RISC-V ISA Simulator:

Design Documentation

Team members:

Bellam Jayanth - 2019CSB1080

Sagina Jaya Govinda Venkata Sai Teja – 2019CSB1114

Rohit Dagar – 2019EEB1186

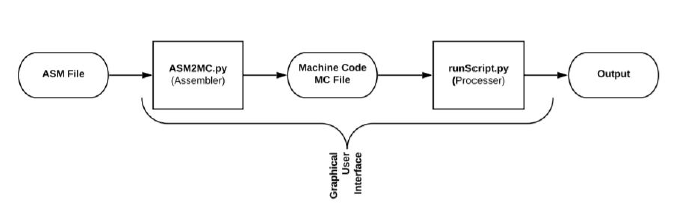
Satia – 2019CSB1118

Keshav Goyal – 2019CSB1165

**Features:**

* **Simulator backend:** Python
* **ISA:** RISC-V RV32IM
* **Size of instruction:** 32 bits
* **Number of registers:** 32
* **Size of registers:** 32 bits

**Overview:**



**Phase 1: Five step instruction execution**

## Input:

Input to the simulator is a machine code file that contains the encoded instruction and the corresponding address at which instruction is supposed to be stored, separated by space.

Functional behaviour:

All the instructions given in the input .mc file is executed as per the functional behaviour of the instructions. Each Instruction will go through the following steps:

* Step 1: Fetch
* Step 2: Decode
* Step 3: Execute
* Step 4: Memory Access
* Step 5: Register Update or Writeback

The simulator prints messages for each stage:

* Fetch: it prints that the instruction is fetched and PC is incremented.
* Decode: it recognize and print the operation, rs1, rs2, rd.
* Execute: executes the operation and print the instruction statement.
* Memory access: print the retrieved memory at the memory address (if required).
* Writeback: print the result updated in the destination register.