Laboratory Assignment #5 – All Sections

Deadline: 02/01/2018, 11:55 a.m.

You will design a sequential circuit for a simple vending machine and implement it using Verilog HDL. The vending machine can deliver 3 different products: tea, coffee and hot chocolate. It has a single coin slot that accepts one coin (25 Krş, 50 Krş or 100 Krş) at a time. The prices of tea, coffee and hot chocolate are 50 Krş, 75 Krş and 100 Krş, respectively.

First, you will use push buttons on the FPGA board to select the product you want to purchase. You will use BTN1 for tea, BTN2 for coffee and BTN3 for hot chocolate. The following table shows the input combinations for products.

Product	BTN1	BTN2	BTN3
no product	0	0	0
Tea	1	0	0
Coffee	0	1	0
Hot Chocolate	0	0	1

Then, you will again use push buttons on the FPGA board to insert coin to the vending machine. You will use BTN1 for 25 Krş, BTN2 for 50 Krş and BTN3 for 100 Krş. The following table shows the input combinations for coins.

Coin Inserted	BTN1	BTN2	BTN3
no coin	0	0	0
25 Krş	1	0	0
50 Krş	0	1	0
100 Krş	0	0	1

You cannot press two push buttons simultaneously. When the total value of inserted coins is 50 Krş or more for the product tea, the vending machine delivers the product and returns coin if necessary. Similarly, when the total value of inserted coins is 75 Krş or more for the product coffee and 100 Krş or more for the product hot chocolate, the vending machine delivers the product and returns coin if necessary.

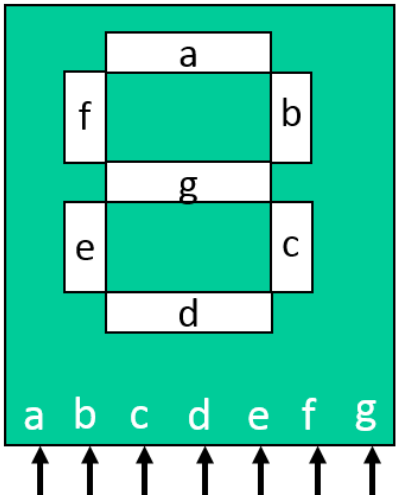
The sequential circuit should start from ‘no product’ state. You should use Reset BTN as an asynchronous reset input to put the sequential circuit into ‘no product’ initial state. When a product is delivered, the sequential circuit should go to ‘product delivered’ state and it should stay in this state until asynchronous reset is applied. This state is used to keep product output as 1 when a product is delivered.

The sequential circuit should have product output. This output should be 1 whenever a product is delivered, otherwise it should be 0. You should use LED4 to show whether product output is 1 or 0. You should use LED7, LED6 and LED5 to show which product is selected. If you select tea, LED7 should be on. If you select coffee, LED6 should be on. If you select hot chocolate, LED5 should be on.

The sequential circuit should also output the total value of inserted coins for a product in terms of Krş and the total value of returned coins for a product in terms of Krş after a product is delivered. These outputs should be shown on Seven Segment Displays (SSDs) on FPGA board. For example, if you inserted two 25 Krş coins for the product coffee, you should see '5 0 - -' on SSDs. Then, if you insert another 50 Krş coin for the same product, the product should be delivered and 25 Krş should be returned. Therefore, you should see '- - 25' on SSDs. An example is shown below.

BTN 3	BTN 2	BTN 1	Reset BTN		LED 7	LED 6	LED 5	LED 4	SSD 4	SSD 3	SSD 2	SSD 1	
0	0	0	1	→	0	0	0	0	-	-	-	-	(Reset)
0	1	0	0	→	0	1	0	0	0	0	-	-	(Coffee selected)
0	0	1	0	→	0	1	0	0	2	5	-	-	(25 Krş inserted)
0	0	1	0	→	0	1	0	0	5	0	-	-	(25 Krş inserted)
0	1	0	0	→	0	1	0	1	-	-	2	5	(50 Krş inserted)
0	0	0	1	→	0	0	0	0	-	-	-	-	(Reset)

An SSD and its control signals 'abcdefg' are shown below. Using 7-bit 'abcdefg' control signals, you can display different digits and signs on an SSD. For example, 'abcdefg' should be '0000001' in order to display 0 and '0000110' in order to display 3. There are four SSDs on the FPGA board. First two SSDs should be used to display the total value of inserted coins for a product while the last two SSDs should be used to display the total value of returned coins for a product. Your Verilog module should have 7-bit output for each SSD; digit1 and digit2 for the first and second decimal digits of the total value of inserted coins for a product, digit3 and digit4 for the first and second decimal digits of the total value of returned coins for a product.



Even though you press a push button on the FPGA board once, more than one logic 1 inputs may be given to your sequential circuit. Verilog codes in the "debouncer.v" and "clk_divider.v" provided to you solve this problem. When simulating your sequential circuit do not use "debouncer.v" and "clk_divider.v". You should use them when implementing your sequential circuit on the FPGA.

You should use the following interface for your Verilog module as in the "vending_ machine.v" provided to you.

```
module vending_machine (clk, reset, BTN1, BTN2, BTN3,
                        product1, product2, product3, delivered,
                          digit1, digit2, digit3, digit4);

  input clk, reset, BTN1, BTN2, BTN3;
  output reg product1, product2, product3, delivered;
  output reg [6:0] digit1, digit2, digit3, digit4;
  ...
endmodule
```

Do the following before the deadline and in the lab session.

Before the deadline

1. Design your sequential circuit and implement it using Verilog HDL.
2. Simulate your Verilog code using ISim. Try as many inputs as necessary to make sure that it is working correctly.
3. Associate your inputs and outputs with the push buttons and LEDs on the FPGA board using "ucf_vending_machine.ucf" provided to you.
4. Synthesize and implement your Verilog code together with the Verilog codes provided to you.
5. Generate the bitstream file.
6. Do not write a report for this lab assignment.
7. Zip your project directory (as much as you've done), name it as "username(s)_SectionInfo_LabNumber", i.e. acmert_hasanazgin_SectionA_Lab4.zip, and submit your zip file through SUCourse.

In the lab

8. Download the generated bitstream file to the FPGA board.
9. Demonstrate your work to the lab assistant as (s)he instructs. Show that your Verilog code is working correctly on the FPGA board.