Fundamentals of Reconfigurable Computing

(ECGR6890)

Ron Sass

University of North Carolina at Charlotte

Fall 2008

Course Outline

Course Outline

Course Outline

Course Outline

Course Outline

Course Outline

Course Outline

Overview

Or, *What the heck is Reconfigurable Computing?*

What Is Reconfigurable Computing?

- ***reconfigurable computing*** — study of computing systems that incorporate user-programmable switches to determine functionality or architecture
- RC is seven less syllables
- in contrast, conventional computing systems use devices that
  - are fixed at manufacture
  - and have their function is pre-determined

Conventional Computing

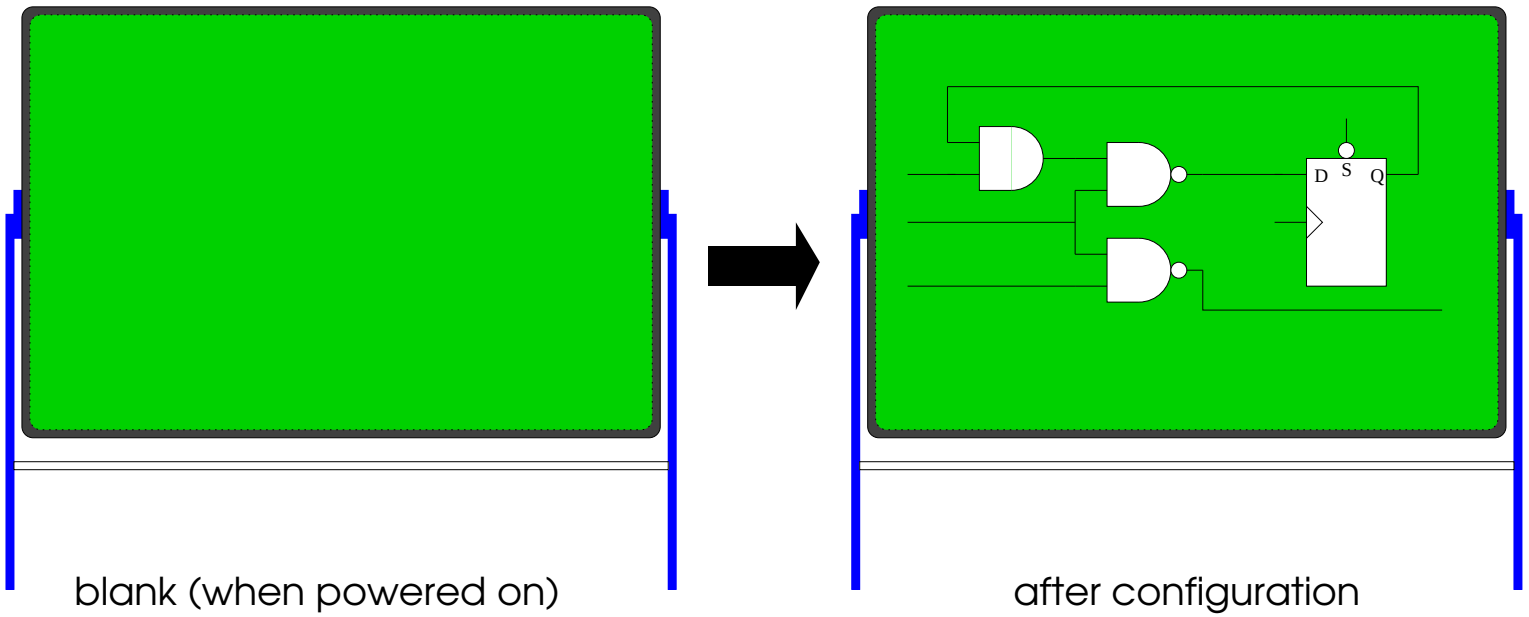
Contrast this with conventional computer architectures:

- von Neumann (EDVAC) introduced the first well-known computer architecture
  - stored program
  - sequential execution
- Harvard architecture
  - separated memory for code and data
  - (still) sequential execution

Most Common Technology for RC

- system composed of microprocessor(s) and Field-Programmable Gate Array(s) (FPGAs)
- ***FPGA*** — an Integrated Circuit (IC) that whose functionality can be configured in the field; i.e., its functionality can be determined after it has already been manufactured and soldered into a product
- many different ways to combine these components but for now envision the abstract picture

10,000-Meter View of An FPGA



FPGA is a blank slate that can be programmed to do almost any digital circuit

Other Configurable Technology

- FPGA-like technologies (CPLDs and other PLAs)
- Coarse-Grain Reconfigurable Arrays (CGRA)
- Polymorphic Computing Architectures (TRIPS, Raw, Monarch)
- Conventional Processor + Configuration Switches

RC Synonyms

- as with any new field (FPGAs are 20 years old) – different names emerge
- other names for the field:
  - custom computing machines (CCMs) (SRC)
  - configurable computing machines (VT)
  - Fixed+Variable (F+V) (Estrin)
  - Field-Programmable ...

