FAMU- FSU College of Engineering

Department of Electrical and Computer Engineering

Fall 2019 Semester

EEL3705L Digital Logic Lab Report

Section No:	1
Lab Instructor:	Rajesh Thomas
Lab No:	PROJECT
Lab Title:	Stopwatch with Lap Timer
Name:	Marc Abad
Partner's Name:	John Mijares
Date Performed:	November 29, 2019
Date Delivered:	November 22, 2019

Contents

1 Introduction

The purpose of this experiment is to develop a significant project using only VHDL and the laboratory development boards. Using these skills that we have learned with stop watches and LCD panels, this project will be a sequential logic design using push buttons to design a working stop-watch and lap timer on the LCD panel. The goal of this experiment is to have a smoothly operating stopwatch that can be synchronized with any accepted stopwatch that accurately times laps.

2 Requirements

Signal Name	**Description**
Clk_50 (inputs)	System Clock
Start	Start/stop button that starts the watch when initially pressed, stops when pressed again, then continues the watch when pressed again.
Reset	Resets stopwatch and lap counter to 0.
Lap	Records the current lap time and updates the lap counter number.
Lcd_en (outputs)	LCD enable
Lcd_rw	LCD Read/Write Select
Lcd_rs	LCD Command/Data Select
Lcd_on	LCD Power on/off
Lcd_data(7 .. 0)	LCD Data
HEX0[0]	Lap Number Segment 0
HEX0[1]	Lap Number Segment 1
HEX0[2]	Lap Number Segment 2
HEX0[3]	Lap Number Segment 3
HEX0[4]	Lap Number Segment 4
HEX0[5]	Lap Number Segment 5
HEX0[6]	Lap Number Segment 6

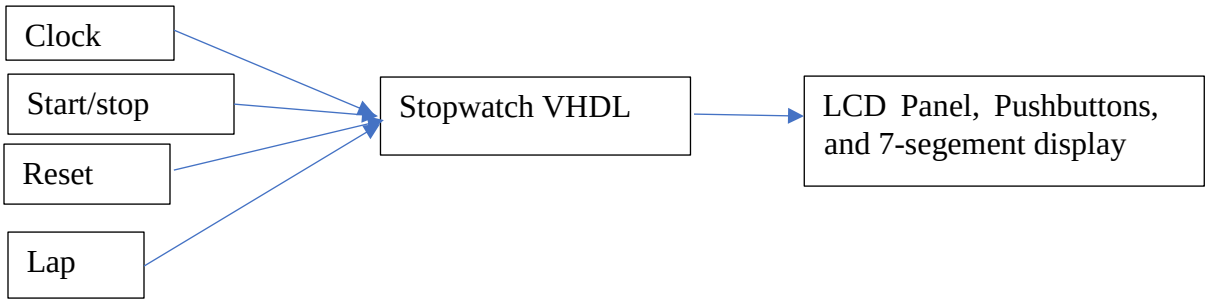
3 Theoretical Design

3.1 Design Narrative

The way this design works is the start/stop button, reset, and lap functions utilize the system clock and the written VHDL code to cause the stopwatch to increase and record

a lap time when activated. These numbers are programmed to display on a LCD panel and a lap number counter on a 7-segment display.

3.2 Top-level design



3.3 Functional description of components

Clock – Utilizes the system's clock function.
Start/Stop – Either brings LCD panel to correct sets state where the lowest digit is increasing the other digits from 0, sets stopwatch to current values, or continues stopwatch from the correct running state.
Reset – Sets stopwatch time, lap time, and lap counter to 00.00 or 0.
Stopwatch VHDL – The code below that utilizes these inputs, outputs, pin assignments, and multiple processes to have a working stopwatch.
LCD Panel – Displays Time and Lap labels, and stopwatch and lap times.
Pushbuttons – Are assigned the start/stop, reset, and lap signals.
7-Segment Display – Displays the lap number counter.

