Course Project Documentation

BE Project 2011-12

Character Writing Robot

TEAM 06

Amarjit Prasad, BE EXTC

Abhishek Kharche, BR COMP

Chinmay Pednekar, BE COMP

Vaibhav Naidu, BE COMP

Table of Contents

1. INTRODUCTION

Autonomous Robots are Robots that can perform desired tasks in unstructured environments without continuous human guidance. Many kinds of Robots have some degree of autonomy. Different Robots can be autonomous in different ways. A high degree of autonomy is particularly desirable in fields such as space exploration, cleaning floors, mowing lawns, and waste water treatment.

The Robot that we are going to built will be able to write with a pen on a paper. The SPARK programming board with ATMEGA16 microcontroller controls the execution of the program. A mechanism is used to hold the pen and write on the paper. As the mechanism available (i.e. gripper) is very heavy for the SPARK Robot to hold, we will make a robust Robot body and a robust mechanism to move the pen in different directions.

The mechanism made will be installed on the Robot body and as the mechanism moves the pen will trace it and eventually form a character. The input to the Robot will be random and will be given by the user from the computer window. The input character is sent to the Robot using wireless communication and then executed at the Robot end.

The whole Robot will work in such a way that all processing part will be done by the computer terminal and the execution part will be done by the Robot controller. Input character is analyzed in the computer and only the line coordinates to draw it is sent through wireless module to the Robot.

The Robot uses an algorithm which makes it write straight lines very precisely, but when it comes to curves it fails to draw them. The curves are therefore formed by many small line segments. Although it will not be an accurate curve but the whole character can be figured out.

The Robot can very easily handle uppercase characters, lowercase character and numbers and even some symbols. New characters can be easily added as long as they can be defined as a set of segments. The GUI made by us can take a whole character string as input and print them one by one on the paper.

2. PROBLEM DEFINITION

To create a Robot equipped with mechanism to write characters on a paper using a pen. The input to the Robot is given by a GUI which takes as input a string of characters to write them on paper.

To create mechanism that is capable of moving pen in different directions. The mechanism should be simple enough to be implemented easily.

There are many projects going on to make Robots more human like. One of the most difficult tasks that humans can do and Robots cannot is writing and that is the main objective of our project.

The main aim of the project is to explore the writing capability of a Robot. To Make a Spark Robot write on a piece of paper with the help of self-developed algorithm and mechanism is our aim in this project. The algorithm will be a modification to the existing stroke method of character generation. The proposed Robot will have three layers one for programming board, other for mechanism and the chassis.

At the end of this report a user will be able to give commands for writing characters on a paper to a Robot through a keyboard. Due to mechanical errors the output characters may not be very accurate but will resemble the character to the most extent.

To make the root write all the characters precisely including curved characters is the objective of our project. Also we will be making the subroutines of all characters available free on internet as a part of open source project suggested by E-Yantra team, IITB.

3. REQUIREMENTS

Hardware Requirements:			
Sr. No.	**Hardware**	**Function**	**Quantity**
1	Spark Robot Programming board (ATMEGA16)	It will be used for execution of the commands given by the user from the computer and to control various features of the Robot like motion, communication, etc.	1
2	Chassis	It will be required to give a robust body to the Robot to hold the mechanism on it.	1
3	DC motors	These will be required in the locomotion of the Robot and also in providing strength to the Robot.	2
4	Mini RC servo motor	It will be used in mechanism for controlled movement of pen to write the characters.	1
5	Stepper motor	Used for moving the mechanism	2
6	ZigBee module	For wireless communication between computer and Robot.	2
7	Acrylic sheet	For making the mechanism	1
8	Gear, caste wheel and other hardware tools	For making the mechanism and the robot body.	NA

Software Requirements		
Sr. No.	**Software**	**Function**
1	AVR editor	For writing the embedded C codes.
2	ZigBee drivers and GUI	For sending the characters code from computer to Robot.
3	Visual C++	For making the GUI
4	Java	Simulation using applets

4. PLANNING AND ESTIMATION

4.1 Stepper motors

4.1.1 Working of Stepper Motors

A stepper motor unlike other DC motors rotate step by step. We can specify how many steps it should rotate. A step is a small portion of its complete revolution. Doing this helps in many applications where we require precise control on the rotation which the DC motors cannot give directly. It also helps maintaining the position of the shaft.

The stepper motor consists of electromagnetic coils that are placed around the shaft which has a permanent magnet. In the figure below we have shown a small example of how the stepper motor is inside. The four terminals that come out of the motor allows us to control these electromagnets.

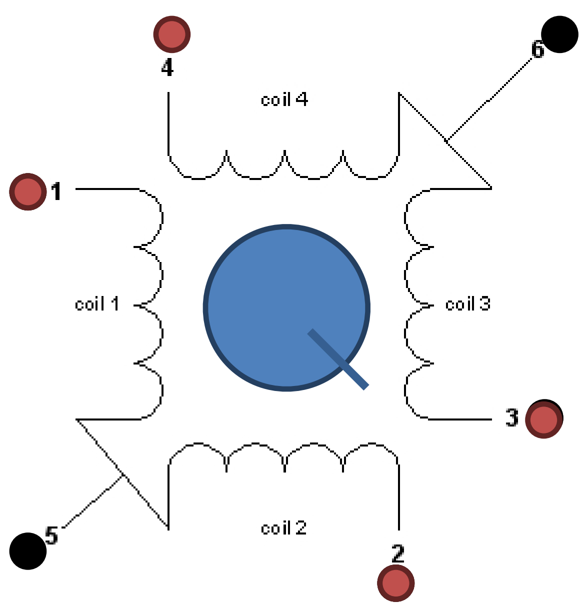

Figure 5.1 : Internal diagram of stepper motor

When we turn the first two electromagnets, the teeth of the shaft is alligned with these magnets. Turning on the next pair of magnets, the teeth of the shafts are aligned with them. Doing these in continuing succession the shaft keeps rotating.

