Principles of Measurement &Instrumentation I Laboratory
PHYS417

Experiment 3- Data Acquisition

Author Batuhan Dereli

1851690

Experiment Date:23.10.2019

Submission Date:30.10.2019

Contents

0.1 Objectives

In this experiment ,we try to get to read and write data from microcontroller and different type laboratory equipment special example for this laboratory oscilloscope is the example .Another case , last week experiment works specially read the analog values on the microcontroller Arduino based .Actually this action is not the enough the for the experimental physics because sometimes we need to read many sensor and save the all operation for the scientific reasons. In this laboratory work procedure we obtain the get data from oscilloscope and at the same time get this data with computer based using port communication and we read the data value at the same time we plot the graph also PY-Visa based on Ni-visa programs provide the recognize the which device connected to our computer usb port because of the this programs we easily connected and with python relative the some codes helped to us get data from devices .

Another point on the laboratory , thanks to microcontroller we learned that how to get temperature or other analog values to convert our digital world , this provide to us at the real time learn our sensors response times and their characteristic specifies because of this experiment we can get any data from microcontroller and also we can read this data from PORT also this port values at the real time we can save on the txt file and save this values .

Last point on the this lab experience at the real time we can drawn the graphs this is the very important for the scientist because our world is the mathematical representation and we are physicist with electronic devices not enough the count or measure our world environment we also study on this data and we need the data values visualize so that this laboratory experiment give to us make modeling sensor or any port values on the graph .

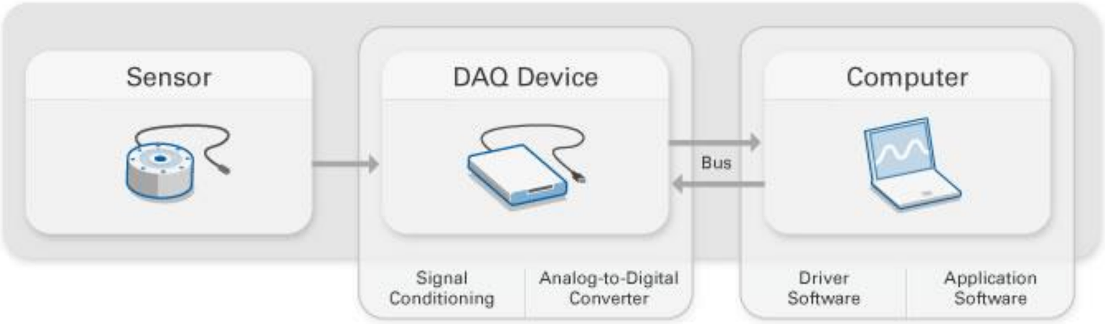


Figure 1 Data Acquisition -Represent to Data Read with Computer

0.2 Introduction

First of all, we have to know that Data Acquisition (DAQ), this is very important for the understanding our world . In the world many data flow our around and somehow scientist or special purposes people need to understand this data basically after that this data will use for the special purposes . In the past, people want to sense this environment with the human sense depends but as close past times electronic devices more accrued and make possible the so sensitive measurements . If we discuss about the electronic history ,first Data Acquisition (DAQ) with the human eyes and the save the results with hand also obtain results with hand mathematical modellings but nowadays almost last 5 decades with computers this method is the old school technical and make possible at the real time respond so that make possible the make so fast and clear measurements .

Today measurement method is the make independent variables make depends each other's .For example ,we have a problem relative the temperature and learn what is the room temperature and we want to make monitoring this problem .However, there are many example for the solving this situation .Moreover , many ways make possible the understand room temperature , our purpose of the measure value of the temperature .First we need sensor also we can call that transducer ,both names are provide to us make possible depends on the transducer quality and special qualities , read and measure in range wanted values . This values are as laboratory and electronic way , we learn that with temperature create EMF or depends the resistance , make some voltages , this voltage values are so important , because if we do not have any computer or microcontroller we can read this voltages and using ITS-90 tables for the standard temperature references tables are make possible convert voltage to temperature with hand operation mathematical results will give to us temperature values .

Unlike that, in the laboratory we have microcontroller and this microcontroller have enough range to convert analog to digital signals , for example Arduino have 10 bits ADC range this value is the nice resolution for our measurement system . With microcontroller data taken after that with RS232 communication serial port USB connection data transfer the at the real processing time and this data after get on the serial monitor high speed bound rate make possible changes observe on the monitor fast and without delay . Moreover , data on the PORT ,so we can use libraries and python is the our laboratory based main computer software so some libraries provide to us connect our computer USB PORT so data reading converting decode from ASCII to understandable to our visible characters .Also basic mathematical graph actions make possible the all data reads at the real time plot and observe how to environment give respond with the transducers .

0.3 Equipment

- Computer
- Oscilloscope
- Thermistor
- 100 K resistor
- Potential Meter
- Arduino UNO (I used Arduino Mega)
- USB 2.0 CABLE TYPE A/B
- LED

0.3-01 SOFTWARE

Python based libraries

- Py visa
- Py Serial
- Matplotlib
- Numpy
- Drow now
- Time
- Pylab

Arduino based

- Arduino IDLE

0.4 Procedure

In this part, we will try to run some programs to learn how to plot graphs using Matplotlib because we need to draw sensor values so first we understand to how to draw values on the python software and Matplotlib is the very useful tool for the plot graph . You can see how to draw on the python which code I use on the code part . First of all , ready codes gives on the lab manual but I prefer the colors and grid also legend parts with my own knowledge.

