

Ramaiah Institute of Technology
(Autonomous Institute, Affiliated to VTU)

Department of CSE

Programme: B.E
Course: Computer Organization

Term: Jan to May 2019
Course Code: CS45

Activity V: Designing an ALU to perform arithmetic and logical functions using Logisim simulator.

Name: Madhur	Marks: /10	Date: 18-05-2020
USN: 1MS18CS063	Signature of the Faculty:	

Objective: To simulate the working of Arithmetic and Logical Unit using simulator.

Simulator Description: Logisim is an educational tool for designing and simulating digital logic circuits. With its simple toolbar interface and simulation of circuits as you build them, it is simple enough to facilitate learning the most basic concepts related to logic circuits. With the capacity to build larger circuits from smaller sub circuits, and to draw bundles of wires with a single mouse drag, Logisim can be used (and is used) to design and simulate entire CPUs for educational purposes.

Activity to be performed by students:

List out the steps in designing ALU

dict out the steps in designing ALU

- 1.) ADD two i/p pins . Name them A and B
2. ADD or , and , ex-or , nor gates and 1-bit-adder.
- 3.) Connect the A's and B's of all the gates to their respective pins
4. Add an output pin and name it Result.
- 5 Add 1 bit Multiplexer with 3 select bits.
6. Connect outputs of all the gates to the mux and 3 last input pins to mux

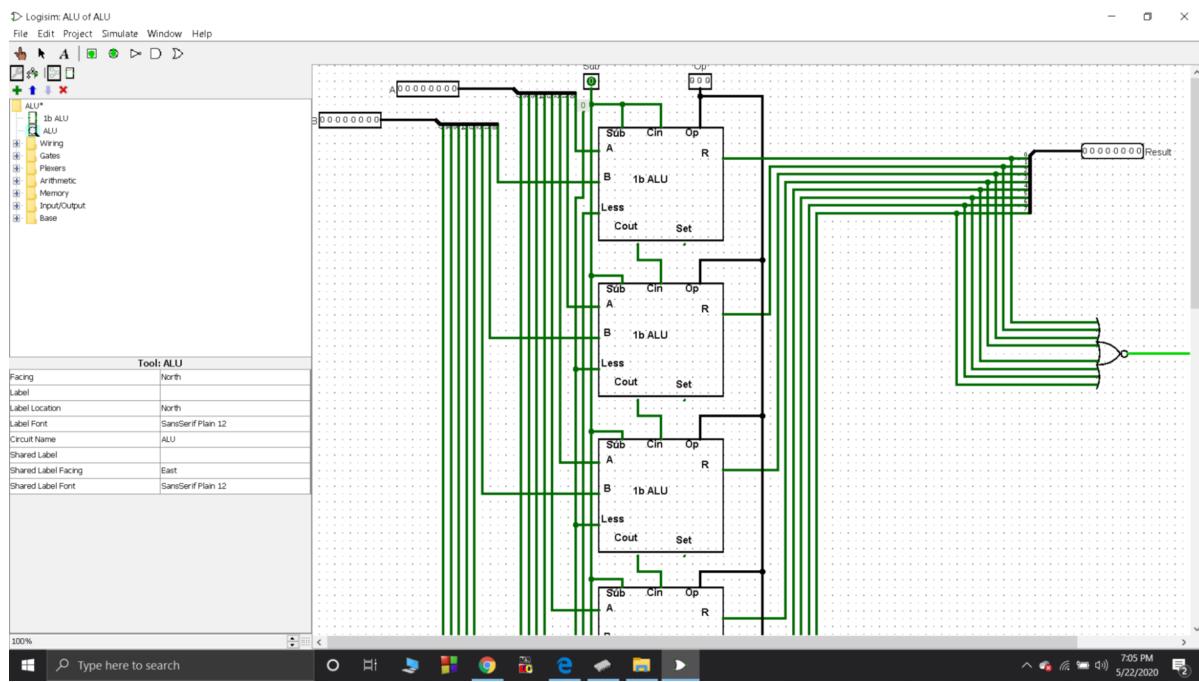
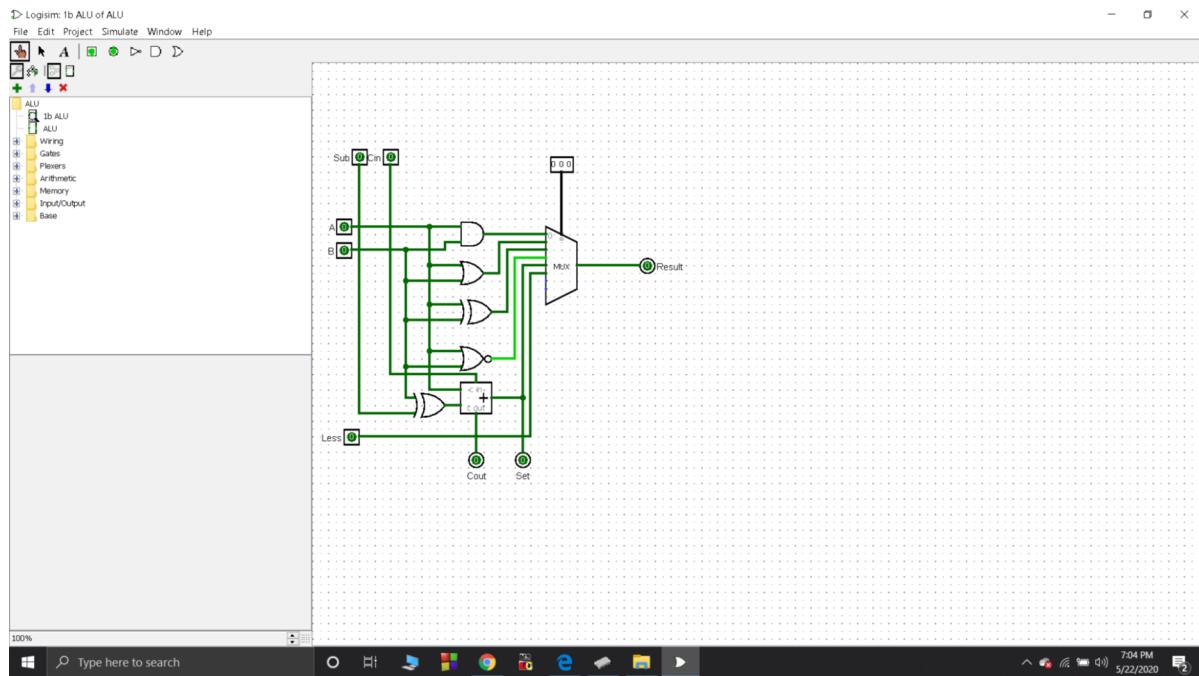


