SYA14 - Neuromorphic Computing

Lab 1

1 Objective

In this lab, you will learn to be familiar with Spiking Neural Network, Verilog HDL, and CAD tools.

2 Prerequisite

The following are the prerequisites of this exercise:

- Linux and command line tool.
- Verilog HDL.
- Modelsim with command line (the example program doesn't support GUI).
- Familiar with CAD tool to export: area cost, timing and power consumption.
- Fully connected model.
- MNIST dataset.

3 Spiking Neural Network Design

The source code is written in Verilog HDL. Simulation scripts using Modelsim are provided.

The SNN model is fully connected (784-48-10) for the MNIST dataset. The weights are pre-trained and converted to an 8-bit fixed point format. The structure of Verilog HDL is as follows:

- top.v: top-level file of SNN.
  - SNPC.v: neuromorphic core acts as a layer.
    - SNPC_control.v: controller of SNPC.
    - xbar.v: crossbar of synapses.
    - LIF_neuron.v: Leaky-Integrate-and-Fire neuron.

4 Exercise 1-1: Spiking Neural Network Design

Description: This design is about a simple spiking neural network (SNN) written in Verilog HDL using ModelSim.

4.1 Step 1

Download the source code from the website: `https://web-ext.u-aizu.ac.jp/misc/neuro-eng/book/NeuromorphicComputing/lab.html` (direct URL: `https://web-ext.u-aizu.ac.jp/misc/neuro-eng/book/NeuromorphicComputing/lab/SNN_RTL-main.zip`)

Unzip the source code into the folder and change the directory to it. The content of the source code is as in Figure 1.

Figure 1: The context of the source code.

- *env.sh*: environment bash file to run CAD tools. Please note this only works with the CAD server zxp007.
- *RTL*: folder of RTL code of SNN.
- *SIM*: simulation folder using Modelsim
- *TB*: folder of testbench

4.2 Step 2

Link tools' paths into the terminal environment using the following command.

```
source env.sh
```

The content of `env.sh` is as in Figure 2. Please note that this code only works with CAD server zxp007.

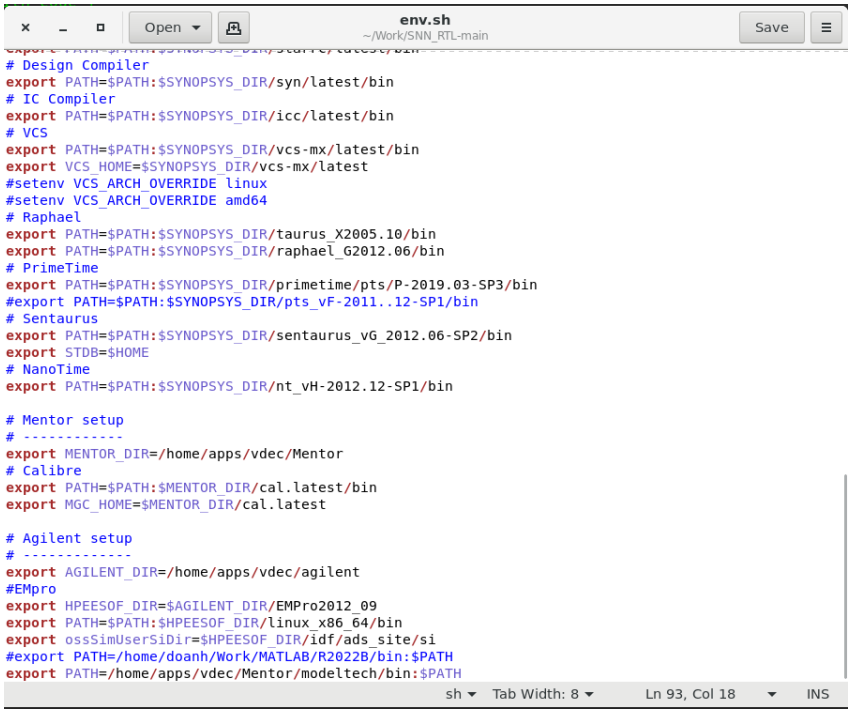

Figure 2: The context of the env.sh file.

4.3 Step 3

Go to the SIM (simulation) directory using the following command.

cd SIM

You can see the content of the simulation folder as in Figure 3.

Figure 3: The context of the SIM directory.

4.4 Step 4

In this step, we will run the simulation. To run in bash, we need executing permission (run only once in the new system):

```
chmod +x run.sh
chmod +x script/MNIST.sh
```

