Digital System Design

by
Dr. Lesley Shannon
Email: lshannon@ensc.sfu.ca
Course Website: http://www.ensc.sfu.ca/~lshannon/courses/ensc350

Simon Fraser University

Slide Set Overview

- Course Motivation

- Logistical Details

- Course Lab Component

- Lecture Topics

What is a digital (electronic) system?

- Uses discrete values (1's and 0's) to represent electrical voltages

- Computed and stored values are discrete:
  - Value ranges are **_contiguous_** and not continuous

- (Symbolic) Logic equations can be used to represent both the control path and data path circuitry

Why does digital system design matter?

Why do digital systems matter?

- ## They're taking over!

  - The ratio of digital systems to analog systems is increasing, where analog circuitry is often just used to convert I/O to the digital domain

  - The number of devices that contain digital systems is constantly increasing
    - With the increasing popularity of portable devices, consumer electronics are exploding

Why do digital systems matter?

- **Significant area of employment for electronics and computer engineers**

  - Computer engineers are primarily concerned with hardware and software

    - **Hardware Rules!**

  - Electronics engineers are primarily concerned with digital and analog electronics

    - Dare to dream digital!

Digital Systems are *EVERYWHERE* …

Personal Computers

Prescott Pentium 4

122 mm^2 with 125 million transistors
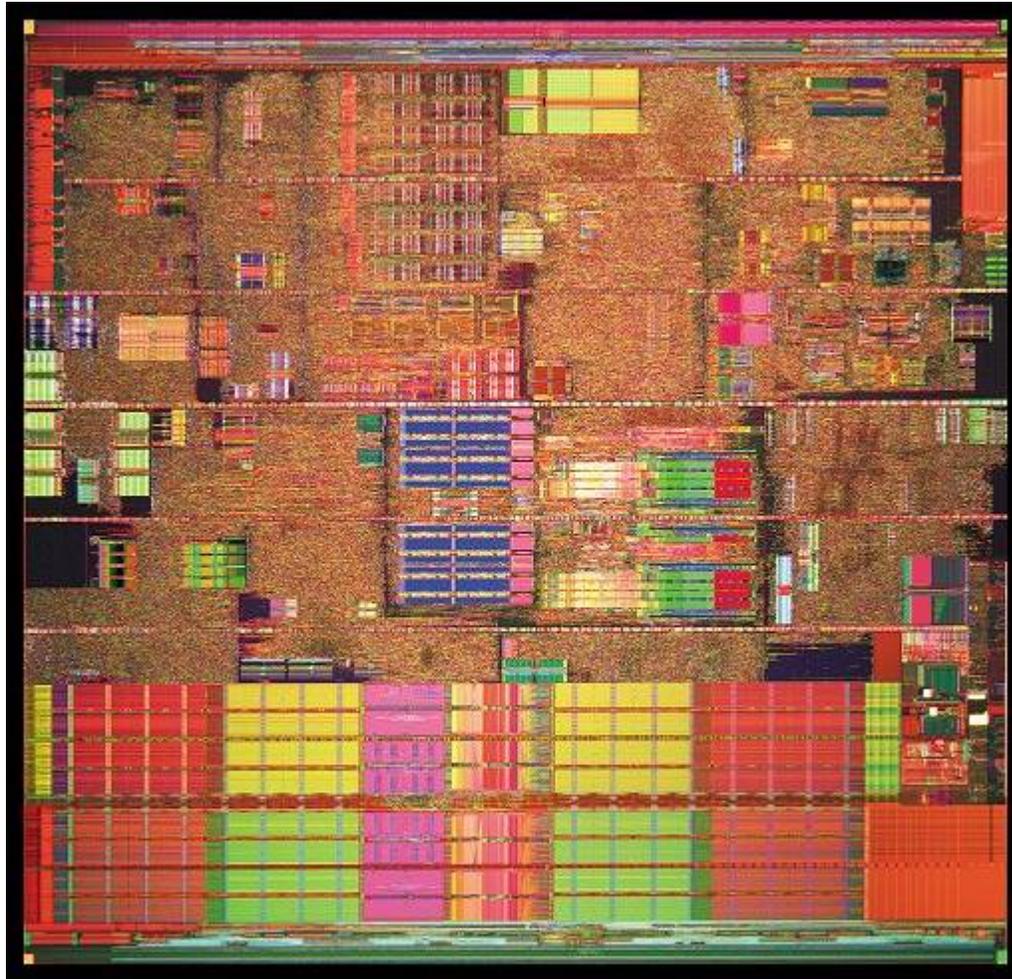
Intel Pentium 4 Prescott

Trace Cache Access, next Address Predict

Instruction Trace Cache

Execution Pipeline Start

Buffer Allocation & Register Rename

Trace Cache Branch Prediction Table (BTB), 1024 entries.