Scanned with CamScanner

- 7 Add i/p to Cin and output pin to Cout.
8. Add an ex-or gate . Connect its o/p to cout . The first input must be connected to B and second to another i/p pin sub.
- 10 Add another input and name it dels . Connect it to Mux.
- 11 Add an output pin and name it Set , connect it to the o/p of adder unit .
- 12 Now we use this 1 bit ALU to build an 8-bit ALU by making 8 copies of it and connecting the pins and multiplexers as mentioned above



Snapshots:



Ramaiah Institute of Technology
(Autonomous Institute, Affiliated to VTU)

Department of CSE

Programme: B.E
Course: Computer Organization

Term: Jan to May 2019
Course Code: CS45

Activity VI: Designing memory system using Logisim simulator.

Name: Madhur	Marks: /10	Date:19-05-2020
USN:1MS18CS063	Signature of the Faculty:	

Objective: To simulate the writing operation on memory.

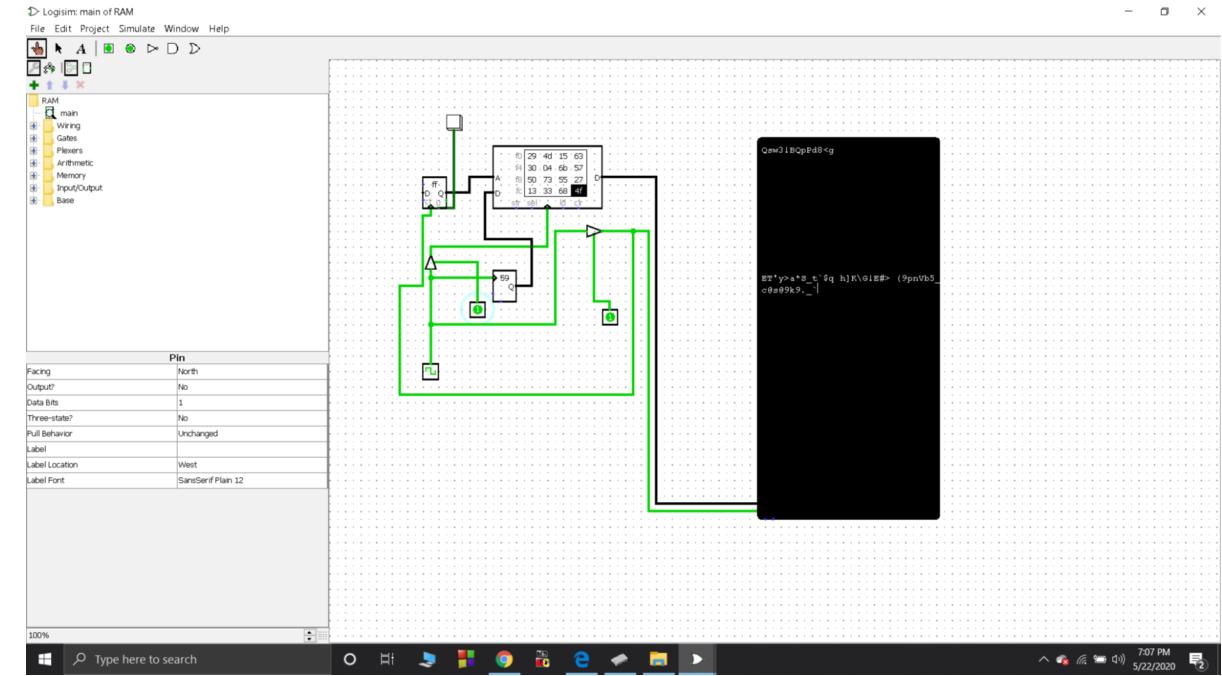
Simulator Description: Logisim is an educational tool for designing and simulating digital logic circuits. With its simple toolbar interface and simulation of circuits as you build them, it is simple enough to facilitate learning the most basic concepts related to logic circuits. With the capacity to build larger circuits from smaller sub circuits, and to draw bundles of wires with a single mouse drag, Logisim can be used (and is used) to design and simulate entire CPUs for educational purposes.

Activity to be performed by students:

List out the steps in designing memory system

1. Add a RAM with separate load and store selected.
2. Add a counter and connect Q to A of the RAM
3. Add a controller buffer and connect its op to the RAM
4. Add a clock and connect to the ip of the buffer.
5. Add a TTY with 32 rows and columns. Make the connections with RAM.
6. Add a 7-bit random number generator, connect Q to D
7. Add another controlled Buffer, connect to TTY. Also add an ip pin to the buffer.
8. Connect the output of the second buffer to the counter.
9. Connect a button to the counter.

Observations and Snapshots:



Ramaiah Institute of Technology
(Autonomous Institute, Affiliated to VTU)

Department of CSE

Programme: B.E
Course: Computer Organization

Term: Jan to May 2019
Course Code: CS45

Activity VII: To simulate advantages of using pipeline technique in executing a program.

Name: Madhur	Marks: /10	Date: 20-05-2020
USN: 1MS18CS063	Signature of the Faculty:	

Objective: To learn and analyze the performance of the CPU by overlapping of instructions using CPUOS-SIM simulator.

Simulator Used: CPUOS-SIM is a software development environment for the simulation of simple computers. It was developed by Dale Skrien to help users to understand [computer architectures](#).

Modern CPU's contain several semi-independent circuits involved in decoding and executing each machine instruction. Separate circuit elements perform each of these typical steps:

- Fetch the next instruction from memory into an internal CPU register.
- Decode the instruction to determine which function sub-circuits it requires.
- Read any input operands required from high-speed registers or directly from memory.
- Execute the operation using the selected sub-circuits.
- Write any output results to high-speed registers or directly to memory.

Separate sections of the CPU circuitry are used for each of these steps. This allows these circuit sections to be arranged into a sequential pipeline, with the output of one step feeding into the next step.

