Multi Product Vending Machine Using Verilog

Submitted
By:

Anupam Verma
2K20/EC/037

Abhishek Gupta
2K20/EC/009

Submitted To:

Ms. Kriti Suneja
Department of Electronics And Communication Engineering
Delhi Technological University,Delhi
Delhi-110042
May,2024

Electronics & Communication Engg. Deptt.
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Bawana Road-Delhi-42

Certificate

We hereby certify that the project titled "Multi Product Vending Machine Using Verilog" which is submitted by Anupam Verma. (2K20/EC/037) and Abhishek Gupta(2K20/EC/009) [Electronics and Communication Engineering], Delhi Technological University, Delhi is a record of the project work carried out by the student under my supervision. To the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

Place : Delhi
Date: 01/05/2022

Ms. Kriti Suneja
(Supervisor)
Department of ECE
Delhi Technological University

Electronics & Communication Engg. Deptt.
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Bawana Road-Delhi-42

Candidate's Declaration

We, hereby, declare that the work embodied in this project entitled "Multi Product Vending Machine Using Verilog" submitted to the Department of Electronics & Communication Engineering, Delhi Technological University, Delhi is an authentic record of our own bonafide work and is correct to the best of our knowledge and belief. This work has been undertaken taking care of engineering ethics.

Name and Signature of the Student:

Anupam Verma
(2K20/EC/037)

Abhishek Gupta
(2K20/EC/009)

Electronics & Communication Engg. Deptt.
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Shahbad Daulatpur, Bawana Road-Delhi-42

ACKNOWLEDGEMENT

In performing Our project, We had to take the help and guideline of some respected persons, who deserve our greatest gratitude. The completion of this project gives us much pleasure. We would like to show our gratitude to Ms. Kriti Suneja, mentor for our project, for giving us good guidelines for reporting throughout numerous consultations. We would also like to extend our deepest gratitude to all those who have directly and indirectly guided us in writing this report. In addition, We would like to thank the Department of Electronics And Communication Engineering Delhi Technological University for giving us the opportunity to work on this topic.

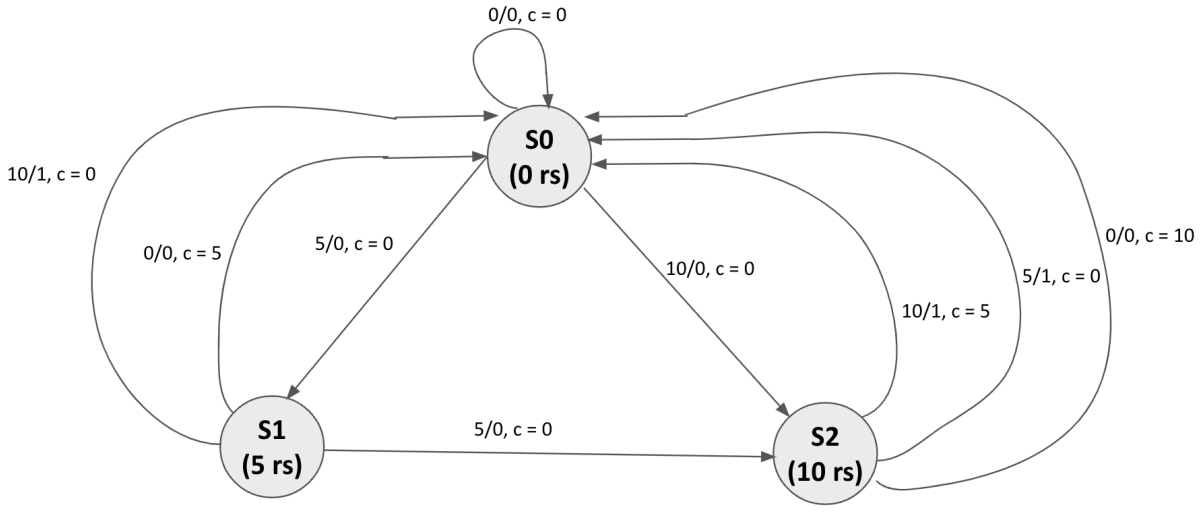
INTRODUCTION TO THE TOPIC

A Vending machine is an automated Machine is an automated machine that provides items such as snacks or beverages after cash or other forms of payment are inserted into the machine. By the means of this project we have designed a two product vending machine which can be extended based upon the needs using VERILOG hardware description language.

For the simplicity of the state diagrams and the algorithm we have assumed that the vending machine has only two products priced at Rs 15 and Rs 20. Also the accepted payment include only 5Rs and 10Rs. We have designed a Mealy machine for each product and implemented that mealy machine using Xilinx Vivado.