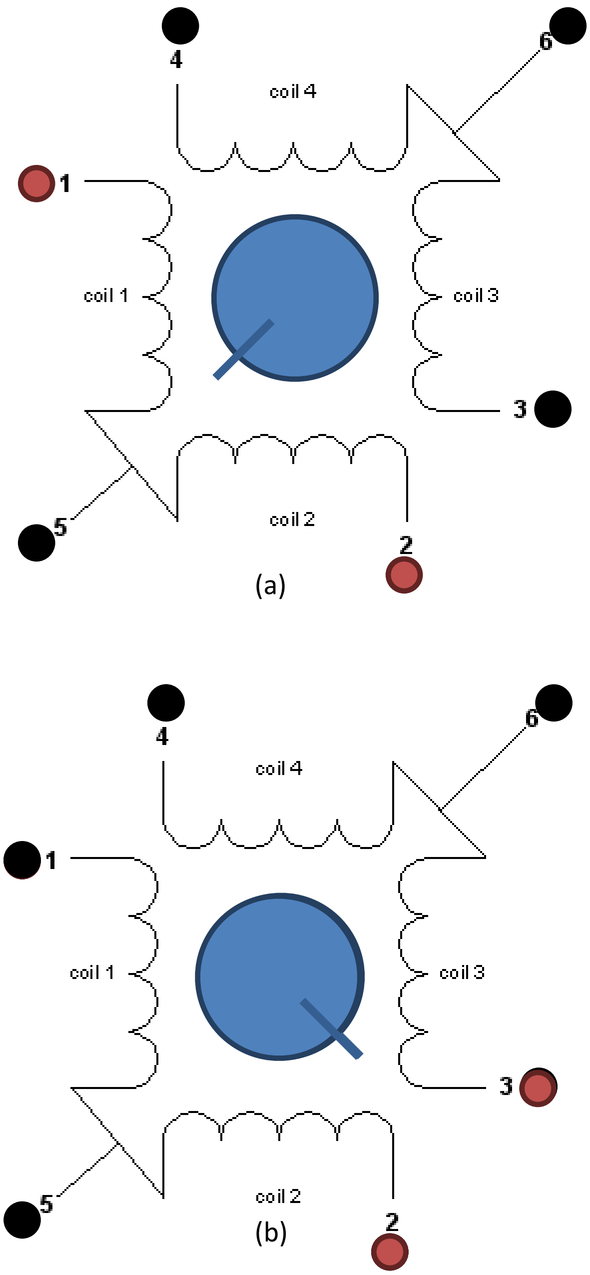

Figure 5.2: Working of stepper motor

(a) Coil 1 and coil 2 are active

(b) Coil 2 and coil 3 are active

4.1.2 Controlling the stepper motor

The SPARK V development board comes with a motor driver L293D. This motor driver is used to control the two DC geared motors used on the development board. Since we are not requiring this much control with the DC motors, we use this motor driver to control the stepper motor. The four outputs of the L293D are connected to the stepper motor controlling pins. The input to this driver is given by PB0, PB1, PB2 and PB3 which were earlier used to control two stepper motors. Since there are 2 stepper motors, we required 2 motor drviers to control them. Therefore we developed our own addon board which splits the original connections given to the L293D into two ports. The second stepper motor is controlled by the pins PA0, PA1, PA2 and PC3. The pins of port A were initially used for the ADC and the PC3 was initally used for a buzzer.

4.2 Mechanical Arm

4.2.1 Introduction

The mechanisms to hold the pen was a very challenging job. The available grippers in the market are either too costly to be bought for a BE project or they are too heavy that its weight cannot be balanced properly on the small body of SPARK V.

We decided to build a mechanism on our own. But the designs available on the internet are very complex. Also these mechanisms have more degree of freedom than required. Though it is not a drawback but why waste such mechanisms in doing a job that is less efficient for them given their specification [10]. The kinematics of these equations are too complex. Even if we build such a design and make the equations, it is hard to debug them because of the complex formulas which execute in loops. So we decided to design the mechanism too on our own.

The mechanism seems to be quite simple because it is designed using simple trigonometric formulas. Iterating these formulas gives a powerful arm that can go to any specified point in a 2D Cartesian plane.

The arm consists of 2 links connected to each other. The first link is mounted on a motor connected to the Robot. The second link is mounted on a motor connected to the previous link. The arm is an R-R arm without any translation involved. The only place where we need translation is after completing each character we need to go to next point in the space. That is done by the DC motors attached to the chassis of the Robot.

