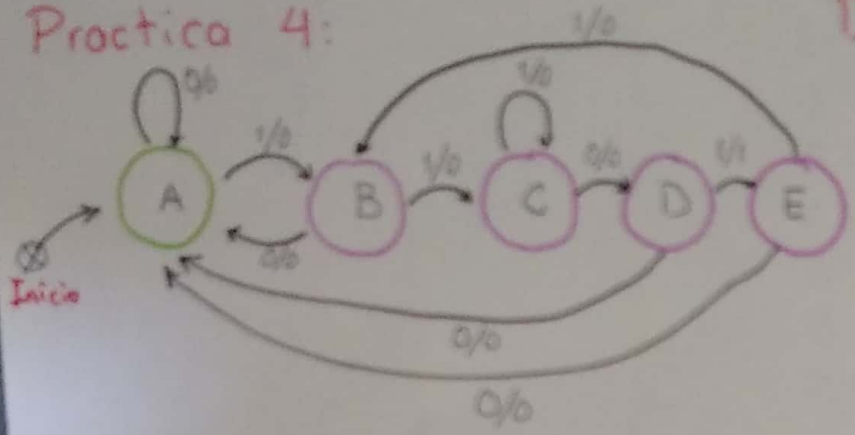
Córdova Pichardo Francisco Oziel

Practica 4: Detector de secuencia

Secuencia

1101



Descripción de la máquina de Mealy por la séxtupla

$M = \{Q, \Sigma, \Delta, \delta, \lambda, q_0\}$

$Q = \{A, B, C, D, E\}$

$\Sigma = \{0, 1\}$

$\Delta = \{0, 1\}$

Edo_inicial = $\{A\}$

$\delta(A,0) = A$	$\lambda(A,0) = 0$
$\delta(A,1) = B$	$\lambda(A,1) = 0$
$\delta(B,0) = A$	$\lambda(B,0) = 0$
$\delta(B,1) = C$	$\lambda(B,1) = 0$
$\delta(C,0) = D$	$\lambda(C,0) = 0$
$\delta(C,1) = C$	$\lambda(C,1) = 0$
$\delta(D,0) = A$	$\lambda(D,0) = 0$
$\delta(D,1) = E$	$\lambda(D,1) = 1$
$\delta(E,0) = A$	$\lambda(E,0) = 0$
$\delta(E,1) = B$	$\lambda(E,1) = 0$

Parcialmente redundantes

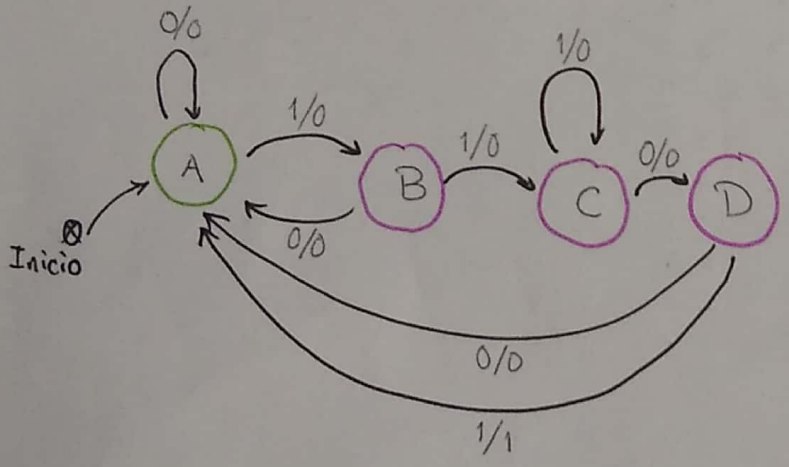
Edo_Act	Entrada	Edo_Sig	Salida
A	0	A	0
A	1	B	0
B	0	A	0
B	1	C	0
C	0	D	0
C	1	C	0
D	0	A	0
D	1	E	1
E	0	A	0
E	1	B	0

A = E
Son redundantes

Tabla resultante

Edo_ACT	E	Edo_Sig	S
A	0	A	0
A	1	B	0
B	0	A	0
B	1	C	0
C	0	D	0
C	1	C	0
D	0	A	0
D	1	A	1

Detector reducido



Asignacion

A = 00
B = 01
C = 10
D = 11

2 ff

codigo secuencial

Edo_Act Q_1	Q_0	E
A 0	0	0
A 0	0	1
B 0	1	0
B 0	1	1
C 1	0	0
C 1	0	1
D 1	1	0
D 1	1	1

