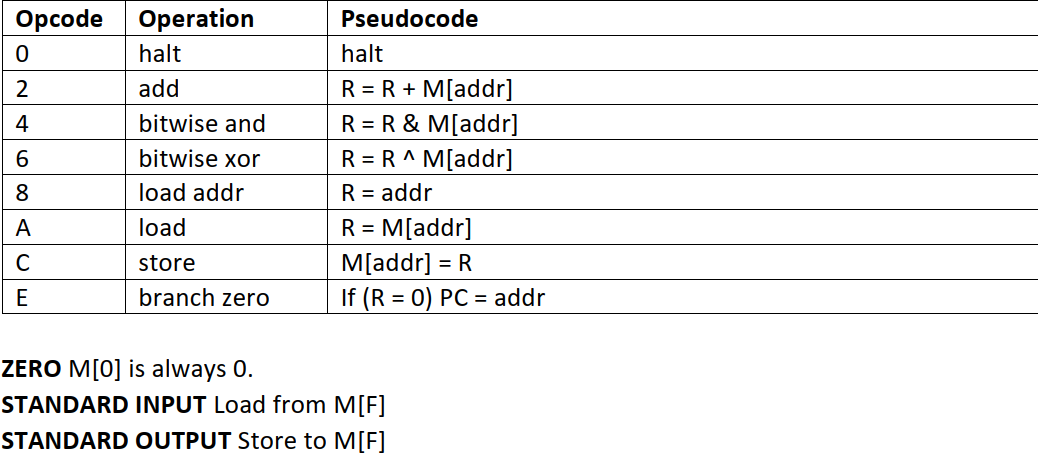
**Specifications:**

* TOY is an 8-bit CPU capable of simple arithmetic operations.
* It has an 8-bit ALU (Arithmetic Logic Unit), capable of Adding, Bitwise AND, and Bitwise XOR operations.
* The CPU has a central Control Unit, that triggers the selection lines for components.
* A clock at the heart monitors the operations of the CPU.
* The system has a 16-byte memory, consisting of 16 8-bit Memory Registers.
* A Program Counter keeps track of the operation.
* It has an Instruction Register and a Register that temporarily holds the data for the operations.
* The circuitry utilizes 2-way and 3-way MUXs to ensure connection between the components.
* The UI is built to allow data entry into specific memory locations, viewing the data stores, and allowing the CPU to function, while keeping track of the number of operations (PC out).

**Prerequisites:**

To run this program, the software Logisim needs to be [installed](https://sourceforge.net/projects/circuit/).

**Instruction Set:**



**How-to:**

* Open the .circ file on Logisim, and open the circuit titled: “TOY\_Bit”.
* In the Simulate tab, enable Auto-Tick, and ensure Tick Frequency is set to 2Hz (ideal).

**Loading into Memory:**

1. Ensure that Run is set to low value (off), or the Run button is not pressed. (This can be confirmed when the ON light is red)
2. Select the 4-bit address by interacting with the input pins.
3. Select the 8-bit data to be entered by interacting with the data input pins.
4. Hold the LOAD button to store this data into the address.
5. Repeat for all memory addresses.
6. To look at the data stored, input the address and hold the LOOK button, or look at the output pins at the lower left corner of the interface.

**Run the CPU:**

1. Turn off Auto-Tick, and ensure that clock is on low, or simply interact with it to make it so.
2. Set the data and address input pins to default, i.e 0.
3. Turn on Auto-Tick at 2Hz frequency.
4. Press and hold the RUN button to allow the CPU to run through the program stored.
5. While Running, the ON light will turn Green, and at the end of program (when memory is 0000 == HALT) the HALT light turns Green.