Return Stacks (4 x16 entries)

Trace Cache next IP's (4x)

Instruction Decoder

Up to 4 decoded uOps/cycle out (from max. one x86 instr/cycle) Instructions with more than four are handled by Micro Sequencer

Raw Instruction Bytes in

Data TLB, 64 entry fully associative, between threads dual ported (for loads and stores)

Front End Branch Prediction Tables (BTB), shared, 4096 entries in total

Instruction TLB's 128 entry, fully associative for 4k and 4M pages. In: Virtual address [47:12] Out: Physical address [39:12] + 2 page level bits

Instruction Fetch from L2 cache and Branch Prediction

Front Side Bus Inter-face, 533..800 MHz

Trace Cache
Fill Buffers

Micro code Sequencer
Micro code Flash & ROM

16k uOps
128 kByte

8 way set associative

8 x 512 sets of 4 uOps

La Grande? uControler, RAM/ROM

Tag comparators
39 bit virtual Tags

LRU
LRU

Misc. Tag Data

uOp Queue
Register Alias History Tables (4x128)
Register Alias Tables

Instruction Queue (for less critical fields of the uOps)

General Instruction Address Queue & Memory Instruction Address Queue (queues register entries and latency fields of the uOps for scheduling)

uOp Schedulers

Parallel (Matrix) Scheduler for the two double pumped ALU's

General Floating Point and Slow Integer Scheduler: (8x8 dependency matrix)

FP Move Scheduler: (8x8 dependency matrix)

Load / Store Linear Address Collision History Table

Load / Store uOp Scheduler (8x8 dependency matrix)

FP, MMX, SSE1..3

Floating Point, MMX, SSE1..3 Renamed Register File 256 entries of 128 bit.

Floating Point Registers

Legacy Floating Point Multiply

Legacy Floating Pnt.Add

512 kByte L2 Cache Block

L2 Phys. Tags

L2 Cache Line Transfer Buffers

L2 Phys. Tags

512 kByte L2 Cache Block

Integer Execution Core

(1) uOp Dispatch unit & Replay Buffer Dispatches up to 6 uOps / cycle

(2) Integer Renamed Register File 256 entries of 32 bit (+ 6 status flags) 12 read ports and six write ports

(3) Databus switch & Bypasses to and from the Integer Register File.

(4) Flags, Write Back

(5) Double Pumped ALU 0

(6) Double Pumped ALU 1

(7) Load Address Generator Unit

(8) Store Address Generator Unit

(9) Load Buffer (96 entries)

(10) Store Buffer (48 entries)

(13) Databus multiplexing

(14) Cache Line Read / Write Transferbuffers and 256 bit wide bus to and from L2 cache

(11) ROB Reorder Buffer 4x64 entries

(12) 16 kByte Level 1 Data cache four way set associative. 1R/1W
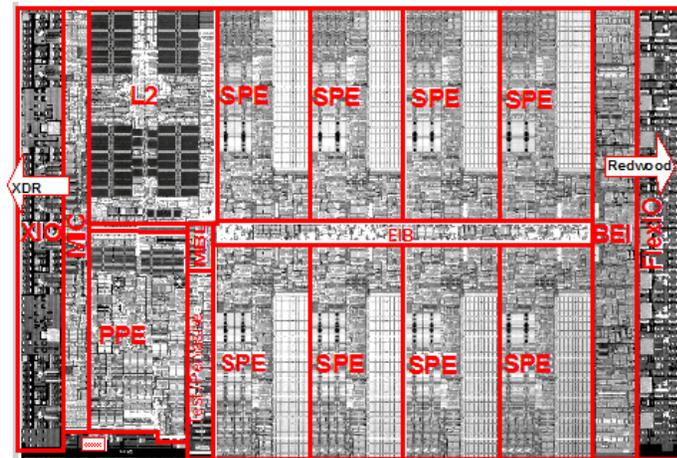
Dual-Core Intel Itanium 2 processor (Montecito)

