

CÓDIGO FUENTE

PRIMERA GAL

```

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 Primera is port(
7     CLK, CLR, INI: in std_logic;
8     NUM : in std_logic_vector (5 downto 0);
9     FF : out std_logic_vector (5 downto 0);
10    LB, EB, EC : out std_logic
11 );
12 end entity;
13
14 architecture aPrimera of Primera is
15 type estado is (EO, E1, E2);
16 signal actual, sig : estado;
17 signal LA, EA, Z : std_logic;
18 begin
19     --Para el registro del estado
20     process (CLK, CLR)
21     begin
22         if CLR = '1' then
23             actual <= EO;
24         elsif rising_edge(CLK) then
25             actual <= sig;
26         end if;
27     end process;
28
29     --Para cambiar el estado
30     process (actual, Z, FF, INI)
31     begin
32         LA <= '0';
33         LB <= '0';
34         EA <= '0';
35         EB <= '0';
36         EC <= '0';
37         Z <= not (FF(5) or FF(4) or FF(3) or FF(2) or FF(1) or FF(0));
38
39         case actual is
40             when EO =>
41                 LA <= '1';
42                 LB <= '1';
43
44                 if INI = '1' then
45                     sig <= E1;
46                 else
47                     sig <= EO;
48                 end if;

```

```

49         when E1 =>
50             EA <= '1';
51
52             if Z = '0' then
53                 sig <= E1;
54
55                 if FF(0) = '1' then
56                     EB <= '1';
57                 end if;
58             else
59                 sig <= E2;
60             end if;
61         when others =>
62             EC <= '1';
63
64             if INI = '1' then
65                 sig <= E2;
66             else
67                 sig <= E0;
68             end if;
69         end case;
70     end process;
71
72     --Para el contador
73     process (CLK, CLR)
74     begin
75         if CLR = '1' then
76             FF <= "000000";
77
78             elsif rising_edge(CLK) then
79                 if LA = '1' then --Carga
80                     FF <= NUM;
81                 elsif EA = '1' then --Corrimiento
82                     for i in 0 to 4 loop
83                         FF(i) <= FF(i + 1);
84                     end loop;
85
86                     FF(5) <= '0';
87                 else --Retención
88                     FF <= FF;
89                 end if;
90             end if;
91     end process;
92 end architecture;

