

Pedro Covelo Rodríguez --- 39493020L  
Computadores 1.  
Álvaro García Sánchez-----70924450V

Trabajo  
Grupo A3.

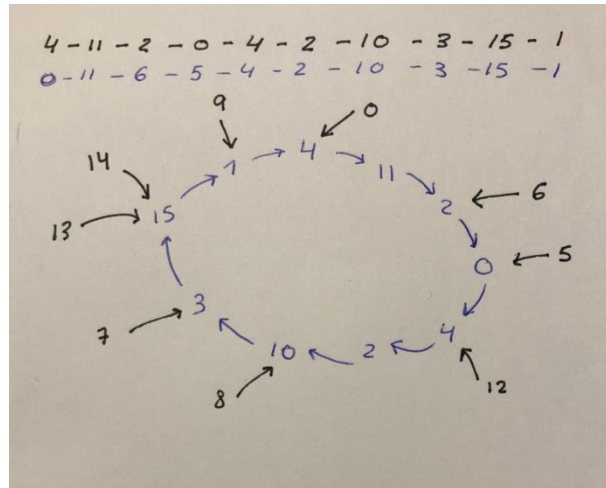
### **Índice:**

1. DISEÑO DEL CONTADOR:
  - . Diagrama de estados.
  - . Tabla de transiciones.
  - . Mapas de Karnaugh.
2. DISEÑO DEL CIRCUITO.
3. DISEÑO DEL CONVERTOR:
  - . Tabla del convertor.
  - . Mapas de Karnaugh del convertor.
4. VERILOG:
  - . Biestable JK.
  - . Creación del módulo principal del contador de 4 bits con biestables JK.
  - . Puertas.
  - . Declaración de los biestables.
  - . Cambios en la circuitería
  - . Módulo de pruebas.
  - . Salida en el terminal.
5. GTKWAVE
6. PRESUPUESTO Y COSTES DEL CIRCUITO

## 1. DISEÑO DEL CONTADOR

### . DIAGRAMA DE ESTADOS:

Se le asignan otros números a los números que no se encuentran en la lista:



### . TABLA CON LAS TRANSICIONES NECESARIAS:

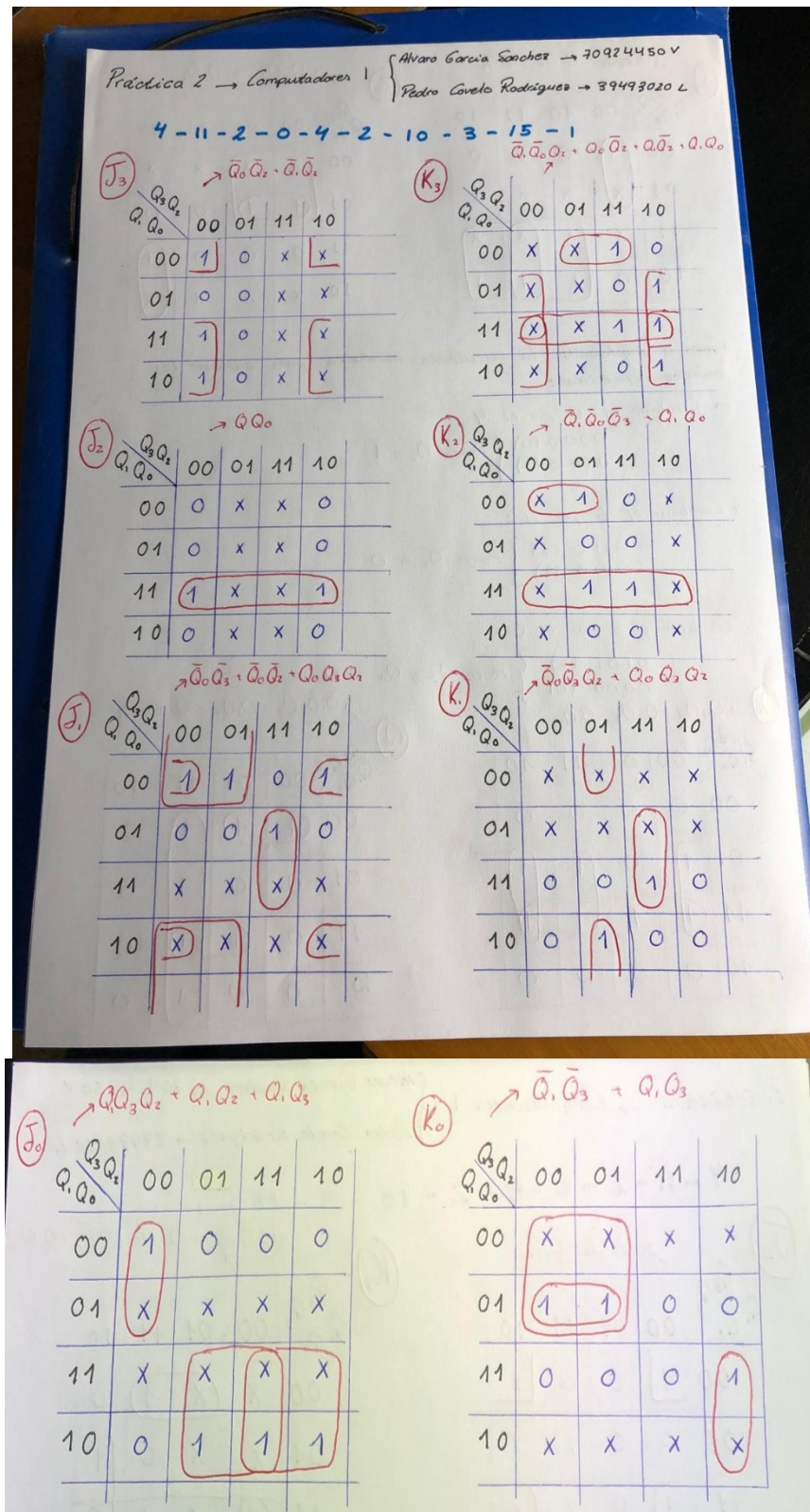
Tabla con las transiciones del biestable JK

	Salida	Sgte.	J3 K3	J2 K2	J1 K1	J0 K0
0	0000	1011 (11)	1x	0x	1x	1x
1	0001	0000 (0)	0x	0x	0x	x1
2	0010	1010 (10)	1x	0x	x0	0x
3	0011	1111 (15)	1x	1x	x0	x0
4	0100	0010 (2)	0x	x1	1x	0x
5	0101	0100 (4)	0x	x0	0x	x1
6	0110	0101 (5)	0x	x0	x1	1x
7	0111	0011 (3)	0x	x1	x0	x0
8	1000	1010 (10)	x0	0x	1x	0x
9	1001	0001 (1)	x1	0x	0x	x0
10	1010	0011 (3)	x1	0x	x0	1x
11	1011	0110 (6)	x1	1x	x0	x1
12	1100	0100 (4)	x1	x0	0x	0x
13	1101	1111 (15)	x0	x0	1x	x0

