ISLAMIC UNIVERSITY OF TECHNOLOGY (IUT)
ORGANISATION OF ISLAMIC COOPERATION (OIC)
DEPARTMENT OF ELECTRICAL AND ELECTRONIC ENGINEERING

PHASE- B

TEAM NAME: **BOOLEAN AUTOCRATS**

TEAM MEMBERS:

Name	IDs
1. K.M. Sirazul Monir	200021247
2. Shafin Ibnul Mohasin	200021244
3. Md. Muhidur Rahman	200021242
4. Sk Asad Al Abir	200021331
5. Tahmid Hasan Muttaky	200021310

Supervisor :Ahmad Shafiullah
Lecturer, EEE

Introduction:

For phase B of Sap-1, we have 2 types of modules to construct.

1. Register:
   a) Accumulator
   b) B- register
   c) output register
2. Ram

Registers are our main part in Sap-1 which stores data and necessary instructions and releases it to the corresponding module for processing as per the clock cycle. On the other hand, ram holds the data, address, and instructions which will be processed in a structured way, unlike registers which store 4 or 8-bit data.

Designing and implementation:

Accumulator: The Accumulator is also known as A register. It is connected to the bus and ALU. As per the clock pulse and controller sequence command, it can take input from the bus, store data, and output it to the bus or ALU. It is basically 4 j-k flipflops with some enable switch and tri-state buffers which can store 8 bit data. We used two 74LS163 ic which has 4 j-k flipflops for 8 bits. 74LS245 IC was used as tri-state buffer. Pin 14,13,12,11 of each register chip was the input side and pin 2,3,4,5 was our output. pin 1,2 of each register was shorted which was used to toggle output enabling. It was active low toggle so we grounded it. Pin 9,10 was used to load data into the register. If enabled, the register will load and hold the 8 bit data. Pin 15 of each ic is for clearing the loaded data.
As for the tri-state buffer, pin 19 is for enabling output. It is active low. pin 1 is for setting direction which we connected to the ground.

B register: B register is similar to Accumulator. However, unlike Accumulator, it cannot output to the bus. Instead, it outputs to the ALU directly. As there's no output path to the bus, tri-state buffer is not required. It also stores 8 bit data.

Output register: It is similar to B register. It is connected to the bus for taking input and outputs it to the output unit as per clock cycle and controller sequence. It cannot output data to the bus. It also stores 8-bit data.

Ram: The Ram or Random access memory is our primary storage for SAP-1. It is basically bunch of j-k flipflops, which can hold total of 128 bit or 16byte of data. It has 16 addresses and each address can hold 8 bits of data so in total 16×8=128 bit of data. If we remove the power source it loses all the data stored in it and that is why it is called RAM.
We used 2 74LS189 chip which can hold 64 bit of data each to make our ram. Pin 3 is Write enable switch which is active low. When it is in the 0 position ram will take input and write data to the memory. when it is 1 the ram becomes read only. The outputs of ram are all inverted. So we used a hex inverter to invert the data from ram.

Software simulation :

We constructed the circuits using Proteus 8. Each module's output was shown using LED.

RAM:

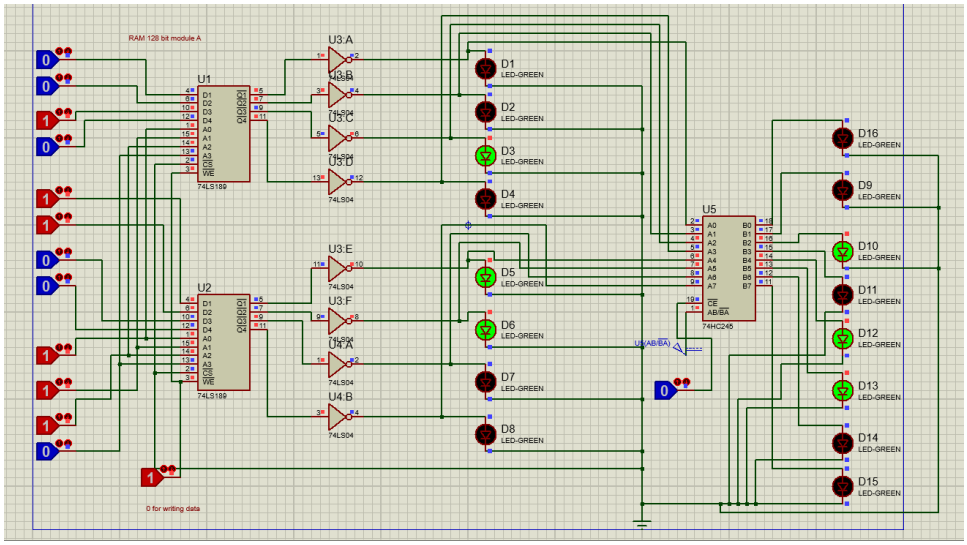

Fig 1: Ram simulation in proteus

Accumulator:

