```
1   ----------------------------------------------------------------------------------
2   -- Company:         FCEFyN - UNC
3   -- Engineer:        Grupo 03 - ED1
4   --
5   -- Create Date:     20:51:56 05/15/2022
6   -- Design Name:     Trabajo Practico 3
7   -- Module Name:     SumadorRestador4bits - Behavioral
8   -- Project Name:    ALU con FPGA
9   -- Target Devices: Spartan 3E-100 CP-132
10  -- Tool versions:   Xilinx ISE 14.7
11  -- Description:     Sumador - Restador - AND - OR de 4 bits
12  --
13  -- Dependencies:
14  --
15  -- Revision:
16  -- Revision 0.01 - File Created
17  -- Additional Comments: (Clock agregado para un control mas estable en las operaciones)
18  --
19  ----------------------------------------------------------------------------------
20  library IEEE;
21  use IEEE.STD_LOGIC_1164.ALL;
22  use IEEE.STD_LOGIC_ARITH.ALL;
23  use IEEE.STD_LOGIC_UNSIGNED.ALL;
24
25  entity SumadorRestador4bits is
26      Port ( BTN1 : in  STD_LOGIC;
27             BTN2 : in STD_LOGIC;
28             CLOCK : in STD_LOGIC;
29             NUM1 : in  STD_LOGIC_VECTOR (3 downto 0);
30             NUM2 : in  STD_LOGIC_VECTOR (3 downto 0);
31             RESULTADO : out  STD_LOGIC_VECTOR (3 downto 0);
32             CARRY : out  STD_LOGIC;
33             NEG : out  STD_LOGIC );
34     end SumadorRestador4bits;
35
36  architecture Behavioral of SumadorRestador4bits is
37
38  signal aux : STD_LOGIC_VECTOR (3 downto 0) := "0000";
39  signal acarreo : STD_LOGIC := '0';
40  signal negativo : STD_LOGIC := '0';
41
42  begin
43
44     process(CLOCK)
45     begin
46
47     if CLOCK = '1' and CLOCK'event then
48
49     acarreo <= '0';                                    -- Reseteamos los valores de...
50     negativo <= '0';                                   -- ...acarreo y negativo para...
51                                                        -- ...evitar arrastre del valor.
52        if BTN1 = '0' then
53
54           if BTN2 = '0' then
55
56              aux <= NUM1 + NUM2;                       -- SUMA
57
```

```
58                if (NUM1 AND NUM2) /= "0000" then       -- Verifica si hubo Acarreo
59                   acarreo <= '1';
60                else
61                   acarreo <= '0';
62                end if;
63
64             else
65
66                aux <= NUM1 AND NUM2;                    -- AND
67
68             end if;
69
70          else
71
72             if BTN2 = '0' then                          -- OR
73
74                aux <= NUM1 OR NUM2;
75
76             else
77
78                if NUM1 < NUM2 then                      -- Verifica si la resta...
79                   negativo <= '1';                      -- ...dará número negativo...
80                   aux <= NUM2 - NUM1;                   -- ...y en ese caso invierte...
81                else                                     -- ...el orden
82                   negativo <= '0';
83                   aux <= NUM1 - NUM2;
84                end if;
85
86             end if;
87          end if;
88       end if;
89       end process;
90                                                          -- Asigna los valores al...
91    RESULTADO <= aux;                                     -- ...finalizar el proceso
92    CARRY <= acarreo;
93    NEG <= negativo;
94
95  end Behavioral;
```