Python Code

Python was used to parse the data received from the sensors.

The below program was used to get a graph based on the values received from the sensor. The data was also saved in a csv format after. Data was then combined from all the locations and the pandas tool was used to find the mean value and the ratio of the mean vs the concentration of CO_2 in clean air.

CO2.php

```
<PHP>
# Importing library
import csv
import pandas as pd
import serial
import time
import matplotlib.pyplot as plt

# make sure the 'COM#' is set according the Windows Device Manager
ser = serial.Serial('insert_device_com', 9600, timeout=1)
time.sleep(2)

data = []
for i in range(100):
    line = ser.readline()   # read a byte string
    if line:
        string = line.decode()  # convert the byte string to a unicode
string
        num = float(string) # convert the unicode string to an int
        num2 = int(num)
        data.append(num2) # add int to data list
ser.close()

# build the plot
plt.plot(data)
plt.xlabel('Time[s]')
plt.ylabel('CO2 [ppm]')
plt.title('CO2 vs. Time')
plt.show()


CO2 = pd.DataFrame(data) #convert data to a pandas dataframe
# opening the csv file in 'w+' mode

file = open('CO2_(insert-initial.csv', 'w+', newline ='')
CO2.to_csv(file,index = False)

#combining data taken from all three locations
```

```
L = pd.read_csv('CO2_L.csv')  #reading the csv files
K = pd.read_csv('CO2_K.csv')
B = pd.read_csv('CO2_B.csv')
df = pd.concat([L,K,B], axis = 1) #combining all the files horizontally

# calculating the mean for each location
mean = df.mean()
ppm = 400 #defining ppm of CO2 in clean air
ratio = mean/ppm #calculating the ratio of ppm of each location with
the ppm of clean air

#adding the mean and ratio to the dataframe
df.loc['mean'] = mean
df.loc['ratio'] = mean/ppm

#converting the dataframe to a csv file
file1 = open('CO2.csv', 'w+', newline ='')
df.to_csv(file1,index = False)
</PHP>
```

The below program was used to get a graph based on the sound in decibels vs time. The data was received from the sensor LM393.

Sound.php

```
<PHP>
# Importing library
import csv
import pandas as pd
import serial
import time
import matplotlib.pyplot as plt

# make sure the 'COM#' is set according the Windows Device Manager
ser = serial.Serial('insert_device_com', 9600, timeout=1)
time.sleep(2)

data = []
for i in range(100):
    line = ser.readline()   # read a byte string
    if line:
        string = line.decode()  # convert the byte string to a unicode
string
        num = float(string) # convert the unicode string to an int
        num2 = int(num)
        data.append(num2) # add int to data list
ser.close()

# build the plot
```

```
plt.plot(data)
plt.xlabel('Time[ms]')
plt.ylabel('Sound[dB]')
plt.title('Sound vs. Time')
plt.show()



</PHP>
```

Back to report

From:
https://student-wiki.eolab.de/ - **HSRW EOLab Students Wiki**

Permanent link:
https://student-wiki.eolab.de/doku.php?id=amc2021:groupl:extras:python:start

Last update: **2023/01/05 14:38**