ACS6502
Group Project Final Report
Group number 16, Domain 1, Autonomous systems
Self - Drawing Manipulator

Ming Zhou, Zhaoda Du, Jiannan Tan

ABSTRACT — The aim of this project is to design a manipulator that can draw picture according to the user's needs. The user only needs to insert an image file in the user interface, and the manipulator can draw it on the paper. Manipulator is now used in many fields, such as industrial production and medical treatment. It can accomplish many tasks that human beings can't, characterized by high precision, large load and long continuous operation time. In order to realize the drawing function of the manipulator in this project, the problems faced are: mechanical and electronic design, the image processing, how to control the joint of the manipulator so that it can accurately guide the end-effector (such as pen) to the corresponding position. The system is divided into two parts, image processing and servo motor control. To assess the performance of the design, a random 30-person survey was conducted. The result is quite successful. The project can improve for other functions, only thing need to replace the end-effector.

1. INTRODUCTION

With the development of computer imaging, the traditional drawing tools have changed. Computer graphics has become a mainstream productivity tool. But ink-pan painting is still very popular in the field of art [1]. It has its own unique artistic value. Many people dream of creating their own ink-pan paintings, but it is very difficult to master such a skill. Even some artists often spend a lot of time to create ink-pan paintings. With this motivation, the group decided to develop a drawing manipulator that can realize the creation of ordinary people's ink-pan painting. There are many similar projects in the field of robot drawing. For example, ISAC [2] learns to draw by tracking human hand movements. Draw-Bot [3] analyzes and records the force between pen and paper to learn how to draw. Both of the above projects have a similar feature. It is necessary to learn the artist's painting process before drawing an ink-pan painting. The disadvantage is that such robots can only draw a single ink-pan painting, which can't meet the flexible needs of ordinary users. In addition, the drawing manipulator can also be used for drawing education. The beginners can learn some drawing skills by repeatedly observing the drawing track of the manipulator. For example, a drawing education mechanical arm developed with Lego building blocks [4].

2. DEVELOPED SYSTEM

2.1 OVERVIEW OF THE SYSTEM

Fig. 1 presents the CAD design of the drawing robot system. The arm 1 and arm 2, which are controlled by motor 1 and motor 2 respectively, should move from 0° to 179°. Before the manipulator begins to draw, the end-effector would be pressed down by the motor 3. After the manipulator completes a continuous curve, the end-effector would be lifted by the motor 3 and prepares for the next point.

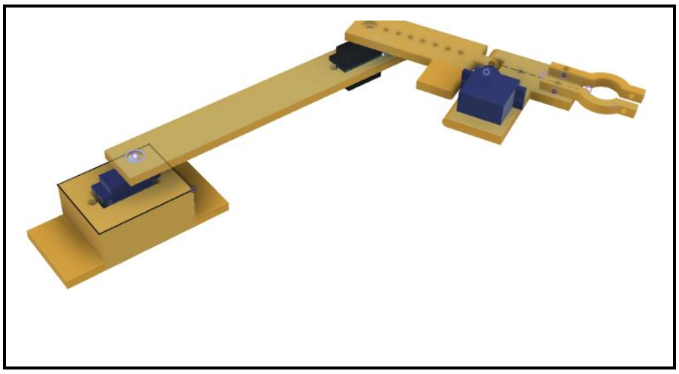


Fig. 1. Overview of the CAD design of the system.

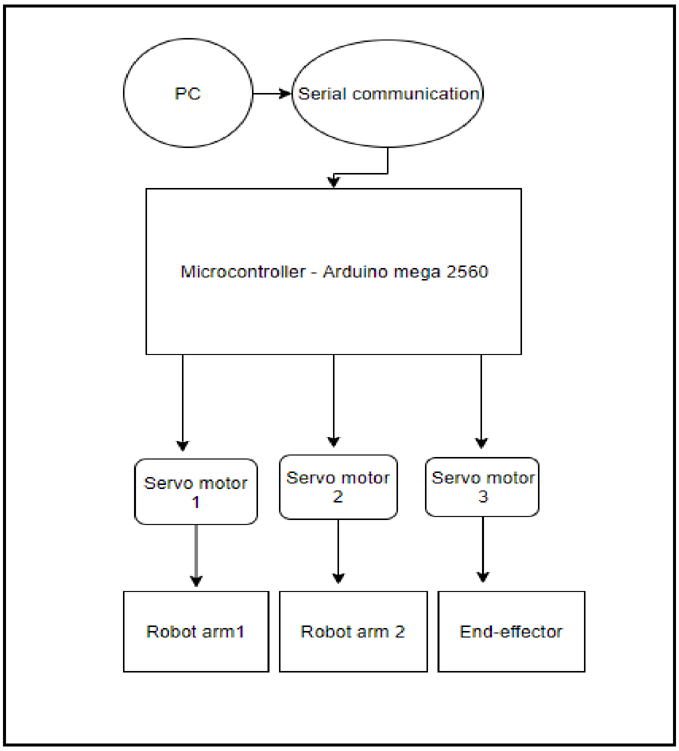


Fig. 2. Components relation diagram describing the system's functional structure.

Fig 2. shows the connection between the joints and MCU to realize the mechatronic behaviour that draws pictures on paper. After the picture was processed by PC or other microcontrollers, the coordinates of the picture would be sent to Arduino Mega 2560 via serial communication.

2.2 MECHANICAL DESIGN

All mechanical parts in this project are manufactured by 3D printer with PLA material. The 3D model is designed from Fusion 360. The appearance of the real manipulator shows in Fig.3 below.

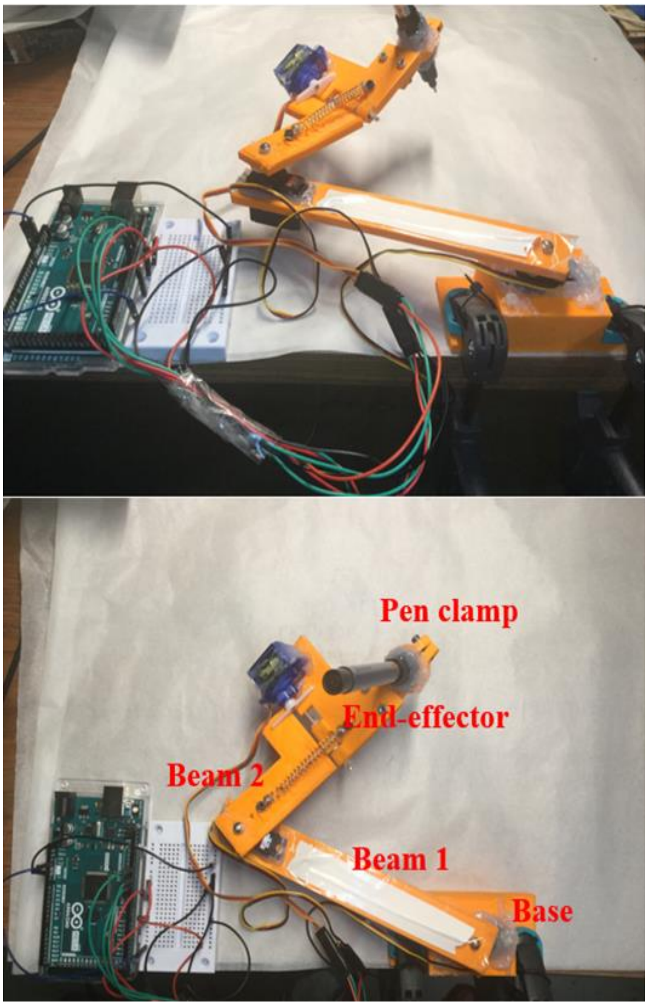


Fig. 3. The real Mechanical design of this manipulator.

