

## ANÁLISIS A MANO

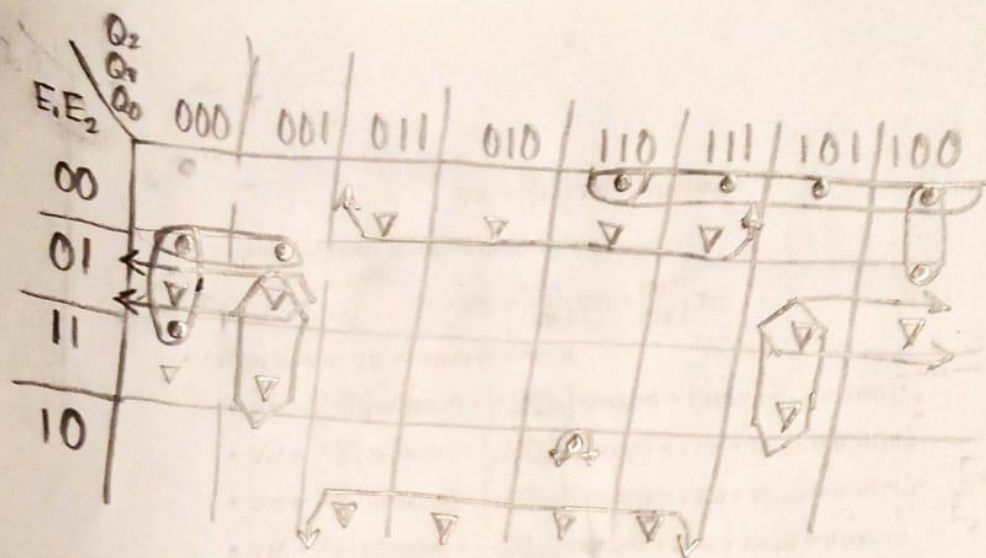
E	Q <sub>0</sub> Q <sub>1</sub> Q <sub>2</sub>	Q <sub>0</sub> Q <sub>1</sub> Q <sub>2</sub>	E, E <sub>0</sub>	S, S <sub>0</sub>	J <sub>0</sub> , K <sub>0</sub>	J <sub>1</sub> , K <sub>1</sub>	J <sub>2</sub> , K <sub>2</sub>	
Q <sub>0</sub>	000	000	00	00	0X	0X	0X	
	000	001	10	00	0X	0X	1X	
	000	100	11	00	1X	0X	0X	
	000	110	01	00	1X	1X	0X	
Q <sub>1</sub>	001	001	10	00	0X	0X	X0	
	001	011	11	00	0X	1X	X0	
	001	110	01	00	1X	1X	X1	
	001	000	00	00	0X	0X	X1	
Q <sub>2</sub>	011	011	11	00	0X	X0	X0	
	011	010	01	00	0X	X0	X1	
	011	001	10	00	0X	X1	X0	
	011	000	00	00	0X	X1	X1	
Q <sub>3</sub>	010	010	01	00	0X	X0	0X	
	010	000	00	01	0X	X1	0X	
	010	011	11	00	0X	X0	1X	
	010	001	10	01	0X	X1	1X	
Q <sub>4</sub>	110	110	01	00	X0	X0	0X	
	110	000	00	00	X1	X1	0X	
	110	001	10	00	X1	X1	1X	
	110	111	11	00	X0	X0	1X	
Q <sub>5</sub>	111	111	11	00	X0	X0	X0	
	111	000	00	00	X1	X1	X1	
	111	110	01	00	X0	X0	X1	
	111	101	10	00	X0	X1	X0	
Q <sub>6</sub>	101	101	10	00	X0	0X	X0	
	101	000	00	10	X1	0X	X1	
	101	110	01	10	X0	1X	X1	
	101	111	11	00	X0	1X	X0	
Q <sub>7</sub>	100	100	11	00	X0	0X	0X	
	100	110	01	00	X0	1X	0X	
	100	000	00	00	X1	0X	0X	
	100	001	10	00	X1	0X	1X	