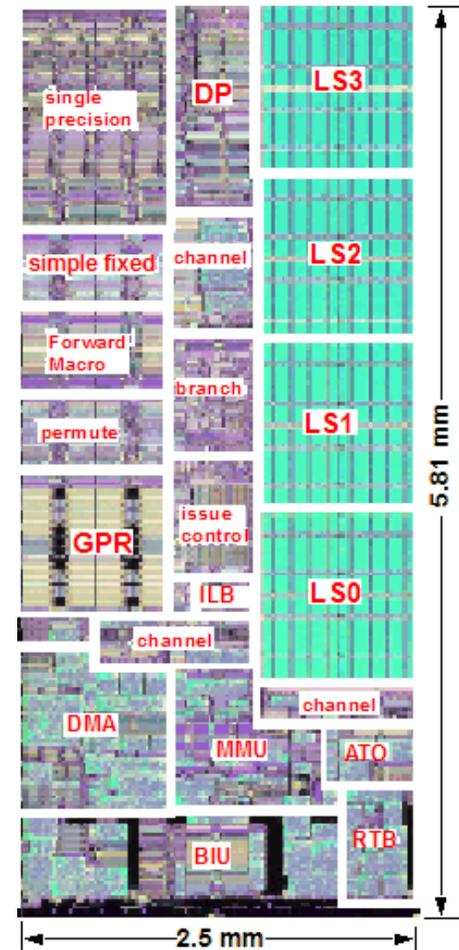**1.72 billion transistors, 27.72 mm x 21.5 mm**
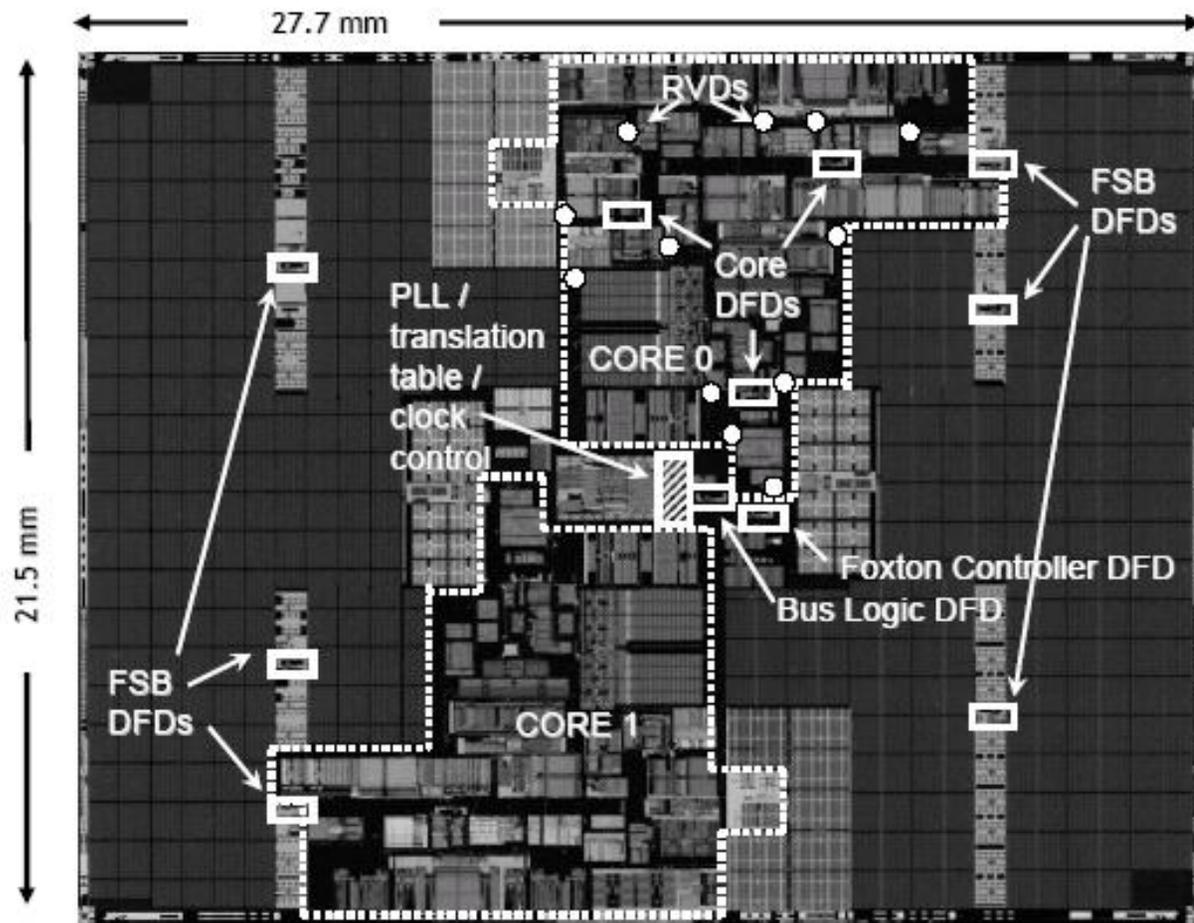
PlayStation 3

Cell Processor (ISSCC'05)

- PowerPC Processing Element (PPE), Synergistic Processing Element (SPE), Internal Element Interconnect Bus(EIB), Shared Memory Interface Controller (MIC), Double precision FPU (DP)m

- 90nm, 4GHz, 256 GFLOPS (single), 25 GLOPS (double), 25 GB/s memory bandwidth (RAMBUS), 50-80W

- Trend is to augment processor with parallel functional units to improve efficiency