FF Tipo D

Edo_Sig Q_1^+	Q_0^+	S
0	0	0
0	1	0
0	0	0
1	0	0
1	1	0
1	0	0
0	0	0
0	0	1

Tabla de exitacion FF tipo D

$Q(t)$	$Q(t+1)$	D
0	0	0
0	1	1
1	0	0
1	1	1

Edo_ACT		E	Edo_SIG		S	D_1	D_0
Q_1	Q_0		Q_1^+	Q_0^+			
0	0	0	0	0	0	0	0
0	0	1	0	1	0	0	1
0	1	0	0	0	0	0	0
0	1	1	1	0	0	1	0
1	0	0	1	1	0	1	1
1	0	1	1	0	0	1	0
1	1	0	0	0	0	0	0
1	1	1	0	0	1	0	0

Entrados

D_1

Q_1/Q_0 E	00	01	11	10
0	0	0	1	0
1	1	1	0	0

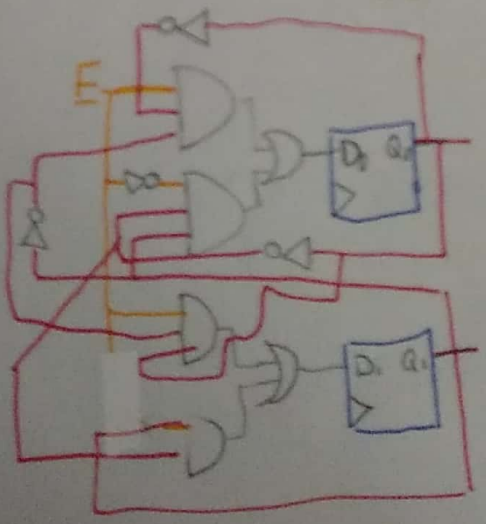
D_0

Q_1/Q_0 E	00	01	11	10
0	0	1	0	0
1	1	0	0	0

$D_1 = \overline{Q_1}\, Q_0 E + Q_1 \overline{Q_0}$

$D_0 = \overline{Q_1}\, \overline{Q_0} E + Q_1 \overline{Q_0}\, \overline{E}$

$S = Q_1 Q_0 E$



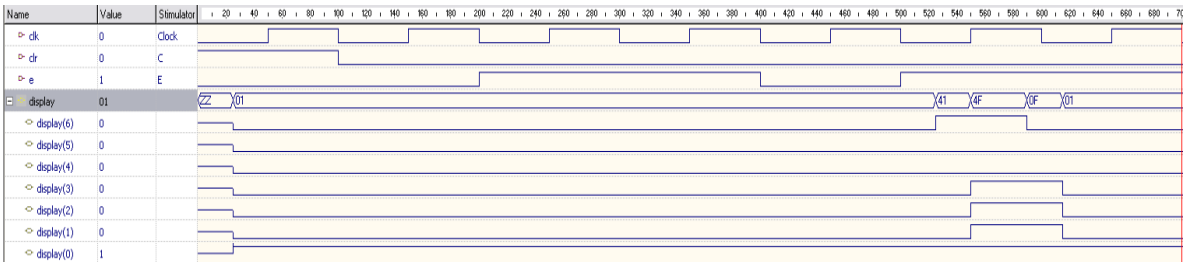
Córdova Pichardo Francisco Uziel

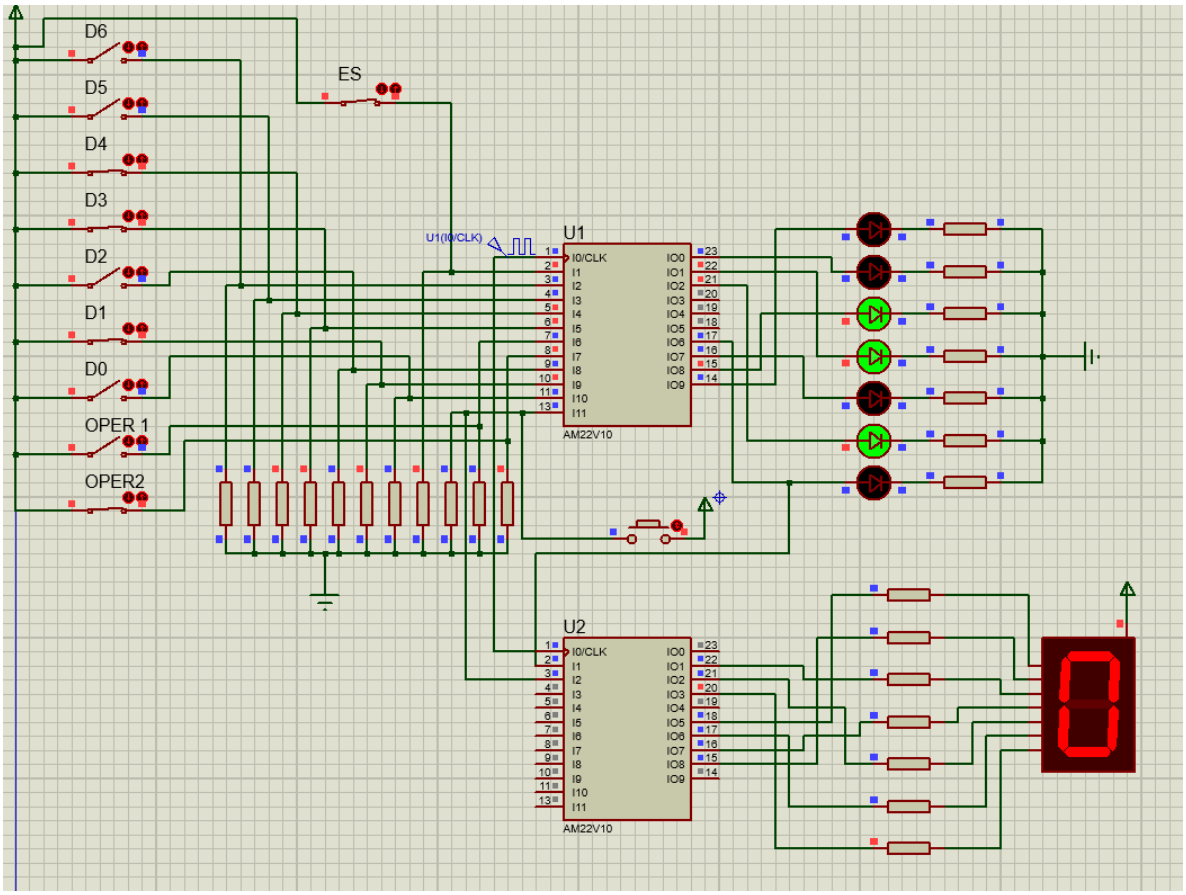
Practica 4

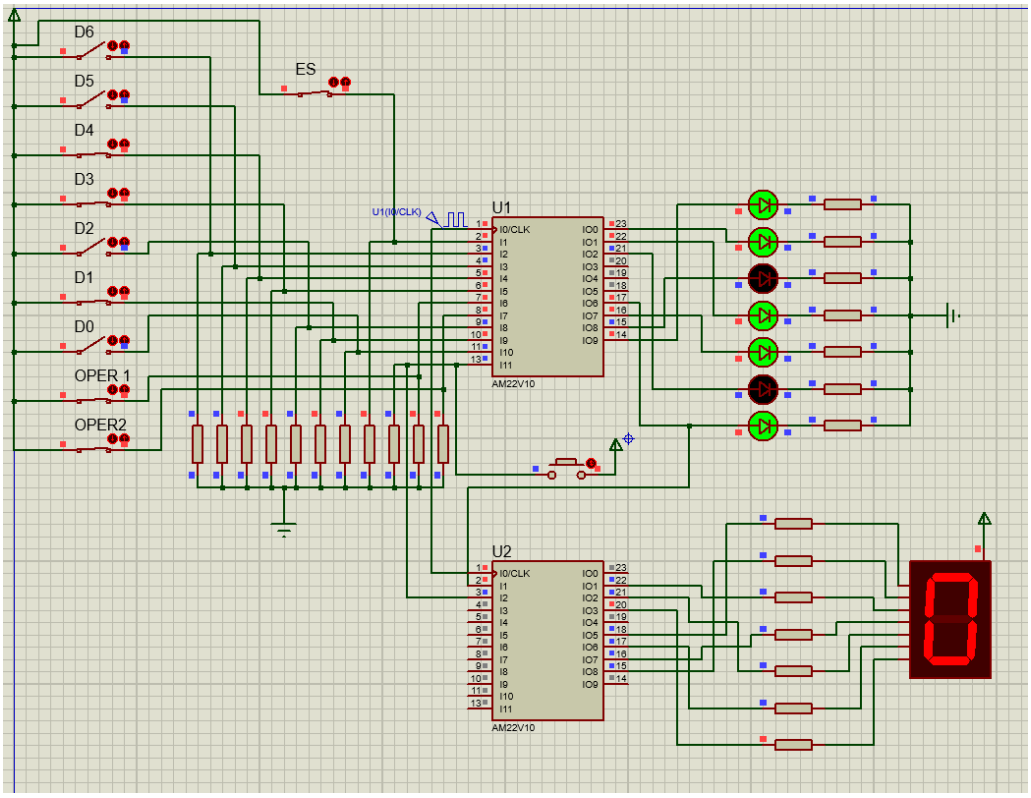
```
 1 library ieee;
 2 use ieee.std_logic_1164.all;
 3
 4 entity PDetector is port
 5 (
 6     E, clk, clr : in std_logic;
 7     display : out std_logic_vector(6 downto 0)
 8 );
 9 end PDetector;
10
11 architecture ADetector of PDetector is
12 signal S : std_logic;
13 signal q : std_logic_vector (1 downto 0);
14 begin
15     process(clk, clr)
16     begin
17         if(clr = '1') then
18             q <= "00";
19             S <= '0';
20         elsif (rising_edge(clk)) then
21             q(0) <= ((not(q(1)))and(not(q(0)))and(E))or((q(1))and(not(q(0)))and(not(E)));
22             q(1) <= ((not(q(1)))and(q(0))and(E))or((q(1))and(not(q(0))));
23         end if;
24     end process;
25     S <= q(1)and(q(0))and(E);
26     display <= "1001111" when (S='1') else
27                "0000001";
28 end ADetector;
```