4 Synthesized Design

```
--John Mijares & Marc Abad
--due: 11/25/2019
-- completed: 11/22/2019
LIBRARY ieee;
USE ieee.std_logic_1164.all;
USE ieee.std_logic_arith.all;
USE ieee.std_logic_unsigned.all;
ENTITY projectlaptest IS
PORT (clk_50, start, stop, reset, lap      : IN std_logic;
lcd_en, lcd_rw, lcd_rs, lcd_on             : OUT std_logic;
lcd_data                                   : OUT std_logic_vector(7 downto 0);
lapnum                                     : out std_logic_vector(6 downto 0));
END projectlaptest;

ARCHITECTURE structure OF projectlaptest IS
SIGNAL new_state, count, clk_cout, clk_cout2     : INTEGER := 0;
SIGNAL state                                     : INTEGER := 1;
```

```
SIGNAL clk, clk2, temp, verify : std_logic          := '1'; --we have a stopwatch and lap clock
signal d1,d2                                        : std_logic_vector(7 downto 0);
signal delay                                        : INTEGER := 24000;
signal delay2                                       : INTEGER := 249999;

TYPE name IS ARRAY (0 to 15) of STD_LOGIC_VECTOR(7 downto 0);
SIGNAL   name1, name2                                          : name;
SIGNAL   clock_count, clock_count2                             : INTEGER:=0;
SIGNAL   clock, run, cont, lapclocksw                          : std_logic := '0';
SIGNAL   b3, b2, b1, b0, c3, c2, c1, c0, e0, e1, e2, e3, laap  : std_logic_vector(3 downto 0);
SIGNAL dd3, dd2, dd1, dd0, lp3, lp2, lp1, lp0 : std_logic_vector(7 downto 0);


BEGIN
-- t    i    m    e     r                              l    a    p
name1                                                                                                <=
(x"74",x"69",x"6D",x"65",x"72",x"20",x"20",x"20",x"20",x"20",x"20",x"20",x"20",x"6C",x"61",x"70");
--    10sec sec  . .1sec
name2 <= (dd3,dd2,x"2e",dd1,dd0,x"20",x"20",x"20",x"20",x"20",x"20",x"20",lp3,lp2,x"2e",lp1);

--CLOCK DIVIDER
clock_divider : PROCESS(clk_50) –utilizes the stopwatch and lap clock time processes
BEGIN
if rising_edge(clk_50) then
  if clk_cout < delay then
    clk_cout <= clk_cout + 1;
  else
    clk_cout <= 0;
    clk <= NOT clk;
  end if;
  if clk_cout2 < delay2 then
    clk_cout2 <= clk_cout2 + 1;
  else
    clk_cout2 <= 0;
    clk2 <= NOT clk2;
  end if;
end if;
END PROCESS clock_divider;

--LCM
  lcd_rw <= '0';
PROCESS(clk,start)
BEGIN
```

```
    if start = '0' then
      if cont = '1' then --continue
        state <= 10;
      end if;
      if run = '0' then
        new_state <= 1; --start
        state <= 0;
        end if;

elsif rising_edge(clk) then

    if state = 0 then -- default state
      lcd_en <= '0';
      lcd_on <= '1';
      state <= new_state;

    elsif state = 1 then
      lcd_en <= '1';
      lcd_rs <= '0';
      lcd_data <= x"30";
      state <= 0;
      new_state <= 2;

    elsif state = 2 then
      lcd_en <= '1';
      lcd_rs <= '0';
      lcd_data <= x"38";
      state <= 0;
      new_state <= 3;

    elsif state = 3 then
      lcd_en <= '1';
      lcd_rs <= '0';
      lcd_data <= x"08";
      state <= 0;
      new_state <= 4;

    elsif state = 4 then
      lcd_en <= '1';
      lcd_rs <= '0';
      lcd_data <= x"01";
      state <= 0;
      new_state <= 5;
```

```
elsif state = 5 then
  lcd_en <= '1';
  lcd_rs <= '0';
  lcd_data <= x"06";
  state <= 0;
  new_state <= 6;

elsif state = 6 then
  lcd_en <= '1';
  lcd_rs <= '0';
  lcd_data <= x"0C";
  state <= 0;
  count <= 0;
  new_state <= 7;

elsif state = 7 then
  lcd_en <= '1';
  lcd_rs <= '0';
  state <= 0;
  count <= 0;
  new_state <= 8;

elsif state = 8 then
  lcd_en <= '1';
  lcd_rs <= '0';
  lcd_data <= x"80";
  state <= 0;
  count <= 0;
  new_state <= 9;

elsif state = 9 then --entry mode set
  lcd_en <= '1';
  lcd_rs <= '1';
  lcd_data <= name1(count);
  count <= count + 1;
  state <= 0;
if count = 15 then
  new_state <= 10;
  end if;

elsif state = 10 then
  lcd_en <= '1';
  lcd_rs <= '0';
  lcd_data <= x"C0";
```

```
  state <= 0;
  new_state <= 11;
  count <= 0;

elsif state = 11 then
  lcd_en <= '1';
  lcd_rs <= '1';
  lcd_data <= name2(count);
  count <= count + 1;
  state <= 0;
if count = 15 then
  new_state <= 10;
end if;
end if;
end if;
END PROCESS;
temp <= NOT start;
timer : PROCESS(run, reset, clk2, b3, b2, b1, b0, lap, c3, c2, c1, c0) –timer that creates the true stopwatch b[3 .. 0] and the hidden lap timer c[3 .. 0]
BEGIN

--run <= '1' when start ='0' else
--'0' when stop = '0';

if rising_edge(temp) then
verify <= NOT verify;
end if;

if verify = '0' then
if reset = '0' then
  b3 <= "0000";
  b2 <= "0000";
  b1 <= "0000";
  b0 <= "0000";
  e3 <= "0000";
  e2 <= "0000";
  e1 <= "0000";
  e0 <= "0000";
  c3 <= "0000";
  c2 <= "0000";
  c1 <= "0000";
  c0 <= "0000";
  laap <= "0000";
  cont <= '0';
```

```
end if;
if start = '0' then --start or continue
  run <= '1';
end if;
elsif verify ='1' then
if rising_edge(clk2) then
if start = '0' then --stop
  b3 <= b3;
  b2 <= b2;
  b1 <= b1;
  b0 <= b0;
  e3 <= e3;
  e2 <= e2;
  e1 <= e1;
  e0 <= e0;
  run <= '0';
  cont <= '1';
end if;

if lap = '0' then --lap
if lapclocksw = '0' then
  c3 <= e3;
  c2 <= e2;
  c1 <= e1;
  c0 <= e0;

  laap <= laap + '1';
if laap = "1001" then
  laap <= "0000";
end if;
elsif lapclocksw = '1' then
  e3<="0000";
  e2<="0000";
  e1<="0000";
  e0<="0000";
end if;
else lapclocksw <= '0';
end if;
  e0 <= e0 + '1';
if e0 = "1001" then
  e1 <= e1+ '1';
  e0 <= "0000";
end if;
if e1 > "1001" then
```

```
  e2 <= e2 + '1';
  e1 <= "0000";
end if;
if e2 > "1001" then
  e3 <= e3 + '1';
  e2 <= "0000";
end if;
if e3 > "1001" then
  e3 <= "0000";
  e2 <= "0000";
  e1 <= "0000";
  e0 <= "0000";
end if;
  b0 <= b0 + '1';
if b0 = "1001" then
  b1 <= b1+ '1';
  b0 <= "0000";
end if;
if b1 > "1001" then
  b2 <= b2 + '1';
  b1 <= "0000";
end if;
if b2 > "1001" then
  b3 <= b3 + '1';
  b2 <= "0000";
end if;
if b3 > "1001" then
  b3 <= "0000";
  b2 <= "0000";
  b1 <= "0000";
  b0 <= "0000";
end if;
end if;
end if;
end PROCESS timer;

with b0(3 downto 0) select –4-bit to hexadecimal conversions
dd0 <= x"30" WHEN "0000",--0
x"31" WHEN "0001",--1
x"32" WHEN "0010" ,--2
x"33" WHEN "0011" ,--3
x"34" WHEN "0100" ,--4
x"35" WHEN "0101" ,--5
x"36" WHEN "0110" ,--6
```

