

ECE586_Team3

Team 3's final project for ECE 486/586 Portland State University, Winter 2022

Braden Harwood, Stephen Short, Michael Weston, Drew Seidel

**Information sourced from Dr. Faust's Final Project Description and Group Discussion*

Table of Contents

- Objective
- Directory Organization
- Getting Started
- Usage
- Test Plan

Objective

The purpose of this project is to simulate a 32-bit RISC-V machine. The program takes in a memory image file and alters the PC, registers, and memory based on instructions. We implemented all of the standard instructions, the RV32M extension package, ecall system calls, and a user interface command line option.

Directory Organization

- docs
 - Contains documentation relating to the project including our specifications document, edge cases tracking, test plan, and table of timing constraints.
- src
 - Contains all of our source files including the main program.
- testing
 - The test plan includes testing every function, including bound cases, overflow, etc. See testplan.pdf for detail.

Getting Started

1. Clone the repository to your computer

```
git clone https://github.com/your_username_/ECE586_Team3.git
```

2. Run make to compile the project

```
make
```

3. For a basic simulation, run the following:

```
./sim.exe
```

By default, program reads from 'program.mem' in directory as input file. By default program counter is 0, stack pointer is 65535. Output will be generated and sent to stdout

For more advanced use, see the Usage section for different flags that can be set during runtime. A few examples:

- Will set the start program counter to 4, and the stack pointer to 4096.

```
./sim.exe -pc 4 -sp 4096
```

- Will read the file 'prog1.mem' into memory. Will set the start program counter to 4, and the stack pointer to 4096.

```
./sim.exe prog1.mem -pc 4 -sp 4096
```

- Will do all of the previous, with the user interface enabled. The user interface allows stepping, setting/clear breakpoints, watchpoints, printing register and memory contents etc.

```
./sim.exe prog1.mem -pc 4 -sp 4096 -ui
```

Usage

sim.exe

The simulation program sim.exe has the following flags: -pc to set pc (defaults to 0). -sp to set stack pointer (defaults to 65536). -ui to enable user interface.

make

make also has additional targets:

- <none> : No target provided defaults to the basic compilation of all
- all : Compiles the simulator sim.exe. Running this mode will print register contents at the end of execution.
- verbose : Compiles the simulator sim.exe, but prints register contents after each instruction.
- debug : Compiles the simulator sim.exe and adds additional decoding text to the output for debugging purposes.
- reg_mem : Compiles the simulator sim.exe and prints debug information, along with register and memory contents after each instruction. If a memory address(s) or register was just written to, they will be highlighted in red, for easy of debugging.
- clean : Removes all object files and the executable generated by make to clean the directory.

Test Plan

Go through the most recent RISC-V specifications document and test each implemented function according to its particular specifications.