Teoría

1. c, d
2. b
3. b, c, d
4. b
5. b

E	$Q_2Q_1Q_0$	$Q_2'Q_1'Q_0'$	$T_2T_1T_0$
0	0 0 0	1 1 1	1 1 1
0	1 1 1	1 1 0	0 0 1
0	1 1 0	1 0 1	0 1 1
0	1 0 1	1 0 0	0 0 1
0	1 0 0	0 1 1	1 1 1
0	0 1 1	0 1 0	0 0 1
0	0 1 0	0 0 1	0 1 1
0	0 0 1	0 0 0	0 0 1

Es $\bar{Q}_0$

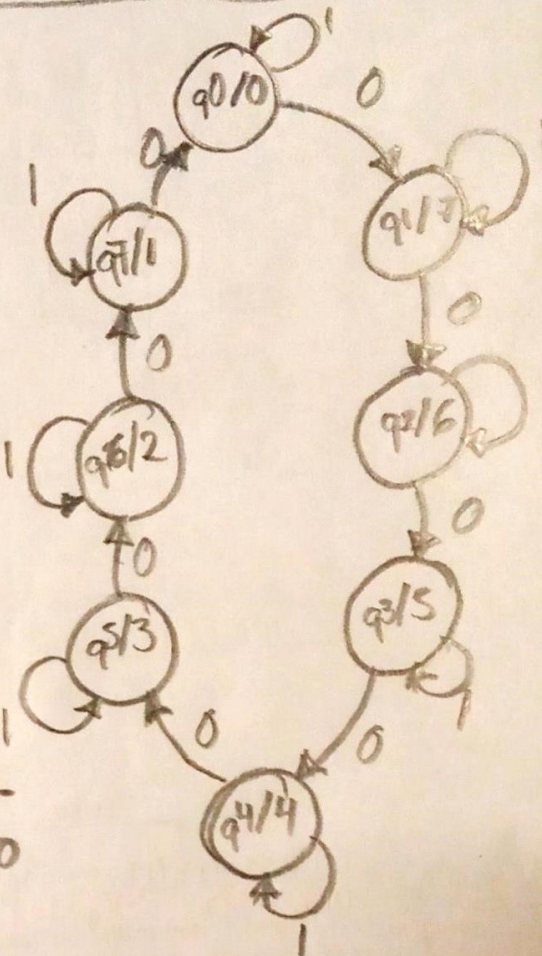
Como el ff T con 0 mantiene a Q se reduce el mapa

Mapa - $\bar{E}$

$T_2 = \bar{E}\bar{Q}_1\bar{Q}_0$

$T_1 = \bar{E}\bar{Q}_0$

$T_0 = \bar{E}$

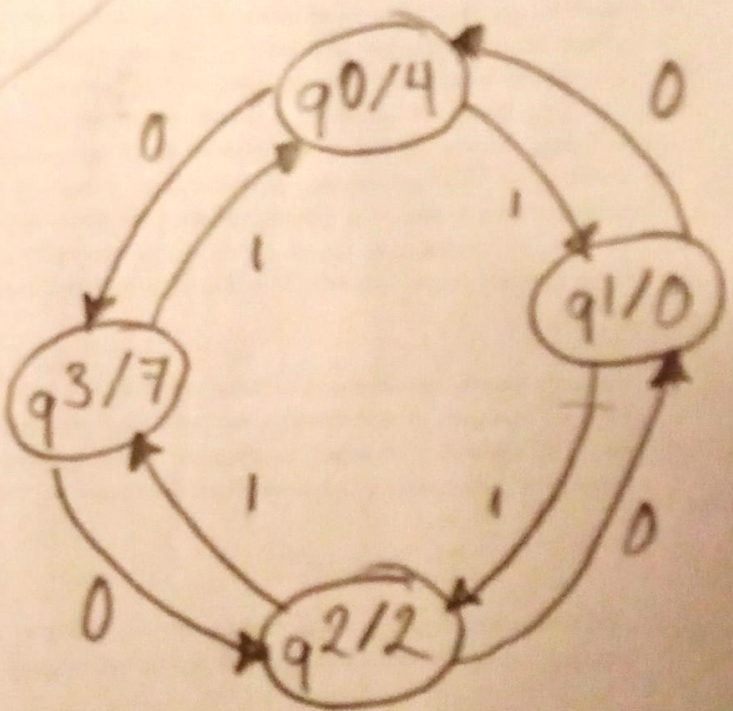


Para T_2 (Se elimina E)