IBM/Sony/Toshiba Cell Processor

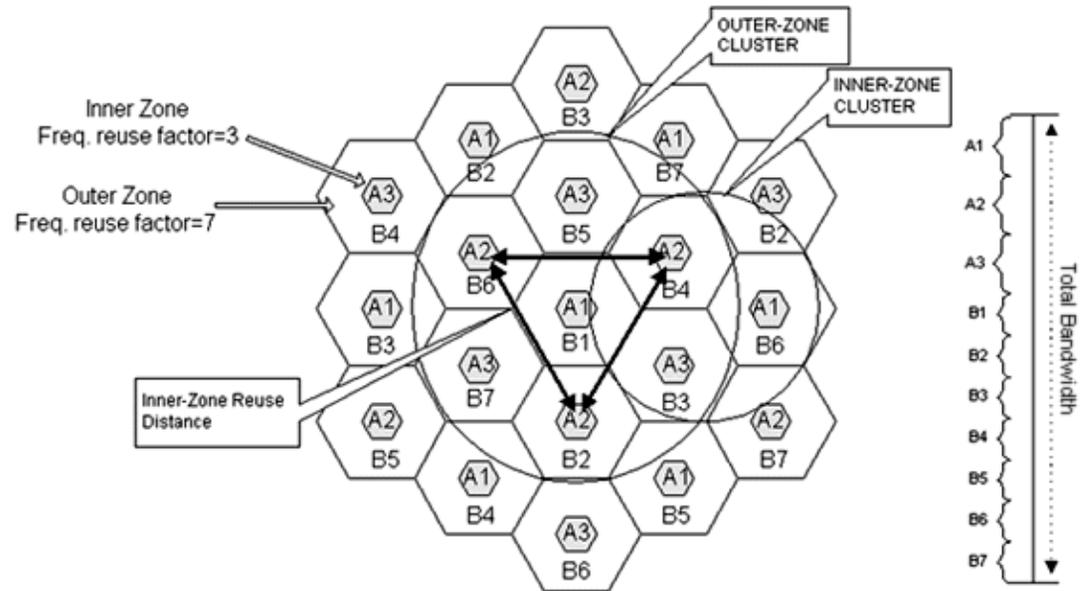Synergistic Processing Element: 4 FMADD/cycle

Source: http://www.realworldtech.com/includes/templates/articles.cfm?ArticleID=RWT021005084318&mode=print

