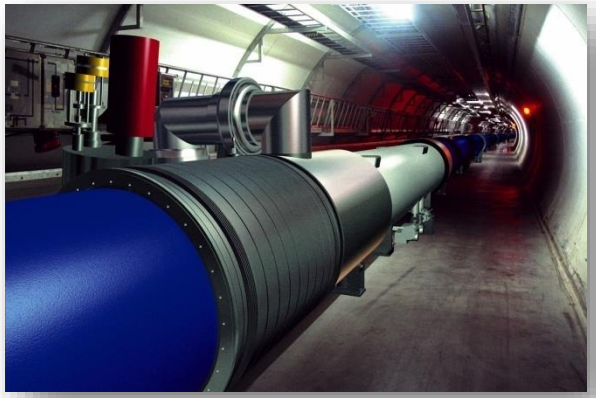


Large Telescope
Mirror Control

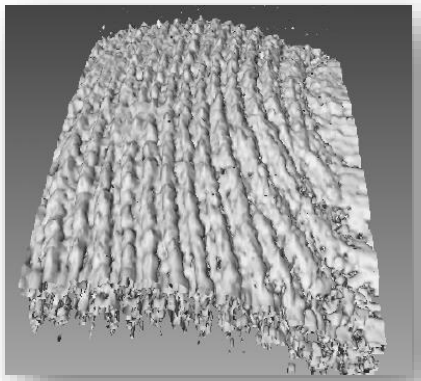


Tokomak
Plasma Control

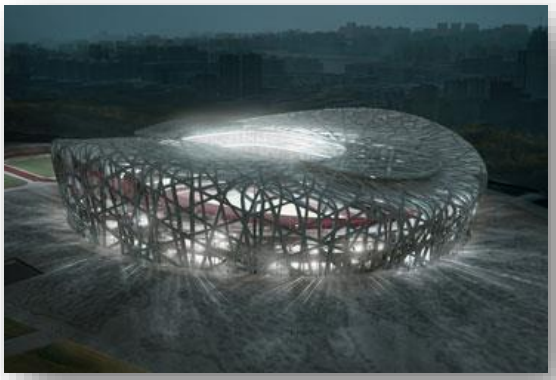
Wind Turbine Sound Source
Characterization



CERN Hadron Collider



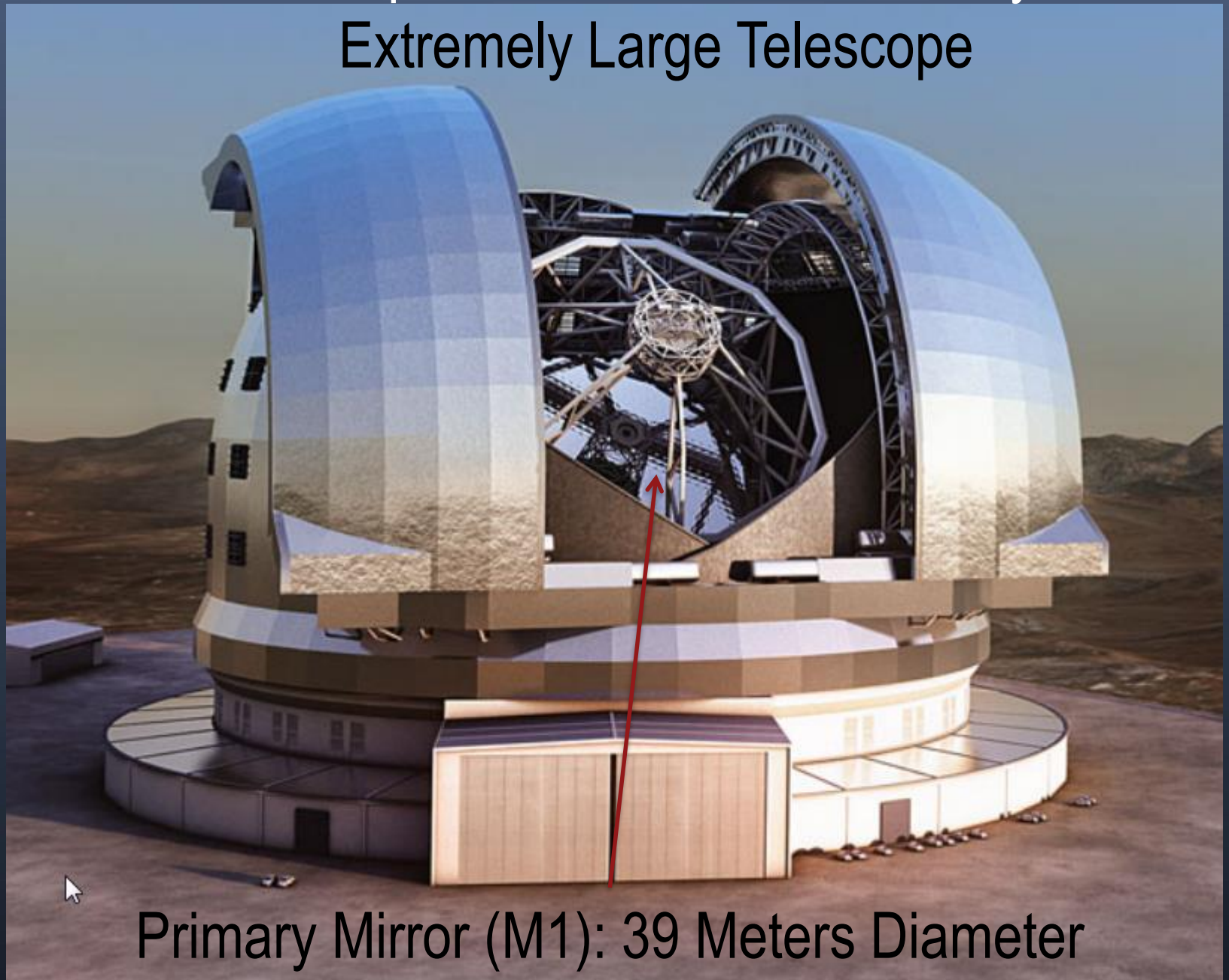
Early Cancer Detection



Structural Health Monitoring

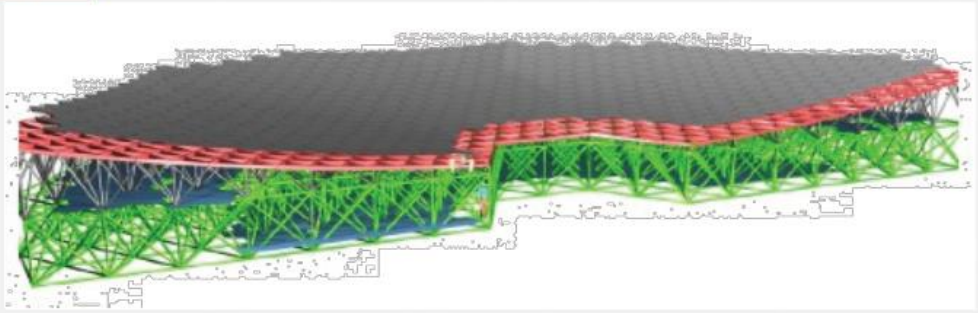
NATIONAL INSTRUMENTS

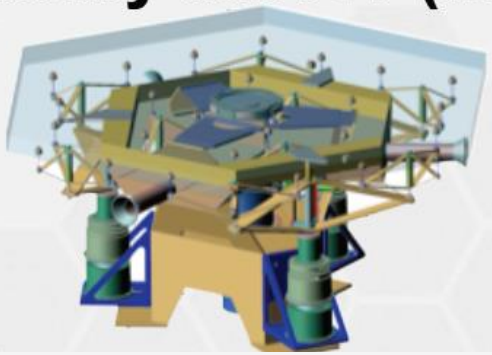
European Southern Observatory
Extremely Large Telescope



Primary Mirror (M1): 39 Meters Diameter

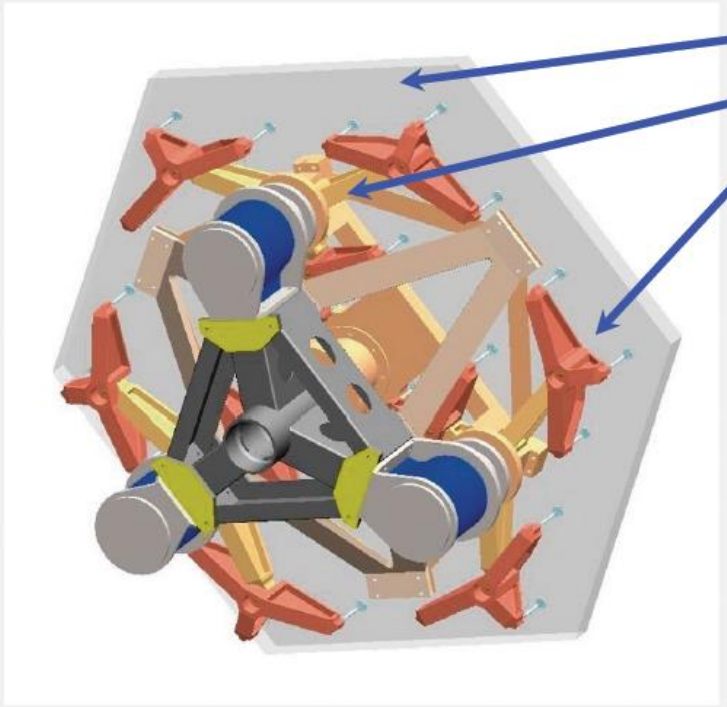
The Primary Mirror (M1)

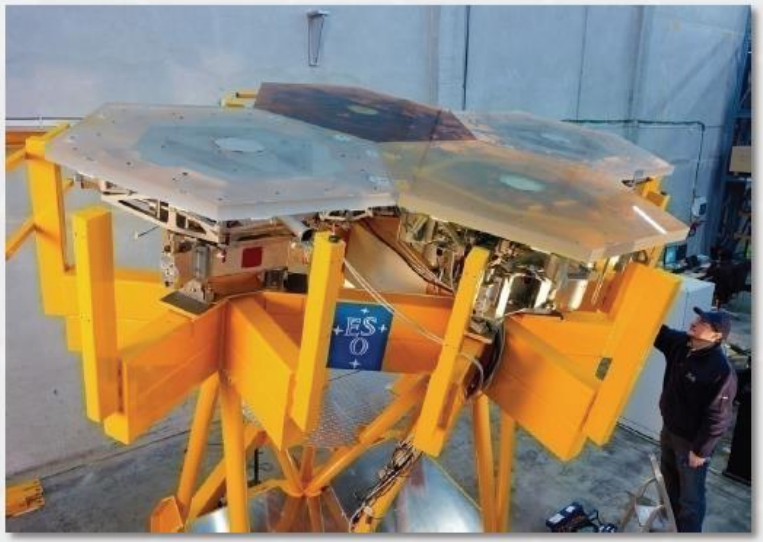


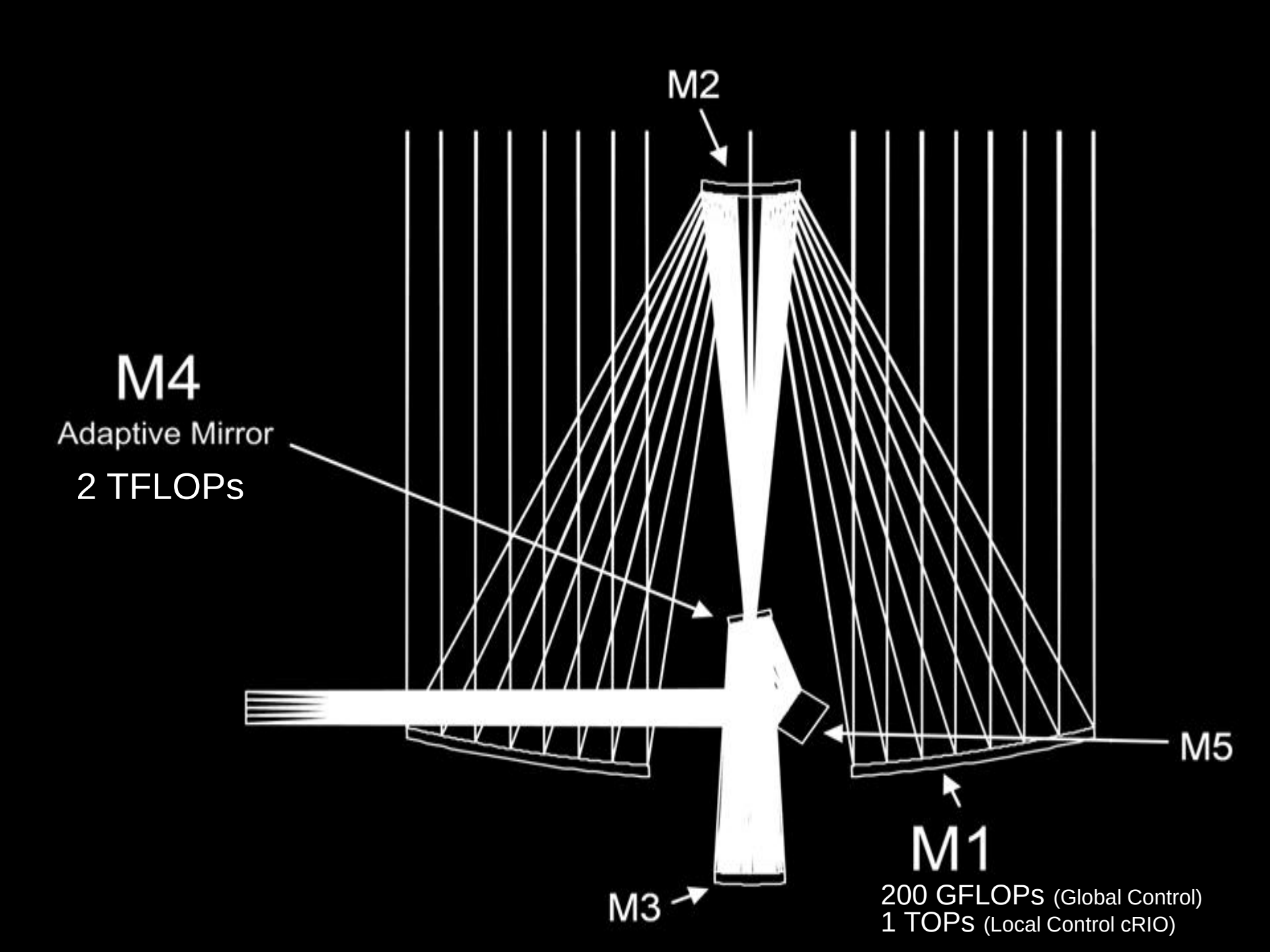


Mirror Segment Active Control (798 Mirrors)

- 6 Edge Sensors (4788 total)
- 3 Actuators (2394 total)
- 2.4K x 4.8K Matrix calculation / 1 ms






M2
M4
Adaptive Mirror
2 TFLOPs
M5
M1
200 GFLOPs (Global Control)
1 TOPs (Local Control cRIO)
M3

Perspective

Pope Election 2005

NATIONAL INSTRUMENTS

Perspective

Pope Election 2013

NATIONAL INSTRUMENTS

Prototyping Is Critical for Algorithm Research

"Experience shows that the real world often breaks some of the assumptions made in theoretical research, so **testbeds are an important tool for evaluation under very realistic operating conditions**"

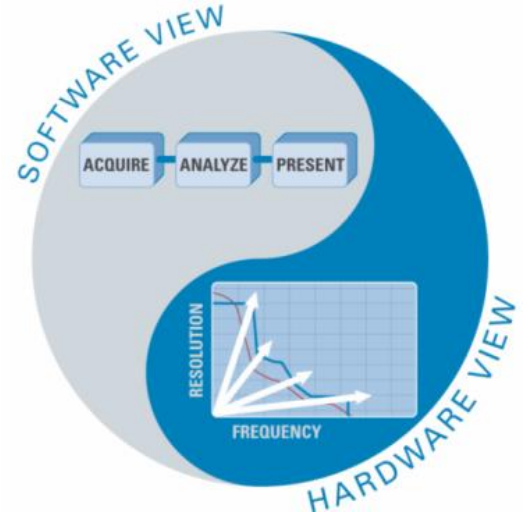
"…development of **a testbed** that is able **to test radical ideas** in a complete, working system **is crucial**"

[1]NSF Workshop on Future Wireless Communication Research

NATIONAL INSTRUMENTS

The National Instruments Vision

"To do for test and measurement
what the spreadsheet did for financial analysis."