Q_2 \ Q_1Q_0	00	01	11	10
0				
1				

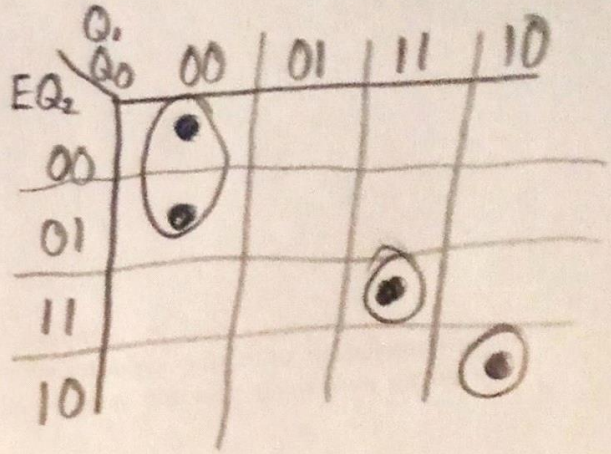
$\bar{Q}_1\bar{Q}_0$

```
architecture Contador of C3 is
begin
  process (CLK, CLR)
  begin
    if CLR = '1' then
       S <= "000";
    elsif rising_edge(CLK) then
       if E = '0' then
          S <= S - 1;
       else
          S <= S;
       end if;
    end if;
  end process;
end architecture;
```

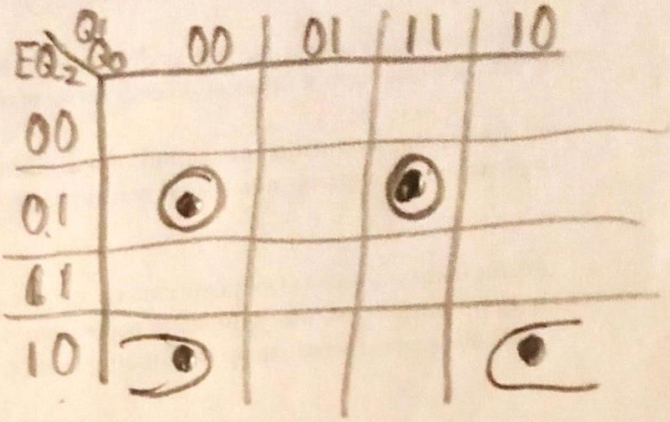


E	$Q_2Q_1Q_0$	$D_2D_1D_0$
1	1 0 0	0 0 0
1	0 0 0	0 1 0
1	0 1 0	1 1 1
1	1 1 1	1 0 0
0	1 0 0	1 1 1
0	0 0 0	1 0 0
0	0 1 0	0 0 0
0	1 1 1	0 1 0