This manipulator contains 5 parts, they are Base, Beam 1, Beam 2, End-effector and Pen clamp(They have been marked named in Figure 1). They can be viewed separately in fusion 360 and the parameters of each part shown in Fig. 4 (top view parameters) and Fig. 5 (side view parameters). (they have been put in appendix)

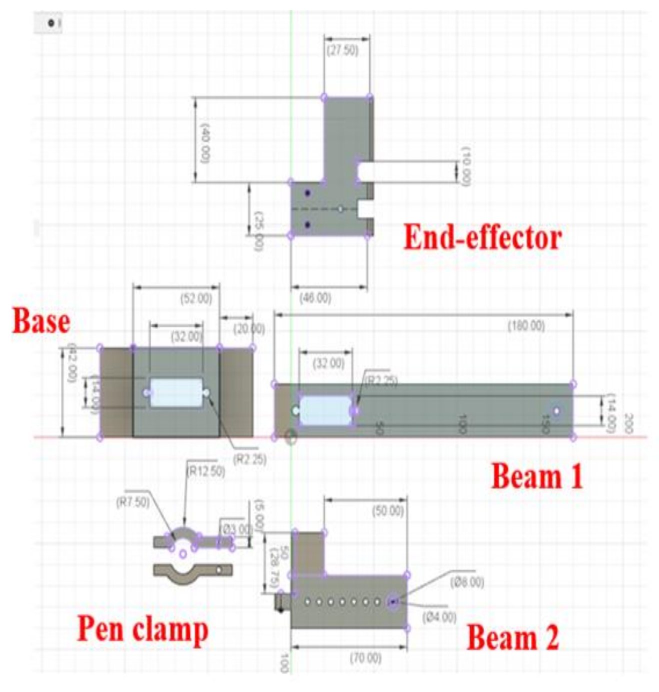


Fig. 4. Top view parameters of each part.

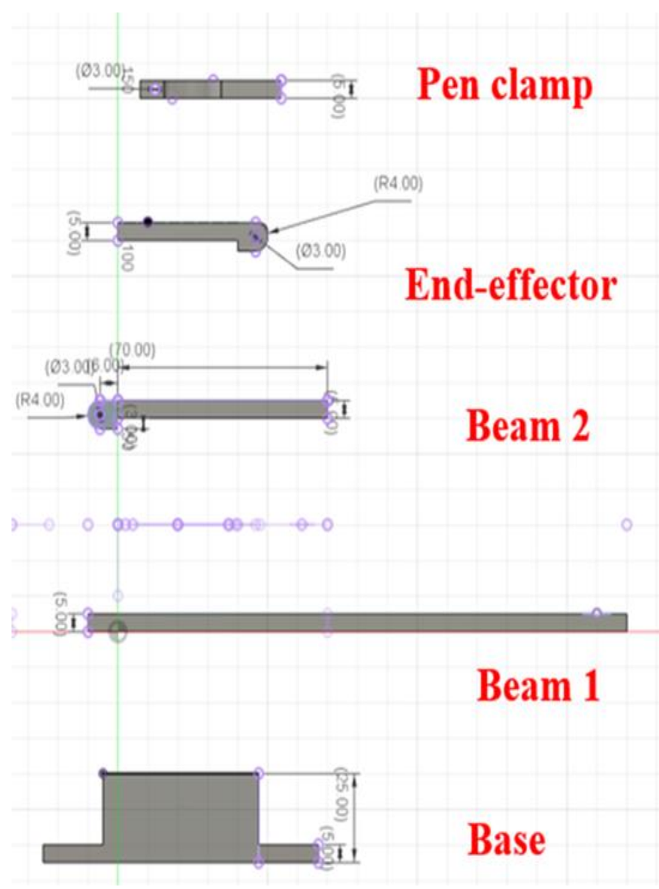


Fig. 5. Side view parameters of each part.

To assemble the parts, screws and hot gun have been used. The structure of this manipulator is strong enough to operate. A spring has been connected between end-effector and beam 2, that's why there are many holes in the centre of beam 2, spring can be stretch according to the torque of the motor on end-effector.

2.3 ELECTRONICS DESIGN

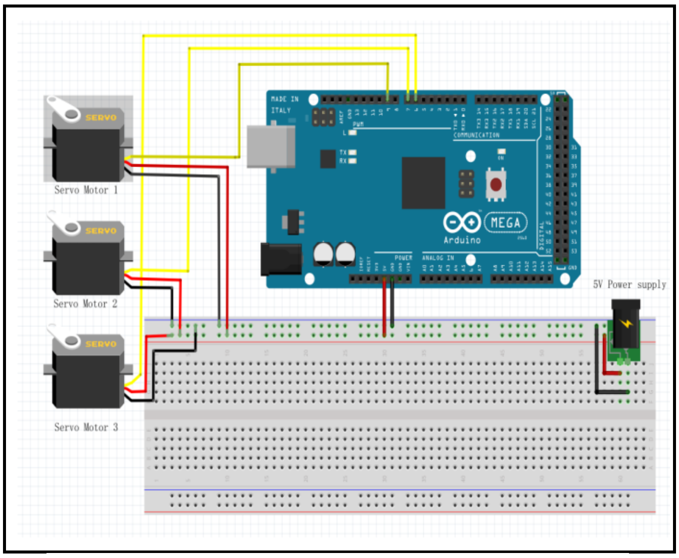


Fig. 6. Schematic and photo of the circuit.

Fig 6. shows the circuit of the system. Two Servo motors (HS-81) [5] have been selected for joint 1 and joint 2, respectively. The torque for these two motors was 3 kg·cm which satisfies the requirements for driving the weights (80g±2g) of all robot arms and a pen. The torque of another servo motor (SG90) [6] for joint 3 was 2.5 kg·cm. Because the motor for joint 3 was used for pressing down or lifting the pen, the requirement of the torque for implementing this operation is smaller than previous joints. Hence, using this servo motor is sufficient.

2.4 BEHAVIOUR DESIGN

In this project, inverse kinematic will be used. The equation of inverse kinematic shows as below.

$$\theta_2 = \arccos\left(\frac{x^2 + y^2 - l_1^2 - l_2^2}{2l_1 l_2}\right)$$

$$\theta_1 = \arctan\left(\frac{y}{x}\right) - \arctan\left(\frac{l_2 \sin\theta_2}{l_1 + l_2 \cos\theta_2}\right)$$

Where:

θ_1 is the turning angle of the motor at joint 1;

θ_2 is the turning angle of the motor at joint 2;

l_1 is the length from the first joint to the second joint;

l_2 is the length from the second joint to pin of the pen;

x and y are the coordinates of the pen.

l_1, l_2 are constant values and they are 145mm and 125mm.x and y are required input variables in this equation.