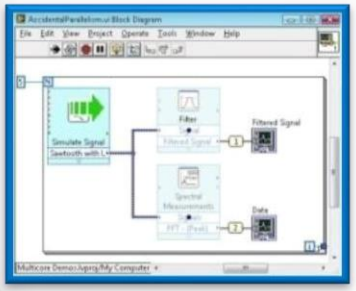
Virtual Instrumentation

with NI LabVIEW™

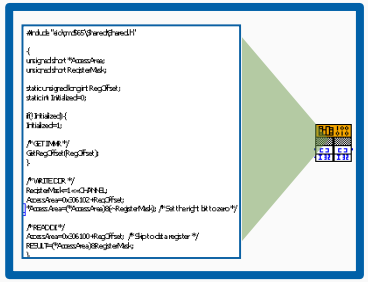
NATIONAL INSTRUMENTS™

High-Level Design Models

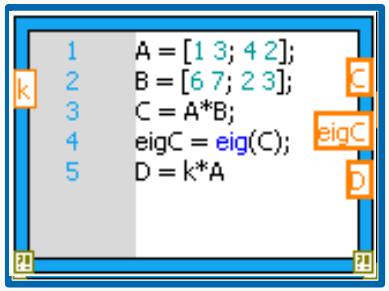
Data Flow



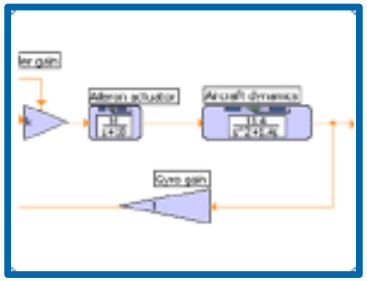
C Code



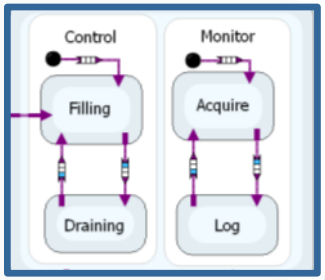
Textual Math



Simulation



Statechart



NATIONAL INSTRUMENTS LabVIEW™

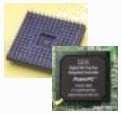
Graphical System Design Platform

PC/Mac/Linux

PXI

CompactRIO

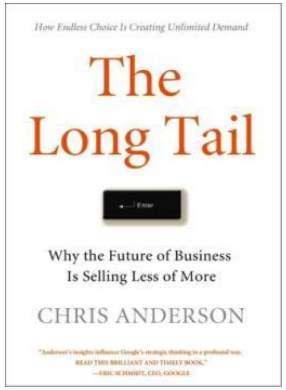
FlexRIO

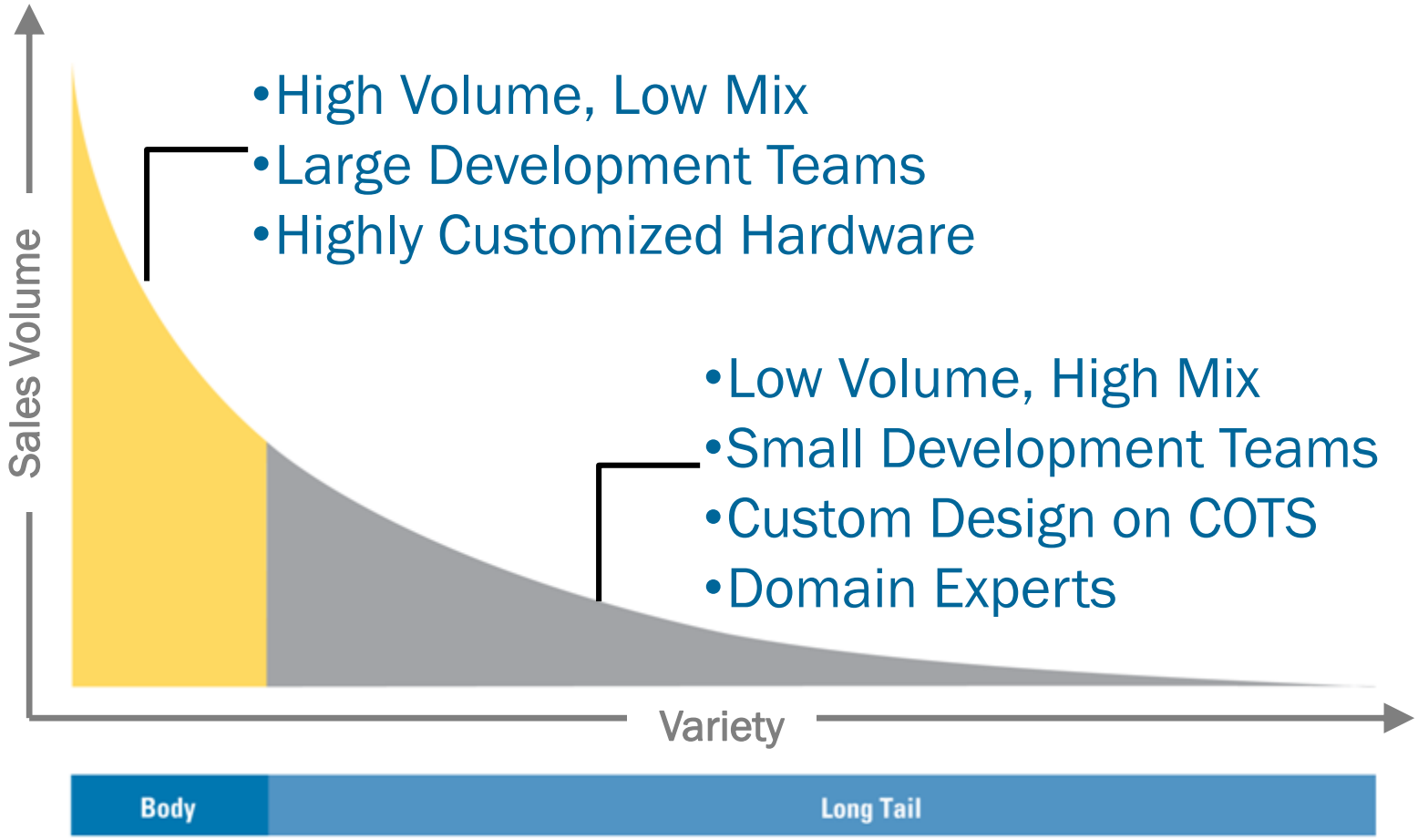


Custom

NATIONAL INSTRUMENTS™

The Long Tail

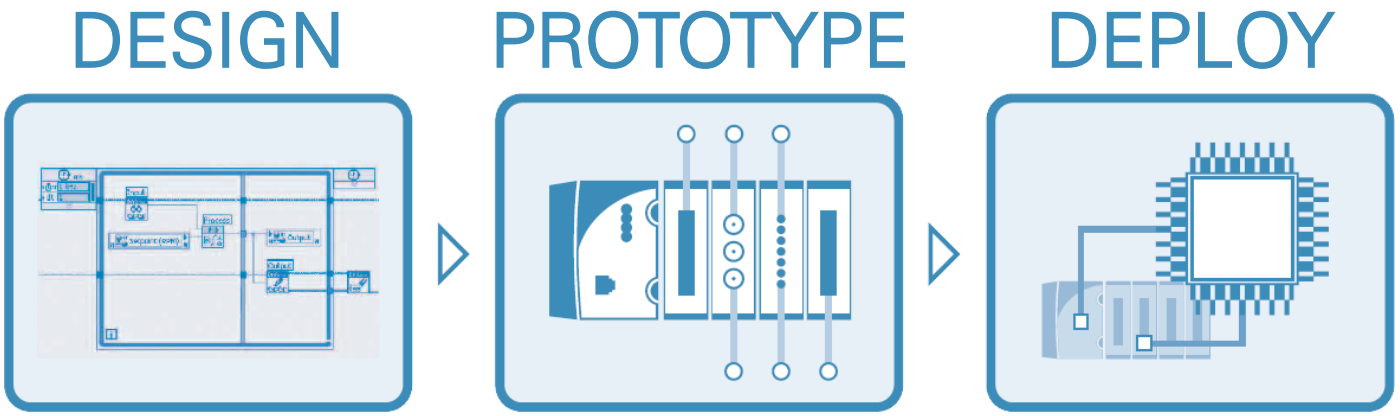


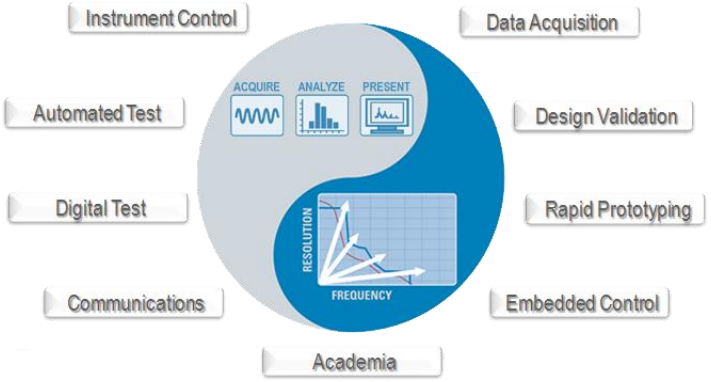


["The Long Tail," Chris Anderson *Wired*, 2004]

NATIONAL INSTRUMENTS

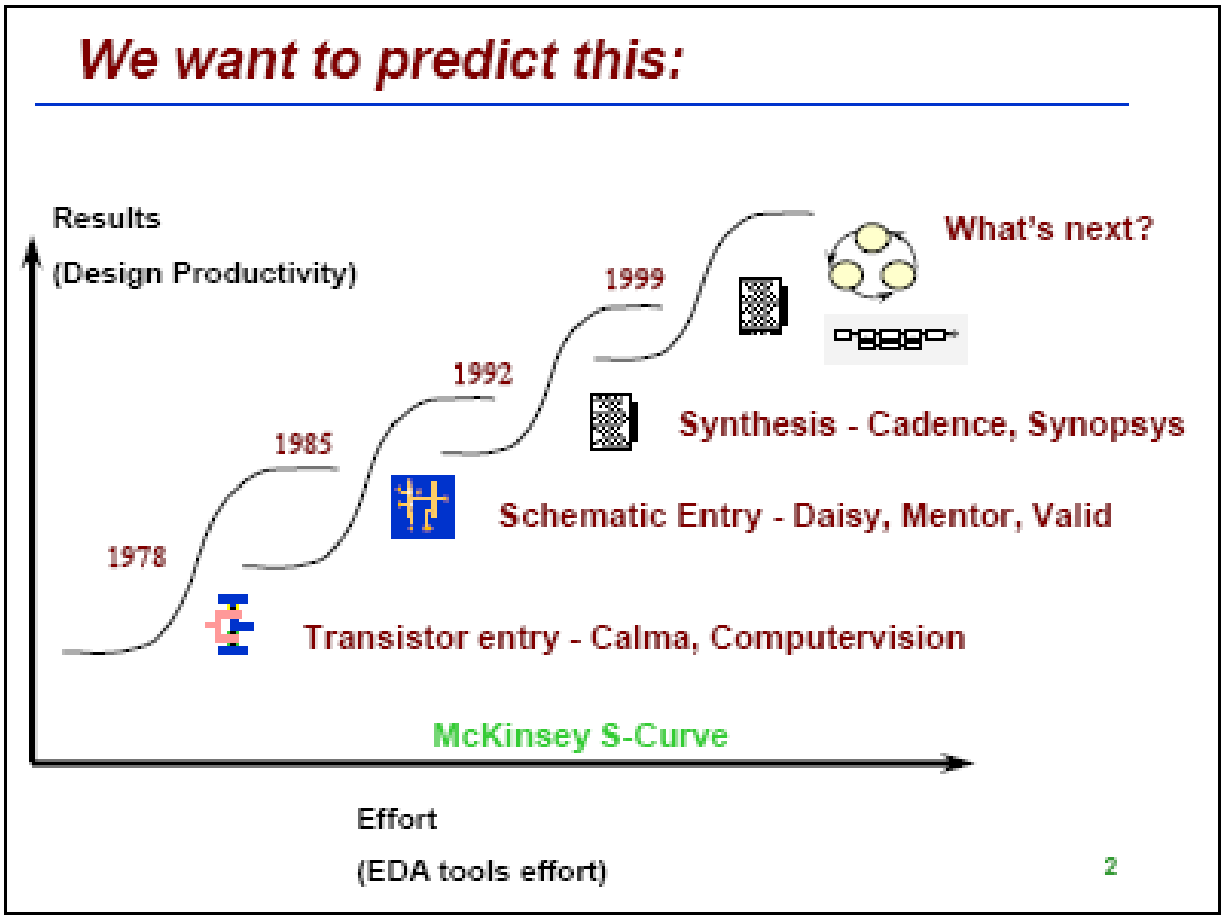
NI Vision *Evolved*: Graphical System Design





NATIONAL INSTRUMENTS

Design discontinuities in EDA tools



[1] Kurt Keutzer, UC Berkeley EECS 244 class

Platform Based Design & Models of Computation

- Constructs for application domain experts
- Structured implementation with the right levels of abstraction
- Separation of concerns between functionality and architecture
- Evolve designs on hardware "generations"
- Design flow that supports analysis, simulation, verification and synthesis

[1] E.A. Lee, "Embedded Software", Revised from UCB ERL Memorandum M01/26, November 1, 2001,
[2] E.A. Lee and S. Neuendorffer, "Concurrent Models of Computation for Embedded Software", Memorandum No. UCB/ERL M04/26, July 22, 2004
[3] Alberto Sangiovanni-Vincentelli, "Quo Vadis, SLD? Reasoning About the Trends and Challenges of System Level Design", Proceedings of the IEEE, Vol. 95, No. 3, March 2007.