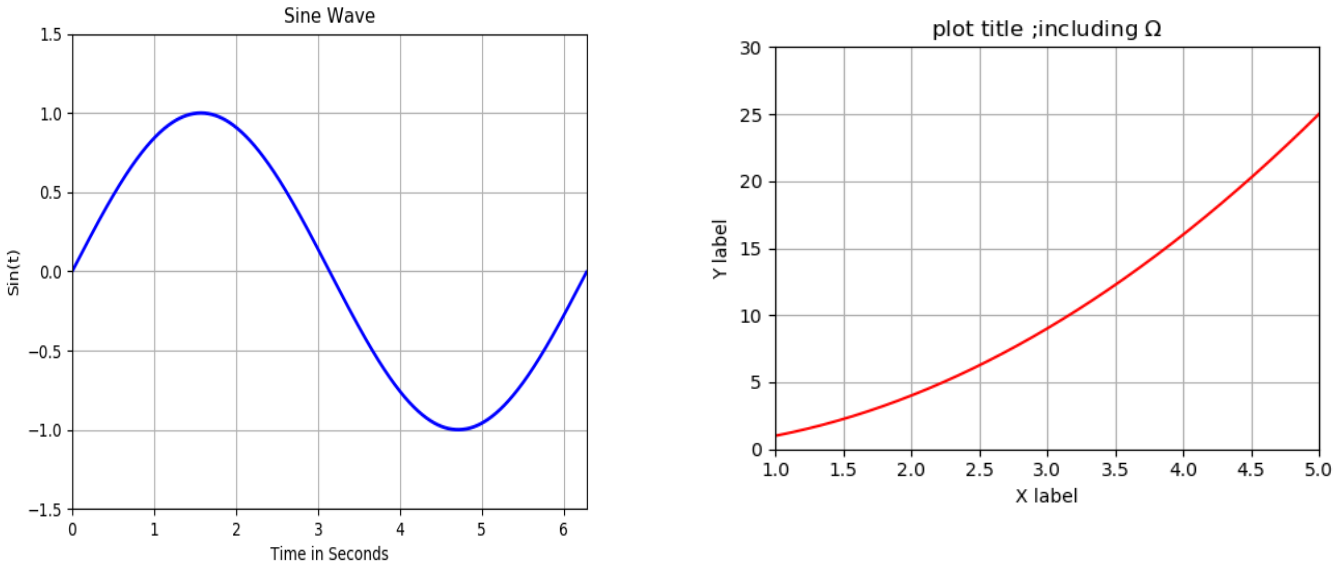


Figure 2 Sin Wave and normal line plot Draw on the Python

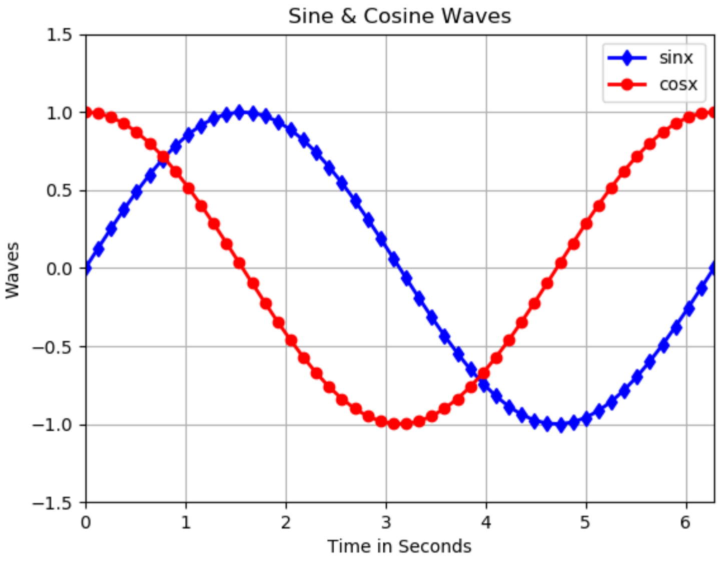
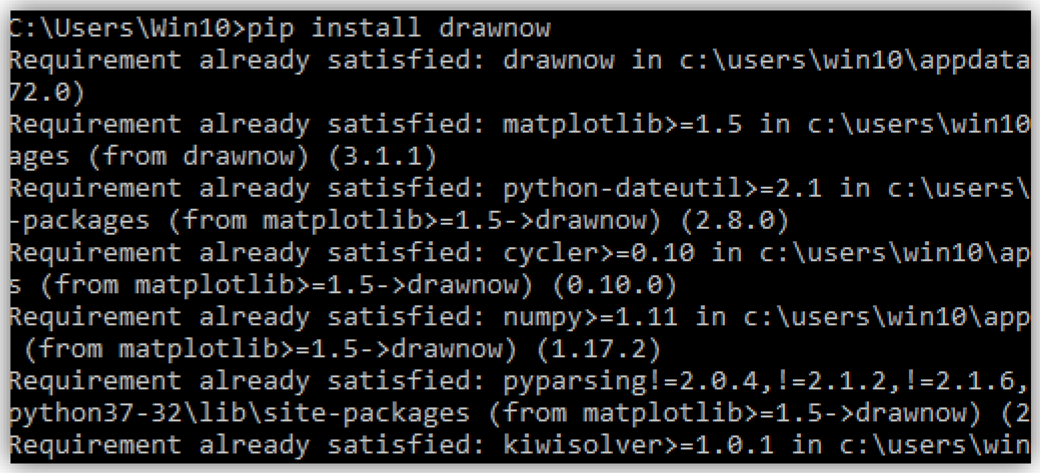


Figure 3 sin-cos graph

With using console CMD on the windows based computer we can write pip install drawnow comment as you can see CMD screen live view at the figure 4 .



```
C:\Users\Win10>pip install drawnow
Requirement already satisfied: drawnow in c:\users\win10\appdata
72.0)
Requirement already satisfied: matplotlib>=1.5 in c:\users\win10
ages (from drawnow) (3.1.1)
Requirement already satisfied: python-dateutil>=2.1 in c:\users\
-packages (from matplotlib>=1.5->drawnow) (2.8.0)
Requirement already satisfied: cycler>=0.10 in c:\users\win10\ap
s (from matplotlib>=1.5->drawnow) (0.10.0)
Requirement already satisfied: numpy>=1.11 in c:\users\win10\app
 (from matplotlib>=1.5->drawnow) (1.17.2)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,
python37-32\lib\site-packages (from matplotlib>=1.5->drawnow) (2
Requirement already satisfied: kiwisolver>=1.0.1 in c:\users\win
```

Figure 4   pip install drawnow procedure successfully completed

Also all install NI VISA library as well and download the software website , we can ready to connect to our instrument device in this laboratory we have aim to connect to oscilloscope .

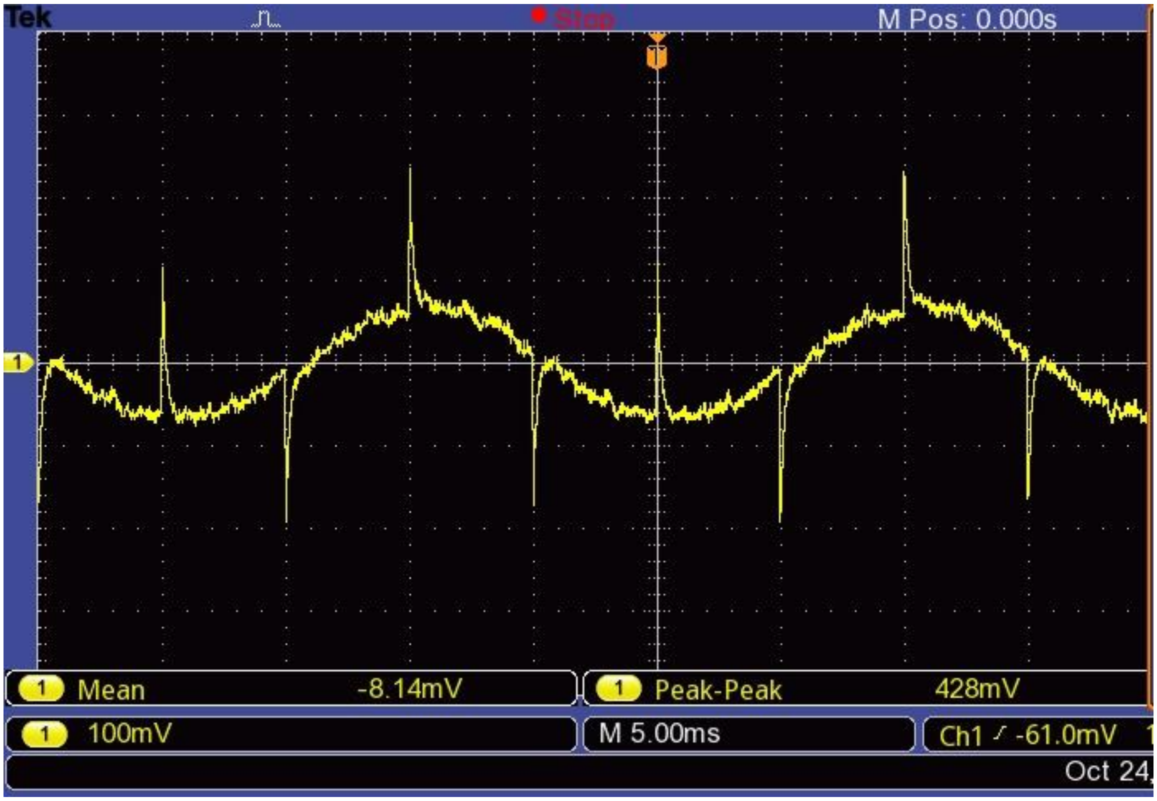


Figure 5 Oscilloscope one channel

We get this value using with python codes . Another modification with this codes we had opportunity the get with matplotlib draw with own graph.