$J_1 = K_1$

$$S_1 = \overline{Q_2} Q_1 \overline{Q_0} \overline{E_0}$$

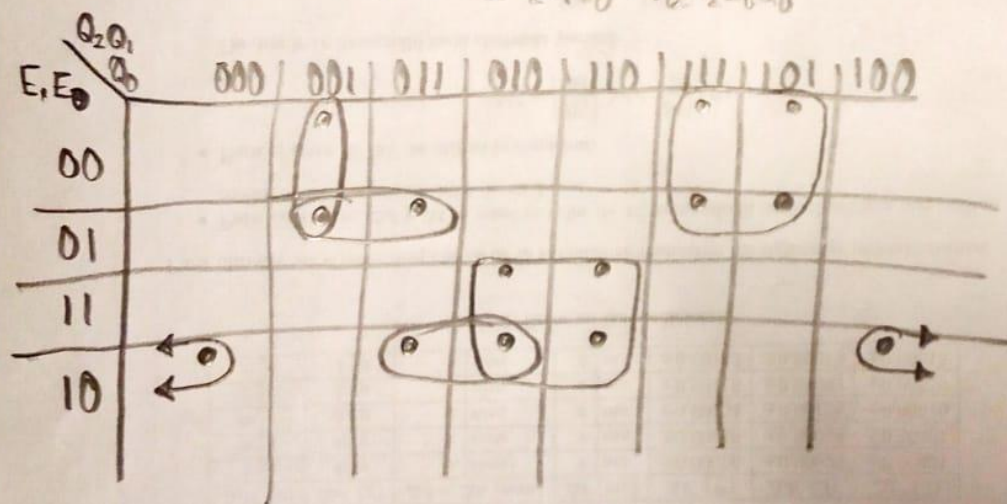
$$S_0 = Q_2 \overline{Q_1} Q_0 \overline{E_1}$$

2



•  $J_2 = K_2 = \bar{E}_1 \bar{E}_2 \bar{Q}_2 \bar{Q}_1 \bar{Q}_0 + \bar{E}_1 \bar{E}_2 Q_2 + \bar{E}_1 Q_2 \bar{Q}_1 \bar{Q}_0 + \bar{E}_1 \bar{Q}_2 \bar{Q}_1 \bar{Q}_0 + \bar{E}_1 E_2 \bar{Q}_2 \bar{Q}_1$

$\nabla J_1 = K_1 = \bar{E}_0 \bar{Q}_1 + \bar{E}_1 E_2 Q_1 + \bar{E}_2 \bar{Q}_2 \bar{Q}_1 \bar{Q}_0 + E_2 Q_2 \bar{Q}_1 \bar{Q}_0$



$J_0 = K_0 = \bar{E}_1 \bar{Q}_2 \bar{Q}_1 \bar{Q}_0 + \bar{E}_1 E_0 \bar{Q}_2 \bar{Q}_0 + \bar{E}_1 Q_2 \bar{Q}_0 + E_1 Q_1 \bar{Q}_0 + E_1 \bar{E}_0 \bar{Q}_1 \bar{Q}_0 + E_1 \bar{E}_0 \bar{Q}_2 Q_1$