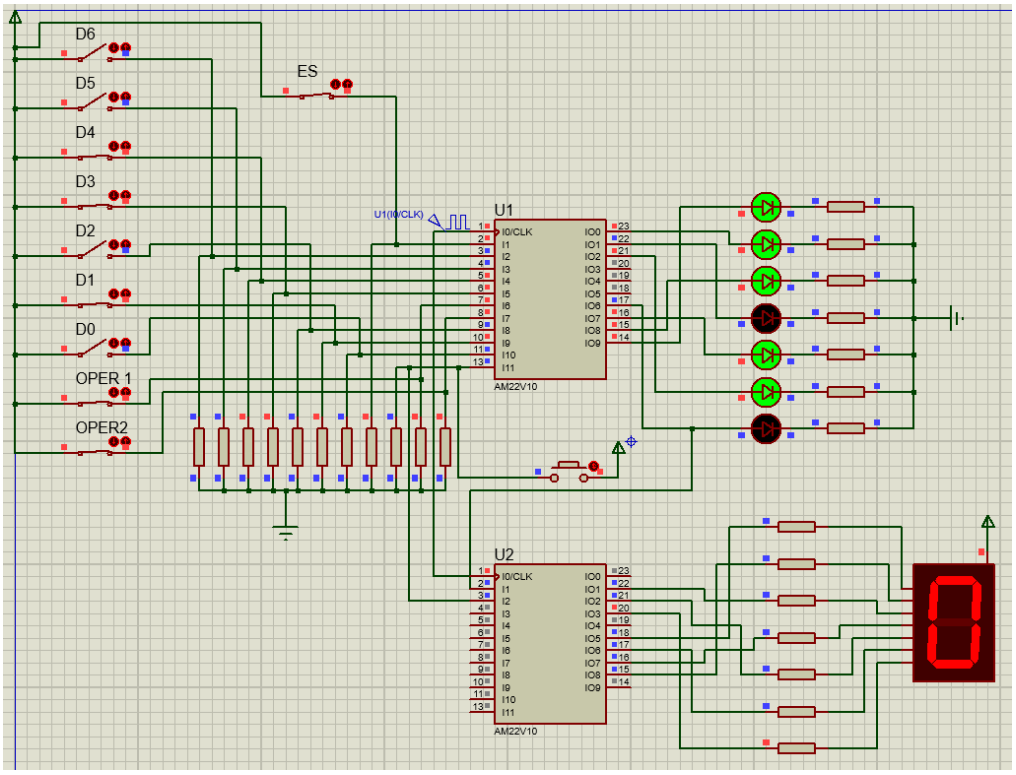
C22V10

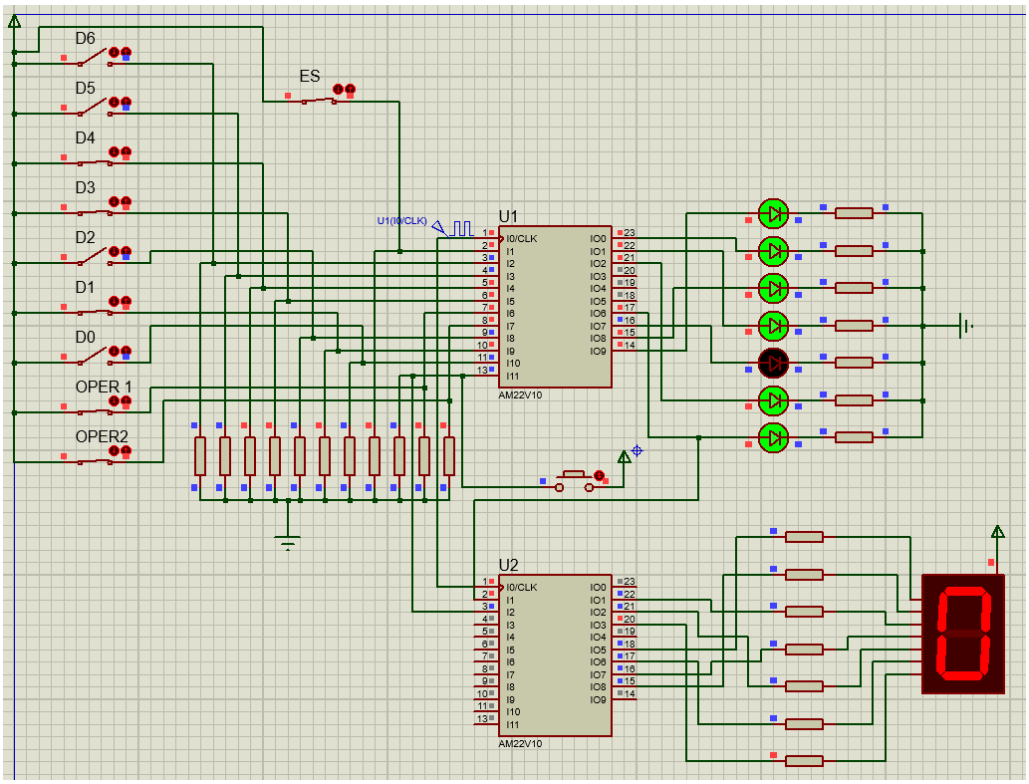
```
        clk =| 1|                                  |24|* not used
          e =| 2|                                  |23|= (q_0)
        clr =| 3|                                  |22|= display(4)
  not used *| 4|                                  |21|= display(2)
  not used *| 5|                                  |20|= display(0)
  not used *| 6|                                  |19|* not used
  not used *| 7|                                  |18|= display(6)
  not used *| 8|                                  |17|= display(1)
  not used *| 9|                                  |16|= display(3)
  not used *|10|                                  |15|= display(5)
  not used *|11|                                  |14|= (q_1)
  not used *|12|                                  |13|* not used
```