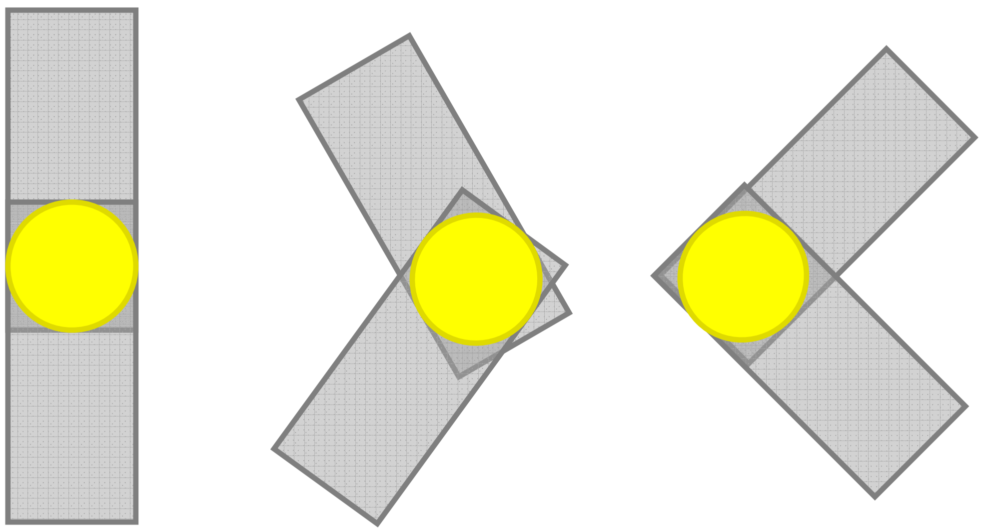

Figure 5.3: Different Configurations of the Arm

The arm can go into different directions thus tracing the strokes we require to draw the characters

4.2.2 Mathematical Calculations

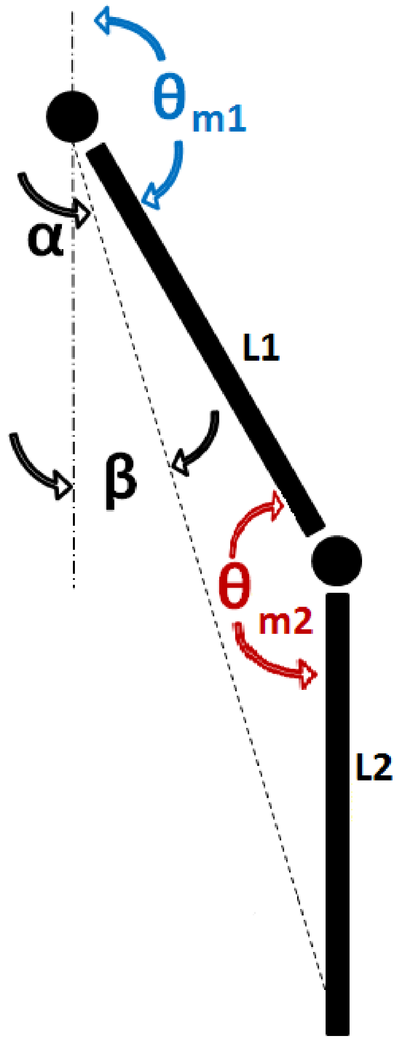

Figure 5.4 : Mathematical Calulations – Naming

As seen in Figure 5.4, the naming convention is Link attached to the Robot is called L1 and the link attached to L1 is L2.

θm1 is the angle made by L1 with the vertical. θm2 is the angle made by L2 with L1.

α is angle made by L1 with the line joining the origin and the tip of L2. B is angle made by line joining tip of L2 with the vertical.

After this, we join the origin with the tip of L2. That gives us a triangle formed by L1, L2 and L (line joining origin and tip of L2)

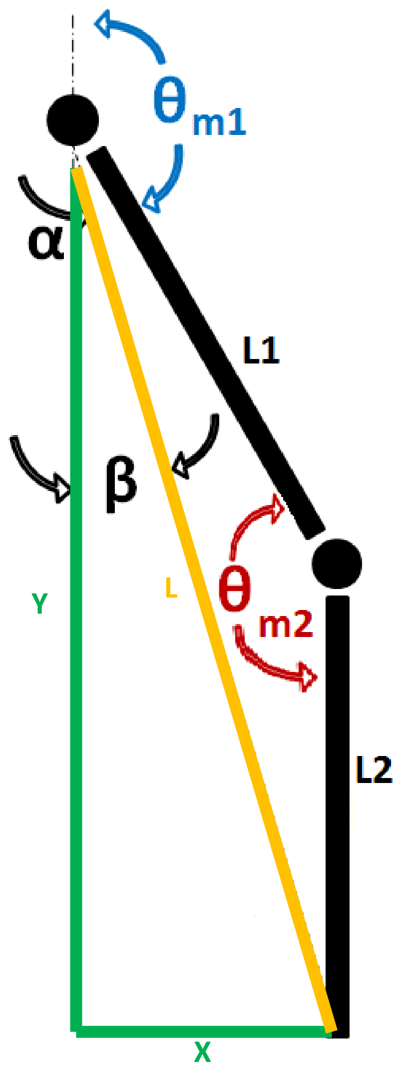

Figure 5.5 : Triangles formed by the links

The working mechanism is completely dependent on the calculations in this part. To start, lets first calculate the length of L

The X span of L is X and Y span of L is Y. (X,Y) are the coordinates of the tip of L2.

