EECS 3215: Embedded Systems

Lab 2: Pulse Position Modulation

Sean O’Brien - 213735741

Michael Founk - 213641204

**Problem Statement**

The problem to be solved in this lab is combining multiple pulse width modulation signals into a pulse position modulation signal. The PPM signal is created by setting one of the pins of the Quartus II board to high and demonstrating this using the oscilloscope. The signal should repeat itself every 30 milliseconds and there should be a way to reset the signal using a push button switch.

**Top-Level Design**

This lab utilizes the clock and a pushbutton as inputs and two pins on the board as output. The clock allows us to time the signal properly and the pushbutton acts as a reset for the signal. We spoke with the TA to clarify the function of the pushbutton and she told us that it is acceptable for the pushbutton to pause the signal because the signal is demonstrated on the oscilloscope which displays a stationary version of the signal.

The code we have written to represent this signal is clock counter which resets itself back to zero every 30ms. The output pin is set to high when the the counter is between 0-6ms and 8-16ms, the pin is at zero for all other times. We use registers to hold the value of the clock counter and the pin, and the pin register is assigned to the pin output. To represent the push-button reset, we enclose the counting logic in an if statement which checks the status of the button. If the button is pressed, then the signal ceases, and when the button is released, the signal continues.

There are two options for representing the PPM signal, and for this lab we opted to represent the second option (signal e). We choose this signal because we would rather work with a signal that is defined over a known amount of time than an impulse. A second reason for deciding to work with the second signal was to ensure that the signal would be easily viewed on the oscilloscope. An issue with the first signal is if we tried to properly approximate an impulse, it would be very thin compared to the duration of the signal, the impulses could have been widened, but we felt we could more accurately represent the second signal.

**Verilog Code**

```
module Lab2(Clk, reset, GPIO_0, ground);
input Clk;
input reset; //on when not pressed
reg r1;
reg[25:0] ClkCount;
output GPIO_0;
output ground;

assign GPIO_0 = r1;

if (~reset)

always@(posedge Clk)
        begin
```

```
                        if (0 < ClkCount &&  ClkCount < 26'd300000)
                                r1 = 1'b1;
                        else if(26'd400000 < ClkCount &&  ClkCount < 26'd800000)
                                r1 = 1'b1;
                        else
                                r1 = 1'b0;
                        if (ClkCount == 26'd1500000)
                                ClkCount = 26'b0;
                        else
                                ClkCount = ClkCount + 26'b1;
                        end

        end
end
endmodule
```

**Pin Assignments**

```
set_global_assignment -name FAMILY "Cyclone II"
set_global_assignment -name DEVICE EP2C35F672C6
set_global_assignment -name TOP_LEVEL_ENTITY Lab2
set_global_assignment -name ORIGINAL_QUARTUS_VERSION "12.1 SP1"
set_global_assignment -name PROJECT_CREATION_TIME_DATE "11:57:26  JANUARY
21, 2019"
set_global_assignment -name LAST_QUARTUS_VERSION "12.1 SP1"
set_global_assignment -name PROJECT_OUTPUT_DIRECTORY output_files
set_global_assignment -name MIN_CORE_JUNCTION_TEMP 0
set_global_assignment -name MAX_CORE_JUNCTION_TEMP 85
set_global_assignment -name ERROR_CHECK_FREQUENCY_DIVISOR 1
set_global_assignment -name VERILOG_FILE Lab2.v
set_global_assignment -name PARTITION_NETLIST_TYPE SOURCE -section_id Top
set_global_assignment -name PARTITION_FITTER_PRESERVATION_LEVEL
PLACEMENT_AND_ROUTING -section_id Top
set_global_assignment -name PARTITION_COLOR 16764057 -section_id Top
set_location_assignment PIN_N2 -to Clk
set_location_assignment PIN_G26 -to reset
set_location_assignment PIN_D25 -to GPIO_0
set_location_assignment PIN_P18 -to ground
set_instance_assignment -name PARTITION_HIERARCHY root_partition -to | -section_id
Top
```