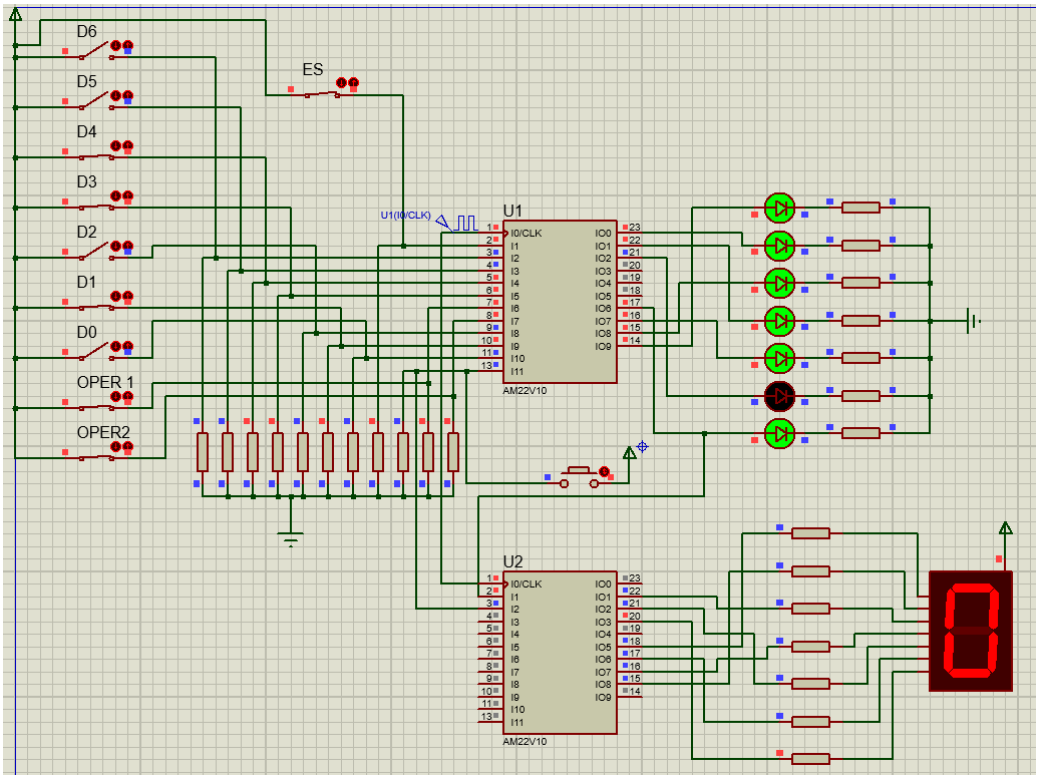

Name
Value
Stimulator
clk
clr
e
display
display(6)
display(5)
display(4)
display(3)
display(2)
display(1)
display(0)
Clock
C
E
ZZ
01
41
4F
0F
01