Knowing this L can be calculated as

$$L = \sqrt{(X^2 + Y^2)}$$

In the triangle formed by L1, L2 and L, we use cosine law to calculate Cosine of $\theta m2$.

$$\cos \theta m2 = \frac{L1^2 + L2^2 - L^2}{2\, L1\, L2}$$

$\theta m2$ can be found out by taking inverse cosine of this value

Also Cosine of α can be calculated using the same formula

$$\cos \alpha = \frac{L1^2 + L^2 - L2^2}{2\, L1\, L}$$

α can be found out by taking inverse cosine of this value

In the triangle formed by X, Y and L

$$\tan \beta = X / Y$$

β can be calculated by tan inverse of this value

Since $\theta m1$, α and β are supplementary angles, $\theta m1$ can be calculated as

$$\theta m1 = 180 - \alpha - \beta$$

4.2.3 Character Mapping

The character space is of a matrix that is 100 x 100 in dimensions. X increases as we move towards right and Y increases as we move downwards. This is the convention that is mostly used with the graphical environments.

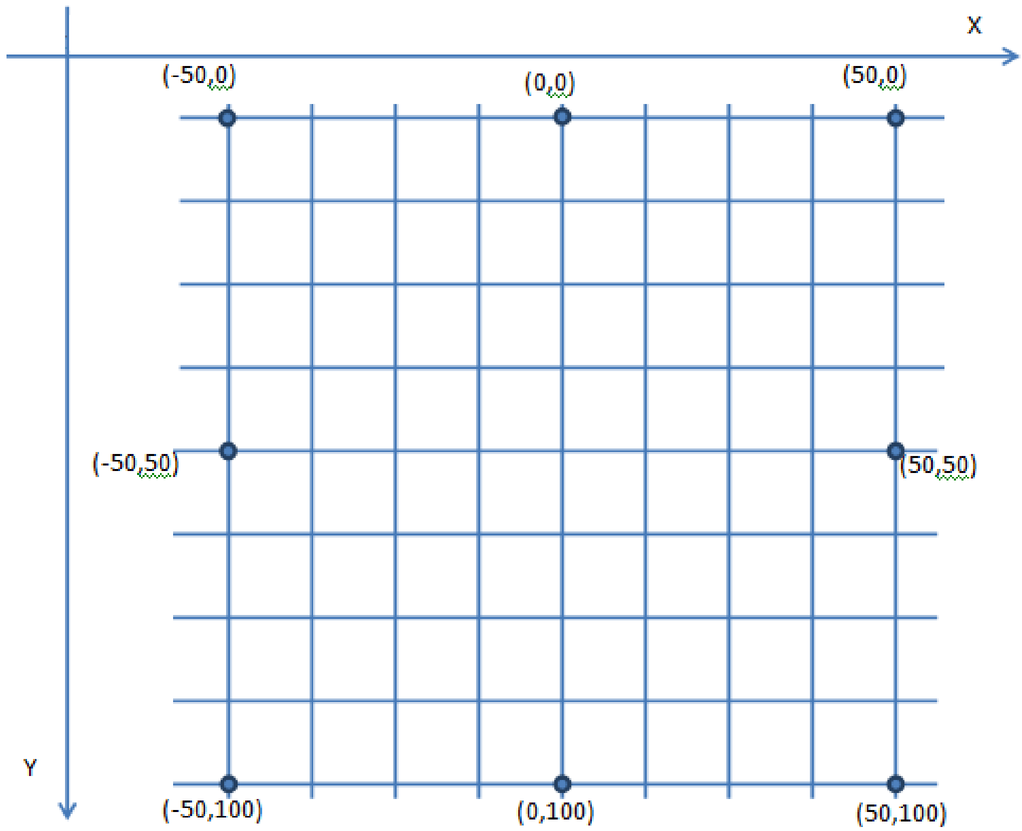

Figure 5.6 : Character Space

In this convention, we have used the point at which the motor is fixed on the Robot as origin so that we can utilize the rotations of the motor on both sides.

Thus we obtain a character space as shown in above figure. Every character that we draw is plotted in the above space.

To draw the character, we plot every point of the lines in this character space. Then we send these points to the Robot. The Robot uses calculations that we mentioned earlier to trace these lines.

As an example, let us plot the character ‘A’

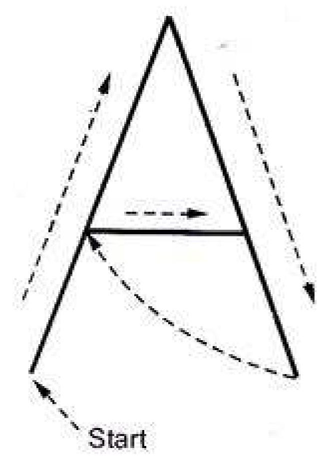

Figure 5.7 : ‘A’ using stroke method

A contains 3 lines that we need to plot. Actually A contains 4 lines. The third is invisible.

To draw this in our character space of 100x100 we will first need to figure out points to which we want to draw the lines.

Let’s say we start at (-50, 100) and go to (0,0). That will be our first line. The second line will be from (0,0) to (50, 100). The third line is from (50,100) to (-25, 50). But this line is invisible. And the fourth line is from (-25, 50) to (25,50).

While drawing this, we must take care that we lift the pen for the lines that are invisible and we place it back for the lines that are visible. Not doing so will draw the invisible lines on the paper too.

