Facilitating Cross-FPGA Platform Designs Using “Virtual” Platforms

Angshuman Parashar
Michael Adler
Joel Emer

VSSAD, Intel

{angshuman.parashar, michael.adler, joel.emer}@intel.com
Outline

• History and Motivation
• Virtual Platforms
• Hybrid Hardware/Software Modules
• Demo
History and Motivation

Model
Top Level Wires

XUP Board
LEDs and Switches
(image source: xilinx.com)

Software Simulator
Processes
(image source: intel.com)

BDPI
History and Motivation

- XUP Model
  - Top Level Wires
- "Sim" Model
  - BDPI

XUP Board
- LEDs and Switches
  (image source: xilinx.com)

Software Simulator
- Processes
  (image source: intel.com)
History and Motivation

Platform Independent Model

Virtual Platform

XUP Front Panel
Top Level Wires

“Sim” Front Panel
BDPI

Plug N Play

front_panel.writeLED()

XUP Board
LEDs and Switches
(image source: xilinx.com)

Software Simulator
Processes
(image source: intel.com)
Virtual Platform

• Set of Abstractions
  – Provide common set of functionalities across multiple physical-platforms
    • Intel FSB
    • Bluesim/Vsim
    • PCI-express
    • XUP
  – Leverage Asim Plug N Play
    • Minimize module replacements/recoding while moving across platforms
  – Functionality + Efficiency
Virtual Platform

Timing + Functional Modules

Virtual Platform

Platform Interface

Virtual Devices

Device 0

Device 1

Device 2

Channel IO

Physical Platform
Virtual Platform on XUP

Timing + Functional Modules

Virtual Platform

Virtual Devices

Front Panel

VGA

Memory

Platform Interface

LEDs

Switches

VGA

On-Board Memory
Virtual Platform on Simulator

**Timing + Functional Modules**

- Decode
- Fetch
- Exe
- Mem

**Virtual Platform**

- Platform Interface
- Virtual Devices
  - Front Panel
  - VGA
  - Memory

**Channel IO**

**Bluesim**

**Std. I/O**

**Soft Connections**
Virtual Platform on Intel FSB

**Timing + Functional Modules**
- Decode
- Exe
- Fetch
- Mem

**Virtual Platform**
- Platform Interface
- Virtual Devices
  - Front Panel
  - VGA
  - Memory
- Channel IO
- FSB

**Software Support**
Hybrid Hardware/Software Modules

• Split module functionality between FPGA and software
• Leverage Virtual Platform Infrastructure
Hybrid Modules

FPGA Modules

- Fetch
- Decode
- Exe
- Control
- Func Model

Software Modules

- Front Panel
- Memory
- Decode
- Front Panel
- Memory

Virtual Platform

HAsim Remote Request/Response (HRRR)

HRRR Client + Server

Channel IO

Hardware

Software
HRRR: Intel FSB

**FPGA Modules**

- Fetch
- Decode
- Exe
- Control
- Func Model

**Virtual Platform**

- Platform Interface
- Front Panel
- Memory
- HRRR Client + Server
- Channel IO

**Software Modules**

- Front Panel
- Memory
- HRRR Client + Server
- Decode
- Channel IO
- Driver

**Hardware**
HRRR: Simulator

**Virtual Platform**
- Platform Interface
- Front Panel
- Memory

**HRRR Client + Server**
- Channel IO

**FPGA Modules**
- Fetch
- Decode
- Exe
- Control
- Func Model

**Software Modules**
- Front Panel
- Memory
- Decode
- HRRR Client + Server

**Hardware**
- **Software**
- UNIX pipe
Hybrid Modules

- **Server module**
  - Publishes the following:
    - `init()` method name (e.g. `fetch_server_init()`)  
    - RRR service string (e.g. “FETCH”)
  - Build process collects these and generates global services table
  - Server main reads services table (during pre-processing) and registers all service modules by calling `init()`

- **Client module**
  - Reads global services table
    - `serviceID = serviceTable.search("FETCH");`
    - `reqID = RRRClient.sendReq(serviceID, params...)`
    - `result = RRRClient.getResp(reqID);`
Hybrid Modules (cont.)

- **Module definition**
  - `hybrid_fetch_unit.awb`
    - `%sources -t BSV -v PUBLIC hardware_fetch_unit.bsv`
    - `%sources -t CPP -v PUBLIC software_fetch_unit.h`
    - `%sources -t CPP -v PRIVATE software_fetch_unit.cpp`

- **Build process**
  - Collects BSVs and generates “hardware” bitfile or simulation binary
  - Collects CPPs and generates “software” binary

- **Runtime**
  - Software binary loads bitfile onto FPGA, or forks off simulation “hardware” binary
  - Software sends “start” HRRR request to Hardware
Demo
Backup Slides
“Sim” Front Panel
Hybrid Front Panel
Running on Bluesim…