```python
import serial
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.animation as animation
import sys, time, math

xsize = 300

# configure the serial port
ser = serial.Serial(
    port='COM5',
    baudrate=115200,
    parity=serial.PARITY_NONE,
    stopbits=serial.STOPBITS_TWO,
    bytesize=serial.EIGHTBITS)
ser.isOpen()

# initial read
initial_read  = ser.readline()
print(int(initial_read[1:len(initial_read) - 2]))
if int(initial_read[1:len(initial_read) - 2]) < 1000:
    # skip one line
    ser.readline()

for x in range(10):
    ADC_num = ser.readline()
    ADC_num = int(ADC_num[1:len(ADC_num) - 2])
    ADC_tmp = ser.readline()
    ADC_tmp = float(ADC_tmp[:len(ADC_tmp) - 2]) / 100
    print(ADC_num, '\t', ADC_tmp)

def data_gen():
    t = data_gen.t
    while True:
        ADC_num = ser.readline()
        ADC_num = int(ADC_num[1:len(ADC_num) - 2])
        if ADC_num > 1000:
            continue
        ADC_tmp = ser.readline()
        ADC_tmp = float(ADC_tmp[:len(ADC_tmp) - 2]) / 100
        t+=1
        # val=100.0*math.sin(t*2.0*3.1415/100.0)
        print(ADC_tmp)
        yield t, ADC_tmp

def run(data):
    # update the data
    t,y = data
    if t>-1:
        xdata.append(t)
        ydata.append(y)
        if t>xsize: # Scroll to the left.
            ax.set_xlim(t-xsize, t)
        line.set_data(xdata, ydata)

    return line

def on_close_figure(event):
    sys.exit(0)

data_gen.t = -1
fig = plt.figure()
fig.canvas.mpl_connect('close_event', on_close_figure)
```

```
  ax = fig.add_subplot(111)
  line, = ax.plot([], [], lw=2)
  ax.set_ylim(0, 100)
  ax.set_xlim(0, xsize)
  ax.grid()
  xdata, ydata = [], []

  # Important: Although blit=True makes graphing faster, we need blit=False to prevent
  # spurious lines to appear when resizing the stripchart.
  ani = animation.FuncAnimation(fig, run, data_gen, blit=False, interval=50, repeat=False)
  plt.show()

```