ECEN 5053

Fall 2017

Lab 2: Rotary Position Sensing

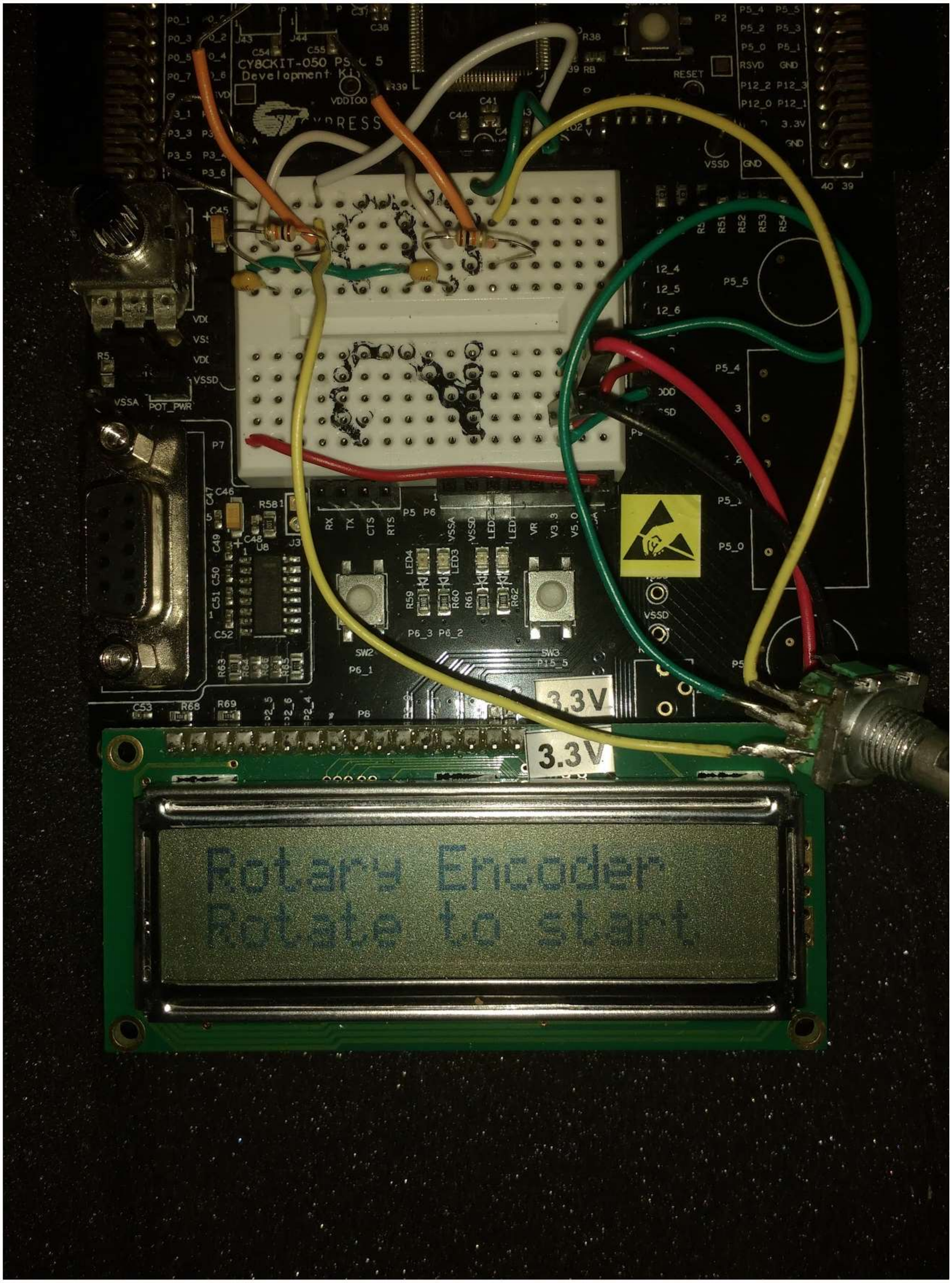
9-21-17

Monish Nene

Angela Maio

Group 7.