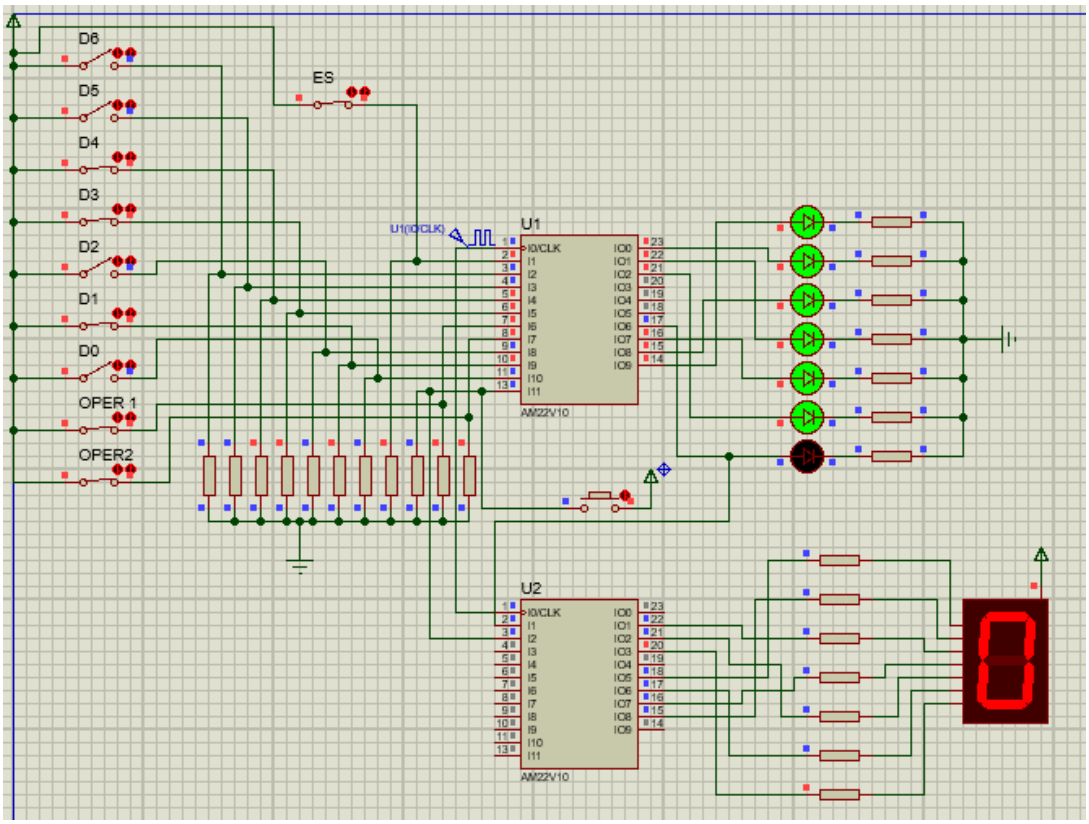

D6
D5
D4
D3
D2
D1
D0
OPER 1
OPER2
ES
U1(I0/CLK)
U1
I0/CLK
I1
I2
I3
I4
I5
I6
I7
I8
I9
I10
I11
IO0
IO1
IO2
IO3
IO4
IO5
IO6
IO7
IO8
IO9
AM22V10
U2
AM22V10