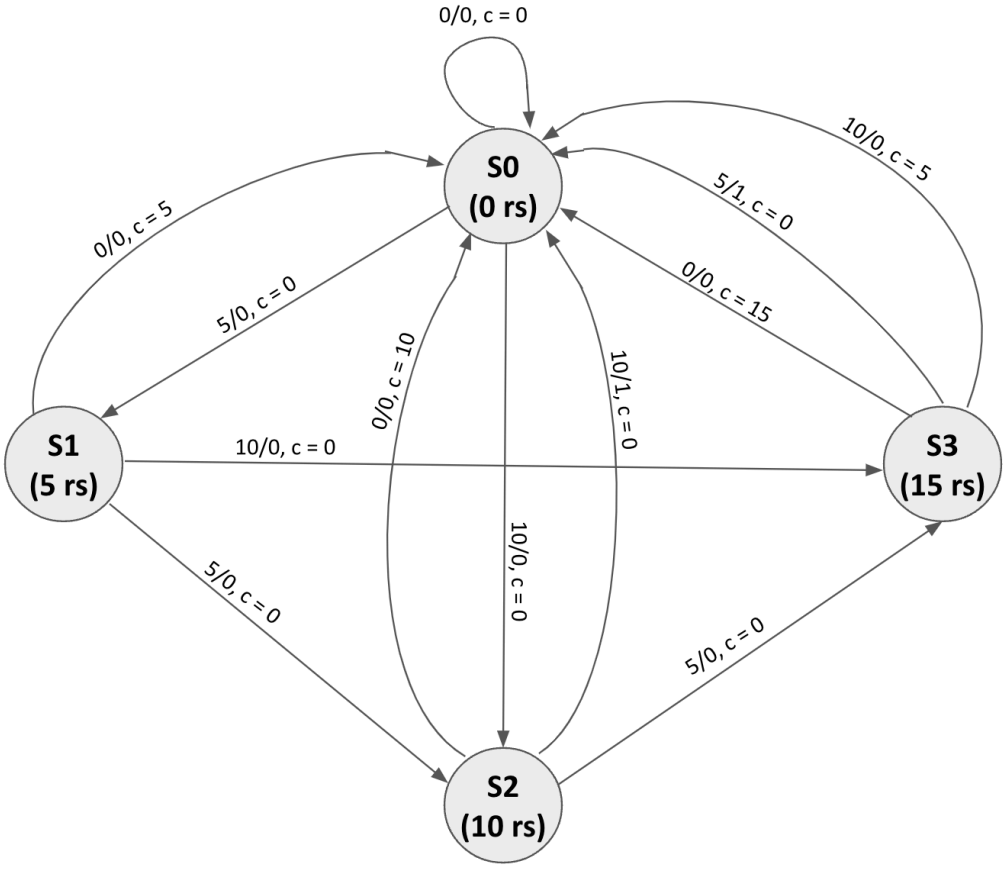
Mealy State Diagram For the First Product

Here the first product is assumed to be priced at 15Rs, and therefore we have three states in the state diagram.



Mealy State Diagram For the Second Product

Here we have assumed the product to be priced at 20Rs, and therefore we have four states in the state diagram.



VERILOG CODE :

```
module vending_machine(
input clk,
input rst,
input [1:0]in, // 01 = 5 rs, 10 = 10 rs
input prod,
output reg out,
output reg[1:0] change //11 = 15 rs
);

parameter s0 = 2'b00, s1 = 2'b01, s2 = 2'b10, s3 = 2'b11;
reg[1:0] state;

always@ (posedge clk)
   begin
      if(rst == 1)
         begin
         state = 0;
         change = 2'b00;
         end
      else
         begin
         if(prod == 1)
            begin
            case(state)
            s0:
               if(in == 0)
                  begin
                  state = s0;
                  out = 0;
                  change = 2'b00;
                  end
               else if(in == 2'b01)
```

```
        begin
        state = s1;
        out = 0;
        change = 2'b00;
        end
    else if(in == 2'b10)
        begin
        state = s2;
        out = 0;
        change = 2'b00;
        end
s1:
    if(in == 0)
        begin
        state = s0;
        out = 0;
        change = 2'b01;
        end
    else if(in == 2'b01)
        begin
        state = s2;
        out = 0;
        change = 2'b00;
        end
    else if(in == 2'b10)
        begin
        state = s0;
        out = 1;
        change = 2'b00;
    end
s2:
    if(in == 0)
        begin
        state = s0;
```

