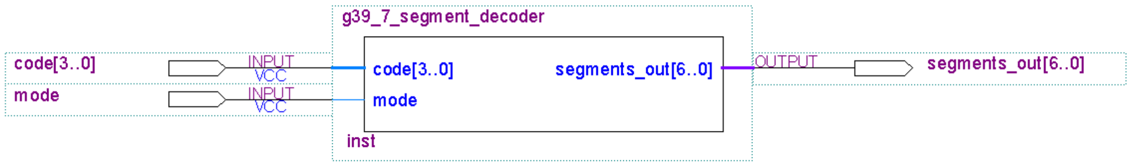
Group# 39 Saleh Bakhit (260632353) Adham El-Khouly (260601197)

ECSE 323 – Digital System Design
7-Segment LED Decoder
g39_7_segement_decoder

Description:

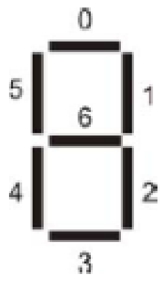
Inputs & Outputs:



code: 4-bit input
mode: 1-bit input
segments_out : 7-bit output

Function:

A 7-Segment LED Decoder works by taking a 5-bit input (code and mode concatenated) that maps onto an LED segment on the Cyclone II board. The mapping happens according to the diagram below:



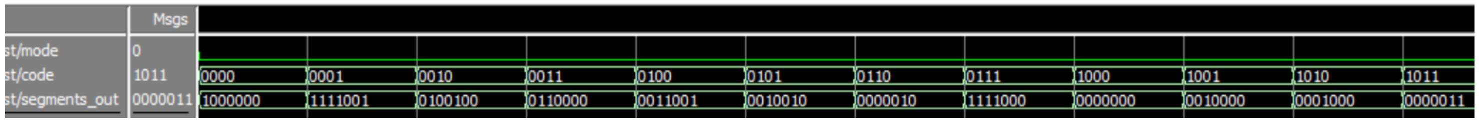
For example to generate the shape for the number "0" then we need to light up segments: 0,1,2,3,4 & 5. Since the circuit is an *active low* circuit then that would mean that the input should be "1000000". This translates to, turn off segment 6, and keep 0 through 5 on. We then proceeded to write the VHDL code for this component using Select "…." When"….." for all needed inputs.

The circuit operates under 2 modes. Mode 0 is just a normal 4 bit hexadecimal representation, meaning that 0 through 15 would map to 0 through F. Mode 1 is a card face value representation. That means that it maps 0 through 12 to A (Ace) through K (King). Since there are only 13 face values in a deck 13 through 15 maps to "---".

Testing:

Simulation and testing were done on all inputs. Those are 32 valid inputs in total. 16 inputs for mode 0, and 16 inputs for mode 1. Everything was tested and shown to the TA during the demo but below some of those tests are explained:

Mode 0:

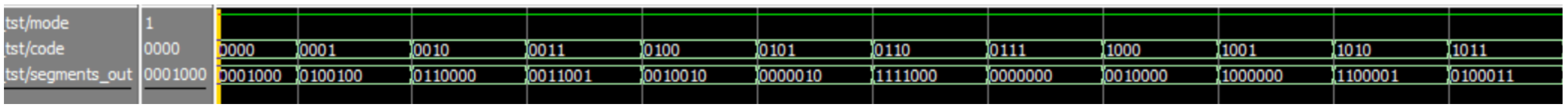


In this output graph the first line is the mode, as it can be seen by the flat green line, it is at 0 the whole time. The second line is the input code, while the last line is the output of the decoder itself.

As it was explained in the description, input codes map onto certain segments being on/off. The figure above shows inputs 0 to 11. The portion of the VHDL code that controlled this part is the following:
*Note: the last bit in the input part represents the mode (after concatenation).

```
"1000000" WHEN "00000", -- code = 0, mode = 0
"1111001" WHEN "00010", -- code = 1, mode = 0
"0100100" WHEN "00100", -- code = 2, mode = 0
"0110000" WHEN "00110", -- code = 3, mode = 0
"0011001" WHEN "01000", -- code = 4, mode = 0
"0010010" WHEN "01010", -- code = 5, mode = 0
"0000010" WHEN "01100", -- code = 6, mode = 0
"1111000" WHEN "01110", -- code = 7, mode = 0
"0000000" WHEN "10000", -- code = 8, mode = 0
"0010000" WHEN "10010", -- code = 9, mode = 0
"0001000" WHEN "10100", -- code = 10, mode = 0
"0000011" WHEN "10110", -- code = 11, mode = 0
```

Mode 1:



```
tst/mode          1        1
tst/code          0000     0000     0001     0010     0011     0100     0101     0110     0111     1000     1001     1010     1011
tst/segments_out  0001000  0001000  0100100  0110000  0011001  0010010  0000010  1111000  0000000  0010000  1000000  1100001  0100011
```

The portion of the VHDL code that controlled this part is the following:

```
"0001000" WHEN "00001", -- code = 0, mode = 1
"0100100" WHEN "00011", -- code = 1, mode = 1
"0110000" WHEN "00101", -- code = 2, mode = 1
"0011001" WHEN "00111", -- code = 3, mode = 1
"0010010" WHEN "01001", -- code = 4, mode = 1
"0000010" WHEN "01011", -- code = 5, mode = 1
"1111000" WHEN "01101", -- code = 6, mode = 1
"0000000" WHEN "01111", -- code = 7, mode = 1
"0010000" WHEN "10001", -- code = 8, mode = 1
"1000000" WHEN "10011", -- code = 9, mode = 1
"1100001" WHEN "10101", -- code = 10, mode = 1
"0100011" WHEN "10111", -- code = 11, mode = 1
```