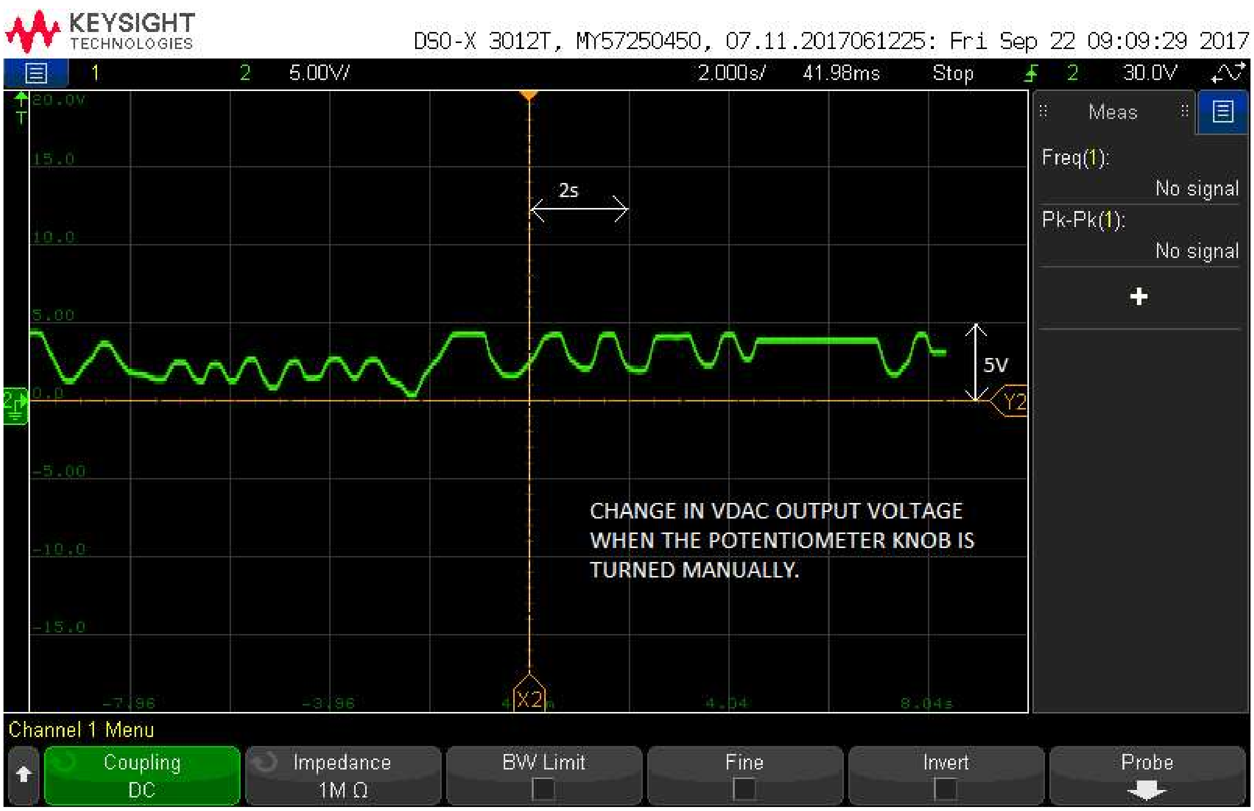
Circuit Diagram:

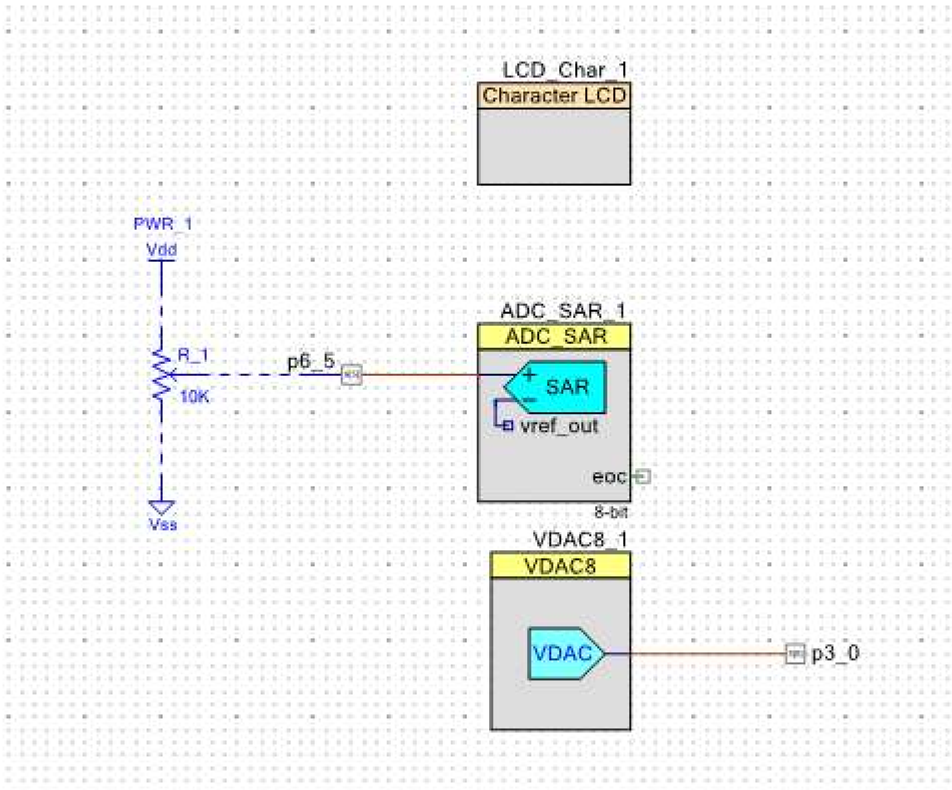

Rotary Encoder
Rotate to start
3.3V
3.3V

Schematic:

Potentiometer

The potentiometer on the PSOC board is internally connected to pin p6_5. p6_5 is connected to ADC which converts the voltage across p6_5 to 8 bit digital value. This value is always positive as the SAR_ADC is in VSS to VDD (single ended) mode. The digital output of the ADC is used to calculate the angle of rotation of potentiometer at that instant. A VDAC is used to again convert the output of the ADC to volts and is shown on the DSO.





Goal: The purpose of this lab is to learn how to interface two different rotary position sensors (a potentiometer and the EN11 rotary encoder) to a microprocessor such as the Cypress PSoC Chip. A secondary goal is to observe and compensate for switch bounce.

Equipment and Parts Required: Cypress PSoC CY8CKIT-050B development board, BI Technologies EN11 Rotary Encoder, oscilloscope.

Preliminaries: The EN11 rotary encoder has pins but they do not fit well on the PSoC dev kit breadboard. You should solder three 5- 10" long wires to pins A- C (locate the data sheet online) and connect the wires to the breadboard area. We want to be able to turn the switch by hand without dislodging the wires.

Challenge 1 (Potentiometer): Interface the R56 potentiometer to the

Demo 1: show this to a TA or the instructor for verification. You may also submit a video to the D2L dropbox to prove you have done this correctly.