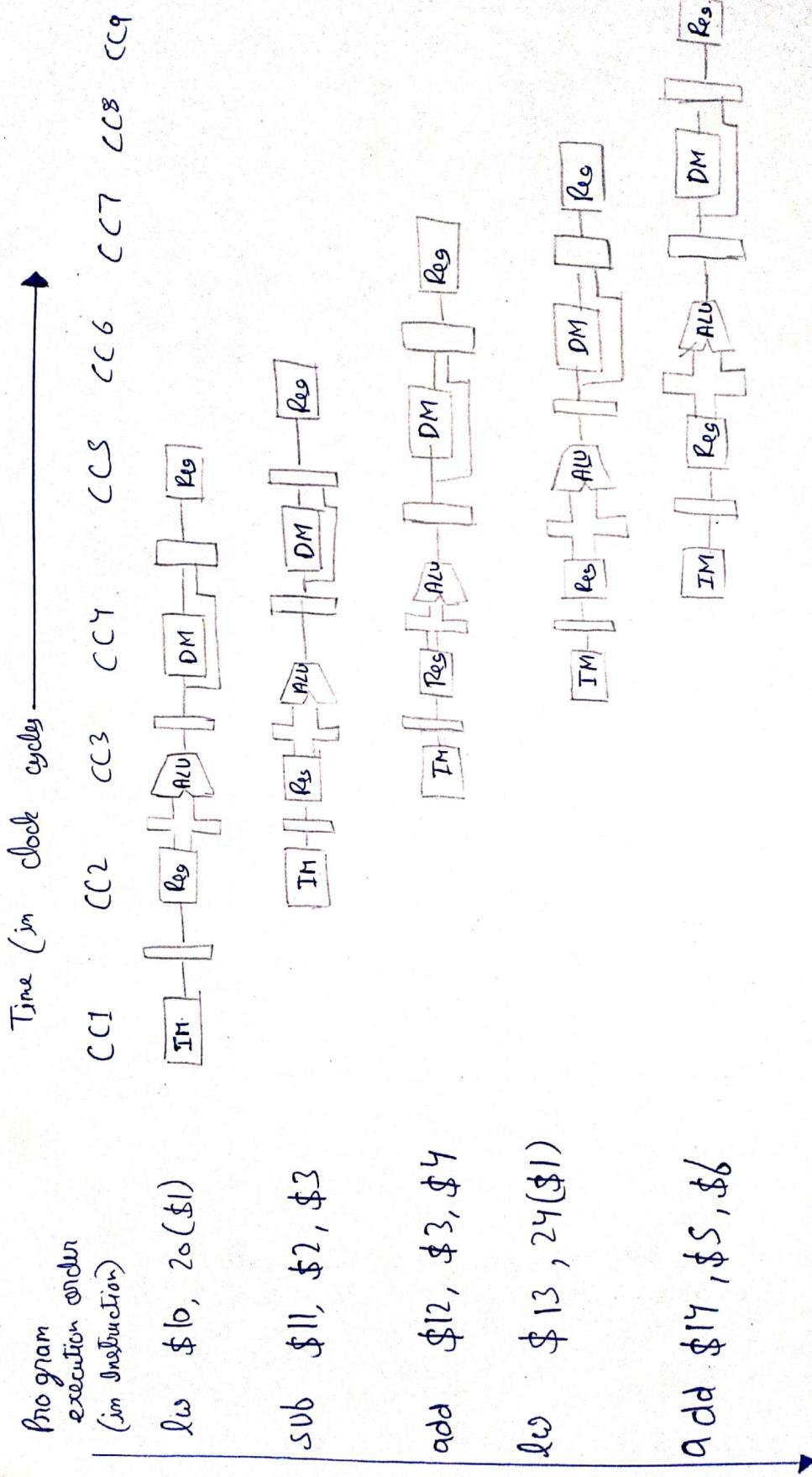
Activity to be performed by students:

With diagram demonstrate the execution of the following instructions using pipelining technique.

lw \$10,20(\$1)
 sub \$11, 42, \$3
 add \$12, \$3, \$4
 lw \$13, 24(\$1)
 add \$14, \$5, \$6

Program execution order (in instructions)	Time (in clock cycles)				
	CC1	CC2	CC3	CC4	CC5
lw \$10, 20(\$1)	Mem fetch	Decode	Execution	Data Access	Write Back
sub \$11, \$2, \$3	Fetch	Decode	Execution	Data Access	Write Back
add \$12, \$3, \$4	Fetch	Decode	Execution	Data Access	Write Back
lw \$13, 24(\$1)	Fetch	Decode	Execution	Data Access	Write Back
add \$14, \$5, \$6	Fetch	Decode	Execution	Data Access	Write Back

Program execution order
(in instruction)



Observations and Snapshots: Take the snap shot of CPU statistics and pipeline design.