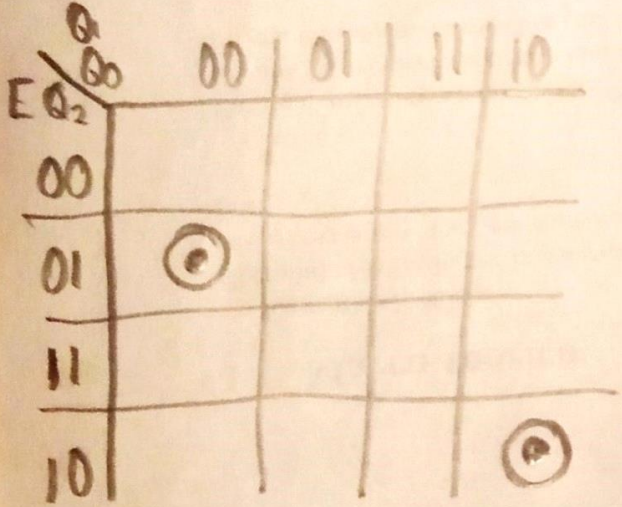
Para D_2



Para D_1



Para D_0



$$D_2 = \bar{E}\bar{Q}_1\bar{Q}_0 + EQ_2Q_1Q_0 + E\bar{Q}_2Q_1\bar{Q}_0$$

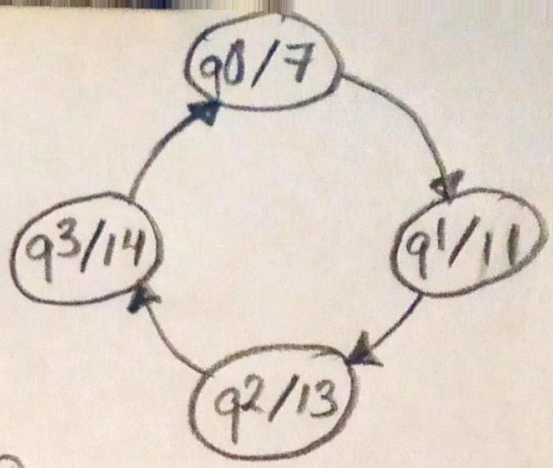
$$D_1 = \bar{E}Q_2\bar{Q}_1\bar{Q}_0 + \bar{E}Q_2Q_1Q_0 + E\bar{Q}_2\bar{Q}_0$$

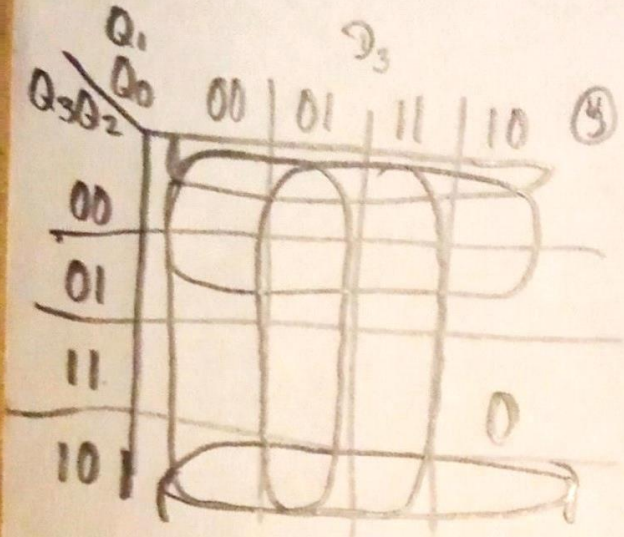
$$D_0 = \bar{E}Q_2\bar{Q}_1\bar{Q}_0 + E\bar{Q}_2Q_1\bar{Q}_0$$

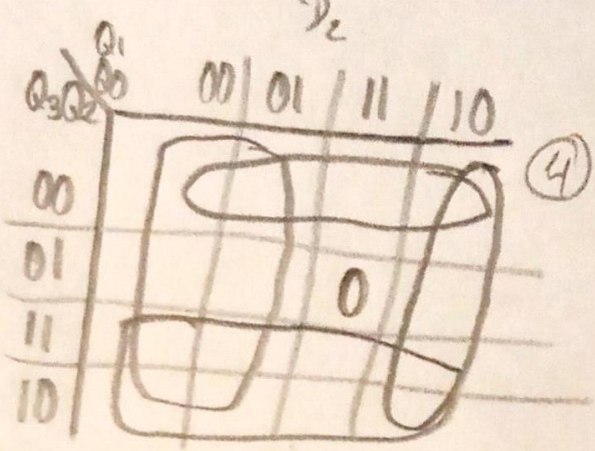
```
architecture Contador of Cont is
begin
  process (CLK, CLR)
  begin
    if CLR = '1' then
       D <= "000";
    elsif rising-edge(CLK) then
       if E = '1' then
          case D is
            when "100" => D <= "000";
            when "000" => D <= "010";
            when "010" => D <= "111";
            when others => D <= "100";
          end case;
       else
          case D is
            when "100" => D <= "111";
            when "111" => D <= "010";
            when "010" => D <= "000";
            when others => D <= "100";
          end case;
       end if;
    end if
  end process;
end architecture;
```

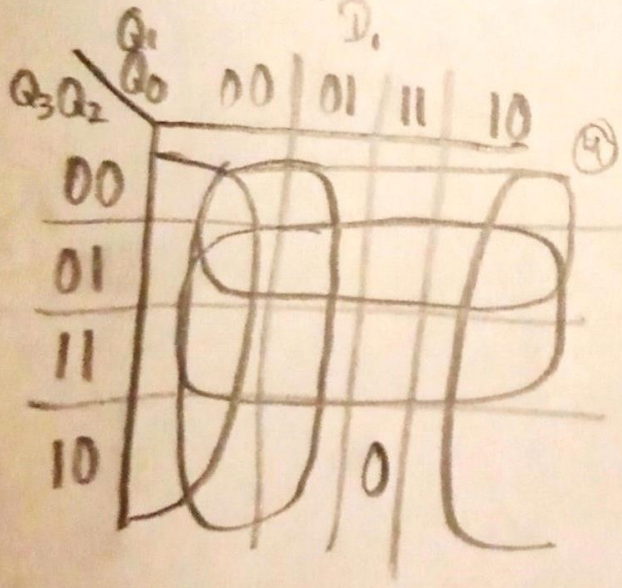
$Q_3 Q_2 Q_1 Q_0$	$D_3 D_2 D_1 D_0$
0 1 1 1	1 0 1 1
1 0 1 1	1 1 0 1
1 1 0 1	1 1 1 0
1 1 1 0	0 1 1 1