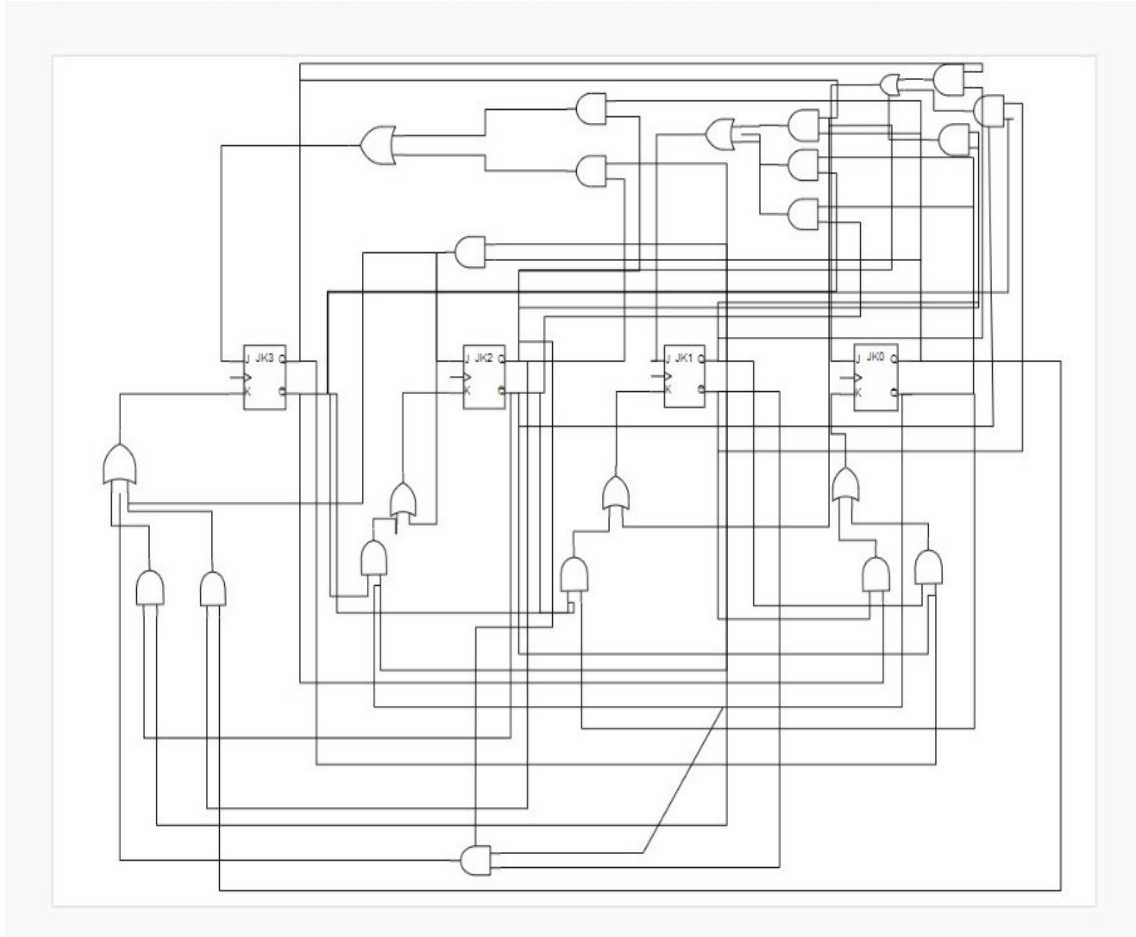
14	1110	1111 (15)	x0	x0	x0	1x
15	1111	0001 (1)	x1	x1	x1	x0

### MAPAS DE KARNAUGH:

A partir de la tabla de transiciones obtenemos los siguientes mapas de Karnaugh y de ellos se pueden sacar las ecuaciones.



## 2. DISEÑO DEL CIRCUITO :



### 3. DISEÑO DEL CONVERSIONOR:

#### TABLA DEL CONVERSIONOR:

	Salida (Q3Q2Q1Q0)	Q3*	Q2*	Q1*	Q0*
jj 0	0000	0	1	0	0
1	0001	0	0	0	1
2	0010	0	0	1	0
3	0011	0	0	1	1
4	0100	0	1	0	0
5	0101	0	0	0	0
6	0110	0	0	1	0
7	0111	0	1	1	1
8	1000	1	0	0	0
9	1001	1	0	0	1
10	1010	1	0	1	0
11	1011	1	0	1	1
12	1100	1	1	0	0
13	1101	1	1	0	1
14	1110	1	1	1	0
15	1111	1	1	1	1

#### MAPAS DE KARNAUGH DEL CONVERSIONOR.

Vamos a implementar la circuitería necesaria para calcular los números repetidos:

→ Cambiar el 0 por el 4:

$$\begin{matrix} 0000 (0) \\ 0100 (4) \end{matrix} \left. \vphantom{\begin{matrix} 0000 (0) \\ 0100 (4) \end{matrix}} \right\} \text{Poner } Q_2 \text{ a } 1$$

