EE2080
8 Bit Computer

Raman Pareek 122101031
Sai Rohan 122101030

ABSTRACT:

In this exercise, basic building pieces including logic gates, registers, and buffers were used to create a straightforward 8-bit computer. Due to its simplicity, this version of the 8-bit computer can only "Add" and "Subtract" 8-bit values and show the outcome.

1. Introduction:

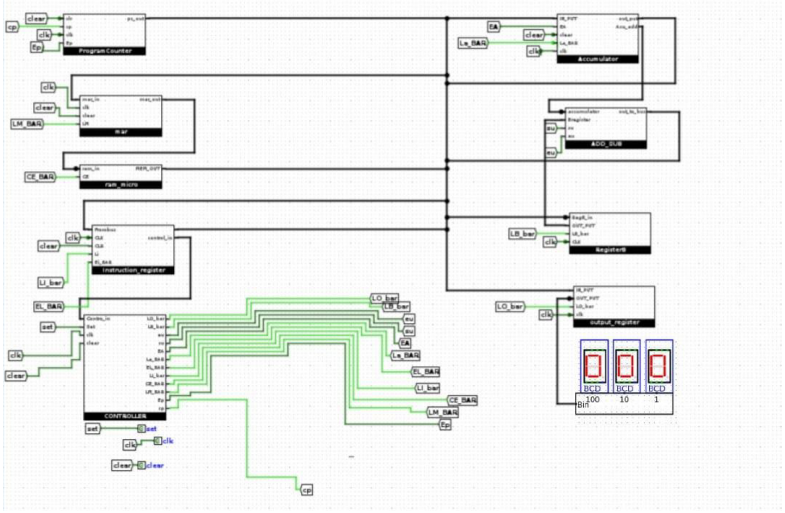


Figure 2: Circuit Block Diagram Realization

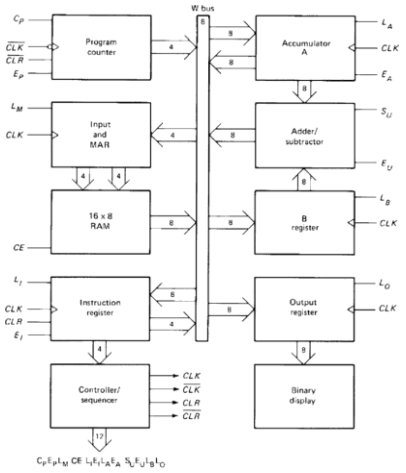


Figure 1: Block Diagram Design Concept

The block diagram in Figure 1 is present in the circuit depicted in Figure 2 above.

The input and output of every module are linked to a common 8-line bus. A tri-state buffer is built into each module to regulate how it connects to the bus. In order to enable the connection of a 4 bit input/output to an 8 bit bus, wire splitters have been employed. Tri-state buffers have been used to prevent short circuits on the bus. The L and E control signals stand for "Load" and "Enable," respectively. Some of the clocked modules are activated at positive clock edges2 and others are activated at negative clock edges3. This was done to make sure that the registers would load with data.

i. LDA - 4b'0000: Load to Accumulator
ii. ADD - 4b'0001: Add and load result to accumulator
iii. SUB - 4b'0010: Subtract and load result to accumulator
iv. OUT - 4b'0100: Output the accumulator value to the Output register
v. HLT - 4b'1111: Stop the clock and halt operations

Timing Diagrams:

The timing diagrams for each instruction are shown in Figures [insert the figures here]. The details are mentionedin 2.9.

2. Implementation and Results:

Program Counter:

The easiest task is to indicate the instruction number using the program counter.

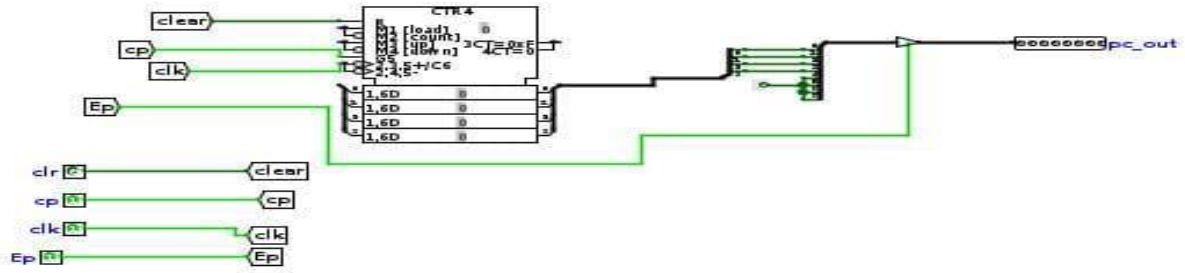


Figure 3: Program Counter Module

The control signal for this module consists of C_p for increment of the count and E_p for the output of the count. This is depicted in Figure 3. It is a clocked module with clear (**CLR**).

