Articulated Miniature Robotic Arm

Sam Ferguson

1 Abstract

This project set out to create a desk-sized miniature robotic arm capable of moving through all three planes to pick up small objects and fluidly move them. After five months of work and a complete redesign, the finished project worked through four degrees of freedom and could pick up and move 400 gram objects through 9.26 cubic feet.

2 Introduction

The idea for this project initially came from industrial articulated robotic arms such as those used in the car manufacturing industry. There is one on display in the Ford dealership located on El Camino in Redwood City, which seemed like an interesting way to investigate robotics. Eventually the project was downsized into the idea of creating a miniature robot small enough to put on a desk. After further research, I found a sample of similar technology called uArm on Kickstarter [1]. The initial goal was an arm with three main rotational joints and a gear-powered claw to pick up and move small objects. In later stages of the project, the arm would also have attachments that can replace the claw such as an electromagnet. Unfortunately, there is little application for this sort of down-sized robotic arm. However, getting a small-scale robotic arm to work is an important first step in creating a full-scale industrial robot.

3 Project Goals: Why Make A Robot?

This project is interesting primarily because it is an in-depth and hands-on way to study robotics. Robotics is an interesting and extensive field with many commercial applications ranging from assembly lines

---

This paper was written for Dr. James Dann's Applied Science Research class in the spring of 2015.

to prosthetics, and creating a miniature articulated robotic arm is a good place to start. The robot will not have the commercial precision, strength, or size of true industrial robots. However, it will exemplify many of the challenges of creating and operating industrial robots on a smaller scale. This project is also an effective way to learn about operating micro controllers and servo motors as well as the mechanical operation of joints, torque, and leverage.

One of the project's primary goals is to have an end product that is as versatile as possible. This overarching goal applies to many different aspects of the robot. Particularly with the design, this goal means the robot will need to operate through all three planes and rotate 360 degrees. The robot's claw should be able to pick up objects of a wide range of materials and sizes. This goal contributes to why the robot should have different attachments with which to replace the claw. Different attachments such as electromagnets or vacuum pumps will increase the robot's versatility and the range of objects it can pick up. While versatility is important, another focuses of the project is to simplify the controlling of the robot. This goal will ultimately be achieved through a well-designed system of levers and a user-friendly controlling interface. The controlling interface will implement kinematics to smooth out its movements.

Aside from these physical goals for the project, the two main aims of the project center around designing the product and creating the controller. The design portion consists of creating the CAD drawing and laser cutting the parts needed. The controller development is a large part of why this project is so enticing. In the first semester I worked intensively with CAD software, laser cutting, and 3D printing. However, I did not explore how to use Arduino micro controllers. Through this project I hope to gain knowledge of how a micro controller functions as the "brain" of a robotic device.

4 History

There are several different types of robots and robotic arms. The type that this project aims to create is referred to as an articulated robot. An articulated robot is a robotic arm that has at least three rotary joints and degrees of freedom. The reason an articulated robot is ideal for this project is that it can perform variety of functions. Articulated robots are commonly seen in many different commercial assembly lines as well as working in labs. In terms of design and function, what sets articulated robots apart from other robots is their rotary joints. Because articulated robots have at least three rotary joints, they are very flexible and have a large range of motion all housed within a small footprint [2]. While articulated robots are the most common, there are other types of robots that perform similar functions such as Delta, SCARA, and Cartesian robots.

Delta robotic arms are known as "pick and place robots" because they are widely commercially known for their ability to pick up and move objects, particularly small ones, at high speed. This is due to their special construction. Unlike the other types of robots, Delta robots are constructed with their actuators and the bulk of their design located above the workspace. This means that the arms can be constructed from lightweight material, allowing very rapid movement across the workspace due to minimal inertia. This type of movement is highly desirable for many commercial applications. One such example is in some 3D printers, which have implemented delta robot technology to print objects quickly. However, Delta robots lack the flexibility of articulated robots, as they are limited to working within the constraint of their frame and often do not have rotational ability. For these reasons, I decided not to focus the attention of this project on a Delta robot.

Figure 1: *A Sample Delta Industrial robot [3].*

SCARA robots look and function closer to how articulated robots do. SCARA (Selective Compliance Assembly Robotic Arm) robots use two parallel revolution joints to move flexibly through the x and z planes. However, SCARA robots' movements are relatively limited in the y plane. In the y plane, they move using a linear joint to pick up and place objects. This linear joint is what distinguishes SCARA robots. These joints often operate at high speeds, move straight up and down, and are capable of rotating quickly. This makes SCARA robots ideal for jobs such as placing screws and bolts, as well as some high-speed pick and place jobs. However SCARA robots have some limitations. Because they use so many parallel revolution joints, the arm functions as a long lever and makes them an ineffective way to transport heavy objects. Furthermore, these joints limit the height of objects that can be picked up. This was the main reason that this project stayed away from constructing a SCARA robot. While in commercial use the speed and precision of SCARAs can be advantageous, for the purpose of this project, they simply don't have the versatility that an articulated robot presents [5].