→ Cambiar el 6 por el 2:

$$\begin{matrix} 0110 (6) \\ 0010 (2) \end{matrix} \left. \vphantom{\begin{matrix} 0110 (6) \\ 0010 (2) \end{matrix}} \right\} \text{Poner } Q_2 \text{ a } 0$$

→ Cambiar el 5 por el 0:

$$\begin{matrix} 0101 (5) \\ 0000 (0) \end{matrix} \left. \vphantom{\begin{matrix} 0101 (5) \\ 0000 (0) \end{matrix}} \right\} \text{Poner } Q_2 \text{ y } Q_0 \text{ a } 0$$

$Q_0 \rightarrow Q_0 Q_2 + Q_0 Q_3 + Q_1 Q_0$

$Q_2 \rightarrow Q_1 Q_0 Q_3 + Q_2 Q_2 + Q_1 Q_0 Q_2$

$Q_2 \backslash Q_0$	00	01	11	10
00	0	0	0	0
01	1	0	1	1
11	1	1	1	1
10	0	0	0	0

$Q_2 \backslash Q_0$	00	01	11	10
00	1	1	1	0
01	0	0	1	0
11	0	1	1	0
10	0	0	1	0

## 4. VERILOG

### BIESTABLE JK

Creación de un módulo de un único biestable JK:

```
7 // -----
8
9 module JK (output reg Q, output wire NQ, input wire J, input wire K, input wire C);
10 not(NQ,Q);
11
12 initial
13 begin
14   Q=0;
15 end
16
17 always @(posedge C)
18   case ({J,K})
19     2'b10: Q=1;
20     2'b01: Q=0;
21     2'b11: Q=~Q;
22   endcase
23 endmodule
```

### CREACIÓN DEL MÓDULO PRINCIPAL DEL CONTADOR DE 4 BITS CON BIESTABLES JK:

```
26 // Creacion del modulo principal del contador de 4 bits con biestables JK
27
28 module Contador4bits(output wire [3:0] Q, input wire C);
29
30   wire [3:0] nQ; // Cables correspondientes a las salidas negadas de los biestables
31
32   wire Qt0, nQt0; // Cables que almacenan la salida temporal del biestable jk0
33
34   wire Qt2, nQt2; // Cables que almacenan la salida temporal del biestable jk2
35
36
37   wire j3,j2,j1,j0,k3,k2,k1,k0; // Cables de entrada a los biestables
38
39
40   wire nq0nq2, q1nq2, nq1nq0q2, q0nq2, q1q0, nq1nq0nq3, nq0nq3, nq0nq3q2, q0q3q2, nq1nq3, q1q3, nq3q2, q1q3nq2; // Cables in
41
42 // -----
```

### PUERTAS:

Se procede a la implementación de las puertas lógicas correspondientes a las entradas y a las salidas de los biestables:

```
42 // -----
43 // Implementacion de las puertas logicas correspo
44
45
46   and a1 (nq0nq2, nQt0, nQt2);
47   and a2 (q1nq2, Q[1], nQt2);
48   or J3 (j3, nq0nq2, q1nq2);
49
50   and J2 (j2, Q[1], Qt0);
51
52   and a3 (nq0nq3, nQt0, nQ[3]);
53   and a4 (nq0nq2, nQt0, nQt2);
54   and a5 (q0q3q2, Qt0, Q[3], Qt2);
55   or J1 (j1, nq0nq3, nq0nq2, q0q3q2);
56
57   and a6 (nq1nq3nq2, nQ[1], nQ[3], nQt2);
58   and a7 (q1q2, Q[1], Qt2);
59   and a8 (q1q3, Q[1], Q[3]);
60   or J0 (j0, nq1nq3nq2, q1q2, q1q3);
61
62   and a9 (nq1nq0q2, nQ[1], nQt0, Qt2);
63   and a10 (q0nq2, Qt0, nQt2);
64   and a11 (q1nq2, Q[1], nQt2);
65   and a12 (q1q0, Q[1], Qt0);
66   or K3 (k3, nq1nq0q2, q0nq2, q1nq2, q1q0);
67
68   and a13 (nq1nq0nq3, nQ[1], nQt0, nQ[3]);
69   and a14 (q1q0, Q[1], Qt0);
70   or K2 (k2, nq1nq0nq3, q1q0);
71
72   and a15 (nq0nq3q2, nQt0, nQ[3], Qt2);
73   and a16 (q0q3q2, Qt0, Q[3], Qt2);
74   or K1 (k1, nq0nq3q2, q0q3q2);
75
76   and a17 (nq1nq3, nQ[1], nQ[3]);
77   and a18 (q1q3nq2, Q[1], Q[3], nQt2);
78   or K0 (k0, nq1nq3, q1q3nq2);
79
```