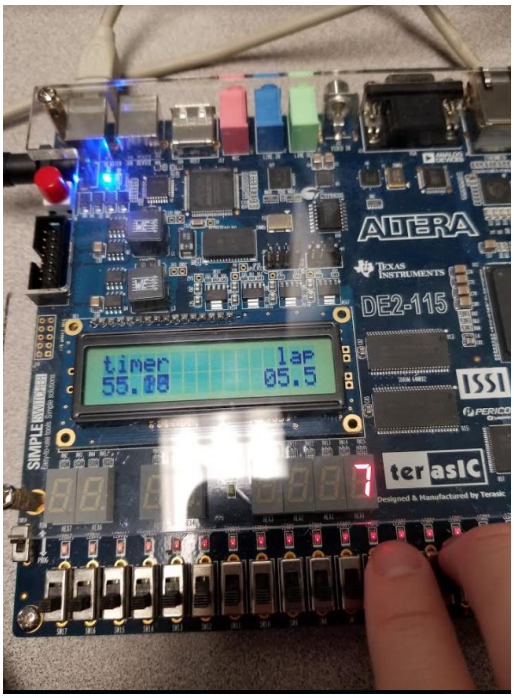
```
x"37" WHEN "0111" ,--7
x"38" when "1000" , --8
x"39" WHEN "1001",--9
x"3A" when others;
with b1(3 downto 0) select
dd1 <= x"30" WHEN "0000",--0
x"31" WHEN "0001",--1
x"32" WHEN "0010" ,--2
x"33" WHEN "0011" ,--3
x"34" WHEN "0100" ,--4
x"35" WHEN "0101" ,--5
x"36" WHEN "0110" ,--6
x"37" WHEN "0111" ,--7
x"38" when "1000" , --8
x"39" WHEN "1001",--9
x"3A" when others;
with b2(3 downto 0) select
dd2 <= x"30" WHEN "0000",--0
x"31" WHEN "0001",--1
x"32" WHEN "0010" ,--2
x"33" WHEN "0011" ,--3
x"34" WHEN "0100" ,--4
x"35" WHEN "0101" ,--5
x"36" WHEN "0110" ,--6
x"37" WHEN "0111" ,--7
x"38" when "1000" , --8
x"39" WHEN "1001",--9
x"3A" when others;
with b3(3 downto 0) select
dd3 <= x"30" WHEN "0000",--0
x"31" WHEN "0001",--1
x"32" WHEN "0010" ,--2
x"33" WHEN "0011" ,--3
x"34" WHEN "0100" ,--4
x"35" WHEN "0101" ,--5
x"36" WHEN "0110" ,--6
x"37" WHEN "0111" ,--7
x"38" when "1000" , --8
x"39" WHEN "1001",--9
x"3A" when others;
with c0(3 downto 0) select
lp0 <= x"30" WHEN "0000",--0
x"31" WHEN "0001",--1
x"32" WHEN "0010" ,--2
```

