PHYS 234 Lab4

Ethan Yang

4-bit Enabled Register

A 4-bit enabled register is a group of 4 flip-flops that only set the output equal to the input on the rising edge of the clock and if enable is 1. The clock is implemented using a button and enable with a switch. We also use 4 switches for the 4-bit input and 4 LED lights to show the 4-bit output. This is programmed with the Verilog below:

```
`timescale 1ns / 1ps
module eelab4(
    input [0:3] a,
    input c, // clock
    input e, // enable
    output reg [0:3] q
    );
always @ (posedge c)
     if (e) q <= a;
endmodule
```

As a result, our input 4-bit number will only be reflected in our 4 LED lights if both the enable switch is flipped on and the clock button is pressed. The circuit will not respond if only 1 of enable or clock is 1.

D Flip-Flop with Synchronous Set and Asynchronous Reset

A synchronous set means the output is set to 1 on the rising edge of the clock if set is 1 and an asynchronous reset means the output becomes 0 as soon as the reset becomes 1. If both set and reset are 1, the output will be 0. We will use an if-else statement under the rising edge of the clock and reset to implement this function, shown below in the Verilog code:

```
`timescale 1ns / 1ps
module eelab4(
    input s, // [0]
    input r, // [1]
    input c, // clock
    input d,
    output reg q
    );
always @ (posedge c or posedge r) begin
    if (r) q <= 0;
```

```
    else if (s) q <= 1;
    else q <= d;
    end
endmodule
```

The results can be shown with the table below. Whenever reset is 1, the output will be 0, and the circuit only outputs 1 when both set and clock is 1 and reset is 0. Lastly, the output will equate to the input d when the circuit is neither set nor reset.

Set	Reset	Clock	Output
0	0	0	D
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0

JK Flip-Flop

For a JK flip-flop, the inputs are J, K, and the clock and the output is Q. On the rising edge of the clock, Q is set equal to a value determined by J and K, which are shown below:

- J = 0, K = 0: Q retains its old value.
- J = 0, K = 1: Q is resetted to 0.
- J = 1, K = 0: Q is set to 1.
- J = 1, K = 1: Q is set to the opposite of its old value.

This can be implemented using an if-else statement at the rising edge of the clock with the Verilog code below:

```
`timescale 1ns / 1ps
module eelab4(
    input j,
    input k,
    input c, // clock
```

```
    output reg q
    );
always @ (posedge c) begin
     if ((j == 0) && (k == 0))
          q <= q;
     else if ((j == 0) && (k == 1))
          q <= 0;
     else if ((j == 1) && (k == 0))
          q <= 1;
     else if ((j == 1) && (k == 1))
          q <= !q;
    end
endmodule
```

The results are visualized with the truth table below:

J	K	Clock	Q
0	0	0	Q
0	0	1	Q
0	1	0	Q
0	1	1	0
1	0	0	Q
1	0	1	1
1	1	0	Q
1	1	1	!Q

The value of Q is only altered at the rising edge of the clock, therefore, whenever clock is 0, Q retains its old value. When the clock is 1 (or at the rising edge of the clock), Q equates to its corresponding value according to the definition earlier.