```

SEGUNDA GAL

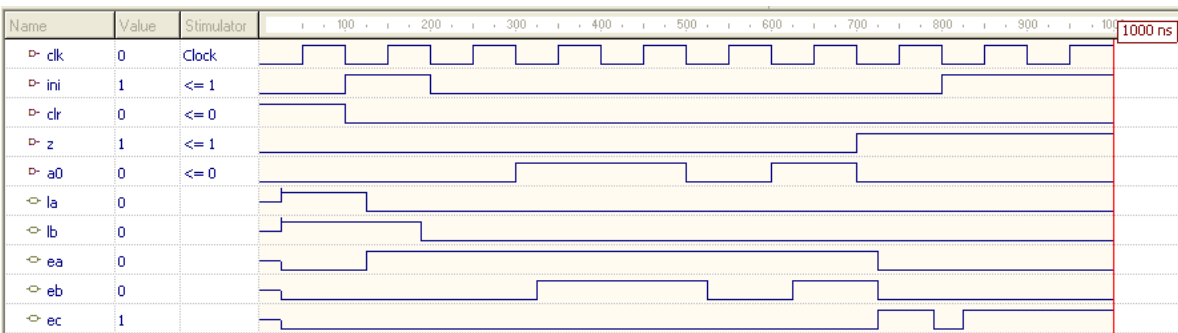
```

1 library ieee;
2 use ieee.std_logic_1164.all;
3
4 entity Segunda is port(
5     CLK, CLR, LB, EB, EC : in std_logic;
6     S : out std_logic_vector (6 downto 0)
7 );
8 end entity;
9
10 architecture aSegunda of Segunda is
11     signal COUNT : std_logic_vector (2 downto 0);
12     signal CODIGO : std_logic_vector (6 downto 0);
13 begin
14     process (CLK, CLR)
15     begin
16         if (CLR = '1') then
17             COUNT <= "000";
18         elsif (rising_edge(CLK)) then
19             if (LB = '0' and EB = '0') then --retiene
20                 COUNT <= COUNT;
21             elsif (LB = '1') then --carga
22                 COUNT <= "000";
23             else --conteo ascendente
24
25                 case COUNT is
26                     when "000" => COUNT <= "001";
27                     when "001" => COUNT <= "010";
28                     when "010" => COUNT <= "011";
29                     when "011" => COUNT <= "100";
30                     when "100" => COUNT <= "101";
31                     when "101" => COUNT <= "110";
32                     when others => COUNT <= "000";
33                 end case;
34             end if;
35         end if;
36
37         case COUNT is
38             when "000" => CODIGO <= "1111110";
39             when "001" => CODIGO <= "0110000";
40             when "010" => CODIGO <= "1101101";
41             when "011" => CODIGO <= "1111001";
42             when "100" => CODIGO <= "0110011";
43             when "101" => CODIGO <= "1011011";
44             when others => CODIGO <= "1011111";
45         end case;
46     end process;
47
48     with EC select
49         S <= CODIGO when '1',
50             "0000001" when others;
51 end architecture;

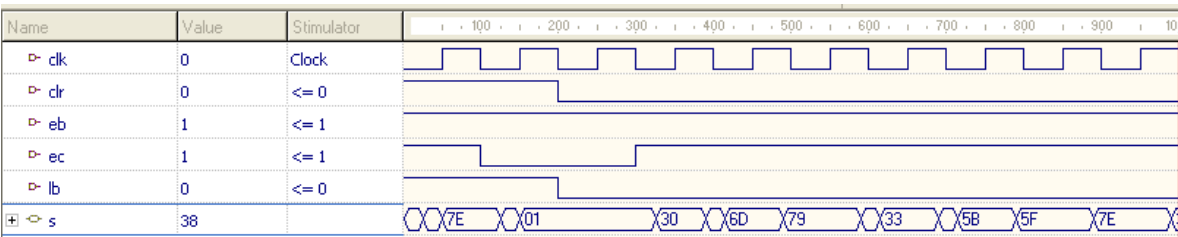
```

