Project Guidelines

Project Schedule

09/30/2010 Submit team members using VSD_ProjMemForm.doc
11/25/2010 Present project proposal
01/20/2010 Complete Final project report due 9am
01/22/2010 Demonstrate final project 10am-6pm

Project Description

In this term project, you will work as a team (with 4 ~ 7 members) to design a general embedded processor that can work with a compiler and communicate with external modules (either memory or accelerators for a specific application.) Requirements are described as bellowed:

1. Basic requirements (60%)

- **Must be a pipelined structure and Andes (or equivalent processor) compatible.**
- **Must be compatible with a compiler and a debugger**
- **Data width shall fit the specification of Andes, i.e., 32 bit.**
- **Operating speed is at least 50 MHz for post-synthesized version.**
- **Its instruction set shall have at least 30 instructions, including branch and I/O instructions.**
- **There shall be at least 6 addressing modes.**
- **The address space for I/O controller registers is part of the same address space as data memory address, i.e., I/O registers are memory mapped.**
- **External main memory shall use non-ideal latency model, including data memory and instruction memory. The delay shall be set to at least 4 times of your processor clock, or at least 80ns. **Note that memory does not need to be synthesizable.**

- **Include interrupt mechanism and interrupt service routine for handling requests from other devices, such as sending data and control signals to DMA controller, or receiving data and control signals from DMA controller.**
- **Gate count shall be larger than 20,000.**
- **Must verify every instruction individually**
- **Must link to Andes compiler without problem for straight code**
- **Must verify by sorting 100 numbers stored in external memory or any equivalent test bench**
- **Must verify by perform Fibonacci series from F_0 to F_{100} stored in external memory**
- **Must verify interrupt and interrupt service routine by receiving data from external devices**
- **Must verify by running a sequence of instructions (at least 1000) that completed a meaningful function based on the target application. If your design is for a general processor, a meaningful function is like add up numbers from 1 to 20 and store back to external memory. If you target 3D-graph application, a meaningful function would be to compute pixel position for a cube when rotating 60 degree horizontally.**

2. Advanced requirements (40%)

- **Include at least 20 more instructions other than those in basic requirements**
- **Implement stack or other mechanisms to facilitate function calls or recursive function.**
- **Implement and verify Cache (either 1 level or 2 levels), Forwarding, or DMA, AHB bus access capability etc. (Note that AHB could be just behavioral model, i.e., not need to synthesize)**
- **Finish layout using SoC Encounter or IC compiler (Originally Astro)**
- **Perform post-layout simulation and show all pre-layout work correctly**
- **Complete code analysis by nLint and reach 100%**
- **Others like O.S. bootable**

NOTE:

- **The course will provide a technology file for simulation and synthesis later.**
- **For those want to do chip layout, your team shall need to access TSMC 0.18um technology in your own lab since we are not allowed to provide this technology for this course.**