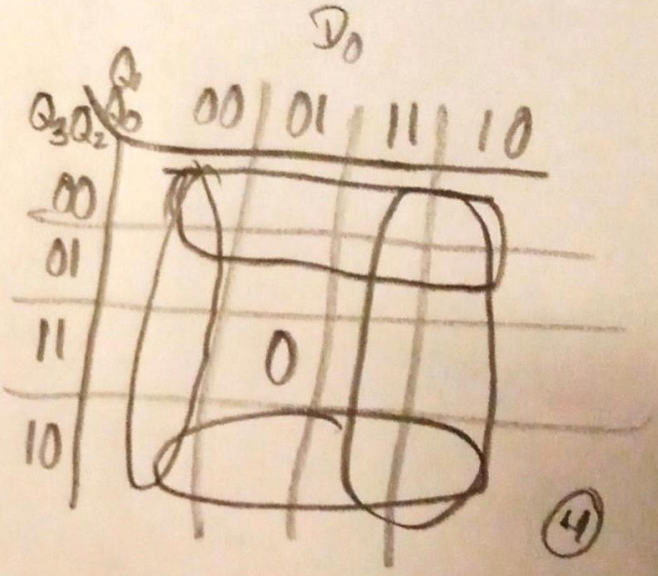
Dibujamos los casos











$$D_3 = \overline{Q_3} + \overline{Q_1} + Q_0 + \overline{Q_2}$$

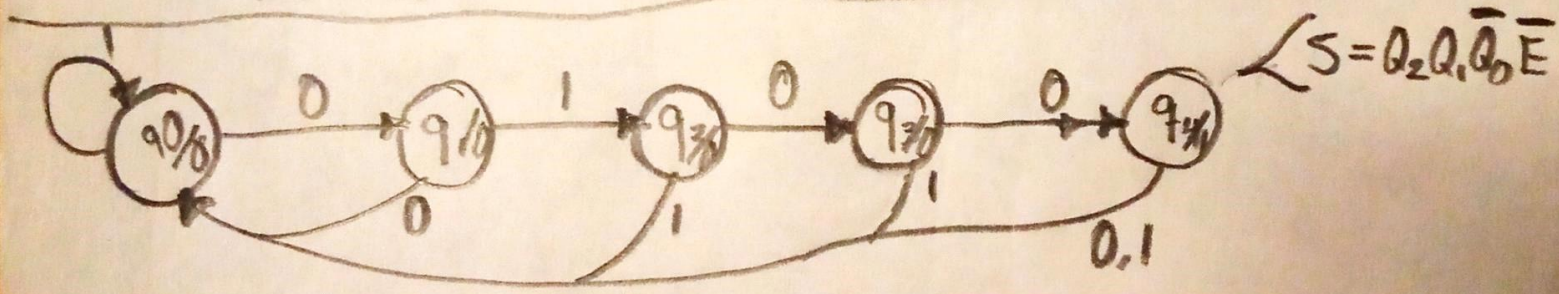
$$D_2 = \overline{Q_1} + Q_3 + \overline{Q_3}\,\overline{Q_2} + Q_1\overline{Q_0}$$

$$D_1 = \overline{Q_3} + Q_2 + \overline{Q_0} + \overline{Q_1}$$

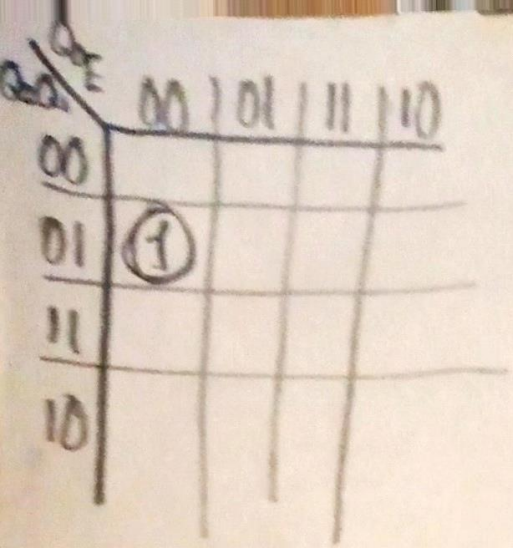
$$D_0 = \overline{Q_3} + Q_1 + \overline{Q_1}\,\overline{Q_0} + Q_3\overline{Q_2}$$

