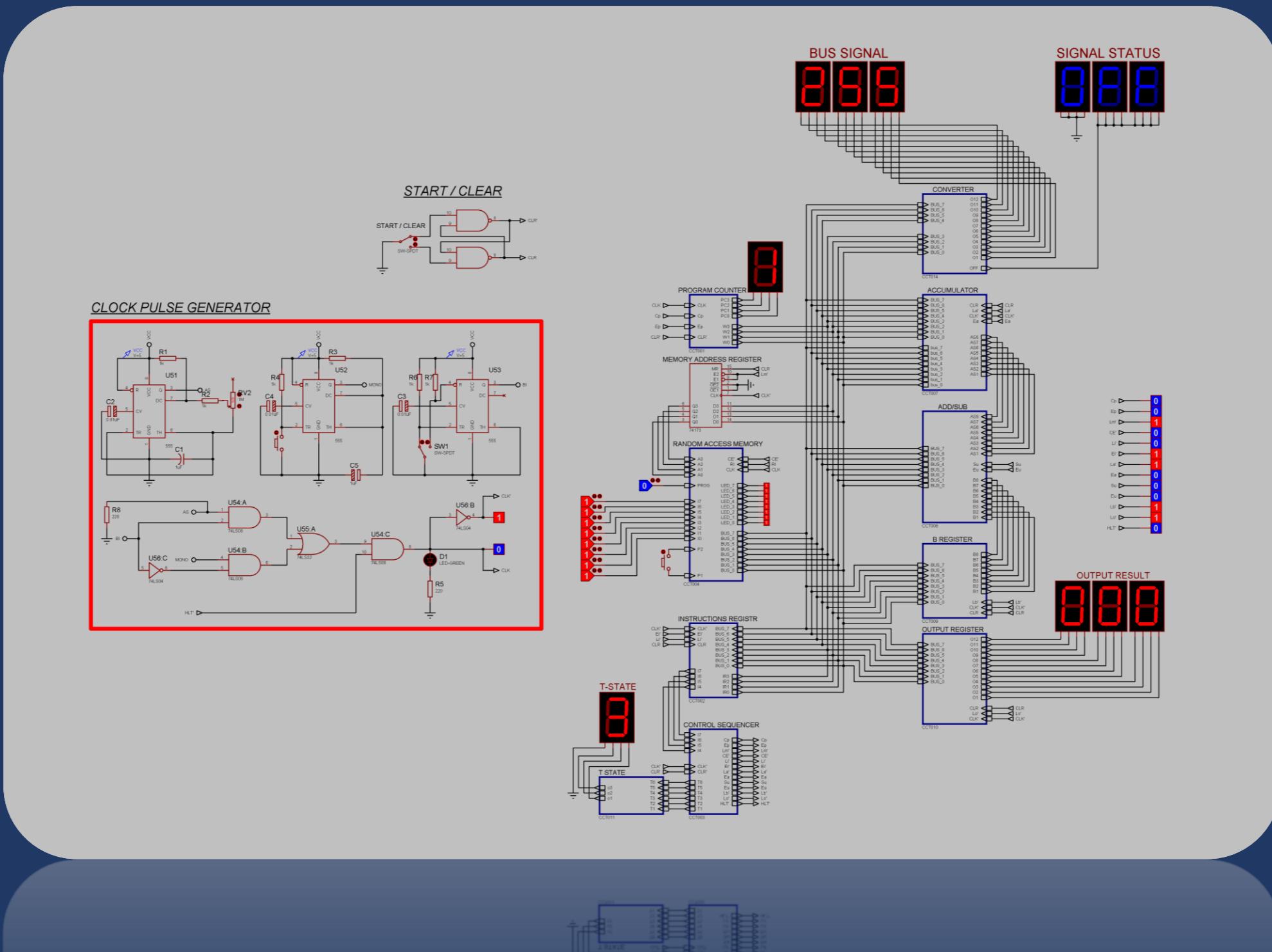


SAP-1 ARCHITECTURE-BASED 8-BIT COMPUTER DESIGN AND IMPLEMENTATION



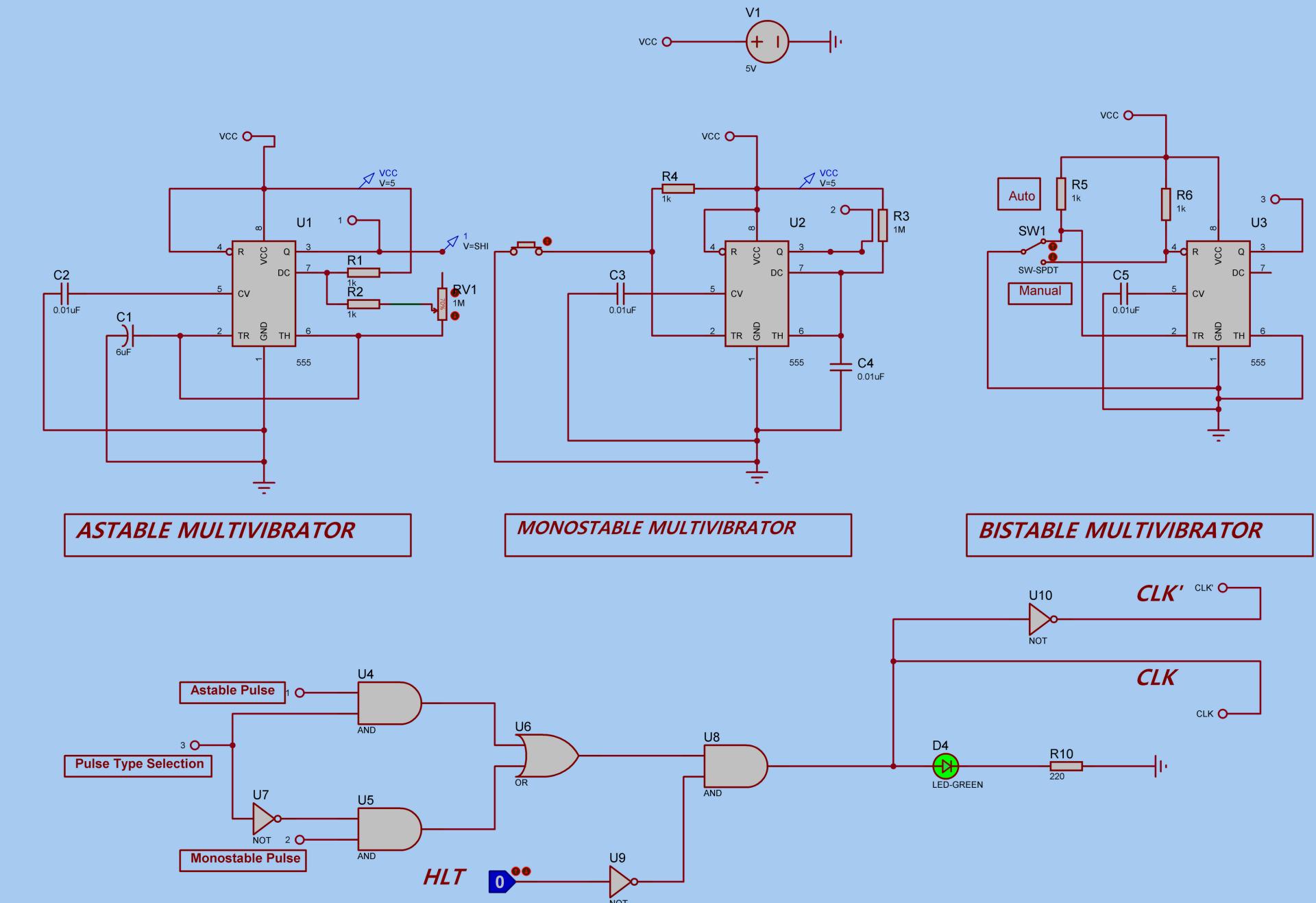
CLOCK PULSE GENERATOR

Astable Multivibrator = Means no stable state.
Mainly not stable in either state. It is basically an oscillating circuit. It oscillates between two states rapidly.

- **Monostable Multivibrator** = Only one stable state. The state which is basically off is the stable state. To get a manual debounced output pulse we can use this.

- **Bistable** = Two stable states. It gives us both low and high debounced output.

- **HLT** = It basically declares that no more instructions are needed to be followed. So, no more execution of commands. It just shuts off the computer.



PROGRAM COUNTER

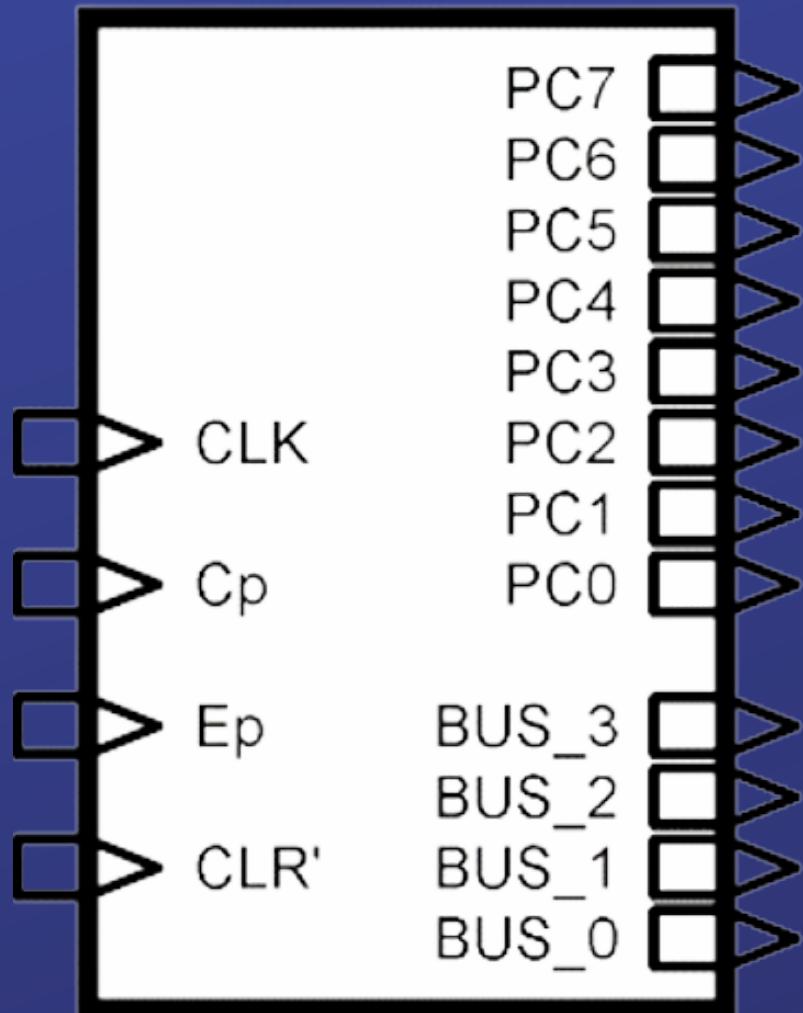


FIG: Block Diagram of Program Counter

I/O pins:

- **PC0 – PC7 = Output pins. Go to display.**
- **BUS_0 – BUS_3 = Output pins. 4-bit data directly go to BUS.**
- **AS1 – AS8 = Input pins. 8-bit data directly comes from Accumulator through these pins.**

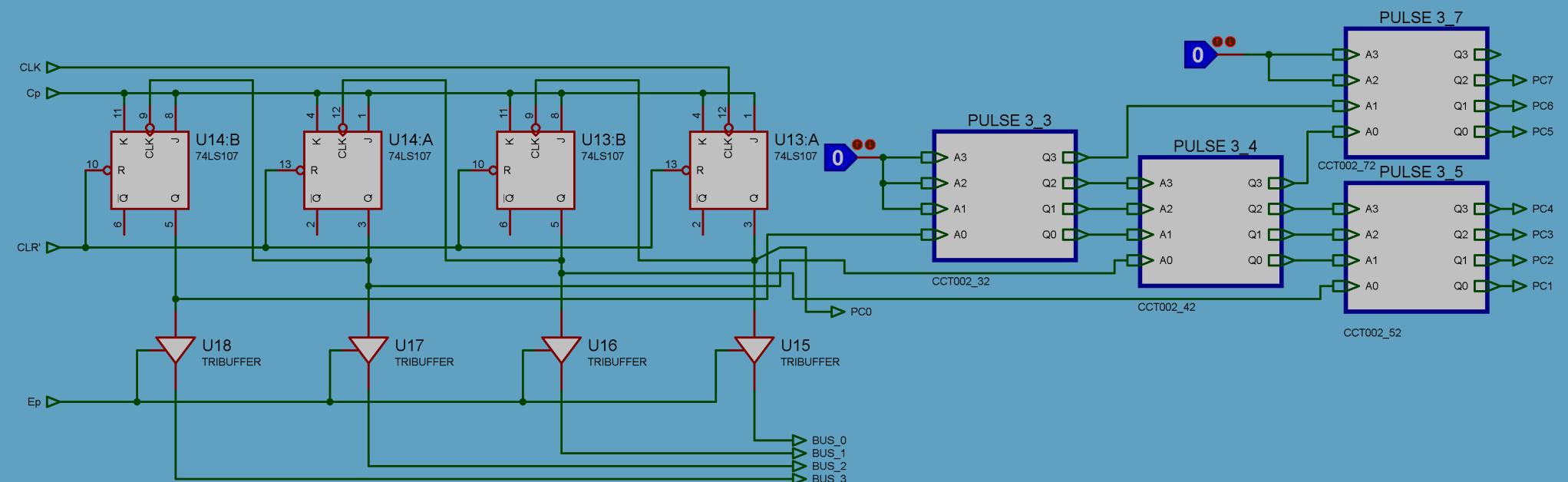
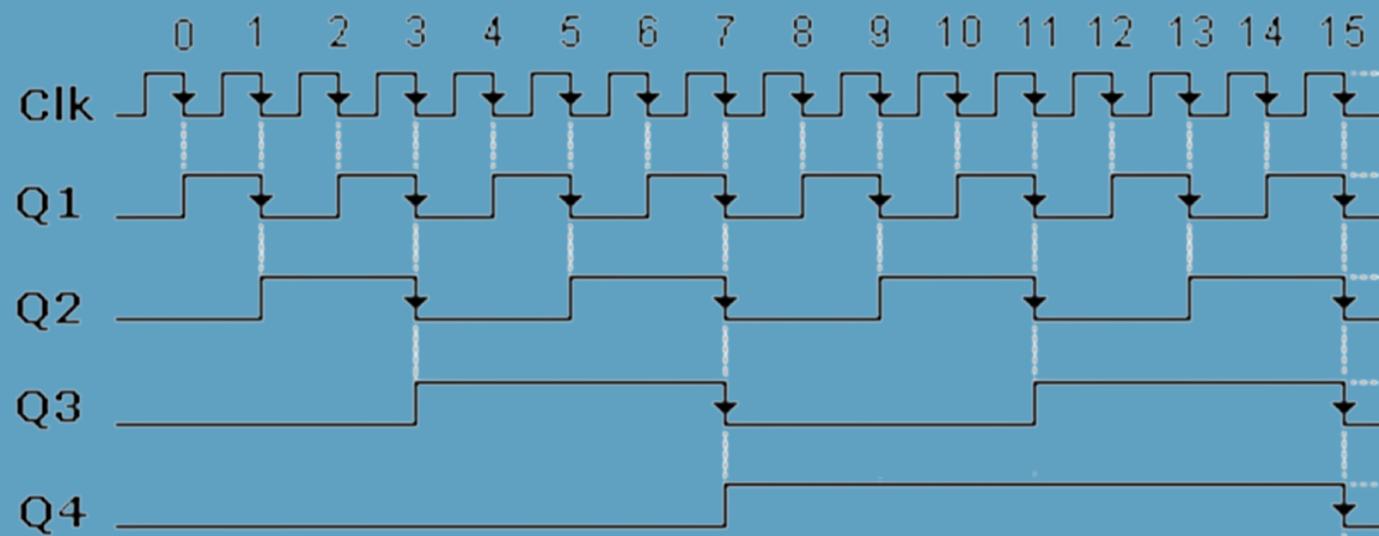
Controlling Pins:

- **CLK = Creating clock pulse.**
- **Cp = Cp pin is used for toggling and incrementing the counter.**
- **CLR' = Clear garbage value.**
- **Ep = Tristate buffer controlling pins. When it will be 1, data will go to BUS from Buffer.**

PROGRAM COUNTER

CHILD SHEET CIRCUIT DIAGRAM

- JK Flipflop Model = 74LS107
- 4 JK Flipflops are used for counting 00-15. Because we used 16x8 ram. That means total 16 bytes data.