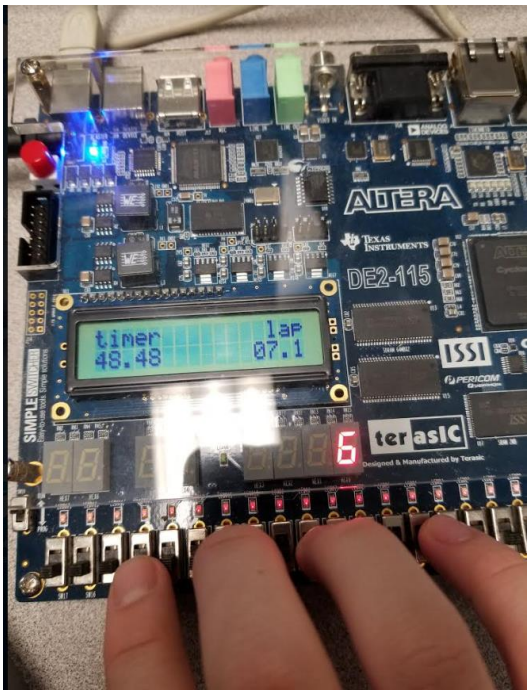
```
x"33" WHEN "0011" ,--3
x"34" WHEN "0100" ,--4
x"35" WHEN "0101" ,--5
x"36" WHEN "0110" ,--6
x"37" WHEN "0111" ,--7
x"38" when "1000" , --8
x"39" WHEN "1001",--9
x"3a" when others;
with c1(3 downto 0) select
lp1 <= x"30" WHEN "0000",--0
x"31" WHEN "0001",--1
x"32" WHEN "0010" ,--2
x"33" WHEN "0011" ,--3
x"34" WHEN "0100" ,--4
x"35" WHEN "0101" ,--5
x"36" WHEN "0110" ,--6
x"37" WHEN "0111" ,--7
x"38" when "1000" , --8
x"39" WHEN "1001",--9
x"3A" when others;
with c2(3 downto 0) select
lp2 <= x"30" WHEN "0000",--0
x"31" WHEN "0001",--1
x"32" WHEN "0010" ,--2
x"33" WHEN "0011" ,--3
x"34" WHEN "0100" ,--4
x"35" WHEN "0101" ,--5
x"36" WHEN "0110" ,--6
x"37" WHEN "0111" ,--7
x"38" when "1000" , --8
x"39" WHEN "1001",--9
x"3A" when others;
with c3(3 downto 0) select
lp3 <= x"30" WHEN "0000",--0
x"31" WHEN "0001",--1
x"32" WHEN "0010" ,--2
x"33" WHEN "0011" ,--3
x"34" WHEN "0100" ,--4
x"35" WHEN "0101" ,--5
x"36" WHEN "0110" ,--6
x"37" WHEN "0111" ,--7
x"38" when "1000" , --8
x"39" WHEN "1001",--9
x"3A" when others;
```