Figure 2: *A Sample SCARA Industrial robot [4].*

Cartesian robots are a third type of robot that function with pick and place ability similar to articulated robots. Often referred to as Gantry robots, Cartesian robots are simple and precise industrial robots. They function with three perpendicularly oriented linear axes of freedom. This means that they too operate freely through the x, y, and z planes. However, there are some large cons that make Cartesians less than ideal for this project. Similar to Delta robots, they operate within their own frame. This limits the range of locations from which objects can be picked up, particularly the height from which objects can be picked up and placed. Also disadvantageous is the fact that the footprint of their frame is often so large. In some situations this is a non-issue, but this was not what this project was seeking. So while they are cheap and more simple, their lack of versatility and large frame make an articulated robot more appropriate for this project [6].

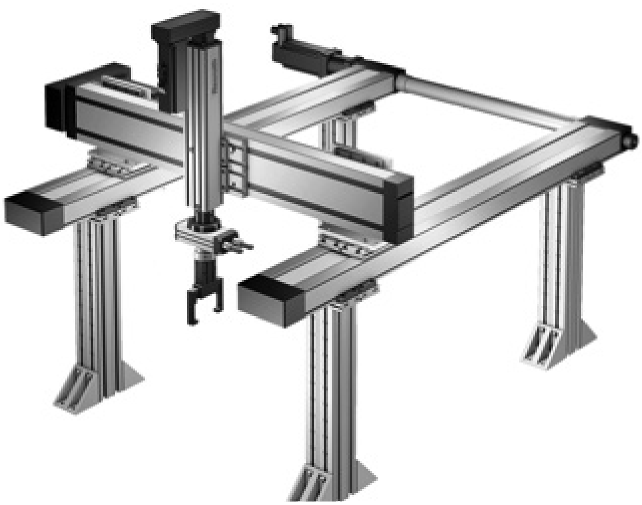

Figure 3: *A Sample Cartesian Industrial robot [6].*

5 Design

A majority of the time in the first and even second phases of this project was spent optimizing the design of the robotic arm. In the initial phases of the project, the 1.0 design pictured in Figure 4 was employed. This design was quite simple, featuring a rotating joint at the base and two main pivot joints on the arm. This basic design provided three degrees of freedom and extended a full 360 degrees around the base. At the end of the arm there was a gear and a servo-powered claw. While the 1.0 design provided the necessary range of motion and seemed simple enough, further work proved that it had unnecessary complications and problems.

When evaluating Version 1.0, one of the apparent issues was the "elbow" joint. The original design would require a large servo fixed onto the elbow. Not only did this look clunky, but it would also add unnecessary weight to the moment arm that the shoulder had to move. The torque of the servo actuating the "shoulder" joint was already being stretched, and this led to a rethinking of the fundamental design of the robot.

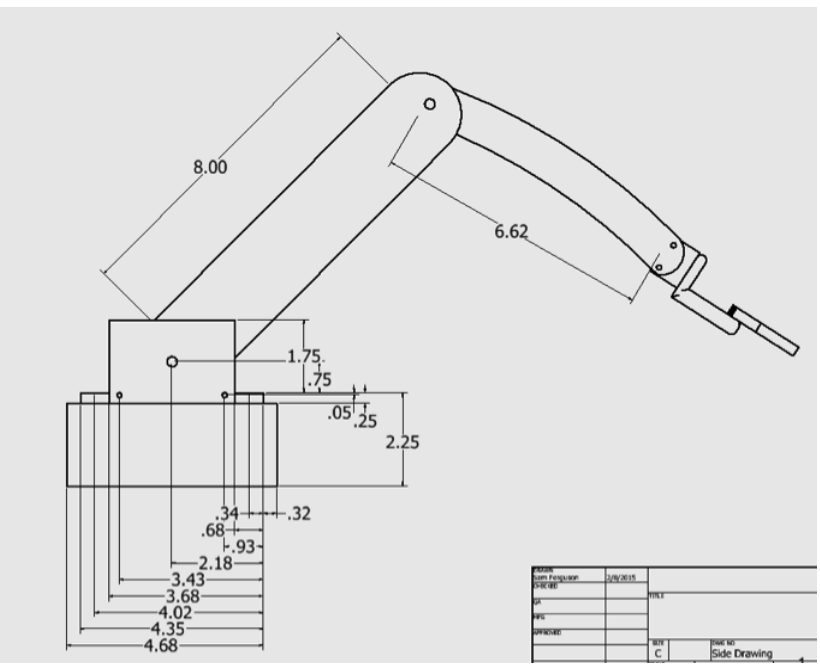

Figure 4: *Side view of CAD drawing of the version 1.0 (dimensions in inches).*

While rethinking the design, one of the main goals of the project kept coming up—simplifying the movement of the robot and making it as fluid as possible. With the elbow servo located at the joint in Version 1.0, not only was the servo adding unnecessary weight but moving each part of the arm was awkward. Another shortcoming of Version 1.0 was the absence of a wrist joint, mainly due to the added weight and controls being an unnecessary complication. However, having no wrist means that the pinchers would often be picking up objects while tilted at unusual angles. The design for Version 2.0 set out to fix all of these problems. What is unique about this side of the arm is that its construction keeps the wrist of the robot parallel to the z plane at all times.

