CS520: APEX CPU Simulator v2.0

Gaurav Kothari
<gkothar1@binghamton.edu>

Department of Computer Science
State University of New York at Binghamton

November 2, 2020

Overview

- A template for a working 5-Stage APEX In-order Pipeline
- Implementation in `C` language
- You can read, modify and build upon given code-base
- You are also free to write your own implementation from scratch

Micro-architecture

- Stages: *Fetch* $\rightarrow$ *Decode* $\rightarrow$ *Execute* $\rightarrow$ *Memory* $\rightarrow$ *Writeback*
- All the stages have latency of one clock cycle
- Includes logic for `ADD`, `LOAD`, `BZ`, `BNZ` and `HALT` instructions
- On fetching `HALT` instruction, *fetch* stage stop fetching new instructions
- When `HALT` instruction is in *commit* stage, simulation stops
- Logic to check data dependencies not included

Files

- `Makefile`
- `file_parser.c` → Functions to parse input file, add new instructions
- `apex_cpu.h` → Data structures declarations, model of CPU, Pipeline stages, code and data memory
- `apex_cpu.c` → Implementation of APEX cpu
- `apex_macros.h` → Macros used in the implementation
- `main.c` → Main function which calls APEX CPU interface
- `input.asm` → Sample input file

How to compile and run ?

- Go to terminal, `cd` into project directory and type:

  `$ make`
- To run:

  `$ ./apex_sim <input_file_name>`
- Report bugs at: `gkothar1@binghamton.edu`