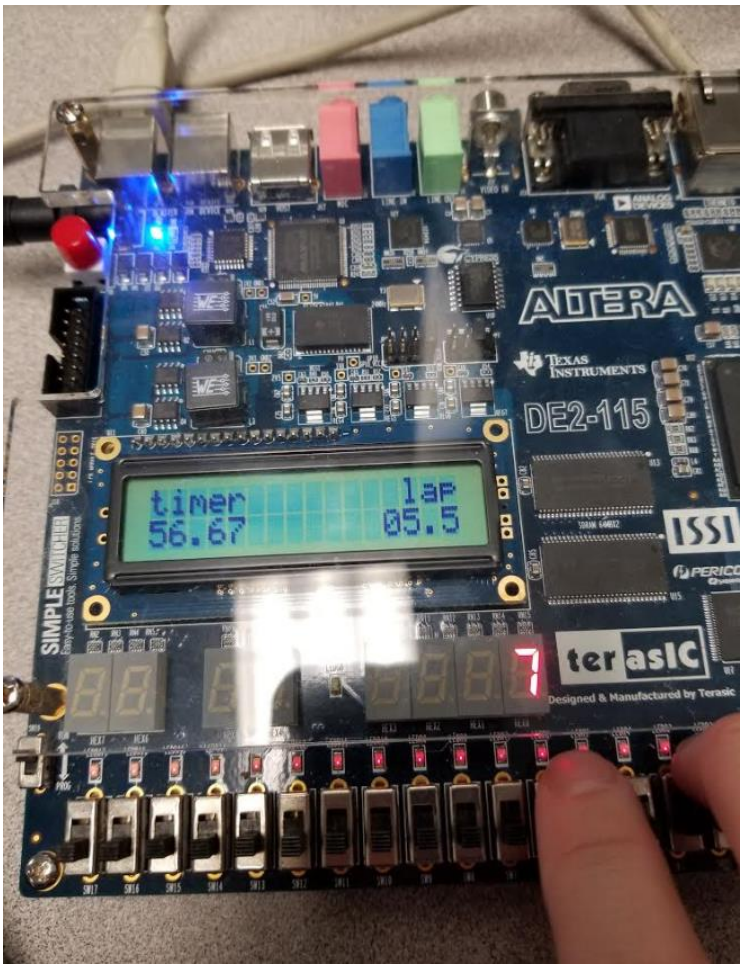
```
with laap(3 downto 0) select
lapnum <= "1000000" WHEN "0000",--0
"1111001" WHEN "0001",--1
"0100100" WHEN "0010" ,--2
"0110000" WHEN "0011" ,--3
"0011001" WHEN "0100" ,--4
"0010010" WHEN "0101" ,--5
"0000010" WHEN "0110" ,--6
"1111000" WHEN "0111" ,--7
"0000000" when "1000" , --8
"0011000" WHEN "1001",--9
"0001111" when others;
end structure;
```