- Here 4-bit Asynchronous Counter Method is used. It works on negative edge triggering. Cp pin is used for toggling and incrementing the counting. It can count from 0000 to 1111 (0-15). First output of the first Flipflop is used for the Clock of next Flipflop.
- PC0 – PC7 pins are connected with binary to BCD converter. So that, decimal number can be seen directly on the display.

MEMORY ADDRESS REGISTER

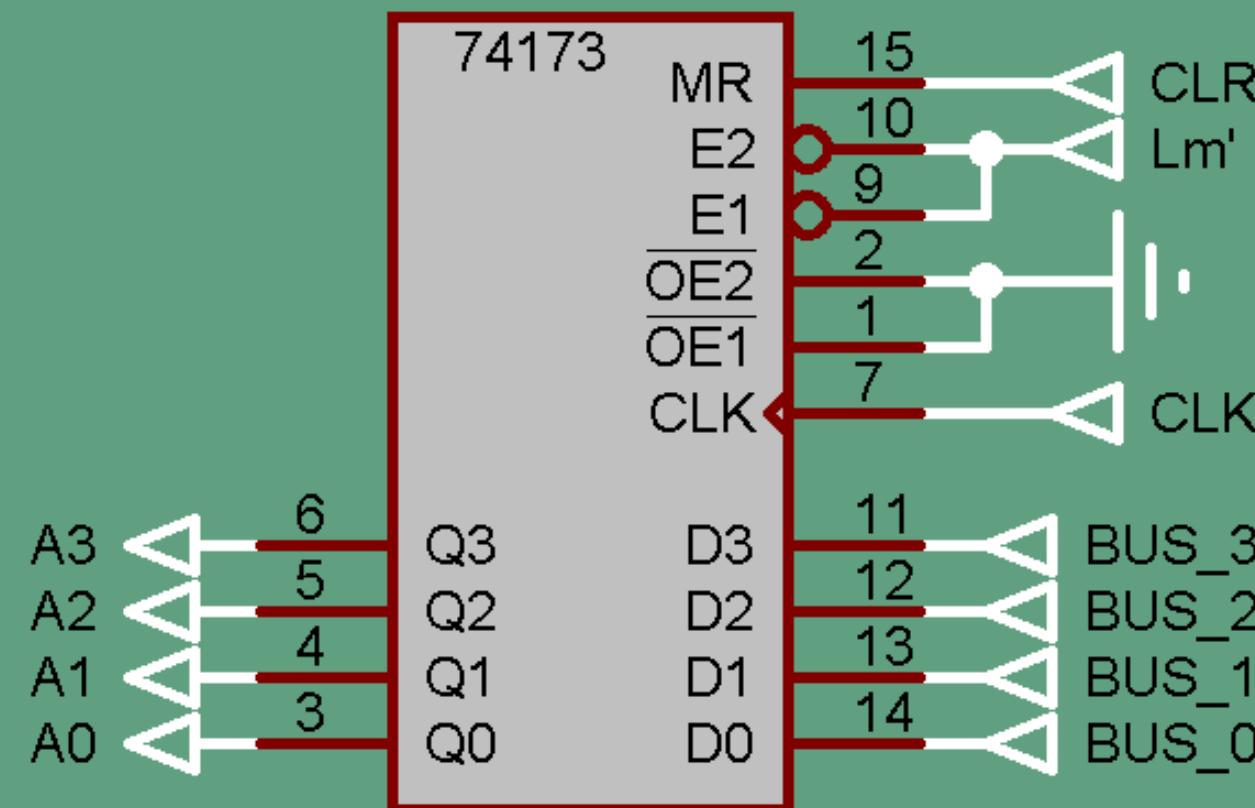


FIG: Block Diagram of MAR

- A D Flipflop is used as MAR. Because an MAR just works with 4-bit of data. And this D Flipflop can handle 4-bit I/O.
- D Flipflop Model = 74173
- OE1', OE2' = Output enables pin. These pins are grounded always.
- E1, E2 = Input enables pin. Connected with Lm'.
- MR = Master Reset Input. Connected with CLR pins.
- CLK' = Creating clock pulse.

I/O pins:

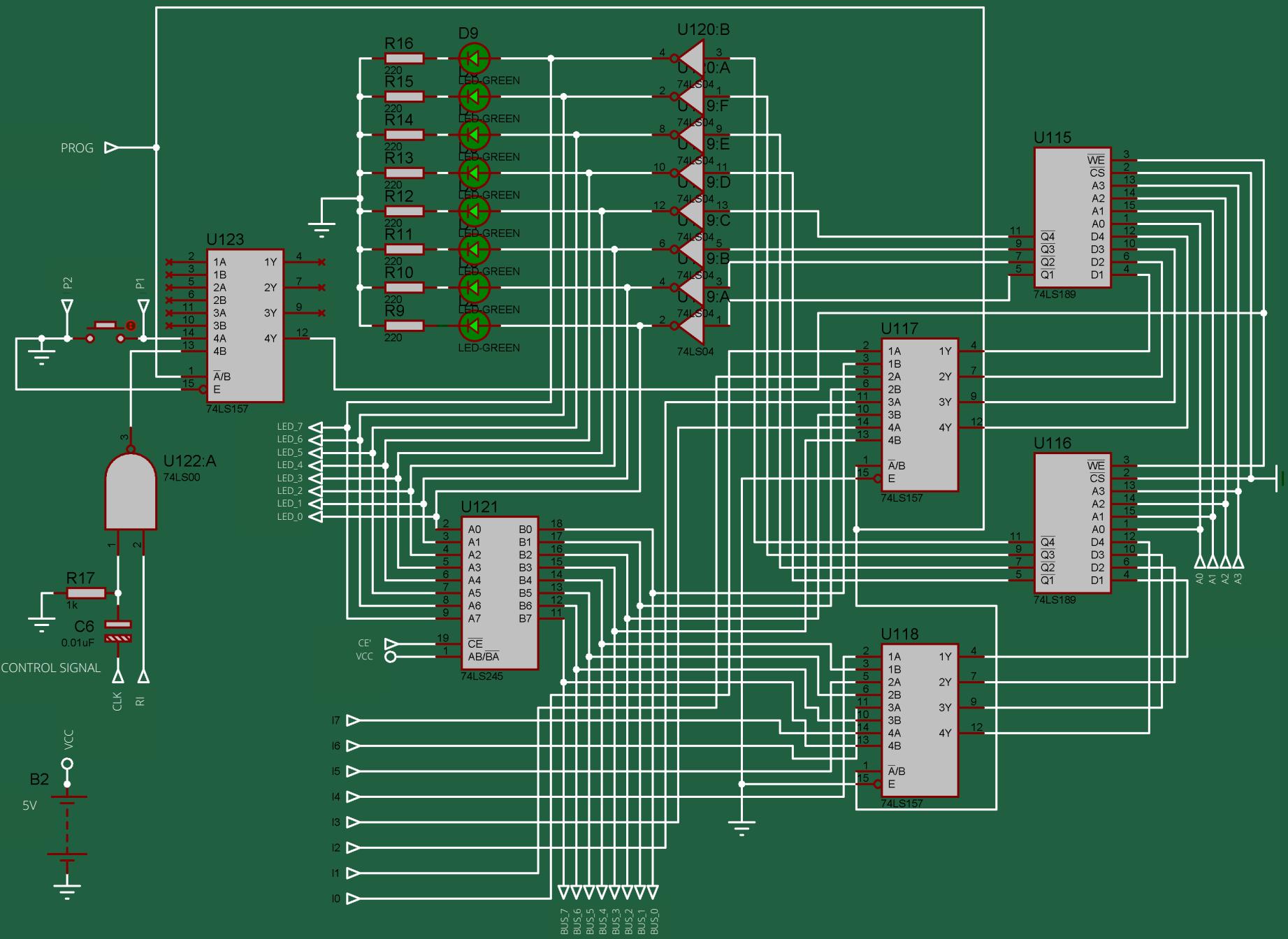
- D0 – D3 = Input pins and connected to the bus. This is controlled by Lm'.
- A0 – A3 = Output pins. 4-bit data directly go to RAM.
- AS1 – AS8 = Input pins. 8-bit data directly comes from Accumulator through these pins.
- Controlling Pins:
- Lm' = When Lm' will be logic 0, data will enter in the MAR through D0 – D3 pins from BUS.

RANDOM ACCESS MEMORY

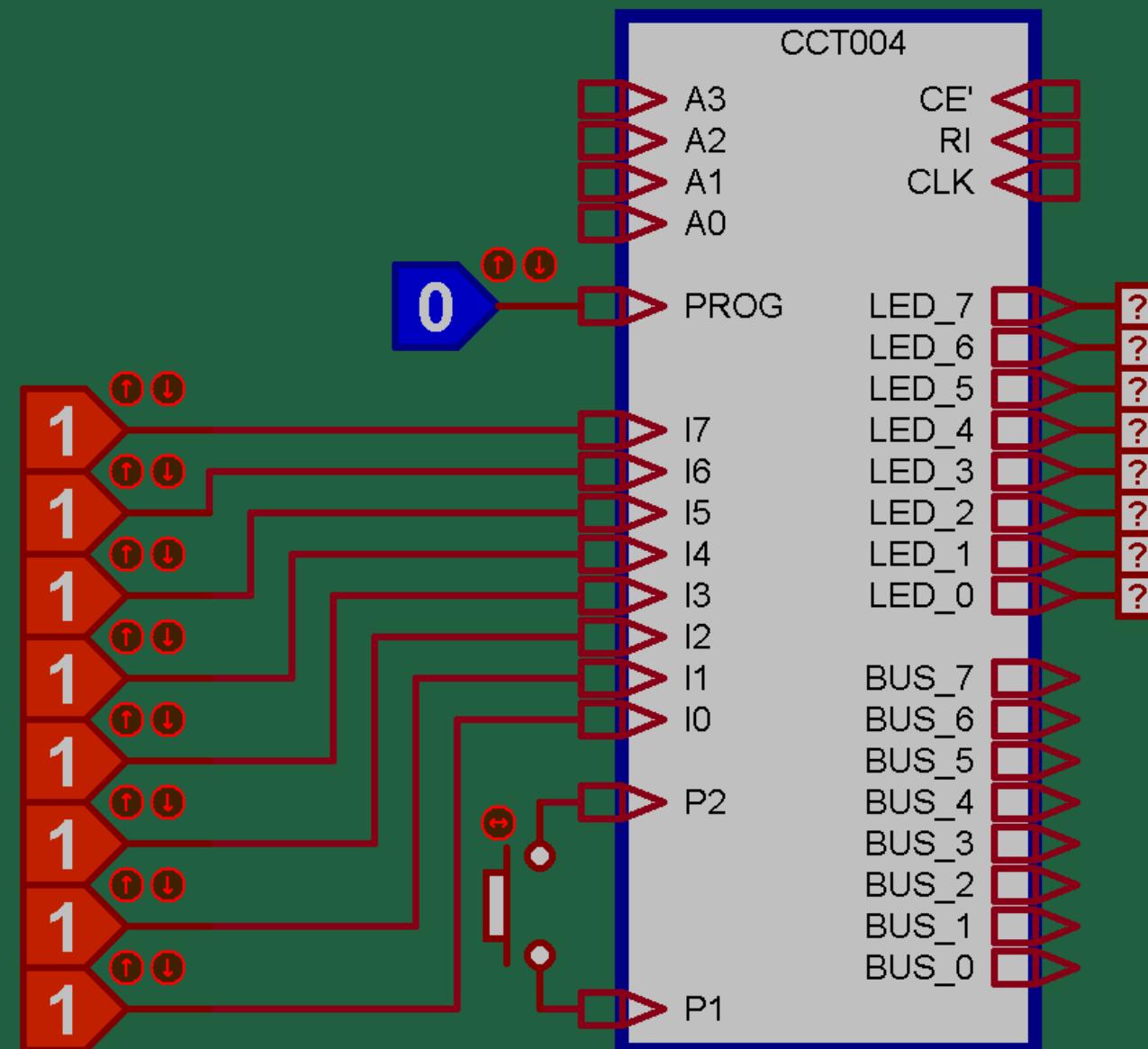
ADDRESS	OPERATION	DATA
0	0000	LOAD 5
1	0001	ADD 6
2	0010	SUB 7
3	0011	OUT
4	0100	HLT
5	0101	STORE 25
6	0110	STORE 05
7	0111	STORE 03
8	1000	DATA STORE
9	1001	DATA STORE
10	1010	DATA STORE
11	1011	DATA STORE
12	1100	DATA STORE
13	1101	DATA STORE
14	1110	DATA STORE
15	1111	DATA STORE

- Address line input A3, A2,A1 & A0. The Memory Address Register (MAR) will provide the address data.
- Data input D3, D2, D1, D0.
- Output Q₃, Q₂, Q₁, Q₀ are inverted. That is why each output is connected to a 74LS04 NOT gate to obtain the actual data.
- 74LS157 data selector/MUX IC is connected to the data input pin. The RAM includes an input pin labeled "PROG." This pin is used to enable Read/Write operation.
- We use the 74LS245 tri-state buffer integrated circuit, which contains eight tri-state buffer circuits. This integrated circuit will generate non-inverting outputs. Each tri-state buffer's input is connected to the RAM's output. The CE' pin of the 74LS245 determines whether the chip is enabled or not.