Input and Memory Address Register:

The "address value" of the instruction or memory in the memory unit is stored and located using the memory address register (Figure 4). The Register is a 4 bit register since the memory unit used in this case is an 18 8 ROM.

At a negative clock edge, the Lm control signal permits writing into the register. The output of the register is directly connected to the ROM's address pointer.

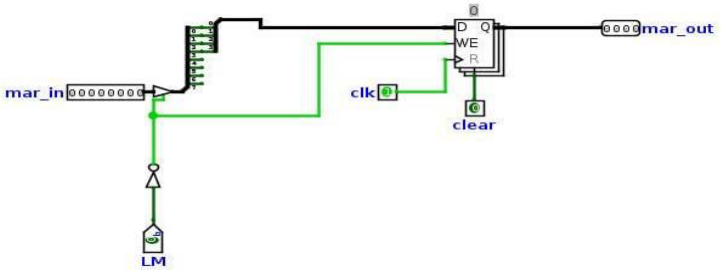


Figure 4: MAR internal circuit

The memory unit houses all the information pertaining to the commands and data that must be processed by the computer.

Here, the memory module only has one control signal, CE, which regulates the module's output. The memory module used is a ROM module, and the MAR module points to its address.

Memory Unit:

The memory unit houses all the information pertaining to the commands and data that must be processed by the computer. Here, the memory module (Figure 7) only has one control signal, CE, which regulates the module's output. The memory module used in Figure 8 is a ROM module, and the MAR module points to its address.

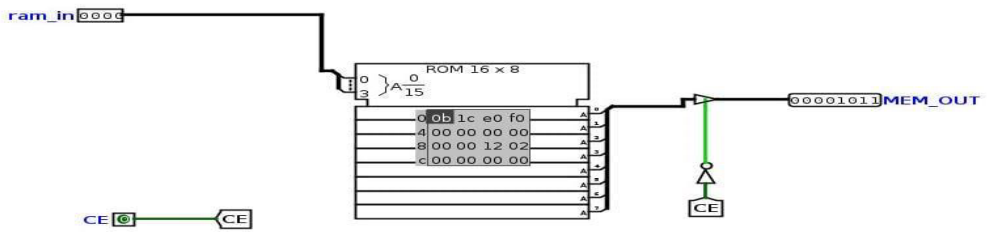


Figure 5 : Internal Circuit in Memory

Instruction Register:

The instruction register acts as a bridge between the ROM and the control unit by storing the instruction from the memory unit. Figure 8 illustrates the two control signals that the module has: Li for loading the instruction from the memory unit onto the register and Ei for sending the memory location back to the MAR Module.

Only an eight bit register and a negative edge clock make up the underlying hardware, as shown in Figure 10. The output has likewise been divided into two four bit lines, with the bus being connected to the least significant line and the control unit receiving the most significant line. It is a clear-clocked module (CLR).

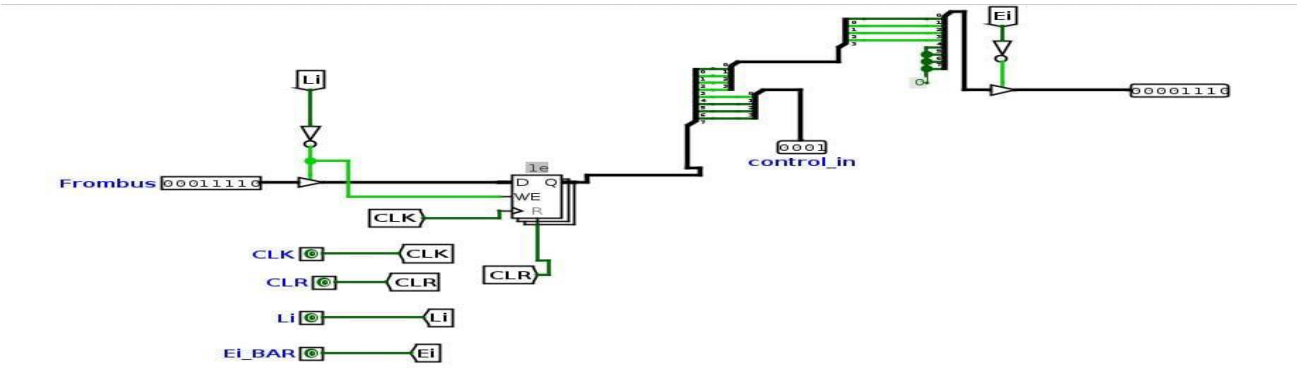


Figure 6: Internal Circuit in Instruction Register

Accumulator:

The Accumulator, a straightforward register that is directly connected to the Arithmetic Unit, stores the data from the ROM. The input and output connections of the register to the bus are controlled, respectively, by the module's two control signals La and Ea, as seen in Figure 11. It is a clear-clocked module (CLR).

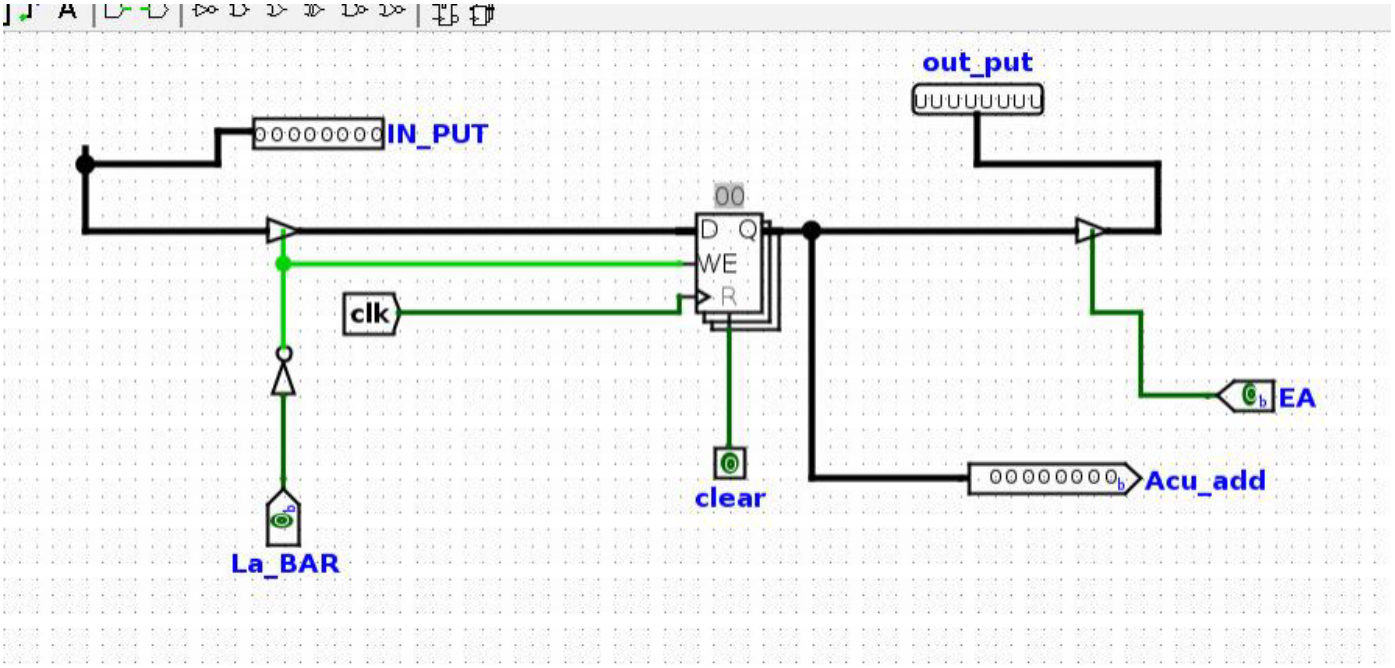


Figure 7: Accumulator Module

Arithmetic Unit:

The task of doing arithmetic calculations falls under the arithmetic unit. Only addition and subtraction using the complement of two can be performed using the intended arithmetic unit. An inverter module is used to accomplish this (Figure 13). The second number is simply inverted by the inverter module, and subtraction is performed by setting the initial carry to 1. The initial carry is set to 0, and because the inverter module functions as a buffer for addition, addition is carried out. Figure 15 illustrates how the Su directs the action. When the number is 0, addition is done, and when it is 1, subtraction is done. The Eu signal, which regulates the output to the bus, is one of the additional control signals shown in Figure 14.
This circuit is not timed.

