

```

1  library ieee;
2  use ieee.std_logic_1164.all;
3  use ieee.std_logic_unsigned.all;
4  use work.archivo_componentes.all;
5
6  Entity Call_Center is
7      Port ( u1,u2,u3,u4,u5,u6,u7,u8: IN std_logic;
8            clock,resetn: IN std_logic; --Por default
9            start: IN std_logic;
10           REALIZAR_LLAMADA: IN std_logic;
11           FINALIZAR_LLAMADA: IN std_logic;
12           HISTORIAL_LLAMADAS: IN std_logic;
13           BORRAR: IN std_logic;
14           TECLADO: IN std_logic_vector (9 downto 0);
15           LLAMADA_EN_PROGRESO: BUFFER std_logic_vector (7 downto 0);
16           OCUPADO: OUT std_logic;
17           DISPLAY_1n: OUT std_logic_vector (3 downto 0);
18           DISPLAY_2n: OUT std_logic_vector (3 downto 0);
19           DISPLAY_3n: OUT std_logic_vector (3 downto 0);
20           DISPLAY_4n: OUT std_logic_vector (3 downto 0);
21           DISPLAY_5n: OUT std_logic_vector (3 downto 0);
22           DISPLAY_6n: OUT std_logic_vector (3 downto 0);
23       );
24 end Call_Center;
25
26 Architecture uno of Call_Center is
27
28     signal usuario: std_logic_vector (7 downto 0);
29     signal num_bcd,cuenta,sal_mux_1,sal_mux_2,sal_mux_3,sal_mux_4,sal_mux_5,sal_mux_6:
30     std_logic_vector (3 downto 0);
31     signal direc_u1,direc_u2,direc_u3,direc_u4,direc_u5,direc_u6,direc_u7,direc_u8:
32     std_logic_vector (3 downto 0);
33     signal dhist_1,dhist_2,dhist_3,dhist_4,dhist_5,dhist_6,dhist_7,dhist_8: std_logic_vector
34     (3 downto 0);
35     signal cuenta_oc,cuenta_hist: std_logic_vector (3 downto 0);
36     signal incremento_u1,incremento_u2,incremento_u3,incremento_u4,incremento_u5,
37     incremento_u6,incremento_u7,incremento_u8: std_logic;
38     signal reset_u1,reset_u2,reset_u3,reset_u4,reset_u5,reset_u6,reset_u7,reset_u8: std_logic;
39     signal ihist_u1,ihist_u2,ihist_u3,ihist_u4,ihist_u5,ihist_u6,ihist_u7,ihist_u8: std_logic;
40     signal rhist_u1,rhist_u2,rhist_u3,rhist_u4,rhist_u5,rhist_u6,rhist_u7,rhist_u8: std_logic;
41     signal igual_oc,icuenta0,icuenta1,icuenta2,icuenta3,icuenta4,icuenta5,igual_hist1:
42     std_logic;
43     signal boton,uf,alguna_llamada,cont6,Fu,num_existe,Re_oc,Re_des,dir_hist_igual,reset_oc,
44     reset_hist,disp_select: std_logic;
45     signal iusuario1,iusuario2,iusuario3,iusuario4,iusuario5,iusuario6,iusuario7,iusuario8:
46     std_logic;
47     signal iusuario_ant1,iusuario_ant2,iusuario_ant3,iusuario_ant4,iusuario_ant5,
48     iusuario_ant6,iusuario_ant7,iusuario_ant8: std_logic;
49     signal igual_h1,igual_h2,igual_h3,igual_h4,igual_h5,igual_h6,igual_h7,igual_h8: std_logic;
50     signal incrementa,cuenta_a_0: std_logic;
51     signal selec_u1,selec_u2,selec_u3,selec_u4,selec_u5,selec_u6,selec_u7,selec_u8:
52     std_logic_vector (1 downto 0);
53     signal adress_u1,adress_u2,adress_u3,adress_u4,adress_u5,adress_u6,adress_u7,adress_u8:
54     std_logic_vector (3 downto 0);
55     signal historial_u1,historial_u2,historial_u3,historial_u4,historial_u5,historial_u6,
56     historial_u7,historial_u8: std_logic_vector (23 downto 0);