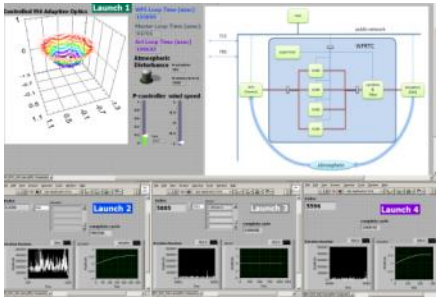
The Y-Chart System Design Methodology

Application Logic

- Algorithm development
- IO characterization
  - Timing characteristics
- Real-time constraints
- Models of Computation integration
- State management

Analysis & Mapping

Performance Evaluation

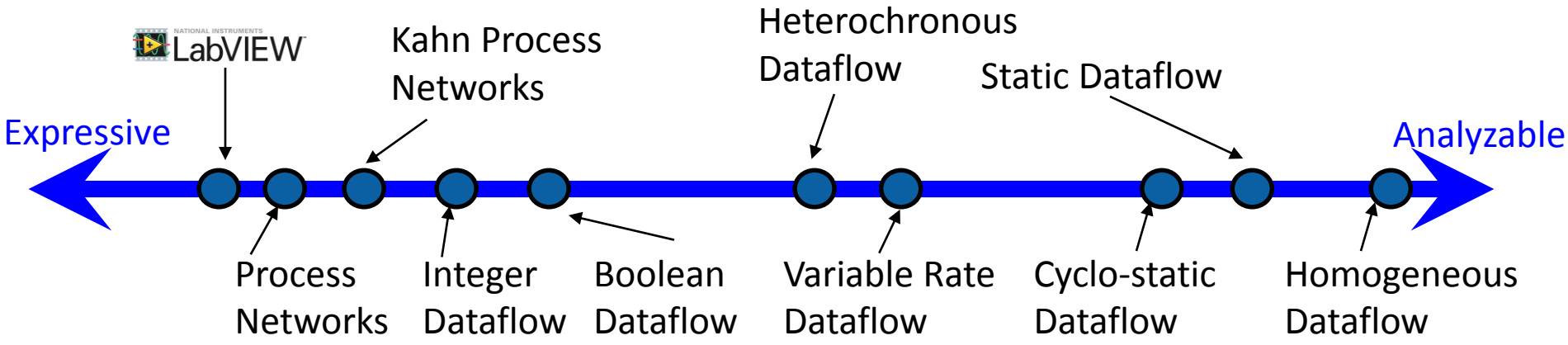


Platform Architecture

- Distributed
- Heterogeneous computing platforms
  - Real-time OS, FPGA, Desktop OS, GPU
- Communication schemes
- Real-time
- IO
- Timing

1. Kienhuis, Deprettere, van der Wolf, and Vissers., "A Methodology to Design Programmable Embedded Systems - The Y-Chart Approach. Embedded Processor Design Challenges: Systems, Architectures, Modeling, and Simulation" - SAMOS, p.18-37, Jan. 2002.
2. Keutzer, Newton, Rabaey, Sangiovanni-Vincentelli, "System-level Design: Orthogonalization of Concerns and Platform-based Design," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 19(12): p. 1523-1543, Dec. 2000.

Dataflow MoCs for Streaming Applications

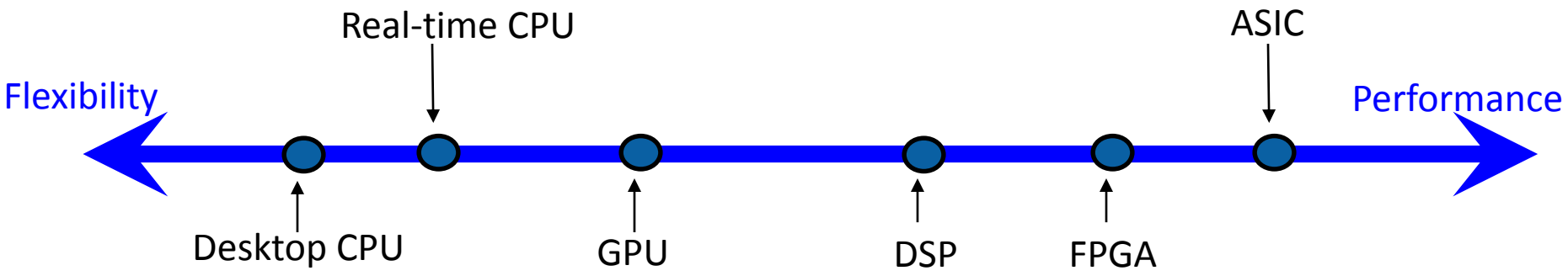


Deter-ministic?	No	Yes
Synchronous?	No	Yes
Deadlock and boundedness decidable?	No	Yes
Static scheduling?	No	Yes

Key trade-off: Analyzability vs. Expressibility

NATIONAL INSTRUMENTS

Platform Architectures

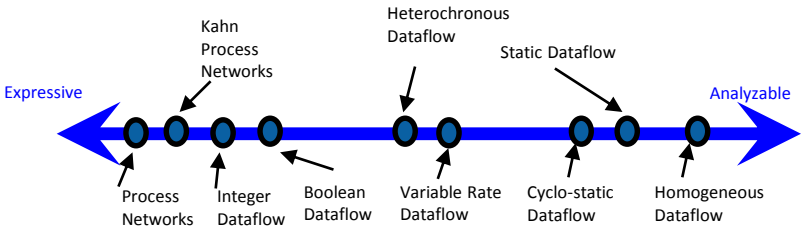


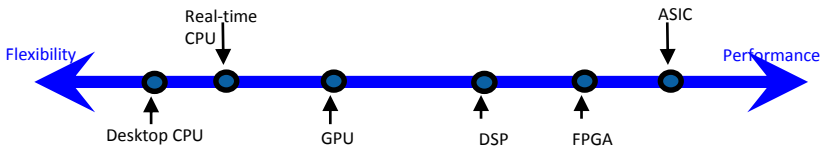
Deter-ministic?	No	Yes
General purpose ?	Yes	No
Floating point math?	Yes	No
Ease of Reprogramability?	Yes	No

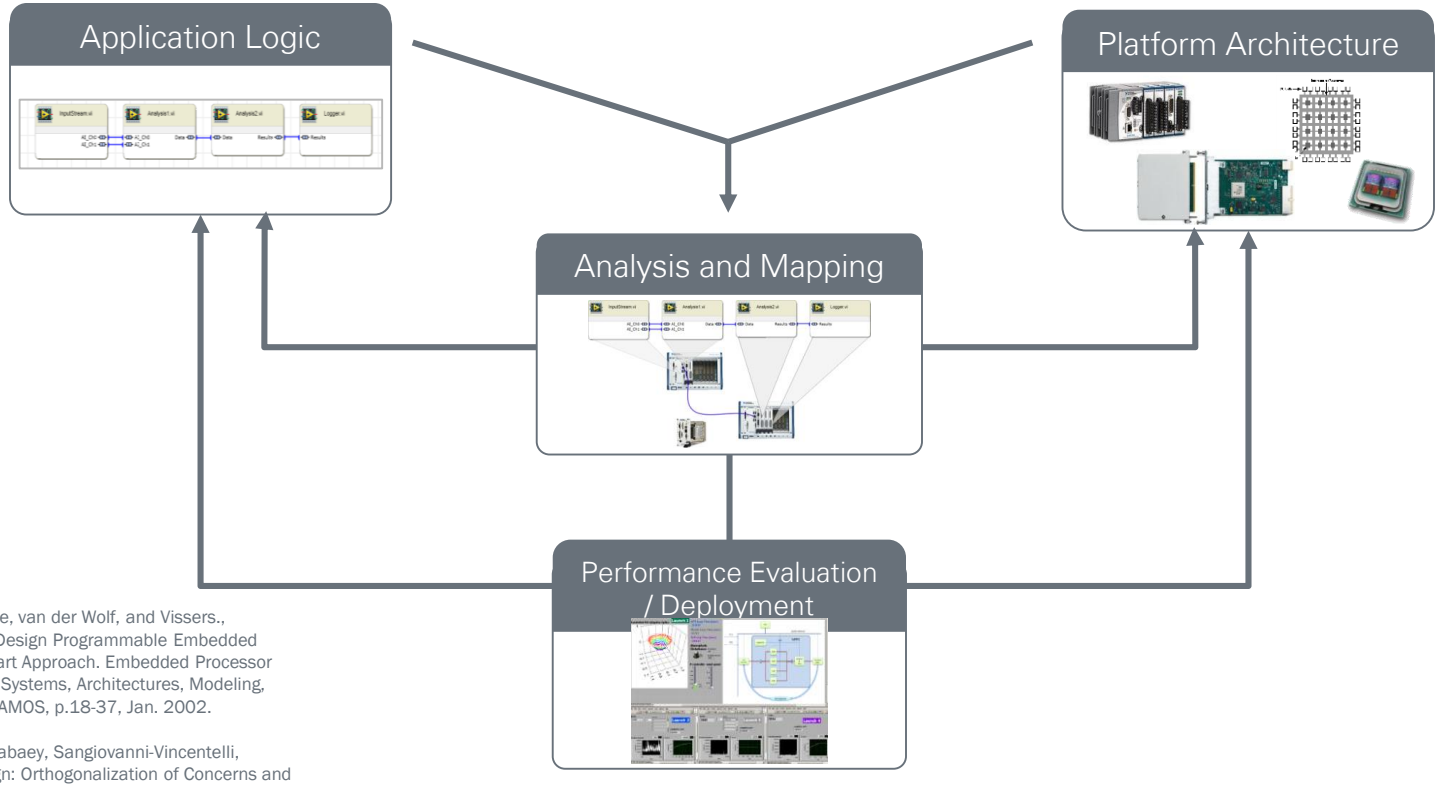
Key trade-off: Flexibility vs. Performance

NATIONAL INSTRUMENTS

The Y-Chart System Design Methodology





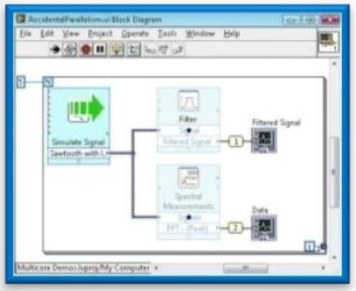


1. Kienhuis, Deprettere, van der Wolf, and Vissers., "A Methodology to Design Programmable Embedded Systems - The Y-Chart Approach. Embedded Processor Design Challenges: Systems, Architectures, Modeling, and Simulation" - SAMOS, p.18-37, Jan. 2002.

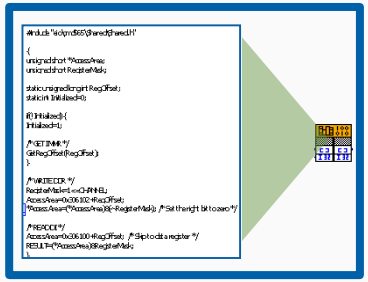
2. Keutzer, Newton, Rabaey, Sangiovanni-Vincentelli, "System-level Design: Orthogonalization of Concerns and Platform-based Design," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 19(12): p. 1523-1543, Dec. 2000.

