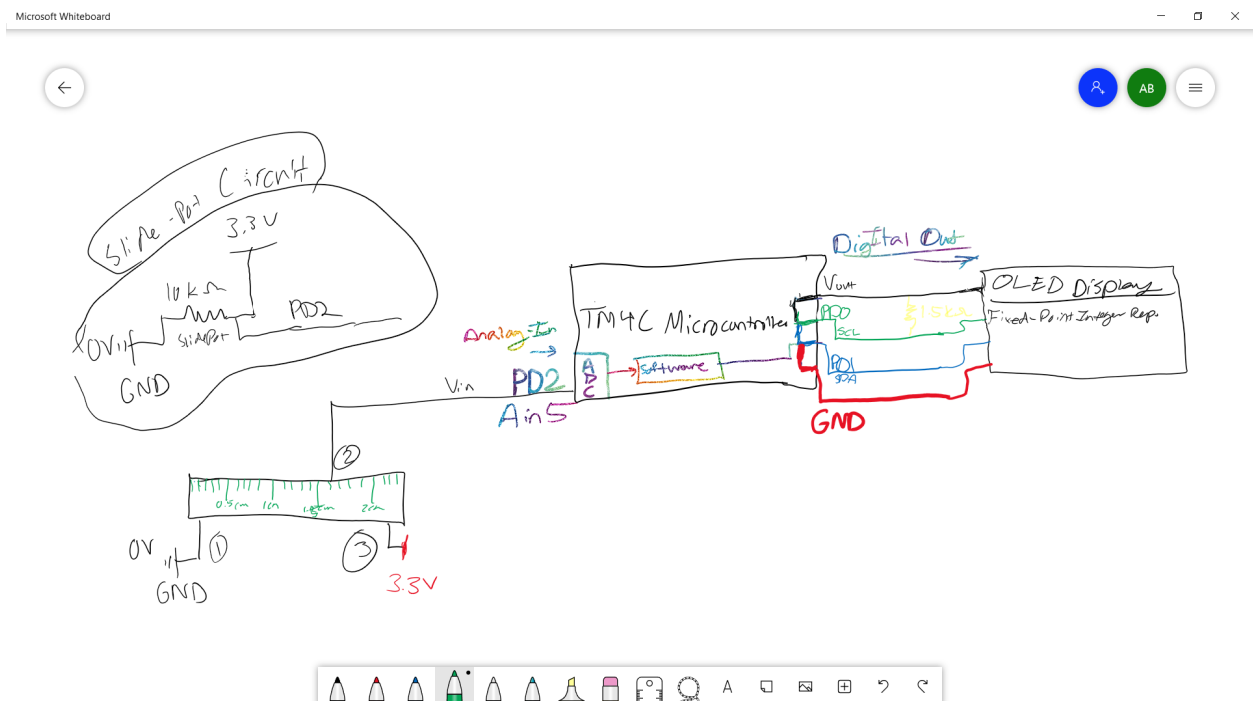


Names: Ayan Basu
 EID: ab73287
 Dr. Yerraballi - Unique #17070
 13 April 2021

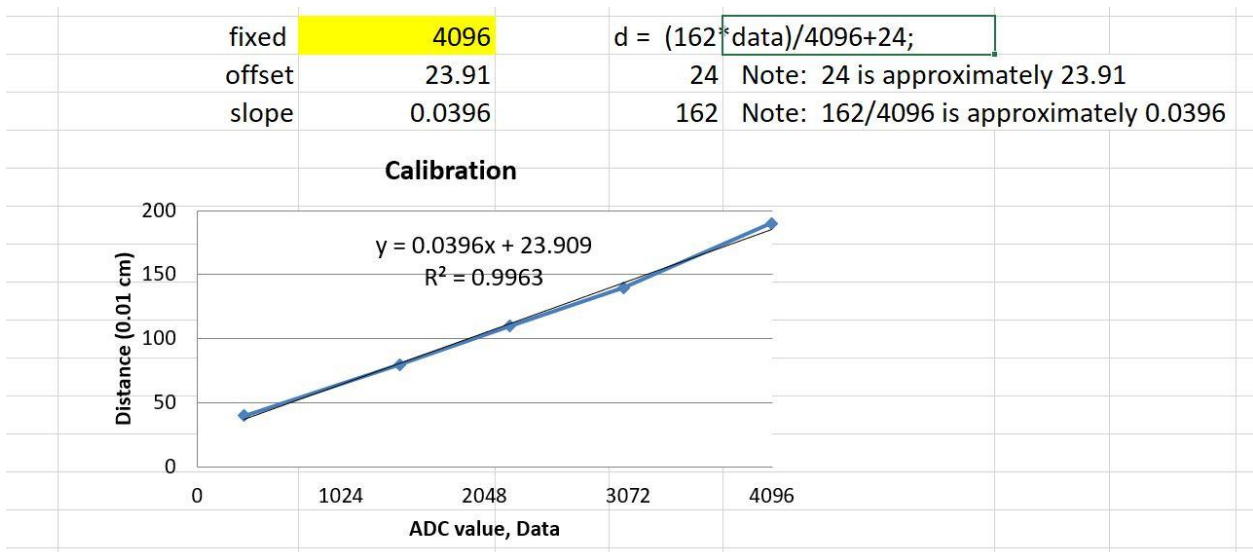
Lab 8 Deliverables

Deliverables