RAM CIRCUIT DIAGRAM



RAM BLOCK DIAGRAM



INSTRUCTION REGISTER

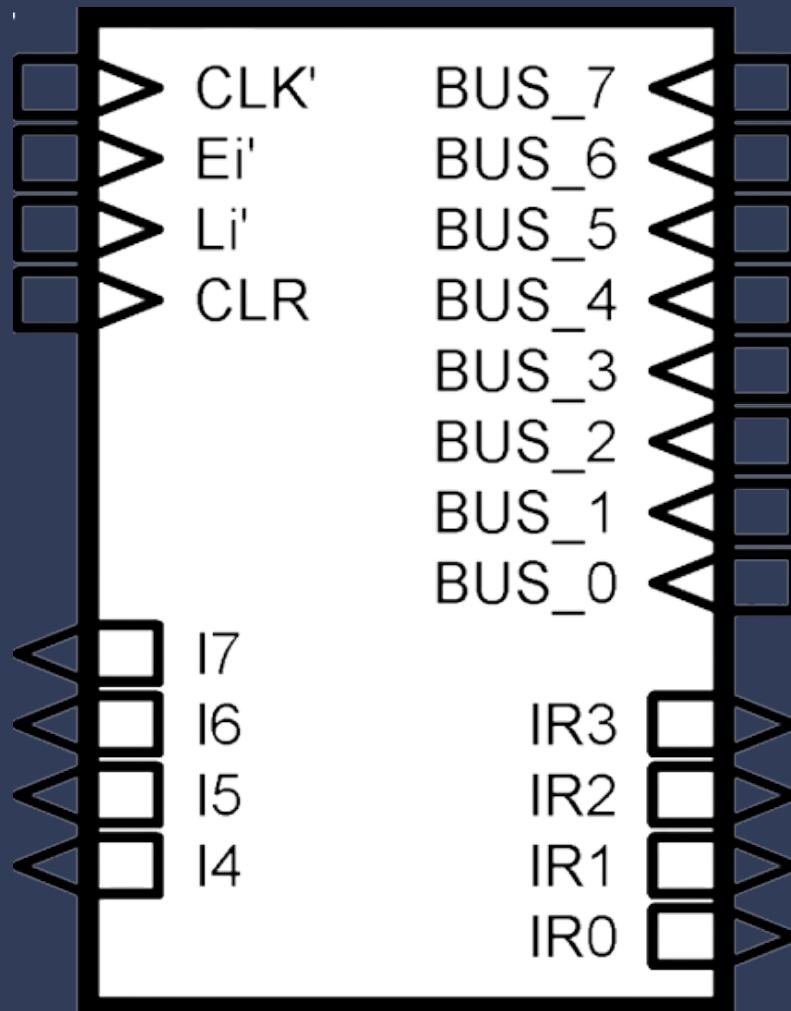


FIG: Block Diagram of
Instruction Register

I/O pins:

- **IR₀ – IR₃ = Output pins. Go to Bus.**
- **I₄ – I₇ = Output pins. Go to Controller Sequence.**
- **BUS₀ – BUS₇ = Input pins. 8-bit data come from BUS. First 4-bit (BUS₀₄– BUS₇) data is sent to Controller Sequencer. Next 4-bit (BUS₀– BUS₃) data (address location) is sent to BUS.**

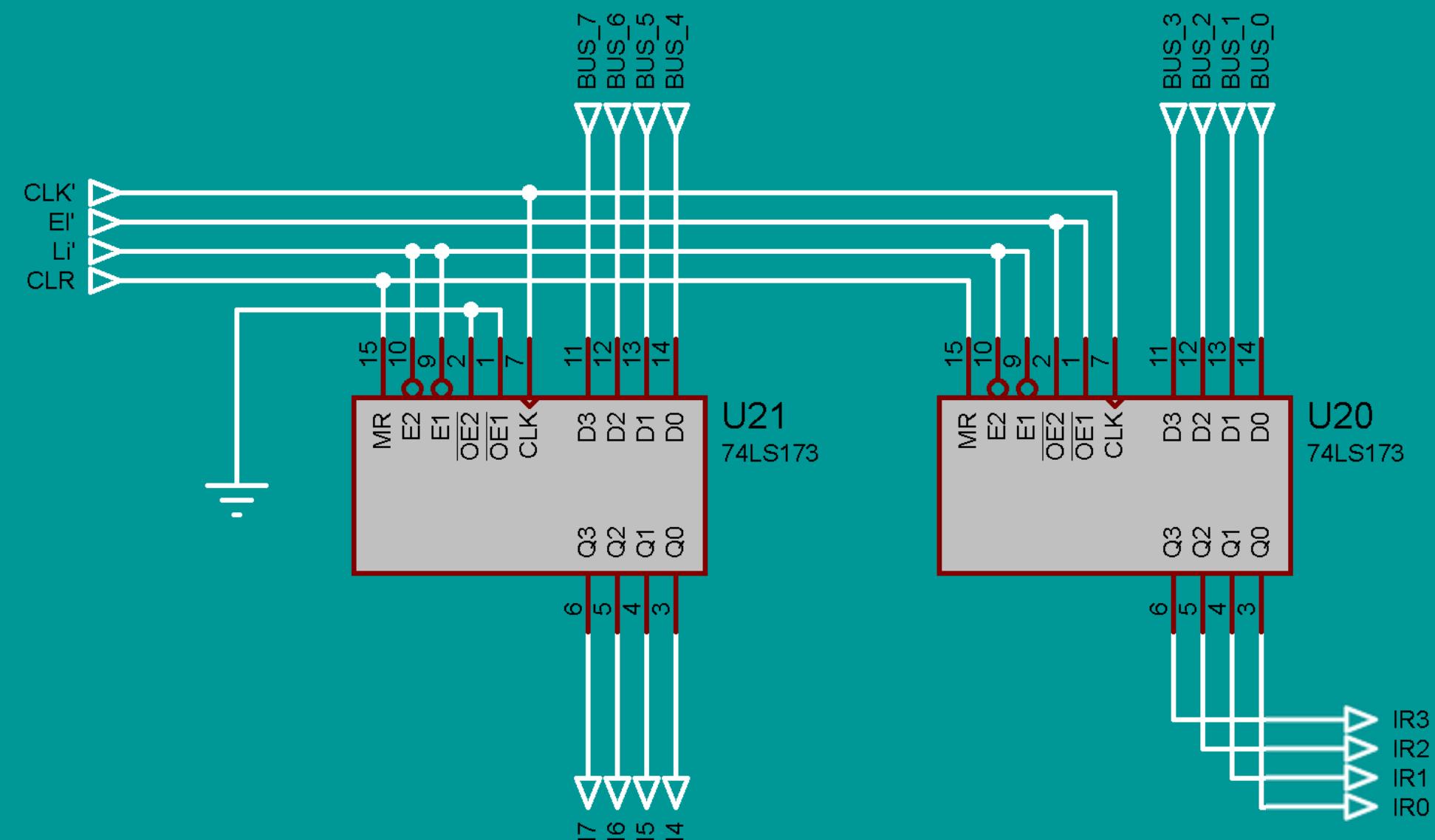
Controlling Pins:

- **Li' = When Li' will be logic 0, data will enter in the Instructions Register from BUS.**
- **CLK' = Creating clock pulse.**
- **Ei' = When Ei' will be logic 0, data will enter in the BUS through IR₀ – IR₃.**
- **CLR = Clear garbage value.**

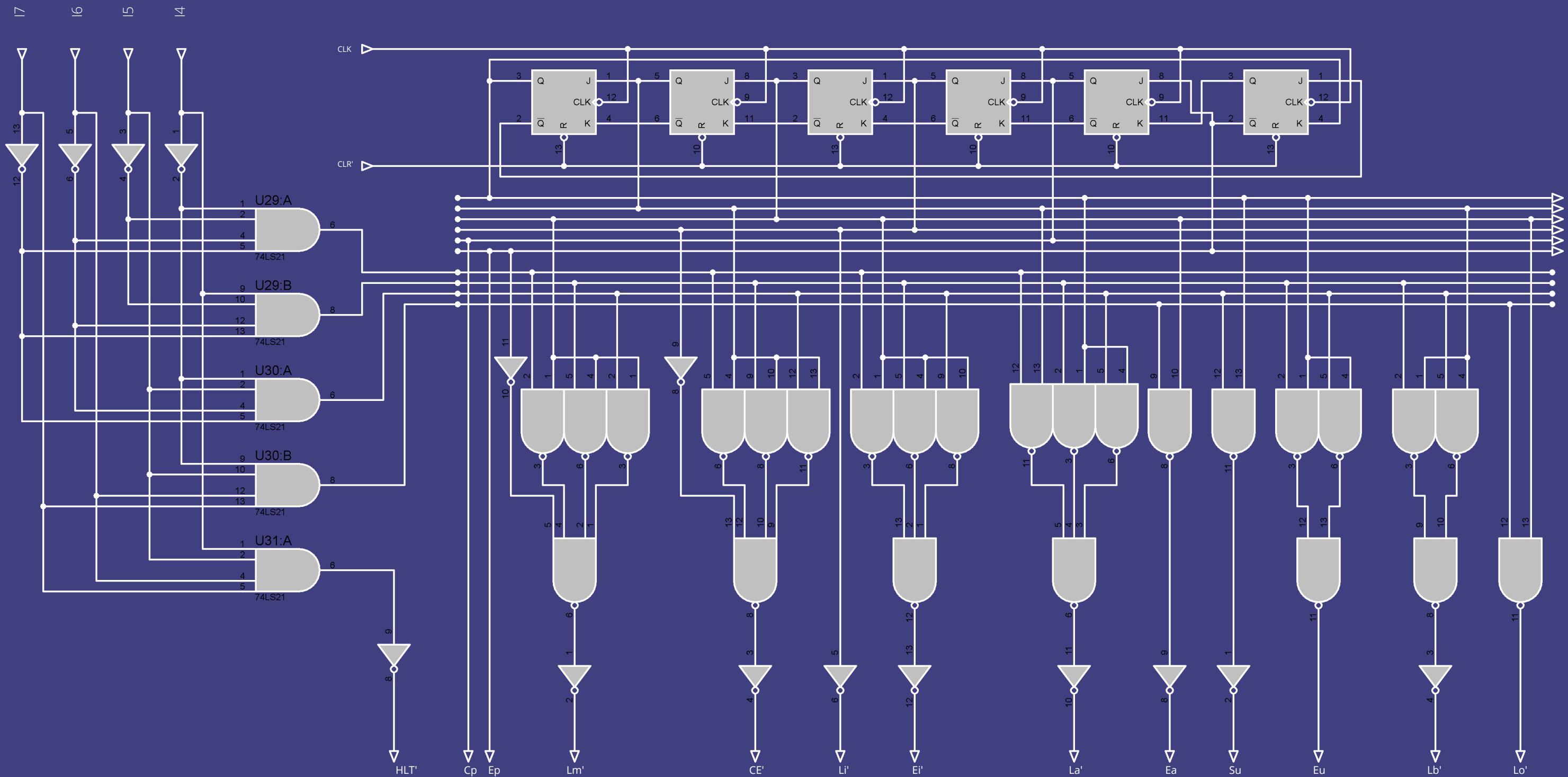
INSTRUCTION REGISTER

- D Flipflop Model = 74LS173
- Two D Flipflops have been used for 8 bit I/O.
- D0 -D3 = Input pins. Q0 -Q3 = Output pins.
- OE1', OE2' = Output enables pin. One of the D flipflop pins are grounded always another one is connected with Ei'.
- E1, E2 = Input enables pin. Connected with Li'.
- MR = Master Reset Input. Connected with CLR pins.
- First 4-bit (BUS_04– BUS_7) data is sent to Controller Sequencer through I4 – I7 pins.
- Next 4-bit (BUS_0– BUS_3) data (address location) is sent to BUS through IR0 – IR3 pins.

CHILD SHEET CIRCUIT DIAGRAM



CONTROL SEQUENCER



CONTROL SEQUENCER- PART 1

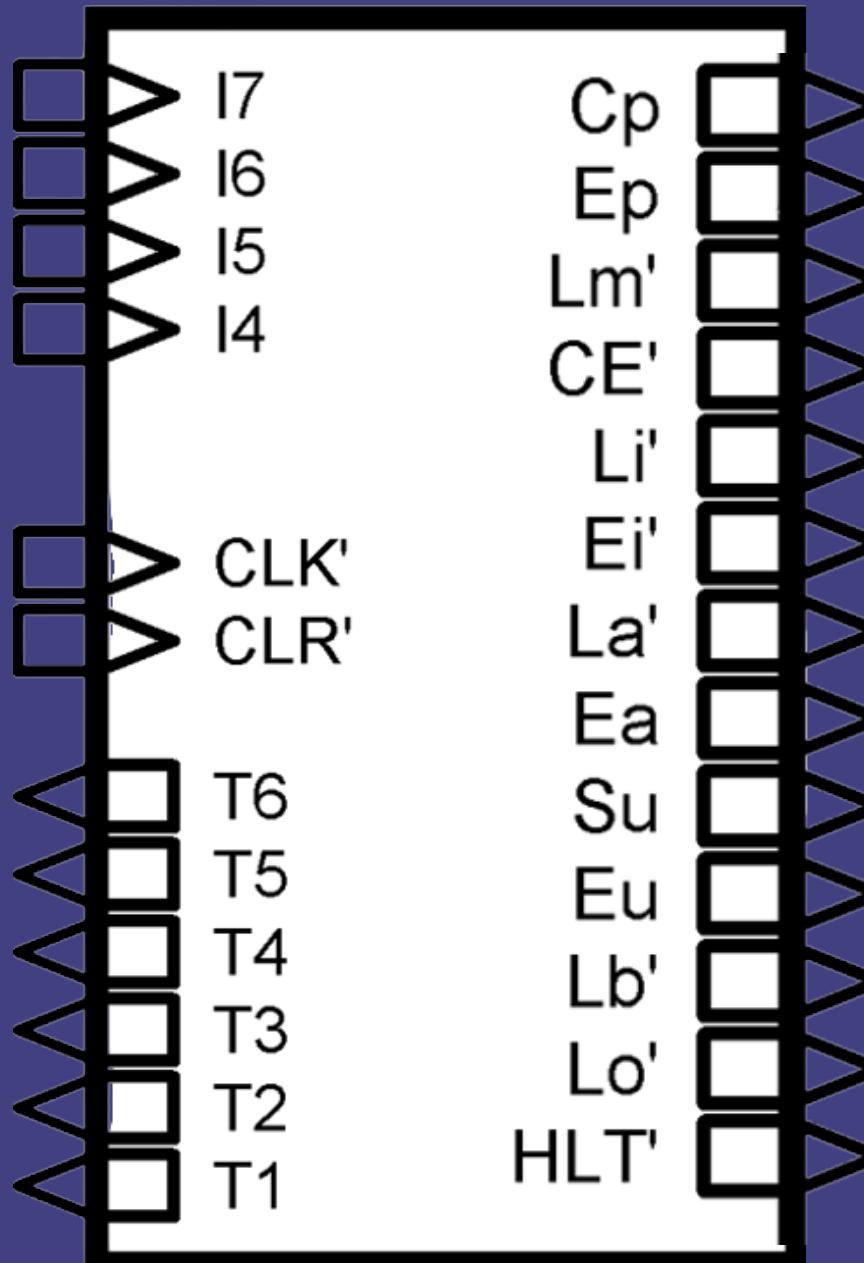


FIG: Block Diagram of
Control Sequencer

I/O pins:

T₁-T₆ = Output pins. Goes to T-state sub circuit.

