Michael Kearns

ELEC-3200

Professor Junsangsri

07/17/20

Lab06

This lab focuses on the utilization of a clock and timing. Clocks are vital for the functionality of sequential logic circuits. This lab is a simple example of a shift register utilizing a clock to shift a hexadecimal value through a 4-byte register every 3 seconds. The inputs to the FPGA board are 4 switches used for the input hex value, and a button. When the button is held down during a rising edge of the clock, the hex value from the switches will be input into the least significant byte of the register.

The internal clock of the FPGA was used for this lab. The internal clock oscillates at a frequency of 50 MHz A simple counter was implemented to count the clock cycles and create a simulated clock that oscillates very 3 seconds. For every 150 million internal clock oscillations, the simulated clock changes once.

A D-flip flop is used to create the shift register because it allows for data storage based on a clock input. Four flip flops are created and connected to create a 4-byte register for the inputted values. With each rising clock edge, the values from the previous flip flop are sent to the next. The first flip flop is filled with either a zero or the input hex value depending on if the button is pressed or not. The value from the fourth and final flip flop does not matter and is not stored or sent anywhere after a shift.

The entire process is visualized using the 7-segment displays on the board. HEX0 displays the current input hex value based on the four switches. HEX 5, HEX 4, HEX 3, and HEX 2, are used to show each value currently stored in each flip flop.