The working process of this project can be expressed in following Flowchart in Fig.7

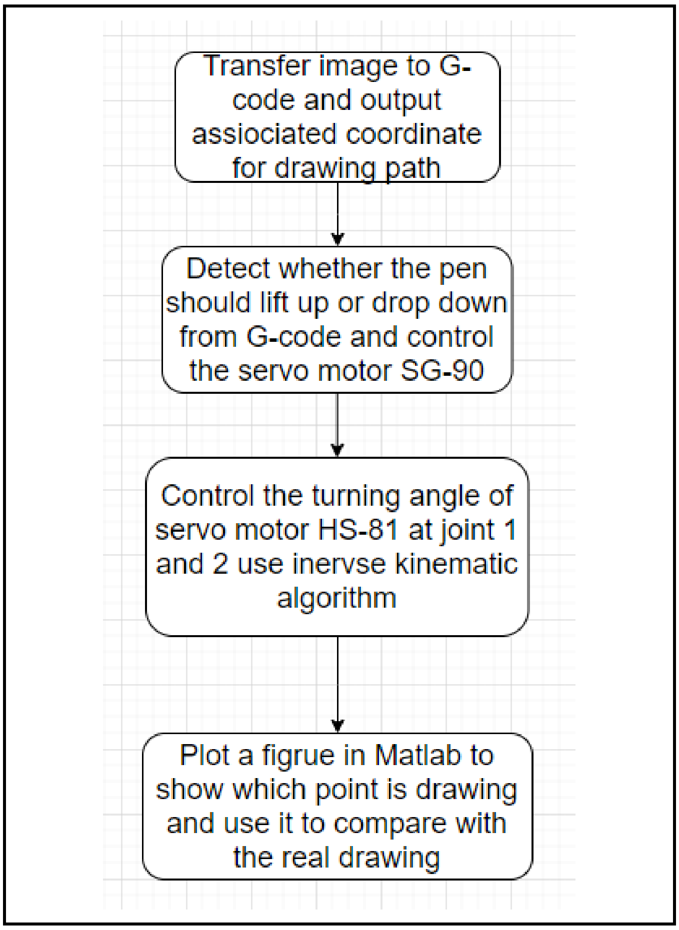


Fig. 7. Flow chart of the main algorithm

According to Fig.7, this project contains four parts.

Firstly, no matter what image has been downloaded from the website, the image will be put into a code (details of code please check Appendix) to transfer the image to G-code and therefore edge of the image will be detected. The coordinates (x, y) of points on edge will be put into a matrix, the matrix then separated into different paths. At the start of each path, a string ‘M3’ will be recorded and at the end of each path, a string’M5’ will occurs. Example of output data can be viewed in Fig. 8. G1 in Fig.8 means on the path. G90 means the start of the whole drawing process. A back up plan incase this code runs any problem in tests is to use a G-code software called ‘laser GRBL’.

```
G90
G0 X50 Y50
M5 S255
F1000
G0 X66.155 Y114.5
M3
G1 X66.51 Y114.5
G1 X66.799 Y114.649
G1 X67.088 Y114.799
G1 X67.769 Y115.057
G1 X68.45 Y115.314
G1 X69.1 Y115.517
G1 X69.75 Y115.72
G1 X69.975 Y115.927
G1 X70.2 Y116.135
G1 X70.2 Y116.252
G1 X70.2 Y116.369
G1 X70.01 Y116.541
G1 X69.82 Y116.713
G1 X69.514 Y116.722
G1 X69.208 Y116.73
G1 X68.179 Y116.218
G1 X67.15 Y115.706
G1 X66.48 Y115.278
G1 X65.809 Y114.85
```

Fig. 8. Example of output data.

Secondly, string ‘M3’ and ‘M5’ will tell the SG90 motor to drop the pen down or lift the pen up. ‘M3’ means start drawing a new path, therefore pen should be drop down on the paper when’M3’ detected. On the contrast, ‘M5’ means lift the pen up to leave the paper and move to the start point of next point. (check the second code in Appendix)

Thirdly, the rotation angles of HS 81 servo motors will be controlled use inverse kinematic. (second code in Appendix).

Fourthly, a figure will be plot through the Matlab, to monitor the process by add point to this figure when running the manipulator. Fig. 9 is example of the plot figure. This is a minor of real figure.

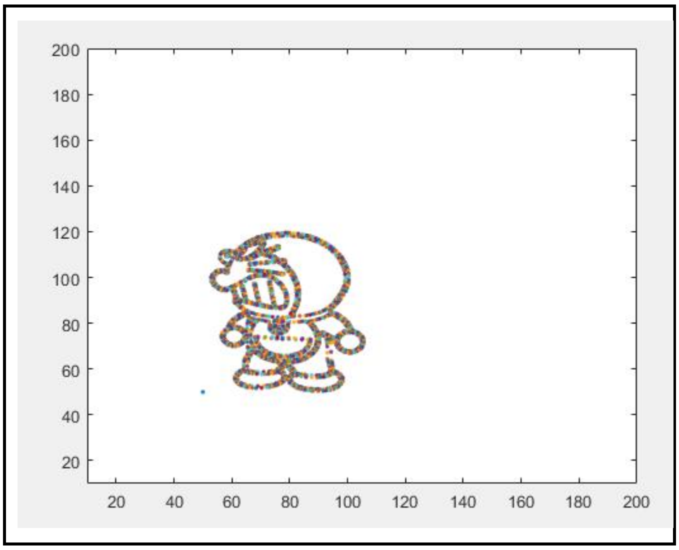


Fig. 9. Example of plot during process.

2.5 CHARACTERISATION

The particularity of this design is that it can't evaluate the result with intuitive data. The group chose to use the form of a questionnaire to 30 randomly selected students from the University of Sheffield. The results are analyzed and evaluated.

- Venue: Diamond, University of Sheffield
- Time: 13:25 - 14:57, 16 / 12 / 2019
- Form: Each of the group are randomly looking for 10 students on the first floor, the second floor and the third floor of the Diamond. Let them evaluate the ink-pan painting drew by the manipulator. There are five levels of evaluation, unrecognized at all, recognized but terrible, not bad, good and perfect.

The pictures evaluated are shown in the figure below. It's a cartoon character image

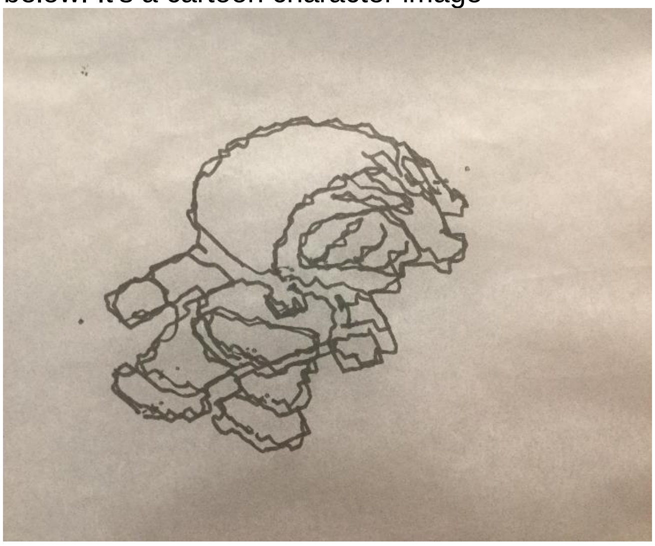


Fig. 10. Drawing results for survey

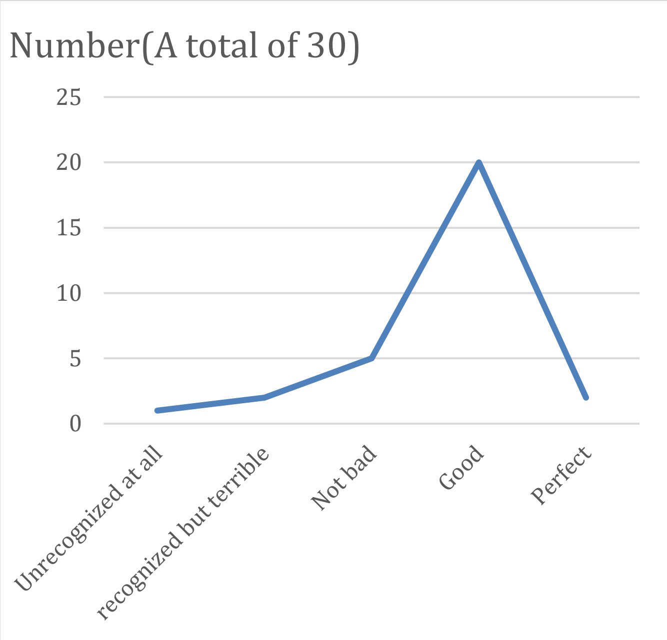


Fig. 11. Statistics of survey results

As can be seen in Fig.11, there are twenty students thought the picture was good, two thought it was perfect, five thought it was not bad, another two thought it could be recognized but it was bad, and one thought it couldn't be recognized.

