Mohammad Kedwaii - mak3799
Zachary Pope - zhp76
EE445L - 15660

Lab 2 Report

A. Objectives

The main goals of this lab were to practice different debugging techniques, understand critical sections, and learn about probability mass functions and the Central Limit Theorem. Some debugging techniques we covered were reading values using an oscilloscope and logic analyzer, writing interrupts to dump data, and setting up breakpoints to run through the lines of the code. By collecting data, we were able to draw PMFs, and by sampling the data at different rates, we could understand the concept of the Central Limit Theorem.

B. Measurement Data

Lab Preparation

2. a. The DCW statements hold the port memory addresses, such as Port F's direction address is 0x4002.5400

b. The entire subroutine takes ~125ns (include the branch) but just toggling PF1 takes ~100ns

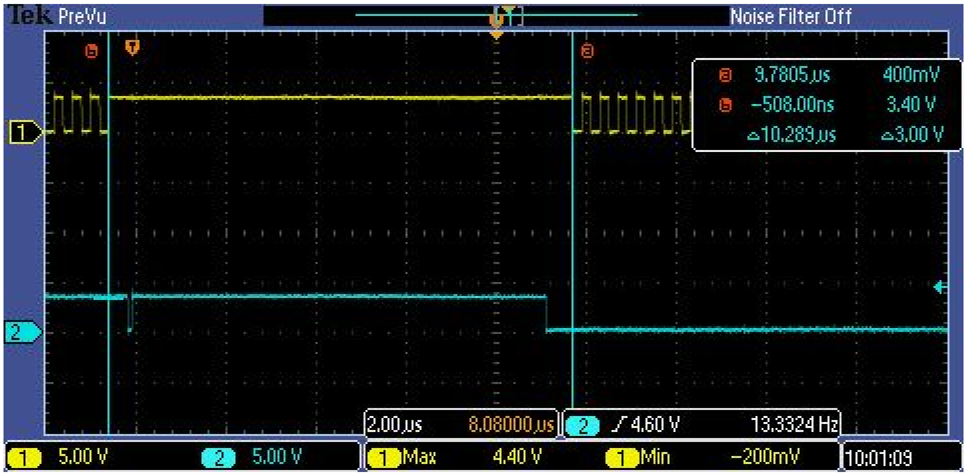
c. The first LDR loads the base address for port F into R0, and the second LDR loads the PORTF_DIR_R address into R0

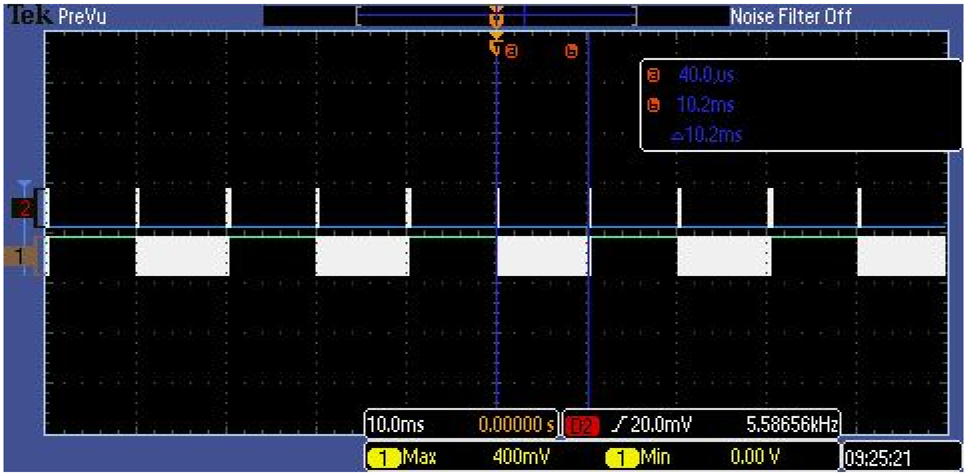
d. Instead of loading Port F's base address, just directly link to the PORTF_DIR_R address to save an LDR call

e. No, a critical section is not created since PF2 is accessed in a bit-specific method (it was defined at the top).