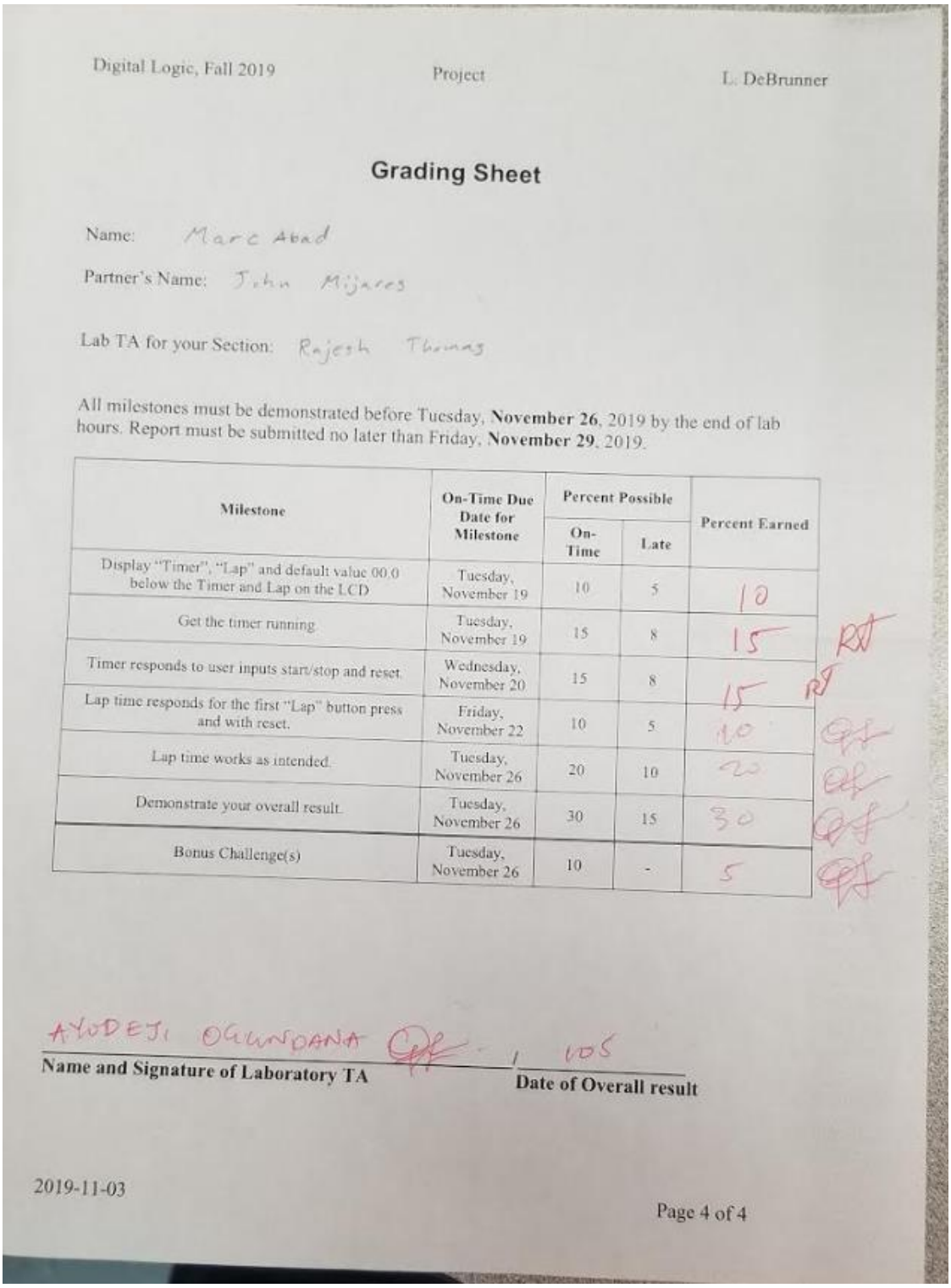
5 Simulation Results

6 Experimental Results






ALTERA
TEXAS INSTRUMENTS
DE2-115
timer
56.67
lap
05.5
ISSI
terasIC
Designed & Manufactured by Terasic



Digital Logic, Fall 2019 Project L. DeBrunner

Grading Sheet

Name: Marc Abad

Partner's Name: John Mijares

Lab TA for your Section: Rajesh Thomas

All milestones must be demonstrated before Tuesday, **November 26**, 2019 by the end of lab hours. Report must be submitted no later than Friday, **November 29**, 2019.

Milestone	On-Time Due Date for Milestone	Percent Possible		Percent Earned
		On-Time	Late	
Display "Timer", "Lap" and default value 00.0 below the Timer and Lap on the LCD	Tuesday, November 19	10	5	10
Get the timer running	Tuesday, November 19	15	8	15 RT
Timer responds to user inputs start/stop and reset.	Wednesday, November 20	15	8	15 RT
Lap time responds for the first "Lap" button press and with reset.	Friday, November 22	10	5	10
Lap time works as intended.	Tuesday, November 26	20	10	20
Demonstrate your overall result.	Tuesday, November 26	30	15	30
Bonus Challenge(s)	Tuesday, November 26	10	-	5

AYODEJI OGUNDANA / 105

Name and Signature of Laboratory TA **Date of Overall result**

2019-11-03

Page 4 of 4

7 Summary

In conclusion, my partner and I were able to develop a working stopwatch and lap recorder. The only design modifications made were to change to a faster and more

suitable delay. The results were successful, and I can conclude that one can correctly design a working stopwatch and lap recorder on a laboratory development board. This test was comprehensive enough to ensure the design meets all and bonus requirements.

8 Lessons Learned

The most important lesson that I learned from this lab was to understand all lab assignments and their applications beforehand.

9 Pre-Lab Assignment

No Pre-Lab Assignment.

END OF DOCUMENT