57     signal numero_actual,numero_ram: std_logic_vector (23 downto 0);
58     signal enable_reg: std_logic_vector (5 downto 0);
59     signal sal_reg_1,sal_reg_2,sal_reg_3,sal_reg_4,sal_reg_5,sal_reg_6: std_logic_vector (3
60     downto 0);
61     signal num1hist,num2hist,num3hist,num4hist,num5hist,num6hist: std_logic_vector (3 downto
62     0);
63     signal clock_ff_d,reset_ff_d,led1,led2,led3,led4,led5,led6,led7,led8,reset_reg: std_logic;
64     signal we_u1,we_u2,we_u3,we_u4,we_u5,we_u6,we_u7,we_u8: std_logic;
65     signal Fu_ad1,Fu_ad2,Fu_ad3,Fu_ad4,Fu_ad5,Fu_ad6,Fu_ad7,Fu_ad8: std_logic;
66     signal cero: std_logic_vector (3 downto 0):="0000";
67     signal uno: std_logic_vector (3 downto 0):="0001";
68     signal dos: std_logic_vector (3 downto 0):="0010";
69     signal tres: std_logic_vector (3 downto 0):="0011";
70     signal cuatro: std_logic_vector (3 downto 0):="0100";
71     signal cinco: std_logic_vector (3 downto 0):="0101";
72     signal CLOCK_1MHz,CLOCK_100kHz,CLOCK_10kHz,CLOCK_1kHz,CLOCK_100Hz,CLOCK_10Hz,CLOCK_1Hz:
73     std_logic;
74     --Numero del usuario 1
75     signal num_u1: std_logic_vector (23 downto 0):="011001110010010100011001";
76     --Numero del usuario 2
77     signal num_u2: std_logic_vector (23 downto 0):="011001110101100000110111";
78     --Numero del usuario 3
79     signal num_u3: std_logic_vector (23 downto 0):="011001110010011001010101";
80     --Numero del usuario 4
81     signal num_u4: std_logic_vector (23 downto 0):="011001110001001101000010";

```

```

68 --Numero del usuario 5
69 Signal num_u5: std_logic_vector(23 downto 0):="011001110010100110000011" ;
70 --Numero del usuario 6
71 Signal num_u6: std_logic_vector(23 downto 0):="011001110010001101010110" ;
72 --Numero del usuario 7
73 Signal num_u7: std_logic_vector(23 downto 0):="011001110001010000111000" ;
74 --Numero del usuario 8
75 Signal num_u8: std_logic_vector(23 downto 0):="011001110100010001010011" ;
76 Begin
77 uf <= (((u1) and not(u2) and not(u3) and not(u4) and not(u5) and not(u6) and not(u7)
and not(u8)) or
78 (not (u1) and (u2) and not(u3) and not(u4) and not(u5) and not(u6) and not(u7)
and not(u8)) or
79 (not (u1) and not(u2) and (u3) and not(u4) and not(u5) and not(u6) and not(u7)
and not(u8)) or
80 (not (u1) and not(u2) and not(u3) and (u4) and not(u5) and not(u6) and not(u7)
and not(u8)) or
81 (not (u1) and not(u2) and not(u3) and not(u4) and (u5) and not(u6) and not(u7)
and not(u8)) or
82 (not (u1) and not(u2) and not(u3) and not(u4) and not(u5) and (u6) and not(u7)
and not(u8)) or
83 (not (u1) and not(u2) and not(u3) and not(u4) and not(u5) and not(u6) and (u7)
and not(u8)) or
84 (not (u1) and not(u2) and not(u3) and not(u4) and not(u5) and not(u6) and not(
u7) and (u8)) );
85 usuario<=u8&u7&u6&u5&u4&u3&u2&u1;--usuario seleccionado
86 numero_actual<=sal_reg_1&sal_reg_2&sal_reg_3&sal_reg_4&sal_reg_5&sal_reg_6;--numero
actual
87 num1hist<=numero_ram(23)&numero_ram(22)&numero_ram(21)&numero_ram(20);--numero 1 del
historial que se obtiene de la RAM
88 num2hist<=numero_ram(19)&numero_ram(18)&numero_ram(17)&numero_ram(16);--el numero de