PART A: Circuit Diagram → Showing the Position Sensor (like Figure 8.1, Part A)



PART B: Four Time Measurements Showing the ADC/LCD Execution Time (Part D - Lab 8 Doc)



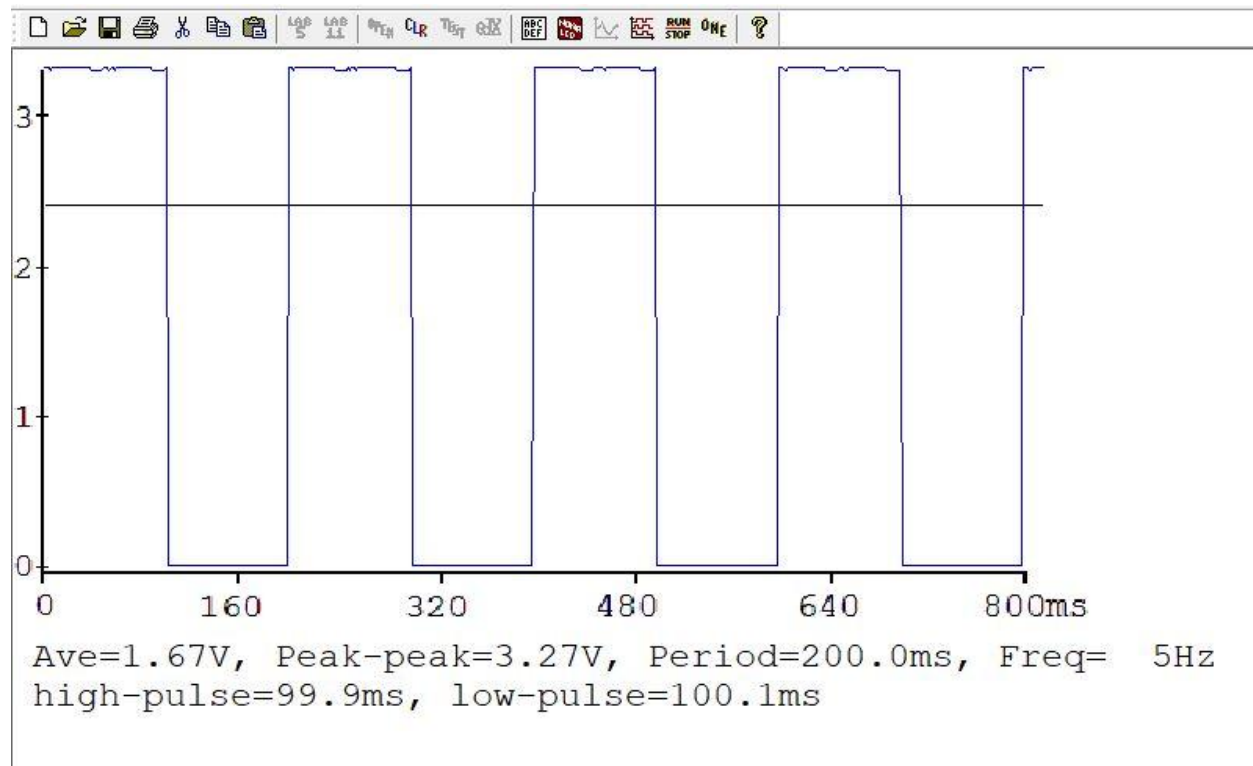
PART C: Calibration Data (like Table 8.1, Part D)

