Keegan Bess
January 26, 2022

Currently Working On:

As I was waiting for the parts I needed for the breadboard to arrive, I started working on creating some sort of GUI that can be used for testing. It is written in Python and has serial communication between the arduino microchip and a PC. This can be used for testing with outputs of velocity, voltage, current or anything else that we want to output that can be calculated from the microchip. Right now, because I only just got the current sensor this morning , I have just been directly testing with known print outputs from the Arduino.

```
void setup() {
  Serial.begin(115200);
}

void loop() {
  Serial.println("current = 5 mA");
  Serial.println("Voltage = 2 V");
  Serial.println("Velocity = 10 mm/s");


  delay(1000);

}
```

Figure 1: This is the simple code that is sent to serial communication

```
from tkinter import *
import serial.tools.list_ports
import functools

root = Tk()
root.config(bg = 'white')

ports = serial.tools.list_ports.comports()
serialObj = serial.Serial(port='/dev/cu.usbmodem14301', baudrate=115200, timeout=None)

dataCanvas = Canvas(root, width = 600, height= 500 )
dataCanvas.grid(row =0 , column = 1, rowspan = 100)

vsb = Scrollbar(root, orient = 'vertical', command = dataCanvas.yview)
vsb.grid(row = 0, column = 2, rowspan = 100, sticky='ns')

dataCanvas.config(yscrollcommand=vsb.set)

dataFrame = Frame(dataCanvas, bg = 'white')
dataCanvas.create_window((10,0), window=dataFrame, anchor='nw')
def checkSerialPort():
    if serialObj.isOpen() and serialObj.in_waiting:
        recentPacket= serialObj.readline()
        recentPacketString = recentPacket.decode('utf').rstrip('\n')
        Label(dataFrame, text = recentPacketString, bg='white').pack()


while True:
    root.update()
    checkSerialPort()
    dataCanvas.config(scrollregion= dataCanvas.bbox("all"))
```

Figure 2: Bare bones code using tkinter package in python

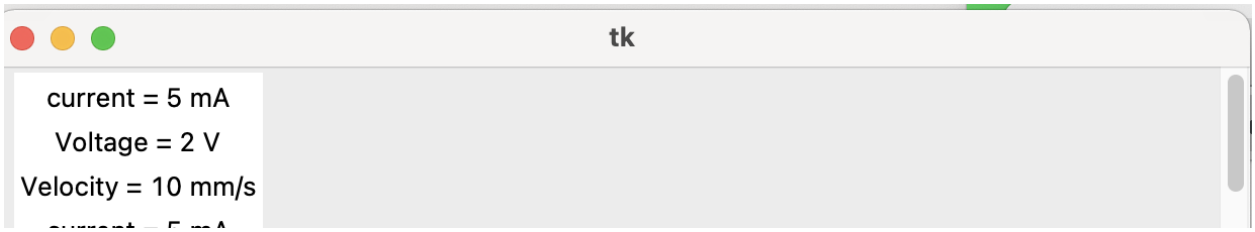


Figure 3: This figure shows the window that appears that continually reads the input from the serial communication with the arduino

I'm still working on making the interface itself prettier and more organized than just a scrolling window, however this is just bare bones to establish that there is serial communication between the arduino and the computer and that there can be a connection through python. I also worked on the alpha test plan which was attached in the email this was sent in.

Future Work:

I finally received all of the parts I ordered for the breadboard design this morning so I will start working on that testing after I acquire the motor. I have to get access to a soldering iron so that I can solder the current sensor pins and motor driver so that I can test it with the motor and the circuit in general. Hopefully if all is well and the breadboard works then I can order the pcb for the beta testing later this semester.

With the software, I want to work through making it look appealing to a user, but ultimately it is for testing to ensure that the current sensor is reading the right values from the motor and also can output whatever the team wants to see, like if they want to see what the velocity of the motor is, without something exterior testing the circuit.