I₄ – I₇ = Input pins. Comes from Instruction Register.

Other output pins = Cp, Ep, Lm', CE', Li', Ei', La, Ea, Su, Eu, Lb', Lo, HLT'. These pins go to their respective sub circuit module and control them accordingly.

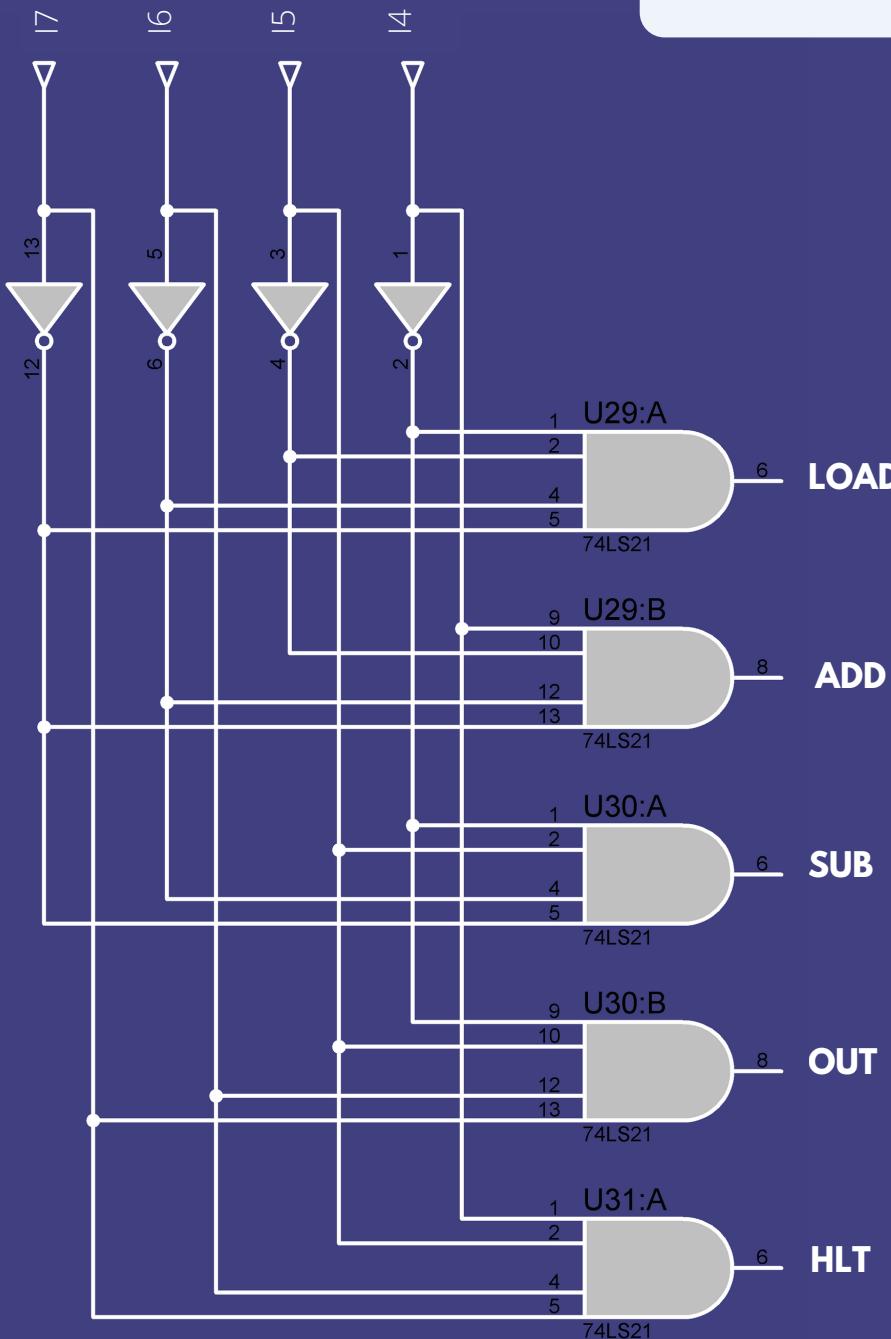
Controlling Pins:

CLK' = Creating clock pulse.

CLR' = Clear garbage value. It is connected with reset pin of JK Flipflops.

CONTROL SEQUENCER PART-2

UNDERSTANDING THE CONTROL SEQUENCER (OPCODE SELECTION)



GATE OUTPUT	OPCODE	OPERATION
$\overline{I_7} \overline{I_6} \overline{I_5} \overline{I_4}$	0000	LOAD
$\overline{I_7} \overline{I_6} \overline{I_5} I_4$	0001	ADD
$I_7 \overline{I_6} I_5 \overline{I_4}$	0010	SUB
$I_7 I_6 \overline{I_5} I_4$	1110	OUT
$I_7 I_6 I_5 I_4$	1111	HLT

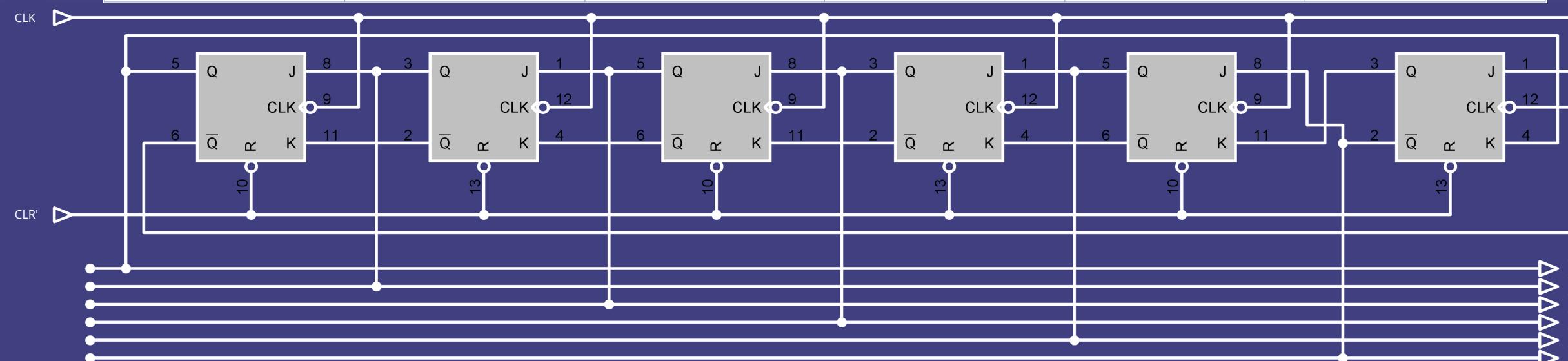
CONTROL SEQUENCER PART-3

UNDERSTANDING THE CONTROL SEQUENCER (T-STATE GENERATION & RING COUNTER)

T-States are basically generated with the help of a Ring Counter which basically made of 6 JK Flipflops. For any time instance only one T states gets high. Rest of them get low.

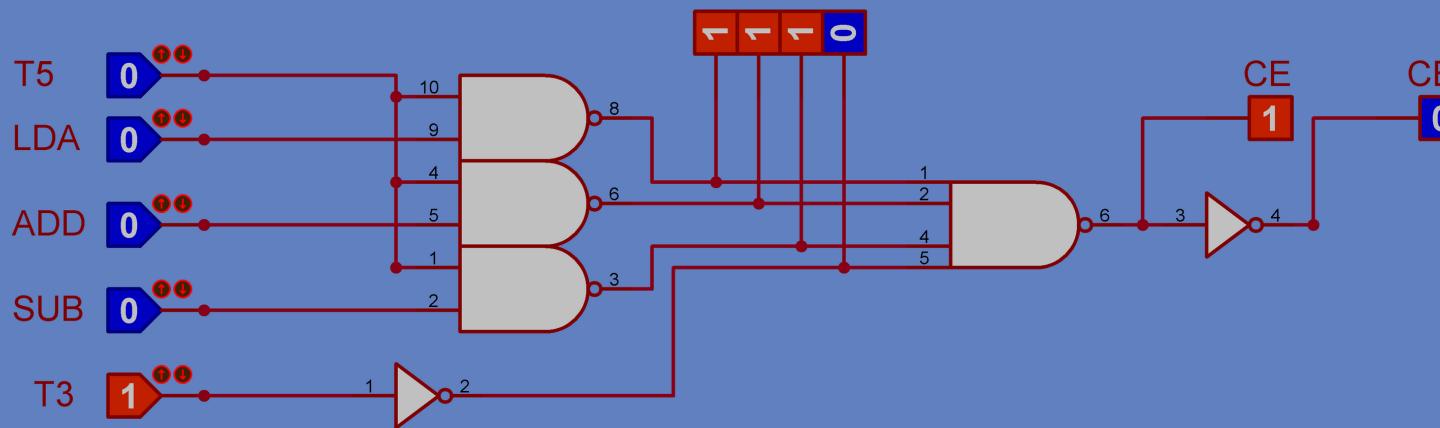
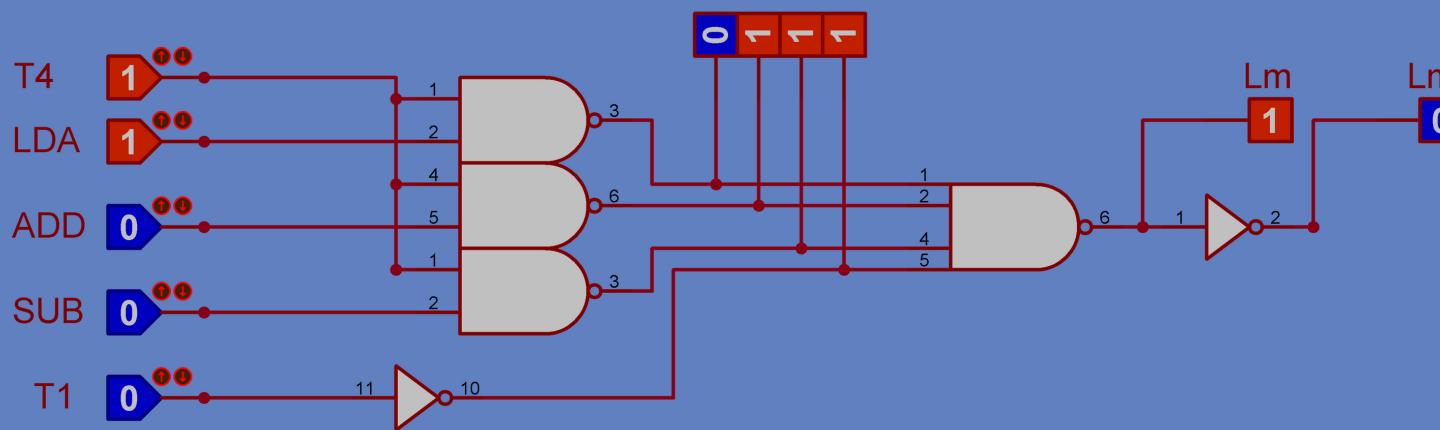
TRUTH TABLE

T1	T2	T3	T4	T5	T6
1	0	0	0	0	0
0	1	0	0	0	0
0	0	1	0	0	0
0	0	0	1	0	0
0	0	0	0	1	0
0	0	0	0	0	1



CONTROL SEQUENCER PART-4

UNDERSTANDING THE CONTROL SEQUENCER (HOW CONTROLLING PINS WORK)



T3 CP

T-STATE



FIG: Block Diagram of T-STATE

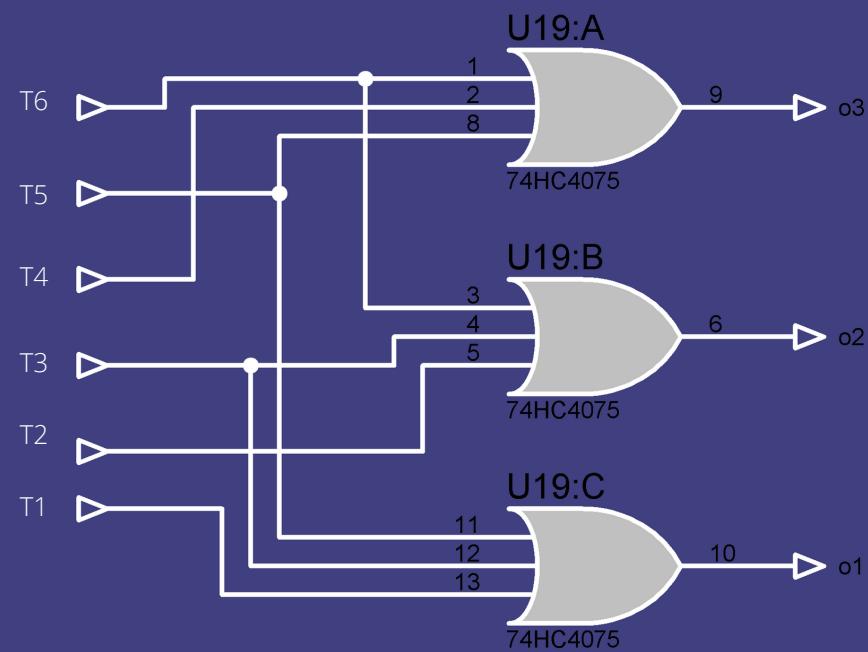


FIG: Circuit Diagram of T-STATE

I/O pins:

- **T1-T6 = Input pins.** These pins come from Controller Sequencer.
- **o1-o3 = Output pins and connected to 7 SEG-BCD.**

TRUTH TABLE

States						Binary Value		Corresponding Decimal	
T1	T2	T3	T4	T5	T6	o3	o2	o1	
1	0	0	0	0	0	0	0	1	1
0	1	0	0	0	0	0	1	0	2
0	0	1	0	0	0	0	1	1	3
0	0	0	1	0	0	1	0	0	4
0	0	0	0	1	0	1	0	1	5
0	0	0	0	0	1	1	1	0	6

ACCUMULATOR

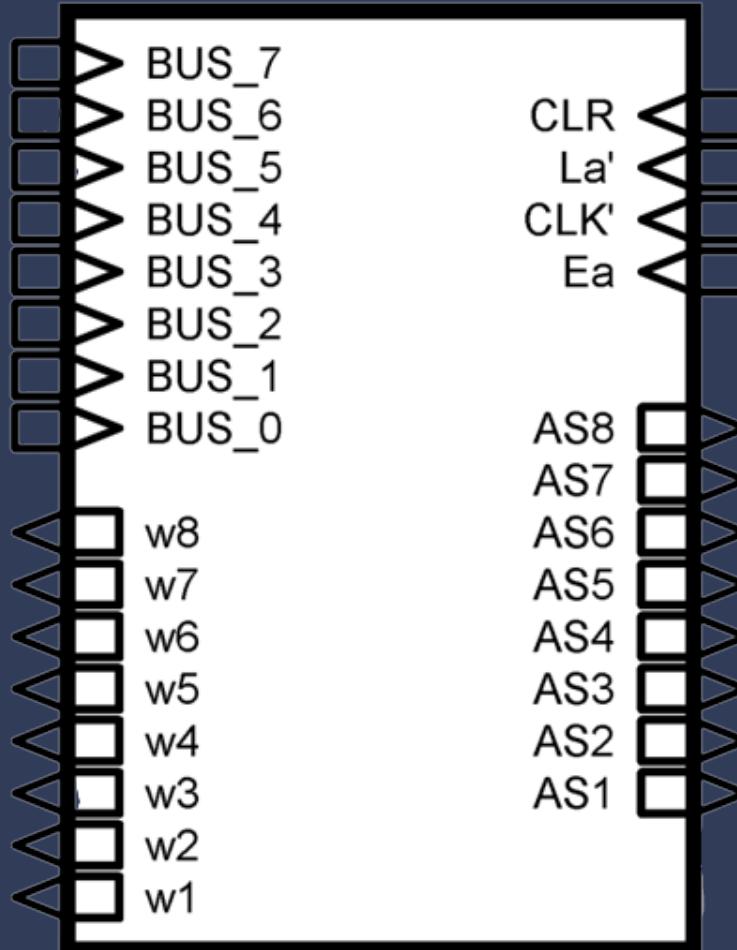


FIG: Block Diagram of
Accumulator

I/O pins:

- **BUS_0 – BUS_7 = Input pins and connected to the bus. This is controlled by La'.**
- **w1-w8 = Output pins and connected to the Bus. Automatically values from B Register go to ALU.**
- **AS1- AS8 = Output pins and connected to the ALU. Automatically values from Accumulator go to ALU.**

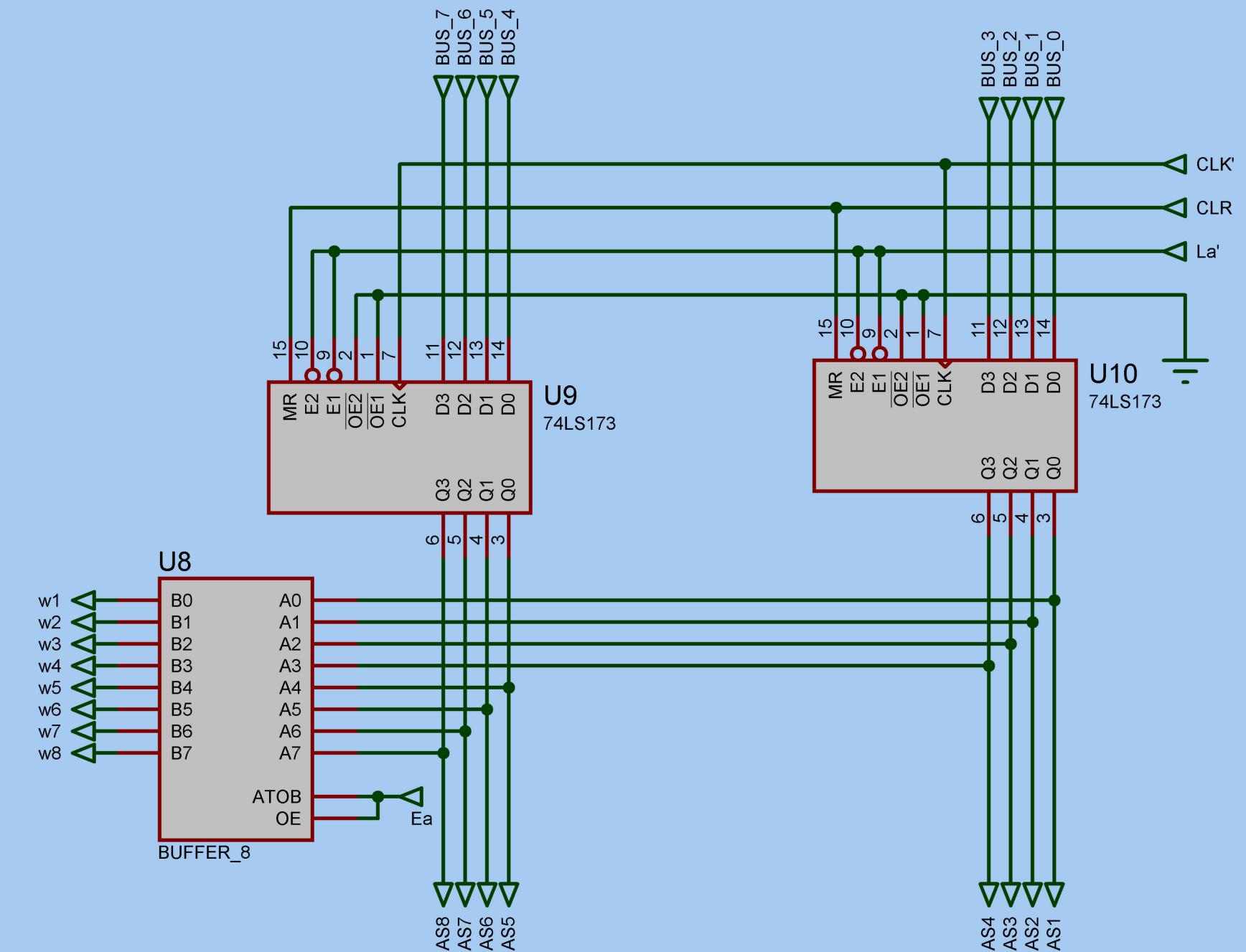
Controlling Pins:

- **La' = When La' will be logic 0, data from bus will enter in the Accumulator.**
- **Ea = This pin actually controls the buffer. When Ea will be logic 1, data from Accumulator will enter in the Bus through w1 – w8 pins.**
- **CLK' = Creating clock pulse.**
- **CLR = Clearing garbage value.**

ACCUMULATOR

CHILD SHEET CIRCUIT DIAGRAM

- D Flipflop Model = 74LS173
- Two D Flipflops have been used for 8 bit I/O.
- D0 -D3 = Input pins. Q0 -Q3 = Output pins.
- OE1', OE2' = Output enables pin. These pins are grounded always.
- E1, E2 = Input enables pin. Connected with La'.
- MR = Master Reset Input. Connected with CLR pins.
- CLK = Connected with CLK pin.



B-REGISTER

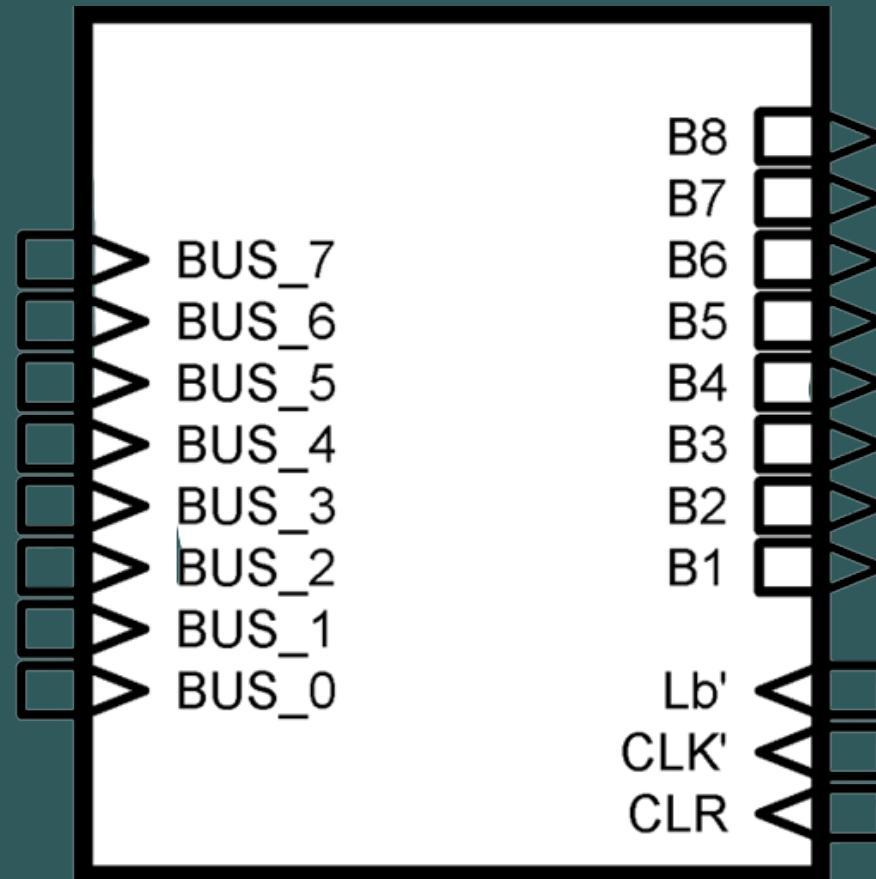


FIG: Block Diagram of
B-Register

I/O pins:

- **BUS_0 – BUS_7 = Input pins and connected to the bus. This is controlled by Lb'.**
- **B1 - B8 = Output pins and connected to the Arithmetic Logic Unit. Automatically values from B Register go to ALU.**

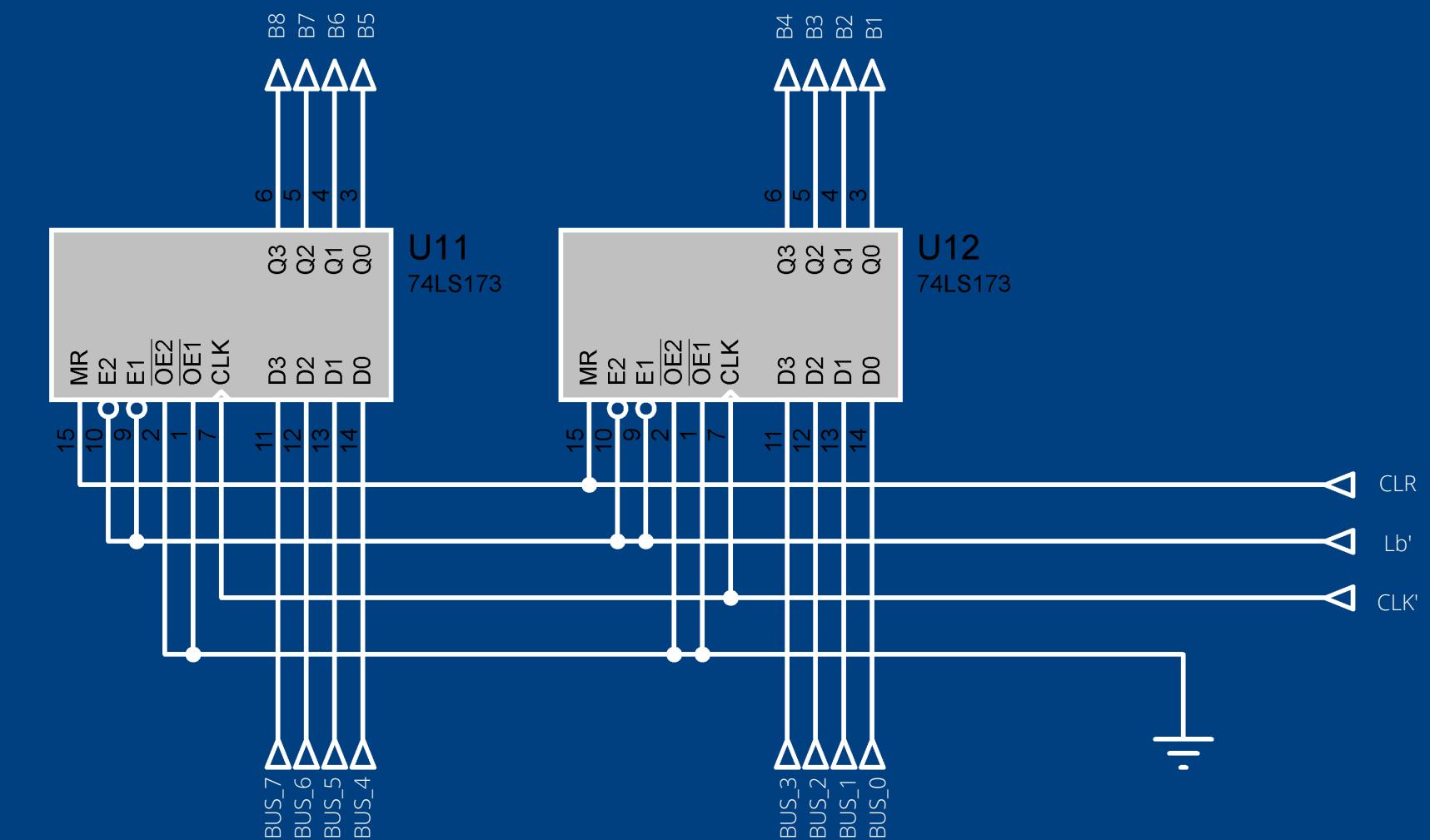
Controlling Pins:

- **Lb' = When Lb' will be logic 0, data from bus will enter in the B Register.**
- **CLK' = Creating clock pulse.**
- **CLR = Clearing garbage value.**

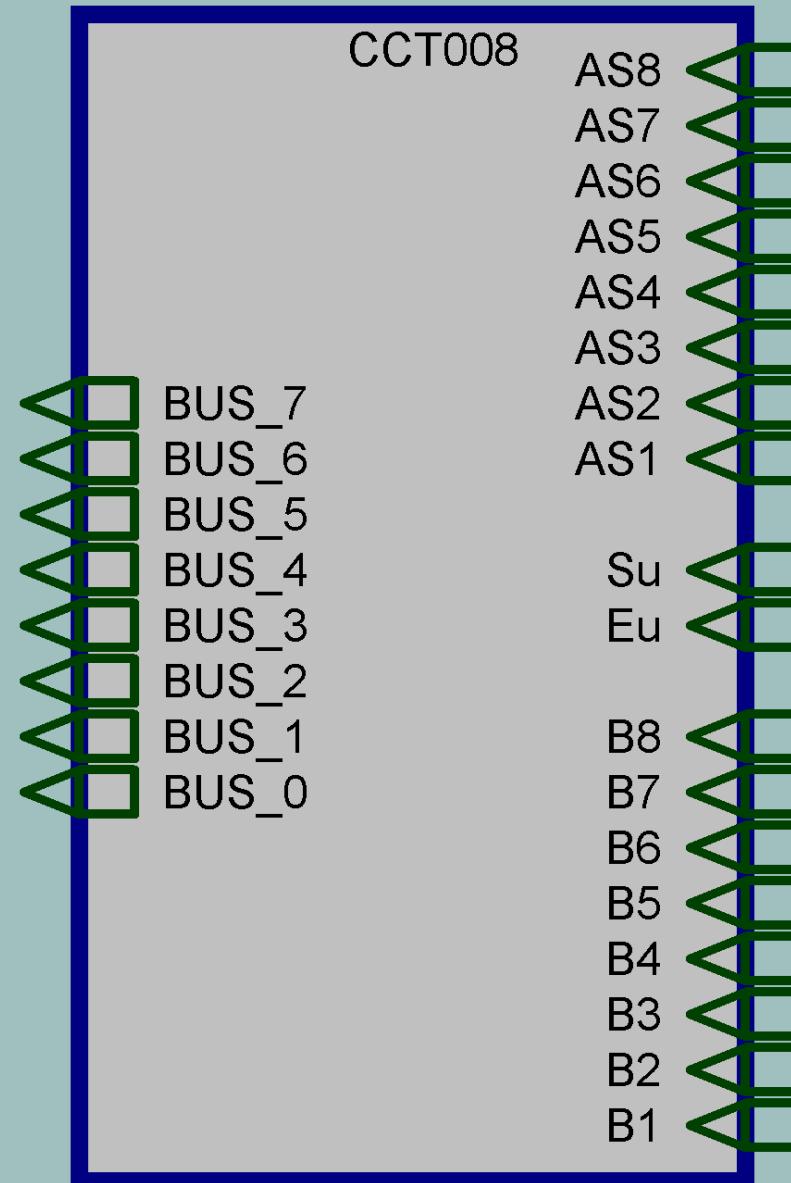
B-REGISTER

- D Flipflop Model = 74LS173
- Two D Flipflops have been used for 8 bit i/o.
- D0 -D3 = Input pins. Q0 -Q3 = Output pins.
- OE1', OE2' = Output enables pin. These pins are grounded always.
- E1, E2 = Input enables pin. Connected with Lb'.
- MR = Master Reset Input. Connected with CLR pins.

CHILD SHEET CIRCUIT DIAGRAM



ARITHMETIC AND LOGIC UNIT



I/O pins:

- **BUS_0 – BUS_7** = Output pins and connected to the bus. This is controlled by Eu.
- **B1 - B8** = Input pins. 8-bit data directly comes from B Register through these pins.
- **AS1 – AS8** = Input pins. 8-bit data directly comes from Accumulator through these pins.

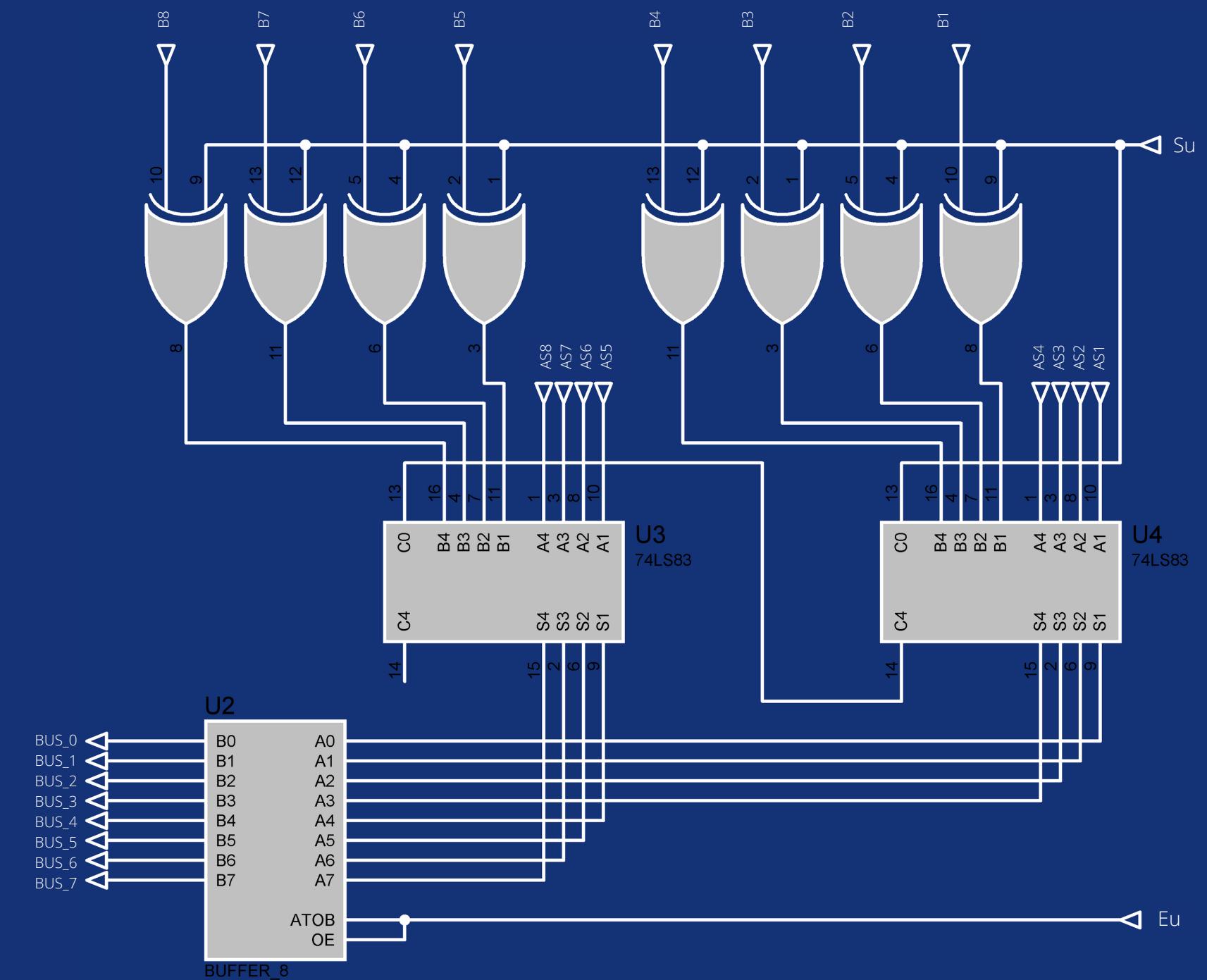
Controlling Pins:

- **SU** = When SU will be logic 0, addition will happen. When 1, subtraction will occur.
- **EU** = When EU will be logic 1, output pins will be active. And stored buffer data will go to the bus.

EU Pin (Logic State)	Operation Name	Explanation
0	Addition	We know, 0 XOR B1= B1. Then B1 + AS1.
1	Subtraction	We know, 1 XOR B1= B1'. Then AS1 – B1 = AS1 + B' + 1

ARITHMETIC AND LOGIC UNIT

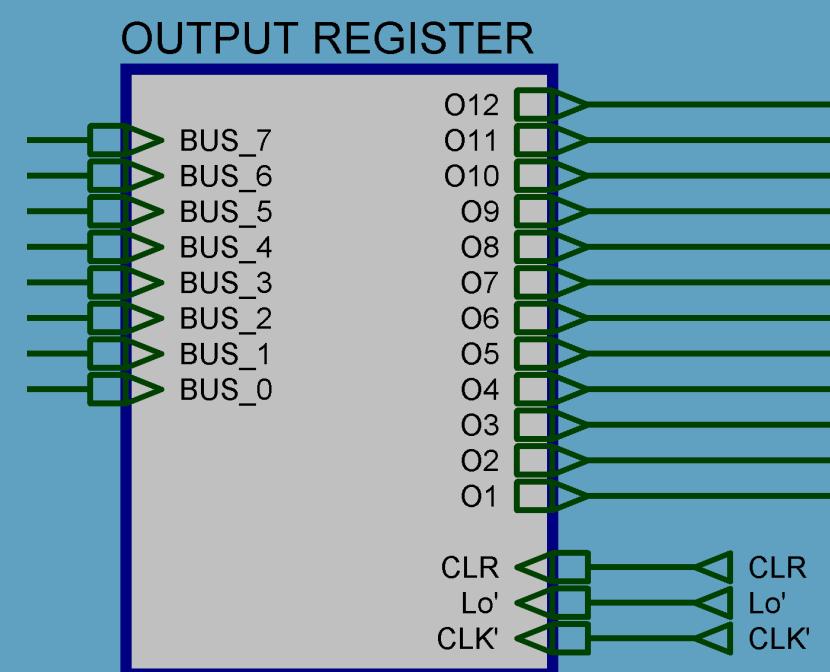
- Adder Model = 74LS83
- Two adders are used for 8 bit add/sub.
- A1- A4 = Input pins. Data coming from accumulator.
- B1 -B4 = Output pins. Data coming from B Register through XOR gate.
- C0 = Carry bit handler pin.



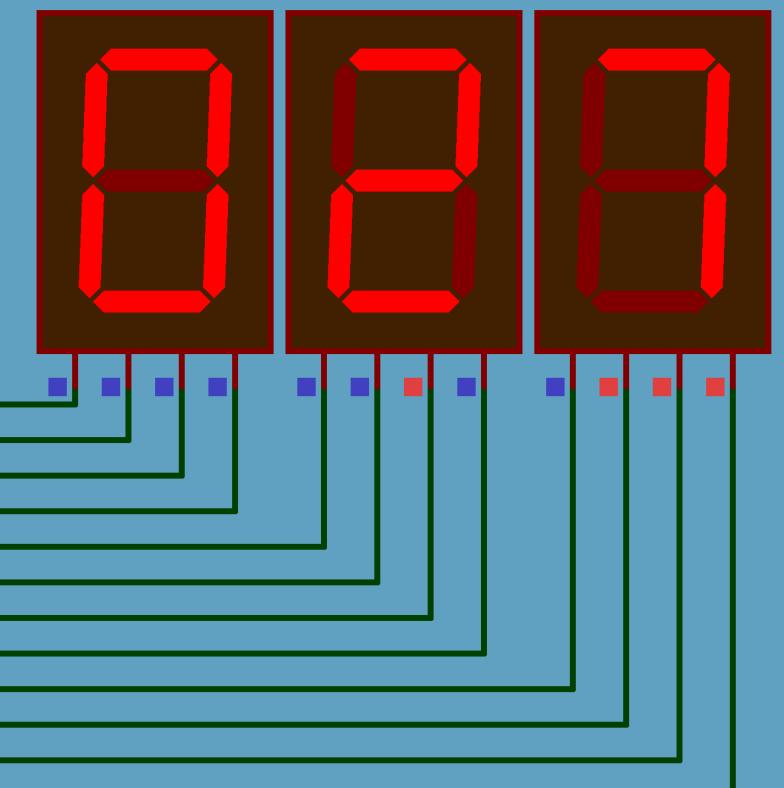
OUTPUT REGISTER WITH DISPLAY

- D Flipflop Model = 74LS173
- Two D Flipflops have been used for 8 bit i/o.
- D0 -D3 = Input pins. Q0 -Q3 = Output pins.
- OE1', OE2' = Output enables pin. These pins are grounded always.
- E1, E2 = Input enables pin. Connected with Lo'.
- MR = Master Reset Input. Connected with CLR pins.
- The "8-bit binary to BCD converter" will convert the 8-bit output to BCD. The display will show the decimal value of the output.