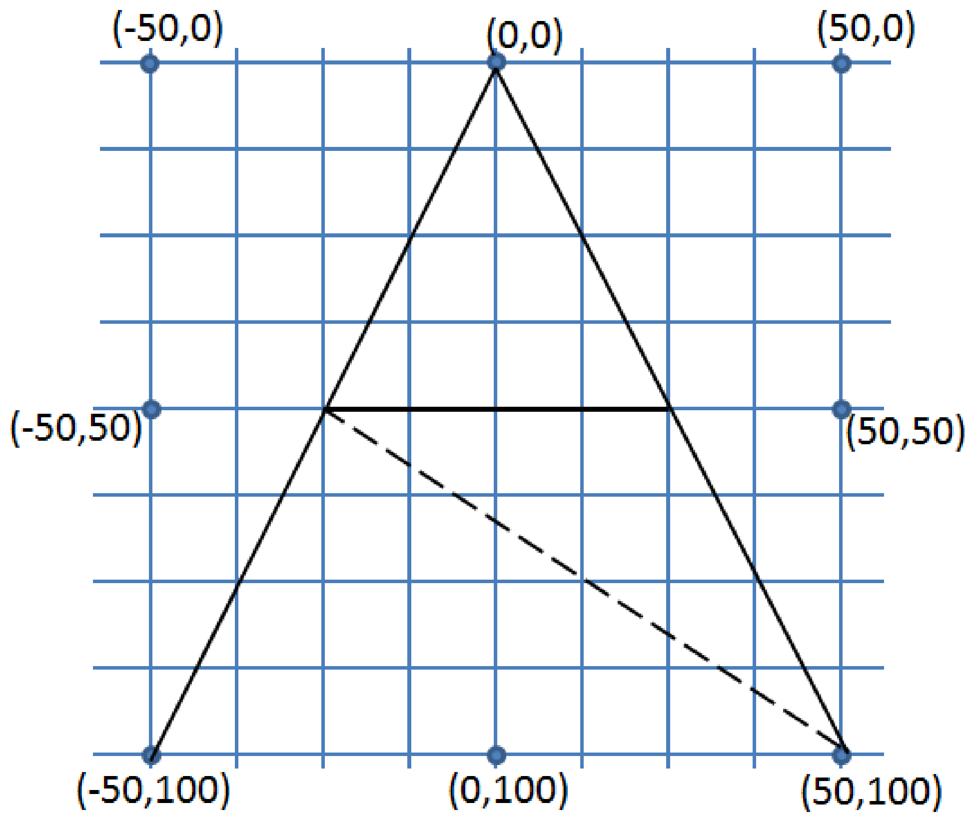

Figure 5.8: Drawing ‘A’ in character space

The steps to do this are as follows:

```
Pen down
Line(-50,100,  0,0)
Line(0,0,  50,100)

Pen up
Line(50,100, -25,50)

Pen down
Line(-25,50,  25,50)
```

The problem with this mechanism is that when the arm moves from one point to other, we have no control on how the motors will rotate between the given two points. So the end result will not necessarily be a straight line. Rather it will never be a straight line. It will be some arbitrary set of curves that we cannot even predict. Therefore the line will look like a scribbling. What we can guarantee is that the line will start and end at the points we specify. To solve this problem we divide the line into 10 equal parts. The co-ordinates of these intermediate points are calculated and 10 lines are drawn instead of one. Now, though the line is crooked in between the effect will not be seen on the entire character.

5. DESIGN

5.1 Data Design

Input data will be only the text to be written on the paper along with the font size. The size of the font is represented in points where each point corresponds to a distance of 5mm on paper.

Depending on the input data, the computer terminal will generate list of co-ordinates for each line segment to be drawn for each character. If instead of a line, a curve is required, appropriate parameters will be passed. This list will be passed to the Robot using wireless communication.

The microcontroller [6] on the Robot will calculate the slope of the lines and will accordingly decide the speed of the motors to draw the lines. In case of a curve, the speed will vary throughout the curve whereas for each line, it will be constant throughout that line.

