# Ahmad Uzair Kochai 20360005

# Robot Drawing Project Plan

# Outline of the Problem to be Solved

The main problem is to convert a string of characters to G-Code for the Robot to output on a piece of paper.

Important given information:

* Limit of the writing area is 100mm by 50mm
* User will input a height for the characters between 4mm and 10mm
* Grid area is 18

Ascii character values and coordinates to draw the characters has been given in the SingleStrokeFont text.

Rows with the number 999 has the ascii number character and number of lines of code to be outputted.

A function named Scale must first be created in order to scale the X and Y coordinates for each character. When the user inputs a height value h between 4 to 10mm, the X and Y coordinates are first divided by 18 then multiplied by the h value.

Example:

User input h = 5mm

X = 12, Y = 9

Scaled Coordinate X = 12 x (5/18)

Scaled Coordinate Y = 9 x (5/18)

The test text has been provided to output the characters onto the paper by the robot.

ASCII values are already built into C so assign ascii\_value = (int) char for each character in the test code.

Afterwards the coordinates and Pen State for each character will be obtained as a pointer array and a second function will be created called GCode.

The function will printf every code into G code as G0/1 Xx Yy S0/1000 in which G0 means it goes to the coordinate without drawing, G1 meaning it will draw a line from the previous coordinate, S0/1000 simply means pen up/down.

Finally, the G-Code will be sent to the Arduino which will then be sent to the robot to carry out the task.

# Key Data Items

|  |  |  |
| --- | --- | --- |
| Name | Data type | Rationale |
| FontData | Structure | Stores the ASCII character coordinates and pen state (PenDown). It contains integer fields for L and M coordinates and PenDown (0 or 1). This allows you to store each character's coordinates and drawing instructions together. |
| \*X | Float | Array of scaled X coordinates for the ASCII characters. Each value represents an X position on the canvas after scaling based on user-defined height. |
| \*Y | Float | Array of scaled Y coordinates for the ASCII characters. Each value represents a Y position on the canvas after scaling based on user-defined height. |
| h (Height) | Float | The height input from the user, used to scale the character dimensions between 4mm and 10mm. Each coordinate pair in FontData is scaled by dividing by 18 and multiplying by h. |
| GCodeLine | char[] | Array (or string buffer) to store formatted G-code commands like "G01 X10 Y20 S1000". This buffer holds the command before sending it to the Arduino. |
| |  | | --- | | ascii\_value |  |  | | --- | |  | | int | Stores the ASCII integer code for each character, used to look up the character’s corresponding coordinates in FontData. |

Extend table as required

# Function Declarations

*Only include functions that you will develop.*

*float Scale( float L, float M, float \*X, float \*Y)*

*Float GCode( float \*X, float \*Y, int PenDown)*

*Parameters:*

*L – input X coordinates*

*M – input Y coordinates*

*\*X – output scaled X coordinates*

*\*Y – output scaled Y coordinates*

*Return value – returns 1 if successful, 0 if failed*

*\*X – input scaled X coordinates*

*\*Y – input scaled Y coordinates*

*\*PenDown – input to identify whether a coordinate has 1 or 0 Pen state*

*Return value – returns 1 if successful, 0 if failed*

# Testing Information

|  |  |  |  |
| --- | --- | --- | --- |
| Function | Test Case | Test Data | Expected Output |
| GCode | Convert coordinates to G-code with pen down | Scaled X and Y coordinates, PenDown = 1 | G-code output with format G1 Xx Yy S1000 indicating pen is down (drawing mode) |
| GCode | Convert coordinates to G-code with pen up | Scaled X and Y coordinates, PenDown = 0 | G-code output with format G0 Xx Yy S0 indicating pen is up (movement without drawing) |
| Scale | Verify scaling accuracy | Original X and Y coordinates | Scaled X and Y coordinates that match the scaling formula X \* (h / 18) and Y \* (h / 18), where h is the input height |
| main | Full ASCII character drawing test | Sample ASCII string input | Complete G-code sequence for each character in the input string, with correct pen up/down commands and scaled coordinates |

*Extend table as required. Note that ‘Function’ includes main()*

# Flowchart(s)

May be included as separate pdf