High-Level Design Models

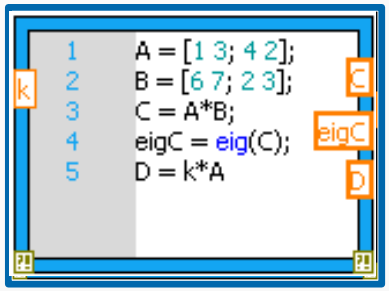
Data Flow



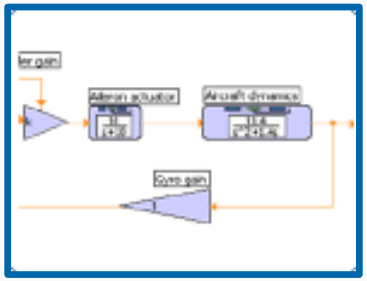
C Code



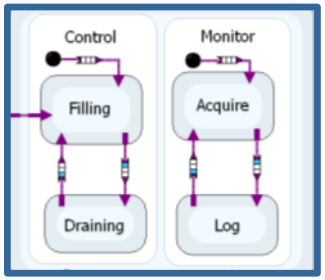
Textual Math



Simulation



Statechart



NATIONAL INSTRUMENTS LabVIEW™

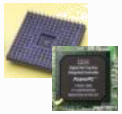
Graphical System Design Platform

PC/Mac/Linux

PXI

CompactRIO

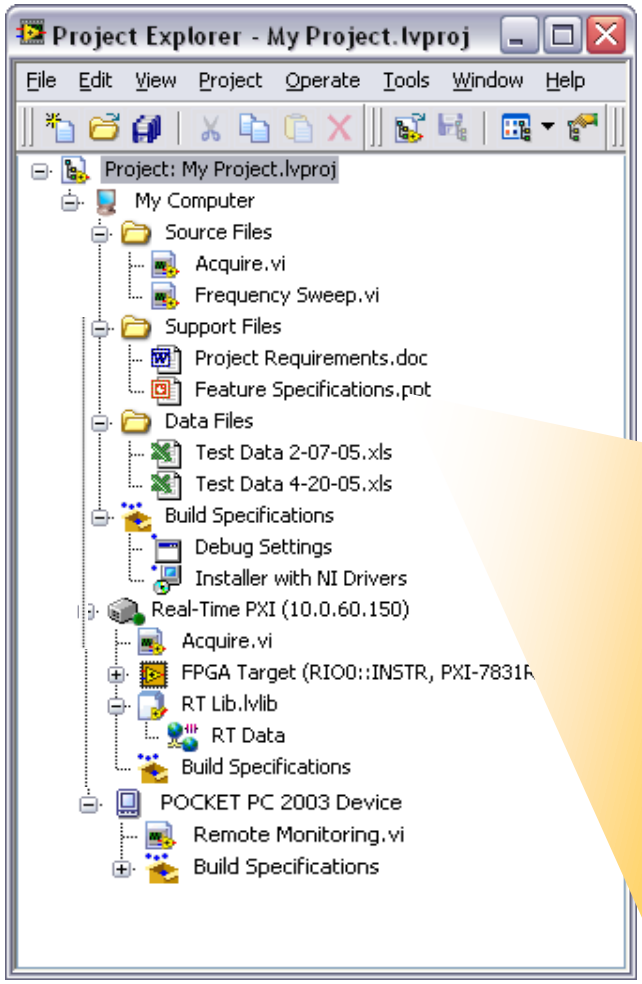
FlexRIO



Custom

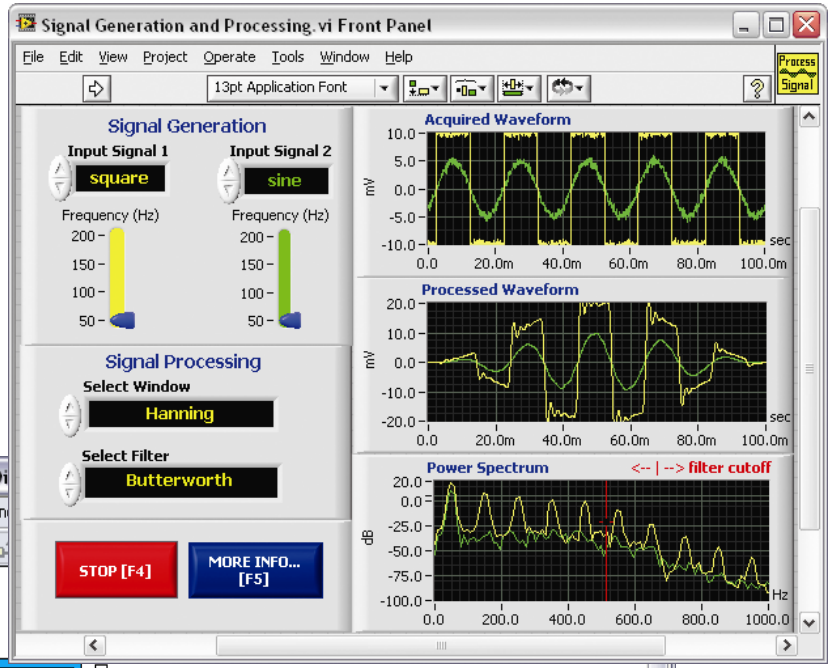
NATIONAL INSTRUMENTS™

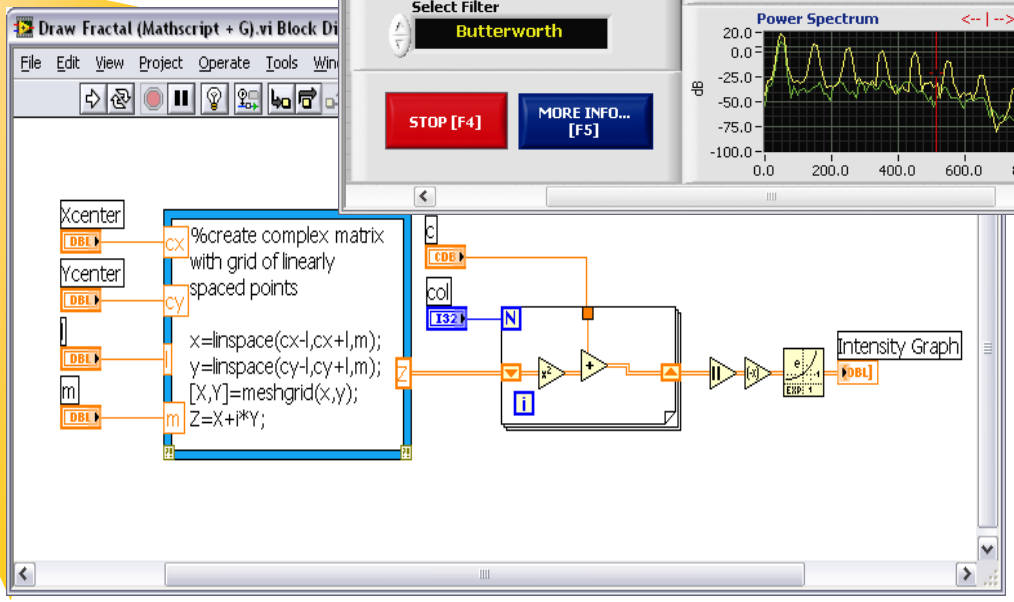
LabVIEW Today



Distributed Computing

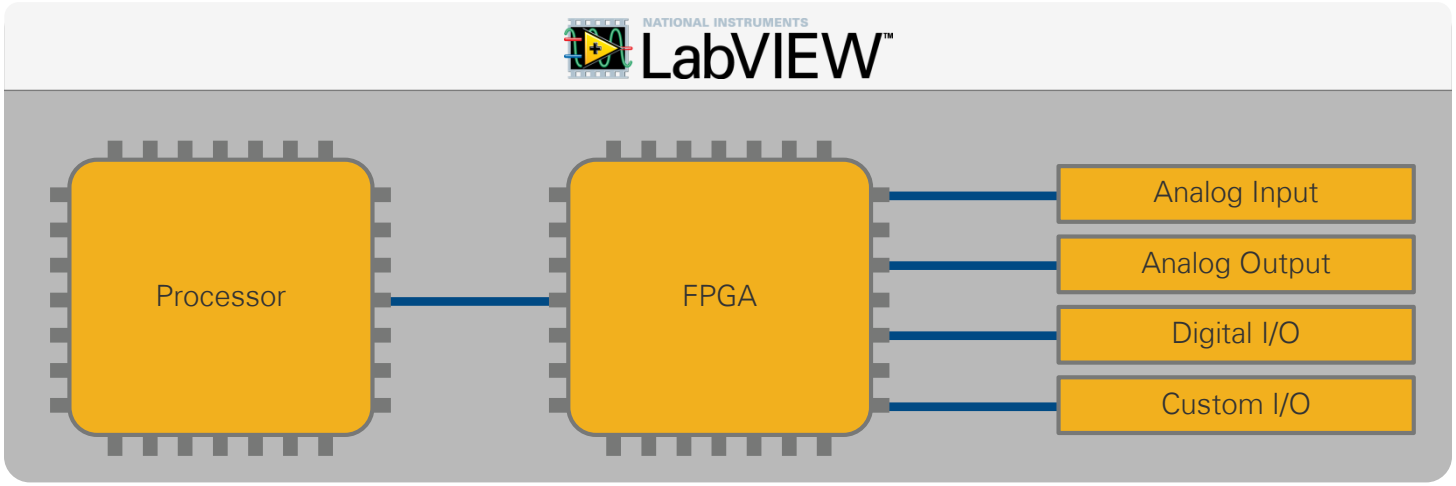
Multiple Programming Models





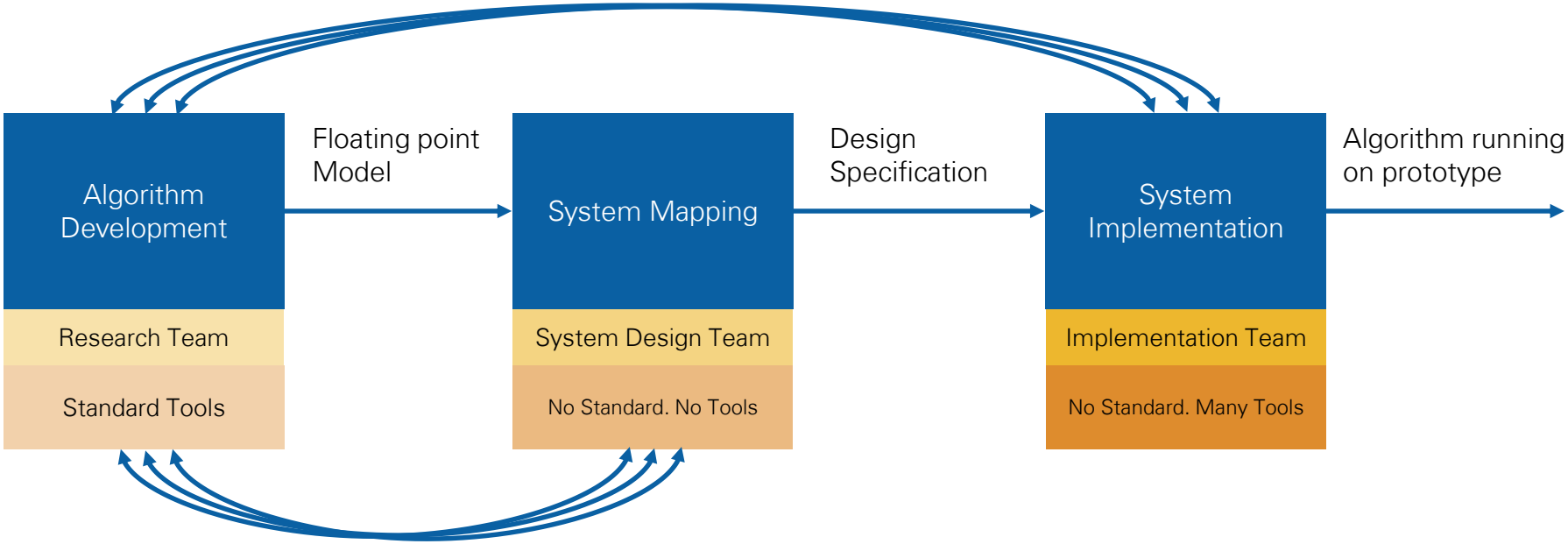
NATIONAL INSTRUMENTS

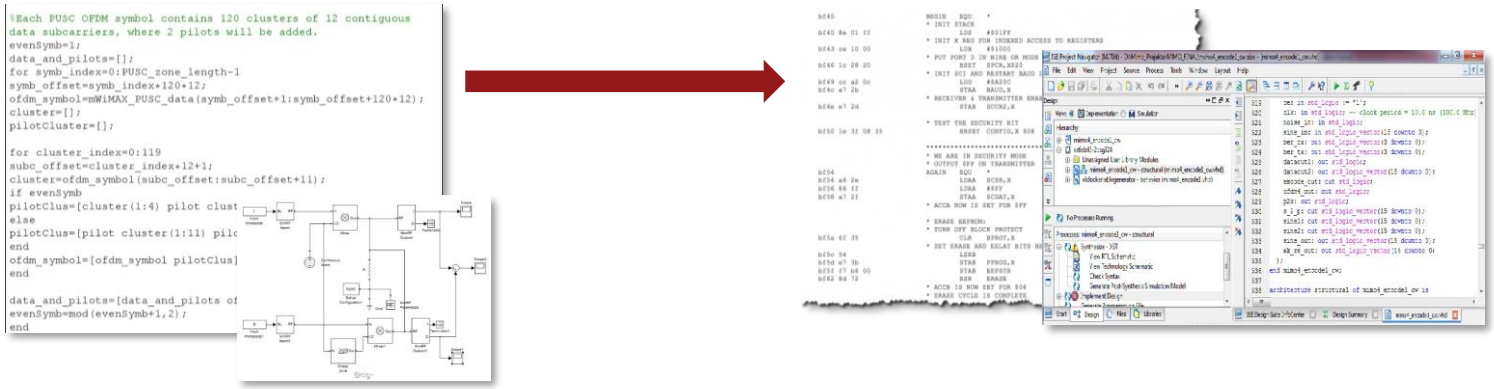
Realizing Our Vision for Instrumentation
Graphical System Design