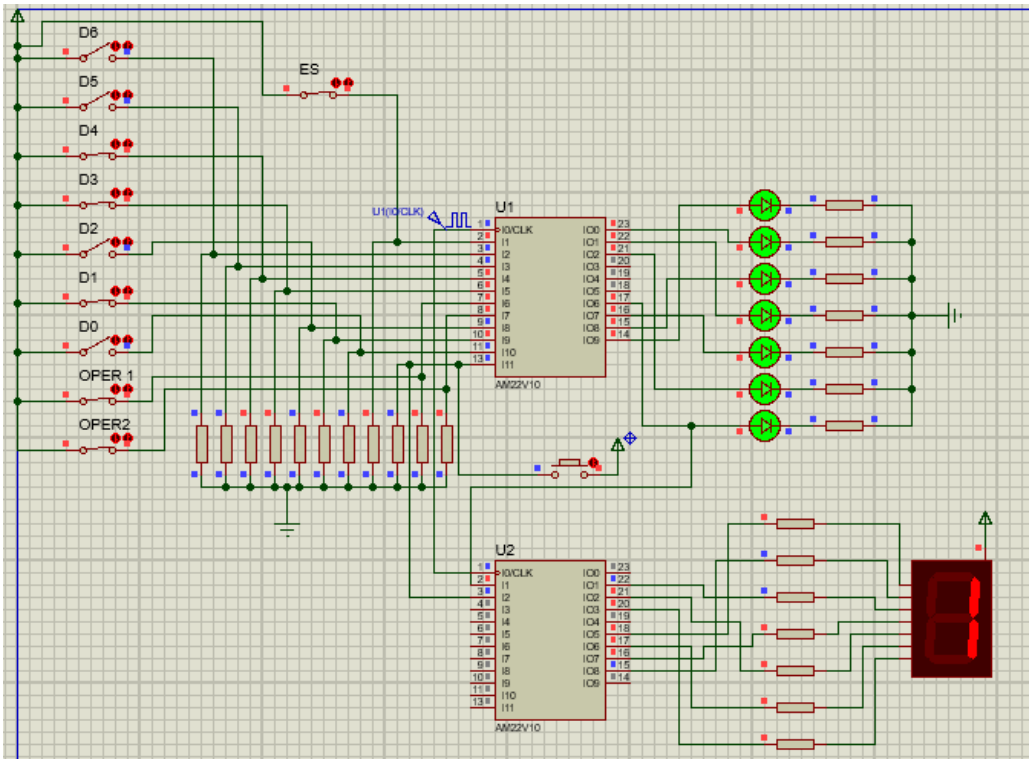

D6
D5
D4
D3
D2
D1
D0
OPER 1
OPER2
ES
U1(IO/CLK)
U1
U2
AM22V10

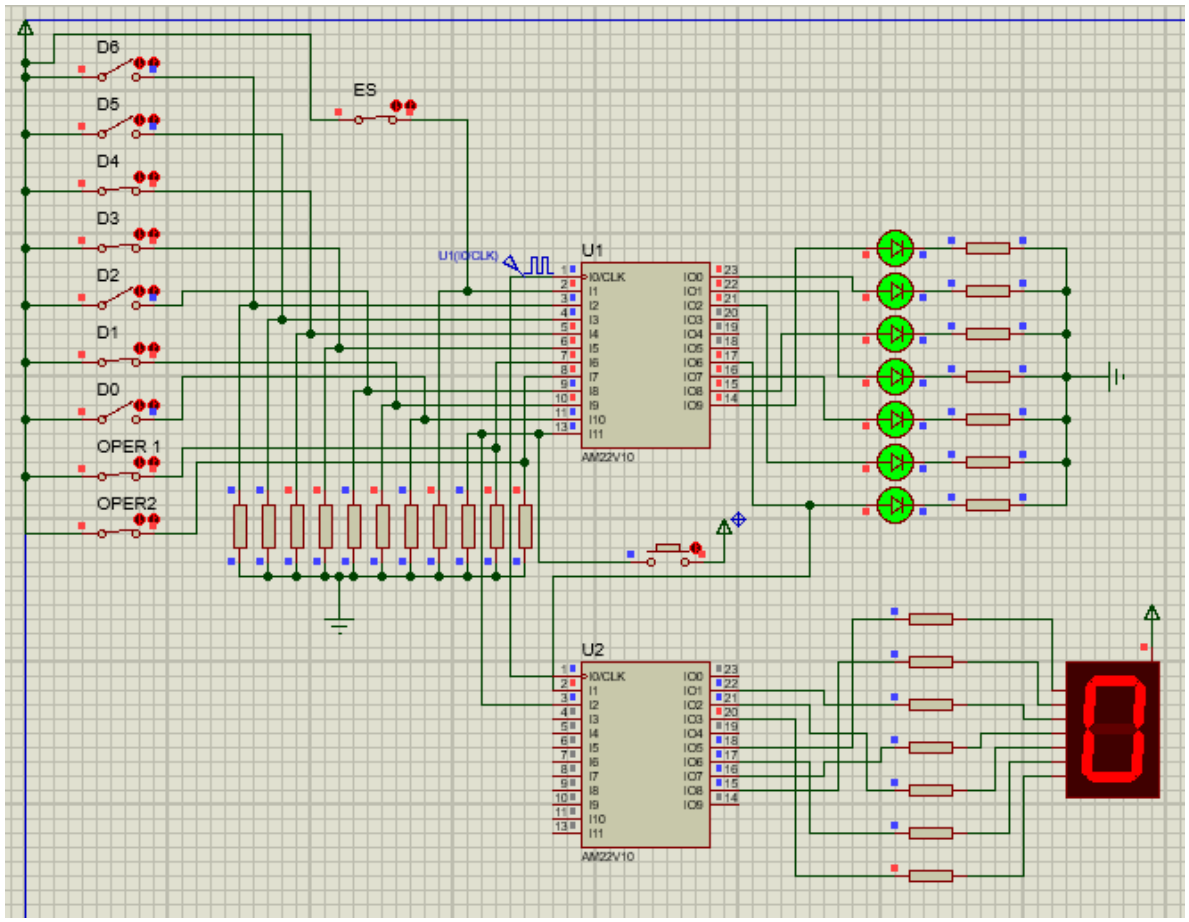

D6
D5
D4
D3
D2
D1
D0
OPER 1
OPER2
ES
U1(IO/CLK)
U1
U2
AM22V10


D6
D5
D4
D3
D2
D1
D0
OPER 1
OPER2
ES
U1(I0/CLK)
U1
I0/CLK
I1
I2
I3
I4
I5
I6
I7
I8
I9
I10
I11
IO0
IO1
IO2
IO3
IO4
IO5
IO6
IO7
IO8
IO9
AM22V10
U2
AM22V10


D6
D5
D4
D3
D2
D1
D0
OPER 1
OPER2
ES
U1(I0/CLK)
U1
I0/CLK
I1
I2
I3
I4
I5
I6
I7
I8
I9
I10
I11
IO0
IO1
IO2
IO3
IO4
IO5
IO6
IO7
IO8
IO9
AM22V10
U2
AM22V10


D6
D5
D4
D3
D2
D1
D0
OPER 1
OPER2
ES
U1
U1(IO/CLK)
IO/CLK
AM22V10
U2
AM22V10


D6
D5
D4
D3
D2
D1
D0
OPER 1
OPER2
ES
U1
U1(IO/CLK)
IO/CLK
AM22V10
U2
AM22V10


D6
D5
D4
D3
D2
D1
D0
OPER 1
OPER2
ES
U1(I0/CLK)
U1
I0/CLK
I1
I2
I3
I4
I5
I6
I7
I8
I9
I10
I11
IO0
IO1
IO2
IO3
IO4
IO5
IO6
IO7
IO8
IO9
AM22V10
U2
I0/CLK
I1
I2
I3
I4
I5
I6
I7
I8
I9
I10
I11
IO0
IO1
IO2
IO3
IO4
IO5
IO6
IO7
IO8
IO9
AM22V10