From this statistic, 90% of the students hold a positive attitude towards the result. About 73% of the students praised it. From the above results, this project can be accepted by people and has certain value.

2.6 OTHER ISSUES, if any

None

2.7 LIST OF OUTSOURCED MATERIALS

The original implementation idea came from [7]. However, the program provided in this article does not work successfully on the platform of this project. Moreover, the image processing and path planning methods proposed in this article are two different methods from this report. In the initial realization of path planning, path planning generation software of laser cutting was used [8]. Based on the operating principle of this non-open source software, a program for image processing and path planning was written in this project.

3. DISCUSSION

This project has fully achieved the purpose of it and received a high level. Audiences that have not seen the original image can recognize what has been drawn by the manipulator. However, the drawing is still not accurate enough, this is limited by the hardware. Servo motor is not the best motor to make a drawing robot, but the limit of budgets is 40 pounds and many step motors are expensive than 20 pounds. This structure and material is not suited for long term use, but enough to finish this project and tests. A stronger structure and material should be used in future design (eg, Aluminium)

4. ACKNOWLEDGEMENT

All the three members of this group want to thank Dr. Shuhe Miyashita, Dr. Dana Damian, Dr Ben Taylor and all the GPAs. The group would also like to thank the 30 students who participated in the survey.

5. REFERENCES

[1] Y. Lu, H. M Lam, Y. Yam, “Preliminary Study on Vision-based Pen and Ink Drawing by a Robotic Manipulator, International Conference on Advanced Intelligent Mechatronics, 2009.

[2] A. Srikaew, M. E. Cambron, S. Northrup, R. A. Peters II, M. Wilkes, and K. Kawamura, “Humanoid drawing robot,” in IASTED International Conference on Robotics and Manufacturing, 1998.

[3] P. Monaghan, “An art professor uses artificial intelligence to create a computer that can draw and paint,” The Chronicle of Higher Education, 2012.

[4] A´. Ha´mori, J. Lengyel, and B. Resko, “3DOF drawing robot using LEGO-NXT”, International Conference on Intelligent Engineering Systems, June 23–25, 2011.

[5] Anon, (2019). [online] Available at: https://www.robotshop.com/media/files/pdf/hs81.pdf [Accessed 20 Dec. 2019].

[6] Ee.ic.ac.uk. (2019). [online] Available at: http://www.ee.ic.ac.uk/pcheung/teaching/DE1_EE/stores/sg90_datasheet.pdf [Accessed 20 Dec. 2019].

[7] Instructables, “Drawing Robot,” *Instructables*, 13-Oct-2017. [Online]. Available: https://www.instructables.com/id/Drawing-Robot. [Accessed: 20-Dec-2019].

[8] LaserGRBL. [Online]. Available: http://lasergrbl.com/. [Accessed: 20-Dec-2019].

6. REFLECTIONS

Considering the difficulty and complexity of this project, we started to prepare for this project very early. In week 3, our group plans to have an offline group discussion every week. At first, everyone put forward various ideas and then analyzed and discussed them. After the experimental classes every week, we asked GTA some ideas about our project. Such as the feasibility of this project, the estimation of budget, what knowledge and technology will be used. We have recognized the painting robot proposed by Tan. We held the view that making this robot is very interesting. After looking up information online, we found that we needed to use much new knowledge, such as image processing, inverse kinematics, and path planning, which made us very excited. In the fifth week, we started the division of labour. Tan is responsible for image processing, Zhou is responsible for circuits and Arduino, and Du is responsible for robot kinematics and mechanical production. It did not go well at first. Du proposed a strategy of traversing all the pixels, but the robot was running very slowly and could not even draw a circle in 10 minutes. When we were at a loss, Tan got the inspiration from iForge's laser cutting machine. After searching the information on the Internet, we found a thing called G-code that can be used to plan the drawing path of the robotic arm, and we figured out how to generate gocode by our own program. After going through mechanical production and debugging, the robot can successfully draw very complicated pictures.

Appendix A: WEEKLY REPORT ATTACHMENT

WEEKLY REPORT 1

ACS6502
Group Project Weekly Report (Week 1, 12/11/2019)

Group 16, (Project domain 1) Autonomous systems, Self-Drawing Manipulator

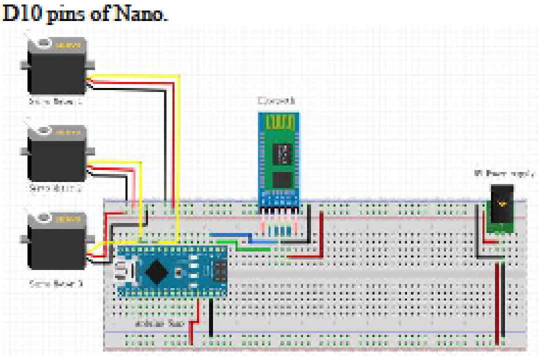
Author: Ming Zhou, Zhaoda Du, Jiannan Tan,

1. *PROGRESS SUMMARY FROM PREVIOUS WEEK AS A GROUP*

Looking back on last week's work, the group completed the design topics and some preliminary plans. The previous week was mainly about finishing the tender proposal, some details has been mentioned in this report.

2. *WORK RECORD OF MEMBERS IN THE WEEK*

Ming Zhou: The main task this week for Ming Zhou is to complete the detailed design plan of the circuit wiring. Because there is no requirement for mobility in this design, the 5V AC-DC power supply is suitable for cost saving. As can be seen a 5V plug-in power supply in the design sketch. The TX and Rx pins of Bluetooth module are directly connected with RX and TX of Nano respectively. Since the module supports wide voltage input of 3.6-6v, external resistances voltage reduction is not required. The PWM input pins of the three servo motors are respectively connected with the D8, D9 and D10 pins of Nano.



Jiannan Tan: This week, the coding environment for robot and debugged the servo motors to figure out their unique attributes have been built up. The coding environment includes Arduino IDE and MATLAB with Arduino Component. Additionally, Servo motors' unique attributes testing included torques ranges and valid rotation angles.

Zhaoda Du: The mechanical parts have been designed on fusion, but only the first edition, it has been putted in tender proposal. The material has been decided which is PLA. The mechanical parts will be manufactured by 3Dprinter.