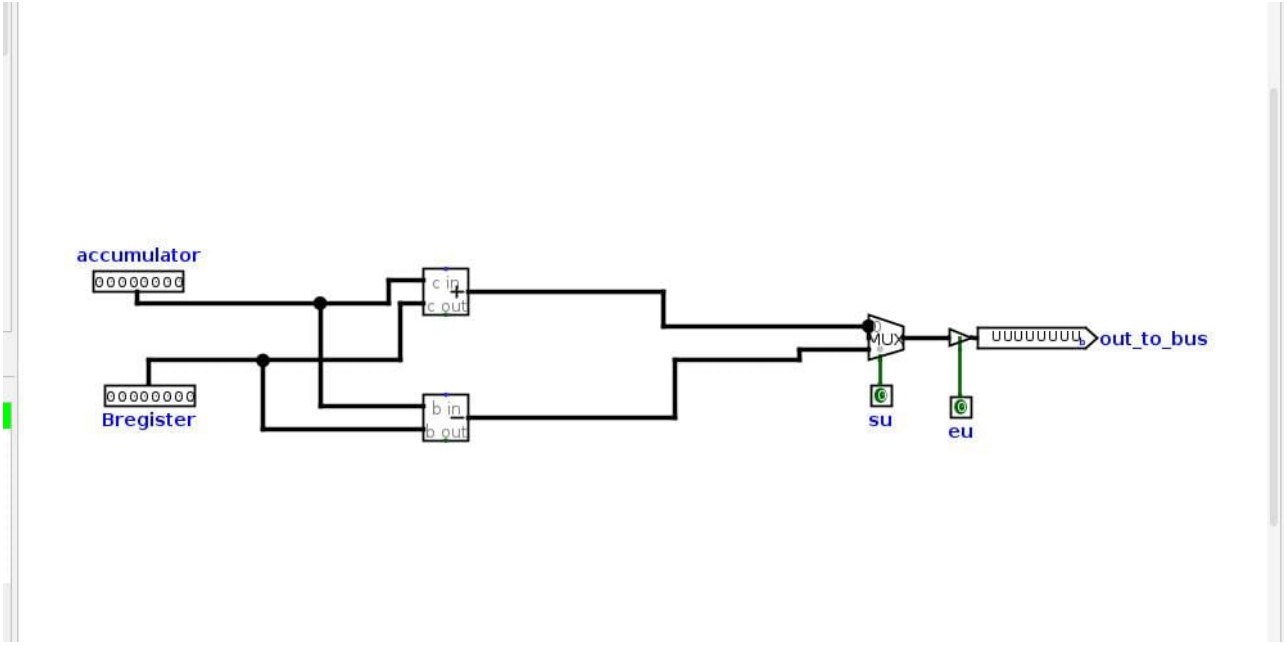


Figure 8: Internal Circuit in Arithmetic Module

Register B

Register B is a straightforward register that is directly connected to the arithmetic unit and stores data from the ROM. According to Figure 16, the module has a single control signal Lb that regulates the register's input to the bus. It is a clear-clocked module (CLR).

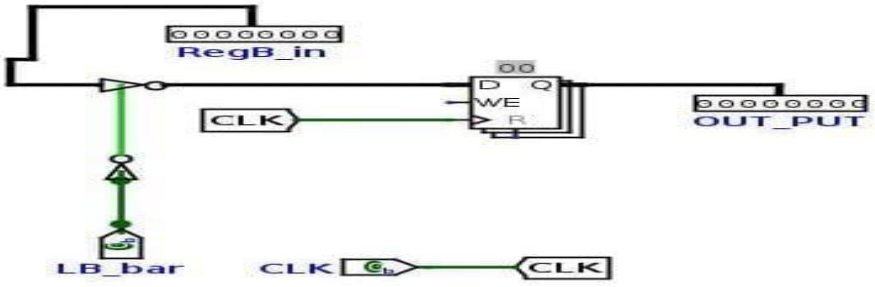
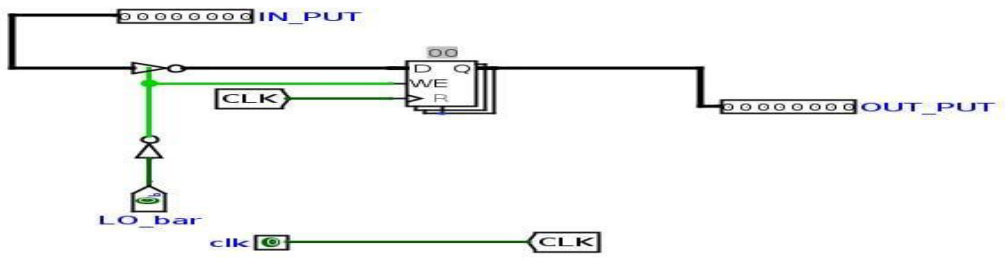


Figure 9: Internal Circuit in Register B

Output

The output serves as a bridge between the output device and computer by acting as a straightforward register to store the output. A series of three seven-segment displays are used in this instance, as shown in Figure 18, to display a decimal value from the output register.