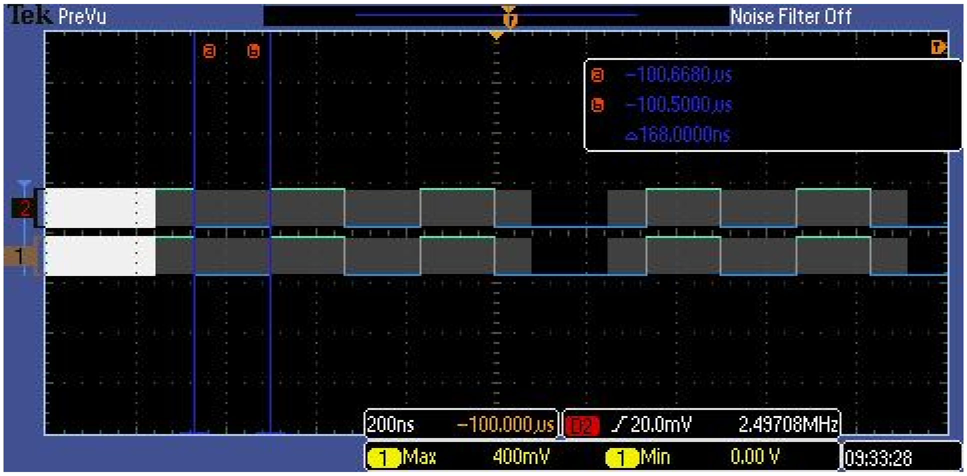
Pictures

PF1 vs PF2 during ISR

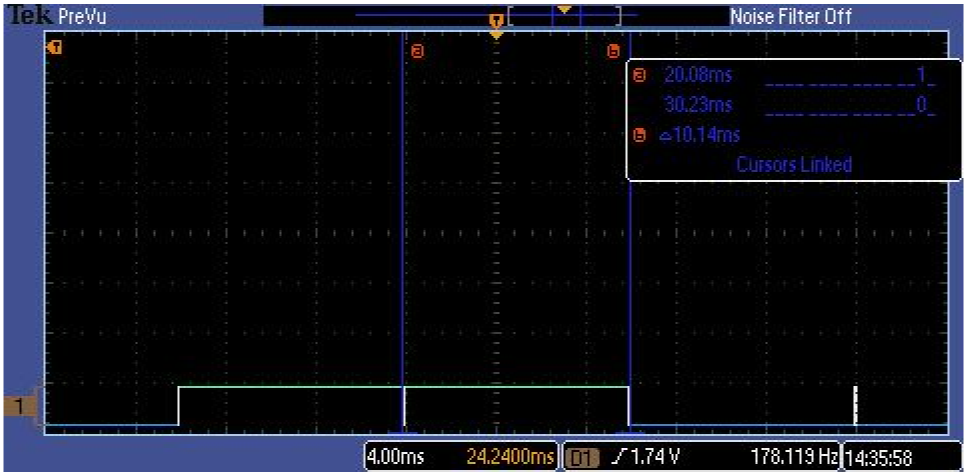




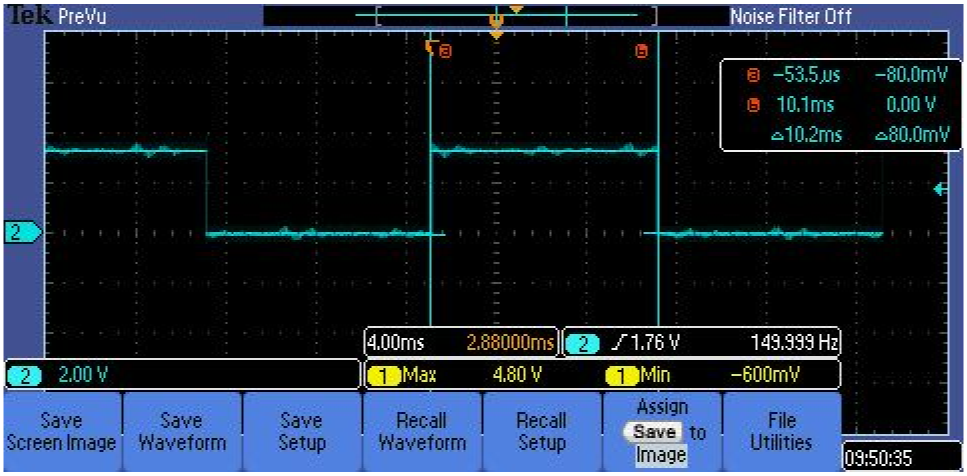
Main toggling PF1



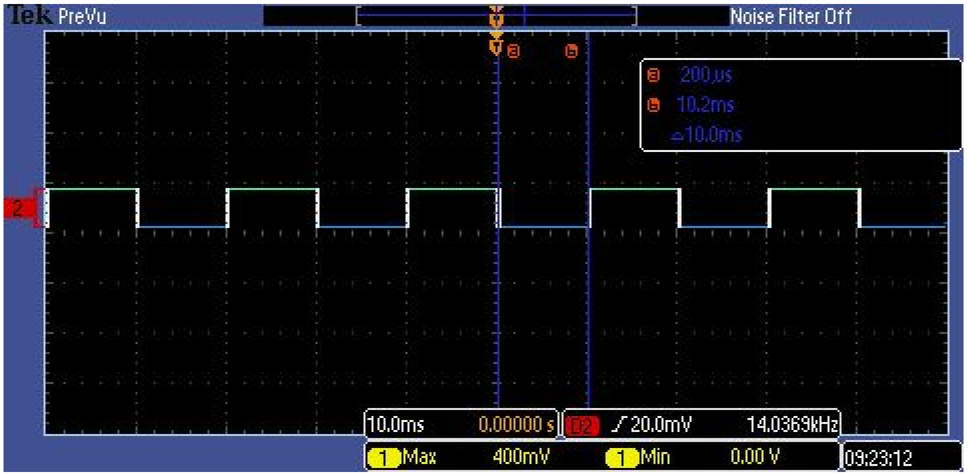
Interrupt with critical section



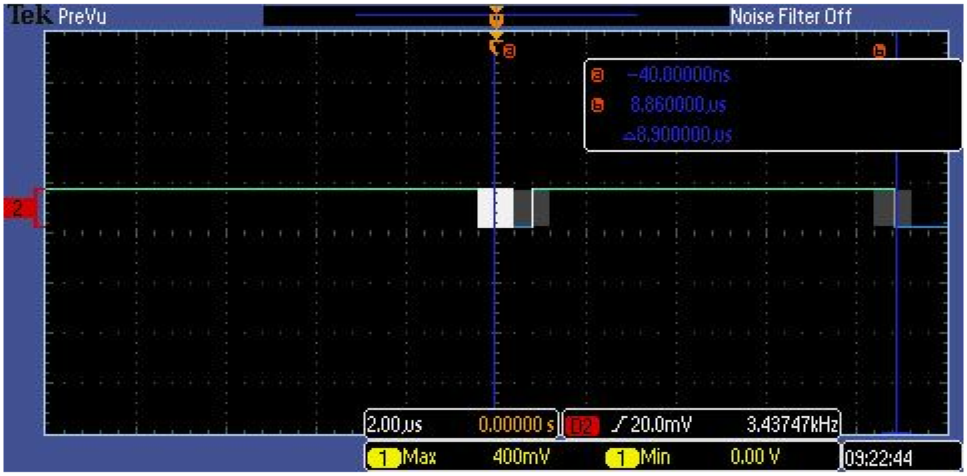
Timer0 interrupt timing at 100Hz (Analog)



Timer0 interrupt timing at 100Hz (Digital)



Interrupt duration without critical section



Time jitter - 52 cycles = 6.5 x 10^{-4} ms
Hardware averaging - By averaging samples into the data that is dumped, we are effectively increasing the number of samples we are using. This causes the PMF to thin out. By the Central Limit Theorem, collecting more samples gives us a distribution whose mean is closer to the actual value and whose variance is closer to zero.

C. Analysis and Discussion

1) **The ISR toggles PF2 three times. Is this debugging intrusive, nonintrusive or minimally intrusive? Justify your answer.**
It is minimally intrusive, since it only adds a few more lines of code to the ISR and does not take up a large amount of time relative to the ISR or the main function..

2) **In this lab we dumped strategic information into arrays and processed the arrays later. Notice this approach gives us similar information we could have generated with a printf statement. In ways are printf statements better than dumps? In what ways are dumps better than printf statements?**
With a printf statement, you can see the values you are getting while you collect them. However, a dump takes far less time than a printf statement because it does not have to send data to the screen or whatever output you have set.

3) **What are the necessary conditions for a critical section to occur? In other words, what type of software activities might result in a critical section?**
A critical section would occur if more than one thread is accessing the same data or memory, such as a global variable. Using an interrupt that accesses a global variable or running multiple instances of the same process would result in a critical section.

4) **Define “minimally intrusive”.**
Minimally intrusive means the debugging method does affect the execution time of the overall project, but the extra time is small enough to not have a significant impact on runtime.

5) **The PMF results should show hardware averaging is less noisy than not averaging. If it is so good why don’t we always use it?**
Hardware averaging is not always used because collecting more samples decreases throughput, since you have to wait to collect enough samples before averaging and passing the value to the ISR and dump.