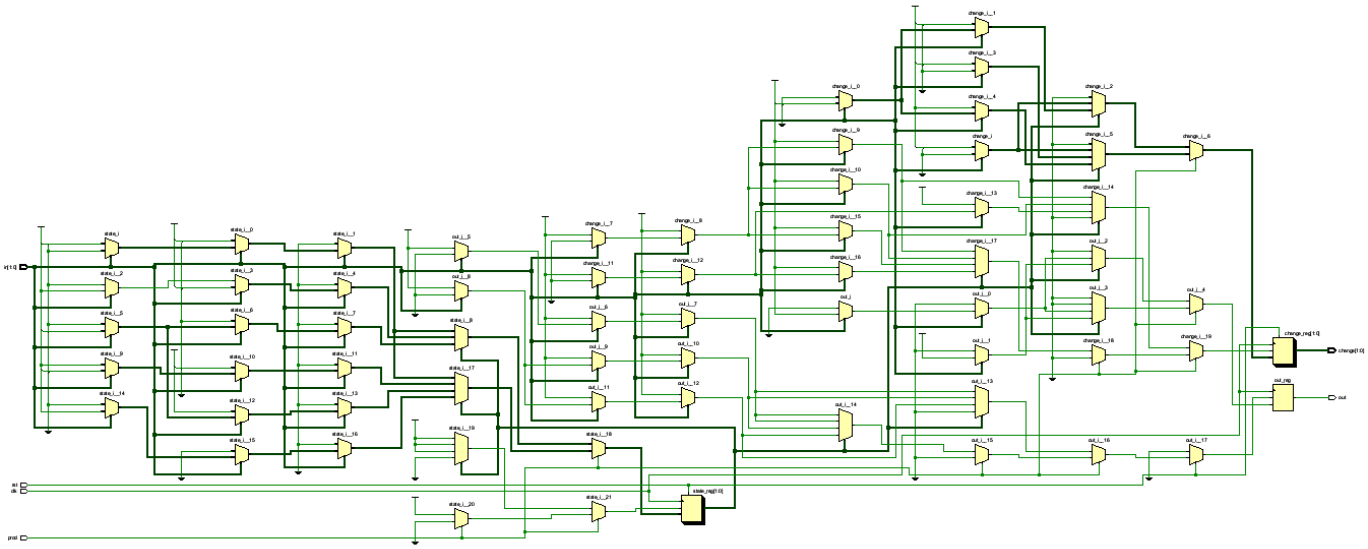
```
            out = 0;
            change = 2'b10;
            end
        else if(in == 2'b01)
            begin
            state = s0;
            out = 1;
            change = 2'b00;
            end
        else if(in == 2'b10)
            begin
            state = s0;
            out = 1;
            change = 2'b01;
            end
    endcase
    end

else if (prod == 0)
    begin
    case(state)
    s0:
        if(in == 0)
            begin
            state = s0;
            out = 0;
            change = 2'b00;
            end
        else if(in == 2'b01)
            begin
            state = s1;
            out = 0;
            change = 2'b00;
            end
```