	A	B	C	D	E	F	G	H	I
1		Do this calibration yourself, fill in yellow-highlighted fields							
2		Position (0.01 cm)	ADC Data			Fixed-Point Output	Error	percent	
3		40	334	111556	13360	37	-0.003	0.30%	
4		80	1445	2088025	115600	81	0.001	0.10%	
5		110	2225	4950625	244750	112	0.002	0.20%	
6		140	3040	9241600	425600	144	0.004	0.40%	
7		190	4090	1.7E+07	777100	185	-0.005	0.50%	
8		y	x	x*x	xy				
9	N	5							
10	sum	560	11134	3.3E+07	1576410				
11									
12		fixed	4096		d = (162*data)/4096+24;				
13		offset	23.91		24	Note: 24 is approximately 23.91			
14		slope	0.0396		162	Note: 162/4096 is approximately 0.0396			

PART D: Observations about the Nyquist Theorem (Lab 8 Doc - Part G)

F

PART E: A Photo/Screenshot Verifying Sampling Rate is 10 Hz (Like Figure 8.8 - Part I)



PART F: Accuracy Data & Accuracy Calculation (Like Table 8.2 - Part J)

True position x_{ti}	Measured Position x_{mi}	Error $x_{ti} - x_{mi}$
0.3 cm (30)	0.32	-0.020
0.6 cm (60)	0.62	-0.020
1.0 cm (100)	0.93	-0.030
1.5 cm (150)	1.52	-0.020
1.6 cm (160)	1.61	-0.010

The Number of Data Sets: 5

The Maximum Error is: 0.030

The Average Error is: 0.020

Average Accuracy (with units in cm): 0.020 cm

Accuracy is measuring the difference between the true value and the measured value

In your lab report, answer these open-ended questions (one or two sentences each):

- 1) What factors limited the accuracy of the device?
 - a) Some factors that contributed to the limitations of measuring accuracy were not limited to: incorrect calibration, not measuring perfectly (the ruler is based on a rough eyeball estimation based on the slide-pot distance)
- 2) What conclusions can you make based on this data in Table 8.2?
 - a) The results shows the affects of the accuracy of a slide-pot data in relation to the calibration that was determined from earlier. The better the calibration (meaning the more tightly woven the data was), the more accurate the measurements are.

LAB 8 Demonstration Questions

1. TA may look at your data and expect you to understand how the data was collected and how the ADC and interrupts work

2. You should be able to explain how the potentiometer converts distance into resistance, and how the circuit converts resistance into voltage.

Trying to find voltage across series of resistors and ground

3. Also, be prepared to explain how your software works and to discuss other ways the problem could have been solved

4. What will you change in your program if the potentiometer were to be connected to a different ADC pin? How would this system be different if the units of measurement were inches instead of centimeters

5. What is your sampling rate?

Dependent on the highest frequency

6. What do you mean by sampling rate?

The highest frequency we can move the potentiometer, sample 2x the highest frequency 10 times per second would be 10hz

7. What is the ADC range, resolution, and precision?

Range is 3.3, Precision - 4,096, Resolution = Range/Precision

8. How do you initialize the SysTick interrupt?

9. How can you change your sampling rate?

10. Be prepared to prove what the sampling rate is using the calculator and the lab manual

11. Explain how, when an interrupt occurs, control reaches the interrupt service routine.

12. Why is it extremely poor style to output the converted data to the LCD inside the SysTick ISR?

13. Where is the interrupt vector located?

14. What are the differences between an interrupt and a subroutine?

Done more automatically, subroutine would need to be done

15. What will happen if you increase your sampling rate a lot?

ISR wouldn't execute, step on its own toes, slows everything down, ISR time > sampling rate

16. At what point do you think your program will crash?

The above would cause it to crash.

17. What is the Nyquist Theorem?

If you have an idea of how fast your potentiometer moves, Need to have at least 2x of the frequency with the periodic sample

18. How does it apply to this lab?

Potentiometer asked to test the nyquist theorem, if you went above 5 times per second, program isn't right

LAB 8 FAQ'

FAQ

1. Is anyone having issues with main2? My program gets stuck at OutDec and when I stop it, it jumps to the hard fault handler.

