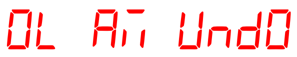
TÉCNICO LISBOA

Projecto e Controlo em Lógica Digital

1st Laboratory

Work #1 – Olá Mundo

Objective: Write "Ola Mundo" in the 7-segment displays
Launch Quartus II and open the project DE2_top;
Change the program to light (statically) the correct segments of the display in such a way that they indicate "Ola mundo". You can light them like this:



Work #2 – 7-Segment decoder

Objective: Decode numbers in binary so that they are displayed in the 7-segment displays in decimal. Use modules in Verilog to implement two decoders. The numbers are input using switches (8) of the board in binary. They should be displayed in decimal in the displays.

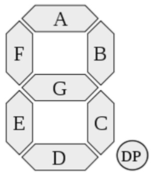
Launch Quartus II and open the DE2_top project;
Identify the lines from the eight switches (SW) as well as the two 7-segment display lines;
Implement a module that has as inputs the "numbers" in binary and as output the lines for the 7-segment displays;
Implement two decoders using four switches for the first number and four switches for the second number.

Decoders using combinational logic (Karnaugh maps)

We start by designing a circuit that, using the switches to represent a number in binary (using 2 bits) shows this number in the 7-segment display. Let's implement this circuit using Karnaugh tables.

After writing and simplifying the karnaugh tables we get:

A = !M2. M1	B = 0
C = M2 . !M1	D = !M2 . M1
E = M1	F = M2 + M1
G = !M2	



```
//use sw 5 and 6 to input a number in binary and show in the 7-segment
display (do the combinatorial logic)

assign      HEX0[0]          =     !SW[6] && SW[5];
assign      HEX0[1]          =     1'b0;
assign      HEX0[2]          =     !SW[5] && SW[6];
assign      HEX0[3]          =     !SW[6] && SW[5];
assign      HEX0[4]          =   SW[5];
assign      HEX0[5]          =   SW[5] || SW[6];
assign      HEX0[6]          =   !SW[6 ];
```

Decoders using "case statement"

Now, let's show in the 7-segment screen :

- 1 if the switches are 0 0
- 3 if the switches are 0 1
- 5 if the switches are 1 0
- 7 if the switches are 1 1

The code is:

```
reg [1:0] sel;
sel[1]=SW[5];
sel[0]=SW[6];
case (sel)
      2'b00 : begin
assign HEX0[0] = 1'b1;
assign HEX0[1] = 1'b0;
assign HEX0[2] = 1'b0;
assign HEX0[3] = 1'b1;
assign HEX0[4] = 1'b1;
assign HEX0[5] = 1'b1;
assign HEX0[6] = 1'b1;
      end

      2'b01 : begin
assign HEX0[0] = 1'b0;
assign HEX0[1] = 1'b0;
assign HEX0[2] = 1'b0;
assign HEX0[3] = 1'b0;
assign HEX0[4] = 1'b1;
assign HEX0[5] = 1'b1;
assign HEX0[6] = 1'b0;
      end
      2'b10 : begin
assign HEX0 = 7'b0010010;
      end
      2'b11 : begin
assign HEX0 = 7'b1111000;
      end
endcase
```