## CÓDIGO FUENTE

### SENSORES

```
1 library ieee;
2 use ieee.std_logic_1164.all;
3 use ieee.std_logic_arith.all;
4 use ieee.std_logic_unsigned.all;
5
6 entity sensor is port(
7     CLK, CLR: in std_logic;
8     E : in std_logic_vector (1 downto 0);
9     D : out std_logic_vector (2 downto 0);
10    U : out std_logic_vector (3 downto 0)
11 );
12 end entity;
13
14 architecture aSensor of sensor is
15 type estado is (Q0, Q1, Q2, Q3, Q4, Q5, Q6, Q7);
16 signal actual, sig : estado;
17 signal S : std_logic_vector (1 downto 0);
18 begin
19     --Para el registro del estado
20     process (CLK, CLR)
21     begin
22         if CLR = '1' then
23             actual <= Q0;
24         elsif rising_edge(CLK) then
25             actual <= sig;
26         end if;
27     end process;
28
29     --Para cambiar el estado
30     process (E, actual)
31     begin
32         case actual is
33             when Q0 =>
34                 if E = "00" then
35                     S <= "00";
36                     sig <= Q0;
37                 elsif E = "01" then
38                     S <= "00";
39                     sig <= Q4;
40                 elsif E = "10" then
41                     S <= "00";
42                     sig <= Q1;
43                 else
44                     S <= "00";
45                     sig <= Q7;
46                 end if;
```

```

47      when Q1 =>
48          if E = "00" then
49              S <= "00";
50              sig <= Q0;
51          elsif E = "01" then
52              S <= "00";
53              sig <= Q4;
54          elsif E = "10" then
55              S <= "00";
56              sig <= Q1;
57          else
58              S <= "00";
59              sig <= Q2;
60          end if;
61      when Q2 =>
62          if E = "00" then
63              S <= "00";
64              sig <= Q0;
65          elsif E = "01" then
66              S <= "00";
67              sig <= Q3;
68          elsif E = "10" then
69              S <= "00";
70              sig <= Q1;
71          else
72              S <= "00";
73              sig <= Q2;

```

```

74          end if;
75      when Q3 =>
76          if E = "00" then
77              S <= "01";
78              sig <= Q0;
79          elsif E = "01" then
80              S <= "00";
81              sig <= Q3;
82          elsif E = "10" then
83              S <= "01";
84              sig <= Q1;
85          else
86              S <= "00";
87              sig <= Q2;
88          end if;
89      when Q4 =>
90          if E = "00" then
91              S <= "00";
92              sig <= Q0;
93          elsif E = "01" then
94              S <= "00";
95              sig <= Q4;
96          elsif E = "10" then
97              S <= "00";
98              sig <= Q1;

```

---

```

99         else
100             S <= "00";
101             sig <= Q5;
102         end if;
103     when Q5 =>
104         if E = "00" then
105             S <= "00";
106             sig <= Q0;
107         elsif E = "01" then
108             S <= "00";
109             sig <= Q4;
110         elsif E = "10" then
111             S <= "00";
112             sig <= Q6;
113         else
114             S <= "00";
115             sig <= Q5;
116         end if;
117     when Q6 =>
118         if E = "00" then
119             S <= "10";
120             sig <= Q0;
121         elsif E = "01" then
122             S <= "10";
123             sig <= Q4;
124         elsif E = "10" then
125             S <= "00";
126             sig <= Q6;
127         .

```

---

```

127     else
128         S <= "00";
129         sig <= Q5;
130     end if;
131     when others =>
132         if E = "00" then
133             S <= "00";
134             sig <= Q0;
135         elsif E = "01" then
136             S <= "00";
137             sig <= Q4;
138         elsif E = "10" then
139             S <= "00";
140             sig <= Q1;
141         else
142             S <= "00";
143             sig <= Q7;
144         end if;
145     end case;
146 end process;
147
148 --Para el contador

```



```

149     process (CLK, CLR)
150     begin
151         if CLR = '1' then
152             D <= "000";
153             U <= "0000";
154         elsif rising_edge(CLK) then
155             case S is
156                 when "01" => --Ascendente
157                     if U = "1001" then --9
158                         U <= "0000";
159                         D <= D + 1;
160                     else
161                         U <= U + 1;
162                     end if;
163                 when "10" => --Descendente
164                     if U = "0000" then --0
165                         U <= "1001";
166                         D <= D - 1;
167                     else
168                         U <= U - 1;
169                     end if;
170                 when others => --retencion
171                     D <= D;
172                     U <= U;
173             end case;
174         end if;
175     end process;
176 end architecture;

