Lab Worksheet

**Lab Number (circle this week’s lab)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

**Name**: Chimzim Ogbondah

**Lab Section**: 1

**Lab Partner Name**: Ruiyu Son

This lab worksheet is the final deliverable for a lab. You will usually have three deliverables for a lab:

1. **Prelab assignment BEFORE LAB**: Posted with the lab manual, typically involves a system sketch, submitted in Canvas before the start of your lab section, may be worked on and used by lab partners in class on Tuesday during lab planning
2. **Demonstrations IN LAB**: Demonstrated/discussed with a TA in lab and recorded using a demo evaluation sheet to be printed and signed in lab (functional demo of a lab milestone, debug demo using debugging tools to explain something about the internal workings of your system, Q&A demo showing ability to formulate and respond to questions)
3. **Postlab assignment AFTER LAB**: Submitted in Canvas before the start of your next lab section, may be reviewed by lab partners in class on Tuesday during lab retrospective, consists of three items (prelab planning boards, lab notes, and lab retrospective)

Deliverable #1 has its own Canvas assignment submission. (10 points)

Deliverable #2 has an evaluation sheet that is printed in lab, used as a checklist, and submitted to your TA. The TA will enter points in Canvas based on the demo evaluation sheet. (40 points)

Deliverable #3 has its own Canvas assignment submission. (30 points)

This worksheet will help you develop the items needed for deliverable #3.

1. **PRELAB PLANNING BOARDS**

Question Board: What are the three priority questions from your lab planning work?

* 1. How will data from the ADC look like on a graph?
  2. What registers are required in order to set up the ADC?
  3. How the formula works to convert a number from 0-4095 to a mm value in distance?

1. Task Board: What are several tasks you identified in your planning (for you and lab partner)?
   1. Going through the Trivia datasheet to understand how to correctly initialize the registers
   2. Understand which registers we needed to initialize to set up the ADC
   3. Figure out which flag register and its bit position to ensure that the ADC was configured (While loop)
   4. Measure two data points so a number could be used in the equation to turn 0-4095 into a distance in millimeters
2. **LAB NOTES**

During lab, keep notes about the following so that you can submit information with this deliverable.

1. Results related to the three priority questions (might be answers, might be more questions, write brief summaries, don’t include code files)
   1. We looked through the lab manual sheet together and at a graph of what the IR sensor data looks like on a graph. Because of this we were able to determine that we needed our two points for the distance equation to be after 10 mm and before 70 mm. We looked at some slides on canvas as well as the registers down below to help us know which registers, we needed to configure the ADC and then used the trivia sheet to ensure that we were setting the right bits to high (1.). After we set up the ADC, we looked back through the Bai book and our prelab at how the distance was supposed to be calculated. We didn’t get to measure these two points however because lab was closing to an end.
2. Any additions, refinements, or corrections to the prelab system sketch based on what you learned (include an updated sketch, or briefly describe at least one update you made)
   1. I didn’t make any real changes to my prelab
3. Description of your debug demo (what did you demo and why, what did you find, a paragraph is fine, may want to include a screenshot)
   1. Initially inside of our read function we have a while loop that takes the ADC flag as an argument checking to see if it is empty or not. At first we just set the mask in the while loop and left it as is because it worked that way for us in lab 5. We couldn’t see any data being printed to the LCD screen and so I stepped through the entire main method then stepping into our **int number = adc\_read();** after this we stepped through each line which brough us to the while loop statement where our green bar never left explaining we had an infinite while loop. We demoed this to the TA and then quickly set a conditional in the while loop, so it only jumped in the loop when the flag was == 1. After doing this we should again how the register was empty and so it stepped over the while loop return an integer value which got placed in our variable *Number.*
4. **LAB RETROSPECTIVE**

Take 10-15 minutes and answer these questions as you think about your lab experience. You don’t need to describe everything, try to pick something notable.

1. What did we set out to do?
   1. We set out to complete the lab, which entailed configuring the ADC creating another function that would return the value of the IR sensor and the graph it so we could turn the data into a numerical distance in millimeters.
2. What actually happened?
   1. We finished initializing the ADC registers, it did take some time as we wanted to be sure we did everything correctly. We looked at the trivia data sheet to make sure we were enabling and clearing the right bits inside the registers based on which ADC we were using. We then wrote the adc\_read() function and debugged it so it no longer had an infinite while loop.
3. Why did it happen?
   1. This happened because we didn’t want to go back and forth on if our software code was working or if we initialized the registers the correct way and so we just referred to the book in order to be sure.
4. What are we going to do next time (to improve)?
   1. I feel like we worked fairly productive for this lab. Using structured pairing I don’t really know if there was anything we could have done collectively to help us in progress and then separately outside of lab time too.