Homework #8

Out: November 28, 2022
Due: December 6, 2022 at 11:59 PM
Points: 185

Serial Divider Design

For this assignment you are to design and implement the CPLD for a serial divider using a Xilinx XC95108 or Lattice ispMACH4128ZE. You must use VHDL to implement the design and you must demonstrate your implementation to a TA. The hardware for testing the design is supplied.

The pinout files are available (divider.ucf for the XC95108) and include all of the pin definitions for the XC95108 and ispMACH4128ZE devices. No logic is included and no internal nodes are defined. There is also a sample VHDL file that adds or subtracts two 16-bit values input from the keypad (addsub16.vhd). You may use this file as the basis for your design.

The VHDL files, including test benches, and any supporting documents must be turned in. You must turn in a hard copy of the design files and also submit all files, including the test benches, electronically via the website.

System Description

For this project you are to implement a serial divider for 16-bit unsigned binary numbers. The user inputs two 16-bit (four hex digits) numbers and then the system calculates their quotient. There is a toggle switch for the user to indicate which value (divisor or dividend) is being input and a push button switch to indicate the calculation should be done. The actual values are input using a 16-key keypad (keys are labeled '0' to 'F'). If more than 16-bits are input for a value the high nibble (four bits) is lost and the new nibble is shifted into the low order bits. The system should implement division using one of the serial division schemes discussed in class. Finally, the input values and results are to be displayed on 7-segment LEDs which are multiplexed by the CPLD. To properly multiplex the display each digit should be on for approximately one millisecond out of every 10-20 milliseconds.

Hardware Description

The hardware consists of a 16-key keypad with a 74C923 keypad controller chip to debounce the keypad and twelve 7-segment LEDs, driven by ULN2003's, for displaying values. In addition there is a toggle switch to indicate whether the dividend or divisor is being input and a push button switch to signal the division is to be done. All of this is controlled by a XC95108 or ispMACH4128ZE. Additionally, there is a 29M16 on the board that can be used to decode the multiplexed digit outputs. The PALASM code (dec412.pds) and programming file (dec412.jdc) are available for this chip, though there is no reason to change this chip. There is also a 16V8 on the board that does the binary nibble to 7-segment decoding. The PALASM code (segdec.pds) and programming file (segdec.jdc) are also available for this chip, though again there is no reason to change them. Both of these chips are jumpered to allow them to be disabled when implementing extra credit features. Notice that if you are using

the 29M16 for digit decoding, IO32 to IO43 on the XC95108 or F8 to F13 and G2 to G14 on the ispMACH4128ZE **must** be disabled using the tri-state control. If you are using the 16V8 for segment decoding, IO47 to IO49 and IO52 to IO55 on the XC95108 or E0, E2, E4, E6, E12, E14, and F0 on the ispMACH4128ZE **must** be tri-stated. The keypad controller chip can also be disabled to allow the keypad scanning and debouncing to be done by the CPLD (extra credit). Notice that the row and column signals to the keypad (IO6 through IO13 on the XC95108 or B0 to B13 on the ispMACH4128ZE) **must** be tri-stated if the 74C923 and not the CPLD is being used to scan and debounce the keypad. See the attached schematics (XC95108 schematic and ispMACH4128ZE schematic) for more details.

Extra Credit

Below are a number of possibilities for extra credit. In order to get the extra credit it must be demonstrated with the working system. It is **not** allowed to *sequentially* demonstrate the extra credit. That is, all extra credit must be demoed at the same time and implemented in the same CPLD design.

Remap Keypad (10 points)

Remap the keypad to a more natural order.

Preserve Dividend and Divisor (20 points)

Preserve the dividend and the divisor so that at the end of the calculation, the dividend, divisor, and quotient are all displayed. Note: the challenge is to get the design to fit.

Preserve Display while Calculating (10 points)

Do not allow the LEDs to change while doing the calculation. It is acceptable to blank the display while doing the division, but it is not acceptable for the display to be blanked the entire time the "Calculate" push button switch is pressed.

Compute and Display the Remainder (30 points)

Display the remainder in addition to the quotient at the end of the calculation. This may be done by displaying the remainder instead of the dividend or divisor or by using the toggle switch to choose between quotient and remainder display after the division is completed.

Divide-by-Zero (10 points)

Detect and report attempts to divide by zero.

Segment Decoding (30 points)

Implement the divider with the segment decoding in the CPLD instead of the external PAL 16V8. The 16V8 should be disabled by removing the appropriate jumper.

Digit Decoding (20 points)

Implement the divider with the digit decoding (a one of 12 decoder) in the CPLD instead of the external PAL 29M16. The 29M16 should be disabled by removing the appropriate jumper.

Keypad Scanning (70 points)

Scan and debounce the keypad with the CPLD (*i.e.* don't use the 74C923 outputs). Note that you will need to simulate a bouncing keypress to test this.

Homework #8 Resources

- VHDL pin definition file for Xilinx XC95108
- Xilinx XC95108 board schematic
- Lattice ispMACH4128ZE board schematic
- 16-bit adder/subtracter example in VHDL
- slides describing adder/subtracter example
- PALASM code for digit decoder
- programming file for digit decoder
- PALASM code for segment decoder
- programming file for segment decoder
- Homework Q&A
- Electronic Submission

Last updated November 24, 2022 01:13 PM by glen@caltech.edu

copyright © 1999 - 2022, Glen George. All rights reserved. Reproduction of all or part of this work is permitted for educational or research use only, provided that this copyright notice is included in any copy.