```

## DISPLAY

```

1 library ieee;
2 use ieee.std_logic_1164.all;
3
4 entity Contador is port (
5     CLK, CLR : in std_logic;
6     U : in std_logic_vector (3 downto 0);
7     D : in std_logic_vector (2 downto 0);
8     A : inout std_logic_vector (1 downto 0);
9     DISPLAY : out std_logic_vector (6 downto 0)
10 );
11 end entity;
12
13 architecture aContrador of Contador is
14     signal FF : std_logic_vector (1 downto 0);
15 begin
16     process (CLK, CLR)
17     begin
18         if CLR = '1' then
19             FF <= "00";
20         elsif rising_edge(CLK) then
21             if FF = "01" then
22                 FF <= "10";

```

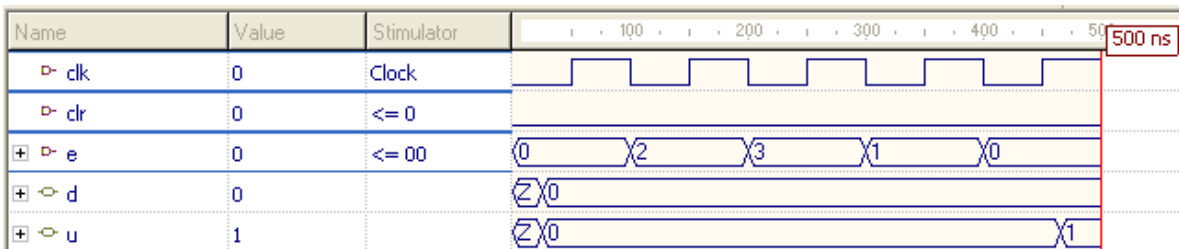
```

23         else
24             FF <= "01";
25         end if;
26     end if;
27
28     A <= FF;
29 end process;
30
31 process (FF, U, D)
32 begin
33     if FF = "01" then
34         case D is
35             when "000" => DISPLAY <= "1111110"; --0
36             when "001" => DISPLAY <= "0110000"; --1
37             when "010" => DISPLAY <= "1101101"; --2
38             when "011" => DISPLAY <= "1111001"; --3
39             when "100" => DISPLAY <= "0110011"; --4
40             when "101" => DISPLAY <= "1011011"; --5
41             when "110" => DISPLAY <= "1011111"; --6
42             when others => DISPLAY <= "1110000"; --7
43         end case;
44     else
45         case U is
46             when "0000" => DISPLAY <= "1111110"; --0
47             when "0001" => DISPLAY <= "0110000"; --1
48             when "0010" => DISPLAY <= "1101101"; --2
49             when "0011" => DISPLAY <= "1111001"; --3
50             when "0100" => DISPLAY <= "0110011"; --4
51
52             when "0101" => DISPLAY <= "1011011"; --5
53             when "0110" => DISPLAY <= "1011111"; --6
54             when "0111" => DISPLAY <= "1110000"; --7
55             when "1000" => DISPLAY <= "1111111"; --8
56             when others => DISPLAY <= "1110011"; --9
57         end case;
58     end if;
59 end process;
60 end architecture;

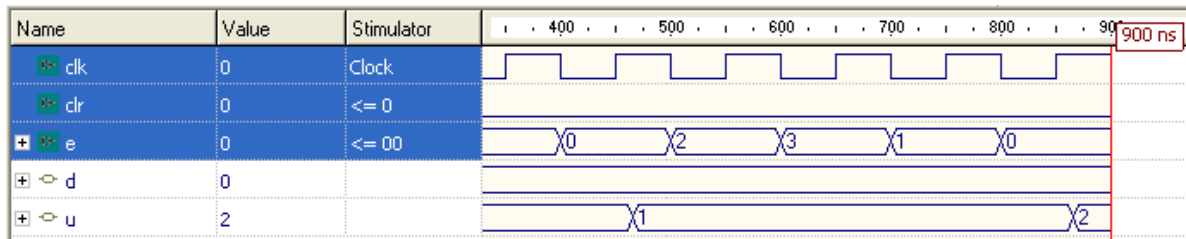
```

