Hardware/Software co-design Reloaded
Home Work 01
06.10.2014
Team member: Shehzad Ahmed and Arka Mallick
==========

Exercise 1: Phase quadrature encoder

According to the given description of the phase encoder, the disk can be graphical represented by the Fig. 1. This disk contains two black bars and 2 transparent bars with separation of 90 degree between each.

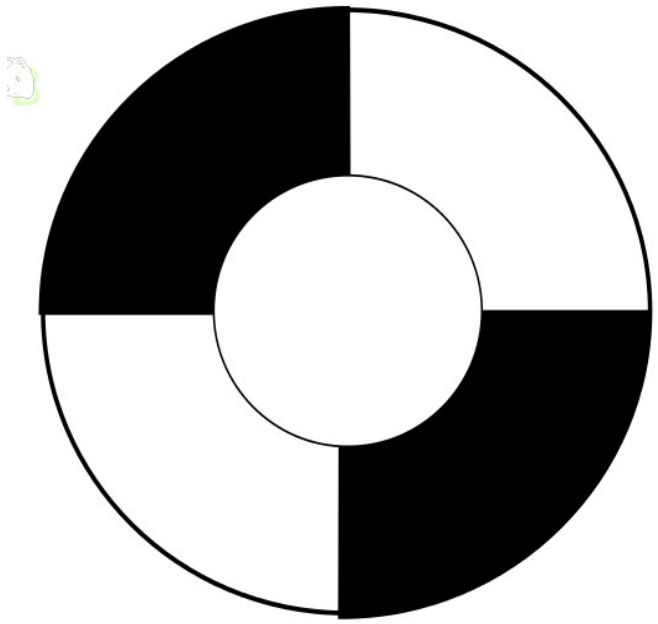
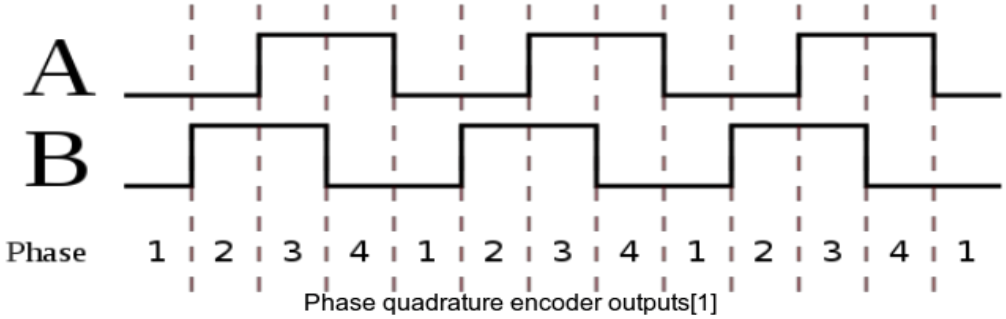


Fig.1 Phase encoder disk

- Determination of maximum resolution of phase quadrature encoder:

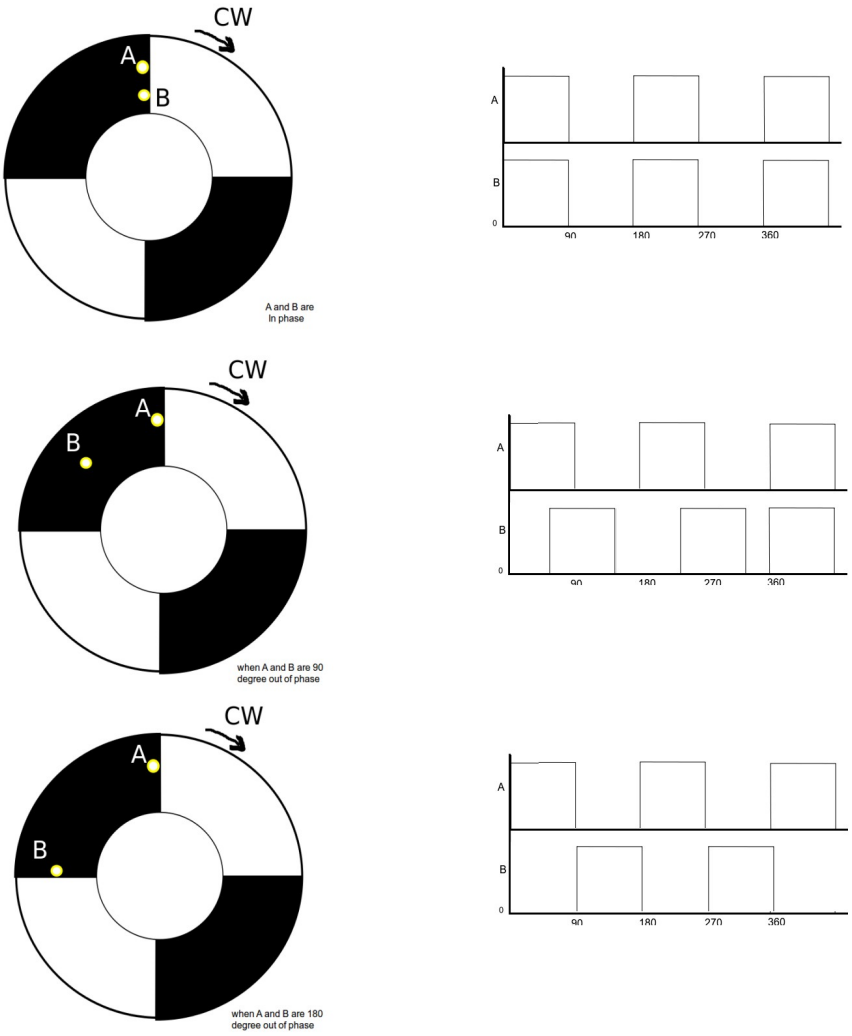
  According to the definition of the phase quadrature encoders, they employ two outputs, say A and B, in order to determine the direction of motion and measure the position of the motors. These two outputs are 90 degree out of phase because of this reason these encoders are named as quadrature encoders.[1]



