ECEC-355

Project 1: RISC-V ISA

Instructor: Anup Das
TA: Ankita Paul {ankita.paul@drexel.edu}

1 Introduction

This project is intended to be a comprehensive introduction to RISC-V ISA. There are two parts to this project:

- In part one, you will translate certain types of RISC-V instructions into their binary representations by hand.
- In part two, you will automate the binary representation translation process using C programming language.

You may work on this project in a team of up to three members. This project is due on **Jul 5, 2021**.

2 Required Reading

Chapter 2. Instructions: Language of the Computer, Sections 2.1 – 2.10, Sections 2.12 – 2.14.

3 Development Environment

- Operating System: Linux.
- Code-base: `https://github.com/Shihao-Song/DREXEL-DISCO-RISC-V-Simulator-C-Impl`. Please *git clone* the repository to your Linux machine:

```
$ git clone https://github.com/Shihao-Song/DREXEL-DISCO-RISC-V-Simulator-C-Impl
```

4 Framework Overview

1. You will be working under directory *DREXEL-DISCO-RISC-V-Simulator-C-Impl/project_1*. To navigate to the directory:

```
$ cd DREXEL-DISCO-RISC-V-Simulator-C-Impl/project_1
```

2. To compile and run the simulator:

```
$ make
$ ./RVSim ../cpu_traces/project_one
```

3. You should be able to see the following output:

```
Loading trace file: ../cpu_traces/project_one

Instruction at PC: 0
0 0 0 0 0 0 0 1 1 0 0 1 0 1 0 1 0 0 0 0 0 1 0 1 0 0 1 1 0 0 1 1
```

4. *../cpu_traces/project_one* contains five representative RISC-V instructions. To take a look at the format of the trace file:

```
$ cat ../cpu_traces/project_one
add x10, x10, x25
ld x9, 0(x10)
addi x22, x22, 1
slli x11, x22, 3
bne x8, x24, -4
```

5 Binary Representation of RISC-V Instructions

Your first task is to translate the instructions inside *../cpu_traces/project_one* into their binary representations by hand. For each instruction, please report the following

1. The type of instruction, i.e., R-type.
2. The binary representation.

6 Extending the Framework

Your next task is to extend *Parser.{h,c}* to support all types of instructions in *../cpu_traces/project_one*. A routine that translates R-type instructions has been given. Compare the output of *RVSim* with *Sec.5* to verify the correctness.

7 Submission

1. Summarize your experiment in Section 5 and 6. Compile your report in PDF format.
2. All the source codes.
3. Zip above and submit through Bblearn.