## SIMULACIONES EN GALAXY

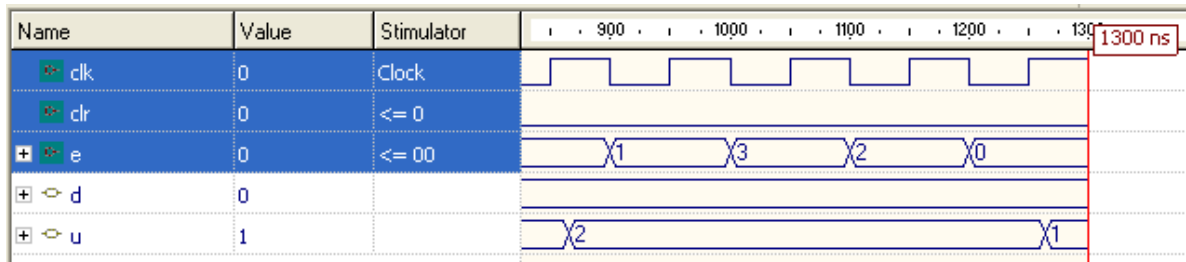
ENTRA 1 PERSONA



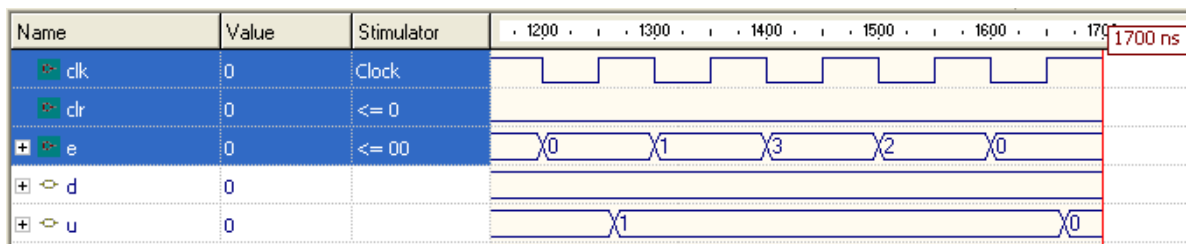
## ENTRA 1 PERSONA



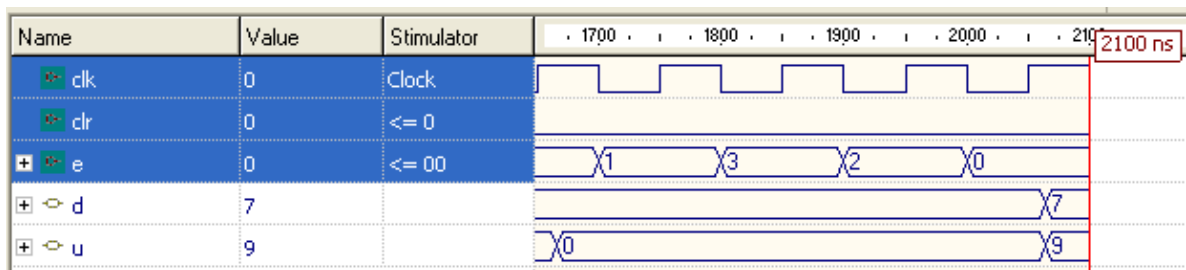
## SALE 1 PERSONA



## SALE 1 PERSONA



## SALE 1 PERSONA





## FSM

## C22V10

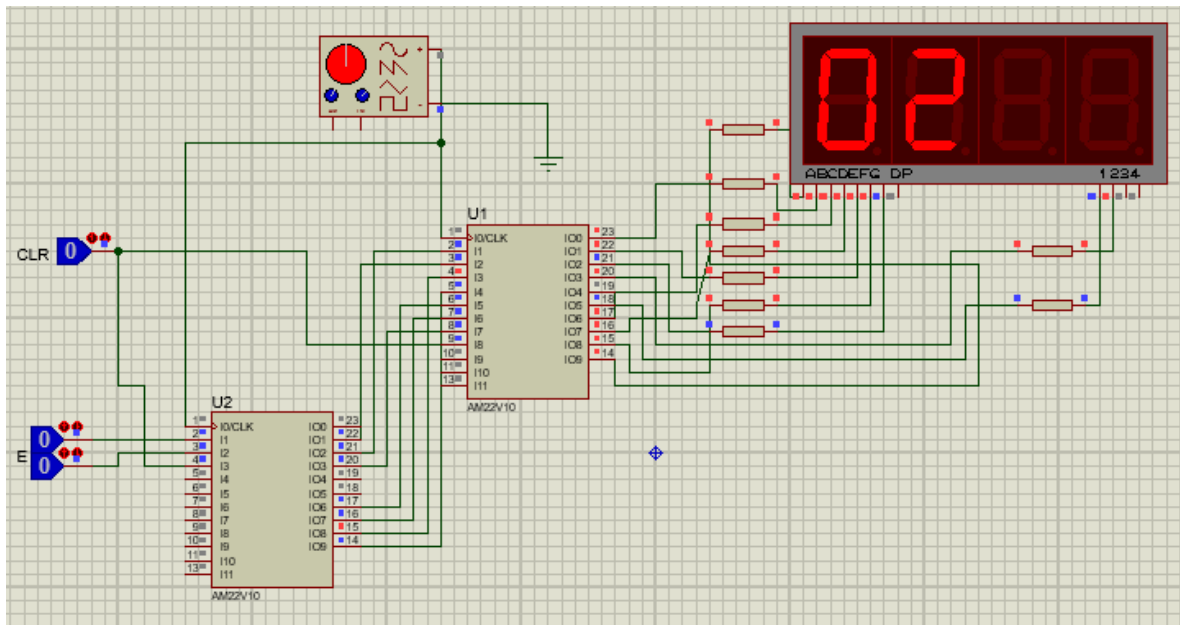