NATIONAL INSTRUMENTS™

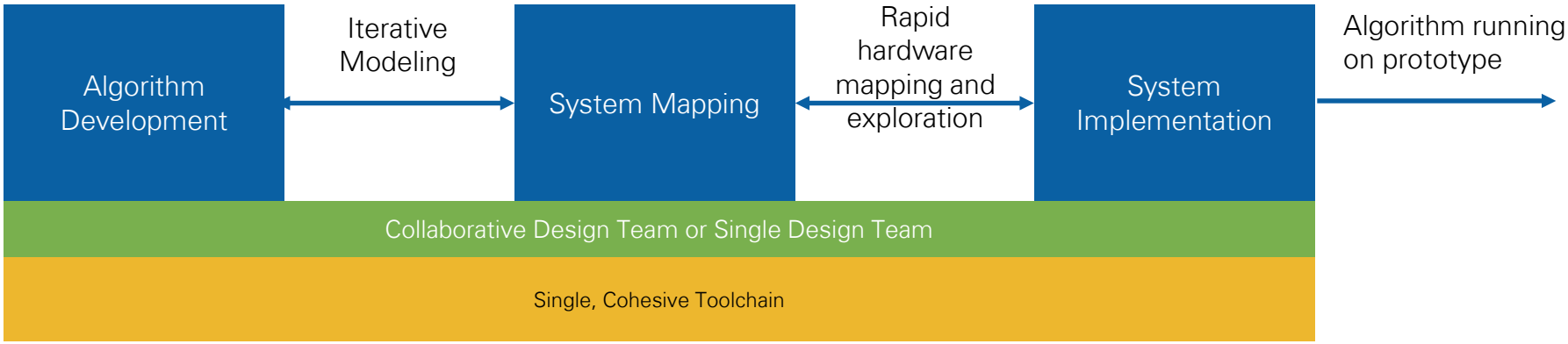
Classical Design Flow

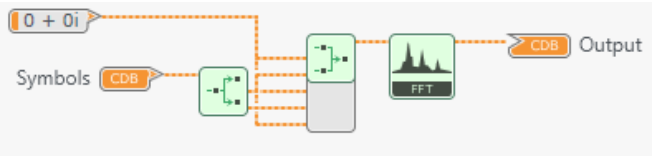


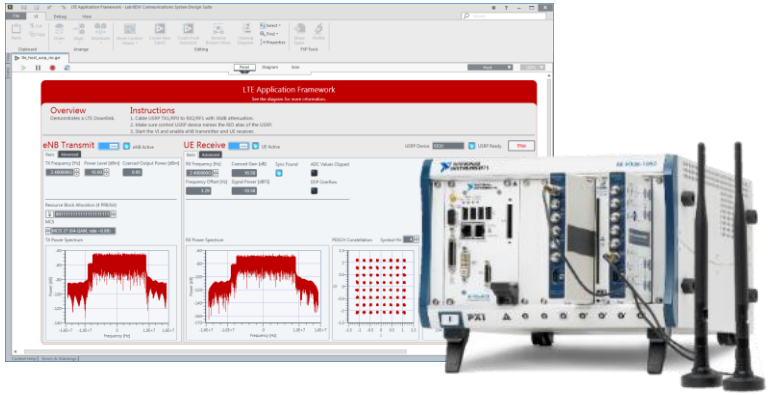


NATIONAL INSTRUMENTS

Platform Enabled Design Flow



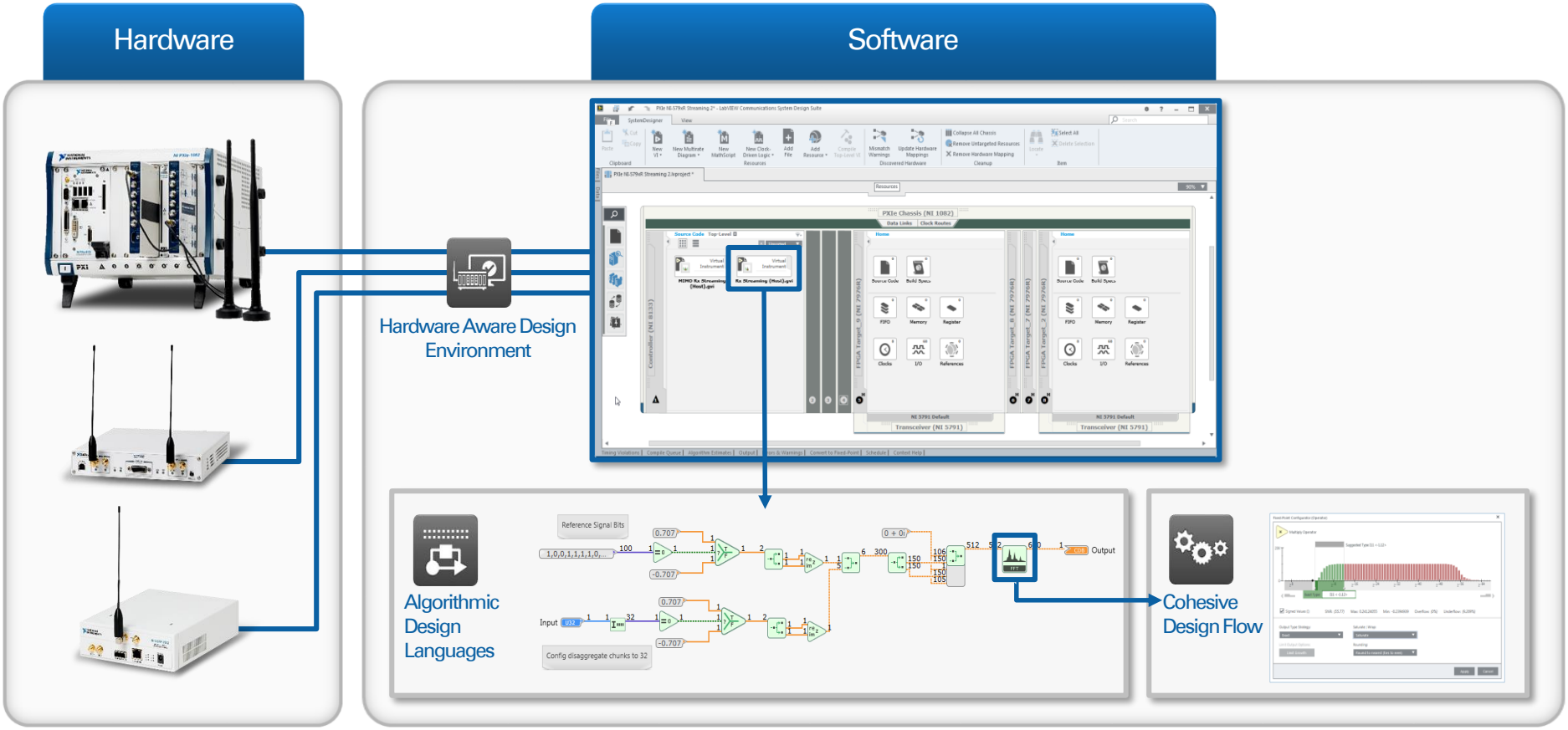




NATIONAL INSTRUMENTS

LabVIEW Communications System Design Suite

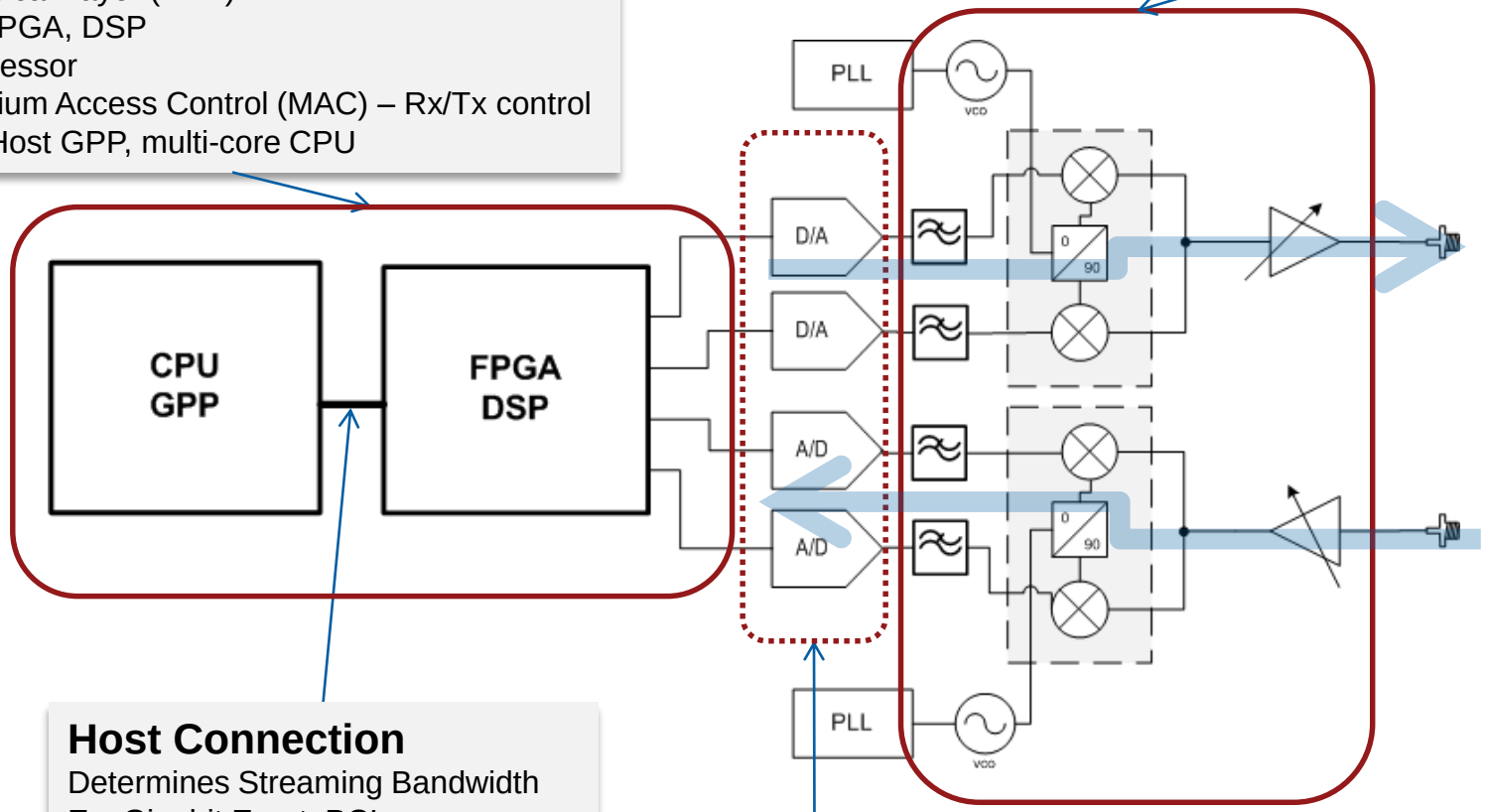
A Platform for Software Defined Radio Development



Software Defined Radio Architecture

RF Front End
General Purpose RF
Dual LOs
Contiguous Frequency Range

Multi-Processor Subsystem
Real-time signal processor
Physical Layer (PHY)
ex FPGA, DSP
Host processor
Medium Access Control (MAC) – Rx/Tx control
ex. Host GPP, multi-core CPU

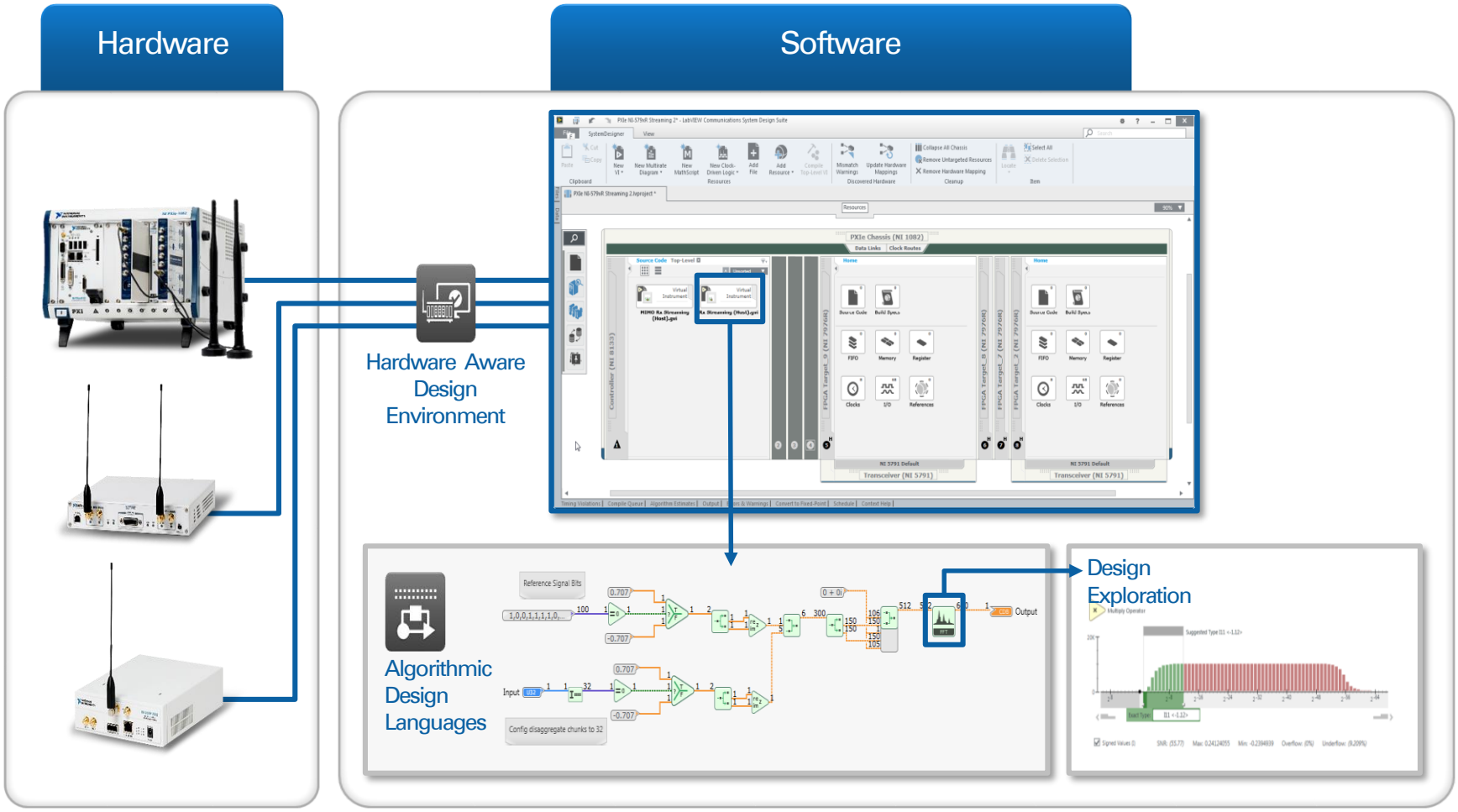


Host Connection
Determines Streaming Bandwidth
Ex. Gigabit E-net, PCIe

Baseband Converters

NATIONAL INSTRUMENTS

Design Tool Wish List



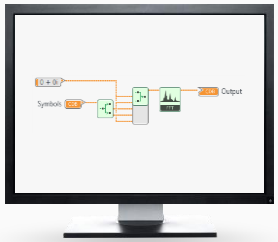
NATIONAL INSTRUMENTS

Unified Design Flow

Single, Cohesive Toolchain

Algorithm Development

- C and .M Support
- Algorithm design languages



System Mapping

- Hardware aware environment
- Easy, iterative design partitioning

Design Exploration

- Design simulation tools built in
- Analyze tradeoffs in implementations

System Implementation

- Deploy to hardware
- Prototype with real-world signals

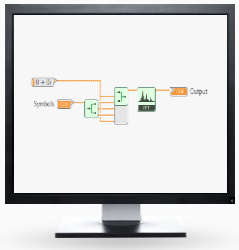
Collaborative Design Team

Unified Design Flow

Single, Cohesive Toolchain

Algorithm Development

- C and .M Support
- LabVIEW (G)
- Multirate (SDF)



System Mapping

- Hardware aware environment
- Easy, iterative design partitioning

Design Exploration

- Design simulation tools built in
- Analyze tradeoffs in implementations

System Implementation

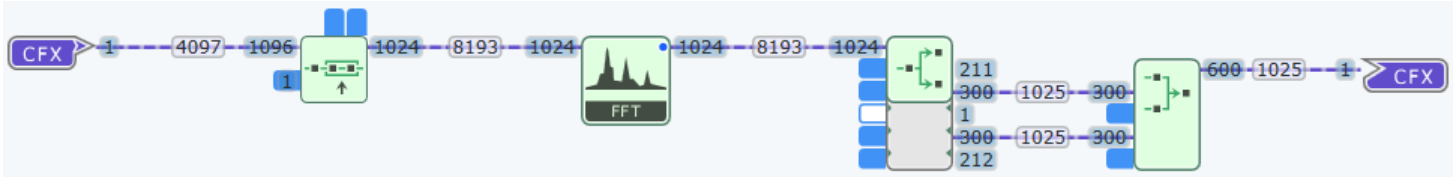
- Deploy to hardware with one click
- Prototype with real-world signals