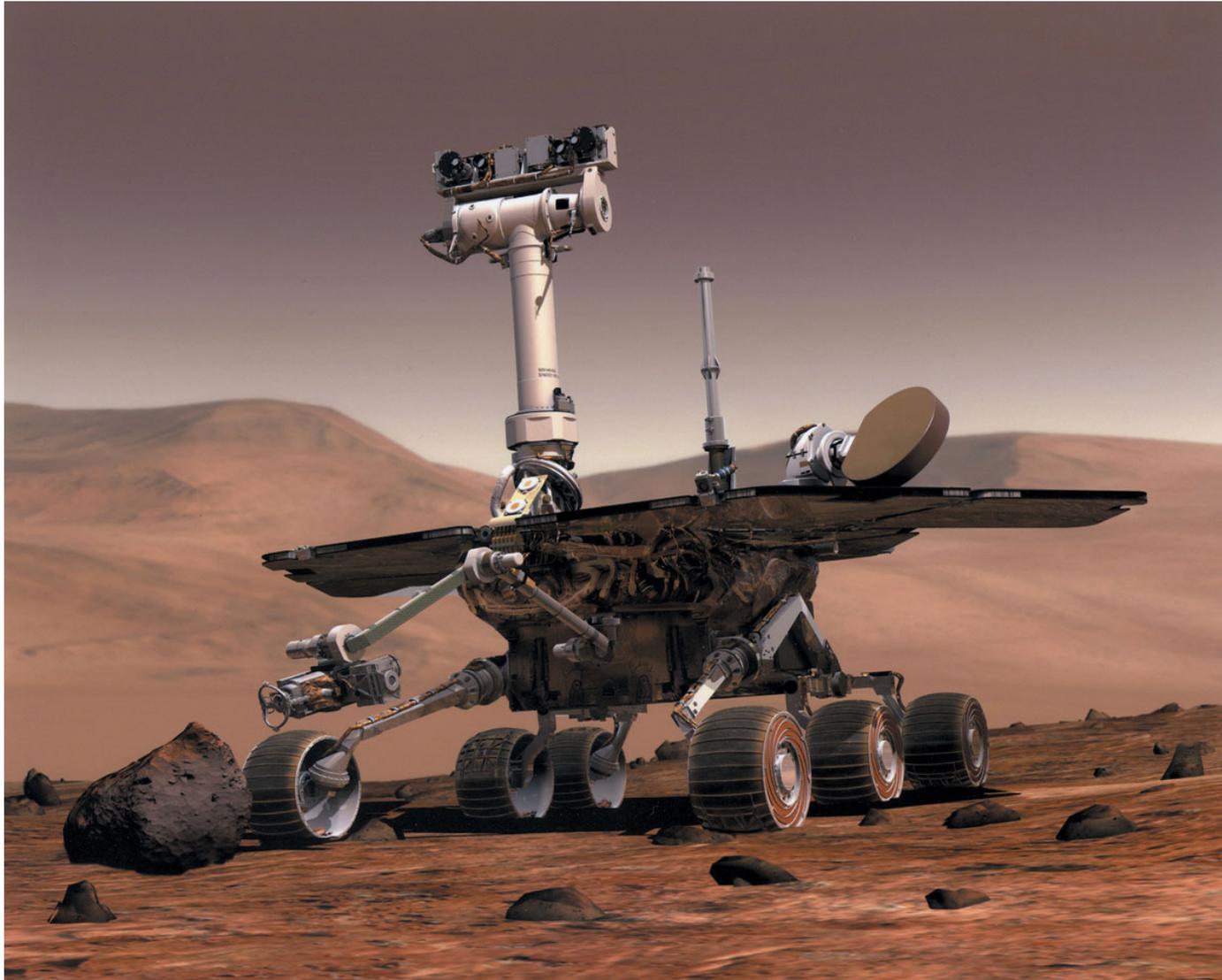# Cellular Applications
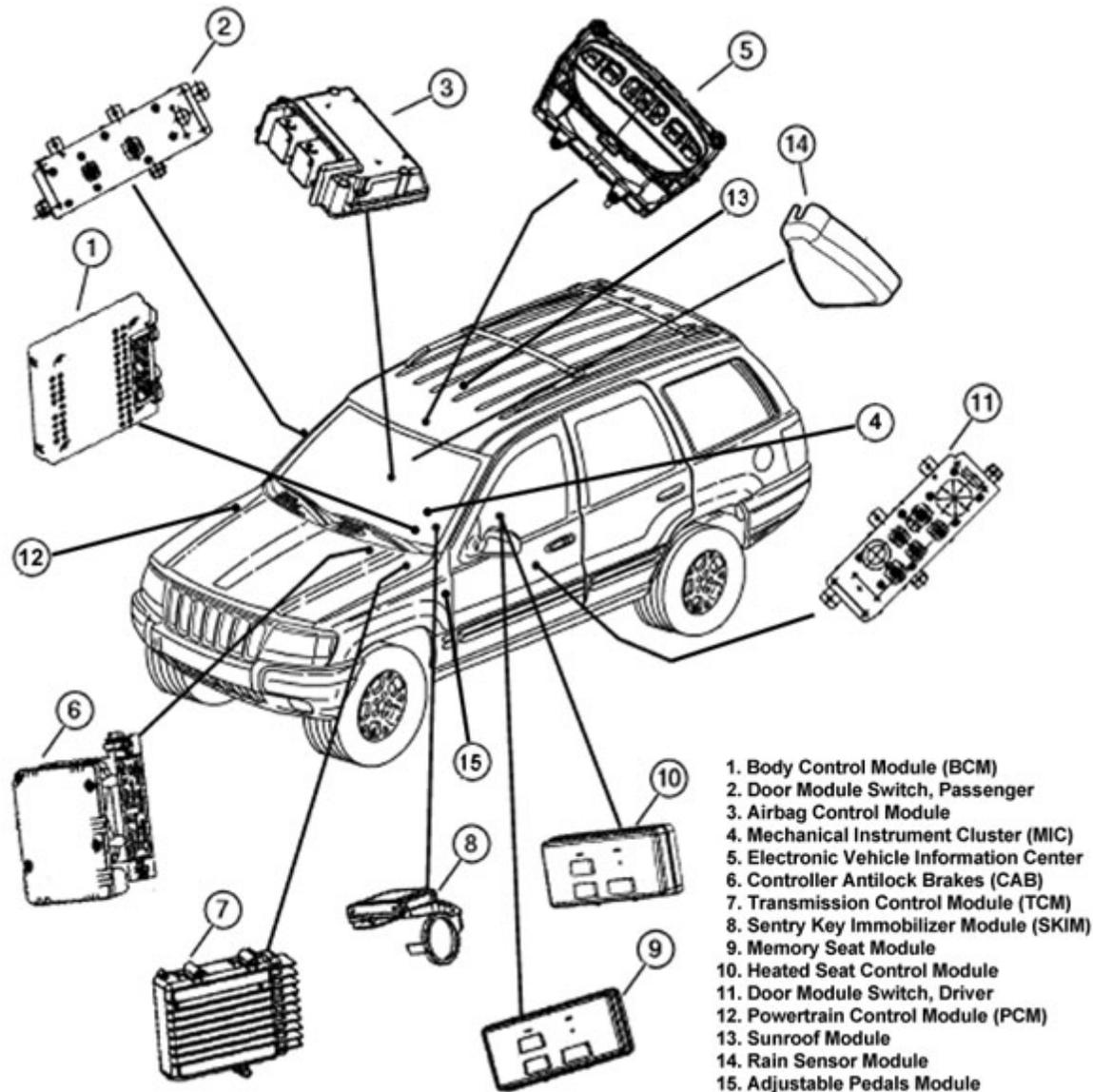
Cellular Applications