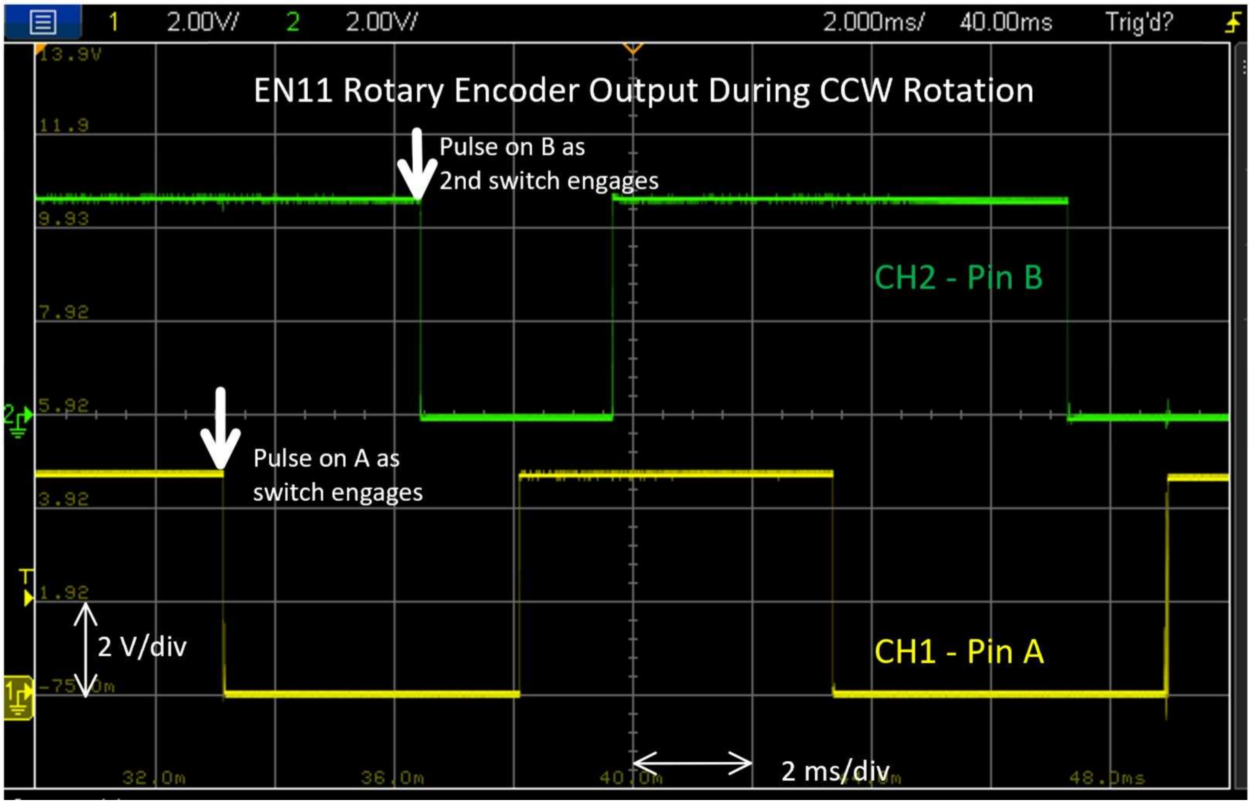
Challenge 2 (Rotary Encoder):

Question #1: In Tabular Format, Discuss the differences and relative merits of the EN11 Rotary Encoder and the Potentiometer

Potentiometer	Rotary Encoder
The output is in the form of change in DC Voltage Level.	The output is in the form of 2 pulses with a phase difference.
Absolute	Incremental
Has a dead band at zero.	Has no dead band.
Requires ADC for Decoding.	Requires Quadrature Decoder or similar code for Decoding.
Effect of noise on decoding process is comparitively low.	Effect of noise is significant and filter must be used.
Decoder code level required is low.	Decoder code level required is high as delay needs to be adjusted.
270 ° angle of rotation with 256 unique positions.	360 ° angle of rotation with 20 unique positions.
Doesn't require homing operation for sensing initial position.	Requires homing operation for sensing initial position.