la ram puede usarse para mostrar el historial
89 num3hist<=numero_ram(15)&numero_ram(14)&numero_ram(13)&numero_ram(12);--o para
conocer el numero anterior
90 num4hist<=numero_ram(11)&numero_ram(10)&numero_ram(9)&numero_ram(8);
91 num5hist<=numero_ram(7)&numero_ram(6)&numero_ram(5)&numero_ram(4);
92 num6hist<=numero_ram(3)&numero_ram(2)&numero_ram(1)&numero_ram(0);
93 boton <= (num_bcd(3) or num_bcd(2) or num_bcd(1) or num_bcd(0)) and uf;--además de
presionar un botón un usuario está seleccionado
94 alguna_llamada <= (LLAMADA_EN_PROGRESO(0) or LLAMADA_EN_PROGRESO(1) or
LLAMADA_EN_PROGRESO(2) or
95 LLAMADA_EN_PROGRESO(3) or LLAMADA_EN_PROGRESO(4) or
LLAMADA_EN_PROGRESO(5) or
96 LLAMADA_EN_PROGRESO(6) or LLAMADA_EN_PROGRESO(7));
97 cont6 <= (not cuenta(3) and cuenta(2) and not cuenta(1) and cuenta(0));
98 Fu <= (uf and alguna_llamada and FINALIZAR_LLAMADA);--usuario seleccionado y alguna
llamada en progreso, sólo ahí puedo finalizar
99 num_existe <= (iusuario1 or iusuario2 or iusuario3 or iusuario4 or iusuario5 or
iusuario6 or iusuario7 or iusuario8);--si el numero actual es igual a al menos un número
en el registro
100 Re_oc <= (num_existe and alguna_llamada and REALIZAR_LLAMADA);--Para mostrar el
ocupado, si el numero escrito existe y ademas existe alguna llamada y presiono realizar
llamada
101 Re_des <= (num_existe and not (alguna_llamada) and REALIZAR_LLAMADA);--para realizar
la llamada el numero escrito debe existir, no debe haber ninguna llamada y debo
presionar realizar llamada
102 dir_hist_igual <= (igual_h1 or igual_h2 or igual_h3 or igual_h4 or igual_h5 or
igual_h6 or igual_h7 or igual_h8);--si la direccion de alguno de los historiales ya se
iguala a la direccion de las ram en las que voy guardando
103 Fu_ad1 <= (uf and u1 and LLAMADA_EN_PROGRESO(0)); --solo finalizo la llamada si hay
una en progreso y de paso el adress de la ram debe aumentar a la siguiente para guardar
104 Fu_ad2 <= (uf and u2 and LLAMADA_EN_PROGRESO(1));
105 Fu_ad3 <= (uf and u3 and LLAMADA_EN_PROGRESO(2));
106 Fu_ad4 <= (uf and u4 and LLAMADA_EN_PROGRESO(3));
107 Fu_ad5 <= (uf and u5 and LLAMADA_EN_PROGRESO(4));
108 Fu_ad6 <= (uf and u6 and LLAMADA_EN_PROGRESO(5));
109 Fu_ad7 <= (uf and u7 and LLAMADA_EN_PROGRESO(6));
110 Fu_ad8 <= (uf and u8 and LLAMADA_EN_PROGRESO(7));
111 DISPLAY_1n <= sal_mux_1;-- tomar en cuenta para la particion funcional
112 DISPLAY_2n <= sal_mux_2;
113 DISPLAY_3n <= sal_mux_3;
114 DISPLAY_4n <= sal_mux_4;
115 DISPLAY_5n <= sal_mux_5;
116 DISPLAY_6n <= sal_mux_6;
117 G0: encoder_Dec_BCD port map (TECLADO,num_bcd);
118 G1_1: contador_14bits port map (incrementa, cuenta_a_0, cuenta);--contador para
cuando presiono una tecla
119 --Contador para el adress del usuario n
120 G1_u1: contador_14bits port map (incremento_u1,reset_u1,direc_u1);
121 G1_u2: contador_14bits port map (incremento_u2,reset_u2,direc_u2);
122 G1_u3: contador_14bits port map (incremento_u3,reset_u3,direc_u3);

```

```

123     G1_u4: contador_1_4bits port map (incremento_u4,reset_u4,direc_u4);
124     G1_u5: contador_1_4bits port map (incremento_u5,reset_u5,direc_u5);
