Reaction Timer

Signature and Grading Sheet

Group #:_____ Name(s):_____ .

Signature

Section 4.2(f): _____.

Grading

- Section 4.1(a)(b): pseudo algorithm and registers (10 points):_____.
  Attach 1-page algorithm and explanation of registers
- Section 4.1(c): ASMD chart (25 points):_____.
  Attach 1-page detailed ASMD chart
- Section 4.1(d): VHDL code (25 points):_____.
  Attach code printout
- Section 4.1(e): Simulation (20 points):_____.
  Attach simulation timing diagram printout.
- Section 4.2 (e) (5 points):_____.
  Attach one-page report printout with # of LEs circled .
- Section 4.2 (f) : demo signature (15 points):_____.

Total points: _____.

Experiment
Reaction Timer

1 Purpose

To use FSMD methodology to design and implement an intermediate-sized digital circuit.

2 Reading

- Chapter 7 of *Embedded SoPC Design with Nios II Processor and VHDL Examples*

3 Project specification

The purpose of this project is to construct a circuit that measures a person's eye-hand reaction time. The reaction time is the period between the time that an indicator comes on and the time that the person presses the stop button. The reaction time should be recorded in 1 milli-second interval up to .999 second (i.e., the range of display is from 0.001 second to 0.999 second) and displayed on LED.

In addition to the clock and reset signals, the input signals of this circuit are

- clear: a pushbutton switch that moves the circuit to initial state.
- start: a pushbutton switch that starts the reaction time test.
- stop: a pushbutton switch that stops the timer.

The circuit outputs include LED indicators and two 7-segment LED displays:

- ready: a green LED that indicates the circuit is ready.
- go: a red LED that indicates the initiation of the reaction timer.
- counter_out: three 7-segment LED displays that show the reaction time (from 000 to 999).

Note that the clear signal synchronous and is different from the asynchronous reset signal. The reset signal should be only used once during power-on.

The reaction timer functions as follows

(a). When a user presses the clear button, the circuit enters an initialization state. The ready LED is on and go LED is off. The counter_out LED displays " HI".

(b). When the user is ready for test and presses the start button. The circuit waits for a random interval between 1 second to 7 seconds. In this waiting interval, the ready LED is off. The counter_out LED displays 000. If the user presses the stop button during this interval (i.e., he/she is cheating), the test is over and the counter_out LED displays " CH" (for "cheating"). The circuit stays there until the clear button is pressed again.

(c). If the user doesn't press the stop button, the circuit turns on the go LED after the random interval and starts the counting timer. The timer increments every 1 milli-second and its content is displayed in counter_out LED in BCD format.

(d). Once the go LED is on, the user should try to press the stop button as soon as possible. When the stop button is pressed, the timer stops counting. The counter_out LED displays the reaction time. The circuit stays there until the clear button is pressed again

(e). If the user fails to press the stop button within 999 milli-seconds, the circuit stops counting and the counter_out LED displays " SL" (for slow). The circuit stays there until the clear switch is pressed again.

The design must be synchronous or 50% will be deducted.

4 Design Procedures

4.1 FSMD design and simulation

(a) Derive a pseudo algorithm.

(b) Determine the registers needed.

(c) Derive the detailed ASMD chart according to the algorithm. This circuit can be done by using 5 to 10 states.

(d) Derive VHDL code according to the ASMD chart.

(e) Use a mod-2 counter for 1-millisecond tick and perform simulation.

4.2 Implementation and testing

(a) Replace the mod-2 counter with the actual counter to generate 1-millisecond tick and derive VHDL code

(b) Use the 50MHz oscillator for clock.

(c) Use a slide switch for the reset signal and three pushbutton switches for three input signals.

(d) Perform pin assignment and synthesize the code.

(e) Look at the compiling report and determine the number of LEs used in circuit.

(f) Download the file to the FPGA device and verify the operation of physical circuit. Demonstrate the circuit to instructor and get signature.