Digital Design Principles

Crosswalk Controller

Name: Rajkaran Singh Grewal

Student No: 8882386

Email: rgrewal2386@conestogac.on.ca

Objective:

- I use VHDL to implement a Crosswalk Controller.

- Introduction to Clock Frequency.

Verilog introduction:

For the program in Verilog, we will be initializing a main module which will have the following parameters: an input clock, four input push-buttons, 6 output arrays for the seven segment arrays, and finally an output led.

Within the module we will have a wire named clock Phase for the clock phase and output of the second module. We will also initialize two registers holding an array of 4 units for counter 1 and counter 2. Where counter 1 will have the initial value of 0000, and the initial value of counter 2 will be 1000 (8 in decimal). We will then initialize four registers for the push state, the state when the push buttons are pressed. We will initialize a module clock cycle for converting the 50 MHz in-built clock to a 1 Hz clock for our use. The module will take an input of our clock and will return the output of the wire clock Phase.

Within the always block where we are checking the posedge of clock Phase, we increment both the counters by one. Then we have an if statement which checks when the counter 1 is greater by the value of 15, and when it is we set the value of counter 1 to zero, we set the value of the first push state to zero and the third push state to zero. We have another if statement which checks when the value of counter 1 is greater than 8 and when it is it will set the value of the first and third push state to zero. The next if statement will check when the counter 2 is greater than 15 and when it is it will set the value of counter 2 to zero and set the value of the second and fourth push state to zero. The next if statement will check when the counter 2 will be greater than 8 and when it is it will set the value of second and fourth push set to zero. The next if statement will check if the value of push button 1 is zero it will set the value of push state 1 to 1 and the led will also be set to one. And in the condition of else we will set the led to zero. The same will continue with the rest of the push buttons. And finally, we will set the value of hex according to the output of the functions from display Seven Segment and displaySevenSegment2. With these functions the value of the input are check through using a case and the output are set according to which of the input we received.

Finally in the second module we will register a counter holding a value of 26 units to zero. And when the counter value reaches 49,999,998 it will set the output clock to one while when the value reaches 49,999,999 the counter will be reset back to zero and the output, we be set to zero as well.

.

Verilog program screenshot: Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated Text

Description automatically generated Text

Description automatically generatedVerilog compilation report: Graphical user interface, text, application

Description automatically generated

Verilog pin planner screenshot:

Table

Description automatically generated Table

Description automatically generated Table

Description automatically generated

Conclusion:

Thus we can conclude that by using the programming language of verilog we will set a four set crossWalk using the simplest manner and using a secondary module.