

8-BIT SINGLE CYCLE PROCESSOR- RISC V

How to use the Programmer to load a program to the Simple Processor

1. Compile the "CO224Assembler.c" according to the instructions given in the C file and generate the executable named "CO224Assembler" (on Linux). Make sure you keep the generated executable in the current directory. You need to do this only once as long as the Assembler code is kept unchanged.
Command: gcc CO224Assembler.c -o CO224Assembler

2. Run the Shell script "generate_memory_image.sh" while passing the assembly program (your program.s file) as the only argument. There is a sample program (sample_program.s) under "programs" sub-directory for your reference.

```
Command: ./generate_memory_image.sh <assembly_file_name>
(e.g. ./generate_memory_image.sh programs/sample_program.s)
```

This script ultimately generates a file named "instr_mem.mem" which contains the machine code of your Assembly program formatted according to a standard file format (.mem) supported by Verilog to initialize a memory array.

3. In your testbench of the CPU of Simple Processor, add a line to read the generated memory content file and initialize the memory array you have declared in the testbench for instruction memory.

```
Verilog syntax to initialize a memory array using the generated .mem
file:
$readmemb("instr_mem.mem", instr_mem);
```

A sample testbench that includes this syntax to read the .mem file is given to you (cpu_tb.v). You may use it for the simulation of your Simple Processor.

(Note: As you may have already realized, once you have setup the files for the simulation by following above steps 1-3, you will only need to do step 2 every-time you need to load a new program to the CPU - given that the Assembler program and the location of instr_mem.mem file remains unchanged)