3. *Remaining Tasks and Mitigation Plan*

Remaining tasks:

For the remaining tasks, the team will preliminarily complete the overall mechanical structure design in the next week, including 3D model printing, polishing and performance debugging. At the same time, complete the selection and purchase of materials (purchase list). The algorithm design of software will be preliminarily completed in the next two weeks, including the image processing, the debugging of Arduino's precise control of the servo motor angle, the calculation and measurement of the specific object (the relationship between the arm length and angle of the drawing manipulator) and the debugging of the wireless communication module. After the successful completion of the above tasks, it is the final integration and testing of each module. Then getting the test results, the team will evaluate the results, and make the final improvement.

Mitigation plan:
The above is under the condition that all tasks are successfully completed. However, it must be taken into account that some plans in the original scheme can't meet the design requirements due to some unexpected reasons.
1. At present, it is impossible to determine whether Bluetooth communication can meet the communication requirements of this design 100% (external interference is not taken into account), so if it is found that it can't meet in the later verification process, cable communication will be used as an alternative.
2. As the 3D model has not been printed yet, it is not known whether the material of the model can meet the design requirements (strength, weight and other factors). If after the 3D model is completed, it is found that the strength is too weak or the weight is more than expected, other materials (such as wood, acrylic board) that can meet the design requirements will be used to make the model by hand.

WEEKLY REPORT 2

ACS6502/231
Group Project Weekly Report (Week 2, 19/11/2019)

Group number 16, Domain 1, Autonomous systems， Self - Drawing Manipulator

Jiannan Tan, Ming Zhou, Zhaoda Du

1. PROGRESS SUMMARY FROM PREVIOUS WEEK AS A GROUP

Almost all mechanical components have been finished via 3D printing in iForge， and they all have been assembled. Meanwhile, coding environment is ready for debugging. Forward kinematic has been designed. Risk assessment has uploaded. The design of the system circuit was completed last week

2. WORK RECORD OF MEMBERS IN THE WEEK

- **Ming Zhou:** This week, the group discussed the system' s strategy while completing 3D model printing. It includes how to process the image, how to track the image data, and then convert it to the angle of each joint of the manipulator. The specific strategy will be discussed, and the preliminary experiments will be organized in the rest of time in this week. In addition, the group is in the process of conforming the final purchase list, which is expected to be completed this week.

- **Jiannan Tan:** Firstly, after visiting the iForge and using 3D printing service there, the workroom reserved before has been used. Additionally, the types of equipment it could provide has been taken notes. Last but not least, the finished 3D printing components have already been collected and assembled. Now, the manipulator is ready for hands-on debugging.

- **Zhaoda Du:** Mechanical model has been printed through the 3D printer in IForge with PLA material. The stiffness of this material has been tested, which is strong enough to handle all conditions when the manipulator operates. Forward kinematic has been calculated. Figure 1 is the 3D printed model.

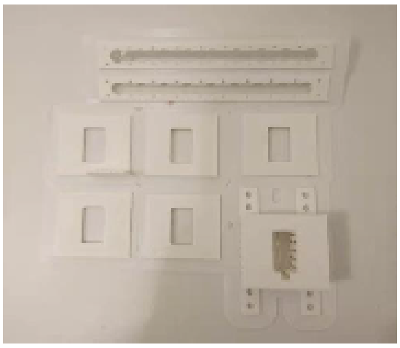


Figure 1 3D Printed Model

3. REMAINING TASKS AND MITIGATION PLAN

Remaining task:

Some of the mechanical parts may need to redesign to fit the motors in the purchase list perfectly. Need to submit the purchase requisition before this Friday. inverse kinematic need to be calculated. When the kinematic is finished, it needs to be applied into Arduino code to operate the manipulator. MATLAB needs to be connected with Arduino to find the coordinates of end joint. Need to think about the frequencies of communication (between MATLAB and Arduino) and the control algorithm output to manipulator, normally communication frequency should be same as control frequency. The greater the frequency is, the better the system will be.

Mitigation Plan:

If the Bluetooth communication doesn't work, Cable communication will be a replacement.

If the mechanical parts not fit the motors or the end joint is too heavy, the material will be changed and the structure will be redesigned.

If the MATLAB can't connected with Arduino or it is too hard when sending coordinate location of end joint, a specific operation will be designed in Arduino code(e.g. draw a circle on paper)

If the frequencies of communication and control can not be the same, the input and the output need to find the best way to fit each other. Decrease the Bluetooth communication frequency if it is bigger than the control frequency.

1

WEEKLY REPORT 3

ACS6502/231
Group Project Weekly Report (Week 3, 26/11/2019)

Group number 16, Domain 1, Autonomous systems, Self - Drawing Manipulator

Jiannan Tan, Ming Zhou, Zhaoda Du

1. PROGRESS SUMMARY FROM PREVIOUS WEEK AS A GROUP

Preliminary inverse kinematics of the manipulator has been calculated. If the redesigned beams made by wood work well, the inverse kinematics could be applied quickly. Meanwhile, the Bluetooth communication between Matlab and Arduino has been tested successfully, after solving several problems during hands-on experiments. Besides, image processing codes for the input picture has been written and run.

2. WORK RECORD OF MEMBERS IN THE WEEK

- **Ming Zhou:** According to the inverse kinematics of the manipulator, this week the group completed the algorithm of solving the joint angles of the manipulator for given coordinates, as shown in Figure 1. The target image is processed into a series of pixels. These pixels will be sent to Arduino, then each joint will be controlled by following algorithm to complete the corresponding action. The code for the corresponding algorithm will be tested this week.

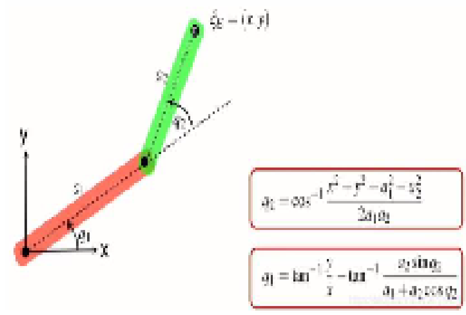


Figure 1: Joint angles equation

- **Jiannan Tan:** In order to send the picture's information to an Arduino, after processing the picture in Matlab, serial communication between Matlab and the Arduino has been carried out. During the tests, some problems have been found, which cost lots of time for debugging. One of them should be reminded is that some reaction time should be given before the character strings are sent via the Matlab, as given in Figure 2 line 5. And parts of image processing in matlab has been done in the week, as shown in Figure 3.

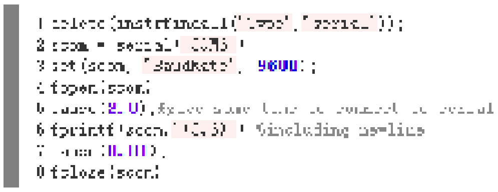


Figure 2: Matlab codes for serials

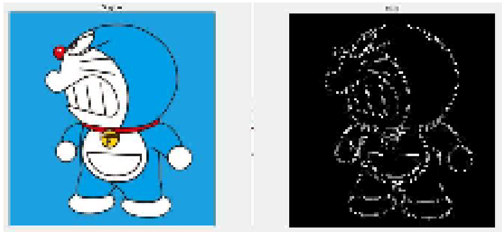


Figure 3: Image processing

- **Zhaoda Du:** The dimension of each part has been redesigned and the material of beam has been changed to wood. To fit the holes from servo motor, pillar drill was used to draw holes on wood board.

3. REMAINING TASKS AND MITIGATION PLAN

Remaining task:

The redesigned beams need to be assembled this Wednesday. Before this Thursday, hands-on debugging of the whole robot should be carried out. The debugging includes Bluetooth communications between the robot and Matlab, inverse kinematics of the manipulator and the quality of the picture drawn by the robot.

