EN2911X: Reconfigurable Computing
Topic 00: Introduction

Prof. Sherief Reda
School of Engineering, Brown University
Fall 2014
Methods for executing computations

Hardware
(Application Specific Integrated Circuits)

Advantages:
• very high performance and efficient

Disadvantages:
• not flexible (can’t be altered after fabrication)
• expensive

Reconfigurable computing

Advantages:
• fills the gap between hardware and software
• much higher performance than software
• higher level of flexibility than hardware

Software-programmed processors

Advantages:
• software is very flexible to change

Disadvantages:
• performance can suffer if clock is not fast
• fixed instruction set by hardware
Temporal vs. spatial based computing

Temporal-based execution (software)

- `add` r2, r4, r6
- `add` r2, r3, r4
- `sub` r5, r3, r1
- `add` r4, r1, r2
- `cmp` r6, r4

Spatial-based execution (reconfigurable computing)

- `add` r2, r4
- `sub` r5, r3
- `cmp` r6, r4
- `add` r2
- `sub` r1
- `cmp` r2, r6

Ability to extract parallelism (or concurrency) from algorithmic descriptions is the key to acceleration using reconfigurable computing.
• **Field-Programmable Gate Arrays (FGPAs)** are one example of reconfigurable devices
• An FPGA consists of an array of *programmable logic blocks* whose functionality is determined by programmable configuration bits
• The logic blocks are connected by a set of *routing resources* that are also programmable
  - Custom logic circuits can be *mapped* to the reconfigurable fabric
Configuring FPGAs

FPGAs can be dynamically reprogrammed before runtime or during runtime (virtual hardware)

- full
- partial

[Maxfield’ 04]
Uses of reconfigurable devices

1. Low/med volume IC production
2. Early prototyping and logic emulation
3. Accelerating algorithms in reconfigurable computing environments
   i. Reconfigurable functional units within a host processor (custom instructions)
   ii. Reconfigurable units used as coprocessors
   iii. Reconfigurable units that are accessed through external I/O or a network

[Compton’ 02]
Why reconfigurable computing is more relevant in emerging computing systems?

• Significant demand for high-performance computation. Large-scale data processing required in many areas (e.g. signal processing, particle physics, weather simulations, bioinformatics)

• Why general-purpose processors are not meeting the demand?
  – Single thread performance is no longer improving (individual core frequencies do not increase due to thermal problems).
  – Thread-level parallelism does not always scale.
  – Consume large amount of power.

• Why reconfigurable fabrics could meet the computational demand?
  – Provide the spatial computational resources required to computationally process large streams of data directly in hardware
  – Leads to power savings
  – Economically feasible
Goals of this class

1. Learn principles of reconfigurable computing.
2. Acquire hands-on experience and useful implementation skills for real-life engineering.
3. Develop/strengthen research skills through readings and class projects.
Topics covered in EN2911X

Need to organize based on topics discussed in class
1. Overview of programmable logic technology
2. Hardware programming languages
3. Reconfigurable computing methodologies
4. Algorithmic acceleration using reconfigurable computing
5. Heterogeneous reconfigurable systems
6. Emerging-technology programmable devices
Programming information could be stored in SRAM. A 4-input Look-Up Table (LUT) is the typical size.
Topic 01: Programmable logic technology overview
Topic 01: FPGA board for lab

No need to design our board; we will use Altera’s DE2 board and Quartus II software.

Features:
- Cyclone II FPGA 35K LUTs
- 10/100 Ethernet
- RS232
- Video out (VGA 10-bit DAC)
- Video in (NTSC/PAL/multi-format)
- USB 2.0 (type A and type B)
- PS/2 mouse or keyboard port
- Line in/out, microphone in (24-bit Audio CODEC)
- Expansion headers (76 signal pins)
- Infrared port
- Memory 8-MBytes SDRAM, 512K SRAM, 4-MBytes flash
- SD memory card slot
- Displays 16 x 2 LCD display
- Eight 7-segment displays
- Switches and LEDs
Topic 02: Hardware programming languages (Verilog)

- Verilog is a hardware description language used to model digital systems
- Similar in syntax to C
- Differs from conventional programming languages as the execution of statements is not strictly linear. Possible to have sequential and concurrent execution statements
- The language can be synthesized into logic circuits

module mux(a, b, select, y);
input a, b, select;
output y;
initial
begin
    always @ (a or b or select) 
        if (select) 
            y = a;
        else 
            y = b;
    end
endmodule
Topic 03: Reconfigurable computing methodologies

System Specification

| software | partitioning | hardware |

compile for target processor

- Graphical State Diagram
- Textual HDL
- Top-level block-level schematic
- Graphical Flowchart
- Block-level schematic

synthesis (compilation)

Mapping (placement & routing)

configuration data
Topic 04: Application acceleration using reconfigurable computing

- Learn about hard and soft processors
- Design multi-core-based reconfigurable computing systems
- Design of on-chip networks for multi-core systems
- Design of custom instructions
- Design of pluggable acceleration function units
Topic 05: Heterogeneous reconfigurable computing systems

1. Understanding the niche of FPGAs among other computing platforms such as GPGPUs, SoC, etc.
2. System-level integration methods for heterogeneous computing fabrics
Topic 06: Emerging reconfigurable systems

- Asynchronous FPGAs
- NanoPLA architectures
- 3D FPGAs
- MRAM FPGAs
- CNT-based FPGAs

[DeHon 04]
Class organization

• Grade assignments:

• Sources: papers, lecture slides, manuals, and book chapters.

• TA: Xin Zhan

• Class website:
  – http://scale.engin.brown.edu/classes/EN2911XF14