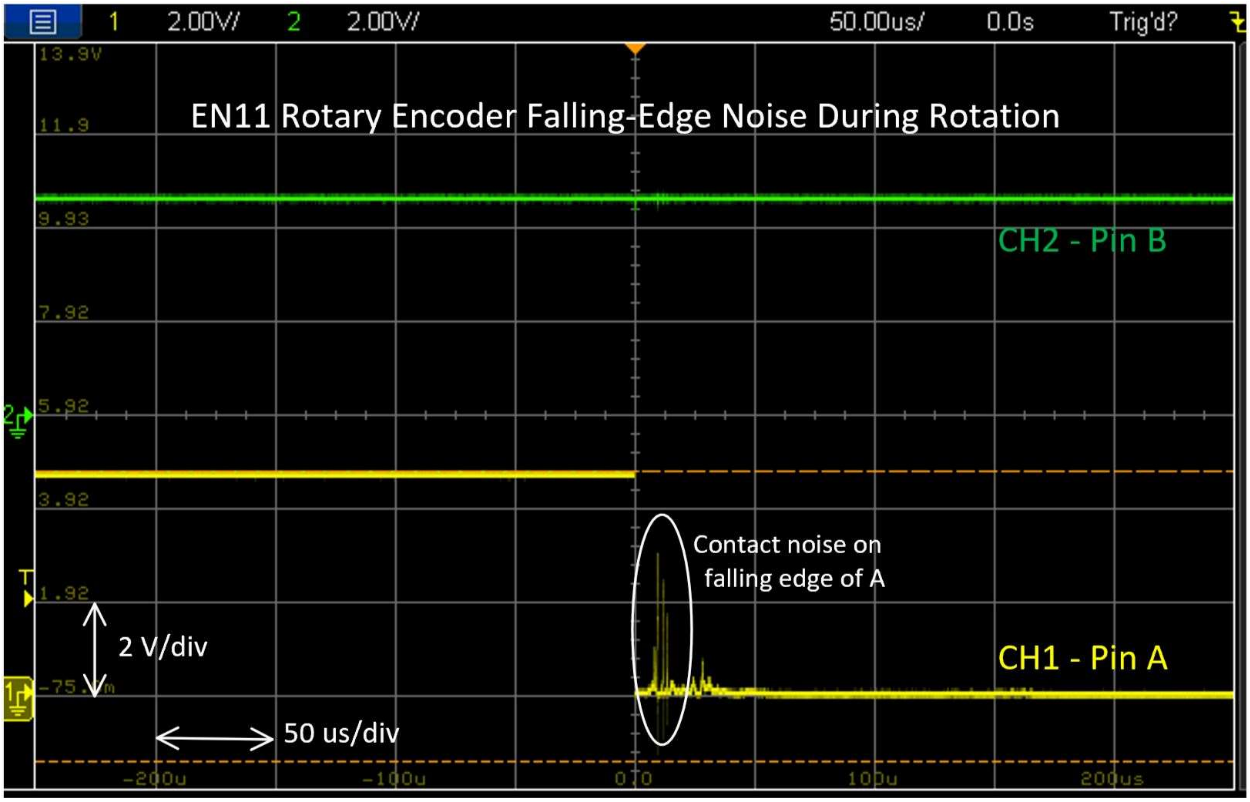
.

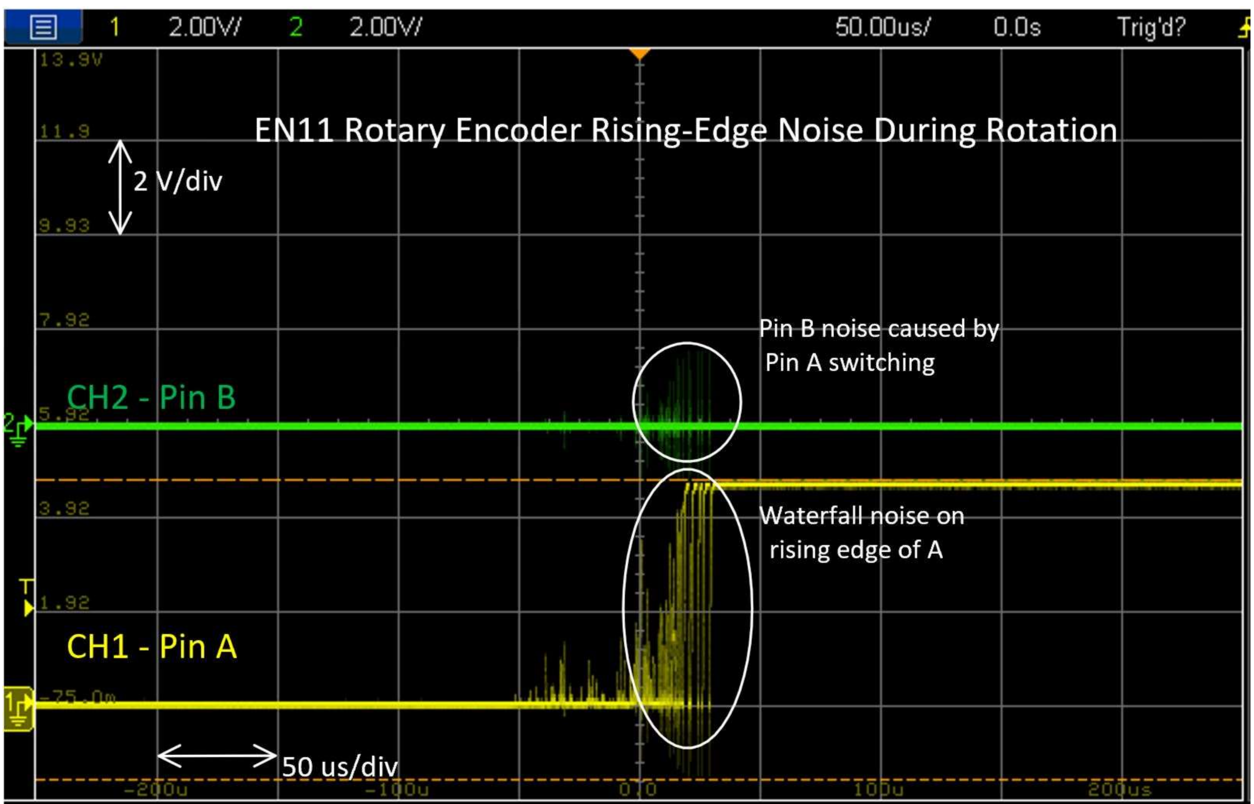
Scope Image 1: On the EN11 rotary encoder, show both channels A (= ch 1 on the scope) and B (=ch 2 on the scope), as the encoder is rotated by hand. Trigger on a rising edge of ch A. Select a slow sweep rate (time/div) and rotation speed so that you see approximately 2- 4 clicks of the EN11 in the image.