Phase quadrature encoder outputs[1]

In order to achieve the desirable response of the phase encoder, lets consider the placement of two outputs A and B in three configuration.

case 1:



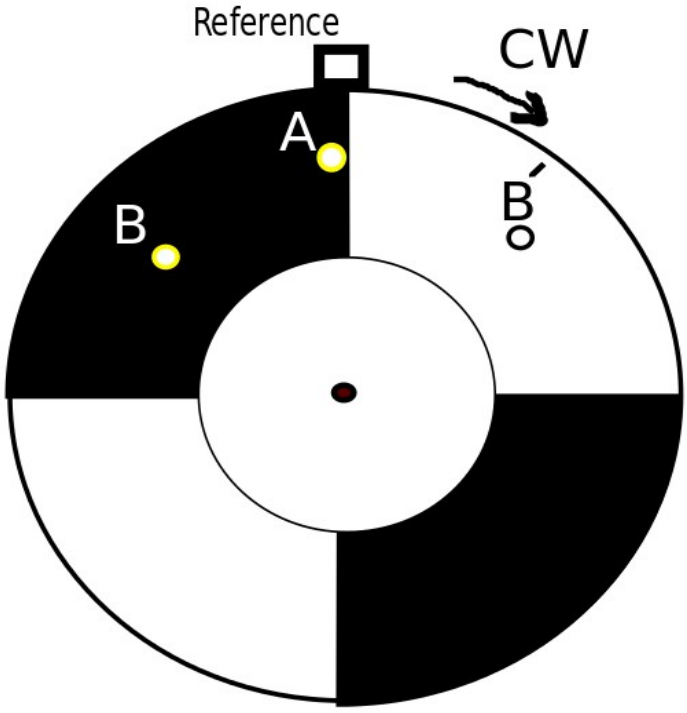
The above diagram shows the effect of placing two channels A and B in three different locations.

- When A and B are placed along the same diameter of the disk, the output channels are in phase and the only possible outputs are 00 and 11.
- Similarly, the third case, there are only two outputs (01 and 10) possible when A and B are place 90 degree apart from each other.
- Both the above case does not satisfy the definition of quadrature phase encoder. But if,

we place A and B 45 degree apart from each other, then we get 4 possible outputs i.e. 00, 10, 11 and 01 (CCW direction) or 10,11,01,00 (CW). This configuration gives two outputs 90 degree out of phase w.r.t each other.

Therefore according to the above argument, the maximum resolution that can b achieved with quadrature encoder is 90 degree.

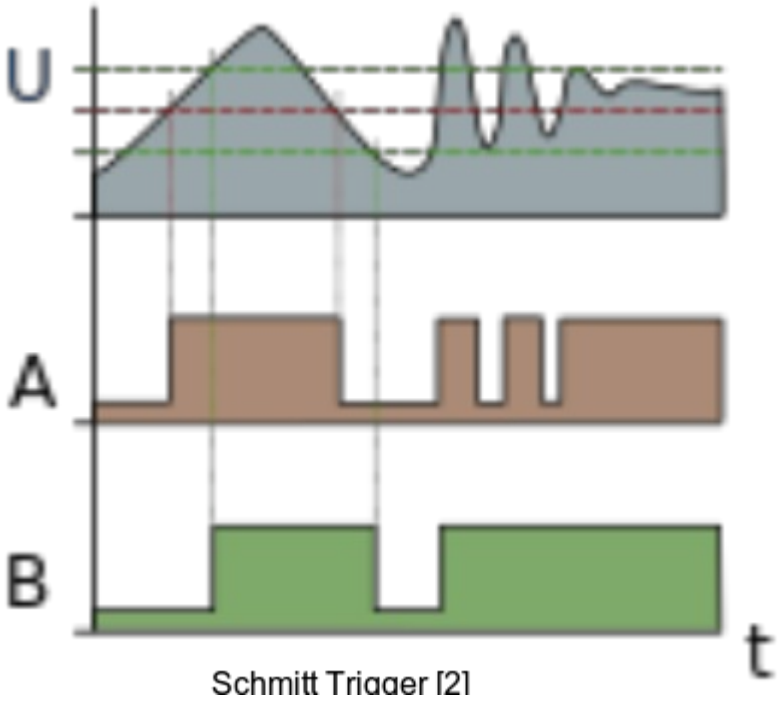
In order to achieve the maximum resolution of 90 degrees, the second barrier can be placed at two position as shown in following figure. The positions of the second barrier can be +/- 45 w.r.t the reference position.