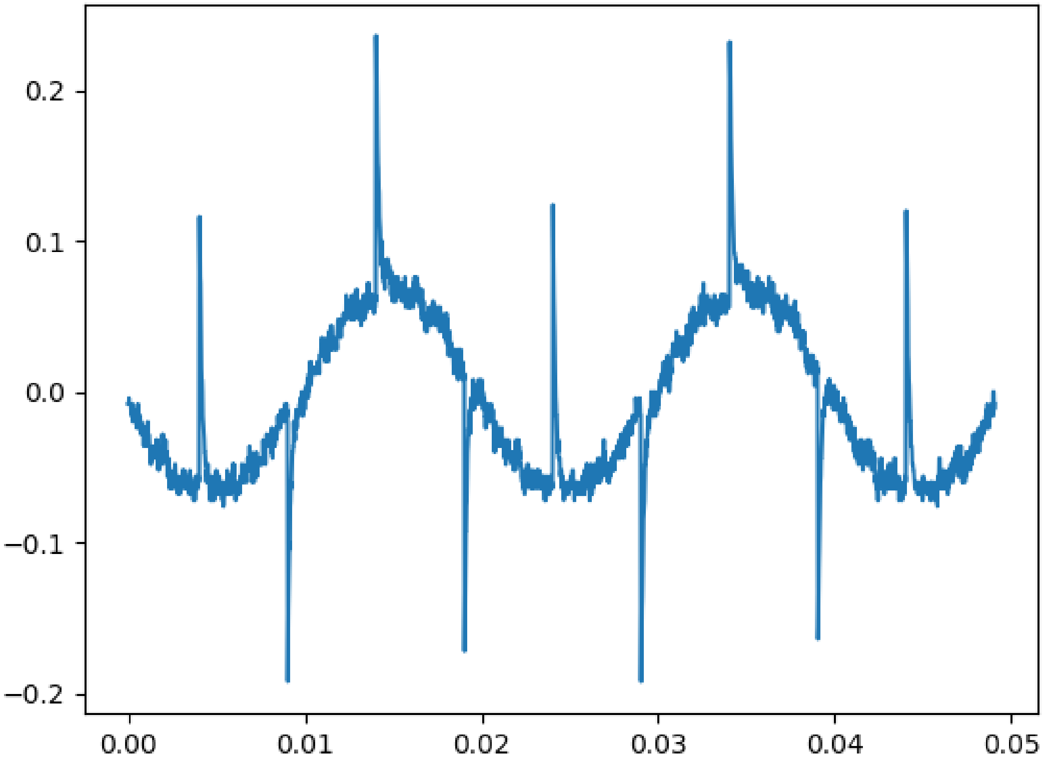
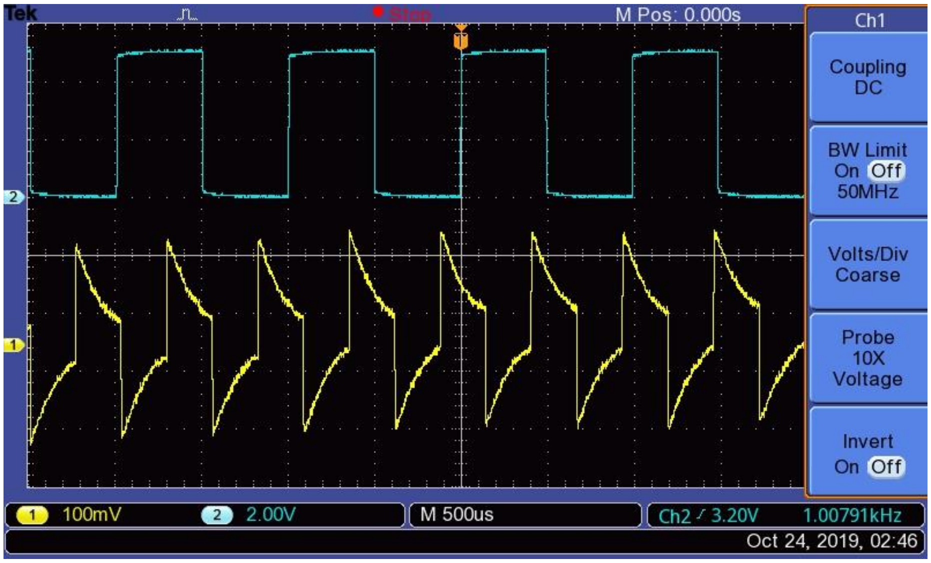


Figure 6 oscilloscope channel 1 same data with graph matplotlib & PYVISA



Coupling DC

BW Limit On [✓ Off] 50MHz

Volts/Div Coarse

Probe 10X Voltage

Invert On [✓ Off]

1 100mV 2 2.00V M 500us Ch2 ⁄ 3.20V 1.00791kHz

Oct 24, 2019, 02:46

Figure 7 oscilloscope 2 channel data taken

Next step we have to create thermistor control circuit because we need analog values for the thermistor sensor and at the same time make plot real time data plot operation .