<u>**Scope Images 2-4:**</u> Use a much faster sweep speed to show switch bounce. Do this a number of times while you watch the trace. <u>Find 3 examples of noise</u> on either channel and include them in your report. At least one example should be so-called "waterfall" noise where multiple edges appear close to each other. Note that this noise will

make accurate position decoding difficult.





<u>Question #2:</u> Typically, how long does the noise occur before or continue after the trigger? How does this compare to the time between clicks when you rotate the switch slowly by hand?

Answer: The noise typically occurs for typically 100us. The pulse width when we rotate by hand varies from 10ms(high speed) to 40ms(low speed). The noise is too less compared to the time between clicks or pulse width. But, the noise duration is more than enough for the code to count multiple times if the noise is not taken into consideration while programming.

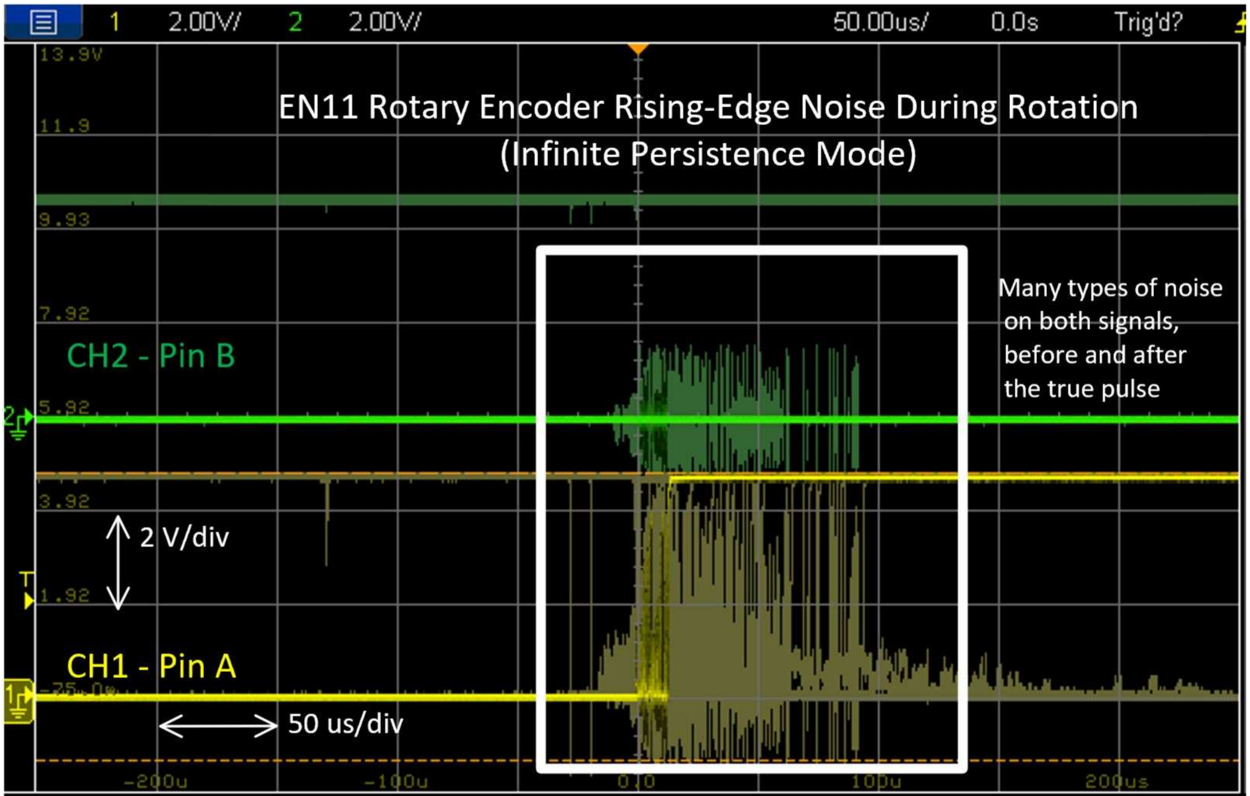
Question #3: Test the EN11 switch and complete the following tables (direction assumes you have oriented the EN11 rotary switch with the wires down and the knob up)

For c terminal connected to Ground

ch A	ch B	direction of motion
rising edge	high	Clockwise
rising edge	low	Anticlockwise
falling edge	high	Anticlockwise
falling edge	low	Clockwise

ch B	ch A	direction of motion
rising edge	high	Anticlockwise
rising edge	low	Clockwise
falling edge	high	Clockwise
falling edge	low	Anticlockwise

Scope Image 5: Set the scope for "infinite persistence mode" so that old traces remain displayed on the screen as new ones are taken. Rotate the switch to collect and display many cases at once. This gives you a much better idea of the variability of the switch.



Question #4: Based on the scope image taken above, how could you reject waterfall noise- i.e. count just once per click while ignoring the multiple edges that can occur close together due to switch bounce?

Answer:

Waterfall noise can be rejected by 3 methods:

1) Using a Passive low pass filter to remove the high frequency noise.
2) Using a delay in the code after it senses the values once and increments or decrements the count. For this the delay between two iterations must be more than the pulse width. But, using this method may skip some counts due to high delay so it is not effective.

3) Comparing the current value of the input with previous value and ignoring if the value is same. For this the delay between two iterations must be less than the pulse width.