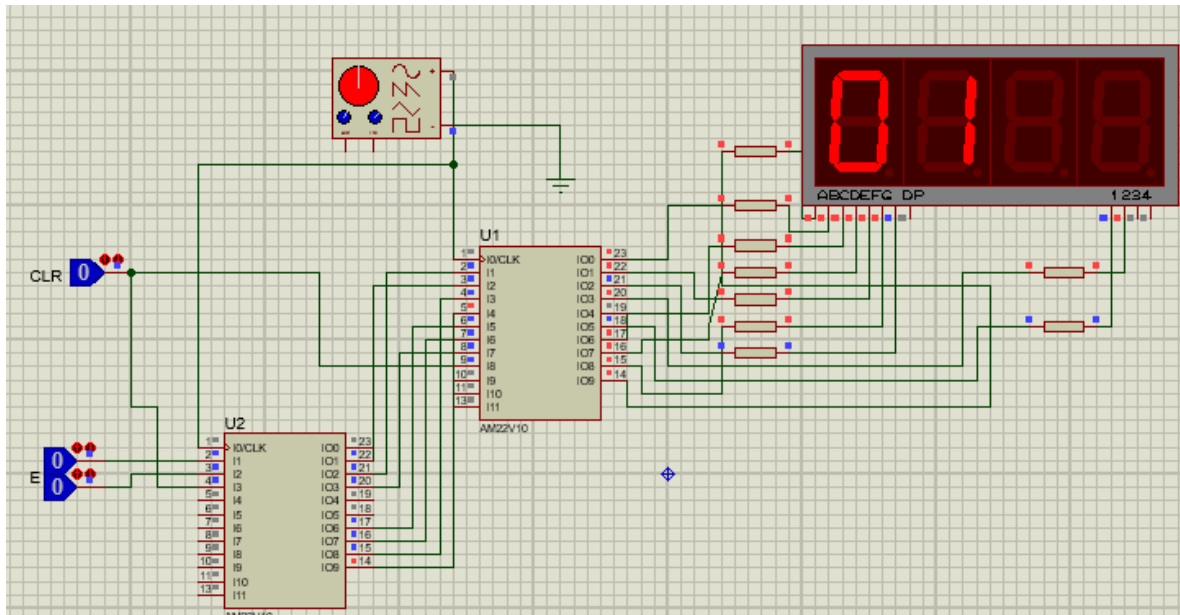
clk =  1	24  * not used
e(1) =  2	23 = (actualSBV_2)
e(0) =  3	22 = u(2)
clr =  4	21 = u(3)
not used *  5	20 = d(0)
not used *  6	19 = (actualSBV_0)
not used *  7	18 = (actualSBV_1)
not used *  8	17 = d(2)
not used *  9	16 = d(1)
not used * 10	15 = u(1)
not used * 11	14 = u(0)
not used * 12	13  * not used

