Signed Binary Arithmetic

Leonardo Aviles, *tue51778@temple.edu*

i. INTRODUCTION

In this lab the task was to perform signed binary arithmetic on the Zybo board. Signed binary is the same as unsigned except the first digit or most significant bit if the value is a 1 the number is a negative decimal number while if the most significant bit is a zero the value is a unsigned number. In signed binary the numbers will range from -128 to 127 decimal.

The specifications of the project are as follows.

- Using Tera Term 2 8-digit number is entered which will be converted to an 8-bit number.
- An operation is entered either addition or subtraction depending on the ASCII character.
- A way to check if there is a overflow when adding or subtracting two numbers that either exceed 127 or 127
- The sum or difference of the two 8-bits is then transmitted to tera term with a line break in between the line where the data is entered.

To complete the project, the modules rcvr.v and txmit.v, are reused. The following new module was implemented.

- "GenData" in this module the code received the data from the user and creates the two 8-bit.
- "AOS" was used to perform the addition or subtraction
- IP blocks adder and a subtract block are used.
- "BinaryBCD" converts the binary value to BCD
- "ValOut" converts the bcd digit to a ASCII character

ii. DISCUSSION

A. Top Module: "Lab3"

This module acted as the controller module sending the output and input data to their respective modules such as the RXD and TXD values.

B. Generate Data: "gendata.v"

In this module in order to convert the ASCII value to a 8 bit binary number the switches and buttons were used in order to have the user enter the data. In order to input the data bit by bit switch 1 controlled the lower nibble and button 3 to 0 controlled the digit placement. For the higher nibble switch 2 was used and when active high the bit would be placed in their respective spot using the buttons. The operation was also found during this module and if addition was the desired

C. Transmit Data "txmit" and Receive Data "rcvr"

The 'rcvr' module was used in order to receive the data from the Tera Term program and store the value into a reg called rbr. While the 'txmit' was used in order to check if the value was ready to be transmitted or received.

D. "AOS":

This module would perform the addition and subtraction of the data bits. In order to find which operation was needed the register OP was used and the desired operation occurred

E. "*BinaryBCD":*

This module would perform the binary to bcd conversation need. In order to do this the divider IP block was used and the code shown in the class powerpoint slides that were provided by Dr. Silage

F. "*ValOut":*

In this module the BCD is converted to ASCII values by adding a hex 30 to each digit. Once the conversion is complete the number is placed in TDIN and transmitted to tera term.

iii. CONCLUSION

The project was not implemented correct when trying to program the Zybo board. The area of error that was found was receiving the data from the user. When using the button and switch I believe that the value would never be update or saved properly. Another area of error could have been when using the addition and subtraction blocks. For this area the module did not update the different values and the addition/subtraction of a two bytes that were filled with zero were done. Once the data was transmitted since the TDIN was full of zeros the null character was printed.

iv. Appendix

Figure 1: Shows the controller and data path and the different signals and data received