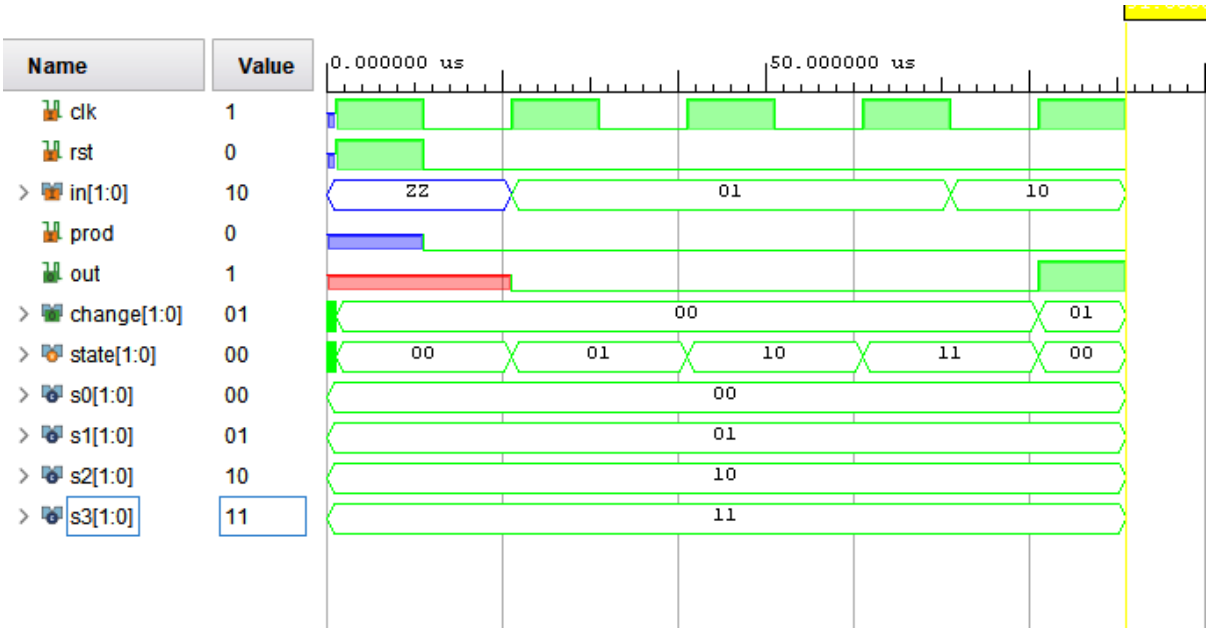
```
    else if(in == 2'b10)
        begin
        state = s2;
        out = 0;
        change = 2'b00;
        end
s1:
    if(in == 0)
        begin
        state = s0;
        out = 0;
        change = 2'b01;
        end
    else if(in == 2'b01)
        begin
        state = s2;
        out = 0;
        change = 2'b00;
        end
    else if(in == 2'b10)
        begin
        state = s3;
        out = 0;
        change = 2'b00;
        end
s2:
    if(in == 0)
        begin
        state = s0;
        out = 0;
        change = 2'b10;
        end
    else if(in == 2'b01)
        begin
```

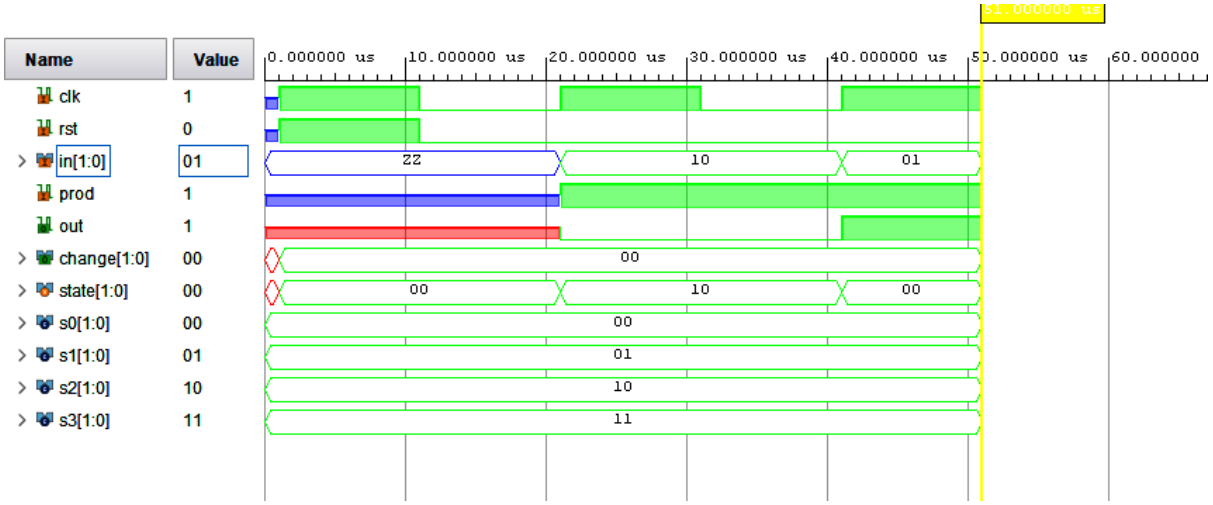
```
                state = s3;
                out = 0;
                change = 2'b00;
                end
            else if(in == 2'b10)
                begin
                state = s0;
                out = 1;
                change = 2'b00;
                end
        s3:
            if(in == 0)
                begin
                state = s0;
                out = 0;
                change = 2'b11;
                end
            else if(in == 2'b01)
                begin
                state = s0;
                out = 1;
                change = 2'b00;
                end
            else if(in == 2'b10)
                begin
                state = s0;
                out = 1;
                change = 2'b01;
                end
        endcase
        end
    end
  end
endmodule
```

SCHEMATIC :



Simulation Results :