SIMULACIONES EN GALAXY

PRIMERA GAL SOLO ASM



SEGUNDA GAL



SIMULACIÓN EN PROTEUS

PRIMERA GAL

C22V10

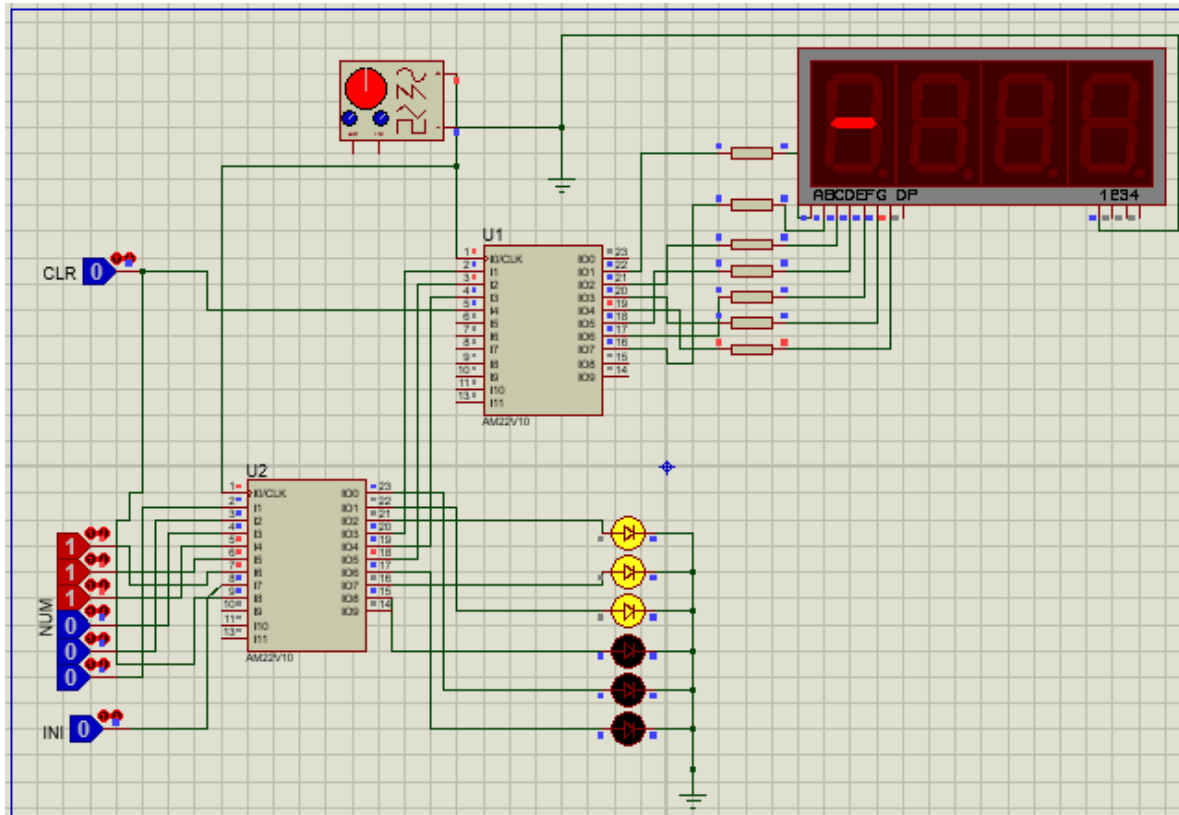
clk	= 1	24 * not used
num(5)	= 2	23 = ff(4)
num(4)	= 3	22 = ff(2)
num(3)	= 4	21 = ff(0)
num(2)	= 5	20 = ec
num(1)	= 6	19 = eb
num(0)	= 7	18 = lb
ini	= 8	17 = ff(5)
clr	= 9	16 = ff(1)
not used *	10	15 = ff(3)
not used *	11	14 = (actualSBV_1)
not used *	12	13 * not used