Collaborative Design Team

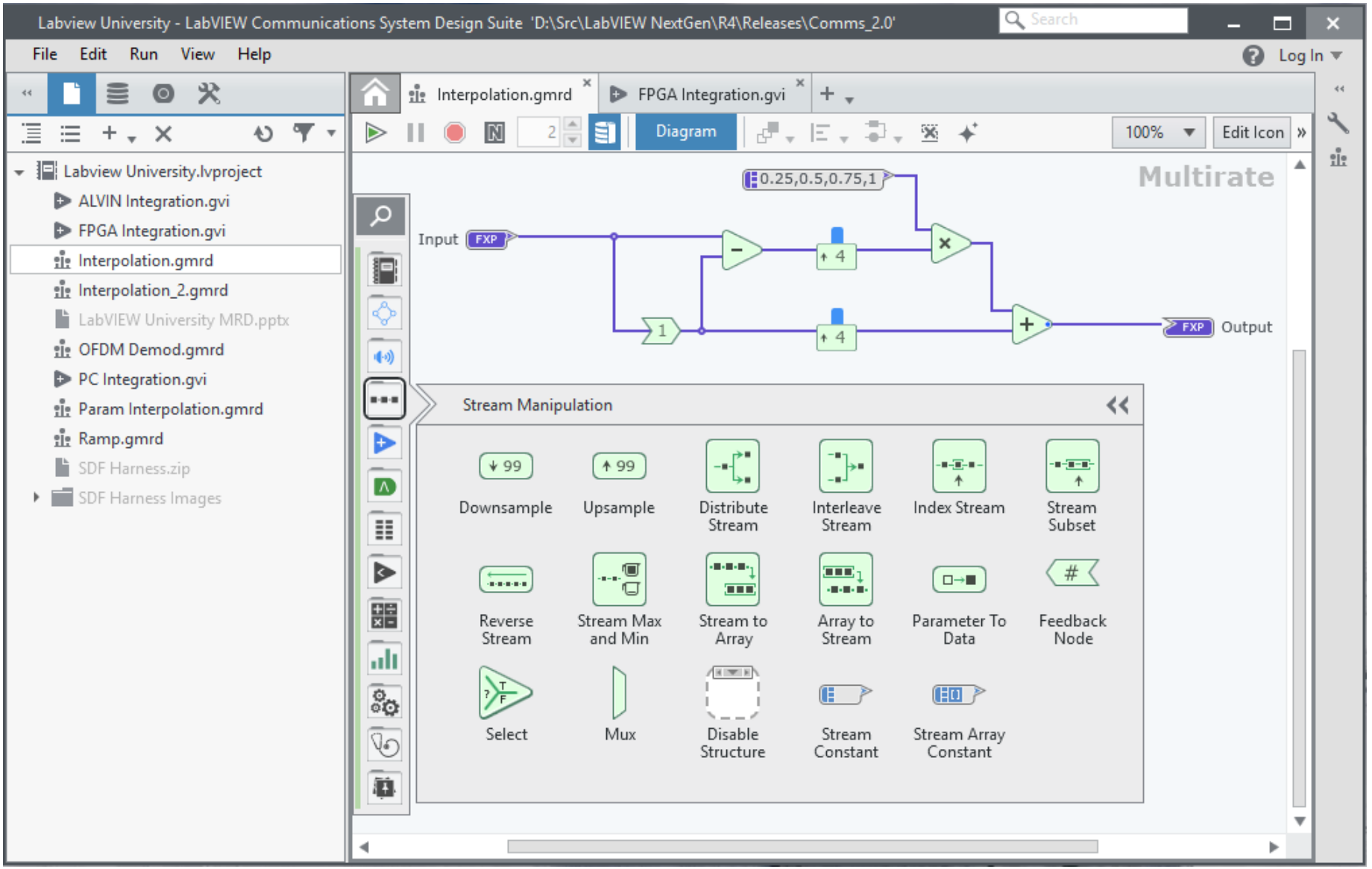
Multi-Rate Diagram

Enable high-level streaming Digital Signal Processing (DSP) algorithm implementation in hardware

- Parameterized SDF (Multi-Rate) dataflow
- Side-effect free
- Concurrent execution