### . DECLARACIÓN DE LOS BIESTABLES:

Procedemos a la implementación de los 4 biestables necesarios para la realización.

```
86
87     JK jk0 (Qt0, nQt0, j0, k0, C);
88     JK jk1 (Q[1], nQ[1], j1, k1, C);
89     JK jk2 (Qt2, nQt2, j2, k2, C);
90     JK jk3 (Q[3], nQ[3], j3, k3, C);
91
92
```

### . CAMBIOS EN LA CIRCUITERIA:

Procedemos a los cambios necesarios en la circuitería cambiando el 0 por el 4, el 6 por el 2 y el 5 por el 0

```
93
94 // -----
95 // Circuiteria que cambia el 0 por el 4, el 6 por el 2 y el 5 por el 0
96
97     and a19 (nq1nq0nq3, nQ[1], nQt0, nQ[3]);
98     and a20 (q3q2, Q[3], Qt2);
99     and a21 (q1q0q2, Q[1], Qt0, Qt2);
100     or Q2 (Q[2], nq1nq0nq3, q3q2, q1q0q2);
101
102     and a22 (q0nq2, Qt0, nQt2);
103     and a23 (q0q3, Qt0, Q[3]);
104     and a24 (q1q0, Q[1], Qt0);
105     or Q0 (Q[0], q0nq2, q0q3, q1q0);
106
```

### . MÓDULO DE PRUEBAS:

```
// -----
6
7 module testContador;
8     reg I, C;
9     wire [3:0] Q;
10    Contador4bits counter(Q,C);
11
12    always
13    begin
14        #10 C=~C;
15    end
16
17    initial
18    begin
19        $dumpfile("Contador_arbitrario.dmp");
20        $dumpvars(2, counter, Q);
21        $display ("Initial 1010");
22        $monitor ($time, " %d", Q);
23
24        C=0;
25        #500 $finish;
26    end
27 endmodule
28
29 // -----
30
```