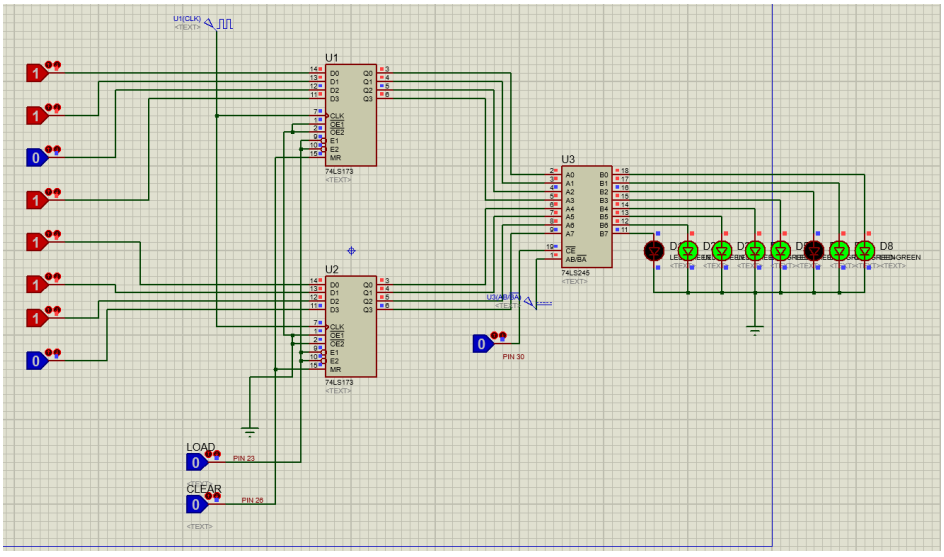

Fig 2: Accumulator

B register:

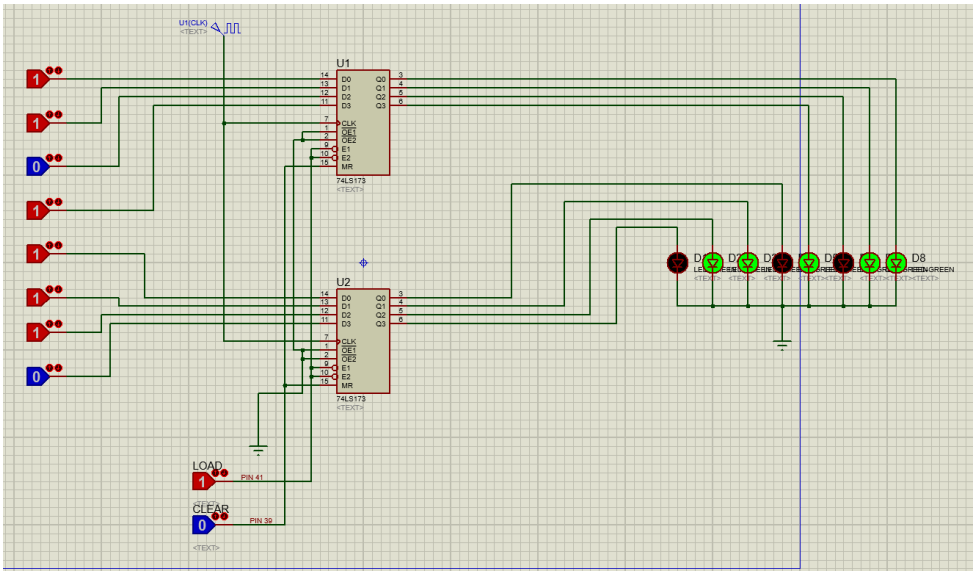

Fig 3: B register

Output register:

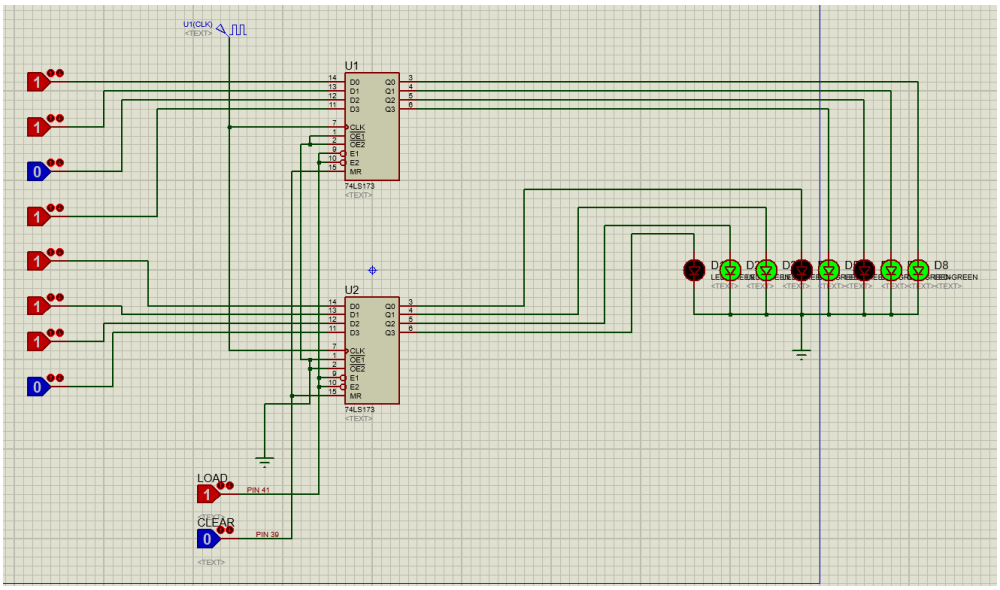

Fig 4: Output register

Hardware implementation:

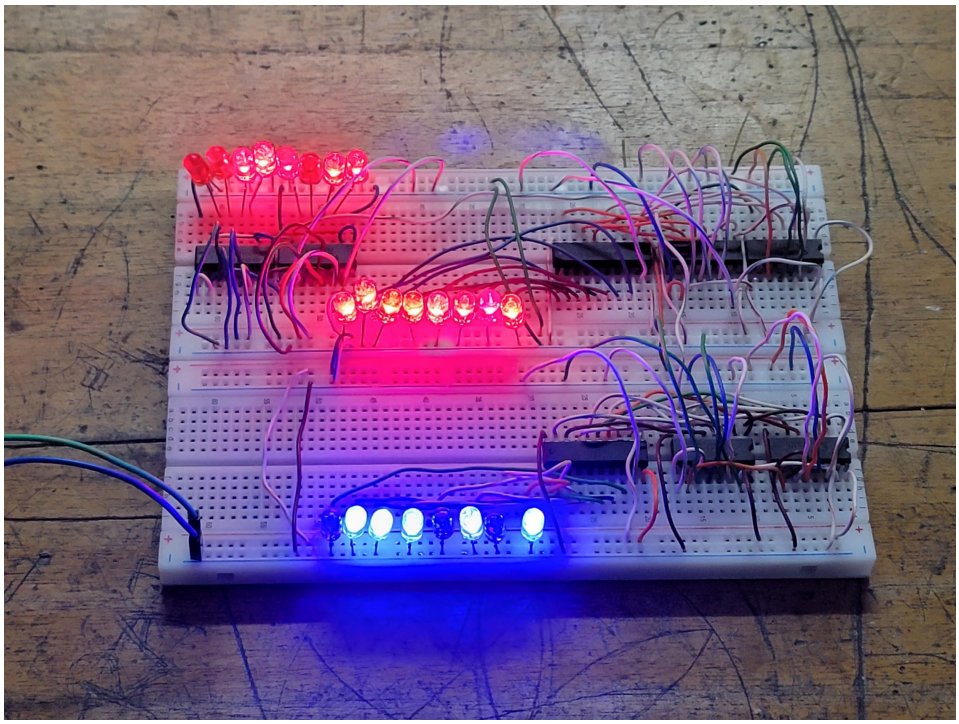

Fig 5: All registers

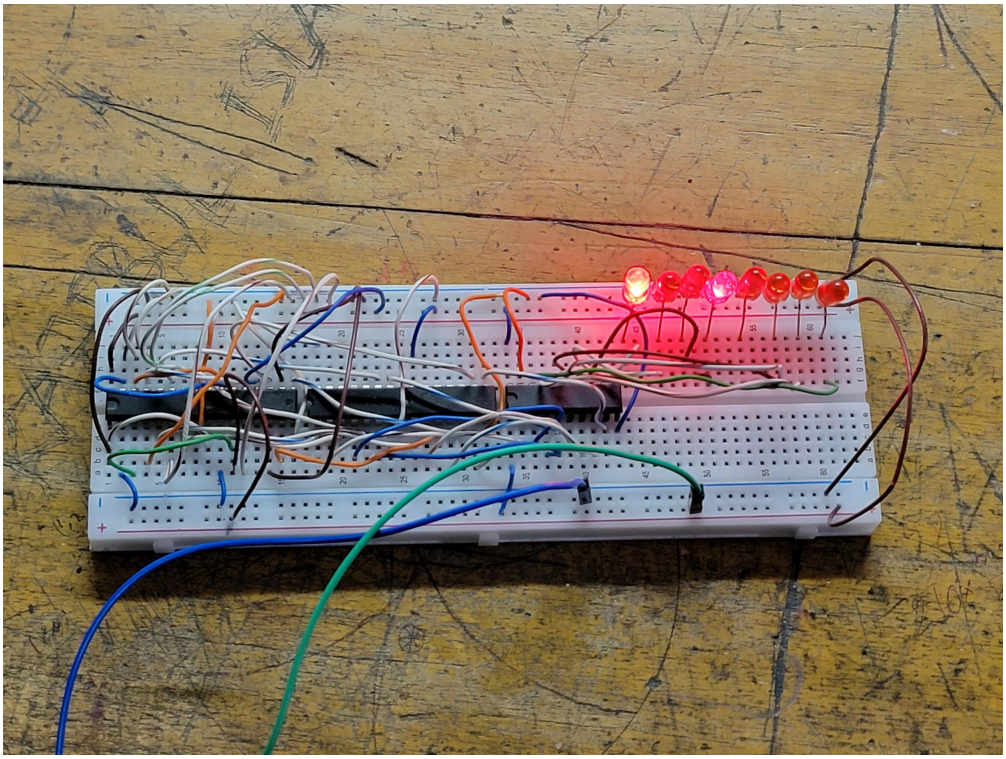

Fig 6: Ram module

Here are the parts list for phase B:

Accumulator:

1. 74LS173 - 2x
2. 74LS245 - 1x
3. LED - 8x
4. Breadboard

B register:

1. 74LS173 - 2x
2. LED - 8x
3. Breadboard

Output register:

1. 74LS173 - 2x
2. LED - 8x

RAM:

1. 74LS189 - 2x
2. 74LS04- 2x
3. 74HC245 - 1x
4. LED - 8x
5. Breadboard

We implemented everything on the breadboard and used the manufacturers datasheet for the pinouts. We stripped USB cables and took 5v and ground from that to power up the circuit. We used wires from Ethernet cable as jumpers. outputs were shown using LED.

Logistics:

Total cost of all the components and their prices:

555	10	0	0
741	20	0	0
jumpers	2	0	0
breadoard	140	3	420
0	25	1	25
2	25	0	0
4	25	3	75
8	30	3	90
32	25	3	75
86	25	0	0
138	35	0	0
139	40	0	0
157	35	5	175
283	40	0	0
107	35	0	0
173	70	7	490
189	70	2	140
245	55	3	165
273	null	0	0
red	5	25	125
blue	5	10	50
yelloow	5	3	15
green	5	5	25
7 seg	50	0	0
6 pin	12	0	0
spdt	6	6	36
8 position	30	0	0
4 position	20	0	0
push	6	4	24
1k	1	0	0
10k	1	0	0
100k	1	0	0
220	1	0	0
470	1	0	0
1M	1	0	0
1M pot	10	0	0
10uF	2.5	0	0
1uF	2.5	0	0
0.1	2.5	0	0
103	3	0	0
104	1.5	0	0
			1930

Our total cost of phase B was BDT 1930 TAKA.

Open discussion:

During building the registers we realized all of were almost same. As we built more of them, we truly understood the use of registers in actual CPU. Register holds data for processing, but we know RAM also holds data for the computer to process. But the difference is that the data held by the register are much smaller and at base level than RAM. Also registers are much, much faster than RAM. However both of the modules use J-k flip-flops for storing data. Also we had some confusions regarding ram addresses and values. Everything cleared up after we designed everything in software and as well as hardware.

The End