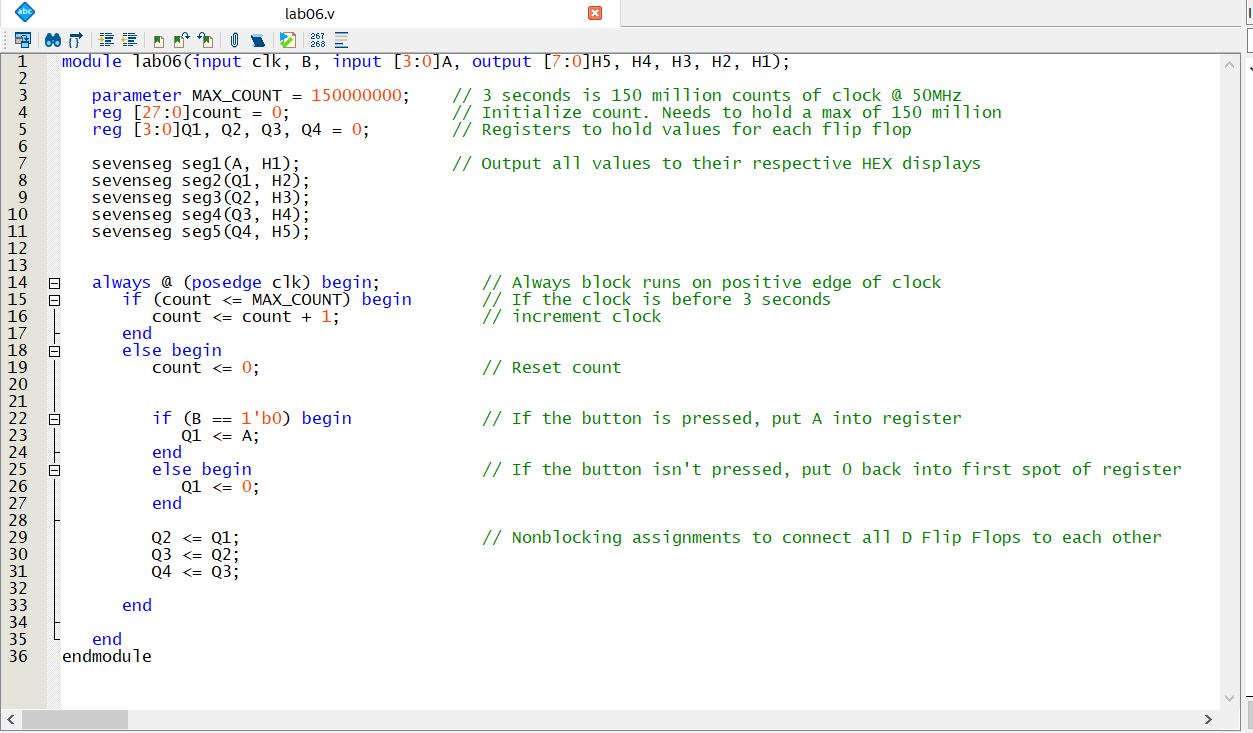


Figure 1: Verilog code for shift register

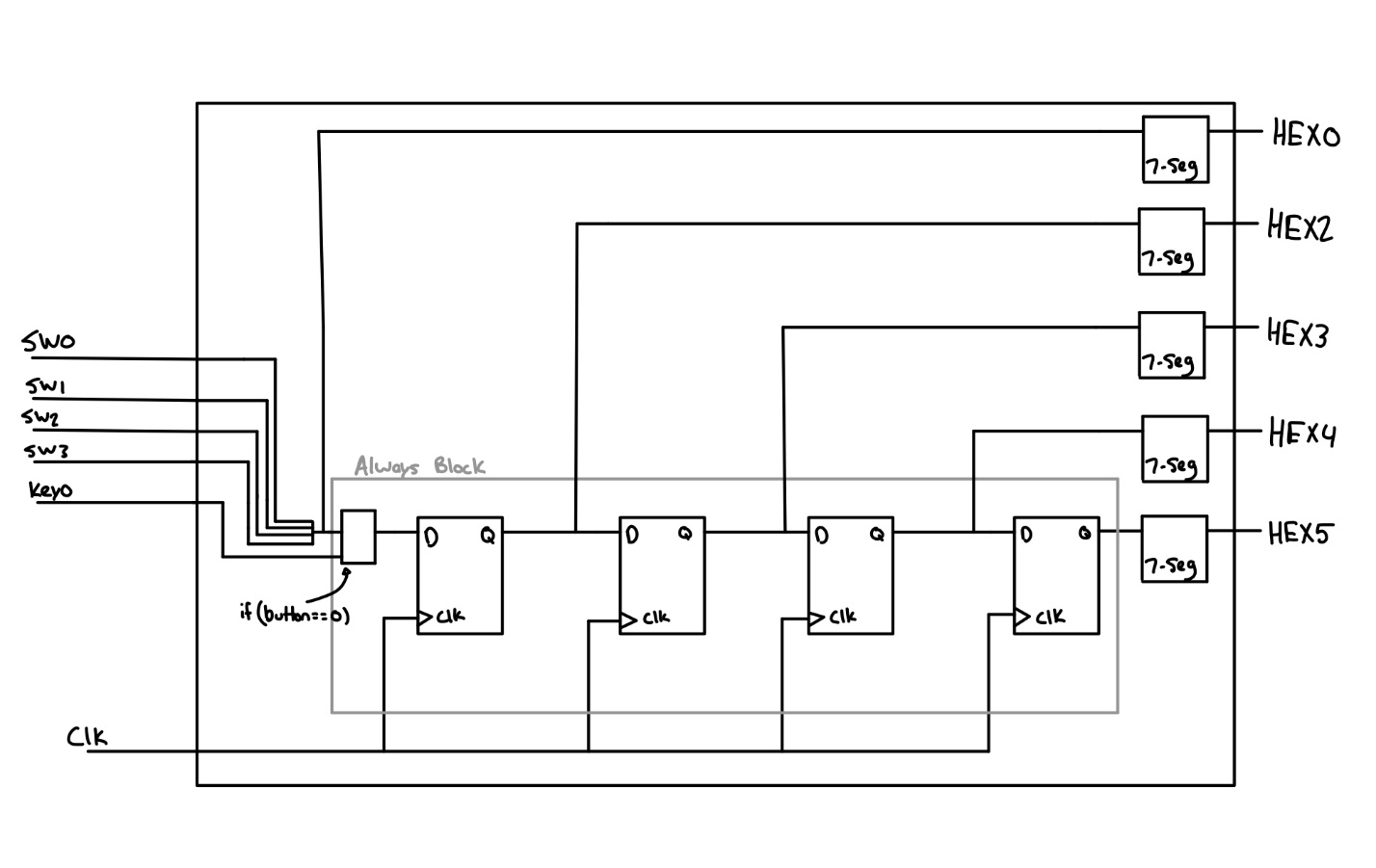


Figure 2: System Block Diagram

Figure 1 shows the Verilog code used to program the board and Figure 2 shows the block diagram that represents the Verilog code. The hex input, and corresponding hex values in each d flip flop are outputs to their respective 7-segment displays. The actual shift register is contained within an always block that triggers on the positive edge of the clock variable. Because of the clock counter that counts the oscillations of the boards internal 50 MHz clock, the always block will run every 3 seconds. There is an if-statement within the always block to check whether the button is being pressed. If the button is being pressed, the input hex value will be loaded into the left most d flip flop. If the button is not pressed, a value of 0 will be loaded into that d flip flop instead. At every clock edge, the value inputted into the register will shift to the next flip flop. These flip flops then output to the 7-segment displays on the board in the opposite direction from left to right to show the value shifting left every 3 seconds.

A demonstration of this program can be viewed here: <https://youtu.be/kQEOGvoqBEM>

The results of this program were as expected, if the button is being pressed down when the clock edge rises, the hex value determined by the switches will be loaded into the register and then subsequently shifted left every 3 seconds. This is a great example of how timing and synchronization works with clocks on FPGA boards.