We can also observe the content of the bash script as in Figure 4

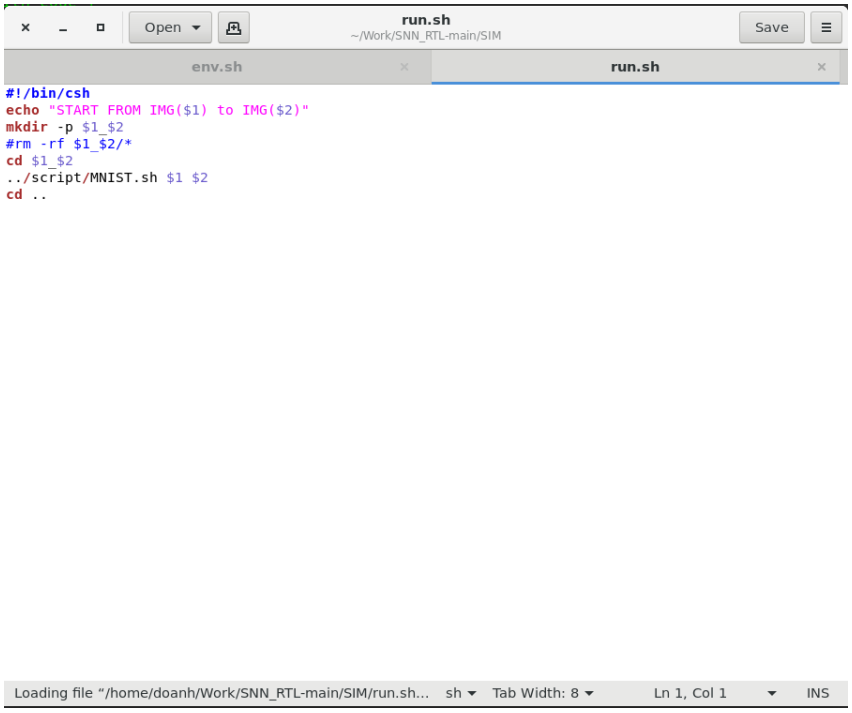

Figure 4: The context of the running script.

Assuming we want to test image index 1 to 10, the following command is used.

```
./run.sh 1 10
```

Following is the display of the terminal with the command:

4.5 Step 5

The output of the simulation can be found at `SIM/output/MNIST0-10K-1-10.txt`. Figure 6 is the output example.

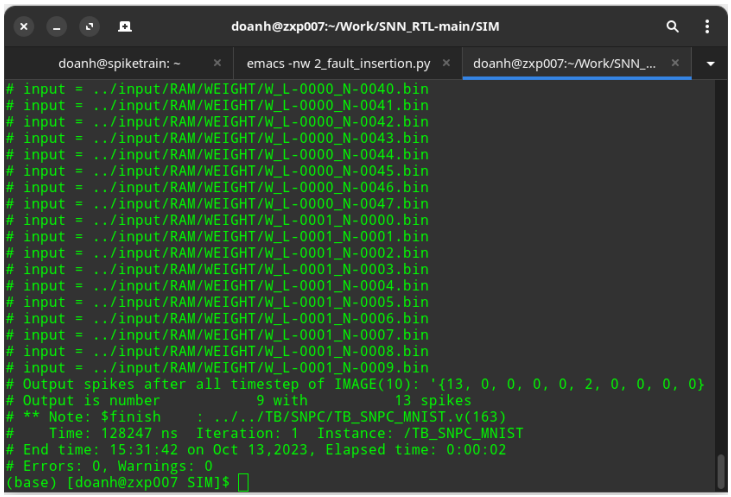

Figure 5: The output of the script.

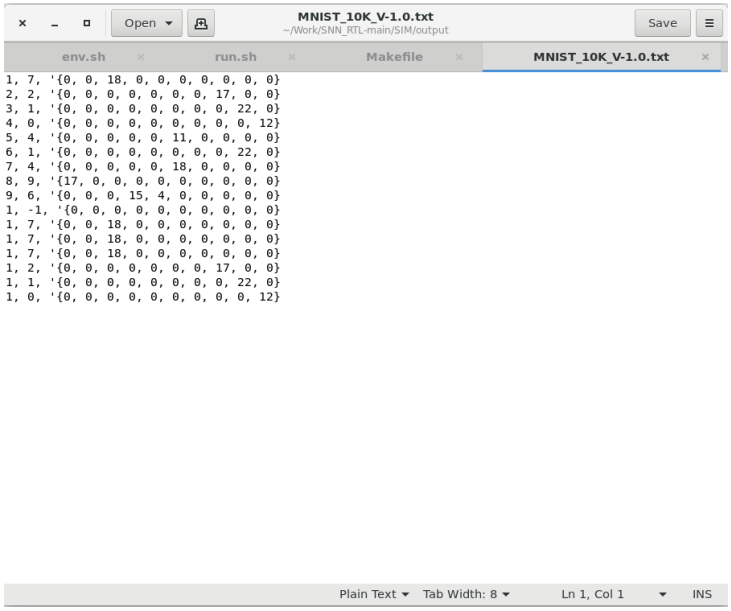

Figure 6: The example of the output file.

5 Exercise 1-2: Area, timing, and power report of the SNN module with FPGA tool

This exercise will utilize the FPGA tool to perform the area, timing, and power.

5.1 Step 1: Wrap the RTL to the FPGA project

In the first step, wrap the design of `top.v` as the top module and add other modules to the FPGA project.

5.2 Step 2: Synthesize the top module

In the second step, synthesize the top module.

5.3 Step 3: Read the report

In the third step, you need to report an evaluation of the area, timing, and power consumption of the SNN module.

6 Submission format and Deadline

Your report should be prepared in English and should contain the following:

1. Your name, your ID, and the Lab #.
2. All reports
3. Submission format: soft copy.

Note: This Laboratory is designed for the book [1]

[1] Book: Neuromorphic Computing Principles and Organization 1st, Edition, ISBN-10: 3030925242, ISBN-13: 978-3030925246, Publisher: Springer, May 2022.