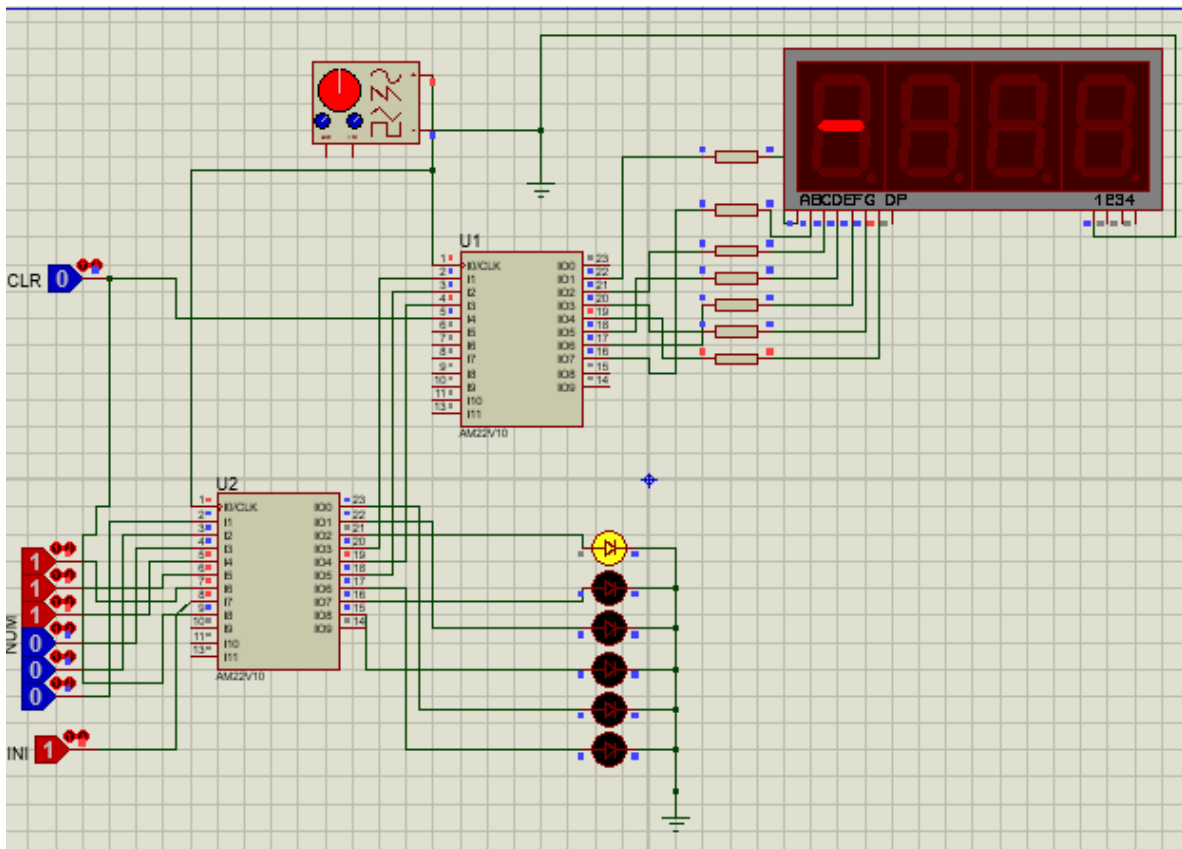
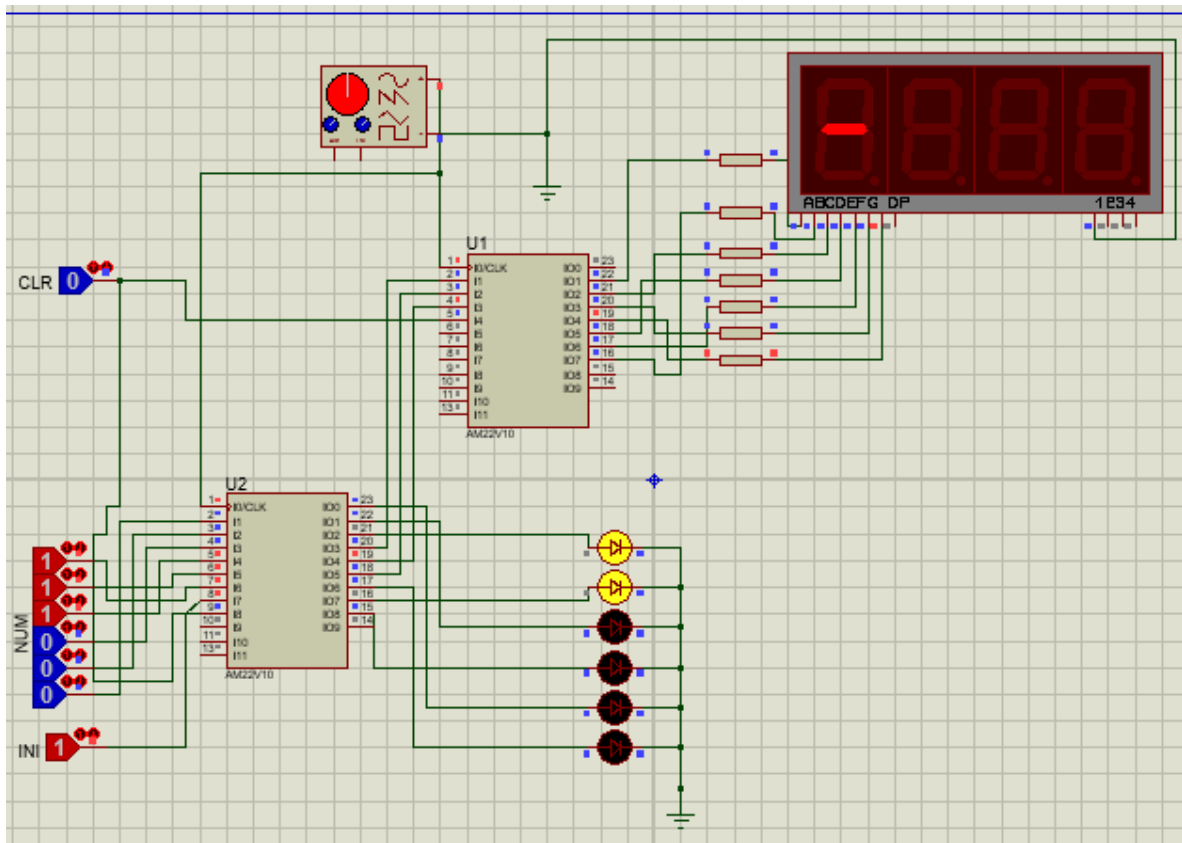
SEGUNDA GAL