Cellphone Networks

&

Cellphones

Outer-Space

Mars Rover

On the road

Automotive Electronics

1. Body Control Module (BCM)
2. Door Module Switch, Passenger
3. Airbag Control Module
4. Mechanical Instrument Cluster (MIC)
5. Electronic Vehicle Information Center
6. Controller Antilock Brakes (CAB)
7. Transmission Control Module (TCM)
8. Sentry Key Immobilizer Module (SKIM)
9. Memory Seat Module
10. Heated Seat Control Module
11. Door Module Switch, Driver
12. Powertrain Control Module (PCM)
13. Sunroof Module
14. Rain Sensor Module
15. Adjustable Pedals Module

7-Series BMW:
  63 Embedded
    Processors

Mercedes S-Class
  65 Embedded
    Processors

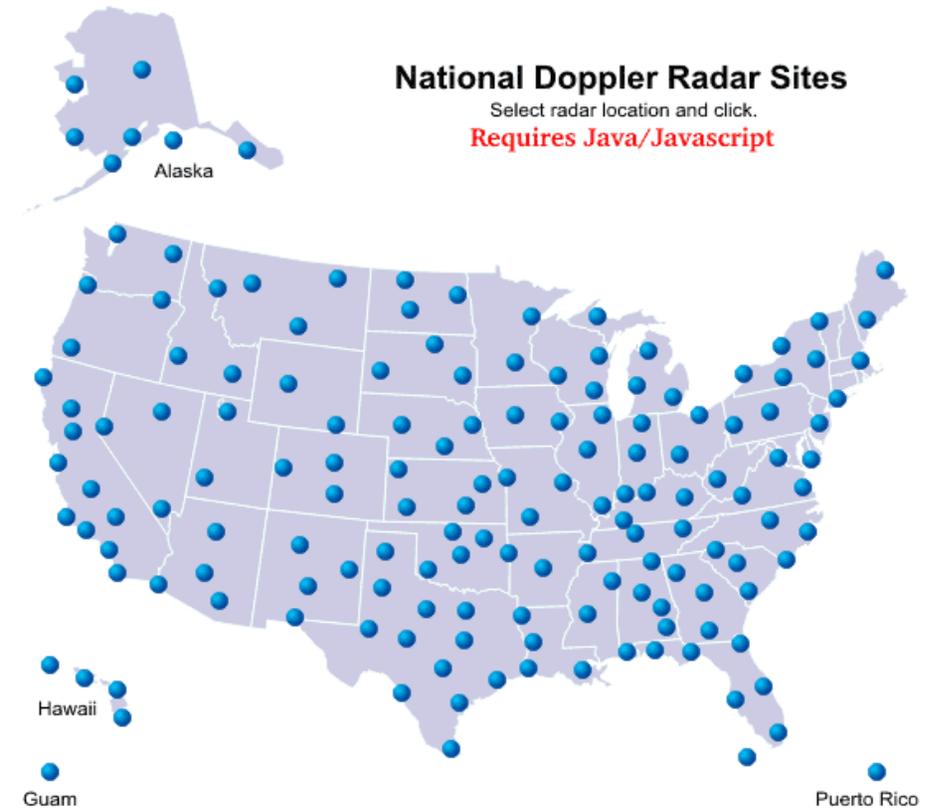More than 80% of the innovation in autos is from innovations in electronics
- Daimler-Chrysler

Automotive Semiconductor Market: US$13.1 billion / year

19

Predicting the weather
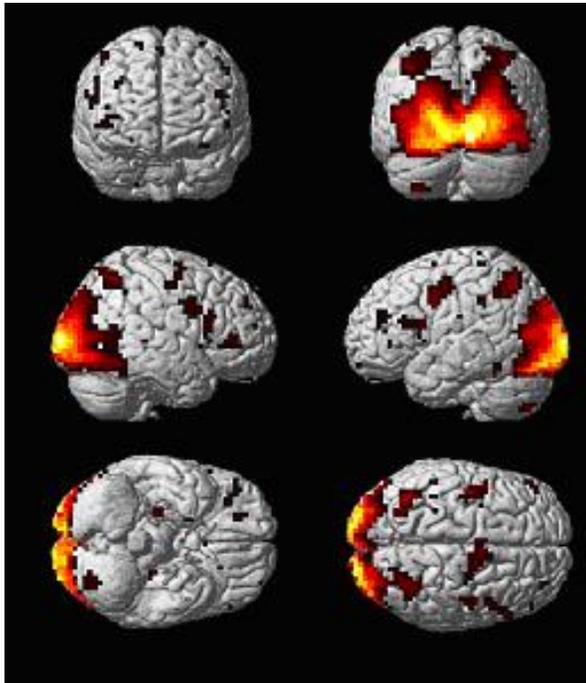
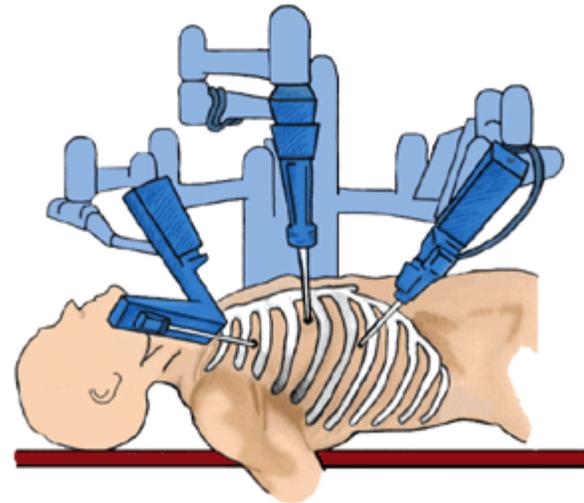# Meteorological Applications

