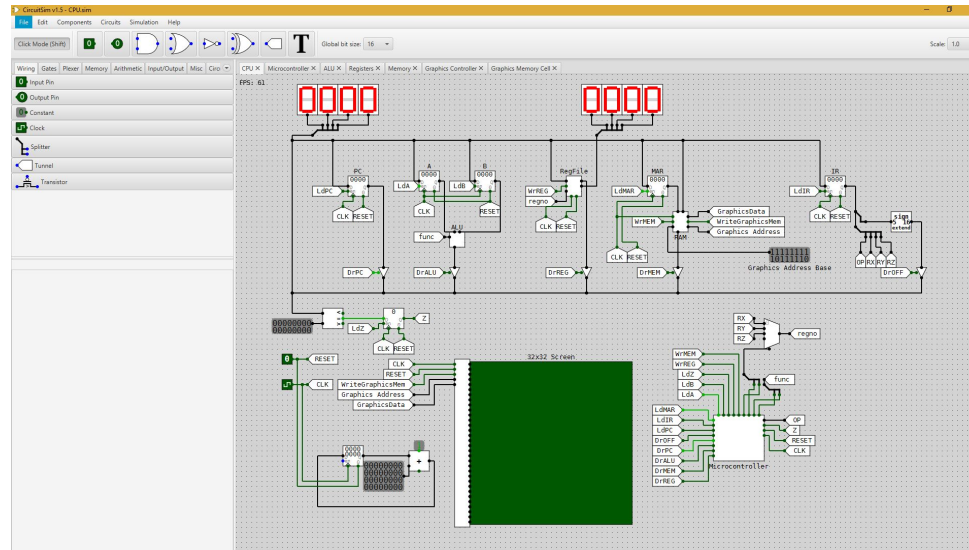# CS 2200

# LAB 1

# CircuitSim Installation

Version 1.9.1 is what we will be using for projects in this class.
More specific instructions on Canvas under Files > Installations
Download link: https://ra4king.github.io/CircuitSim/download
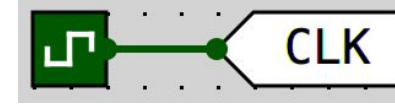
# C Installation

- Most recommended: Docker
  - Follow the installation guide under Canvas > Files > Installations > Docker Setup
- If Docker doesn't work, you can install in your local environment:
  - Windows: (recommend wsl) gcc + gdb
  - Mac: gcc + lldb
  - Linux: gcc + gdb

NOTE: all files/folders that you want to run in the Docker VM must be placed in the workspace folder

# **Tips** for CircuitSim

- Use **subcircuits** where necessary, they save space and abstract away logic. Please do not use subcircuits for single registers!
- **Tunnels** are an excellent way to clean up your circuit, and make it more comprehensible.
- **Labels** all you components, makes things easy to see and understand when wiring.
- Place **probes** in important areas in order to visualize the inputs into a component.
- Since 1.8.5, CircuitSim has introduced **breakpoints** that allow you to stop the clock when a line reaches a certain value, this may be very useful for debugging.
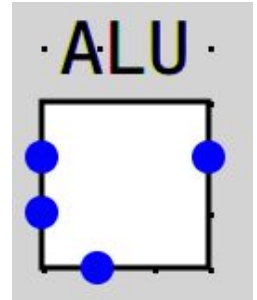
# **Practice** CircuitSim

- Build a 32-bit accumulator register that increments 1 every clock cycle. After you start the clock, it should be able to stop at 1000 (hex: #3e8)
  - You might need these components: clock, register, constant, adder, breakpoint
  - To start the clock, you can click Simulation > Clock Enabled. You might want to change the clock frequency to an appropriate speed.

# Practice CircuitSim SOLUTION



BP set to value of 1000
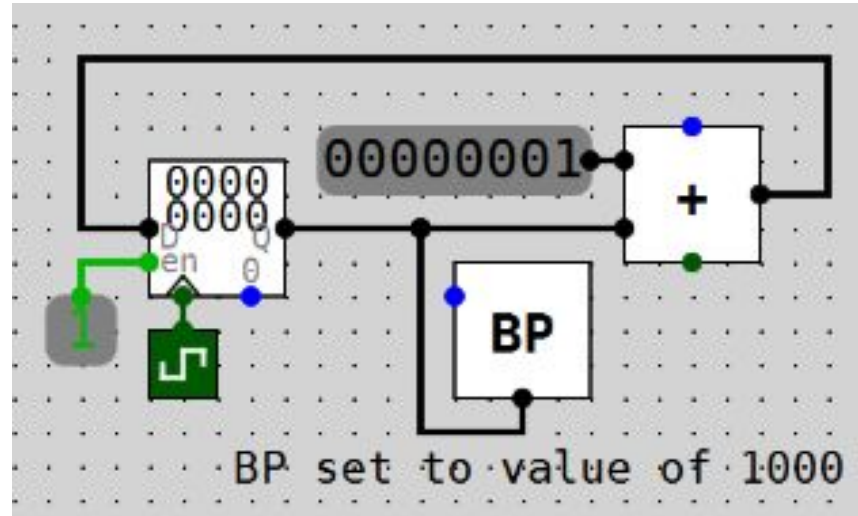
# **Practice** C

- Write a program that sums up 1 to 100 and prints the result.
- Compile the program with gcc using this command:
    - gcc -g ./test.c -o test
- Run it in NOT gdb:
    - ./test

Remember: You have to `#include <stdio.h>` to use printf

NOTE: make sure you put your c files in the workspace folder in order to have access through the VM

# Practice C **SOLUTION**

```c
#include <stdio.h>

int main(void) {
    int sum = 0;
    for (int i = 1; i <= 100; i++) {
        sum += i;
    }
    printf("%d\n", sum);
}
```

Prints 5050