1. For Task 1, what did you measure as the pushbutton's make bounce and break bounce times on the oscilloscope? If you were not able to detect any bounce, then say so.

I could not detect any bounce, there was some noise but were no bounces. However, when I pressed the pushbutton for increment, I could see the bounce because it sometimes would increment two times.

2. For Task 2 you were provided the subroutine var_delay and told that the delay duration is approximately 0.1 ms * r16. Briefly describe how you would efficiently verify the validity of this relationship using simulation.

3. Why is the flip-flop set when you press the pushbutton in Task 2? If you press the pushbutton once, could the flip-flop be set more than once? Why?

The output will stay 1 until the clear button is active. So the flipflop will be set just once.

4. What happens, as displayed on the bargraph, in Task 3 as you change from one 8-bit input value to the next?

The bargraph displays what I have switched on.

5. Briefly explain what the purpose of the flip-flop is in Task 4? Could you reliably accomplish conditional I/O without a flip-flop involved?

The purpose is to debounce from the pushbutton and display the output at a one time

6. In Task 4, what should be done with the asynchronous input PRE of the 74HC74 and why?

7. What is the minimum pulse width you must create at the CLR input of the 74HC74 to clear the 74HC74 flip-flop? Must this be a positive pulse or negative pulse? Explain you answer.

Must be a positive pulse because if I clear and come back to quickly it will still read the input which is always 1

8. How do you know that the 74HC74 flip-flop can be reliably operated at 3.3V? Explain your answer.

Because it says on the table

9. Briefly describe the circuit you used to interface the pushbutton to the microcontroller in Task 4 and why you used that particular circuit.

I used d flipflop to debounce the make bounce coming from pushbutton which is debouncing in hardware and for break bounce I have debounced it with software and programed that it purposely delays and wait until bounce to go away.

10. What is the polling rate for Task 1 in your program? That is, how many times per second is the pushbutton read during the polling of the pushbutton’s output?

About 10 times.