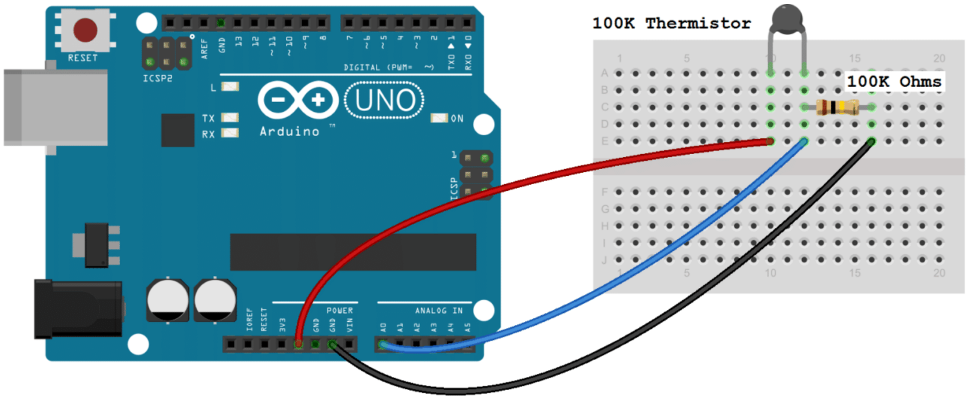


Figure 8Thermistor Arduino Connection

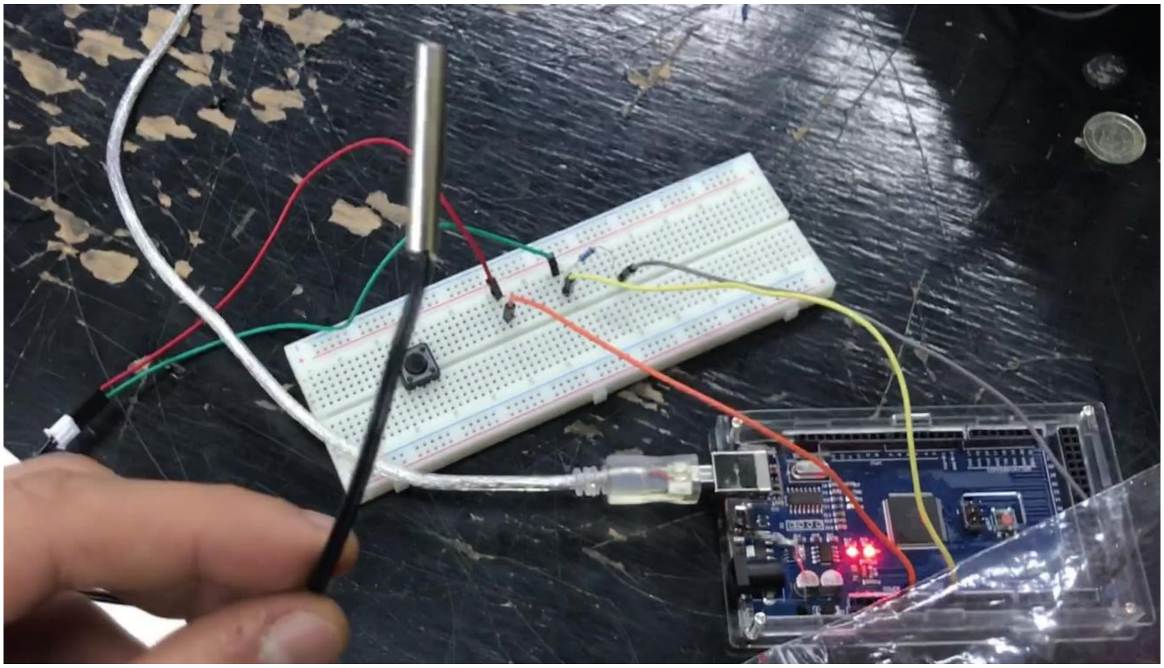


Figure 9 Real during the experiment live circuit connection

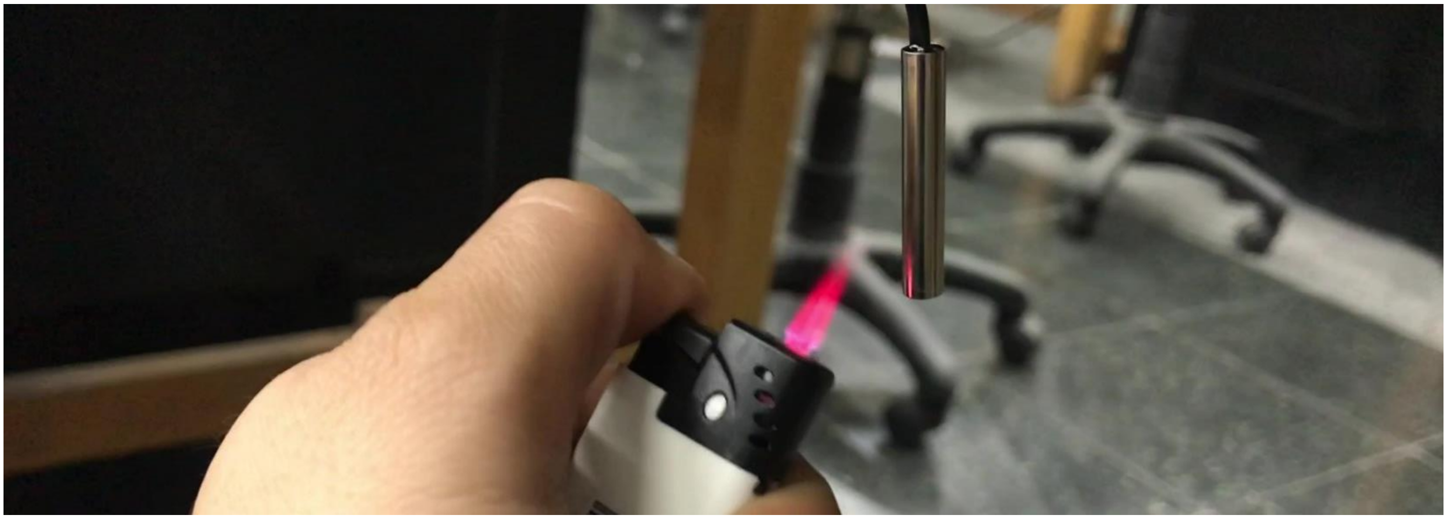


Figure 10 Thermistor temperature is increasing with fire

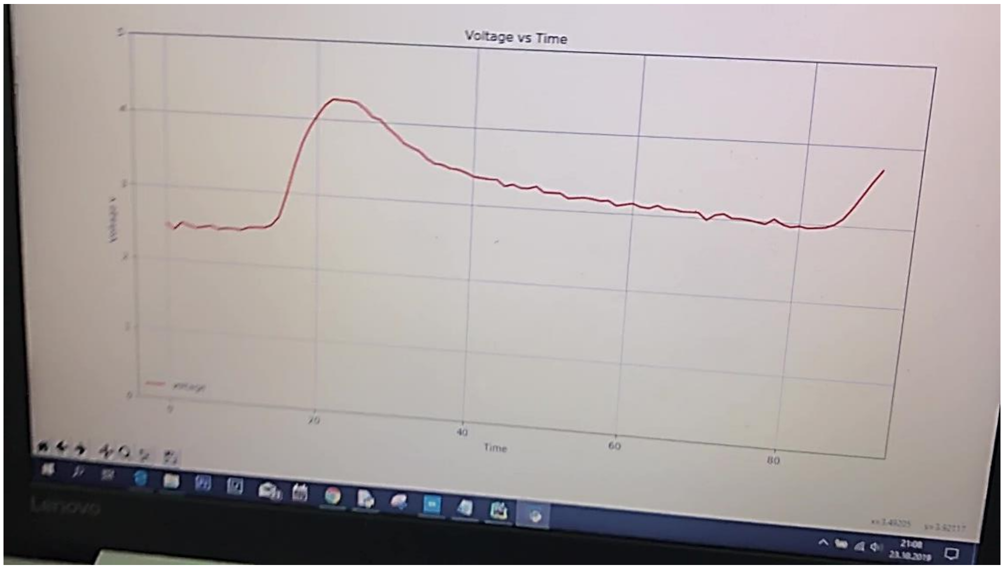


Figure 11 During the experiment at the same time data respond with positive increasing with fire and voltage observe versus time relation

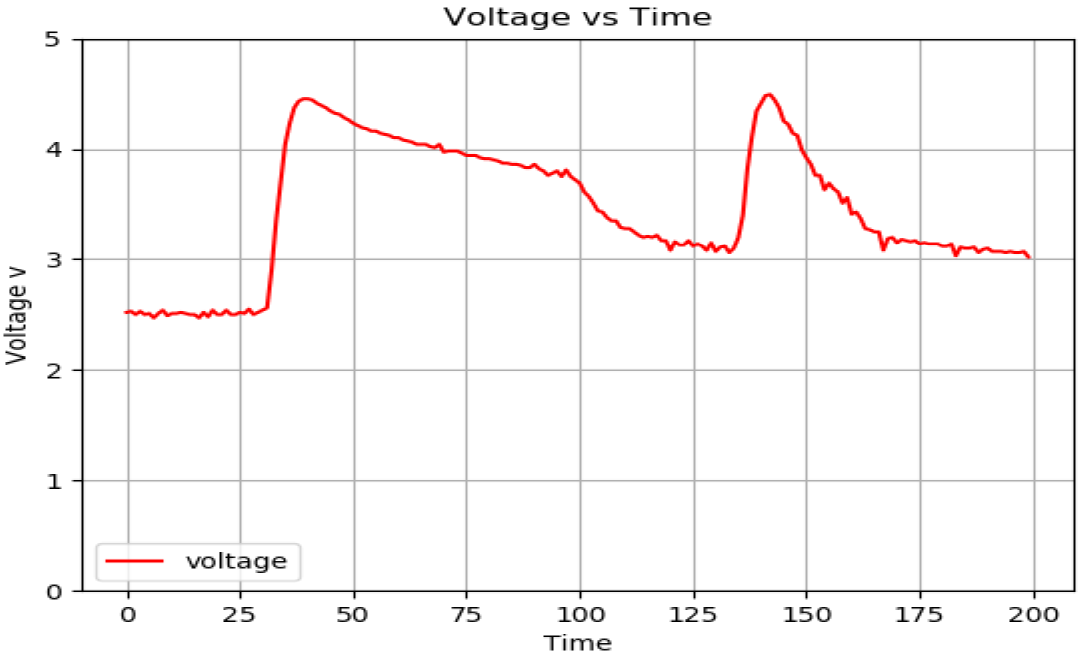


Figure 12 data plot with python software real time

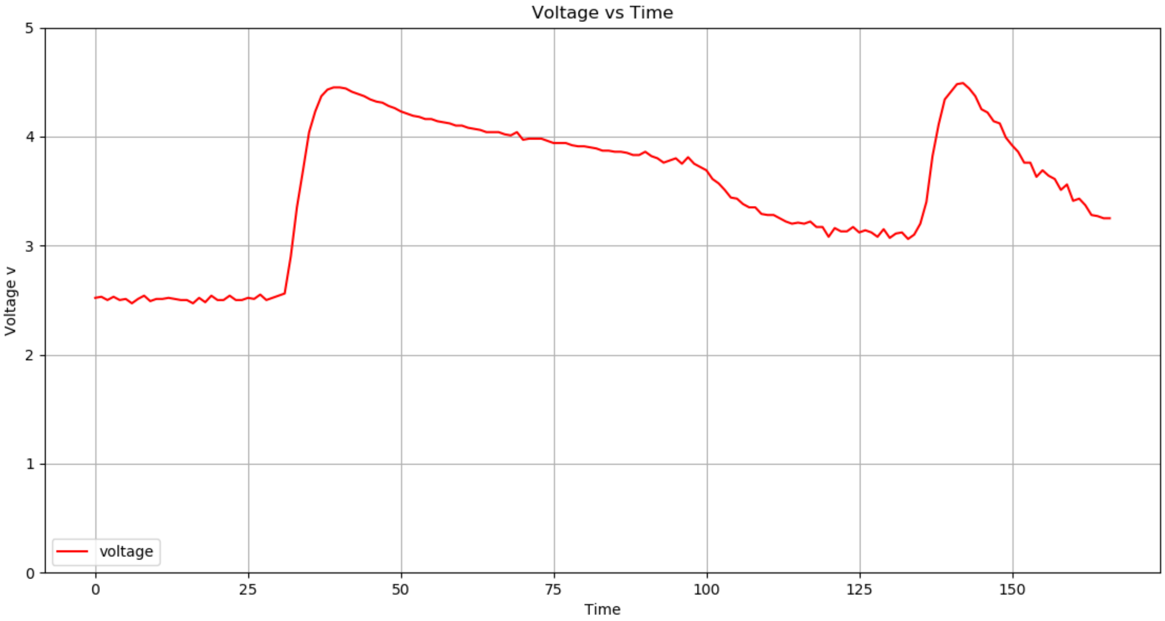


Figure 13 Thermistor Data Read real time and observe responses with increase temperature voltage also increasing proportional