5.2 Architectural Design

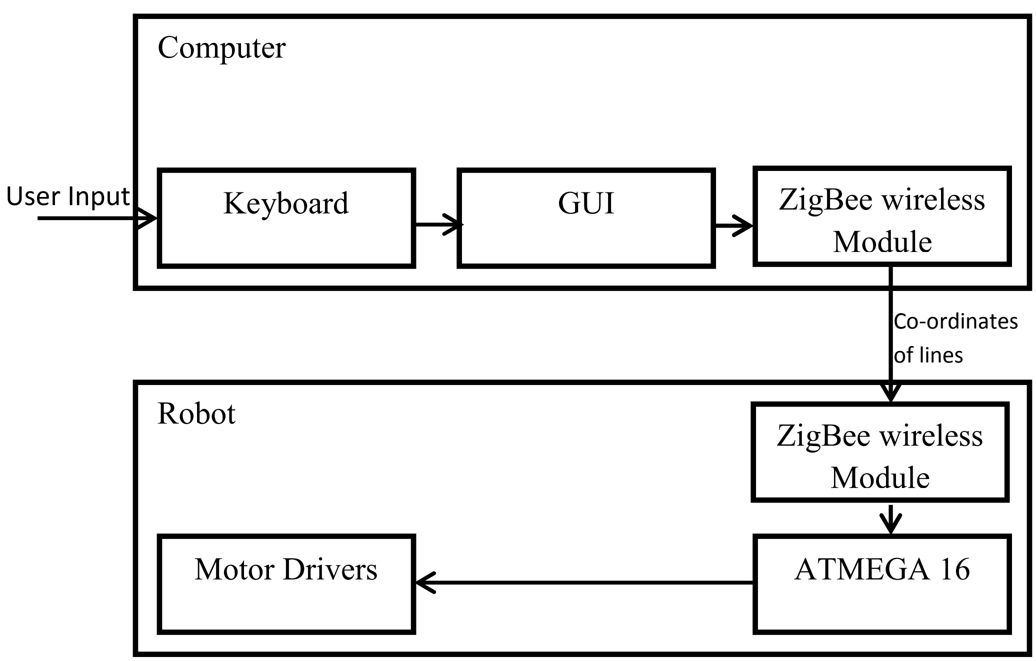

Figure 6.1 : Functional Blok

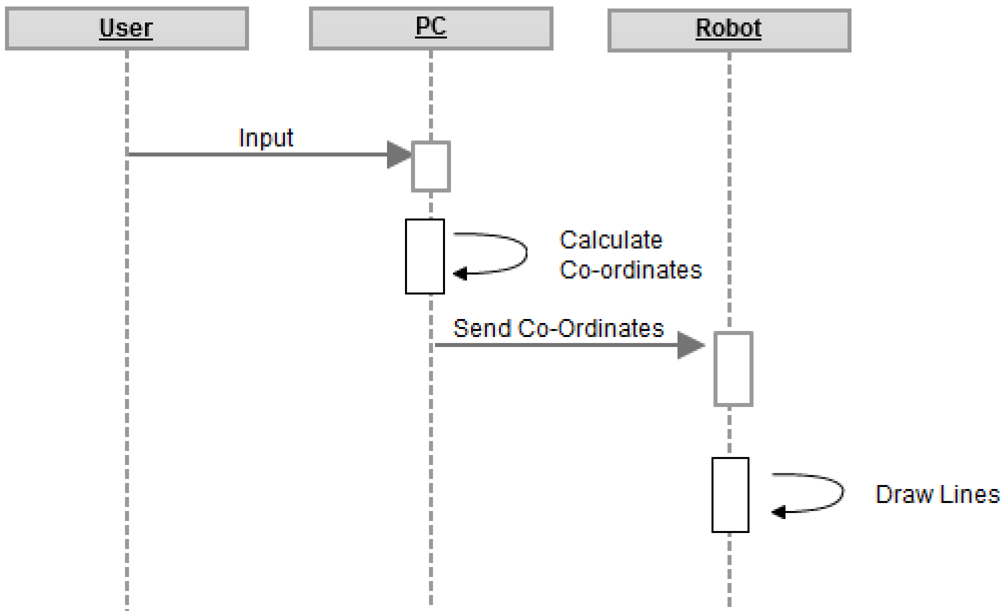

Figure 6.2: Sequence Diagram

There are three actors in this scenario.

User: User is the person who will give the input such as text and the font size to using the application developed for the PC.

PC: PC is a personal computer running windows based operating system and the application. When the user inputs any string, the application running on the PC will generate an intermediate data to be passed to the Robot.

Robot: The Robot will accept the data passed from the PC using a wireless ZigBee[2] module. It will use the data passed from the computer to generate lines on the paper to form the characters in the input string.

5.3 Interface Design

The Graphical User Interface designed for the PC will be similar to the following figure.

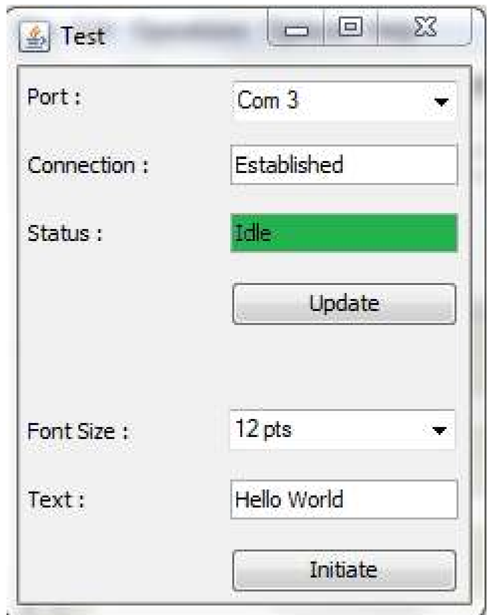

Figure 6.3: Graphical User Interface

The Port field is used to identify the Serial port of the computer on which the ZigBee[2] module is plugged. It is a dropdown combo box that will list all the ports available on the computer.

Once the port is specified, the application will ping the Robot to check the status of the Robot. When the connection is established to the Robot, the connection field will indicate so.

The applications pings the Robot to check whether the Robot is busy writing any alphabets currently or it is free to next the next command. If the Robot is busy, the field will appear red in color and if the Robot is idle, the status field will appear green in color.

The font size field is again a dropdown combo box which will consist a list of predefined sizes that the Robot can write. The user must select the size of the characters so that the application can calculate the co-ordinates.

