

## Serial data logger for Arduino / ESP-X modules

The aim the project is to make a program which can acquire data serially via com ports and store them in an excel based file format.

Hardware used:

1. Arduino /Esp-X module
2. Laptop / PC
3. Any sensor (not essential)

Software used:

1. Python 3.X
2. Arduino IDE / Thonny / ESPlorer
3. Spyder / Python IDLE / Thonny
4. MS office / Open office / Libre Office

### Data Acquiring:

Acquire data using any sensor and print it serially in any of the baud rate. In this example the program uses the noise in the ADC of Arduino:

```
void setup() {  
    // put your setup code here, to run once:  
    Serial.begin(9600);  
    pinMode(A0,OUTPUT);  
    digitalWrite(A0,LOW);  
}  
  
void loop() {  
    // put your main code here, to run repeatedly:  
    Serial.println(analogRead(A0));  
    delay(1000);  
}
```

## Preparing Logger:

The logger is based on python language, you can use any of your favourite IDE for coding.

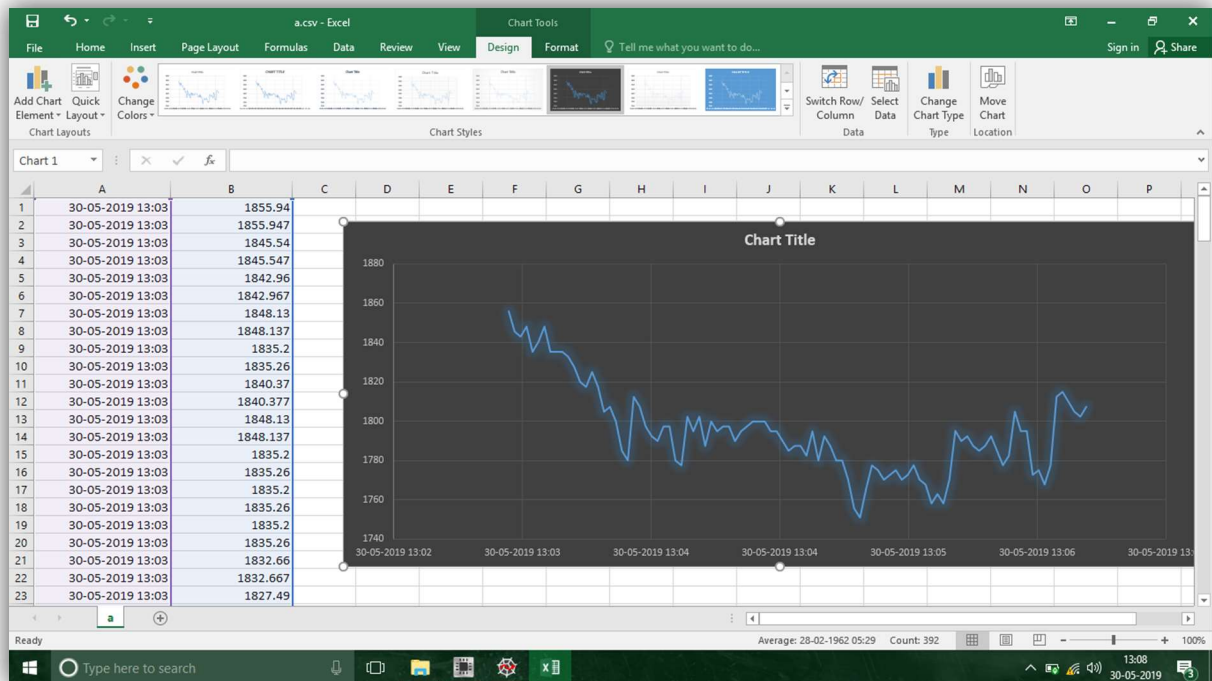
First thing first, in order to get data serially, one must open a portal or port to the external real world (kind of creepy one). Luckily python supports serial data management, you can use this using **serial** library. For the sake of time tracking there must be a timestamp, which can be generated using the **datetime** library. Then the file must be opened in **csv format** (preferred) for the data to be stored in the pc.

1. Open the serial port with proper com port selection and baud rate.
2. Open the file in CSV format with proper file location.
3. Store the serially acquired data in a variable (Note: - it will be in Byte format)
4. Convert it to string.
5. Extract system time.
6. Append the system time and data with comma (,).
7. Write it to a file.
8. Repeat from step 3
9. When everything is over, close the serial connection and file.

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Thu May 30 08:36:36 2019
4 task: serial to csv for time analysis
5 @author: Aswath
6 """
7 import serial, datetime
8 x=datetime.datetime.now()
9 x="Logger_"+x.strftime("%d_%b_%Y_%H:%M:%S")+".csv" # Dynamic File name
10 print("file name : "+x+"\n")
11 try:
12     ser=serial.Serial("com4",9600)          # Estabbling Serial connection
13     f1=open(x,"w")                          # Openning file
14     while 1:
15         line=ser.readline();                # Reading data serially
16         line=line.decode('utf-8')           # Byte to string conversion
17         line=line.replace('\n','')          # data formatting
18         x=datetime.datetime.now()           # system timestamp
19         appended_Data=x.strftime("%d/%m/%Y %H:%M:%S")+","+line
20         f1.write(appended_Data)             # Storing data in a file
21         print(line)
22 finally:
23     f1.close()
24     ser.close()
```

## Results:

The following CSV file is created by the entire project. This data can be used to plot data and analyse the trend of the data. [Note: this output is not from the above arduino program]



## Future expansion:

1. GUI can be made ready for easy setup.
2. Multiple channel and multiple file acquisition.
3. More efficient file format can be used for acquisition.