125     G1_u6: contador_1_4bits port map (incremento_u6,reset_u6,direc_u6);
126     G1_u7: contador_1_4bits port map (incremento_u7,reset_u7,direc_u7);
127     G1_u8: contador_1_4bits port map (incremento_u8,reset_u8,direc_u8);
128     --Contador para el historial address del usuario n
129     G1_hist_u1: contador_1_4bits port map (ihist_u1,rhist_u1,dhist_1);
130     G1_hist_u2: contador_1_4bits port map (ihist_u2,rhist_u2,dhist_2);
131     G1_hist_u3: contador_1_4bits port map (ihist_u3,rhist_u3,dhist_3);
132     G1_hist_u4: contador_1_4bits port map (ihist_u4,rhist_u4,dhist_4);
133     G1_hist_u5: contador_1_4bits port map (ihist_u5,rhist_u5,dhist_5);
134     G1_hist_u6: contador_1_4bits port map (ihist_u6,rhist_u6,dhist_6);
135     G1_hist_u7: contador_1_4bits port map (ihist_u7,rhist_u7,dhist_7);
136     G1_hist_u8: contador_1_4bits port map (ihist_u8,rhist_u8,dhist_8);
137     G1_ocupado: contador_1_4bits port map (CLOCK_1Hz,reset_oc,cuenta_oc);--cuenta hasta 2
segundos
138     G1_historial: contador_1_4bits port map (CLOCK_1Hz,reset_hist,cuenta_hist);--cuenta 4
segundos
139     G2_resta_u1: mux_2selector_restador port map (direc_u1,direc_u1,dhist_1,selec_u1,
address_u1);--la RAM puede trabajar de 3 maneras
140     G2_resta_u2: mux_2selector_restador port map (direc_u2,direc_u2,dhist_2,selec_u2,
address_u2);--guardando el numero y aumentando el address
141     G2_resta_u3: mux_2selector_restador port map (direc_u3,direc_u3,dhist_3,selec_u3,
address_u3);--verificando la direccion anterior (ultimo numero guardado para saber a que
numero llamo determinado usuario)
142     G2_resta_u4: mux_2selector_restador port map (direc_u4,direc_u4,dhist_4,selec_u4,
address_u4);--y un conteo para mostrar el historial en el display
143     G2_resta_u5: mux_2selector_restador port map (direc_u5,direc_u5,dhist_5,selec_u5,
address_u5);
144     G2_resta_u6: mux_2selector_restador port map (direc_u6,direc_u6,dhist_6,selec_u6,
address_u6);
145     G2_resta_u7: mux_2selector_restador port map (direc_u7,direc_u7,dhist_7,selec_u7,
address_u7);
146     G2_resta_u8: mux_2selector_restador port map (direc_u8,direc_u8,dhist_8,selec_u8,
address_u8);
147     G2_h1: mux_8s port map (historial_u1,historial_u2,historial_u3,historial_u4,
historial_u5,historial_u6,historial_u7,historial_u8,usuario,numero_ram);--depende del
usuario que quiera ver el historial, entonces solo ahi se va a verificar el ultimo
numero guardado en la ram de dicho usuario
148     G2_sal_d1: mux_1selector port map (sal_reg_1,num1hist,disp_select,sal_mux_1);
--seleccionar en el display:
149     G2_sal_d2: mux_1selector port map (sal_reg_2,num2hist,disp_select,sal_mux_2);--el
numero escrito actual o
150     G2_sal_d3: mux_1selector port map (sal_reg_3,num3hist,disp_select,sal_mux_3);
--mostrar el historial
151     G2_sal_d4: mux_1selector port map (sal_reg_4,num4hist,disp_select,sal_mux_4);
152     G2_sal_d5: mux_1selector port map (sal_reg_5,num5hist,disp_select,sal_mux_5);
153     G2_sal_d6: mux_1selector port map (sal_reg_6,num6hist,disp_select,sal_mux_6);
154     G3_h1: comparador_4bits port map (direc_u1,dhist_1,igual_h1);