5.4 Procedural Design

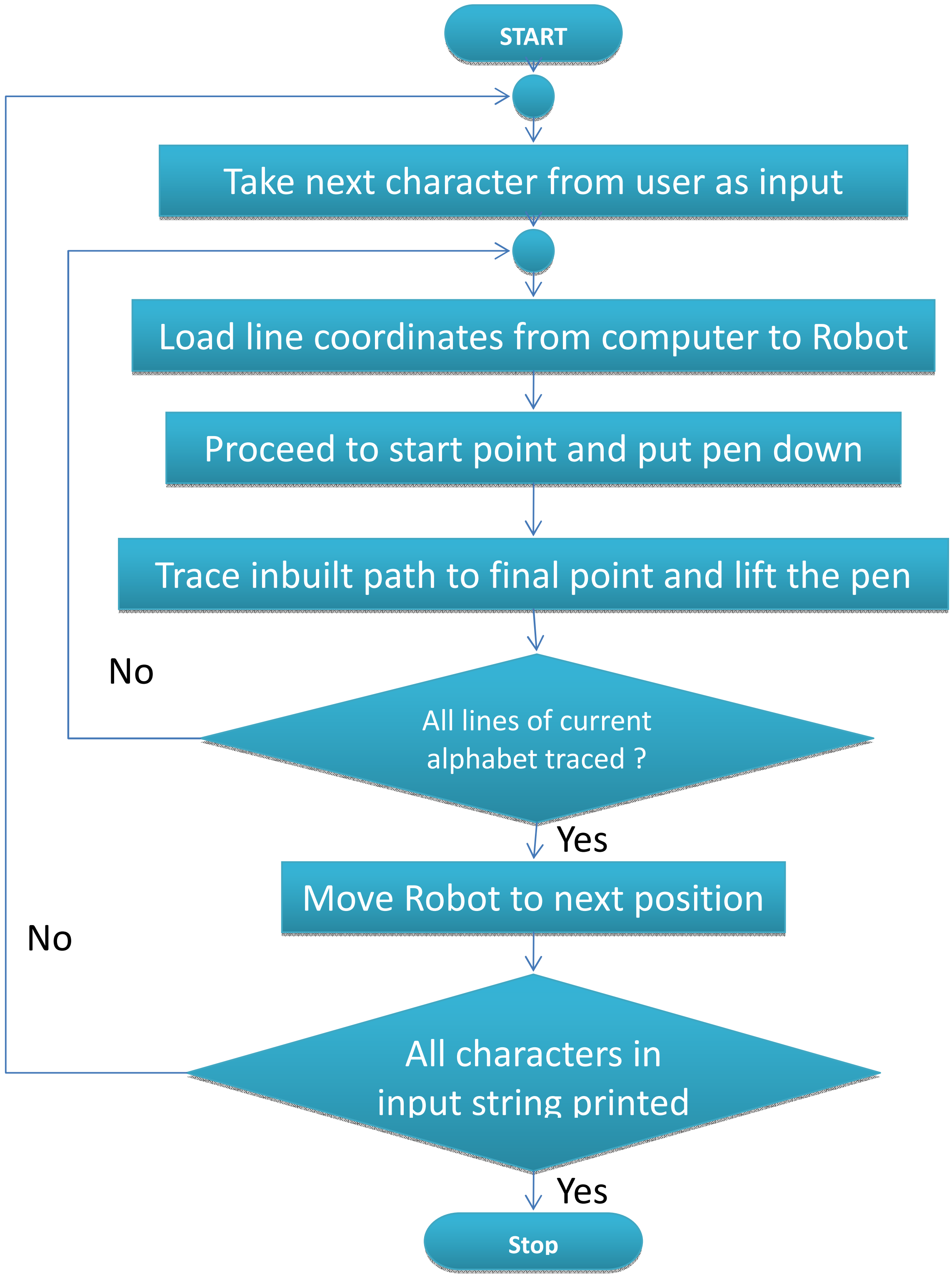

Figure 6.4 : Flow Chart

The flow chart is very simple and straight forward. The user uses the application we have developed on his computer. He selects the font size, the text to write and the port on which the zigbee module is connected. When the user signals the Robot to write the characters, the computer will pass a series of co-ordinates and will also tell the Robot to draw either a curve or line through those points. The Robot will intercept the co-ordinates and will calculate the slope of the lines and then accordingly vary the speed of the motors to generate the desired curves or lines creating a character on the paper.

The pen is initially in air. For each line or curve, it moves to touch the paper till the shape is traced and is then again lifted in the air. The lifting of the pen is also done by a motor controlled by the algorithm itself.

After each character, the Robot moves a step ahead depending upon the size of the character and then repeats all the procedure for the next character

6. RESULTS

6.1 Hardware

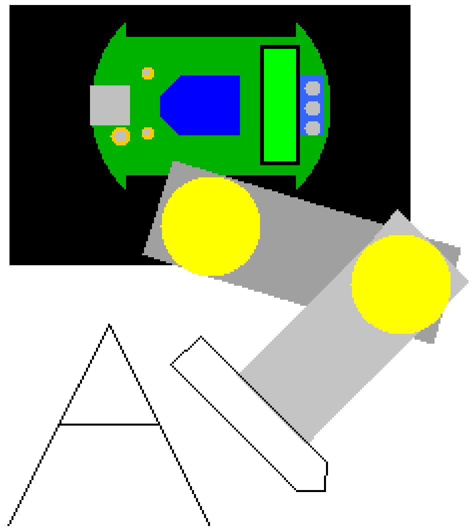

Figure 8.1 : Robot Design

The top view of the mechanism will look as follows. The two RC servo motors are used to move the rack and pinion arrangement which will control the movement of the pen in the X-Y Cartesian plane. Two motors are used, one for each axis. A third motor will be mounted on top of this which will be used to move the pen up or down.