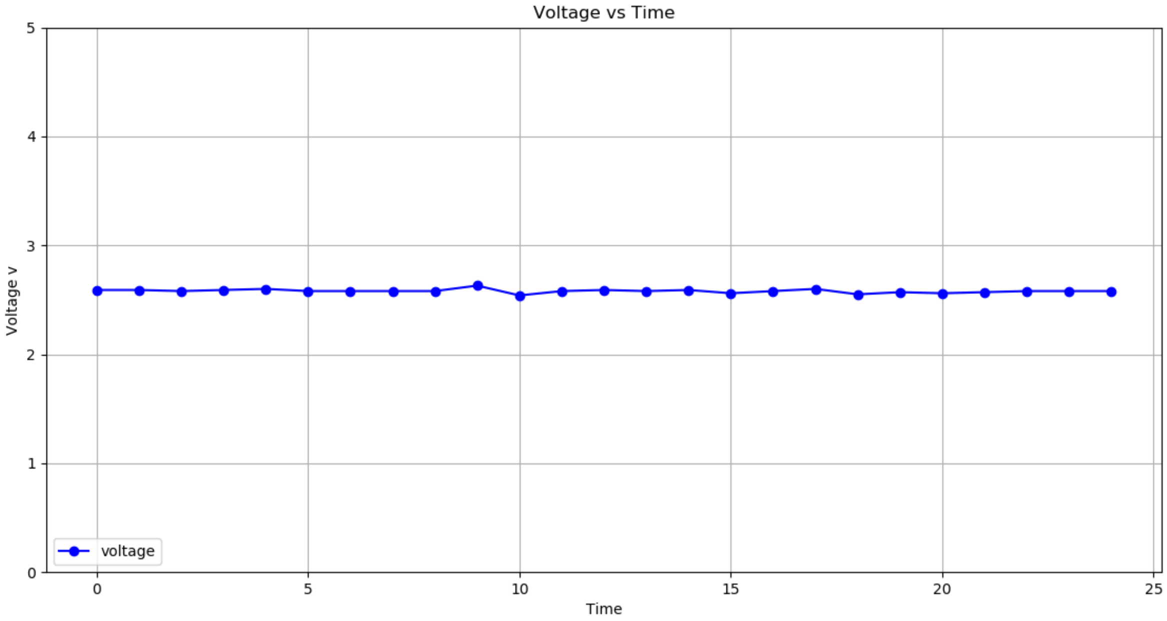
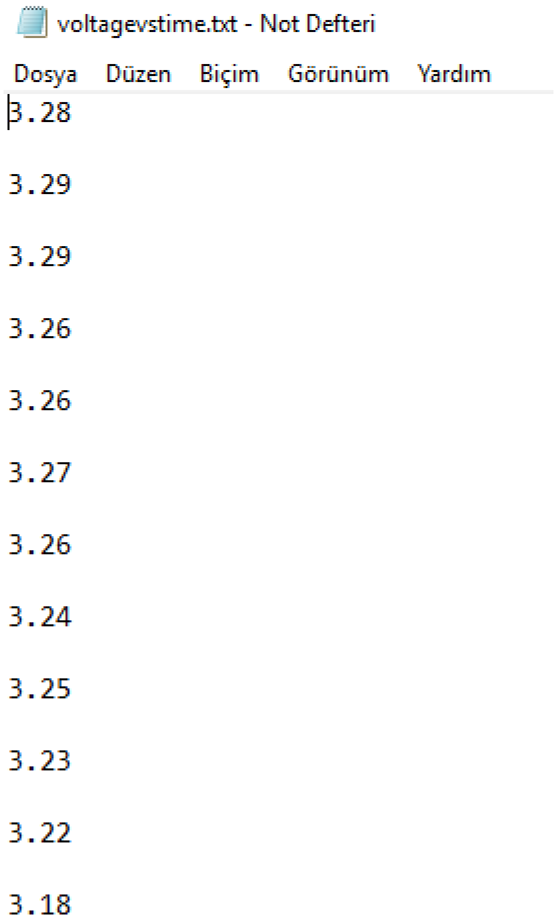
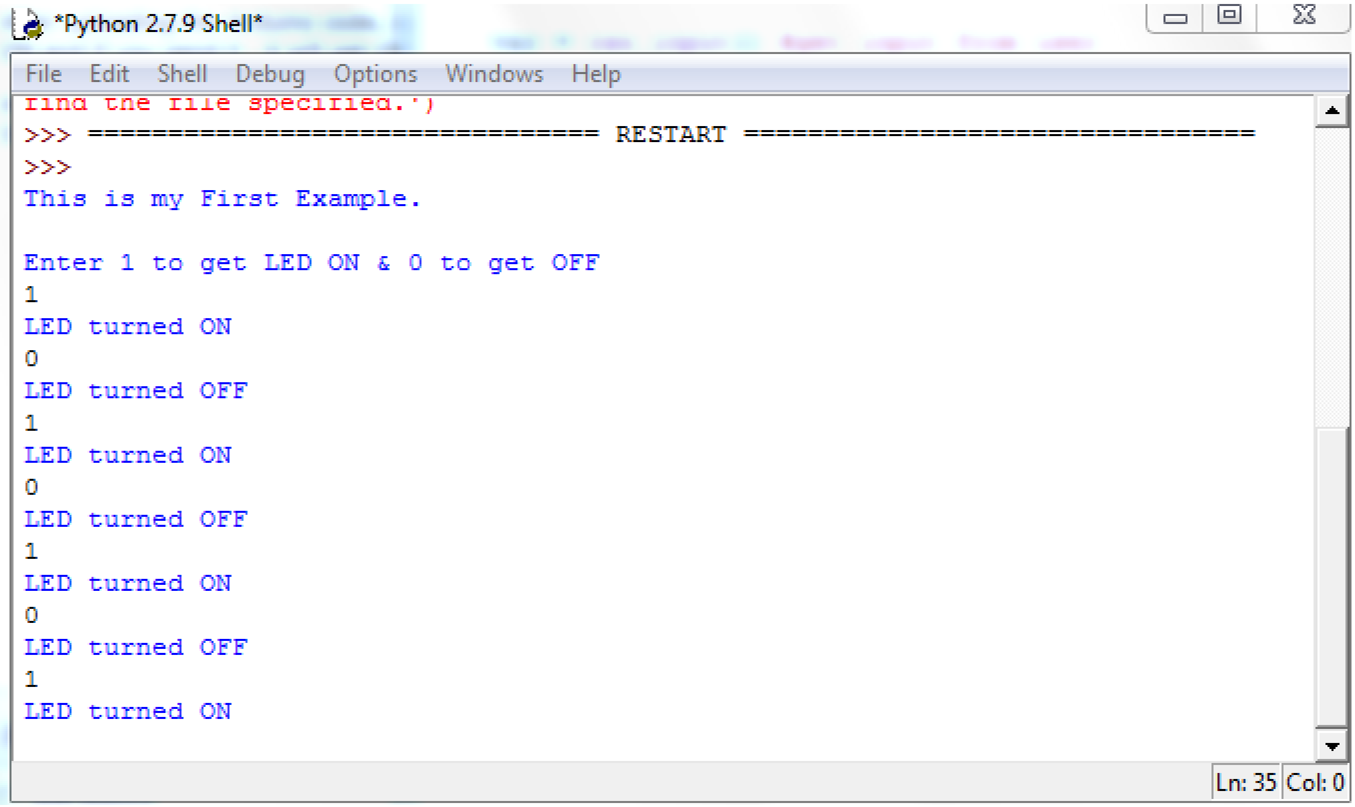


Figure 14 without any interrupt or external any touch ,this graph observe at room temperature in ELECTRONIC LABORATORY



```
3.28

3.29

3.29

3.26

3.26

3.27

3.26

3.24

3.25

3.23

3.22

3.18
```

Figure 15 Voltage Values Saved on the txt file



```
find the file specified.')
>>> ================================ RESTART ================================
>>> 
This is my First Example.

Enter 1 to get LED ON & 0 to get OFF
1
LED turned ON
0
LED turned OFF
1
LED turned ON
0
LED turned OFF
1
LED turned ON
0
LED turned OFF
1
LED turned ON
```

