Graphical System Design

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

- Ph.D. UT Computer Sciences 1997
  - Parallel, functional programming
- Assistant Professor 1997-99
  - ECE Dept. SMU, Dallas
- Principal SW Architect, National Instruments, 1999 -
  - Real-time systems
  - Programming with time
  - Real-time networking
  - Models of Computation
  - High Level Synthesis for FPGAs
  - Research projects with Drs. Gerstlauer and Evans
Agenda for Today

- Embedded system design
- Platform based design
- Models of Computation
- Real-time streaming applications
- System design tools
National Instruments

- Revenue: $1.14 Billion in 2013
- Global Operations: Approximately 6,870 employees; operations in more than 40 countries
- Broad customer base: More than 35,000 companies served annually
- Diversity: No industry >15% of revenue
- Culture: Ranked among top 25 companies to work for worldwide by the Great Places to Work Institute
We Enable Graphical System Design
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
European Southern Observatory
Extremely Large Telescope

M1 Mirror
42 meters
ESO ELT M1 Mirror Control

984 MIRRORS

3,000 ACTUATORS

6,000 SENSORS

3k x 6k MATRIX

1 MILLISECOND
M4
Adaptive Mirror
2 TFLOPs

M1
200 GFLOPs (Global Control)
1 TOPs (Local Control cRIO)
Optical Coherence Tomography Research

Early Cancer Detection with LabVIEW & PXI

K. Ohbayashi
Kitasato University,
Center for Fundamental Sciences

~ 1.5 M FFTs / sec for Real-Time Performance
Business Trends
The Long Tail

- High Volume, Low Mix
- Large Development Teams
- Highly Customized Hardware

- Low Volume, High Mix
- Small Development Teams
- Custom Design on COTS
- Domain Experts

[“The Long Tail,” Chris Anderson Wired, 2004]
Design discontinuities in EDA tools

We want to predict this:

Results (Design Productivity)

1978

1985

1992

1999

Synthesis - Cadence, Synopsys

Schematic Entry - Daisy, Mentor, Valid

Transistor entry - Calma, Computervision

McKinsey S-Curve

Effort (EDA tools effort)

2

What’s next?

[1] Kurt Keutzer, UC Berkeley EECS 244 class
Addressing Design Discontinuity

• New Methodology maps from the higher level abstraction down to the reliable foundation

  ▪ A design-entry approach that offers 10X productivity improvement.
  ▪ A functional verification approach that offers 10-100X speed-up in verification
  ▪ An implementation approach that is predictable and reliable

National Instruments Vision Evolved

“To do for embedded what the PC did for the desktop.”

Graphical System Design

Virtual Instrumentation
- Complex instrumentation
- RF
- Digital
- Distributed

Real-time measurements
- Embedded monitoring
- Hardware in the loop

Embedded Systems
- Industrial control
- RT/FPGA systems
- Electronic devices
- C code generation

ni.com
The Next 30 Years: Expanding LabVIEW into System Design
Trends in Embedded Software
Alberto Sangiovanni-Vincentelli, UC Berkeley

“The design of embedded systems is becoming more difficult as design complexity increases, time-to-market pressures continue, and development teams with diverse backgrounds are assembled. The platform-based design methodology (PBD) is a technique to combat these challenges.” [5]

“Given the cost and risks associated to developing hardware solutions, an increasing number of companies is selecting hardware platforms that can be customized by reconfiguration and/or by software programmability. In particular, software is taking the lion’s share of the implementation budgets and cost. In cell phones, more than 1 million lines of code is standard today, while in automobiles the estimated number of lines by 2010 is in the order of hundreds of millions.” [6]

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,
The Y-Chart System Design Methodology

Application Logic  →  Analysis & Mapping  →  Performance Evaluation  →  Platform Architecture


Platform Dimensions

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

- Algorithm development
- IO characterization
  - Timing characteristics
- Real-time constraints
- Integrating Models of Computation
- State management
Trends in Embedded Software

Edward Lee, UC Berkeley

“The principal role of embedded software is interaction with the physical world. Consequently, the designer of that software should be the person who best understands that physical world.” [domain expert] [1]

“The engineers that write embedded software are rarely computer scientists. They are experts in the application domain with a good understanding of the target architectures they work with.” [1]

“Design of embedded software will require models of computation that support concurrency.” [1]

“In embedded software, concurrency and time are essential aspects of a design.” [2]

Distributed Computing

LabVIEW Today

Multiple Programming Models

ni.com

NATIONAL INSTRUMENTS
Models of Computation

Idea

Model of Computation

Implementation

Multiply / Accumulate
MMSE Equalizer (matrix inversion) problem:

$$\hat{x} = \left(\hat{H}^*\hat{H} + \sigma^2 I\right)^{-1}\hat{H}^*y$$
“You need a high-level language to bring FPGA to the masses, people who are not essentially hardware programmers. We think LabVIEW is a very good tool that enables the domain experts to program FPGAs…”

– Ivo Bolsens, CTO, Xilinx
LabVIEW-Based System Design Tools

• Current project at National Instruments
  ▪ Exploration of concepts and directions

• 2 arcs of exploration
  ▪ System Design Tool
  ▪ DSP algorithm development
Looking Closer at DSP Design

- Focus on DSP streaming applications running on FPGAs
- Explore the right Models of Computation (MoCs)
- Provide analysis and optimization
  - Throughput, latency, area
- Allow simulation and test bed creation
- Generate performing VHDL code
- Provide debugging capabilities
MoCs for Streaming Applications

Key trade-off: Analyzability vs. Expressibility
Dataflow Execution

• LabVIEW Dataflow
Multirate Execution

• LabVIEW Execution

• Asynchronous Execution
Platforms for FPGA-based System Design

HDL/RTL

LabVIEW Today

LabVIEW DSP Design Module Early Access Program
Making Applications Faster to Design

Karplus-Strong Plucked String Algorithm

DSP Diagram
High-Speed Streaming is Complex Today

- Challenges
  - LabVIEW G model
    - Original specification from algorithm designer
    - Not feasible for highly efficient implementation on FPGA targets
  - Implementation challenges
    - Floating to fixed point conversion
    - Array data to point-by-point data conversion
    - Explicit concurrency representation
    - FPGA target constraints
    - Integration with internal and third-party IP

ni.com
Domain Expert Expectations for High-Speed Streaming

- High-level DSP representation that matches algorithm theory
  - Algorithms written independently of hardware target
  - Deal in domain terms of token rate, throughput, and latency
- Explore high-level design tradeoffs without diving into implementation details
  - Tune performance with high-level constraints
  - Access the details if needed
OFDM Transmitter
20 MHz LTE Transmission Band

- Rapidly prototype PHY layer baseband DSP
- Co-design PHY layer algorithms with higher MAC layer protocols
- Seamless integration with wide array of RF hardware
DSP Design Module Value
Enable an algorithm designer to specify an intuitive diagram that generates real-time DSP implementations on FPGAs
RF/Communications PHY FPGA Software

Graphical System Design Platform

LabVIEW DSP Diagram Module

LabVIEW FPGA

LabVIEW IP

LabVIEW FPGA IP Builder

3rd Party IP

RIO Hardware
Summary

• Complexity of system design
• New productivity tools
• System design
  ▪ Design distributed systems
• DSP Design
  ▪ Design DSP algorithms