By: Chloe Atwood
Date: 12/08/2024
ID Number: 893387514
Lab Section: G

CPRE 281 Extra Credit Lab Write-Up

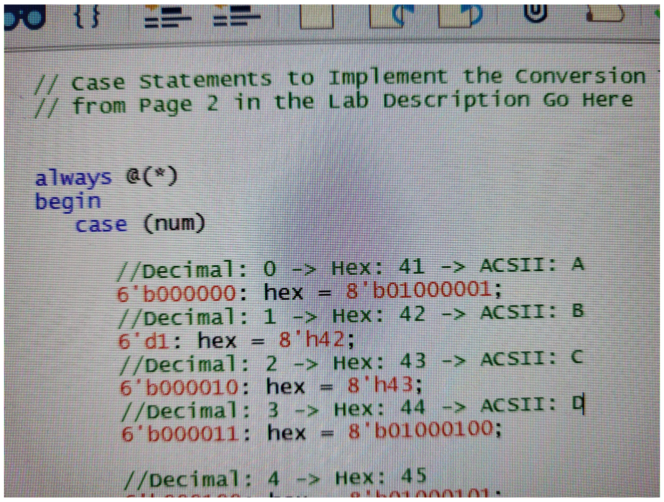
Purpose:

The purpose of this Extra Credit Lab was to take the 4-bit register created in Lab 12 and turn it into a 6-bit register. This would then be used to convert a 6-bit binary input into an 8-bit Base64-Hex output that would then be displayed on an LCD. The goal was to be able to display the letters C and A.

Procedure:

The first step in this process was to alter some files from Lab 12 so that the register would be able to support 6-bits instead of just 4-bits. This was accomplished by first adding two more registers to the reg4b file. All that was done for this was adding two more copies of the register symbol and connecting them to the CLK, IN, OUT, LOAD, and CLRN lines. The buses for the IN and OUT ports had to be changed to 6-bit instead of 4-bit as well. Once that was done, a couple of the Verilog files from Lab 12 had to be changed in order to support the new 6-bit register. The Mux8_4b file needed its inputs and outputs changed from 4-bit to 6-bit values, and the regfile file required the DATAP and DATAQ wires to be changed from 4-bit wires into 6-bit outputs, and the original DATAP and DATAQ outputs needed to be removed. The input and output bit sizes in the rest of the file also had to be changed to support 6-bits. The last thing that needed to be done to the Lab 12 files was creating a symbol file for the new register so it could be used in the wiring diagram.

The next step was to create the base64.v converter that would convert the binary input to the appropriate 8-bit base64-hex output. This was done in a case statement by setting each 6-bit binary value to its equivalent in hex. There are a couple of different ways this could be done. As seen in the image below, a couple of different methods were tested, and all successfully worked.



