Lab Notebook

FPGA Capstone Project

## Academic Integrity

*By signing below you acknowledge the CU Honor Code, "On my honor, as a University of Colorado Boulder student I have neither given nor received unauthorized assistance" applies to this assignment.*

Signed: Din Carmon

# Module 1

## Setup

Author: Din Carmon

Date: 8/5/22

Procedure/Description of Test:

Observations:

* On Connecting the board to the computer the following happens:
  + 7-Seg display go from 0 to F in hex format with a ~0.5 sec display for a digit
  + Led light pattern is: even light, uneven light, all lights and repeat
  + D4 (5V led) and conf led are on.

Data:

Images/Drawings:

תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי  
תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

Results:

|  |  |
| --- | --- |
| DE10\_LITE\_Small | |
| Fmax | 300.48MHz |
| Logic Utilization % | <1% |
| # Flip-Flops | * 18 in spi\_ee\_config * 2 in reset delay * 1 for Cont * 3 in oLED * **24 FF in total** |
|  |  |

|  |  |
| --- | --- |
| DE10\_LITE\_Default | |
| Fmax | 96.1MHz |
| Logic Utilization % | 1% |
| # Flip-Flops | * 18 in spi\_ee\_config * 2 in reset delay * 1 for Cont * 3 in oLED * 1 in VGA\_Audio\_PLL * 7 in VGA\_OSD\_RAM * 13 in VGA\_Controller * **45 FF in total** |
|  |  |

Questions:

1. Record your observations of the board behavior once the FPGA is programmed. Does it behave as you might expect?

* Yes. The Default version takes more logic elements due to more peripherals.
* In the default version pulling SW0 up makes the board respond to the accelerometer. Lights are like an air bubble in water
* Connecting the VGA outputs a picture.

Conclusions:

Lessons Learned (What did you learn?):

* How to program an FPGA
* How to have fun!!

## Part 1

Author: Din Carmon

Date: 11.5.22

Procedure/Description of Test:

* HEX0, HEX1 as a 7-SEG display of SW3-0 and SW7-4 binary value respectively

Observations:

* The display matches our desired application
* A binary value bigger than 9 output a 7-seg display of 0

Data:

Images/Drawings:

תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

Results:

|  |  |
| --- | --- |
| Fmax | No constraints were written |
| Logic Utilization % | <1% |
| # Flip-Flops | 0 |
|  |  |

Questions:

1. Based on your observations of the board behavior once the FPGA is programmed, does it behave as you might expect?

* Yes

1. Explain the reason for the number of flip-flops used in the design.

* In principle FF are needed for memory. However there is no clock / memory in our design.

Conclusions:

Lessons Learned (What did you learn?):

* Functions in VHDL
* Basic process rules in VHDL

## Part 2

Date: 11.5.22

Procedure/Description of Test:

* SW3-0 translate to 2-digit binary representation.

Observations:

* As designed

Data:

תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

Images/Drawings:

Results:

|  |  |
| --- | --- |
| Fmax | No constraints were written |
| Logic Utilization % | <1% |
| # Flip-Flops | 0 |
| With V = 0, z |  |
| With V = 0, A |  |
| With V = F, z |  |
| With V = F, A |  |
|  |  |

Questions:

1. Based on your observations of the board behavior once the FPGA is programmed, does it behave as you might expect?

* YES,

1. Does this design use more or less logic than the design in Part 1? Why?

Less logic. Before it used 15. Now 9 logic elements

Conclusions:

Lessons Learned (What did you learn?):

* If else in VHDL

## Part 3

Date: 12.5.22

Procedure/Description of Test:

* Showing in LEDR an adder output

Observations:

* Works as desired

Data:

Images/Drawings:

תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

Results:

|  |  |
| --- | --- |
| Fmax | No constraints were written |
| Logic Utilization % | <1% |
| #Flip-Flops | 0 |
|  |  |

Questions:

1. Based on your observations of the board behavior once the FPGA is programmed, does it behave as you might expect?

* YES

Conclusions:

Lessons Learned (What did you learn?):

* Nothing new really…

## Part 4

Date: 12.5.22

Procedure/Description of Test:

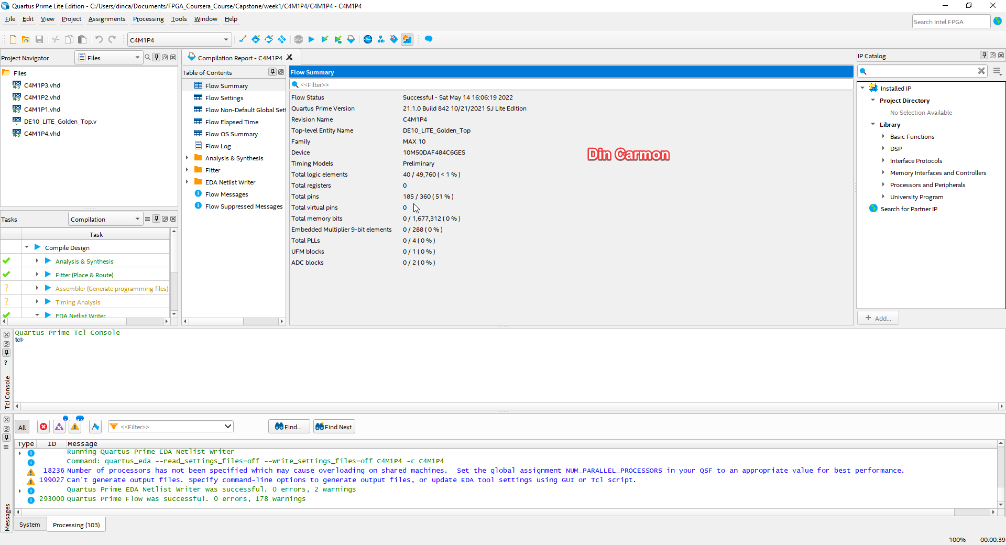
* Combination of all the last parts

Observations:

* As expected

Data:

Images/Drawings:



Results:

|  |  |
| --- | --- |
| # Logic Cells | 40 |
| Logic Utilization % | <1% |
|  |  |
|  |  |

Questions:

1. Based on your observations of the board behavior once the FPGA is programmed, does it behave as you might expect?

* Yes

Conclusions:

Lessons Learned (What did you learn?):

* Using perused modules is awesome

## Part 5

Date: 14.5.22

Procedure/Description of Test:

* Combination of all the last parts

Observations:

Data:

Images/Drawings:

תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

Results:

|  |  |
| --- | --- |
| # Logic Cells | 40 |
| Logic Utilization % | <1% |
|  |  |
|  |  |

Questions:

1. Based on your observations of the board behavior once the FPGA is programmed, does it behave as you might expect?

* Yes!

1. How does this compare to the number of Logic Cells in Part 4?

* Same! The compiler is smart.

Conclusions:

Lessons Learned (What did you learn?):

* You can trust the compiler to create the required architecture.

# Module 2

## PWM

Author:

Date:

Procedure/Description of Test:

Observations:

Data:

Images/Drawings:

Results:

|  |  |
| --- | --- |
| Fmax |  |
| % Logic Utilization |  |
| Total registers |  |
|  |  |

Questions:

1. Did the LED change brightness depending on the setting of the first 3 switches?

Conclusions:

Lessons Learned (What did you learn?):

## ADC

Author:

Date:

Procedure/Description of Test:

Observations:

Data:

Images/Drawings:

Results:

|  |  |
| --- | --- |
| Fmax |  |
| %Logic Utilization |  |
|  |  |
|  |  |

Questions:

1. Does the board behave as you expected?

2. Is this a good voltmeter as is?

3. What could you change in either the board hardware or FPGA logic to make it perform better?

Conclusions:

Lessons Learned (What did you learn?):

# Module 3

## NIOS II Hardware Design

Author:

Date:

Procedure/Description of Test:

Observations:

Data:

Images/Drawings:

Results:

|  |  |
| --- | --- |
| Fmax |  |
| Logic Utilization |  |
|  |  |
|  |  |

Questions:

1. What is the purpose of the Avalon Memory-Mapped Clock-Crossing Bridge?

Conclusions:

Lessons Learned (What did you learn?):

# Module 4

## NIOS II Software Design and System Test

Author:

Date:

Procedure/Description of Test:

Observations:

Data:

Images/Drawings:

Results:

|  |  |
| --- | --- |
| Fmax |  |
| Logic Utilization |  |
|  |  |
|  |  |

Questions:

1. Is the control of the 10 LEDs implemented in hardware or in software?
2. Is the control of the 7-segment LEDs done by hardware or by software?
3. How much memory is required to run your Nios II program? Can you fit it into the onchip RAM if you redesign the onchip RAM block?
4. Your Nios II processor is running at what clock speed? How much faster can it run in your MAX10 design?

Conclusions:

Lessons Learned (What did you learn?):