Figure 16 LED Blink python IDE

0.5Calculation

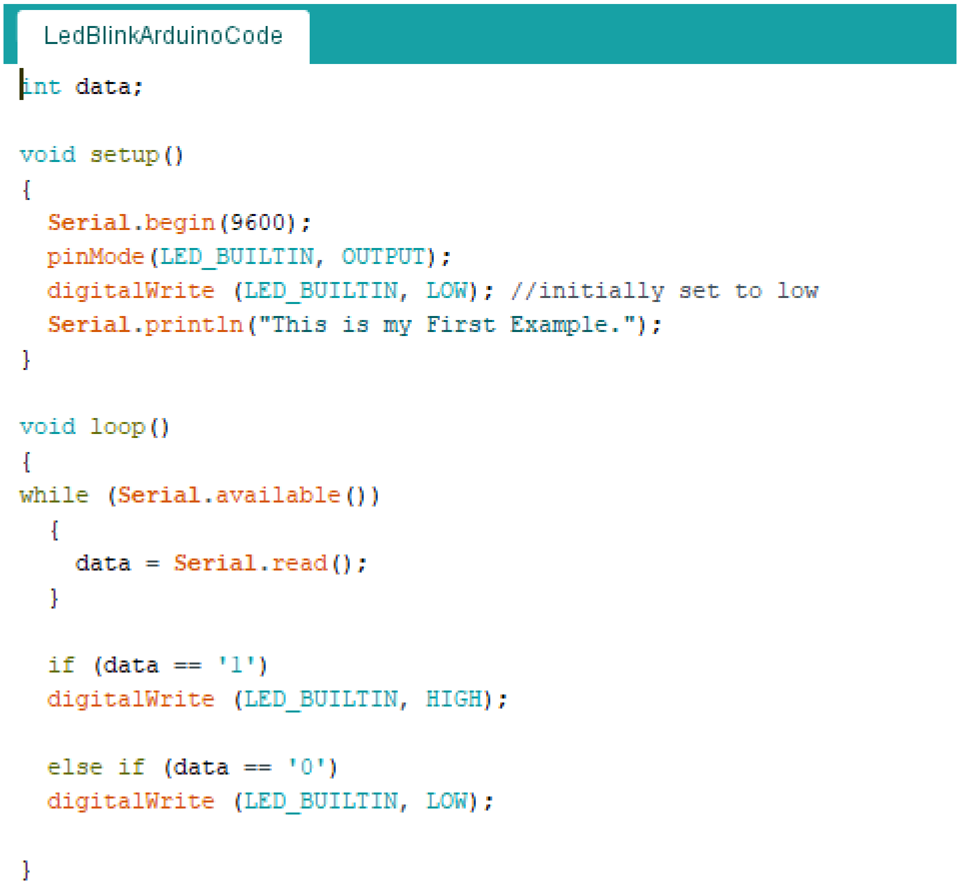
There is no calculation.

0.6Discuss&comments

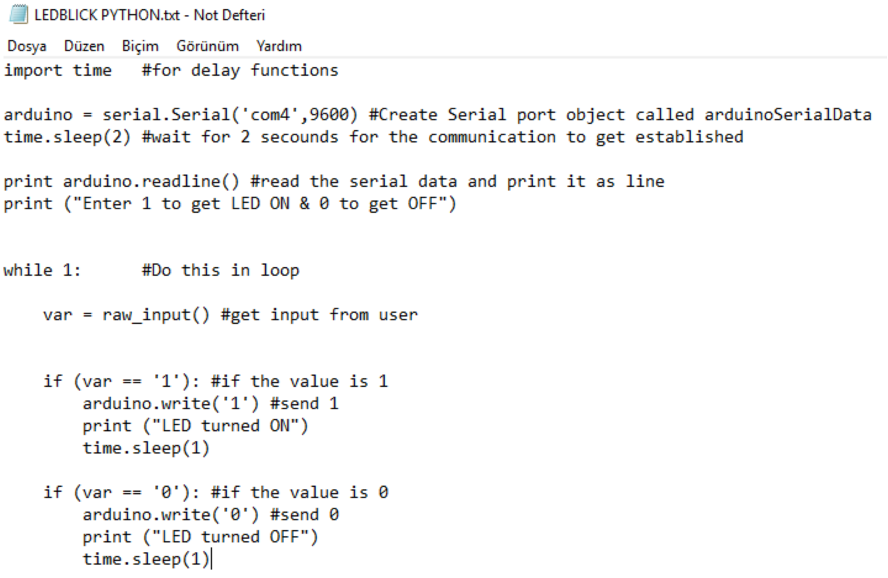
In this experiment , I was know that using Excel and MATLAB software data save methods .After this laboratory , I can take any microcontroller data and use this information provide to me understand how to communicate devices with computer and is there what kind problems and solving methods on the serial communication . It was seems to hard to me but after all data and live plot operations I recognize that basic strings are converting ascii characters and this are decode in the software after we are taking all information logical .Also with any device defined with NI-visa in the lab we can get their port information and also take information so basic idea was saving data in the computer and this lab experiment will help my midterm project for characterizing sensor I need data take so I will use this lab information.

0.7CODES

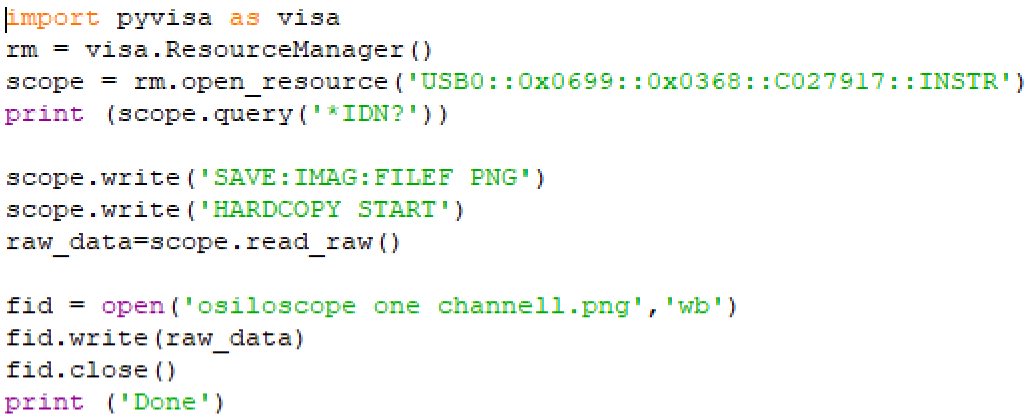
Figure 17 PYTHON ARDUINO SERIAL CONTROL CODE



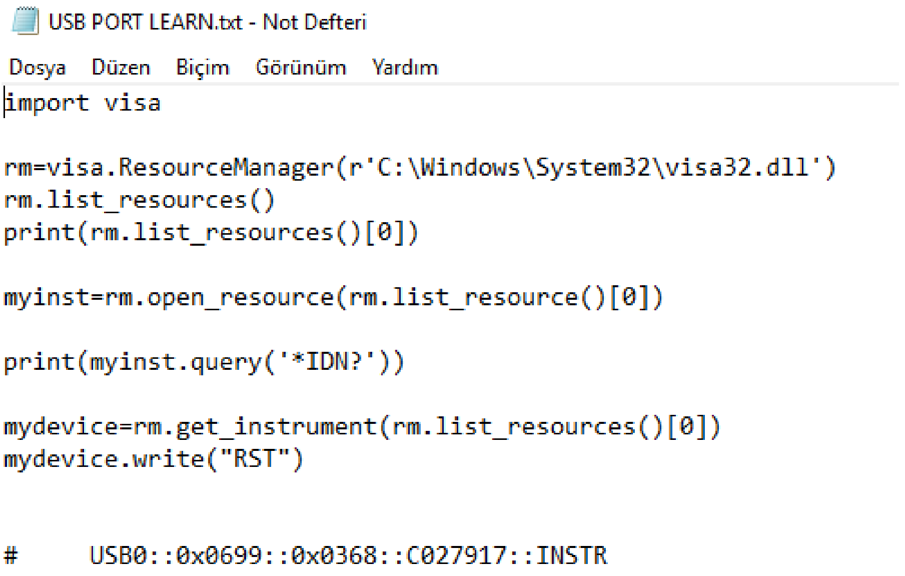
```
int data;

void setup()
{
  Serial.begin(9600);
  pinMode(LED_BUILTIN, OUTPUT);
  digitalWrite (LED_BUILTIN, LOW); //initially set to low
  Serial.println("This is my First Example.");
}

void loop()
{
while (Serial.available())
  {
    data = Serial.read();
  }

  if (data == '1')
  digitalWrite (LED_BUILTIN, HIGH);

  else if (data == '0')
  digitalWrite (LED_BUILTIN, LOW);

}
```


