

```
1  -----
2  -- Company:
3  -- Engineer:
4  --
5  -- Create Date:    12:29:05 06/02/2010
6  -- Design Name:
7  -- Module Name:    top - Behavioral
8  -- Project Name:
9  -- Target Devices:
10 -- Tool versions:
11 -- Description:
12 --
13 -- Dependencies:
14 --
15 -- Revision:
16 -- Revision 0.01 - File Created
17 -- Additional Comments:
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 ---- Uncomment the following library declaration if instantiating
26 ---- any Xilinx primitives in this code.
27 --library UNISIM;
28 --use UNISIM.VComponents.all;
29
30 ENTITY top IS
31     PORT ( CLK      : in  STD_LOGIC;
32           SW1       : in  STD_LOGIC;
33           SW2       : in  STD_LOGIC;
34           SW3       : in  STD_LOGIC;
35           LCD_E     : out  STD_LOGIC;
36           LCD_RS    : out  STD_LOGIC;
37           LCD_RW    : out  STD_LOGIC;
38           LCD_D4    : out  STD_LOGIC;
39           LCD_D5    : out  STD_LOGIC;
40           LCD_D6    : out  STD_LOGIC;
41           LCD_D7    : out  STD_LOGIC;
42           SF_CE0    : out  STD_LOGIC;
43           LED0      : out  STD_LOGIC;
44           LED1      : out  STD_LOGIC;
45           LED2      : out  STD_LOGIC;
46           LED3      : out  STD_LOGIC;
47           LED4      : out  STD_LOGIC;
48           LED5      : out  STD_LOGIC;
49           LED6      : out  STD_LOGIC;
50           LED7      : out  STD_LOGIC);
51 END top;
52
53
54 ARCHITECTURE Behaviour OF top IS
55
56 component CLK_DIV is
57     Port (
58         CLK_IN   : in  STD_LOGIC;
59         CLK_OUT  : out STD_LOGIC);
60 end component;
61
```

```
62
63     component START_UP is
64         Port ( CLOCK      : in  STD_LOGIC;
65              ST_UP      : in  STD_LOGIC;
66              EN         : out STD_LOGIC;
67              RS         : out STD_LOGIC;
68              DONE       : out std_logic;
69              D_BUS      : out STD_LOGIC_VECTOR (3 downto 0));
70     end component;
71
72     component LCD_CTRL is
73         Port ( CLOCK      : in  STD_LOGIC;
74              ENABLE      : in  std_logic;
75              D_C         : in  STD_LOGIC;
76              LCD_DATA    : in  STD_LOGIC_VECTOR (7 downto 0);
77              EN         : out STD_LOGIC;
78              RS         : out STD_LOGIC;
79              D_BUS      : out STD_LOGIC_VECTOR (3 downto 0));
80
81
82     end component;
83
84     component CHAR_GEN is
85         Port ( CLOCK      : in  STD_LOGIC;
86              START      : in  STD_LOGIC;
87              D_BUS      : out  STD_LOGIC_VECTOR (7 downto 0));
88     end component;
89
90
91     --variabelen
92     signal CLK_1MHz      : std_logic;
93     signal READY        : std_logic;
94     signal DATA_COM    : std_logic;
95     signal INT_LCD_DATA : std_logic_vector (7 downto 0) := "11111111";
96
97     signal EN_startup   : std_logic;
98     signal RS_startup   : std_logic := '0';
99     signal D0_startup   : std_logic;
100    signal D1_startup   : std_logic;
101    signal D2_startup   : std_logic;
102    signal D3_startup   : std_logic;
103
104    signal EN_ctrl      : std_logic;
105    signal RS_ctrl      : std_logic := '0';
106    signal D0_ctrl      : std_logic;
107    signal D1_ctrl      : std_logic;
108    signal D2_ctrl      : std_logic;
109    signal D3_ctrl      : std_logic;
110
111    BEGIN
112
113
114
115    SF_CE0      <= '1';
116    LCD_RW      <= '0';
117    DATA_COM   <= '1';
118    --INT_LCD_DATA <= "00100001";
119
120
121    G1 : CLK_DIV PORT MAP (
122                                     CLK_IN => CLK,
```

```
123             CLK_OUT => CLK_1MHz);
124
125
126     G2 : START_UP PORT MAP ( CLOCK      => CLK_1MHz,
127                             ST_UP      => SW1,
128                             EN         => EN_startup,
129                             RS         => RS_startup,
130                             DONE       => READY,
131                             D_BUS(3)  => D3_startup,
132                             D_BUS(2)  => D2_startup,
133                             D_BUS(1)  => D1_startup,
134                             D_BUS(0)  => D0_startup);
135
136
137     G3 : LCD_CTRL PORT MAP ( CLOCK      => CLK_1MHz,
138                             ENABLE     => READY,
139                             D_C        => DATA_COM,
140                             LCD_DATA   => INT_LCD_DATA,
141                             EN         => EN_ctrl,
142                             RS         => RS_ctrl,
143                             D_BUS(3)  => D3_ctrl,
144                             D_BUS(2)  => D2_ctrl,
145                             D_BUS(1)  => D1_ctrl,
146                             D_BUS(0)  => D0_ctrl);
147
148
149     G4: CHAR_GEN PORT MAP ( CLOCK      => CLK_1MHz,
150                             START      => SW2,
151                             D_BUS      => INT_LCD_DATA);
152
153
154
155     LCD_E <= (EN_startup or EN_ctrl);
156     LCD_RS <= (RS_startup or RS_ctrl);
157     LCD_D7 <= (D3_startup or D3_ctrl);
158     LCD_D6 <= (D2_startup or D2_ctrl);
159     LCD_D5 <= (D1_startup or D1_ctrl);
160     LCD_D4 <= (D0_startup or D0_ctrl);
161
162
163     END Behaviour;
```