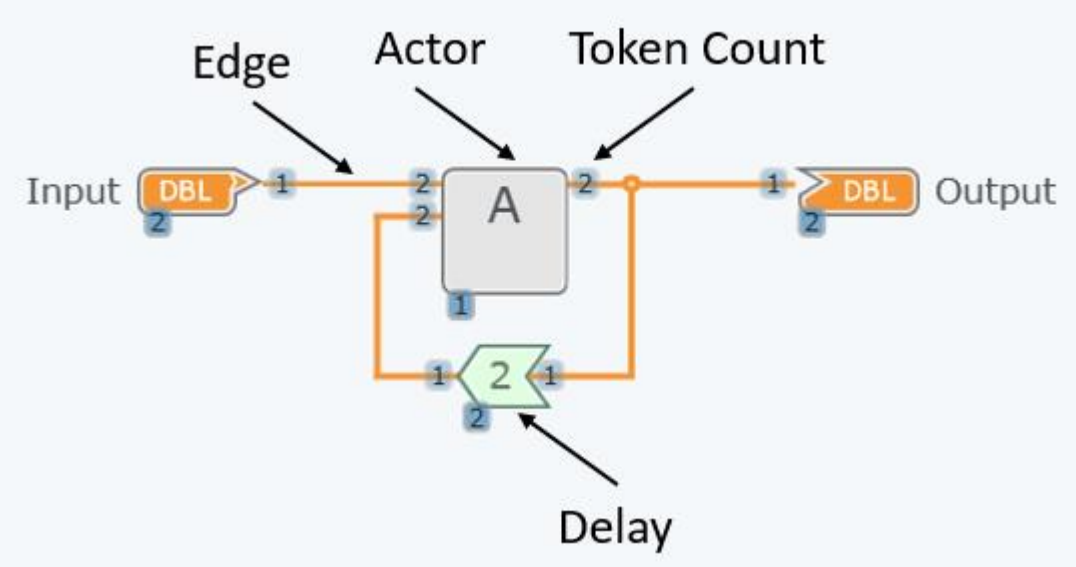


NATIONAL INSTRUMENTS



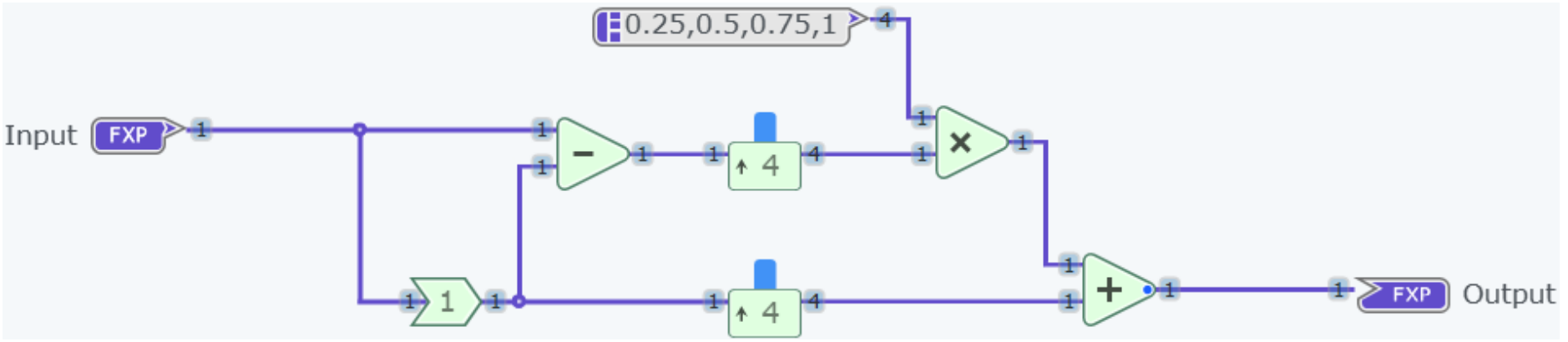
NATIONAL INSTRUMENTS

SDF (MRD) Model of Computation



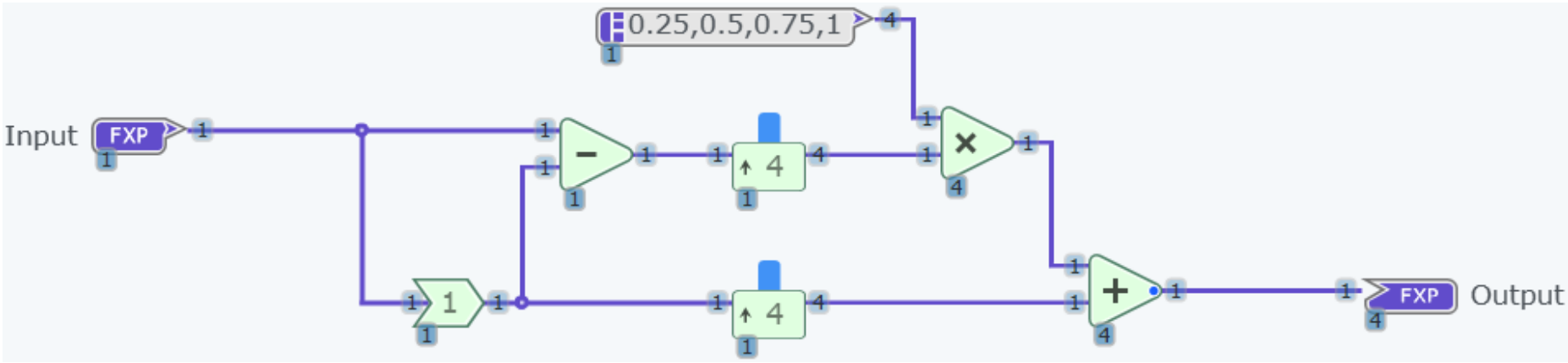
NATIONAL INSTRUMENTS

Example: Linear Interpolation



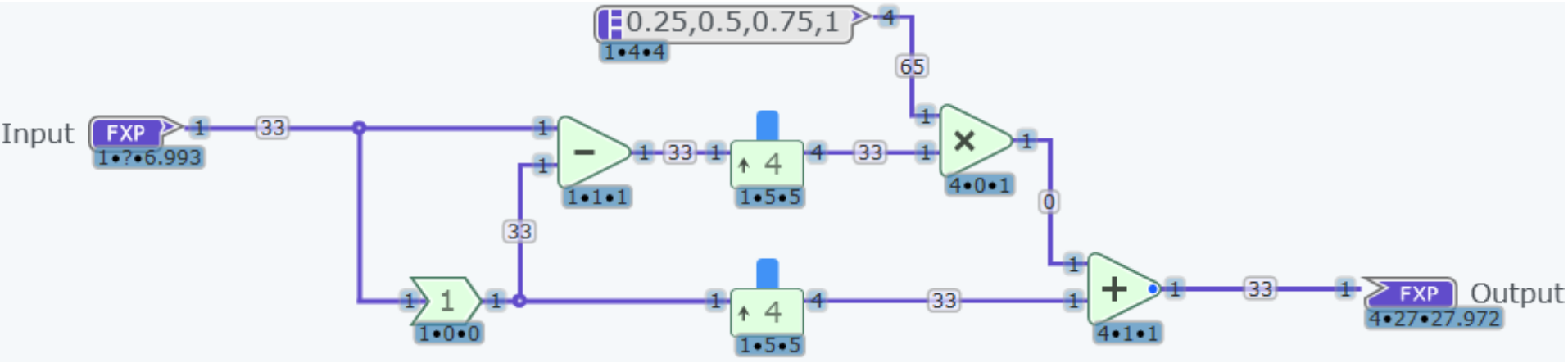
NATIONAL INSTRUMENTS

Example: Linear Interpolation

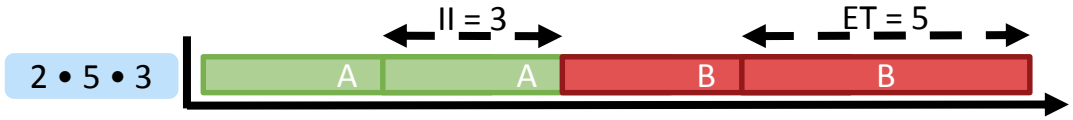


NATIONAL INSTRUMENTS

Example: Linear Interpolation

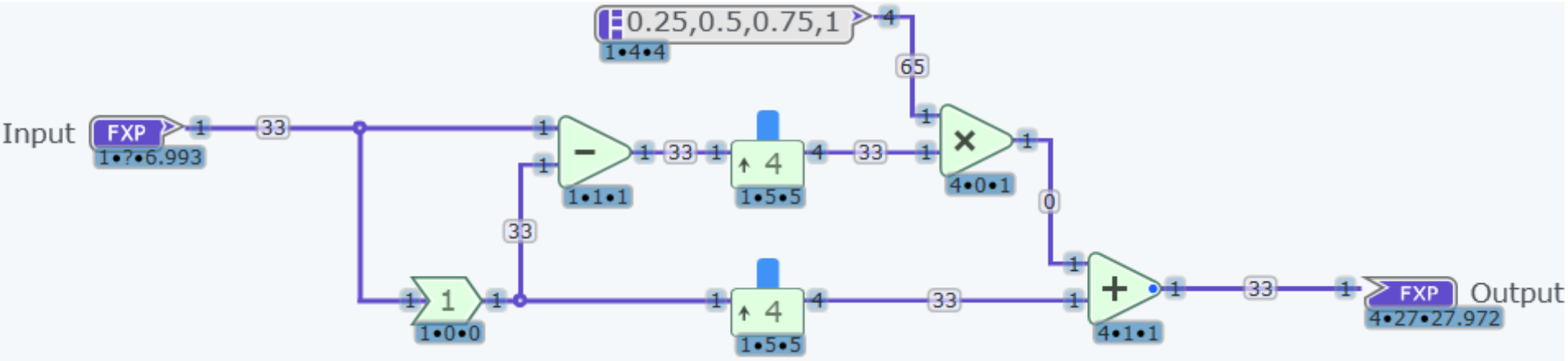


Repetition Count • Execution Time • Initiation Interval

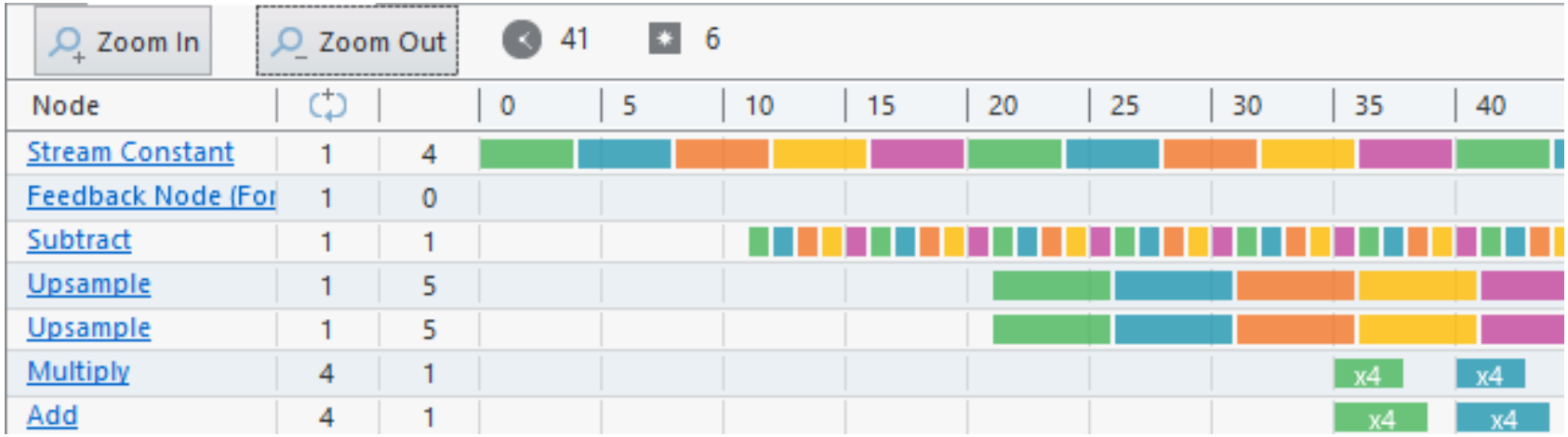


NATIONAL INSTRUMENTS

Example: Linear Interpolation

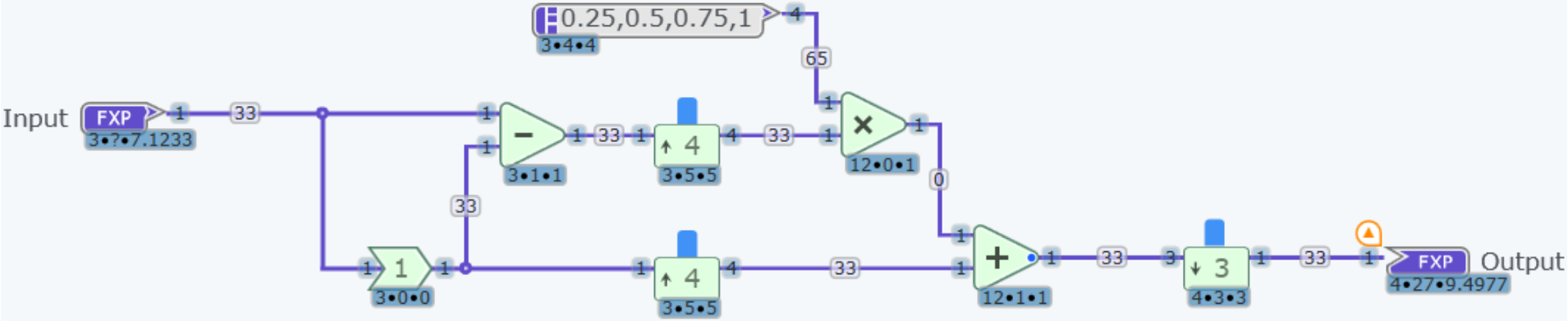


Repetition Count • Execution Time • Initiation Interval

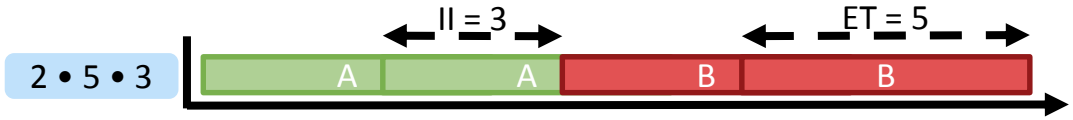


NATIONAL INSTRUMENTS

Example: Linear Interpolation

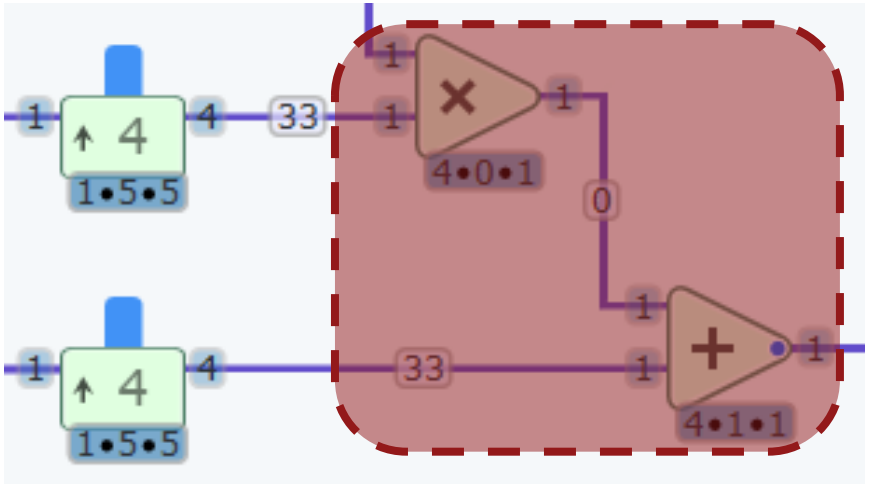


Repetition Count • Execution Time • Initiation Interval



NATIONAL INSTRUMENTS

Example: Linear Interpolation Optimization and Timing



Clump

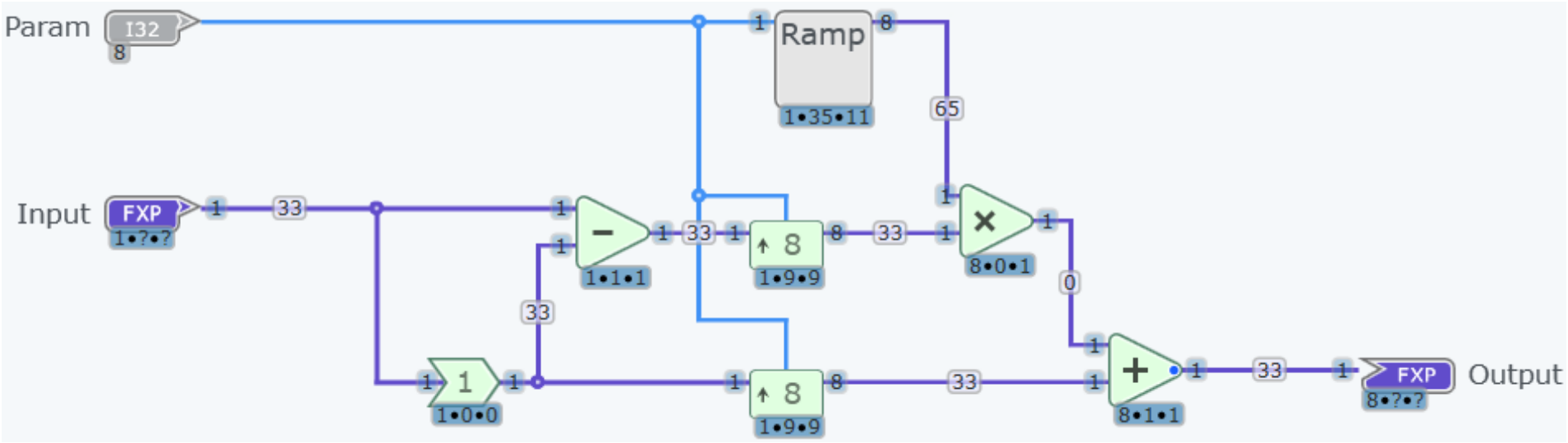
Zoom In | Zoom Out | 41 | 6

Node			0	5	10	15	20	25	30	35	40
Stream Constant	1	4									
Feedback Node (For	1	0									
Subtract	1	1									
Upsample	1	5									
Upsample	1	5									
Multiply	4	1								x4	x4
Add	4	1								x4	x4

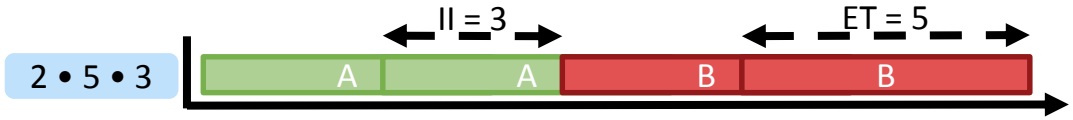
Schedule

NATIONAL INSTRUMENTS

Example: Parametric Linear Interpolation

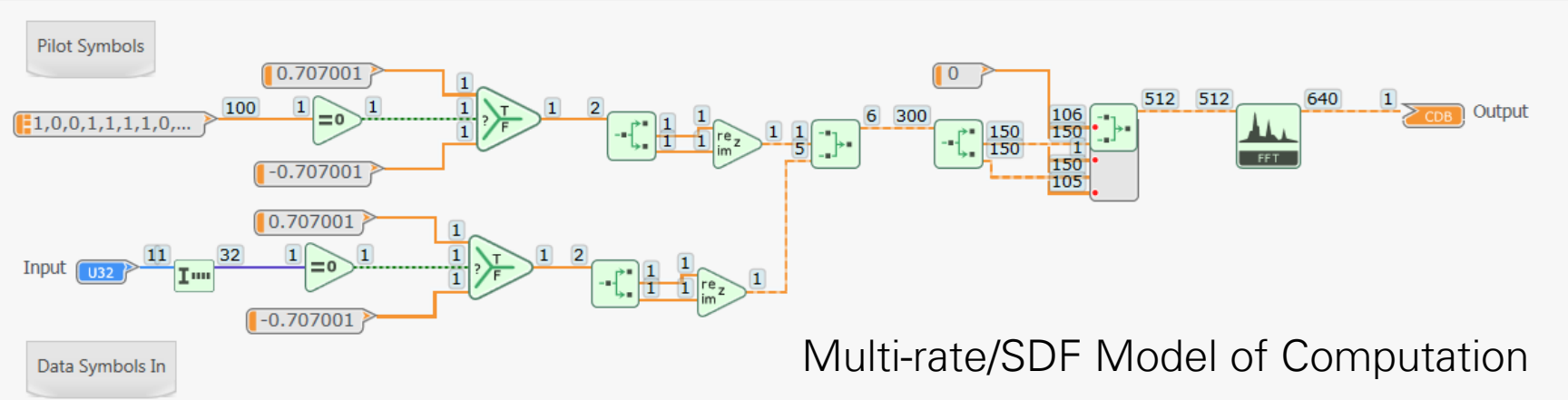


Repetition Count • Execution Time • Initiation Interval



NATIONAL INSTRUMENTS

Example: OFDM Transmitter



Multi-rate/SDF Model of Computation

5 MHz, LTE-Like Design

- **Symbol Mapping**: 4 QAM
- **Data/Pilot Structure**: 1 Pilot (reference) for every 5 Data Symbols
- **Frame Structure**: 512 Elements [106 Zeros, 150 Data/Pilot, 1 Zero, 150 Data/Pilot, 105 Zeros]
- **Cyclic Prefix Length**: 128

NATIONAL INSTRUMENTS

Unified Design Flow

Single, Cohesive Toolchain

Algorithm Development

- C and .M Support
- Algorithm design languages

System Mapping

- Hardware aware environment
- Easy, iterative design partitioning

Design Exploration

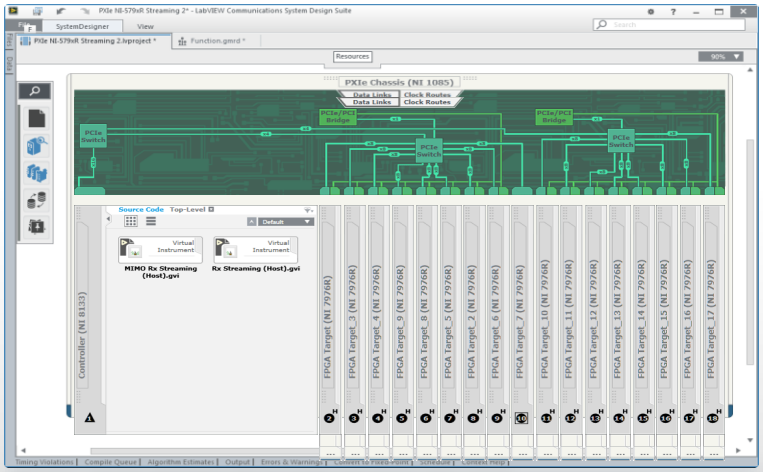
- Design simulation tools built in
- Analyze tradeoffs in implementations

System Implementation

- Deploy to hardware with one click
- Prototype with real-world signals

Collaborative Design Team

Hardware Aware Design Environment



Interactive, visual representation of the physical system which:

- Enables system discovery and verification of system setup
- Provides hardware documentation and visualization of available resources
- Allows for design partitioning and deployment
- Enables articulation of system architecture

NATIONAL INSTRUMENTS

Unified Design Flow

Single, Cohesive Toolchain

Algorithm Development

- C and .M Support
- Algorithm design languages

System Mapping

- Hardware aware environment
- Easy, iterative design partitioning

Design Exploration

- Design simulation tools built in
- Analyze tradeoffs in implementations

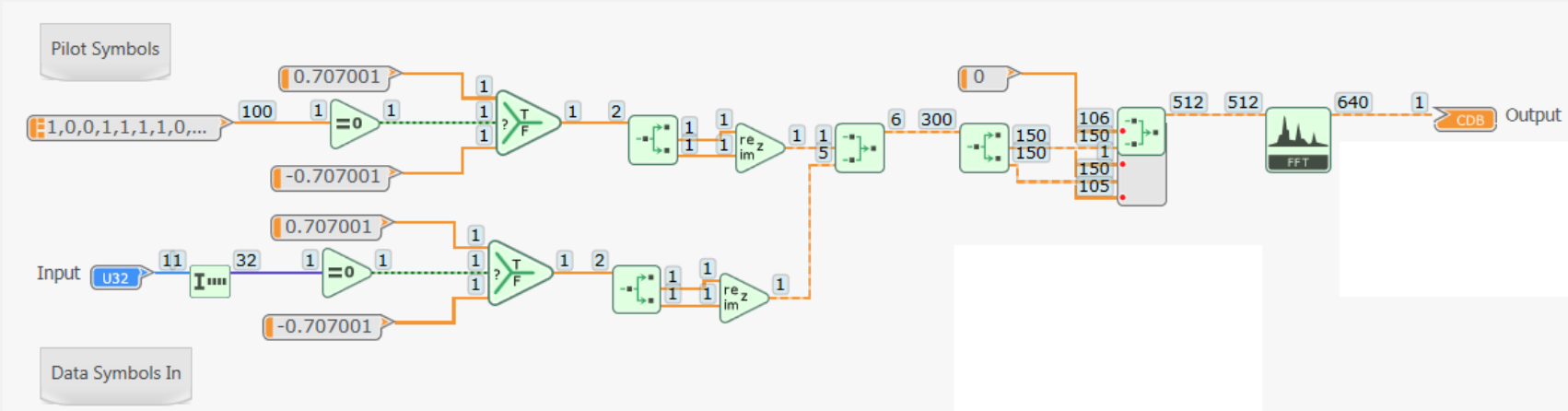
System Implementation

- Deploy to hardware with one click
- Prototype with real-world signals

Collaborative Design Team

NATIONAL INSTRUMENTS

Design Exploration for FPGA Deployment



Floating point design ➡ Fixpoint design

Performance constraints:

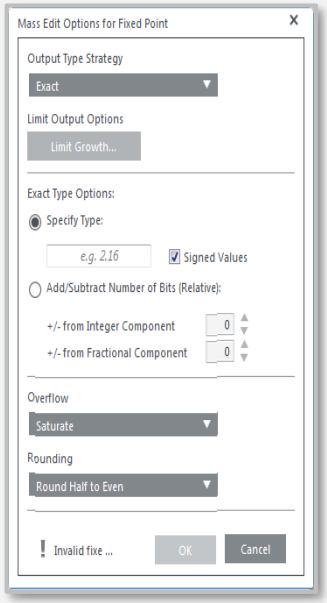
Throughput, latency, resources

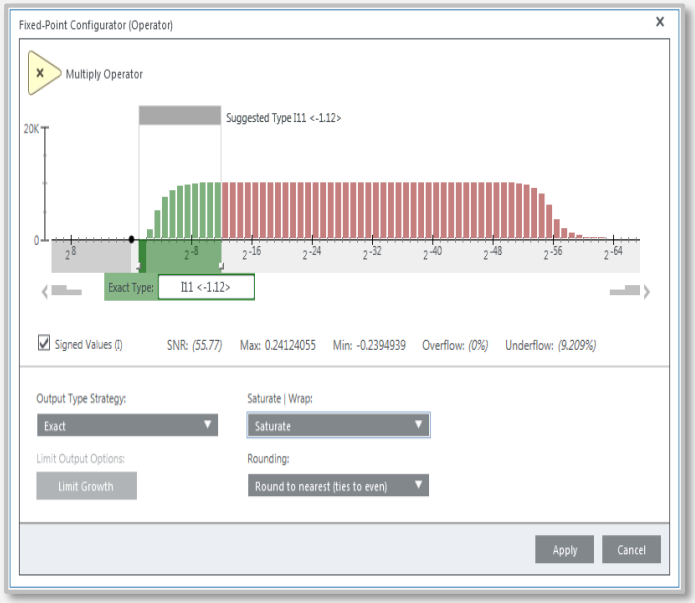
Simulation capability

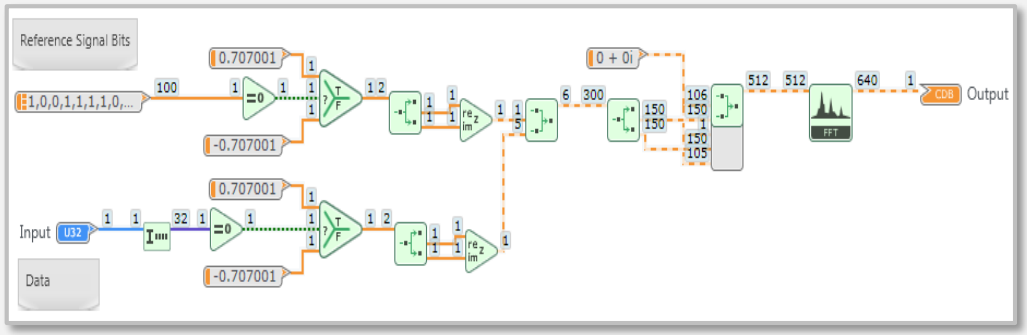
NATIONAL INSTRUMENTS

Design Exploration for FPGA Deployment

Float to Fix Point Conversion with a **data-driven approach**





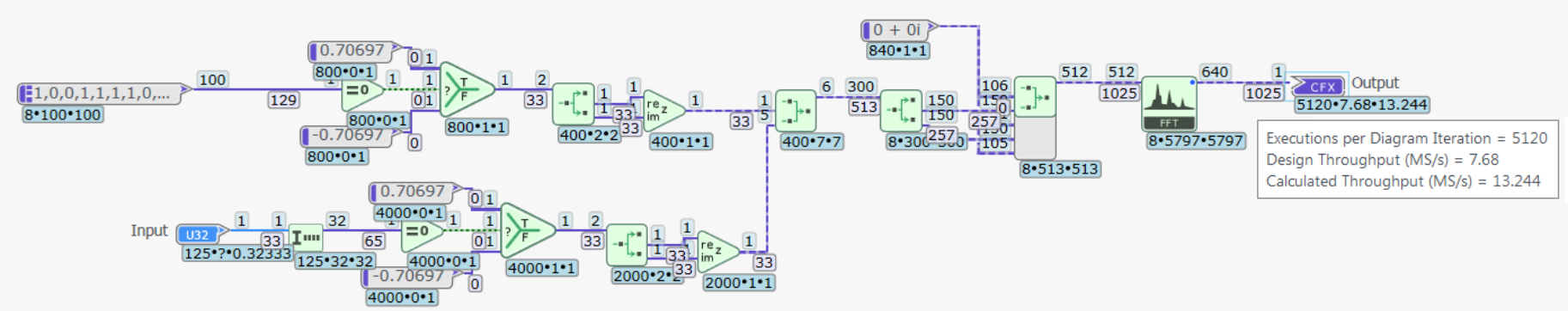


NATIONAL INSTRUMENTS

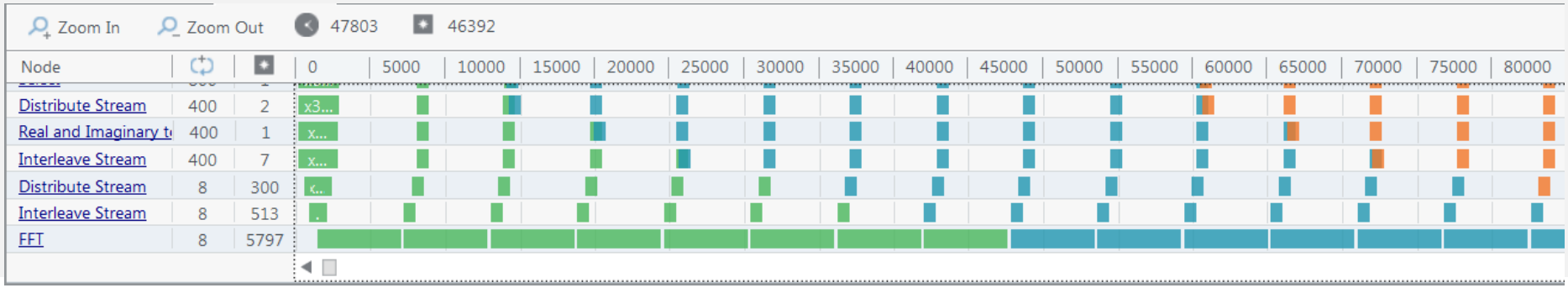
Design Exploration for FPGA Deployment

Feedback on design based on constraints:

- Actual throughput
- Buffer sizes



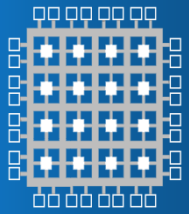
Schedule View to analyze where the design can be further optimized

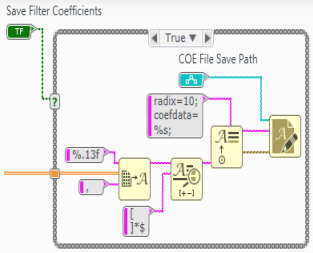


NATIONAL INSTRUMENTS

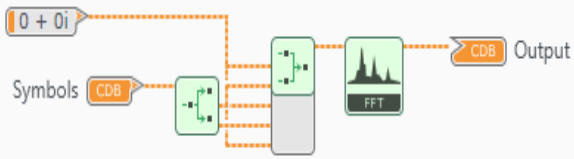
Algorithm Design Languages: FPGA

FPGA

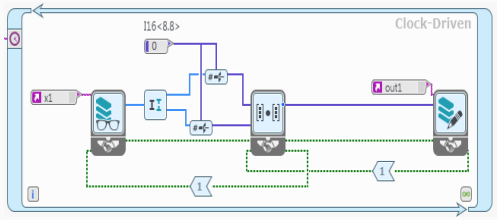




- FPGA IP: Container for preparing graphical dataflow for FPGA
- Can simulate and estimate resources



- Multirate diagram: Algorithmic design language for defining synchronous dataflow – ideal for stitching IP
- Compiler handles buffering and data transfer



- Clock Driven Logic: Low level language for optimizing performance on FPGA

NATIONAL INSTRUMENTS

Unified Design Flow

Single, Cohesive Toolchain

Algorithm Development

- C and .M Support
- Algorithm design languages

System Mapping

- Hardware aware environment
- Easy, iterative design partitioning

Design Exploration

- Design simulation tools built in
- Analyze tradeoffs in implementations

System Implementation

- Deploy to hardware with one click
- Prototype with real-world signals

Collaborative Design Team

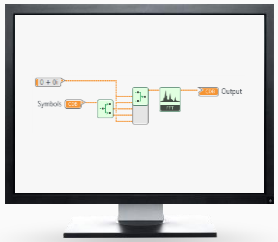
NATIONAL INSTRUMENTS

Unified Design Flow

Single, Cohesive Toolchain

Algorithm Development

- C and .M Support
- Algorithm design languages



System Mapping

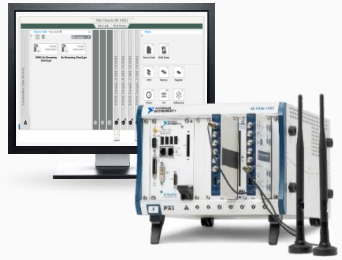
- Hardware aware environment
- Easy, iterative design partitioning

Design Exploration

- Design simulation tools built in
- Analyze tradeoffs in implementations

System Implementation

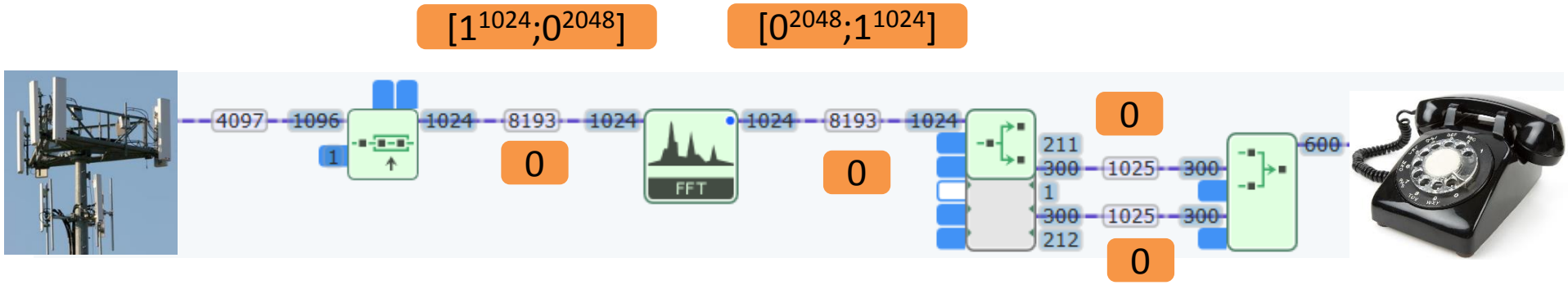
- Deploy to hardware with one click
- Prototype with real-world signals



Collaborative Design Team

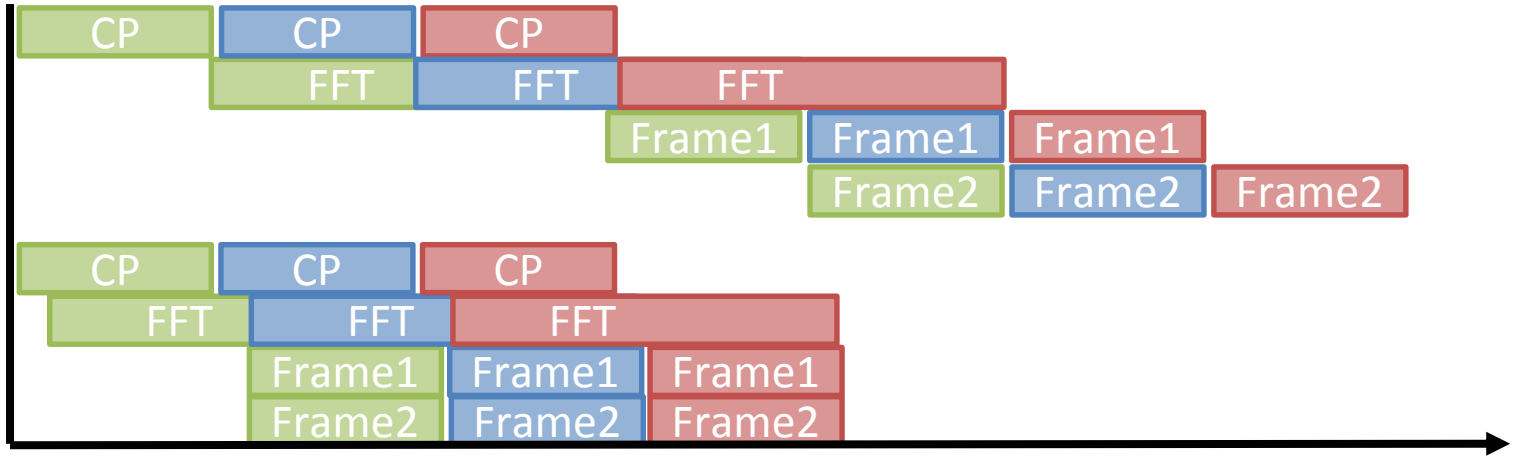
NATIONAL INSTRUMENTS

Access Pattern Optimizations on FPGA



Without Access Patterns

With Access Patterns



Stavros Tripakis, Hugo Andrade, Arkadeb Ghosal, Rhishikesh Limaye, Kaushik Ravindran, Guoqiang Gerald Wang, Guang Yang, Jacob Korneup, Ian Wong. "Correct and Non-Defensive Glue Design using Abstract Models". CODES+ISSS '11: Proceedings of the seventh IEEE/ACM/IFIP international conference on Hardware/software codesign and system synthesis, ACM, 59-68, October, 2011.

NATIONAL INSTRUMENTS

Graphical System Design

A Platform-Based Approach for Measurement and Control

Test

Monitor

Embedded

Control

Mechatronics

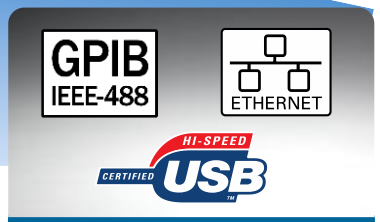
NATIONAL INSTRUMENTS

LabVIEW™

Desktops and PC-Based DAQ

PXI and Modular Instruments

NI CompactRIO and Custom Designs



Open Connectivity With Third-Party I/O