# ME 310 : Microprocessors and Automatic Control Lab

Prasanna Gandhi
Gandhi@iitb.ac.in

***Experiment 4: Interfacing encoders using IC HCTL 2017 and response of motor***

**Objectives:**

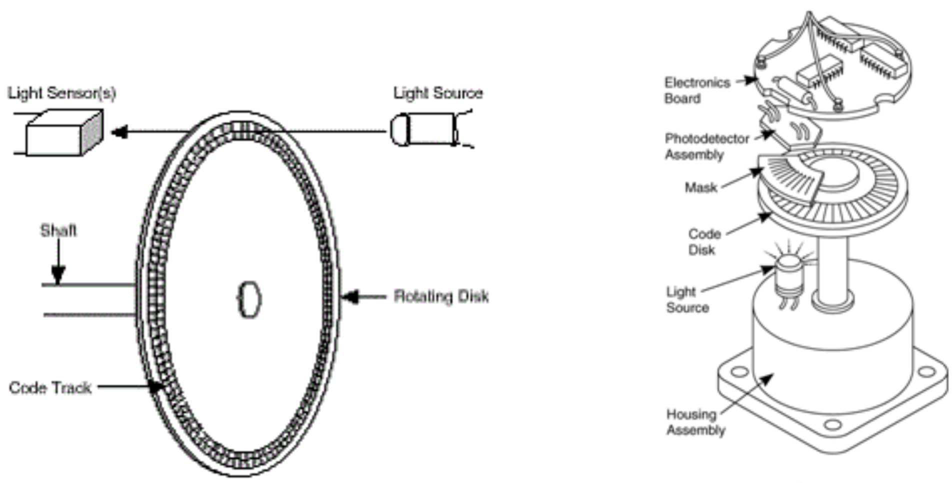
1. Know fundamentals of Encoders and Decoders (IC HCTL 2017)
2. Understand how to interface encoders with XEP 100 and read MSB, LSB and combine them to get encoder count
3. Read the position of the motor and learn how to process this data to get the angular velocity of the motor
4. By varying the duty cycle / voltage, see the effect on the angular velocity and then by using this plot determine the friction and damping of the system

DC motors are at heart of several automatic control/ mechatronic applications and form basic actuator elements (example robots). So far we have learnt how to drive them using microprocessor and our goal is to control them in closed loop finally. For feedback we need to learn to measure the position of motors. Two of the most common devices used for the same are encoders and potentiometers. In this experiment we will be using encoders to read the angular position of the motor. Then we will be processing this data to get the angular velocity of the Motor.

**Background knowledge required:**

1. Thorough understanding of the Encoders and read datasheet of decoder HCTL 2017
2. Thorough knowledge of programming in Matlab and generating PWM
3. Use of easyscope
4. Fundamentals of dynamics

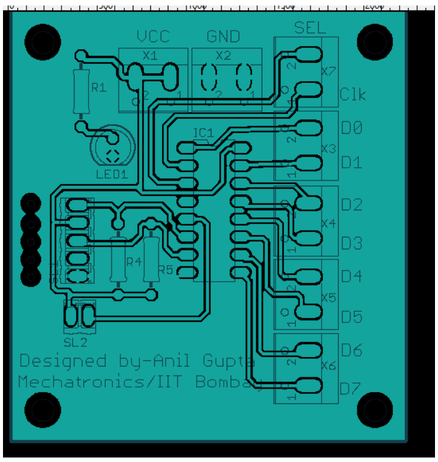
**Encoders**

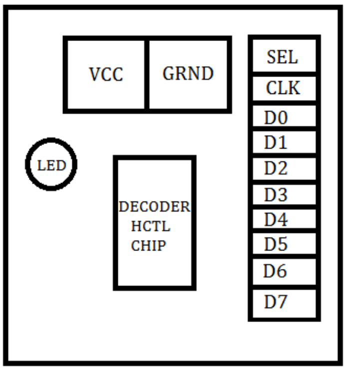



Fundamentals of Encoder

An optical encoder is an electromechanical device which has an electrical output in digital form. Fundamental of its working is shown in the figure. It has a shaft mechanically coupled to an input driver which rotates a disc rigidly fixed to it. A succession of opaque and clear segments are marked on the surface of the disc. Light from infrared emitting diodes reaches the infrared receivers through the transparent slits of the rotating disc. The received analog signal is then processed electronically converted into digital form of one pulse per passed slit. These pulses are needed to be counted and then the position can be known.
Q: How can we know the direction of rotation?? There are two channels A and B for this. These are electronically 90deg out of phase thus we are able to see which is leading signal and know the direction.

**Decoders:**




A decoder is a circuit that converts signals A and B of encoders into the corresponding counts. The encoder generates the number of pulses which are given to pins of a decoder chip (here HCTL 2017) and count is generated as a digital data on the output pins of this chip. This data can be read in microcontroller through digital i/o port B (for example).

