import serial

import time

import numpy as np

import matplotlib.pyplot as plt

from scipy.stats import norm

# Replace 'COM14' with your serial port name

serial\_port = 'COM5'

baud\_rate = 9600

#parity = serial.PARITY\_ODD

parity = serial.PARITY\_NONE

stop\_bits = serial.STOPBITS\_ONE

bytesize = serial.EIGHTBITS

def main():

try:

# Initialize serial connection

ser = serial.Serial(port=serial\_port,

baudrate=9600,

parity=parity,

stopbits=stop\_bits,

bytesize=bytesize,

timeout=1)

# Receive 100000 Byte

prev\_rx = 0

data\_cnter = [0] \* 256

for i in range(15000):

rx\_data = int.from\_bytes(ser.read(2), byteorder='little')

if rx\_data != prev\_rx:

data\_cnter.append(rx\_data)

prev\_rx = rx\_data

#data\_cnter[rx\_data] += 1

#print(rx\_data)

print(data\_cnter)

# Plot the histogram of the data

n, bins, patches = plt.hist(data\_cnter, bins=10, density=True, alpha=0.6, color='g')

# Calculate the mean and standard deviation of the data

mu, std = np.mean(data\_cnter), np.std(data\_cnter)

# Create a range of values (x) from the minimum to maximum bin edges

xmin, xmax = plt.xlim()

x = np.linspace(xmin, xmax, num=1000)

# Calculate the normal distribution values for each x

p = norm.pdf(x, mu, std)

# Plot the normal distribution curve

plt.plot(x, p, 'k', linewidth=4)

plt.show()

except serial.SerialException as e:

print(f"Error opening the serial port: {e}")

except KeyboardInterrupt:

print("\nProgram terminated by user.")

finally:

# Close the serial connection

if 'ser' in locals() or 'ser' in globals():

ser.close()

print("Serial connection closed.")

if \_\_name\_\_ == "\_\_main\_\_":

main()