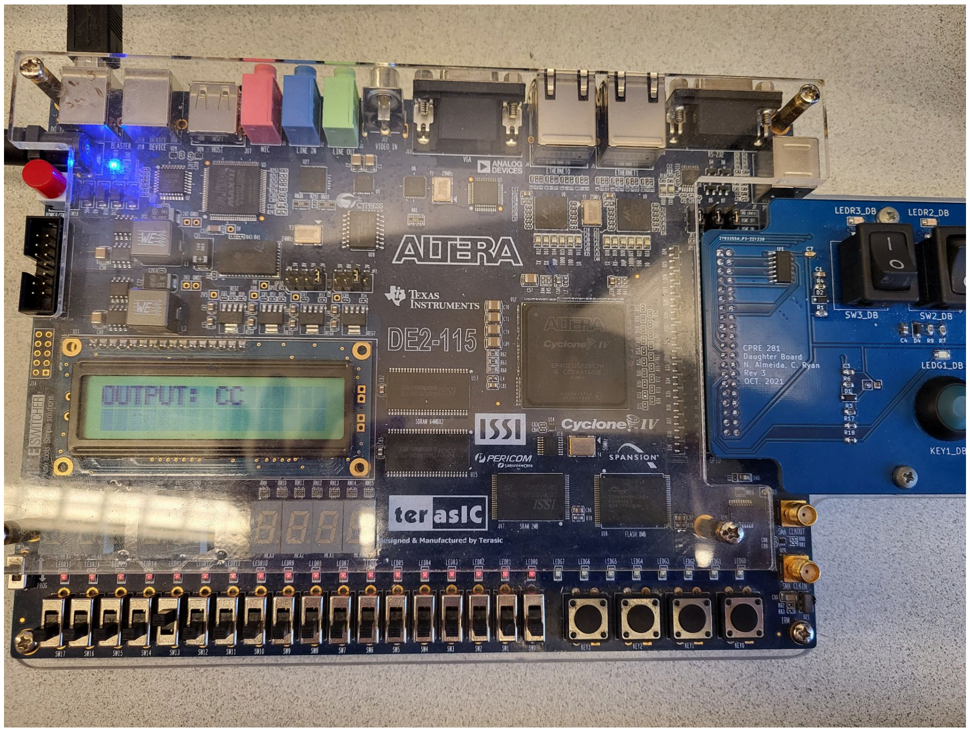
```
// Case Statements to Implement the Conversion
// from Page 2 in the Lab Description Go Here


always @(*)
begin
   case (num)

      //Decimal: 0 -> Hex: 41 -> ACSII: A
      6'b000000: hex = 8'b01000001;
      //Decimal: 1 -> Hex: 42 -> ACSII: B
      6'd1: hex = 8'h42;
      //Decimal: 2 -> Hex: 43 -> ACSII: C
      6'b000010: hex = 8'h43;
      //Decimal: 3 -> Hex: 44 -> ACSII: D
      6'b000011: hex = 8'b01000100;

      //Decimal: 4 -> Hex: 45
      6'b000100: hex = 8'b01000101;
```

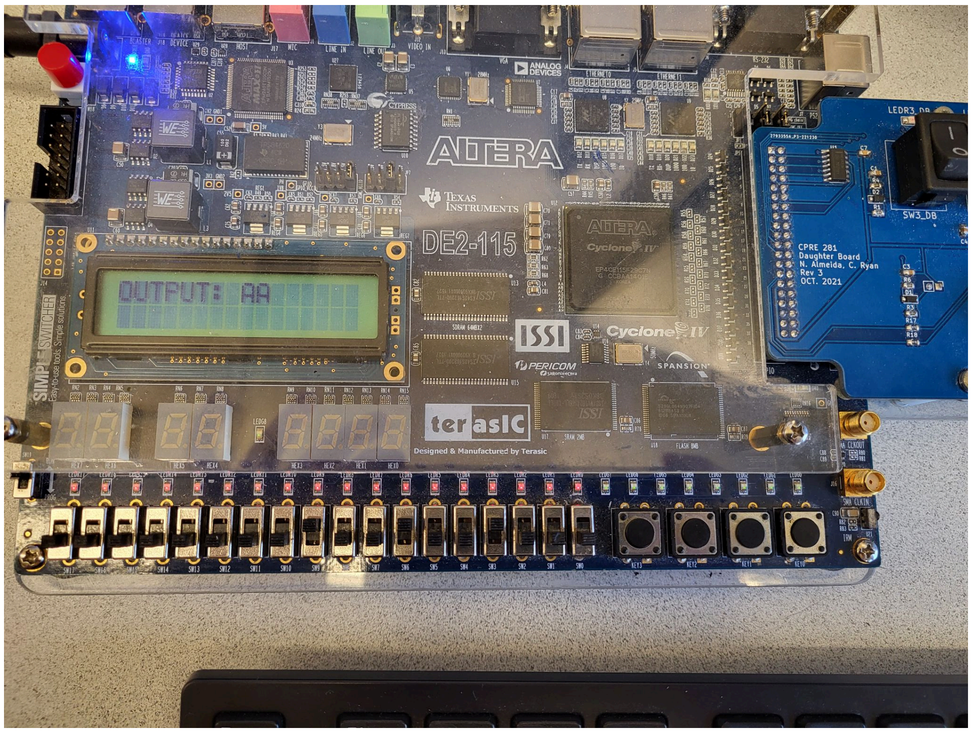
The final steps involved adding the register symbol to the already started wiring diagram and assigning the rest of the pins so that it could be tested with the LCD. The following chart displays the pin assignments:

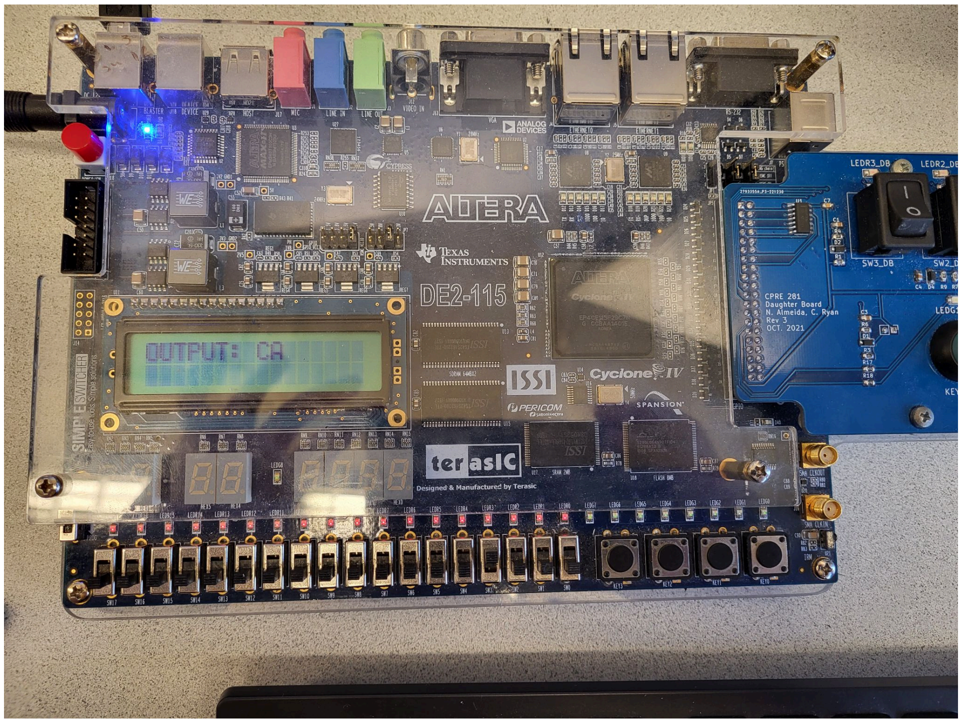
Signal Name:	FPGA Pin No:	Description	Pin Assigned:
SW17	PIN_Y23	Slide Switch[17]	InData[5]
SW16	PIN_Y24	Slide Switch[16]	InData[4]
SW15	PIN_AA22	Slide Switch[15]	InData[3]
SW14	PIN_AA23	Slide Switch[14]	InData[2]
SW13	PIN_AA24	Slide Switch[13]	InData[1]
SW12	PIN_AB23	Slide Switch[12]	InData[0]
SW11	PIN_AB24	Slide Switch[11]	WA2
SW10	PIN_AC24	Slide Switch[10]	WA1
SW9	PIN_AB25	Slide Switch[9]	WA0
SW8	PIN_AC25	Slide Switch[8]	RP2
SW7	PIN_AB26	Slide Switch[7]	RP1
SW6	PIN_AD26	Slide Switch[6]	RP0
SW5	PIN_AC26	Slide Switch[5]	RQ2
SW4	PIN_AB27	Slide Switch[4]	RQ1
SW3	PIN_AD27	Slide Switch[3]	RQ0
KEY3	PIN_R24	Push-Button[3]	CLK
KEY2	PIN_N21	Push-Button[2]	WR
KEY0	PIN_M23	Push-Button[0]	CLRN
SW3_DB	PIN_AB22	Rocker Switch[3]	LCDON

Results:

The result of this Lab was being able to save the letter C to Reg[0] and the letter A to Reg[1] and display them on the LCD as initials:






OUTPUT: CA
ALTERA
Texas Instruments
DE2-115
ISSI
Cyclone IV
PERICOM
SPANSION
terasIC
Designed & Manufactured by Terasic
ANALOG DEVICES
CPRE 281
Daughter Board
N. Almeida, C. Ryan
Rev 3
OCT. 2021
LEDR3_DB
LEDR2_DB
SW3_DB
SIMPLESWITCH