CSE585 Handout 1/12/00

Reading for Next Week

The hardware platform

XESS board is a contains a large XILINX FPGA (programmable logic device) that can be used to implement and test digital designs. The board includes a microcontroller that we won't be using. These documents will be more useful to you when you are in the lab than they are just to study alone.
http://www.xess.com/FPGA/prgmdl40-v1_2.pdf
http://www.cs.washington.edu/education/courses/477/99sp/docs/xsschpwb.pdf

Also check out the CSE567 web-site for Spring '99 with Carl Ebeling. Follow the "project info" link
http://www.cs.washington.edu/education/courses/567/CurrentQtr/project/info.html#framework
We will be using the same hardware setup and CMOS video processing lab project later.

Verilog

Verilog is a very flexible language. The flexibility is there to make it easy to use, but it is also possible to make things more difficult than they need to be. I recommend that you explore some of the links on this web-page. http://www.cs.washington.edu/education/courses/370/99au/tools/verilog.html
Especially the Handbook on Verilog HDL by Daniel Hyde of Bucknell. I Understand that Jeffery Hightower's overview of the use of Verilog for synchronous sequential logic is quite good.

Assignment 2 Due at 1:00pm 1/19 (before class next week).

Implement BITS from week-1 in Verilog twice: once as a purely combinational system (hint: there should be no clock) and once as a synchronous sequential system (with a clock) as we did in class. Take your best crack at this, and don't worry if you are completely lost. We will catch you up in lab next week. E-mail your Verilog code to me by Wednesday 1/19 at 1:00pm. I will pick a couple of the best ones to show in class. You don't need to simulate, verify or debug your code, that is what we will be doing in lab. Just send me the text in e-mail. We will review some your best efforts in lecture next week before going into the lab. Your Verilog code should not contain any simulation directives ($) such as $finish, $monitor, etc., and it should not contain any explicit timing delays (#5 for example).

Week 3 Preview

We will implement BITS in lab next week using the complete detailed synthesis, simulation, and timing design flow in the Xilinx Foundation tools. We will provide you with a lab guide next week so don't worry about studying the tools in advance. You will figure it out in lab with your lab-mates. For those who get done quickly, we will provide some additional challenges!

Project Ideas

- Feasibility Study for a Product Concept
  - Architecture Level Cost/Performance Trade-offs.
  - Major HW and SW components.
  - Rough market potential
  - Likely competition, threats.
  - Implementation method (ASIC, FPGA, Standard IC's on PCB)

- Design Project
  - Prototype a system. Palm Pilot based, FPGA Based, etc.

- Case Study of an Existing Product
  - Review the architecture of existing interesting systems.
    - Sega Dreamcast machine
    - RIO Diamond
    - Palm VII
    - The Sony Dog
    - iMac
    - Or any other system that you might be interested in a can get data for

- Interesting Questions
  - Options for adding TCP-IP to embedded applications (existing or imagined)
  - Smallest system that can run a JVM, or operating systems such as Linux/WinCE

- Emerging CAD Technologies (Research Report)
  - Formal verification
  - Static timing analysis
  - Synthesis
  - Physical Design/Logic Synthesis Integration
  - HW/SW Codesign/Coverification

- Intellectual Property Market and Technologies
  - VSIA IP interchange standards (deliverables, contracts, etc)
  - Catalog of existing IP, future trends.