155     G3_h2: comparador_4bits port map (direc_u2,dhist_2,igual_h2);
156     G3_h3: comparador_4bits port map (direc_u3,dhist_3,igual_h3);
157     G3_h4: comparador_4bits port map (direc_u4,dhist_4,igual_h4);
158     G3_h5: comparador_4bits port map (direc_u5,dhist_5,igual_h5);
159     G3_h6: comparador_4bits port map (direc_u6,dhist_6,igual_h6);
160     G3_h7: comparador_4bits port map (direc_u7,dhist_7,igual_h7);
161     G3_h8: comparador_4bits port map (direc_u8,dhist_8,igual_h8);
162     G3_nu1: comparador_24bits port map (numero_actual,num_u1,iusuario1);--Si el numero
actual le pertenece al usuario 1
163     G3_nu2: comparador_24bits port map (numero_actual,num_u2,iusuario2);--Si el numero
actual le pertenece al usuario 2
164     G3_nu3: comparador_24bits port map (numero_actual,num_u3,iusuario3);
165     G3_nu4: comparador_24bits port map (numero_actual,num_u4,iusuario4);
166     G3_nu5: comparador_24bits port map (numero_actual,num_u5,iusuario5);
167     G3_nu6: comparador_24bits port map (numero_actual,num_u6,iusuario6);
168     G3_nu7: comparador_24bits port map (numero_actual,num_u7,iusuario7);
169     G3_nu8: comparador_24bits port map (numero_actual,num_u8,iusuario8);--Si el numero
actual le pertenece al usuario 8
170     G3_hu1: comparador_24bits port map (numero_ram,num_u1,iusuario_ant1);--Si el numero
anterior de la ram (para finalizar la llamada) le pertenece al usuario 1
171     G3_hu2: comparador_24bits port map (numero_ram,num_u2,iusuario_ant2);--Si el numero
anterior de la ram (para finalizar la llamada) le pertenece al usuario 2
172     G3_hu3: comparador_24bits port map (numero_ram,num_u3,iusuario_ant3);
173     G3_hu4: comparador_24bits port map (numero_ram,num_u4,iusuario_ant4);
174     G3_hu5: comparador_24bits port map (numero_ram,num_u5,iusuario_ant5);
175     G3_hu6: comparador_24bits port map (numero_ram,num_u6,iusuario_ant6);
176     G3_hu7: comparador_24bits port map (numero_ram,num_u7,iusuario_ant7);
177     G3_hu8: comparador_24bits port map (numero_ram,num_u8,iusuario_ant8);--Si el numero
anterior de la ram (para finalizar la llamada) le pertenece al usuario 8
178     G3_oc: comparador_4bits port map (dos,cuenta_oc,igual_oc);--agregado
179     G3_cuentn1: comparador_4bits port map (cuenta,cero,icuenta0);--si la cuenta está en
cero

```

```

180      G3_cuentn2: comparador_4bits port map (cuenta,uno,icuenta1);--si la cuenta está en uno
181      G3_cuentn3: comparador_4bits port map (cuenta,dos,icuenta2);--si la cuenta está en dos
182      G3_cuentn4: comparador_4bits port map (cuenta,tres,icuenta3);--si la cuenta está en
tres
183      G3_cuentn5: comparador_4bits port map (cuenta,cuatro,icuenta4);--si la cuenta está en
cuatro
184      G3_cuentn6: comparador_4bits port map (cuenta,cinco,icuenta5);--si la cuenta está en
cinco
185      G3_hist: comparador_4bits port map (cuatro,cuenta_hist,igual_hist1);--si ha llegado a
los 4 segundos
186      G4: flip_flop_D port map (clock_ff_d,reset_ff_d,led1,LLAMADA_EN_PROGRESO(0));
187      G5: flip_flop_D port map (clock_ff_d,reset_ff_d,led2,LLAMADA_EN_PROGRESO(1));