There is only one control signal, Lo, in the constructed module, and it allows writing to the register on a negative clock edge. It is a clear-clocked module (CLR). A Binary to Decimal Decoder, a Decimal to Seven Segment Converter, a Decimal to Seven Segment Converter, and finally a Seven Segment Display are all

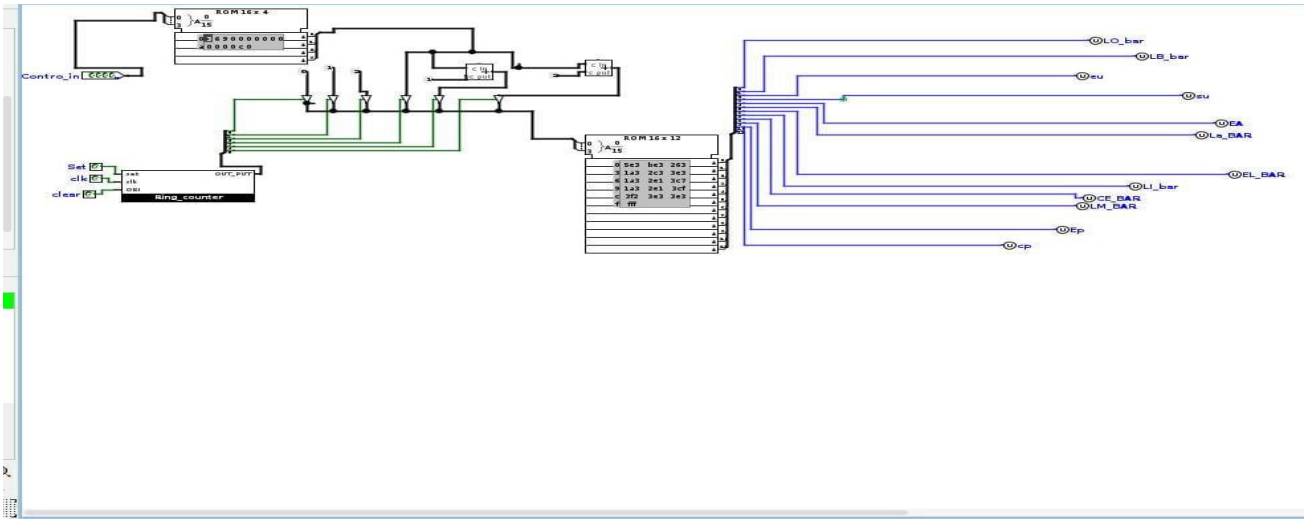


attached to the register's output.

Figure 10: Internal Circuit in Output Register

Control Unit:

The control unit which will act as brain. It tells the computer memory and the adder/subtractor block and input and output devices how to respond to the instructions that have been sent to the processor. It also controls the Tristate buffers of each block which means which pin need to be enabled at particular state .There is a ring counter in the controller which is used for to get continuous cycling process(T1 to T6) process where the first three states(T1 to T6) will be common but (T4 to T6) will depends on the instruction that we get from the IR output. There will be 16*12 ROM which is used for to which control pin should be enabled at a particular state.



FETCHING CYCLE:

(First 3 clock cycles T1 , T2 , T3):

i)Address state: In this state the data and instructions which are present at the program counter will go to the memory address register through the 8 bit bus by enabling Cp pin and Ep pin which is acting as the tristate buffer between bus and pc counter.

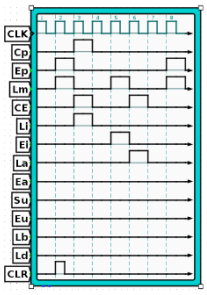
ii)Increment state: Now there is no data& instructions at the pc counter .So during this state the counter counts and also the pc counter points to next address(increments the current address by 1) of the instructions at the pc counter .

iii)Memory state: The data & instructions will go to the ROM(16*8) which contains 16 locations of 8-bit length where the first 8 address locations will be the instructions and next 8 address locations are data and these data will go to the instruction register through the 8 bit bus by enabling of CE_bar and LI_bar.

