Lab 0x03 - Parsing Vector Image Files

This assignment is to be completed in your lab groups. You will have *two* weeks to complete this assignment. For this lab you will implement the design you developed in HW 0x01.

Assignment

For this assignment you will be generating `.hpgl` files and parsing them with your Micropython Firmware. Additionally you will use the algorithm developed in homework to convert the coordinates from the graphics file into proper motor coordinates.

1. Using a vector drawing program of your choice, generate a simple `.hpgl` file containing some basic shapes like lines and polygons. The free program Inkscape would be a good choice. Play with the export settings to properly understand the units in the exported `.hpgl` file.
2. Move the `.hpgl` file to your Nucleo board. You may eventually consider automating the file transfer with a Python script running on the PC. If the file is not immediately available in your Micropython REPL try unplugging and plugging back in the Nucleo after the file is completely finished transferring.
3. Write some Micropython code to parse the `.hpgl` file to extract the commands. To get started, consider first splitting the contents of the `.hpgl` file at semicolons to separate each command, then splitting each command at commas to isolate the commands from the arguments.

   Note: You do not need to interpret every command that the `.hpgl` file includes. You may strategically choose to ignore certain commands, but at a *minimum* you must be able to interpret "pen-up" and "pen-down" commands.
4. After splitting the data into a set of commands with arguments, convert the set of commands to a list of coordinates to feed to the stepper drivers. This will require several steps:
   - Application of any sort of offset or scaling needed to match the units from the `.hpgl` file to the units on your drawing surface.
   - Interpolation of coordinates to help the robot draw straight lines, as discussed in lecture.
   - Conversion of position coordinates to motor axis coordinates in units of steps or microsteps, whichever is required for your configuration of the TMC4210 from Lab 0x02. That is, port the Newton-Raphson algorithm from HW 0x02 to work in Micropython.
5. You do not need to actually spin your motors for this assignment! Combine code from HW 0x02 and Lab 0x01 to allow data transfer between the Nucleo and your PC. Send the motor angles from the Nucleo to the PC in comma separated format, as in Lab 0x01, and then create an animation of the robot drawing the graphic using your code from HW 0x02.

Requirements and Deliverables

To receive full credit on this assignment you will submit a PDF to Canvas in memo format. Your memorandum should be quite brief for this assignment. As with previous assignments, describe the methodology and approach succinctly.

Also include the following content as attachments, or where appropriate, in the body of the memo:

1. Include a the original vector image that you exported as an `.hpgl` file.
2. Include a plot generated from the motor angle data transmitted from the Nucleo to the PC.
3. Include an animated GIF, similar to HW 0x02, showing the image being plotted.