Doppler Weather Radar

Inside you

Biomedical Applications

Medical Applications

Diagnostic Applications

To build any of these types of systems, you need specialized hardware

In this course, we'll be looking at how to approach hardware design…

What should you know already?

- Logic Gates, Combinational Blocks, and Storage

  - Gates, multiplexers, decoders, encoders, registers, flipflops

    - We'll do a ***real*** crash course review

- Synchronous Logic

  - Counters, State Machine Fundamentals, etc

- Fundamental computing concepts

  - We won't be delving deeply into processor architecture, but you should know the appropriate terminology

What should you know already?

- VHDL

  - We'll do a review focusing on ***synthesizable*** HDL versus HDL for testbeds

- Numbers and Arithmetic

  - Base 2 and 16, floating point/fixed point, etc.

- Combinational logic fundamentals

  - Logic equation evaluation, Karnaugh maps, masking, etc.

Why take this course?

- It's "*required*" – a core course for the computer and electronics options

Why take this course?

- It's **_cool_** !!

- Strong ties between the lectures (theory) and the labs (application)

- Develop practical hardware design skills for digital circuit development

  – No "hack" jobs please

  – Methodical and meticulous guarantees long term success

Why take this course?

- Competency in hardware design opens up another possible area for employment:

  "… the computer systems design industry is expected to lead **_all_** service industries in terms of output and employment growth."

  – From: *Looking-Ahead: A 10-Year Outlook for the Canadian Labour Market (2006-2015)*, released by Human Resources and Social Development Canada

What will you learn?

- Hardware Description Languages and Simulation (possibly synthesis)
  - i.e. How to design with HDL.

- Field Programmable Gate Array architecture and technology (Altera, Xilinx)

- Advanced state machine concepts

- Digital design techniques

What will you learn?

- An introduction to floating point arithmetic architecture

- Digital signal processing algorithm implementation

- Design for high performance digital systems

- Computer bus interfacing

By the end of this course, you should be able to take a circuit specification from your project manager:

-divide the specification into appropriate modules

-create a set of testing criteria and testbenches

-utilize some of the basic techniques for circuit optimization

You should also be able to create any necessary mathematical processing units and bus interfacing

About Me

- Microelectronics Group: Computer Engineering Option

- Started in fall '06

  – I've taught ensc '350, '351, '460 ('452 as of last fall)

- My research group is the Reconfigurable Computing Lab

- My interests include:

  – Reconfigurable Computing, Application-Specific Architectures, Design Methodologies, FPGAs, Embedded Systems, Computing System Design

Reconfigurable Computing Lab

Logistics

My Contact Info

- Office:
  - ASB8819
- Email:
  - lshannon@ensc.sfu.ca
- Homepage:
  - http://www.ensc.sfu.ca/~lshannon/
- Office Hours:
  - Mondays 10:30am - 12:30pm
- Teaching Assistants
  - David Dickin
    - Office Hours: Tuesday and Thursday 2-3pm

Lectures

- Monday 8:30-10:30am AQ4150
- Wednesday 9:30-10:30am AQ4150
- 2 in-class pop quizzes (15%)
  - You will receive 3 – 10 days notice
- 1 in-class Midterm (20%)
  - Date: Monday, February 23rd
- 1 Final Exam (35%)
  - Wednesday, April 15th, 2009 8:30-11:30am

Labs

- ## Lab Projects (30%)
    - ### Teams of **2**
    - ### First start off with the tutorials
        - For the tools (complete by January 28th)
    - ### Multiple projects/assignments covering:
        - State Machines, Datapath and Controlpath design, and Testbenches
    - ### All projects will be demonstrated for the TA
    - ### Larger project(s) may include a report
    - ### More details in the coming weeks

Lab Equipment

- Each pair will be given an Altera DE-2 70 boards to sign out for the term
  - First need to form a lab group of 2
  - Sign up sheet on the sign up sheet posted on the RCL lab door (ASB 8803.1)
  - Don't forget to fill out the appropriate paperwork with Fred