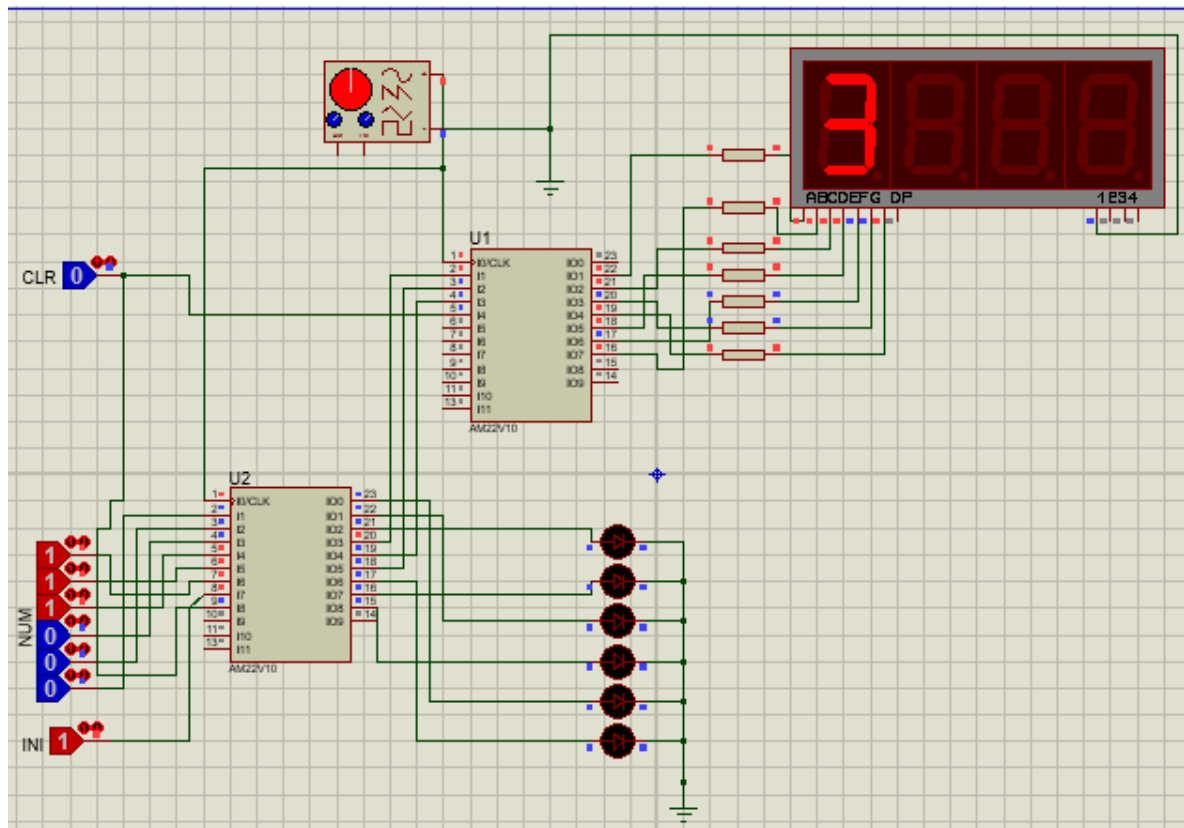
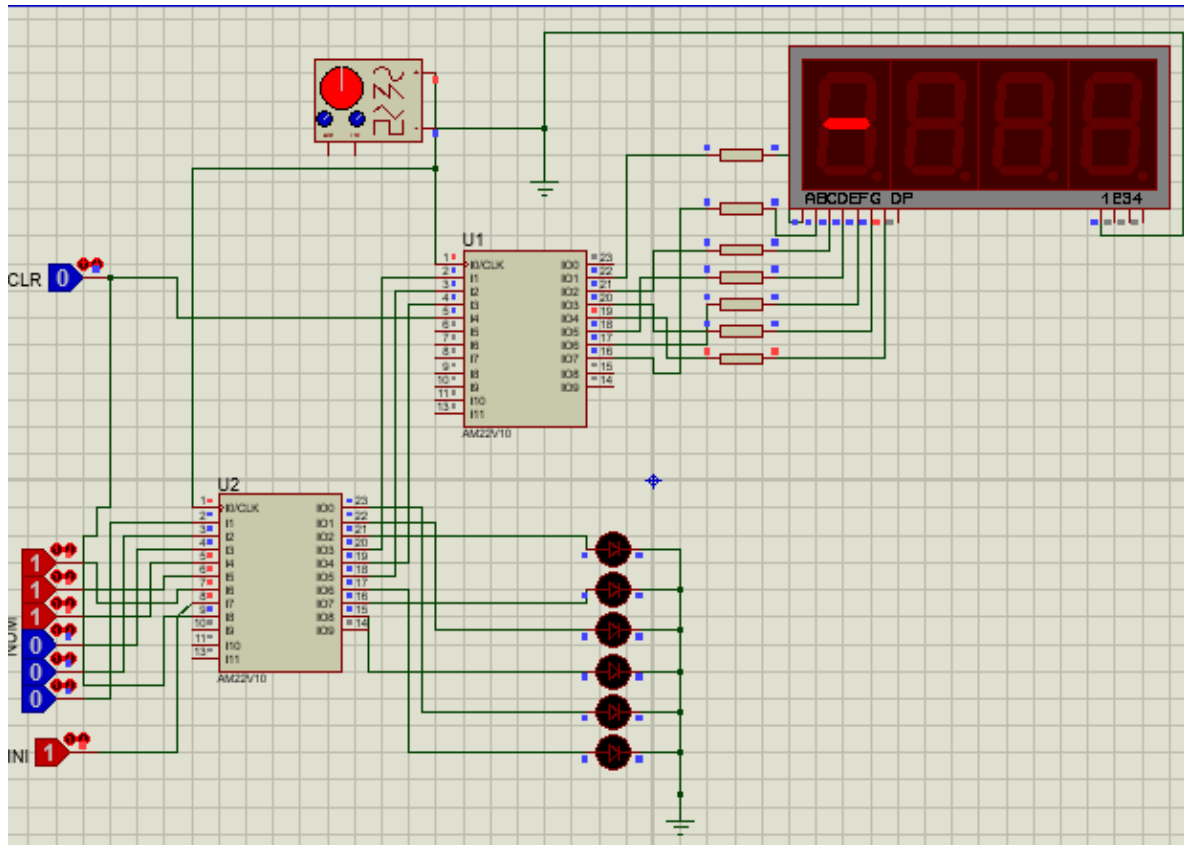
C22V10

clk = 1	24 * not used
ec = 2	23 = (count_1)
lb = 3	22 = s(6)
eb = 4	21 = s(4)
clr = 5	20 = s(1)
not used * 6	19 = s(0)
not used * 7	18 = s(3)
not used * 8	17 = s(2)
not used * 9	16 = s(5)
not used * 10	15 = (count_0)
not used * 11	14 = (count_2)
not used * 12	13 * not used

CIRCUITO ENTERO







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?

Aproximadamente, 14

3. ¿Cuántos pines de entrada/salida de cada PLD 22V10 se usan en el diseño?

Para el primero 19, y para el segundo 15

4. ¿Cuántos términos producto ocupan las ecuaciones para cada señal de salida?

Para el primero 34, y para el segundo 22

5. ¿Cuántos FF' s ocupa el autómata de control de la microarquitectura?

2

6. ¿Qué puedes concluir de esta práctica?

Los autómatas sin duda tienen un gran potencial cuando se combinan con otras cosas, un contador de 1 básicamente es lo que se necesita en muchas cosas, como en Redes de Computadoras.