Mitigation Plan:

During recording the demo, if the Bluetooth communication between the robot and Matlab couldn't work, serial communication with cable would be a replacement. Additionally, if the robot couldn't process the pixel information from Matlab successfully, for example, the picture drawn by the robot not match the original photo, a specific operation would be designed in Arduino code(e.g. draw a circle on paper).

WEEKLY REPORT 4

ACS6502/231
Group Project Weekly Report (Week 4, 03/12/2019)

Group number 16, Domain 1, Autonomous systems, Self - Drawing Manipulator

Jiannan Tan, Ming Zhou, Zhaoda Du

1. PROGRESS SUMMARY FROM PREVIOUS WEEK AS A GROUP

Last week, the group completed the design and 3D printing of the new mechanical structure. The reason is that the structural strength of the early mechanical structure can not meet the design requirements. In addition, the team completed the edge detection of the target image and the communication between MATLAB and Arduino. The construction of the main algorithm framework of the manipulator inverse kinematics.

2. WORK RECORD OF MEMBERS IN THE WEEK

- **Ming Zhou:**

This week, the group basically completed the inverse kinematics algorithm of the conversion between the end-effector coordinates and the joint angles of the manipulator, and implemented it in the form of code. After the angle of each joint of the manipulator is obtained, the Arduino drives the servo motors to rotate and drive the links to rotate. Under the cooperative motion of the two joints, the end-effector reaches the target coordinate. In fact, the motion track of the end-effector is a straight line between points, but this design reduces the visual image brought by these lines with a certain sampling density. So far, the prototype has entered the final experimental stage, while the final demo is also in the assembly.

- **Jiannan Tan:**

The manipulator assembled last Friday was able to draw some simple patterns. Firstly, three existing servo motors were used for debugging and figuring out their unique performance. The performance is about how to rotate to a designated angle. Although these motors rotate the same angle, the duty cycles they need from the input signal are different. Without knowing their performance, respectively, the quality of the picture drawn by the manipulator would be influenced. Subsequently, according to the inverse kinematics of the planer two-segment manipulator calculated last week, codes about the control of these three servo motors were written. Finally, the end effector of the manipulator could reach the designated point on paper.

- **Zhaoda Du:**

Inverse kinematic has been applied into MATLAB code, the number of points that has been selected from circle to use in inverse kinematic calculation as coordinate(x,y) are 36, which means choose a point in a circle every 10 degrees. In MATLAB, it can be achieved as:
theta=linspace(0,2*pi,36);
x = (radius*cos(theta)+xc);
y = (radius*sin(theta)+yc);
where xc and yc are the centre of the circle.
The manipulator has been tested use MATLAB code to drawing a continuous shape on paper(eg. Circle, rectangular).
The shape tested in figure 1 as below is a circle.

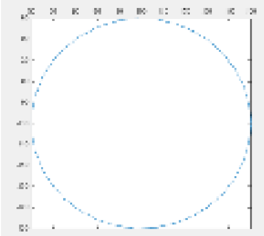


Figure 1 Tested Shape

New 3D printed model has been done shows in figure 2 which has a big stiffness value, density percentage is 30%.

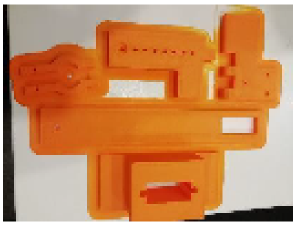


Figure 2 3D print model

3. REMAINING TASKS AND MITIGATION PLAN

Remaining task:
The angle of motor at the first joint will be negative if the lowest point in the continuous shape is too close to x axis, but MATLAB only accept values from 0 to 1(mapping to 0-180 degree). This problem need to be fixed.
Let the manipulator draw something more complex that a circle, fore example, use image processing toolbox to edge detect any figure from internet and draw the edge.

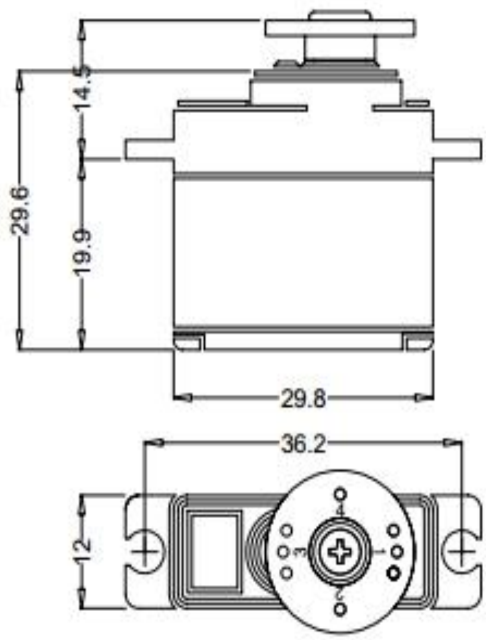
Mitigation Plan:
The project may not able to handle to draw a complex figure , then the simple shape like circles will be used.
Rectangular or triangle will be added as different shape with different center coordinate.
A code on Arduino IDE will be used as backup code.

Appendix B:
Technical Data Sheets of Servo Motor HS-81 and Servo Motor Sg90

ANNOUNCED SPECIFICATION OF HS-81 SUB MICRO SERVO

1.TECHNICAL VALUES

CONTROL SYSTEM	:+PULSE WIDTH CONTROL 1500usec NEUTRAL	
OPERATING VOLTAGE RANGE	:4.8V TO 6.0V	
OPERATING TEMPERATURE RANGE	:-20°C TO +60°C	
TEST VOLTAGE	:AT 4.8V	AT 6.0V
OPERATING SPEED	:0.11sec/60° AT NO LOAD	0.09sec/60° AT NO LOAD
STALL TORQUE	:2.6kg.cm(36.10oz.in)	3kg.cm(41.66oz.in)
OPERATING ANGLE	:40°/ONE SIDE PULSE TRAVELING 400usec	
DIRECTION	:CLOCK WISE/PULSE TRAVELING 1500 TO 1900usec	
IDLE CURRENT	:8.8mA	9.1mA
RUNNING CURRENT	:220mA AT NO LOAD	280mA AT NO LOAD
DEAD BAND WIDTH	:8usec	
CONNECTOR WIRE LENGTH	:160mm (6.29in)	
DIMENSIONS	:29.8x12x29.6mm	(1.17x0.47x1.16in)
WEIGHT	:16.6g(0.58oz)	