188      G6: flip_flop_D port map (clock_ff_d,reset_ff_d,led3,LLAMADA_EN_PROGRESO(2));
189      G7: flip_flop_D port map (clock_ff_d,reset_ff_d,led4,LLAMADA_EN_PROGRESO(3));
190      G8: flip_flop_D port map (clock_ff_d,reset_ff_d,led5,LLAMADA_EN_PROGRESO(4));
191      G9: flip_flop_D port map (clock_ff_d,reset_ff_d,led6,LLAMADA_EN_PROGRESO(5));
192      G10: flip_flop_D port map (clock_ff_d,reset_ff_d,led7,LLAMADA_EN_PROGRESO(6));
193      G11: flip_flop_D port map (clock_ff_d,reset_ff_d,led8,LLAMADA_EN_PROGRESO(7));
194      G12_1: registro_sostenimiento port map (clock,reset_reg,enable_reg(0),num_bcd,
sal_reg_1);
195      G12_2: registro_sostenimiento port map (clock,reset_reg,enable_reg(1),num_bcd,
sal_reg_2);
196      G12_3: registro_sostenimiento port map (clock,reset_reg,enable_reg(2),num_bcd,
sal_reg_3);
197      G12_4: registro_sostenimiento port map (clock,reset_reg,enable_reg(3),num_bcd,
sal_reg_4);
198      G12_5: registro_sostenimiento port map (clock,reset_reg,enable_reg(4),num_bcd,
sal_reg_5);
199      G12_6: registro_sostenimiento port map (clock,reset_reg,enable_reg(5),num_bcd,
sal_reg_6);
200      G14_u1: RAM port map (clock,we_u1,address_u1,numero_actual,historial_u1);--usuario 1
historial
201      G14_u2: RAM port map (clock,we_u2,address_u2,numero_actual,historial_u2);--usuario 2
historial
202      G14_u3: RAM port map (clock,we_u3,address_u3,numero_actual,historial_u3);--usuario 3
historial
203      G14_u4: RAM port map (clock,we_u4,address_u4,numero_actual,historial_u4);--usuario 4
historial
204      G14_u5: RAM port map (clock,we_u5,address_u5,numero_actual,historial_u5);--usuario 5
historial
205      G14_u6: RAM port map (clock,we_u6,address_u6,numero_actual,historial_u6);--usuario 6
historial
206      G14_u7: RAM port map (clock,we_u7,address_u7,numero_actual,historial_u7);--usuario 7
historial
207      G14_u8: RAM port map (clock,we_u8,address_u8,numero_actual,historial_u8);--usuario 8
historial
208
209      MSS: MSS_Call_Center port map
210      (clock,resetn,start,boton,BORRAR,Fu,HISTORIAL_LLAMADAS,cont6,Re_oc,Re_des,
igual_oc,dir_hist_igual,igual_hist1,
211      REALIZAR_LLAMADA,FINALIZAR_LLAMADA,icuenta0,icuenta1,icuenta2,icuenta3,icuenta4,
icuenta5,u1,u2,u3,u4,u5,u6,u7,u8,
212      Fu_ad1,Fu_ad2,Fu_ad3,Fu_ad4,Fu_ad5,Fu_ad6,Fu_ad7,Fu_ad8,iusuario1,iusuario2,
iusuario3,iusuario4,iusuario5,iusuario6,
213      iusuario7,iusuario8,iusuario_ant1,iusuario_ant2,iusuario_ant3,iusuario_ant4,
iusuario_ant5,iusuario_ant6,iusuario_ant7,
214      iusuario_ant8,incrementa,reset_reg,cuenta_a_0,incremento_u1,incremento_u2,
incremento_u3,incremento_u4,incremento_u5,incremento_u6,
215      incremento_u7,incremento_u8,we_u1,we_u2,we_u3,we_u4,we_u5,we_u6,we_u7,we_u8,
reset_oc,OCUPADO,reset_hist,disp_select,rhist_u1,
216      rhist_u2,rhist_u3,rhist_u4,rhist_u5,rhist_u6,rhist_u7,rhist_u8,ihist_u1,ihist_u2,
ihist_u3,ihist_u4,ihist_u5,ihist_u6,ihist_u7,
217      ihist_u8,led1,led2,led3,led4,led5,led6,led7,led8,clock_ff_d,selec_u1,selec_u2,
selec_u3,selec_u4,selec_u5,selec_u6,selec_u7,selec_u8,
218      enable_reg);
219  end uno;

```