5

```
architecture Contador of Anillo is
begin
  process (CLK, CLR)
  begin
    if CLR = '1' then
        S <= "0000";
    elsif rising_edge(CLK) then
        case S is
          when "0111" => S <= "1011";
          when "1011" => S <= "1101";
          when "1101" => S <= "1110";
          when others => S <= "0111";
        end case;
    end if;
  end process;
end architecture;
```

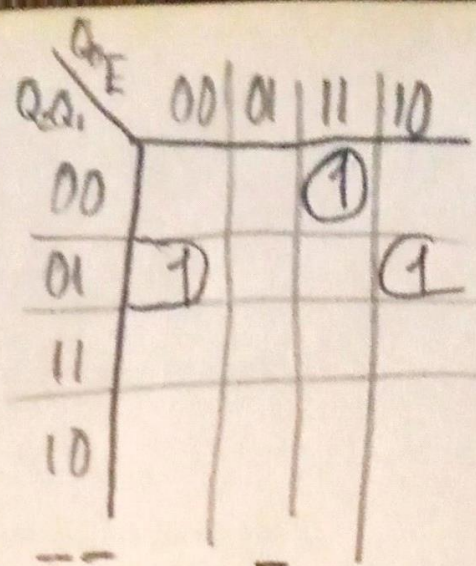


	$Q_2Q_1Q_0$	E	$D_2D_1D_0$	$Q_2Q_1Q_0$	E	$D_2D_1D_0$
q0	000	0	001	000	1	000
q1	001	0	000	001	1	011
q^2	011	0	010	011	1	000
q^3	010	0	110	010	1	000
q4	110	0	000	110	1	000



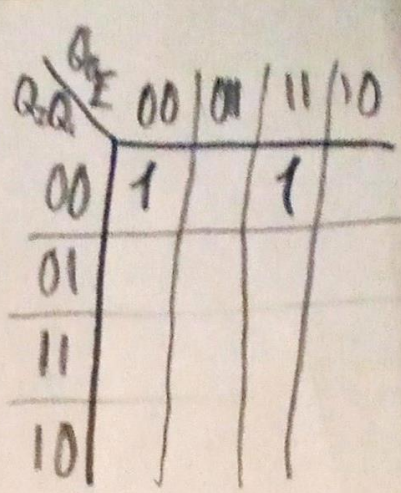
Q_2Q_1 \ Q_0E	00	01	11	10
00				
01	(1)			
11				
10				

$D_2 = \bar{Q}_2 Q_1 \bar{Q}_0 \bar{E}$



Q_2Q_1 \ Q_0E	00	01	11	10
00			(1)	
01	[1]			(1)
11				
10				

$D_1 = \bar{Q}_2\bar{Q}_1 Q_0 E + \bar{Q}_2 Q_1 \bar{E}$



Q_2Q_1 \ Q_0E	00	01	11	10
00	1		1	
01				
11				
10				

$D_0 = \bar{Q}_2\bar{Q}_1\bar{Q}_0\bar{E} + \bar{Q}_2\bar{Q}_1 Q_0 E$

```
architecture Contador of C3 is
begin
  process (CLR, CLK)
    if CLR = '1' then
       F <= "000";

    elsif rising_edge(CLK) then
       case F is
         when "000" =>
           if E = '0' then
              F <= "001";
           else
              F <= "000";
           end if;
         when "001" =>
           if E = '0' then
              F <= "000";
           else
              F <= "011";
           end if;
         when "011" =>
           if E = '0' then
              F <= "000";
           else
              F <= "010";
           end if;
         when "010" =>
           if E = '0' then
              F <= "110";
           else
              F <= "000";
           end if;
         when others =>
              F <= "000";
       end case;
    end if;

    S <= F(2) and F(1) and not(F(0))
         and not E;

  end process;
end architecture;
```