## Contador

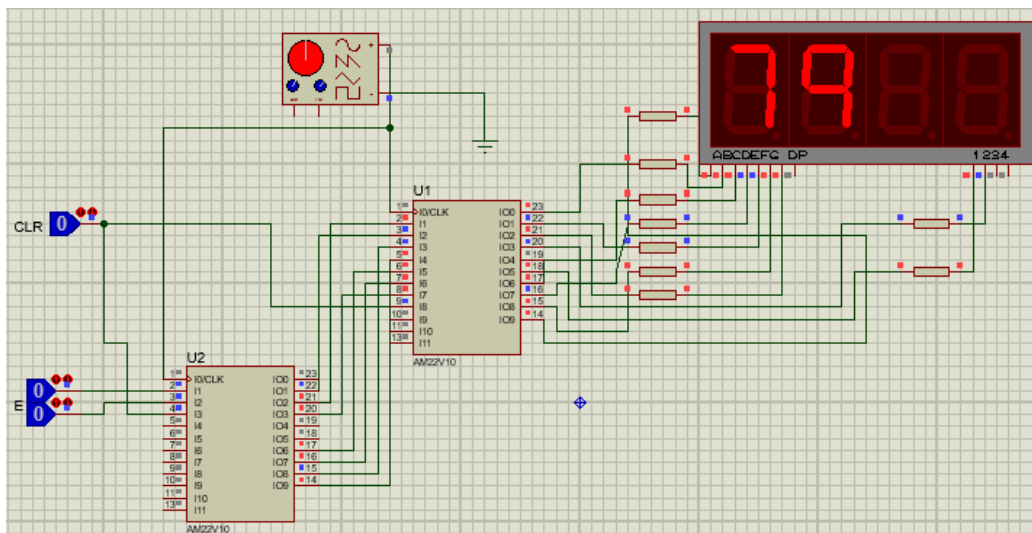
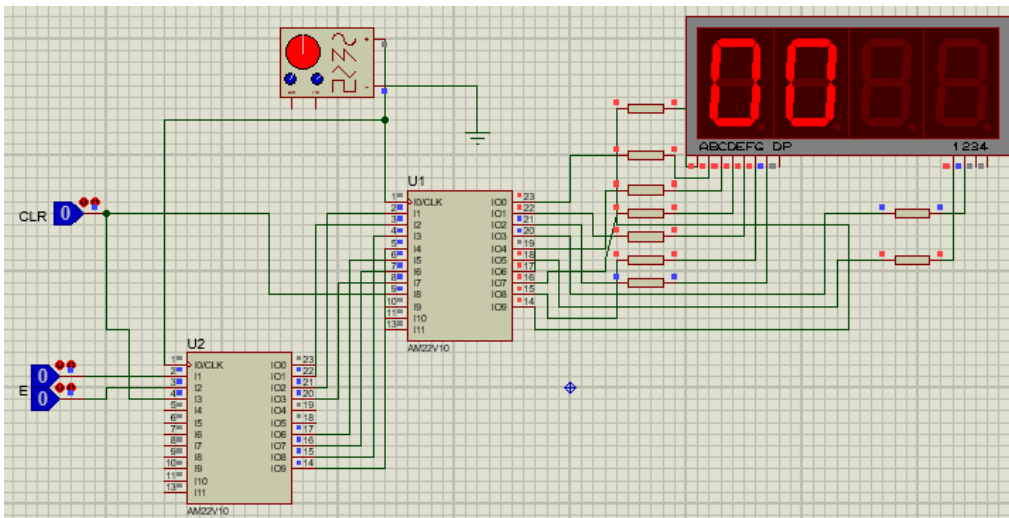
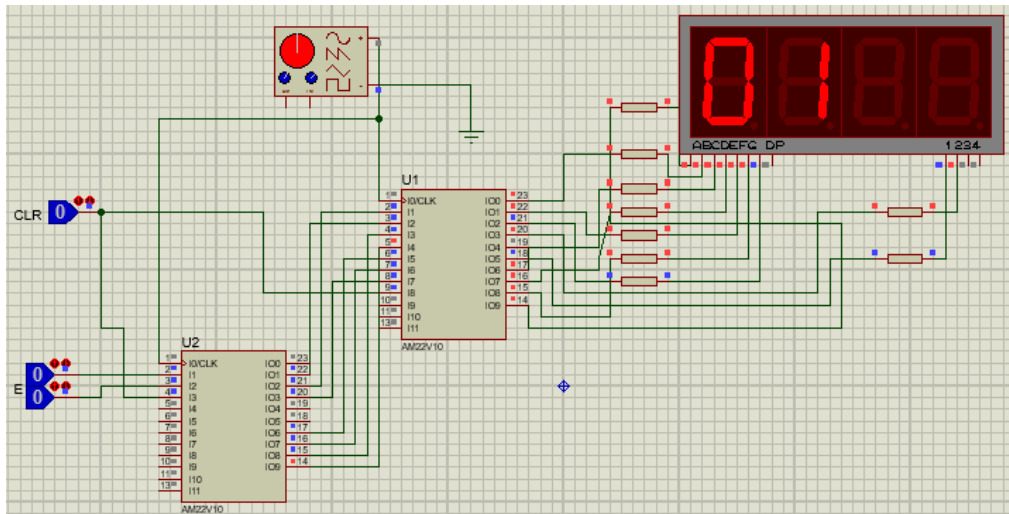
## C22V10

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

## ENTRA 2 PERSONA



# SALEN 3 PERSONAS



## CUESTIONARIO

1. ¿Cuántos dispositivos PLD 22V10 son necesarios para el desarrollo de esta práctica?

2

2. ¿Cuántos dispositivos de la serie 74xx (TTL) ó 40xx (CMOS) hubieras necesitado para el desarrollo de esta práctica?

Para el primero 22, mientras que para el segundo 13.

3. ¿Cuántos pines de entrada/salida del PLD1 22V10 y PLD2 22V10 se usan en el diseño?

El primero 14, el segundo 18.

4. ¿Cuántos términos producto ocupan las ecuaciones para cada señal de salida y que porcentaje se usa en total del PLD1 22V10 y PLD2 22V10?

El primero 88, el segundo 51.

5. ¿Qué puedes concluir de esta práctica?

Las aplicaciones son mucho más amplias de lo que imaginaba.