Make sure you have disabled the interrupts for the ADC. Remember you will be doing a software trigger on the ADC read. To check if the ADC_INTERRUPT configuration is the issue, you could do a DisableInterrupt() call before writing to the LCD and follow it with EnableInterrupt().

2. Where should put the code for the SysTick Init and Handler?

SystickInit can be in any file as long as you can call from the main(). As for the handler, it should be able to share the Mailbox and the corresponding Status Flag with the main(). So, it should be in the same file where main() function exist. Ideally, we would like to see a corresponding .h & .c file for Systick module. Modular codes are reusable and thus preferred.

3. I'm confused about where exactly we're supposed to solder wires to the potentiometer. Is it the skinny pins on the ends, the shorter pins on the sides, or the holes at the ends?

You solder on the pins on the end. There should be 3 pins. Two are close together and on the opposite side is 1. The other pins on the sides can be ignored. The pins that are on the long sides, that are extensions of the frame, aren't actually pins I believe. One of the pins on the side with two pins goes to V_in; the other two remaining pins on the potentiometer go to either ground or 3.3V. The V_in pin is specific but the other two can be connected to either ground or 3.3V.

You have an option of simply plugging the slide pot into the breadboard, without the need to solder.

4. When I was making the measurements to calibrate the measured length with the ADC samples, when I measure the ADC samples for the distances 1.8 - 2.0 cm, they all have the same exact range of values of ADC samples. Does anyone know what I can do about this?

The ends of the ADC tend to be non-linear. You need to either implement a piece-wise function or see if you can gather enough points to make an accurate conversion. (i.e 2 does not read as 1.8)

5. When testing the main2 on the screen, is it supposed to be unstable? (i.e. going from 2022-2023-2021 very fast like a blink)

That should be fine if the final product doesn't blink too much. (i.e the screen doesn't flicker between 1.8 and 1.9, but a slow flicker between 1.855 and 1.856 is tolerable.) Also double check your connections and try pressing the slide-pot into the bread board to make sure your design is as stable as possible.

Activate hardware averaging. See the ADC0_SAC_R register.

6. What are ADCMail and ADCStatus? They are not, to my knowledge, defined in any of the programs. Are we supposed to define them ourselves? If so, what do we do with them later? I think it's safe to assume that there is an alternate name in one of the files somewhere.

The term mailbox and "passing data" just refers to how we are getting data from one function to another. In this case, if we are running out SysTick handler at 10Hz and reading ADC_In in that call, how can we get the result of ADC_In to our main loop? Likewise how does the main loop know when there is new data in the mailbox? By writing to a global variable and setting a flag each time ADC_In is read within the SysTick_Handler, our main loop can poll that flag and output the global when it is set, then clear the flag. If your systick is in another C file, make sure to extern it inside of that file's corresponding header file and include it in your main C file.

7. Our solution works in simulator, but it jumps to the hardfault handler on the real board somewhere during a waitforinterrupt(). What would be a reason for this?

Every time you have to wait for the clock or another register to initialize, add more wait commands(NOPs). The hardware may take longer to initialize everything since there are more registers and polling needed. Also, do not use RCGC0 RCGC1 RCGC2 registers anywhere in your software. Using these registers will lead to hard faults. Using RCGCGPIO and RCGCADC, along with friendly coding solved their problems.

8. What if I break one of the pins on the pot? Assuming it is not pin2, you could make a weird (nonlinear) transducer with just pins 1,2. connect pin 2 to 220 resistor, other end of 220 ohm connects to 3.3V; connect pin 2 to your input PD2; connect pin 1 to ground

let d be the integer part of distance 0 to 200 (meaning 0 to 2.0cm)

Let R be the resistance between pins 1,2, $0 < R < 10000$, linear with distance, let $R = 10000 * d / 200$

$V = R / (220 + R)$ (goes from 0 to 3.22), but it will be nonlinear

Let N be the 12-bit ADC (0 to 4095)

$$3.3 * N / 4096 = 50d / (220 + 50d)$$

$$(220 + 50d) * 3.3 * N / 4096 = 50d$$

$$220 * 3.3 * N / 4096 = 50d - 50d * 3.3 * N / 4096$$

$$220 * 3.3 * N / 4096 / 50 = d * (1 - 3.3 * N / 4096)$$

$$220 * 3.3 * N / 4096 / 50 / (1 - 3.3 * N / 4096) = d$$

calibrate this way, let $d = A * N / (1 - B * N)$, where A and B are calibration coefficients