LEDBLICK PYTHON.txt - Not Defteri
Dosya Düzen Biçim Görünüm Yardım

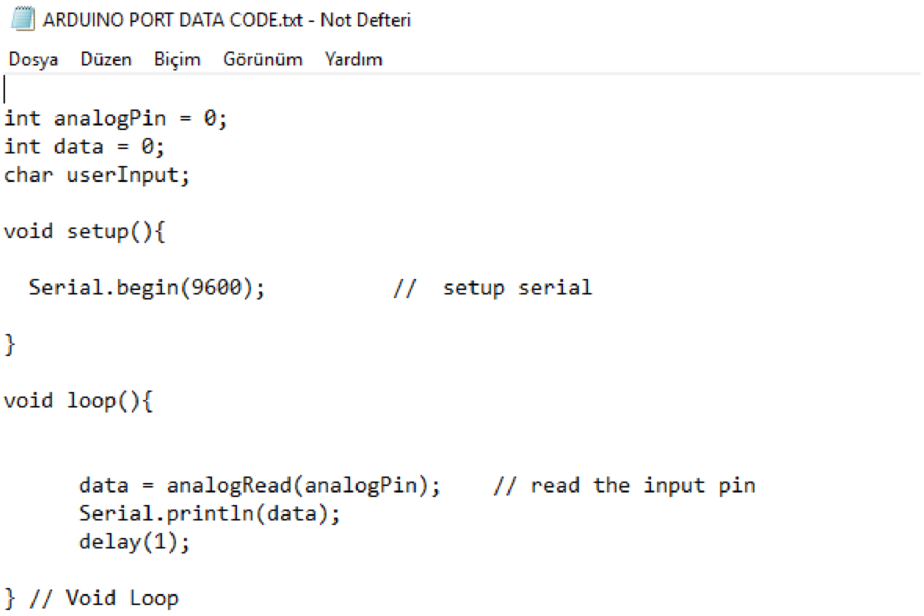
```
import time   #for delay functions

arduino = serial.Serial('com4',9600) #Create Serial port object called arduinoSerialData
time.sleep(2) #wait for 2 secounds for the communication to get established

print arduino.readline() #read the serial data and print it as line
print ("Enter 1 to get LED ON & 0 to get OFF")


while 1:      #Do this in loop

    var = raw_input() #get input from user


    if (var == '1'): #if the value is 1
        arduino.write('1') #send 1
        print ("LED turned ON")
        time.sleep(1)

    if (var == '0'): #if the value is 0
        arduino.write('0') #send 0
        print ("LED turned OFF")
        time.sleep(1)
```



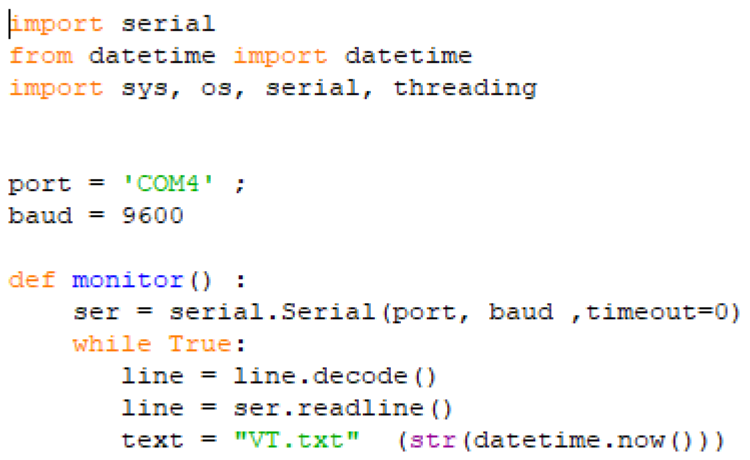
```
import pyvisa as visa
rm = visa.ResourceManager()
scope = rm.open_resource('USB0::0x0699::0x0368::C027917::INSTR')
print (scope.query('*IDN?'))

scope.write('SAVE:IMAG:FILEF PNG')
scope.write('HARDCOPY START')
raw_data=scope.read_raw()

fid = open('osiloscope one channell.png','wb')
fid.write(raw_data)
fid.close()
print ('Done')
```



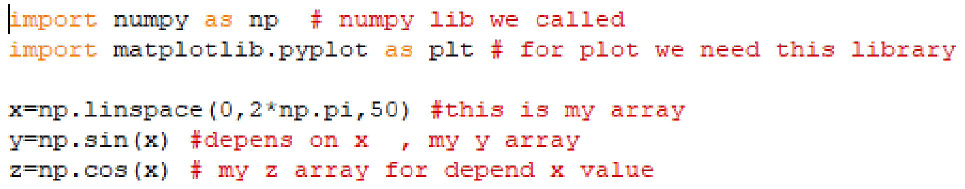
```
import visa

rm=visa.ResourceManager(r'C:\Windows\System32\visa32.dll')
rm.list_resources()
print(rm.list_resources()[0])

myinst=rm.open_resource(rm.list_resource()[0])

print(myinst.query('*IDN?'))

mydevice=rm.get_instrument(rm.list_resources()[0])
mydevice.write("RST")


#     USB0::0x0699::0x0368::C027917::INSTR
```



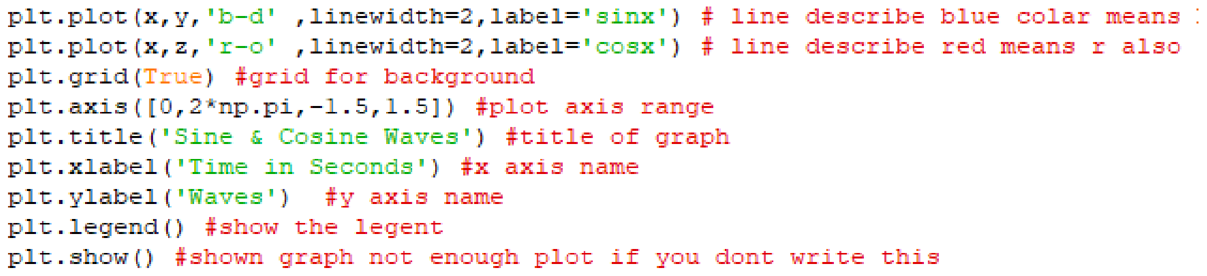
```

int analogPin = 0;
int data = 0;
char userInput;

void setup(){

  Serial.begin(9600);          //  setup serial

}

void loop(){


      data = analogRead(analogPin);    // read the input pin
      Serial.println(data);
      delay(1);

} // Void Loop
```



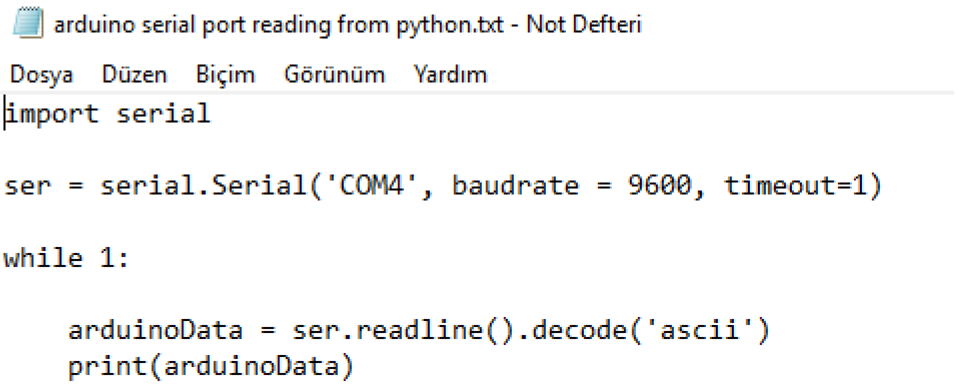
```
import serial
from datetime import datetime
import sys, os, serial, threading


port = 'COM4' ;
baud = 9600

def monitor() :
    ser = serial.Serial(port, baud ,timeout=0)
    while True:
        line = line.decode()
        line = ser.readline()
        text = "VT.txt"   (str(datetime.now()))
```



```
import numpy as np  # numpy lib we called
import matplotlib.pyplot as plt # for plot we need this library

x=np.linspace(0,2*np.pi,50) #this is my array
y=np.sin(x) #depens on x  , my y array
z=np.cos(x) # my z array for depend x value
```