Exercise 2: Rectification using Schmitt-trigger

According to the definition of schmitt-trigger[2],

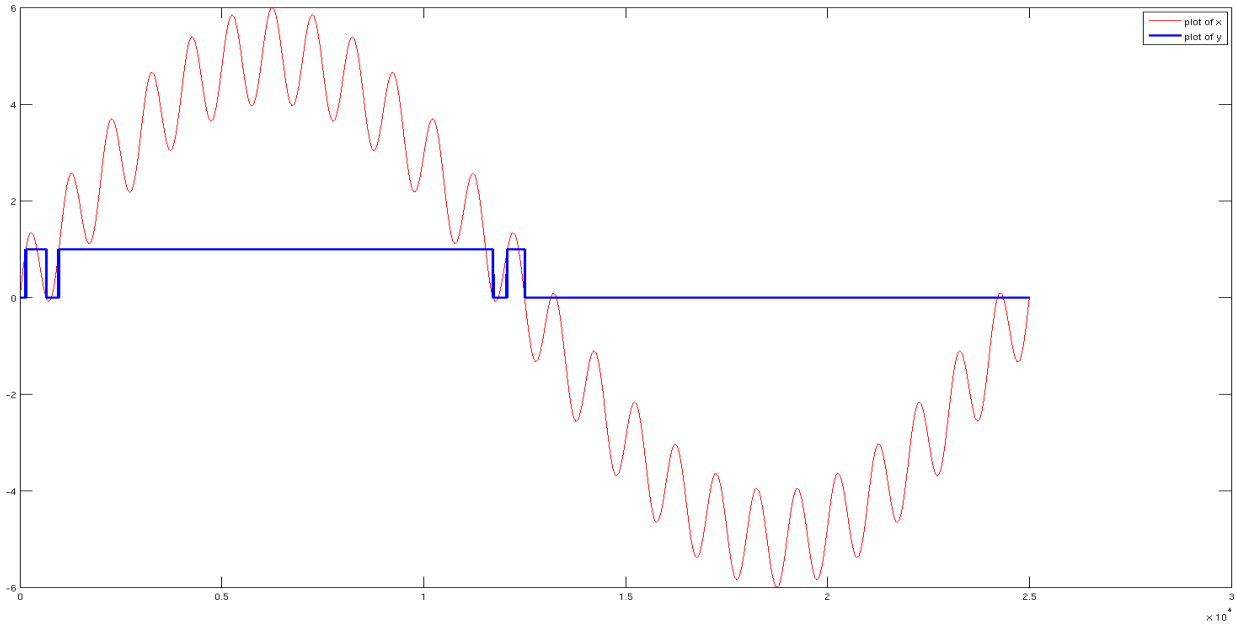
- It is a comparator which converts an analog signal to a digital output signal. The output of this comparator retain its value until the input crosses lower and upper chosen thresholds. When the input is below a certain lower threshold, the output is low and if the input goes above the upper bound of the threshold, the output becomes one.
- The dual threshold action is called hysteresis and it also implies that the Schmitt trigger possesses memory like a latch or flip-flop.



Schmitt Trigger [2]

A nice implementation of schmitt-trigger action is found in [3]. The sample codes has been attached along with this file. The result of applying given sum of sinusoidal signal to is shown in the following image.i.e:

s(t) = 1*sin(2*pi*f1*t) + 5*sin(2*pi*f2*t) , where f1=1KHz and f2=40Hz


plot of x
plot of y
6
4
2
0
-2
-4
-6
0
0.5
1
1.5
2
2.5
3
× 10^4

Exercise 3: Nugget Finder initial design

Here are the following modules that I think are needed.

1. Motion module for Exploration:
    - Navigation and vision closed loop controller
      searching for nuggets → collects if the nuggets is in view → stops after exploring whole area but when ?
    - Need of path planning strategy
    - Need of odometer sensor and camera mounted on board

2. Wall finder and avoidance module:
    - Avoid going into the walls
    - constraint: collection of nuggets near the wall
    - Needed ultrasonic or bumper sensor

3. Safety and security module:
    - Use of Watchdog timer to avoid rover defect
    - Current sensing of the motors

4. Coordinator Module
    - In order to make coordination between the modules if needed.

References:

[1]: http://en.wikipedia.org/wiki/Rotary_encoder
[2]: http://en.wikipedia.org/wiki/Schmitt_trigger
[3]: http://www.mathworks.de/matlabcentral/fileexchange/44419-schmitt-trigger