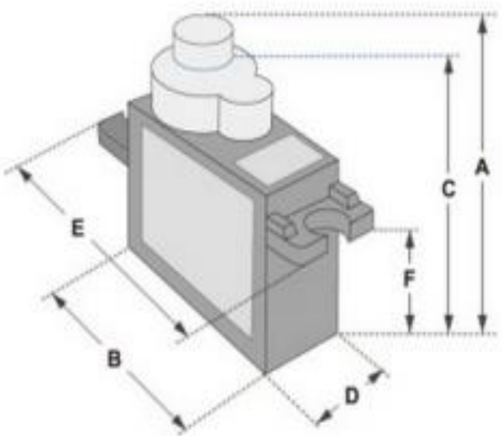


2.FEATURES
3-POLE FERRITE MOTOR
HYBRID I.C
DIRECT POTENTIOMETER DRIVE

3.APPLICATIONS
AIRCRAFT UP TO 15 POUNDS
1/10TH SCALE STEERING

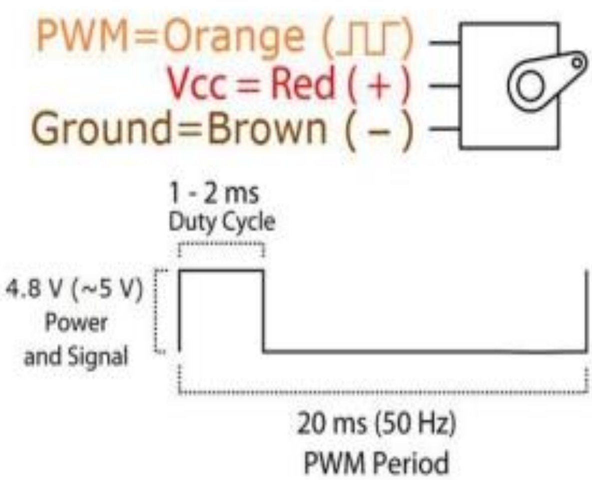
SERVO MOTOR SG90 DATA SHEET

Tiny and lightweight with high output power. Servo can rotate approximately 180 degrees (90 in each direction), and works just like the standard kinds but smaller. You can use any servo code, hardware or library to control these servos. Good for beginners who want to make stuff move without building a motor controller with feedback & gear box, especially since it will fit in small places. It comes with a 3 horns (arms) and hardware.

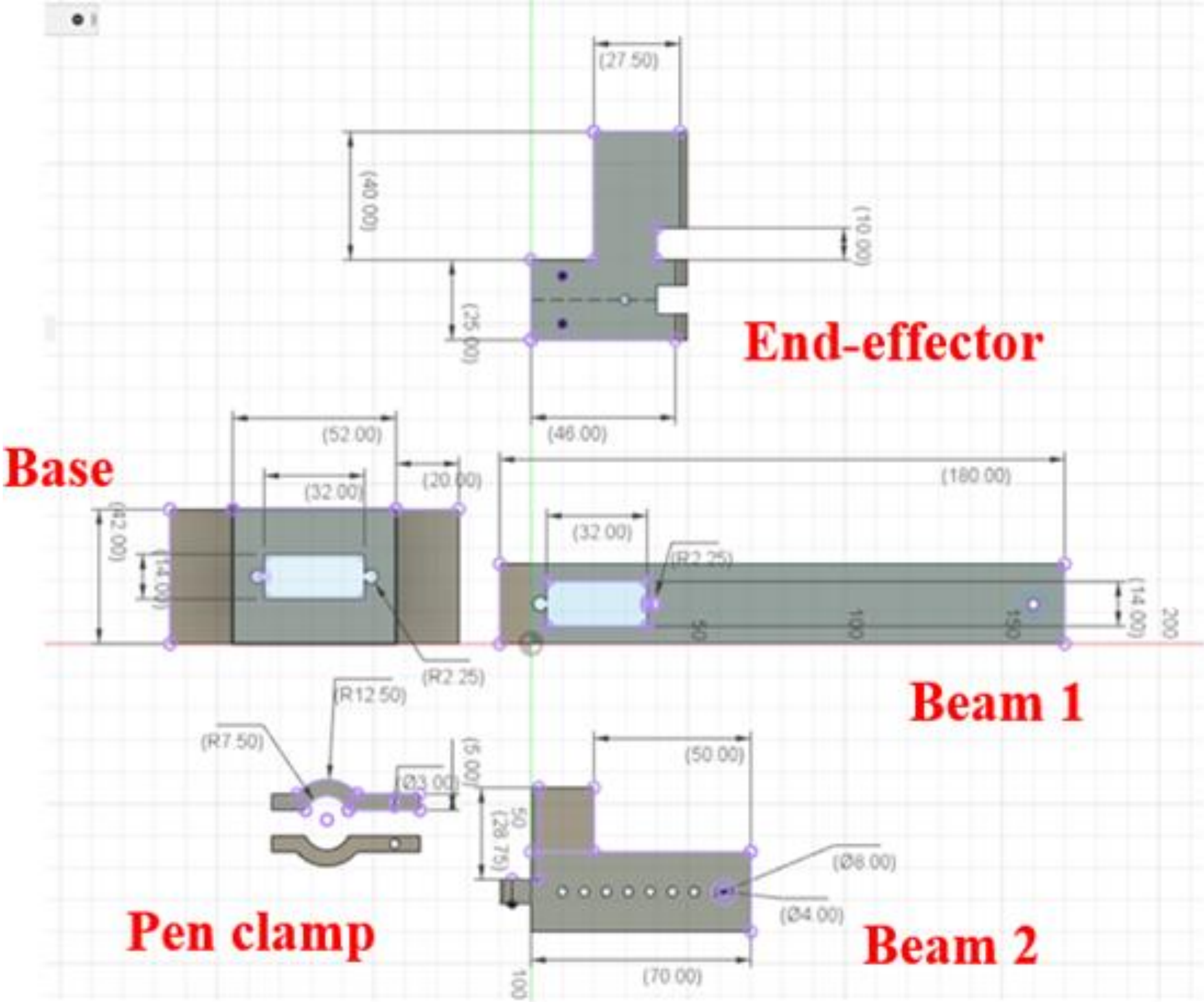


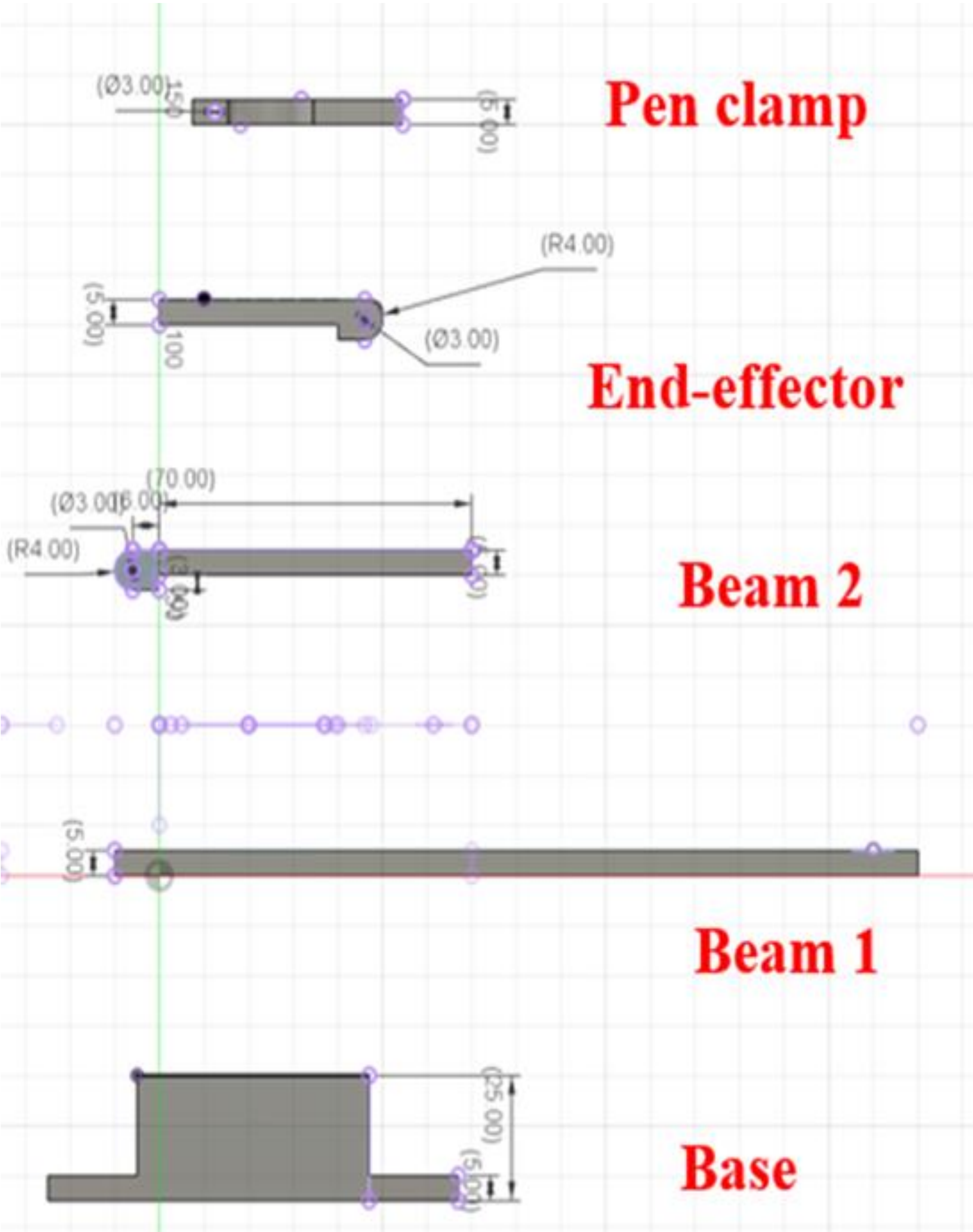
Dimensions & Specifications
A (mm) : 32
B (mm) : 23
C (mm) : 28.5
D (mm) : 12
E (mm) : 32
F (mm) : 19.5
Speed (sec) : 0.1
Torque (kg-cm) : 2.5
Weight (g) : 14.7
Voltage : 4.8 - 6