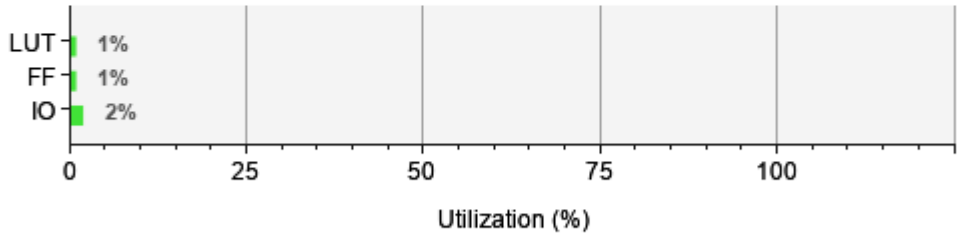




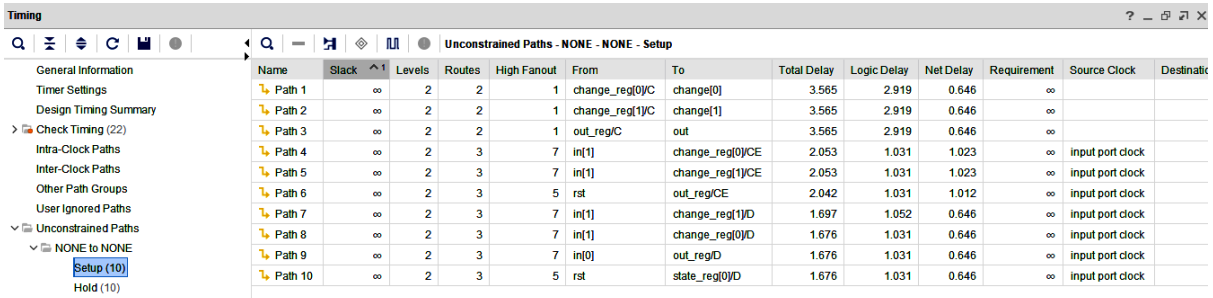
Utilization Summary :

Summary

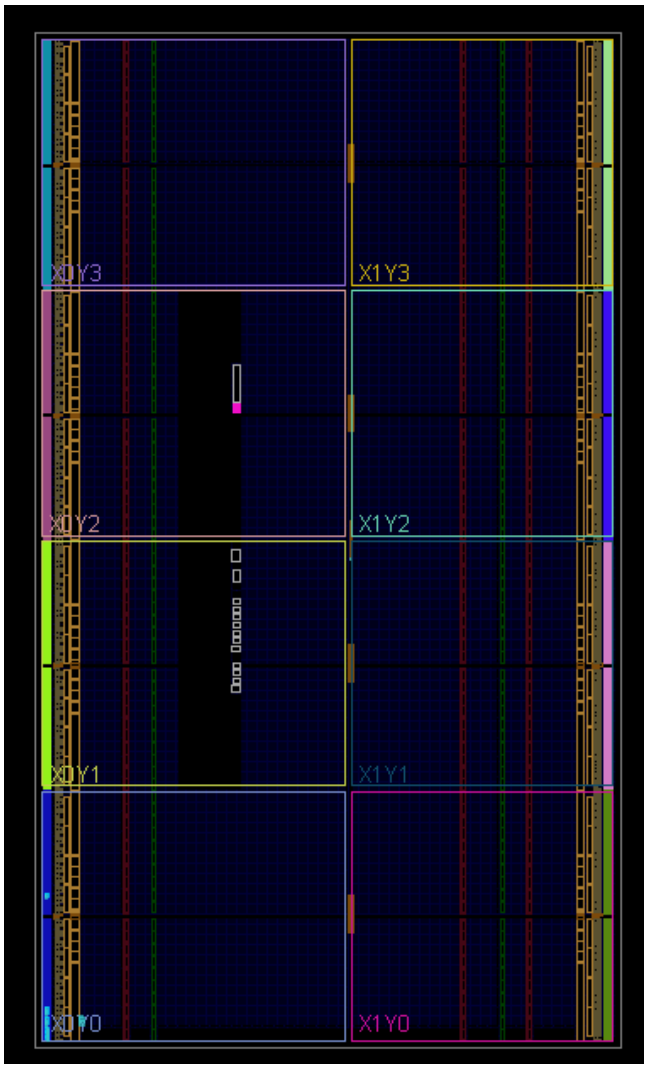
Resource	Utilization	Available	Utilization %
LUT	6	64000	0.01
FF	5	128000	0.00
IO	8	400	2.00



Timing Summary:



Device layout After Implementation:



References :

- Verilog HDL by Samir Palnitkar
- A Verilog HDL Primer by J. Bhasker
- www.asic-world.com