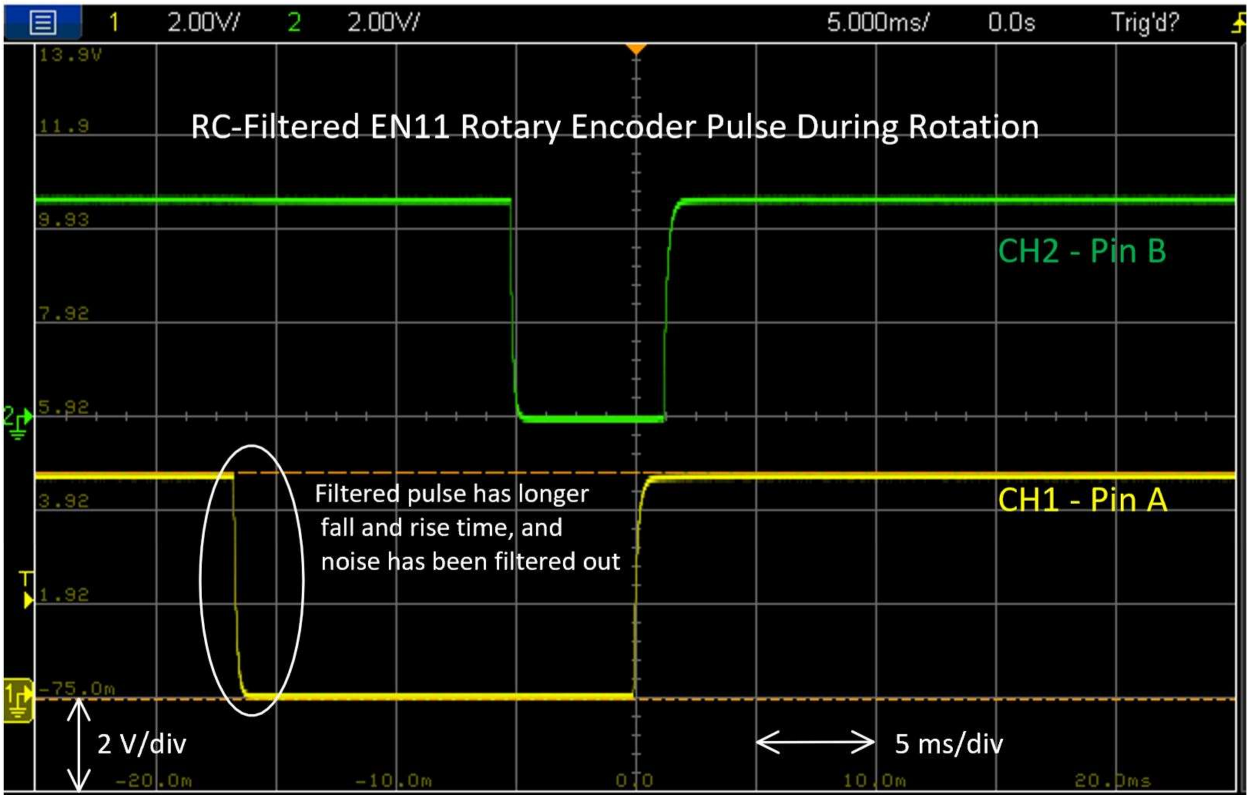
Verification: We will test your design by carefully rotating the EN11 encoder and counting 20 clicks in each direction, while watching both the hardware and software counts. Take a video of this and submit it with your report. (Make sure the video file size is not too large and uses a common format like mp4).

Demo 2: Show the instructing team/ upload a video of your system counting pulses using the Quadrature Decoder to filter the noise.

Demo 3: Show the instructing team/ upload a video of your system counting pulses using software to filter the noise.

Demo 4: Show the instructing team/ upload a video of your system counting pulses using a passive analog filter (you need to figure out if you need a low/high/band pass filter) to filter the noise.

(You can probably setup your circuit to display 2/3 of the required counts at the same time; You must also display rotation in the [0-360) range)



Question #5: Discuss in a few paragraphs, the differences and relative merits of all 3 methods.

Quadrature decoder – This method is the easiest to implement. The Quadrature decoder removes the waterfall noise as a default property. The output is accurate. This method cannot be implemented without PSOC or an actual quadrature decoder so the hardware cost for this method is more.

Software decoding – It is a challenging task to remove waterfall noise in this method. The delay between 2 iterations of loop must be adjusted properly. The previous input must be checked to avoid same pulse getting counted twice. This can be implemented without any block in PSOC but accuracy is less.

Analog Filter – A passive analog filter can be used before the output of the rotary encoder is given to the pins. The filter required is a low pass filter that removes the high frequency waterfall noise and switching noise that occurs due to friction. If the software decoding is used along with a passive low pass filter of correct value, we get accurate results.

<u>**** Extra Credit #2 ** (2 points):**</u> The microcontroller can use the EN11 to measure not only position, but also speed. Use software or hardware to <u>measure the period</u> of each pulse produced by *either* ch A or ch B of the EN11 rotary switch. Convert this count to rpm and display it on the LCD. Demonstrate that the count is correct by turning the switch slowly with a cordless drill (TA has one/ Ask Kai or the Launch Point in the ITLL) while the LCD shows the speed in rpm. You must be within 20% of the perceived RPM to receive credit.

######################### End of Lab ########################