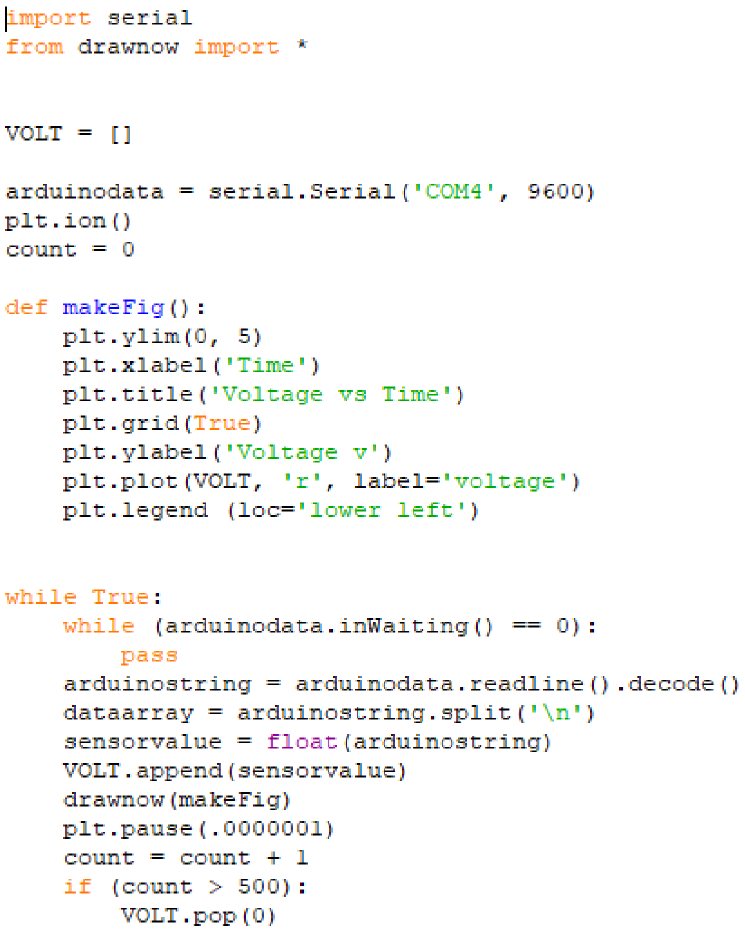


```
plt.plot(x,y,'b-d' ,linewidth=2,label='sinx') # line describe blue colar means
plt.plot(x,z,'r-o' ,linewidth=2,label='cosx') # line describe red means r also
plt.grid(True) #grid for background
plt.axis([0,2*np.pi,-1.5,1.5]) #plot axis range
plt.title('Sine & Cosine Waves') #title of graph
plt.xlabel('Time in Seconds') #x axis name
plt.ylabel('Waves')  #y axis name
plt.legend() #show the legent
plt.show() #shown graph not enough plot if you dont write this
```

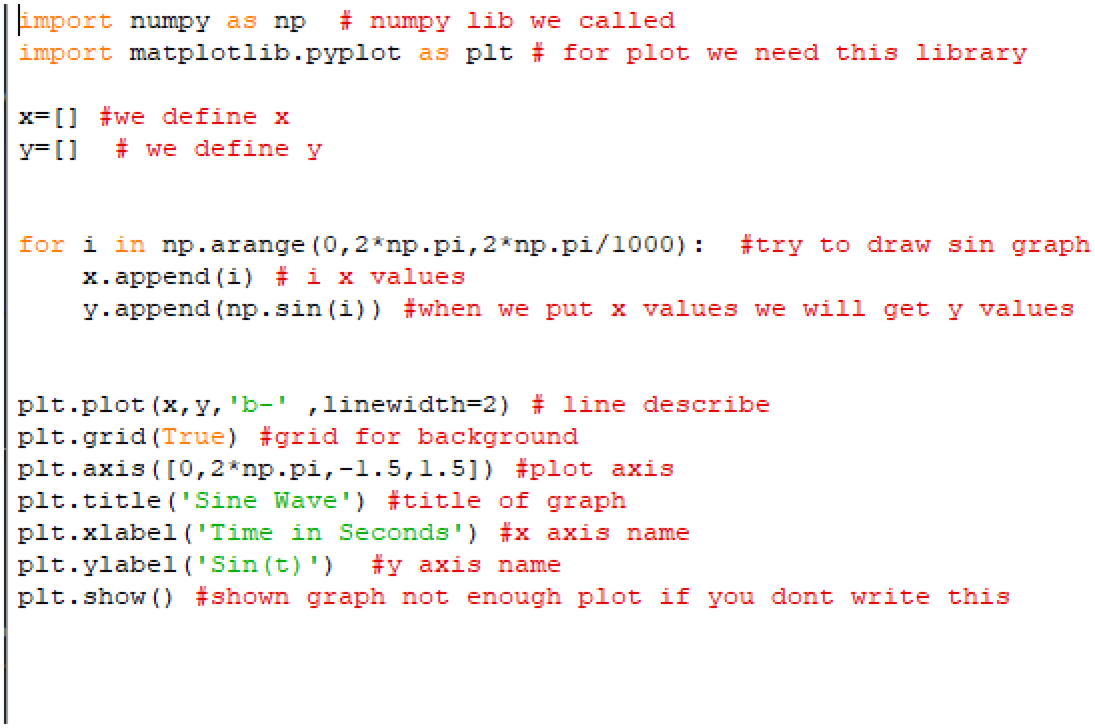


arduino serial port reading from python.txt - Not Defteri
Dosya Düzen Biçim Görünüm Yardım

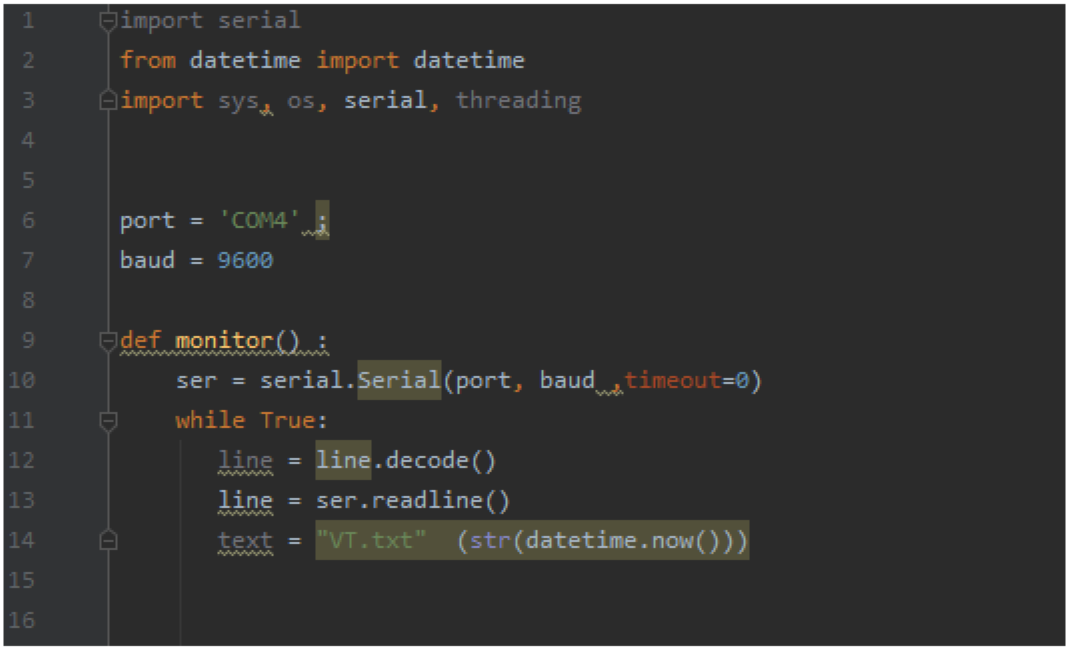
```
import serial

ser = serial.Serial('COM4', baudrate = 9600, timeout=1)

while 1:

    arduinoData = ser.readline().decode('ascii')
    print(arduinoData)
```



```
import serial
from drawnow import *


VOLT = []

arduinodata = serial.Serial('COM4', 9600)
plt.ion()
count = 0

def makeFig():
    plt.ylim(0, 5)
    plt.xlabel('Time')
    plt.title('Voltage vs Time')
    plt.grid(True)
    plt.ylabel('Voltage v')
    plt.plot(VOLT, 'r', label='voltage')
    plt.legend (loc='lower left')


while True:
    while (arduinodata.inWaiting() == 0):
        pass
    arduinostring = arduinodata.readline().decode()
    dataarray = arduinostring.split('\n')
    sensorvalue = float(arduinostring)
    VOLT.append(sensorvalue)
    drawnow(makeFig)
    plt.pause(.0000001)
    count = count + 1
    if (count > 500):
        VOLT.pop(0)
```



```
import numpy as np  # numpy lib we called
import matplotlib.pyplot as plt # for plot we need this library

x=[] #we define x
y=[]  # we define y


for i in np.arange(0,2*np.pi,2*np.pi/1000):  #try to draw sin graph
    x.append(i) # i x values
    y.append(np.sin(i)) #when we put x values we will get y values


plt.plot(x,y,'b-' ,linewidth=2) # line describe
plt.grid(True) #grid for background
plt.axis([0,2*np.pi,-1.5,1.5]) #plot axis
plt.title('Sine Wave') #title of graph
plt.xlabel('Time in Seconds') #x axis name
plt.ylabel('Sin(t)')  #y axis name
plt.show() #shown graph not enough plot if you dont write this
```



```python
import serial
from datetime import datetime
import sys, os, serial, threading


port = 'COM4' ;
baud = 9600

def monitor() :
    ser = serial.Serial(port, baud ,timeout=0)
    while True:
        line = line.decode()
        line = ser.readline()
        text = "VT.txt"  (str(datetime.now()))


```