

CSE4117 Microprocessors

HW2

PART 1 (Logisim) (50 points)

You will implement in Logisim a system which has a Vertebrate CPU and its associated I/O circuits. You will also write the specified software that will run on this system.

In particular, your system will contain the following components:

- A Vertebrate CPU + a PIC for interrupt processing.
- One 4-bit switchboard (SW)
- Two 1-bit switches, PB1 and PB2
- One 4*7-segment display
- One timer

Your system will have the following tasks:

1. Implement SHL "shift left" instruction in Vertebrate. You decide the format of the instruction.
2. You have to enter digits 0-9 and operations +, -, * from the 4-bit switch. For each one of these, set the switchboard to the data you want to enter and press PB1.
 - 0-9 will be represented by 0000-1001 in binary
 - Addition (+) key is represented by 1111
 - Subtraction (-) key is represented by 1110
 - Multiplication (*) key is represented by 1101
 - Equal (=) key is represented by 1100

You should write a program which behave exactly like a calculator which is capable of adding, subtracting and multiplying numbers. You should only enter positive numbers and your results should always be positive. There will be no privilege in calculations, i.e., if you enter $3+2*10$ your result will be 50, not 23. In other words, + and - will have the same privilege.

Note that you have each number digit by digit. For example, if you want to enter 34, you will first enter the number 3 (i.e., 0011) then the number 4 (i.e., 0100) from the switchboard. On the software side, you have to combine the digits to obtain the entered number. (i.e. **number = number*10+new_digit = number<<3 + number<<1+ new_digit**). You **must** use your newly designed shift left instruction here.

Use polling for your calculator. Then your code will continuously poll the 4- bit switchboard (4-bit switchboard and PB1 switch) and read the entered digit as soon as PB1 is pressed.

3. You must also implement a timer. The timer must send an interrupt at approximately every 1 second passed to CPU and update the value of time in its 16-bit data register. CPU must read the timer value from this data register in its ISR routine.

4. If the switch PB2 is pressed, the CPU will display the calculator output. Otherwise, it will display the timer output.

5. **ALL DATA DISPLAY MUST BE IN DECIMAL.** So you should convert your results into decimal format before displaying. The conversion process must be written in software.

6. **DO NOT** use built-in register in Logisim, you have to design your register using S-R and/ or D FFs provided by Logisim. You can use memory, encoders, decoders, multiplexers, gates, adders etc. directly from Logisim.

7. Your system must work with the 4 KHz clock of Logisim.

8. Update the assembler (in marmaralecture.com) as necessary.

PART 2 (FPGA) (50 points or 75 points)

You will implement in FPGA a system which has a CPU and its associated I/O circuits. You will also write the specified software that will run on this system.

You MUST write your program in SystemVerilog, NOT in Verilog.

Implement a CPU on FPGA and connect it with a 4x7-segment display. You will also use the two buttons on the FPGA board.

At the start, CPU must display the number 1 on 4x7 segment. When the user presses the left button, the number on display is incremented by 1. When the user presses the right button, the number will be multiplied by 2.

--- If you do this by continually polling both buttons, and use BIRD, you will get 50 points.

--- If instead you do this by controlling increment button with polling but multiply by 2 button with interrupt and use Vertebrate, you will get 75 points.

As in question 1, you have to update the assembler and display all your numbers in decimal. Conversion to decimal must be done by a software subroutine.