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**: Ruyiu Sun

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**
2. Question Board: What are the three priority questions from your lab planning work?
   1. Understand how the UART receives and transmits data
   2. How to set the UART pins in C so they can transmit and receive data
   3. Understanding how the get data and receive works with the Putty
3. Task Board: What are several tasks you identified in your planning (for you and lab partner)?
   1. Configure the UART so that way we could send and receive data
   2. Figure out how the get data function works with the PIN on the microcontroller
   3. Debug through both having the UART send a character back to the putty and then sending characters to the putty from the keyboard.
4. **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. During the lab time through going back and forth with the TA we saw that we did our prelab section wrong for the UART data sheet part which led us to not set the bits to their correct value. Upon understanding our mistake, we then reset them to 0x02 and then 0x01 so that way one could transmit data and the other one could receive. It was the first two bits that needed to be set because PB0 and PB1 corresponded with bits 0 1. We did our debug demo by showing the putty receiving and sending pits. The first time we set a character equal to a and then put it in the function. This showed us that the value was being held in the register and so whenever we would send a character to the microcontroller it would send an ‘a’ back. We then were able to type whatever by using the other method and saw it appear on the putty screen. This helped knowing what PB0 and PB1 worked with transmitting and receiving.
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 made corrections to my data sheet for the table part. At first I was following things from the slides on where to but 1s or 0s and then Xs. After talking with the TAs I asked why the UART needed to be set to 0b0000 0011 and he said because of the PINS, from their I talked and showed currently on my pre lab that I knew the Pins for Rx and Tx and then I was able to understand more clearly their bit positions and how it is just two wires that work to transmit and receive data. Form there I was able to update my prelab table, so It looked more like are C code.
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. showing the putty receiving and sending pits. The first time we set a character equal to a and then put it in the function. This showed us that the value was being held in the register and so whenever we would send a character to the microcontroller it would send an ‘a’ back. We then were able to type whatever by using the other method and saw it appear on the putty screen. This helped knowing what PB0 and PB1 worked with transmitting and receiving.
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 and understand how the putty works while also reaffirming how to set different parts pins on the microcontroller in c
2. What actually happened?
   1. There was trial and error with understanding how the PINs lined up in the port and why the bit positions made sense. After restructuring we were able to have the right bit assignments and then show how the putty was receiving different data from, the keyboard or a given character in our C program.
3. Why did it happen?
   1. This happened because of how our pre labs were. After fixing the prelab/understanding more things went over well
4. What are we going to do next time (to improve)?
   1. Ask more questions prior to lab about the prelab. Work with my lab partner on the prelab.