Position "0" (1.5 ms pulse) is middle, "90" (~2ms pulse) is middle, is all the way to the right, "-90" (~1ms pulse) is all the way to the left.



Parameters of mechanical elements




Pen clamp
(Ø3.00)
(5.00)
(R4.00)
(5.00)
(Ø3.00)
End-effector
(70.00)
(Ø3.00)
(6.00)
(R4.00)
Beam 2
(5.00)
Beam 1
(25.00)
(5.00)
Base

Code

Code for G-code transfer(python)

```python
from __future__ import absolute_import, division, print_function, unicode_literals
import os
import sys
import cv2
import termcolor
import ast
import copy
import argparse


class ImageToGcode():
    def __init__(self,
                 img,
                 verbose=False):
        self.img = cv2.imread(img,0)
        self.output = ""
        self.outFile = os.path.splitext(os.path.abspath(img))[0]+".gco"
        self.spread = 3.175
        self.nozzles = 12
        self.increment = self.spread/self.nozzles
        self.printArea = [200, 200]
        self.feedrate = 1000
        self.black = 255
        self.offsets = [0, 0]
        self.debug_to_terminal()
        self.make_gcode()

    def make_gcode(self):
        self.output = "M106" #Start Fan
        nozzleFirings = [0 for x in range(0, self.img.shape[1])]
        nozzleFirings = [copy.copy(nozzleFirings) for x in range(0, 4)]
        scan = range(0, self.img.shape[0])
        scan.reverse()
        for y in scan:
            for x in range(0, self.img.shape[1]):
                color = self.img[y,x]
                if color == self.black:
                    nozzleFirings[3][x] += 1 << y % self.nozzles
                else:
                    pass
            if y % 12 == 0 and y > 0:
                for headNumber, headVals in enumerate(nozzleFirings):
                    for column, firingVal in enumerate(headVals):
                        if firingVal:
                            #print(headNumber)
                            currentOffset = self.offsets
                            self.output += "G1 X"+str(self.increment*column-currentOffset[0])+" Y"+str(y/12*self.spread-currentOffset[1])+" F"+str(self.feedrate)+"\n"
                            self.output += "M400\n"
                            self.output += "M700 P"+str(headNumber)+" S"+str(firingVal)+"\n"
                            #print (self.output)
                nozzleFirings = [0 for x in range(0, self.img.shape[1])]
                nozzleFirings = [copy.copy(nozzleFirings) for x in range(0, 4)]
        f = open(self.outFile, 'w')
        f.write(self.output)
        f.close()

    def debug_to_terminal(self):
        print("Rows: "+str(self.img.shape[0]))
        print("Cols: "+str(self.img.shape[1]))
        print("Spread: "+str(self.spread)+"mm")
        print("Nozzles: "+str(self.nozzles))
        print("Print Area: "+str(self.printArea)+"mm")
        rowStr = ""
        for y in range(0, self.img.shape[0]):
            rowStr = ""
            for x in range(0, self.img.shape[1]):
                color = self.img[y, x]
                if color == self.black:
                    rowStr += " "
```

```
                else:
                    rowStr += termcolor.colored(" ", 'white', 'on_white')
            print (rowStr)


if __name__ == "__main__":
    #Setup Command line arguments
    parser = argparse.ArgumentParser(prog="image-to-gcode.py",
                                     usage="%(prog)s [options] input...",
                                     description="Convert bitmaps to gcode."
                                     )

    parser.add_argument("input",
                        help="input file, defaults to stdin"
                        )
    parser.add_argument('--version',
                        action='version',
                        version="%(prog)s 0.0.1-dev"
                        )


    #Always output help by default
    if len(sys.argv) == 1:
        parser.print_help()
        sys.exit(0)  # Exit after help display

    args = parser.parse_args()

    imageProcessor = ImageToGcode(img=args.input,
                                  )
```

Code for control algorithm(Matlab)

```
clear all;
 % port at which your arduino is connected
 port = 'COM9';
% model arduino board
 board = 'Mega2560';
% creating arduino object with servo librarytheta2
 arduino_board = arduino(port, board, 'Libraries', 'Servo');
% creating servo motor object
    servo_motor1 = servo(arduino_board, 'D9');
    servo_motor2 = servo(arduino_board, 'D7');
    servo_motor3 = servo(arduino_board, 'D6');
    theta3 = 0.63;
    theta4 = 0.4;
    L1=145;
   L2=125;
   fid=fopen('dola1.nc');

while ~feof(fid)
 str = fgetl(fid);
 if ((str(1)=='G')&&(str(2)=='1'))||((str(1)=='G')&&(str(2)=='0')) % control rotation angle of HS81 servo motors
   data=regexp(str,'\d*\.?\d*','match');
   a=data{2};
   b=data{3};
   X=str2double(a);
   Y=str2double(b);
     disp([X,Y]);
   plot(X,Y,'.');
   axis([10 200 10 200]);
   hold on;
   theta2 = (acos((X^2+Y^2-L1^2-L2^2)/(2*L1*L2)))*(1/180)*(180/pi);
   theta1 = (atan(Y/X)-atan((L2*sin(theta2))/(L1+L2*cos(theta2))))*(1/180)*(180/pi);
   writePosition(servo_motor1, theta1);
   writePosition(servo_motor2, theta2);
   pause(0.01);
```

```
 end
 if (str(1)=='M')&&(str(2)=='3') % drop the pen down

     writePosition(servo_motor3, theta3);
     pause(0.5);
 end
 if ((str(1)=='M')&&(str(2)=='5'))||((str(1)=='G')&&(str(2)=='9')) % lift pen up
     writePosition(servo_motor3, theta4);
     pause(0.5);
 end
end
```