What does the 'RE' in Reconfigurable Mean?

- two broad classes of Reconfigurable Computing machines based on when the switches are configured
  - ***Compile-Time Reconfiguration*** (CTR) — the architecture is defined before the application begins and remains fixed throughout the execution
  - ***Run-Time Reconfiguration*** (RTR) — the architecture is initially defined before the application begins but may change during execution

Conventional Computing Terms

- ***hardware*** — the electro/mechanical machine (the physical components of a computing system)
- ***software*** — the information that specifies the behavior of a computing system
- ***program*** (n) — the expression of software
- ***programming*** (v) — the act of creating a program

Reconfigurable Computing Terms

in programmable logic device, the behavior of the chip is specified like software but we use different terms:

- ***design*** — the information that specifies the behavior of a reconfigurable system
- ***configure*** (v) — transfer of a design to a reconfigurable device

Confusing Terms/Dual Meaning

- ‘program’ (v) is sometimes used a synonym for ‘configure’ as in...

  I am programming (configuring) the chip now.

- ‘design’ is short for *hardware design* and sometimes ‘hardware’ is subtituted for ‘design’ as in...

  Here’s the hardware part of my reconfigurable system.

RC Application Areas

- concisely: the very small and the very large
  - embedded computing systems
  - high-performance computing systems
- in other words: constrained systems

Embedded Computing Systems

- highly integrated: implement peripherals on chip plus customized operations
- replace traditional digital signal processing
- increasingly networked (support for different communication protocols)

Embedded Examples

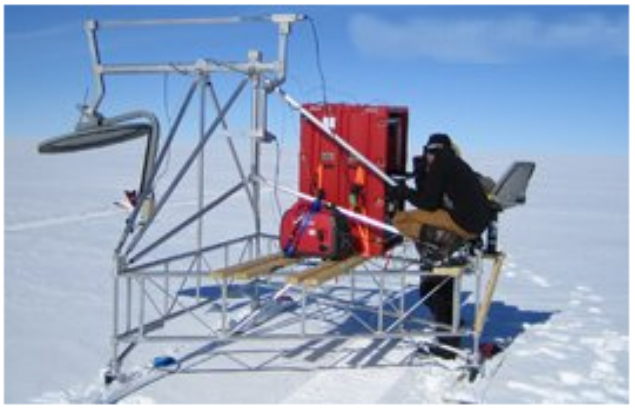
- soft radio

Embedded Examples

- soft radio
- Mars rover expeditions

Embedded Examples

- soft radio
- Mars rover expeditions
- measuring polar sheet ice depth



Embedded Examples

- soft radio
- Mars rover expeditions
- measuring polar sheet ice depth
- digital cameras

High-End Computing (HEC) Systems

- "hardware is faster than software" is a fallacy while it is not always true – a lot of times it is!
- processor-level parallelism
- power, space constrained systems

HEC Examples

- earliest machines
  - Splash-2
  - PAM
- contemporary supercomputing companies
  - Silicon Graphics (RASC)
  - Cray (XD-1)
  - SRC (SRC-6)
- research machines (Reconfigurable Computing Cluster)

Advantages of RC Technology

custom architectures have a number of big advantages:

- specialization produces more efficient (faster) circuits; especially if specialization occurs at run-time*
- less energy is spent on work that doesn't contribute to the solution
- trading space for time allows for lower clock rates and more energy efficient circuits
- smaller physical space; time-multiplexing offers greater gains*

*requires Run-Time Reconfiguration

Disadvantages of RC Technology

Sounds great, eh!? flexibility comes at a cost

- every circuit in an FPGA has a longer propagation delay
- automated tools are much less efficient that custom-designed CMOS
- Law of Diminishing Returns (Amdahl's Law) limitations
- I/O bandwidth can easily become the limiting factor

Summary

Net result:

- FPGA implementations incur a 10× to 100× penalty
- To get a performance improvement, one has first overcome this gap before we make gains on standard processor architectures

Outline for Today

- Introduction to Reconfigurable Computing (done :-)
- Mosaic Linux / Our Tools
- Review of Digital Systems