### . SALIDA EN EL TERMINAL:

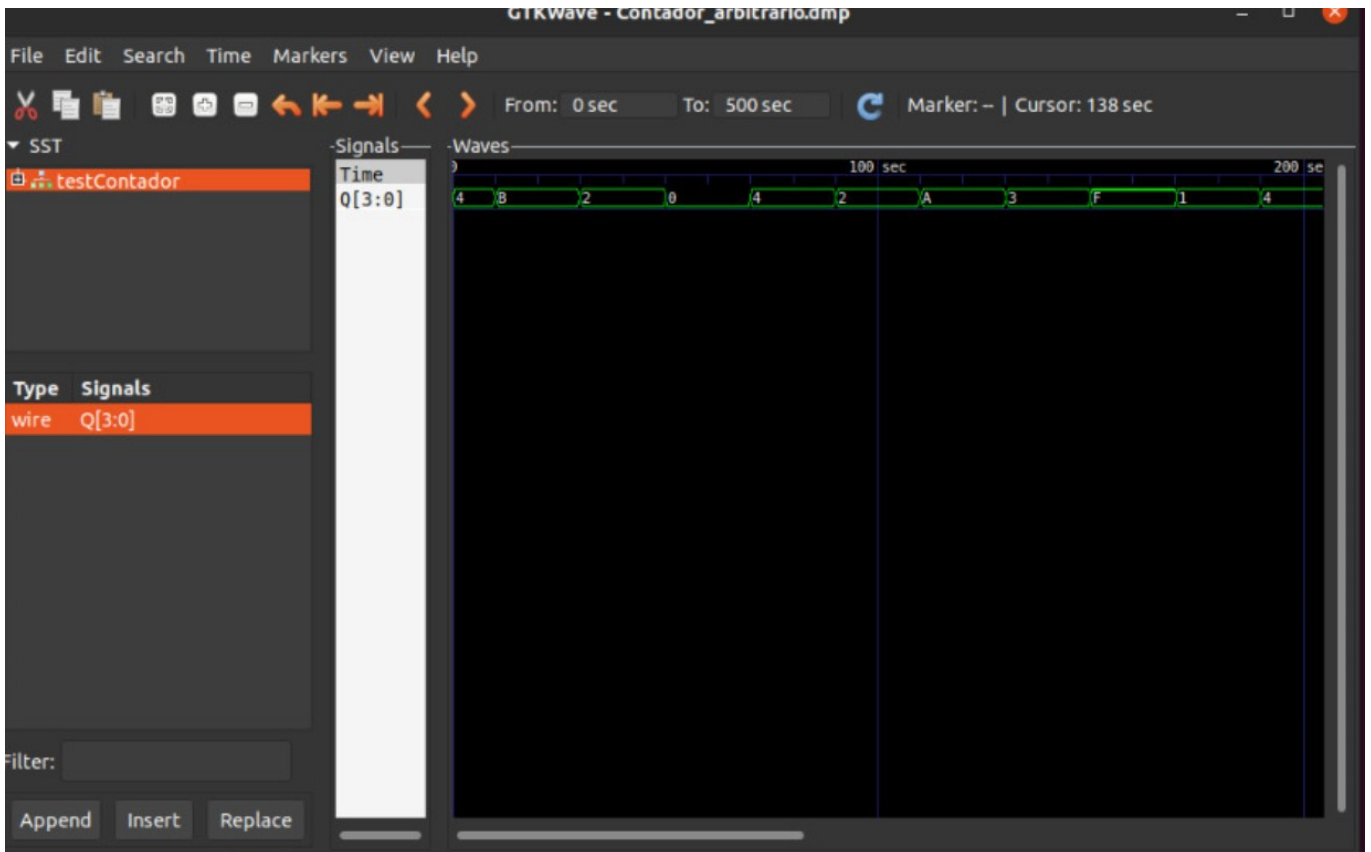
Finalmente esta sería la salida en el terminal con el módulo de pruebas.

```
alvaro@alvaro-X541UJ:~/Escritorio/Trabajo_contador$ iverilog FUNCIONA.v -o fun
alvaro@alvaro-X541UJ:~/Escritorio/Trabajo_contador$ ./fun
VCD info: dumpfile Contador_arbitrario.dmp opened for output.
Initial 1010
      0    4
     10   11
     30    2
     50    0
     70    4
     90    2
    110   10
    130    3
    150   15
    170    1
    190    4
    210   11
    230    2
    250    0
    270    4
    290    2
    310   10
    330    3
    350   15
    370    1
    390    4
    410   11
    430    2
    450    0
    470    4
    490    2
alvaro@alvaro-X541UJ:~/Escritorio/Trabajo_contador$
```



## 5. GTKWAVE

### 5.1. GTKWAVE.



## 6. COSTES DEL CIRCUITO

Analicemos brevemente los costes del circuito y el precio total de sus componentes:

MATERIAL	CANTIDAD EMPLEADA	PRECIO POR UNIDAD	PRECIO TOTAL
Biestable JK	4	0,647 €	2,588€
Puerta AND	25	0,537€	13,428€
Puerta OR	9	0,497€	4,473€
		TOTAL	20,489€

JK:SN74LS112AN-

Biestable, PresetyClear, Salida Complementaria, 74LS112, JK, 15 ns, 30 MHz, 8 mA

AND:SN74LS08N-

PuertaAND,74LS08,2Entradas,8mA,4.75Va5.25V,DIP-14

OR: SN74LS32N - Puerta OR, 74LS32, 2 Entradas, 8 mA, 4.75 V a 5.25 V, DIP-14