Control Signals of Instructions

The instructions are as follows:

1. LDA



4 **Clock-** loads the address location onto the MAR. Control signals L_m and E_i have to be given as 1 whereas the rest are 0.

5 Clock- the contents of the memory unit are loaded to the accumulator. Control signals CE and L_a are given as 1 and others are 0.

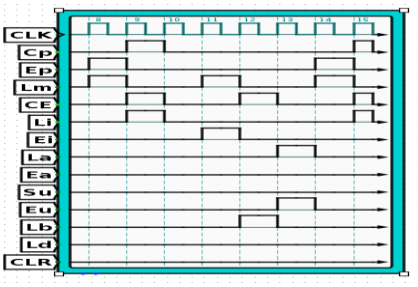
6 Clock- No change

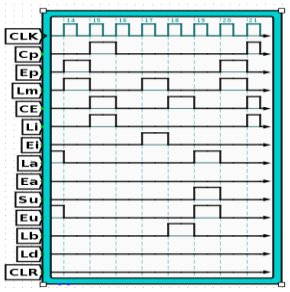
2. ADD

4 clock--loads the address location into the MAR. Control signals Lm and E_i have to be given as 1 whereas the rest are 0.

5 clock-- the contents of the memory unit are loaded to Register B. Control signals CE and L_b are given as 1 and the others are 0.

6 **clock**-the addition is performed and result is stored in the accumulator. Control signals L_a and E_u are given as 1 and rest as 0.



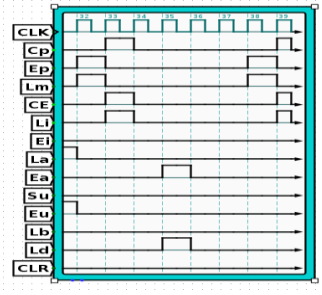


3.SUB

4 **clock --**loads the address location onto the MAR. Control signals L_m and E_i have to be given as 1 and the rest are 0.

5 **clock**-- the contents of the memory unit are loaded to Register B. Control signals CE and L_b are given as 1 and the others are 0.

6 **clock**-- the subtraction is performed and result is stored in the accumulator. Control signals L_a S_u and E_u are given as 1 and rest as 0



.

4.OUT

4 clock--the value in the accumulator is loaded onto the output register. Hence the control signals E_a and L_d are given as 1,whereas the rest are 0.

5 clock--no change

6 **clock** --no change

Instruction	State	C_P	E_P	$\overline{L_M}$	$\overline{CE}$	$\overline{L_I}$	$\overline{E_I}$	$\overline{L_A}$	E_A	S_U	E_U	$\overline{L_B}$	$\overline{L_O}$	Hex Code
Fetch	T_1	0	1	0	1	1	1	1	0	0	0	1	1	5E3
	T_2	1	0	1	1	1	1	1	0	0	0	1	1	BE3
	T_3	0	0	1	0	0	1	1	0	0	0	1	1	263
LDA (0000)	T_4	0	0	0	1	1	0	1	0	0	0	1	1	1A3
	T_5	0	0	1	0	1	1	0	0	0	0	1	1	2C3
	T_6	0	0	1	1	1	1	1	0	0	0	1	1	3E3
ADD/SUB (0001/0010)	T_4	0	0	0	1	1	0	1	0	0	0	1	1	1A3
	T_5	0	0	1	0	1	1	1	0	0	0	0	1	2E1
	T_6	0	0	1	1	1	1	0	0	0/1	1	1	1	3C7/3CF
OUT (1110)	T_4	0	0	1	1	1	1	1	1	0	0	1	0	3F2
	T_5	0	0	1	1	1	1	1	0	0	0	1	1	3E3
	T_6	0	0	1	1	1	1	1	0	0	0	1	1	3E3

5)HLT

This circuit has been implemented slightly differently as it is required to stop the clock. Hence when the instruction for this is read, the AND gate is turned off to switch the clock off.

Conclusion:

After this experiment one can get knowledge about how internally the architecture of 8 bit microcomputer is made and how the different components are working together. Memory allocation and storage of data in memory is known deeply. The importance of control unit and how control unit can manage signals can be known in this experiment. Application of knowledge learned in digital systems course is important in this experiment. There will be fetch and execute cycles to run and instruction. Speed of microcomputer depends on different factors like clock speed, instruction set, memory used eye.