BLOCK DIAGRAM

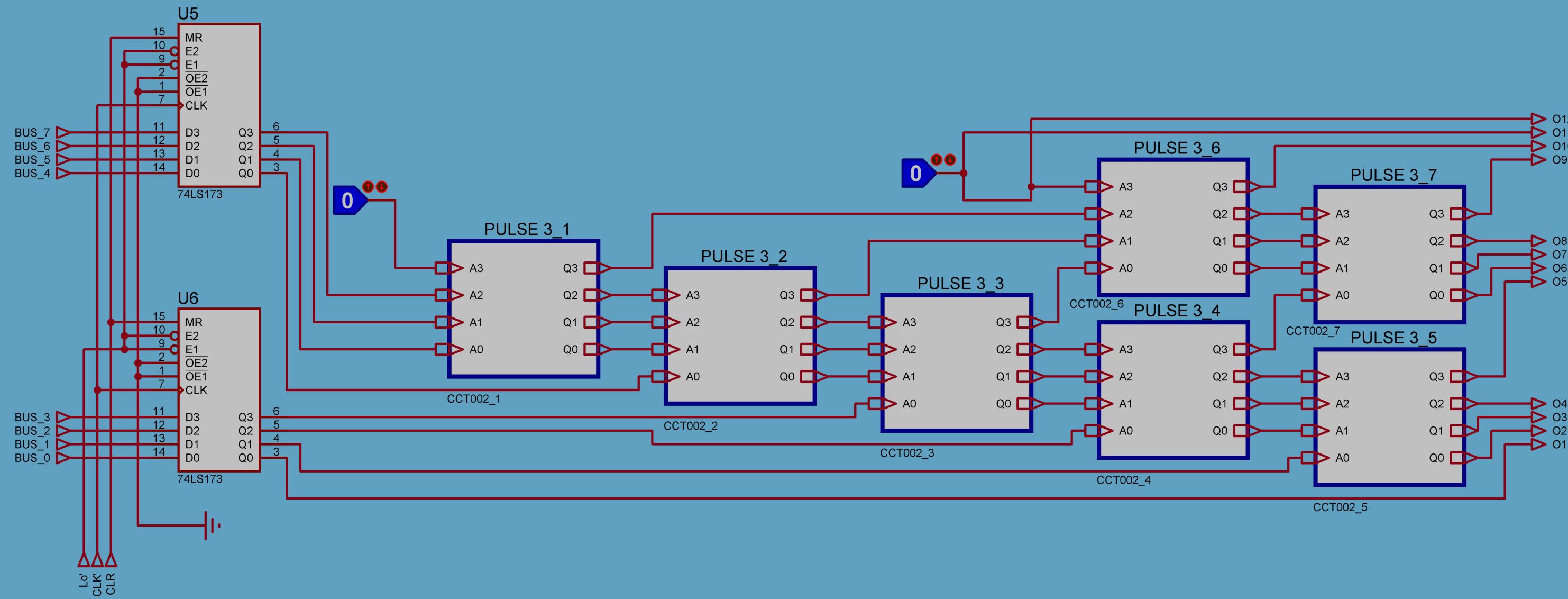


OUTPUT RESULT



OUTPUT REGISTER WITH DISPLAY

CIRCUIT DIAGRAM



CONVERTER

SHIFT 3 FOR BINARY TO BCD CONVERSION

The "Shift 3" circuit is based on a double dabbling algorithm. If the binary value is greater than 4, this circuit ignores it. Otherwise, the binary input is added by 3. Invalid input values are presumed to be between 10 and 15. However, it doesn't really matter what happens at that place. It's fine to add 3 in certain situations.

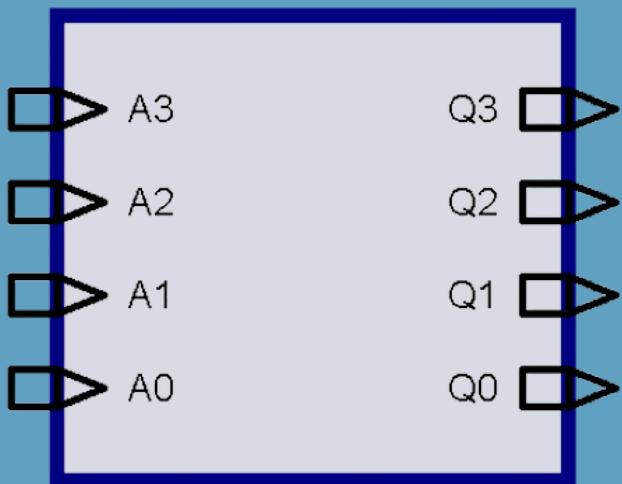
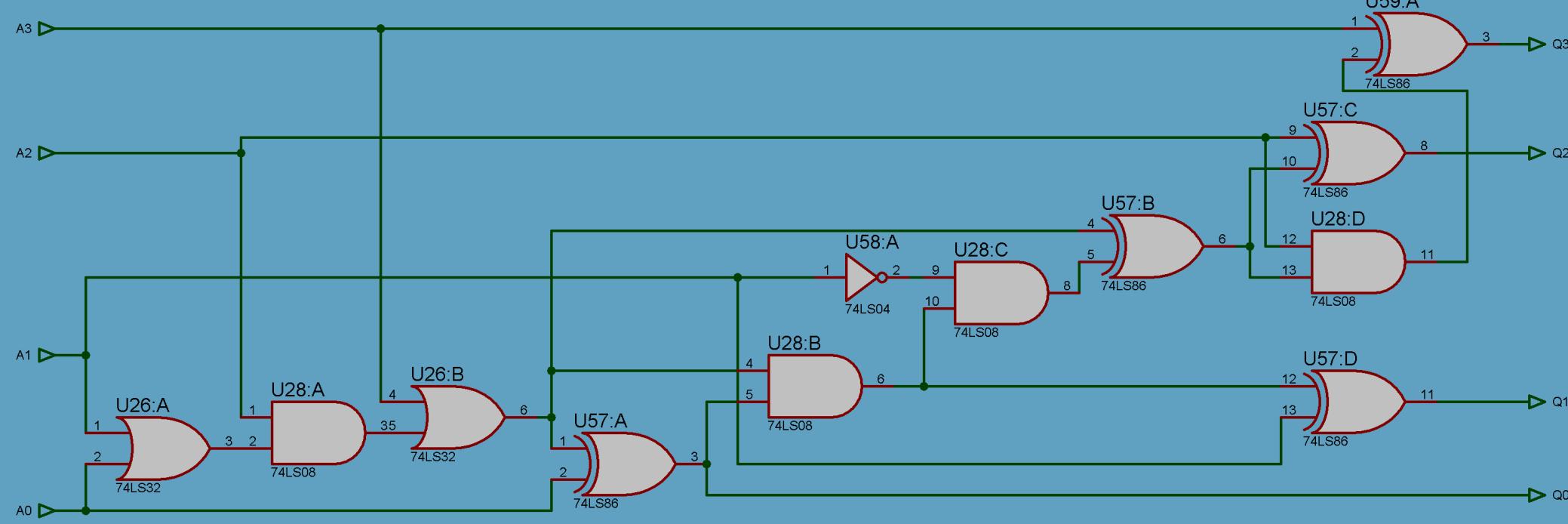


FIG: Block Diagram of Shift 3 Circuit

TRUTH TABLE FOR SHIFT 3 CIRCUIT

A3	A2	A1	A0	Q3	Q2	Q1	Q0
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	0
0	0	1	1	0	0	1	1
0	1	0	0	0	1	0	0
0	1	0	1	1	0	0	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	1	0
1	0	0	0	1	0	1	1
1	0	0	1	1	1	0	0
1	0	1	0	X	X	X	X
1	0	1	1	X	X	X	X
1	1	0	0	X	X	X	X
1	1	0	1	X	X	X	X
1	1	1	0	X	X	X	X
1	1	1	1	X	X	X	X

CONVERTER

BINARY TO BCD CONVERSION

FOR 4-BIT BINARY

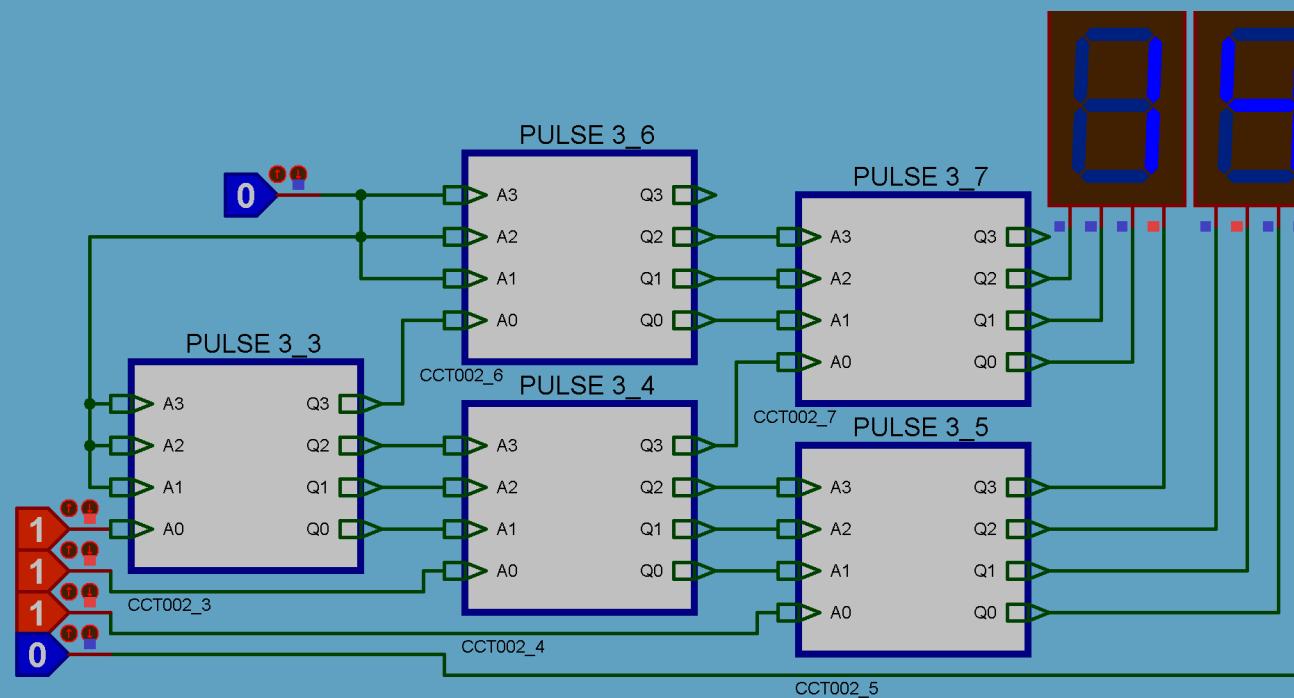
OPERATION	TENS	UNITS	BINARY
START			1110
SHIFT 1		1	110
SHIFT 2		11	10
SHIFT 3		111	0
ADD 3		1010	0
SHIFT 4	1	0100	
BCD	1	4	

FOR 8-BIT BINARY

OPERATION	HUNDREDS	TENS	UNITS	BINARY
START				1111 1111
SHIFT 1				1 1111 111
SHIFT 2				11 1111 11
SHIFT 3				111 1111 1
ADD 3				1010 1111 1
SHIFT 4			1	0101 1111
ADD 3			1	1000 1111
SHIFT 5			11	0001 111
SHIFT 6			110	0011 11
ADD 3			1001	0011 11
SHIFT 7		1	0010	0111 1
ADD 3		1	0010	1010 1
SHIFT 8	10	0101	0101	
BCD	2	5	5	