On Input side, the HCTL chips needs a clock input which is generated by programming one of the PWM channels. Moreover a pin RST should be maintained high (except for resetting count). There are two additional pins SEL and OE which need digital input depending on the output function mentioned below. The output of HCTL chip is a 16 bit number, but for convenience of pin connections and saving number of i/o pins, data is given out 1 byte at a time on 8 pins of HCTL. A couple of control bits (SEL, OE) are used in a specified sequence (given in Figure 10 of datasheet) to select Higher eight bits of the data (MSByte) data or Low 8 bits of the data (LSB).

Two variables U_DATA and L_DATA are used in the program below to read MSB and LSB respectively. Q: Think of algorithm needed to combine MSB and LSB in one variable?? How will you do that?

You have been given program below for the operations mentioned above.

**Program:**

```
unsigned int i=1,j=1;
long count=0;
int En_Read=0x0000;
int L_DATA=0x00,U_DATA=0x00;
void PWM_init(void);
void PWM_for_Motor(void);

void main(void) {
  /* put your own code here */
   PWM_init();
   PWM_for_Motor();
   DDRA=0xFF;
   DDRB=0x00;
   for(;;) {

        PORTA=0x00;          //Enable=0 ,Select Pin=0     {SEL=PA2,OE=PA3}
        U_DATA=PORTB;
        PORTA=0x04;          //Enable=0 ,Select Pin=1     {SEL=PA2,OE=PA3}
        L_DATA=PORTB;
        En_Read=(U_DATA << 8)| L_DATA;  // Combining lower  byte and Upper
                                          Byte into  a 16-bit binary number
        PORTA=0x08;          //Inhibit Logic reset  Enable Pin=1, Select Pin = X

        for (i=1;i<334;i++);  // Delay of approx. 1ms to get time reference for data
        count++;              // Reference value update
   }
```

**Functions used to generate PWM for motor and decoder clock:**

```
void PWM_for_Motor() {
  PWMCLK_PCLK5=0;
    PWME_PWME5=1;
    PWMPOL_PPOL5=1;
    PWMPRCLK=0x01;
    PWMDTY5=0x80;
    PWMPER5=0xFF;
 }
```

Function used to generate Clock for decoder circuit:

```
void PWM_init() {
    PWMCLK_PCLK0=0;
    PWME_PWME0=1;
    PWMPOL_PPOL0=1;
    PWMPRCLK=0x00;
    PWMDTY0=0x04;
    PWMPER0=0x08;
}
```

***Before proceeding forward, please answer the following Questions:***

a) Which channel of PWM is used to run the motor? Which channel is used to generate clock for decoder HCTL?

b) What is PWM frequency and duty cycle for the clock?

c) Which port is used to read MSB and LSB?

d) Pin SEL and OE should be connected to which pin of which Port?

***Show your answers to TA before proceeding further***.

**Things to do?**

1. Read the codes and the handout thoroughly; try to question each and every line of the code to gain an insight of the encoder system. Connect the appropriate PWM pin of XEP to the clock pin on the decoder circuit. Give power (VCC, **only from microprocessor**) and ground connections to the circuit. Connect the SEL pin and OE pin to the port as identified above. Connect pins D0 to D7 to the XEP microprocessor PORT B. Without connecting the motor wires run the program. Now manually turn the motor and see the change in En_Read, U_DATA and L_DATA. Turn the output shaft of gearbox for 6 turns and note the change in the 3 variables above. Can you explain the variation you observe in these variables? Can you guess what is resolution of the encoder? Motor is connected to gear box with reduction ratio of 3.71. Call your TA and discuss.

Additional inputs:

**Steps for alternative way of Visualizing Data on Terminal:**

1> After running the program, open Component drop down menu
2> Click on Visualization
3> Right Click on Add new
4> In Instruments select Chart
5> Now to visualize data of En_Read with respect to time, just drag and drop the variable
6> Change mode to periodical with a refresh rate of 1 ms to get the plot

Explore this way to visualize motor encoder data as it gets generated.

2. Now use easyscope to see signals given to and coming on the pins of HCTL chip. Sketch signals on OE, SEL, Ch A, D0 pins and justify what you observe when motor is rotated! Show the signals to TA.

3. Now connect the motor circuit. After compiling the program and running the motor, write the command "cf log_command.txt" in the command window. It is assumed that the file log_command.txt (written below) is placed in project directory (same place where *.mcp of the project sits).
   Wait for some time till the log file is written. Now open the log file and try to analyze the results.

   **Log file to be separately created:**

Create a text file named 'log_command.txt' in the operating directory. Type the following in 'log_command.txt' :

```
DEFINE loop = 0
FOR loop = 1..10000,1
fprintf(log2.txt,"%d %d\n", count, En_Read)
ENDFOR
```

In log_command.txt, number 10000 indicates the number of times the logging has to be done, counter and En_Read are variable names (which are to be logged). Each reading is printed 16 times in this file for some reason. Use matlab to import and remove multiple readings and to observe the data carefully. Counter indicates the number of cycles passed before the next data is printed. Develop program to process data to finally get smoothly varying motor angle in rad as a function of time. You may need to use calibration in part 1. Show the data to TA and verify.