- ***You are responsible for the well being and proper care of your FPGA board and the equipment in the lab***
  - ***Destruction/vandalism will result in SEVERE grade penalties***

Course Webpage

- Course Webpage:
  - http://www.ensc.sfu.ca/~lshannon/courses/ensc350/.
- Course Handouts will be posted here.
- Lecture Slides will also be posted before each class.
- We will be using WebCT for bulletin board postings only
  - Do **_NOT_** send technical email to myself or the TAs
- A Course Mailing List has been created for me to send any last minute updates (ENSC350)
  - Students will **_not_** be allowed to email to the list to prevent spam
    - Use the Bulletin Board to communicate with each other

Course Bulletin Board

- All students **must** regularly check the course bulletin board.

- All course related questions **must** be posted to the bulletin board.

- You should also post any answers you know

- Postings are set to be anonymous (you can sign your name if you like).

- Announcements and hints regarding the labs and project will be posted here.

- Please observe appropriate bulletin board etiquette (**be respectful**).

- Good citizenship in this course is expected and will be rewarded.

Course Bulletin Board

- The different Bulletin Board Categories are:
    - Course Work Related Topics
        - Lectures
        - Administrative details
        - Lab projects and tutorials

    - Uncategorized Topics
        - Default Topic
        - SFU Cafe

Course Reference Material

- ## Textbook:

   **A Custom Courseware copy of Douglas J. Smith's *HDL Chip Design: A Pracitical Guide for Designing, Synthesizing & Simulating ASICs and FPGAs Using VHDL or Verilog* is recommended for this course.

- ## Check the course web page for online VHDL and Verilog references

   **This is a new reference book I'm suggesting. I will appreciate any feedback people have regarding this book. For previous offerings, the course textbook has been Sundar Rajan's Essential VHDL: RTL Synthesis Done Right, 2nd edition S & G publishing (this book is still supposed to be available from SFU bookstore).

Course Grade Breakdown

- 30 % Lab Component

- 15 % Two Pop Quizzes

- 20 % Midterm

- 35 % Final Exam

-   5 % Class Participation ***Bonus***

Class Participation *Bonus*

- ## 5 % Class Participation:

  - This course requires team work, and it will be rewarded:

    - Making and responding to posts on the course bulletin board

    - In-lecture participation

    - Helpng "the team" (fellow classmates) in the lab

IMPORTANT NOTES

Penalties for Copying and Plagarism

While **I** promote team work and helping each other, code copying and plagiarism will **_NOT_** be tolerated. Any individual found copying code from other class members or the web will receive an automatic **_0_** on the **_entire_** lab component. Similarly, anyone found copying during quizzes, midterms, or the final will receive a **_0_** for the **_entire_** testing component. Furthermore: **_Any_** code copying or plagiarism will result in an automatic **_0_** for the class participation bonus

NO RECORDING OF ANY KIND!!

The lectures and displays (and all material) delivered or provided in this course (ensc350) by Professor Shannon, including any visual or audio recording thereof, are subject to copyright owned by Professor Shannon. ***It is prohibited to record or copy by any means, in any format, openly or surreptitiously, in whole or in part, in the absence of express written permission from Professor Shannon any of the lectures, materials provided or published in any form during or from this course.***

This copyright ***also applies to any guest lecturers*** invited to the class.

Missed Evaluations

Assuming that an evaluation is missed for medical reasons that are verified by a Doctor's note, the following rules apply:

-The value of one missed pop quiz will be counted towards the attended pop quiz.

-The value of a missed midterm is counted towards the final exam. Please note that the final exam will be harder than the midterm.

-If a person misses the final exam and **_if_** a makeup exam is deemed necessary, I will err on the side of making this exam **_harder_** to make it fair to the rest of the class.

-As you will have multiple weeks for your lab projects, there will be **_no extentions_** for project completion or reports. This means you should plan ahead and start early (not cram at the last minute). Only the missed lab **_demo_** is excused by the note.

Grade Award Breakdown

- \> = 90%   A+
- 85 – 89% A
- 80 – 84% A-
- 75 – 79% B+
- 70 – 74% B
- 65 – 69% B-
- 60 – 64% C+
- 55 – 59% C
- 50 – 54% D
- < 50%   F

Final comment on grades:

- I do **_not_** scale:

  - Pro: If everyone does well, everyone gets a good mark

  - But there's always the flip side …

Class Surveys

- This is only the second offering of this course with the new lab equipment. I will be looking for feedback on the labs and how to make this better.
  - There are quite likely still kinks to be worked out
  - For this, I really would appreciate your honest input
  - I'm interested to find out what you think of the
    - Course Material (pace and depth of presentation)
    - Visual Aids and Presentation Style
    - In-Lab and Lab Projects Experience
      - What can I do to make it better?
  - I'll post class surveys at a later date.

Something to think about: Why/What would you like to learn about digital system design?