6.2 Software

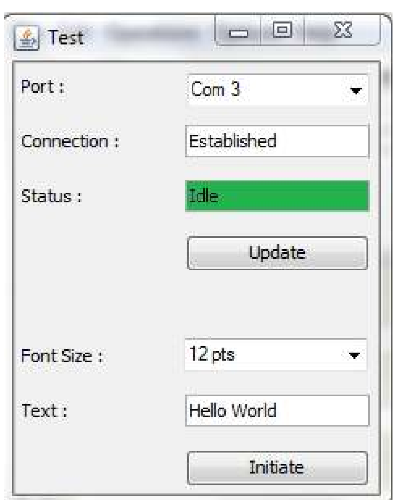

Figure 8.2 : Software Application

The Port field is used to identify the Serial port of the computer on which the ZigBee module is plugged. It is a dropdown combo box that will list all the ports available on the computer.

Once the port is specified, the application will ping the Robot to check the status of the Robot. When the connection is established to the Robot, the connection field will indicate so.

The applications pings the Robot to check whether the Robot is busy writing any alphabets currently or it is free to next the next command. If the Robot is busy, the field will appear red in color and if the Robot is idle, the status field will appear green in color.

The font size field is again a dropdown combo box which will consist a list of predefined sizes that the Robot can write. The user must select the size of the characters so that the application can calculate the co-ordinates.

The Text field is where the user gives the actual string that the Robot is supposed to write on the paper.

7. LIMITATIONS

If we increase the font of character, it is not efficient because the lines are crooked which will be seen if the lines are large. So for drawing large characters, high precision motors are required and dividing each line segment into more small parts is the only solution.

The maximum area to which the robot can draw is limited by the length of its arm. The work environment of this arm is a semicircle of the radius defined by sum of length of both the links, Drawing characters that are larger than this will require changing the length of the arms.

All characters are stored as a subroutines containing information about the line segments required to draw those characters. So to draw the characters, we will have to specify these points for each character. This is a very hectic and time consuming task. Since we are using stroke method of character generation, there is no alternative for this.

All characters cannot be defined as a set of line segments. Some require only curves or are too small to be defined by lines eg: comma(,), full stop(.), brackets((,)). Drawing these characters won't be possible as long as we make this mechanism robust enough to draw curves.

8. DISCUSSION OF THE SYSTEM

8.1 What has worked as per plan?

We made our own mechanism as the mechanism available in the market is too heavy for the robot to carry**.** Two stepper motors were used for the mechanism using sensor pins of the microcontroller ATMEGA 16.

Wireless communication using XBee was used for the user to send character to the robot. Commands to the robot are sent by the user through a GUI.

8.2 What we added more than discussed in SRS?

We made an applet showing virtual demonstration of how our robot would actually write. It also shows the approximation we assumed and the result of our project.

As only one motor driver IC was present on the SPARK V robot and we had to run two stepper motor, servo motor and DC motor simultaneously so we spited the IC into two motor driver suing two enables present in L293D.

8.3 Changes made in plan:

We changed the rack and pinion arrangement with the acrylic hand mechanism as the rack and pinion arrangement was too bulky and didn’t had proper support to be placed over the robot.

To demonstrate virtually how our project works we made a java applet, which shows the approximation we assumed and the end result of our project.

To make the project open source we uploaded all the codes, documents, readme included in our project at following link of GitHub:

https://github.com/amarjit1991/character-writing-robot

9. APPLICATIONS

9.1 Libraries for Future Use

The main aim of IIT Bombay behind this project is to create libraries that can be used in future by anybody who is developing on a similar platform. This project is open source so anybody can implement this in their existing project.

9.2 Plotters

A plotter is a computer printing device for printing vector graphics. Pen plotters print by moving a pen or other instrument across the surface of a piece of paper. This means it is a drawing tool that moves in X-Y co-ordinate plane which is exactly what our Robot does. In today's world where everything needs to be automated, this technique can be modified to work with plotters that can work without any human intervention.

9.3 Vinyl Cutters

Vinyl Cutter is a device that has a blade attached to it that is used to cut shapes from vinyl sheets. In India it is commonly used for making license plates where the each character in the number plate is carved from a vinyl sheet.

10.CONCLUSION

Character writing Robot project has been successful till now as it is recommended as one of the important task by the e-yantra faculty of IITB. As manual work for writing the characters is been reduced greatly, this project is widely accepted. This project has various applications including poster making and making number plates using vinyl cutter. Also this project was never implemented with wireless technology, it provides advanced functions which will be very useful in the future development.

REFERENCES

1. Selective Compliant Assembly Robot
   http://www.robotmatrix.org/SCARARobotic.htm
2. Robot Inverse Kinematics
   http://www.learnaboutrobots.com/inverseKinematics.htm
3. SCARA
   http://en.wikipedia.org/wiki/SCARA
4. Inverse kinematics
   http://en.wikipedia.org/wiki/Inverse_kinematics
5. Spark V, ATMEGA 16:
   http://www.nex-robotics.com/products/spark-v-robot/spark-v.html
6. XBee wireless module:
   http://www.nex-robotics.com/products/wireless-devices/xbee-wireless-communication-module-chip-antenna.html