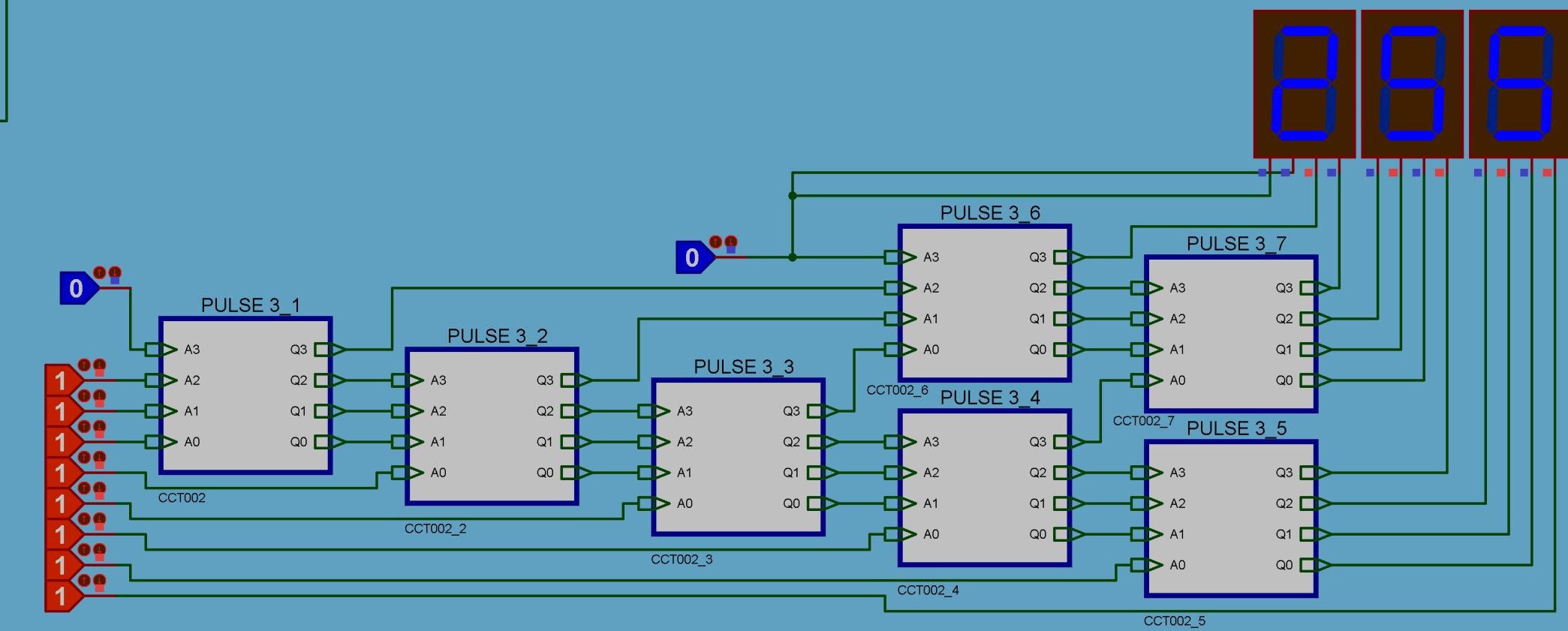
CONVERTER

BINARY TO BCD CONVERSION



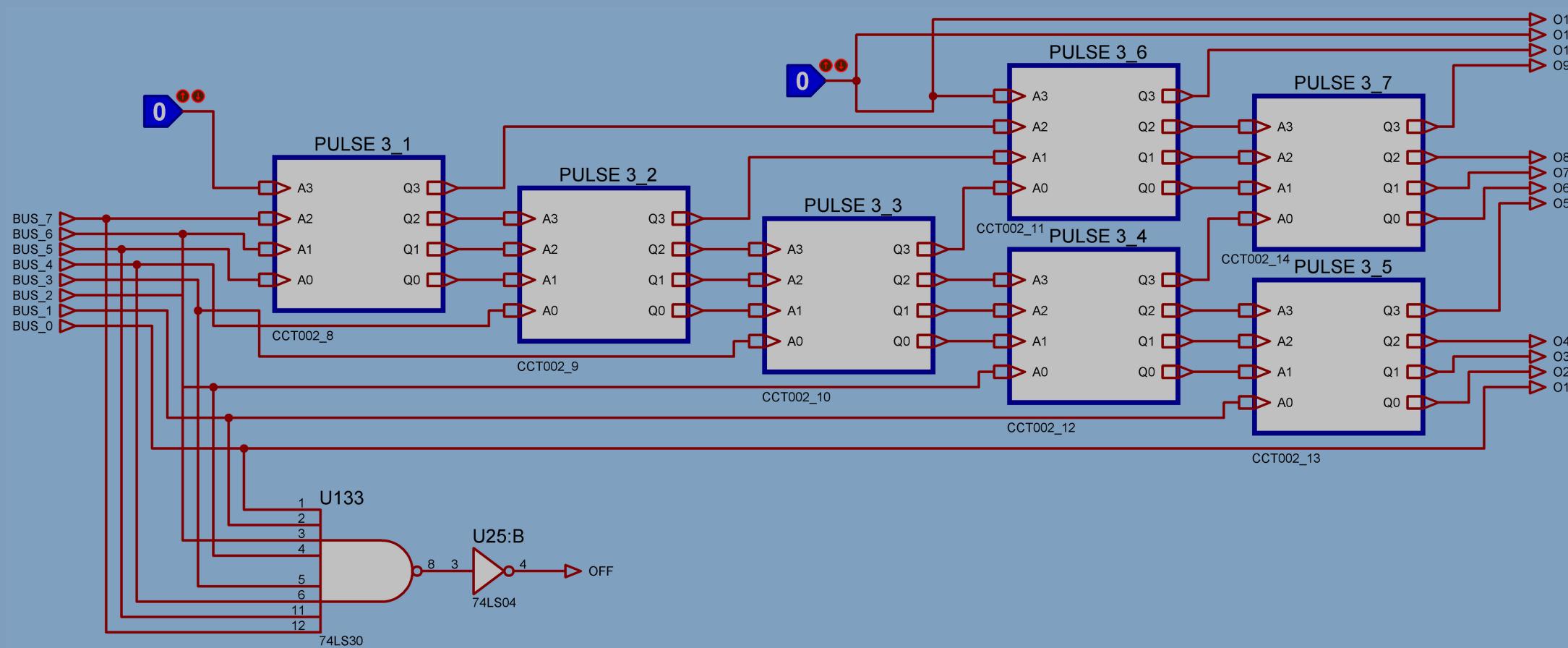
8-BIT BINARY TO BCD

4-BIT BINARY TO BCD

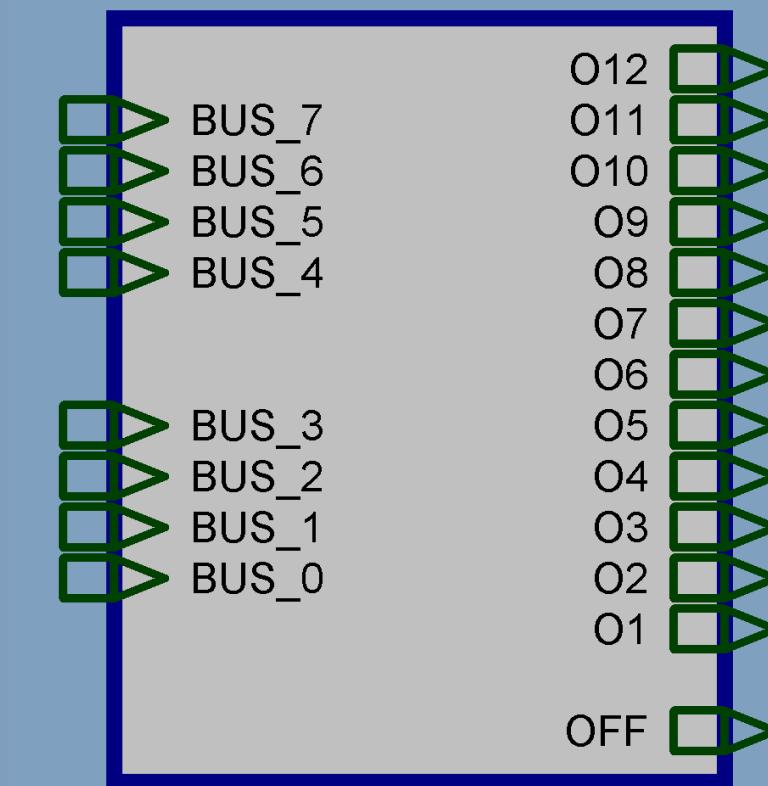


CONVERTER

CONVERTER CIRCUIT DIAGRAM

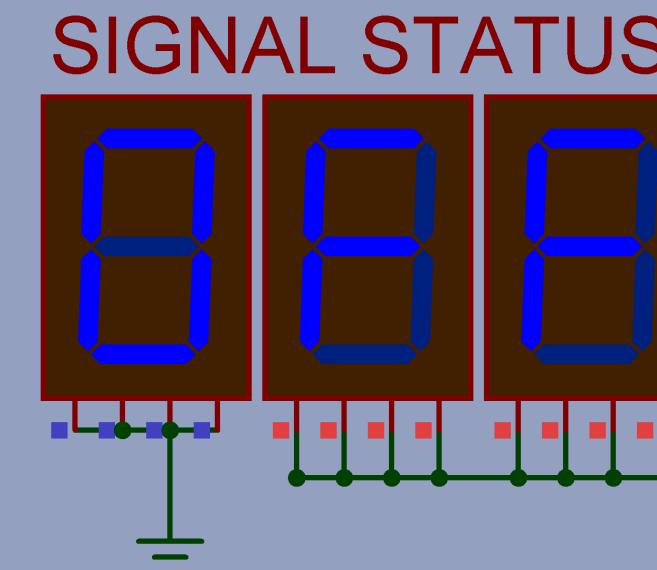


CONVERTER BLOCK DIAGRAM



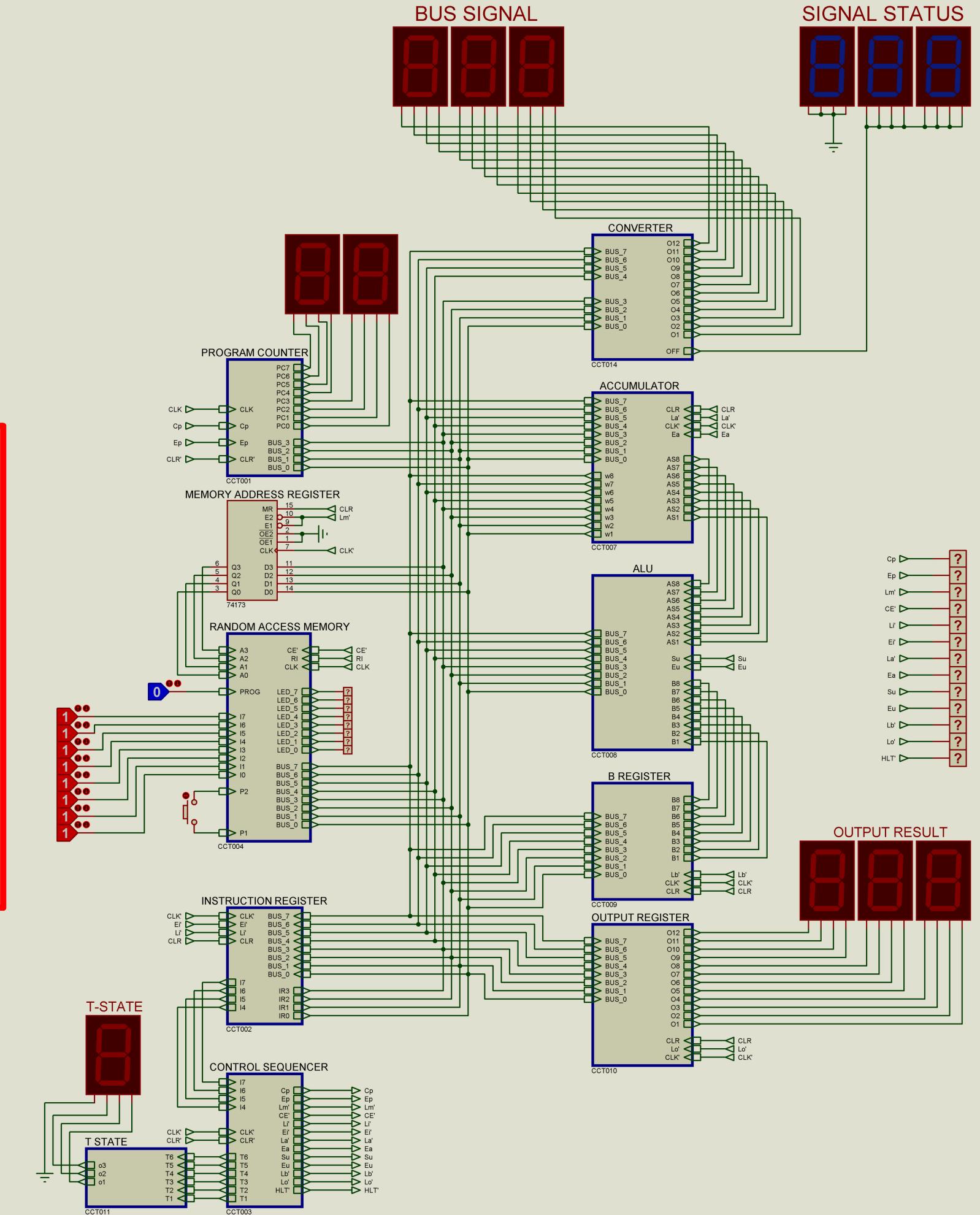
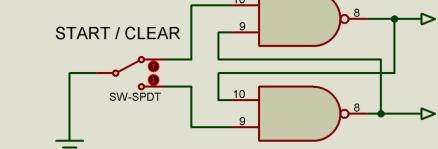
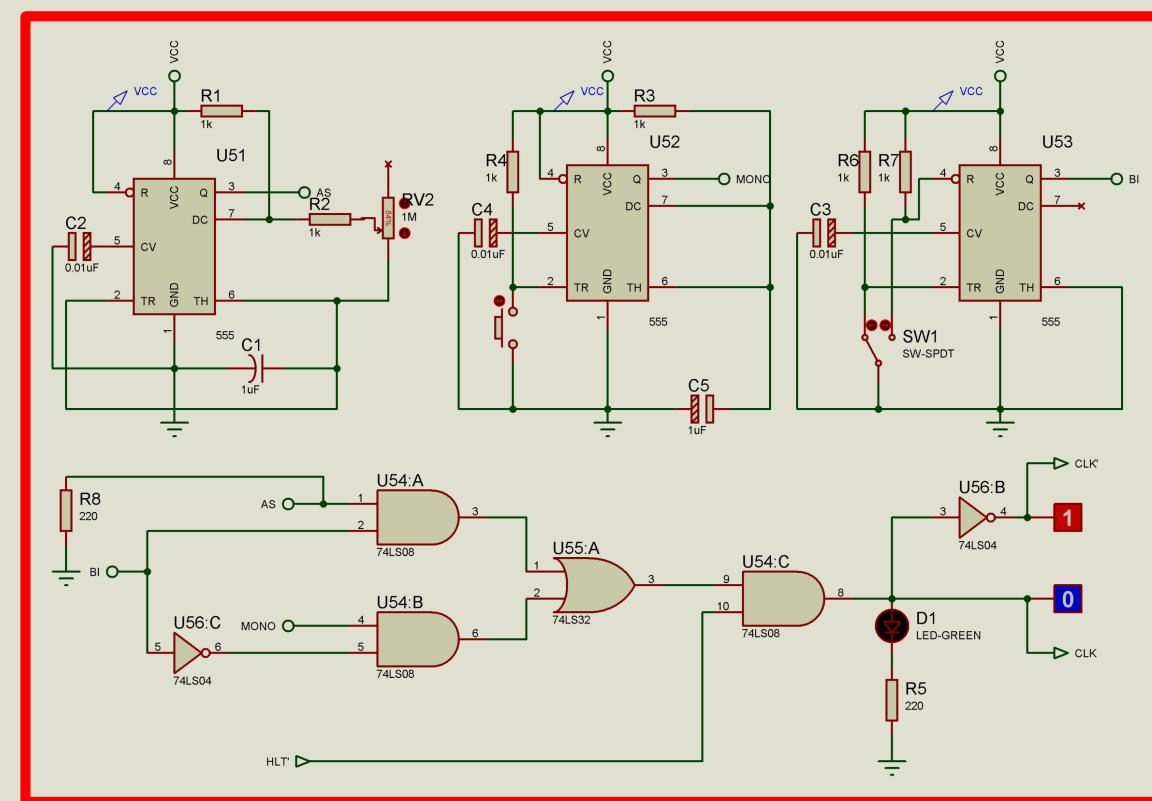
BCD SIGNAL & SIGNAL STATUS

The data contained in the bus will be displayed on the "BUS SIGNAL" display. Let's pretend, we're operating the HLT operation. The bus will receive the output (Binary: 1111111, Decimal: 25). As we're using an 8-bit binary to BCD converter, the display will show 25. Again, when we operate the HLT operation, the HLT's "SIGNAL STATUS" will show "OFF," which signifies that the computing process has been completed and it is off now. A 74LS30 NAND gate and 74LS04 NOR gate are connected to this "OFF" pin in order to show "OFF". The BCD 7-segment display's "OFF" pin is connected to the display's associated pins through the "OFF" pin.



SAP FINAL MODEL

CLOCK PULSE GENERATOR



DISCUSSION

OUTCOME

- **8-bit bus line with a converter to show current bus value in decimal.**
- **We have used three 7-seg BCD display to show “OFF” at the end of all instructions being executed.**
- **Auto and manual clock pulse generation.**
- **Program counter capable of counting 0-15 (decimal) with a 7-seg BCD display to show output.**
- **Memory address register to handle address bits generated from Program Counter.**
- **16*8 bits programmable ram.**
- **Control Sequencer which controls all other sub circuit modules depending on clock pulse through different control pins.**
- **T-state subcircuit is also used to properly designate current state of an instruction with a 7-seg BCD display to show output.**
- **Accumulator is used to send data to Arithmetic Logic Unit and to the bus for final output.**
- **B Register is used to send data to Arithmetic Logic Unit.**
- **Accumulator is implemented for addition and subtraction calculation.**
- **Output register will show the final output through a 7-seg BCD display. It outputs value in equivalent decimal value.**

DISCUSSION

LIMITATIONS

- **Total state number was not reduced. Need optimization in total state number.**
- **It cannot subtract a bigger number from a smaller one. Cannot output a negative value either.**
- **Cannot handle overflow bit.**

FUTURE DEVELOPMENT

- **Total state number can be reduced.**
- **Output of negative value can be shown.**