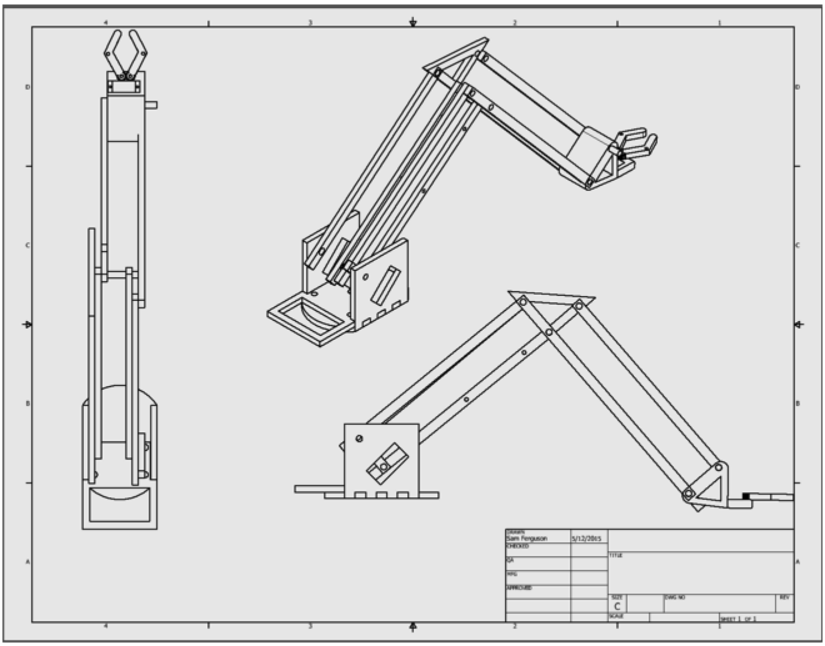

Figure 5: *Different views of Version 2.0 (dimensions in inches).*

As seen in Figure 5, Version 2.0 employs a much more complex design that involves a system of levers. The structure of Version 2.0 has two main sides of the arm which are connected at strategic pivot points. The key to the construction of each side is the geometry of parallelograms. On the left side of the robot, shown in Figure 7, there are two main parallelograms shown: one for the forearm and one for the upper arm. There is also an elbow in the form of an isosceles triangle. When attached to the base, the top support in the upper arm is anchored to a pivot on the base, and the lower support is attached to a servo. By rotating the servo, the entirety of the upper arm moves similarly to the way it would in Version 1.0. However, what makes the left side of the arm so special in Version 2.0 is that the nature of the parallelograms and isosceles triangle at the elbow keep the wrist parallel to the z plane. For example, let's say the upper arm extends outward. In Version 1.0 this would have kept the angle at the elbow constant, pushing the forearm perpendicular to the ground. In Version 2.0, when the lower support of the upper arm is pushed outwards, it also extends the upper arm outwards. However, as it extends the top support in the upper arm pulls back on the top of the elbow, rotating the hypotenuse of the triangle back towards the base of the robot. This in turn pulls back on the top support in the forearm and the top of the wrist. Because of this geometry, the wrist stays level at all times and adds increased control over how the arm picks up objects.

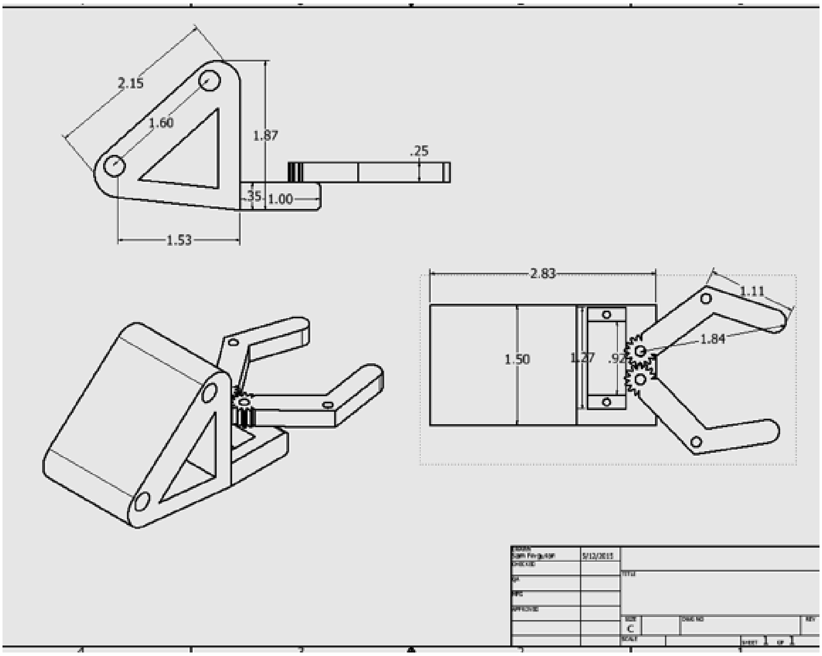

Figure 6: *Different views of the hand with pinchers.*

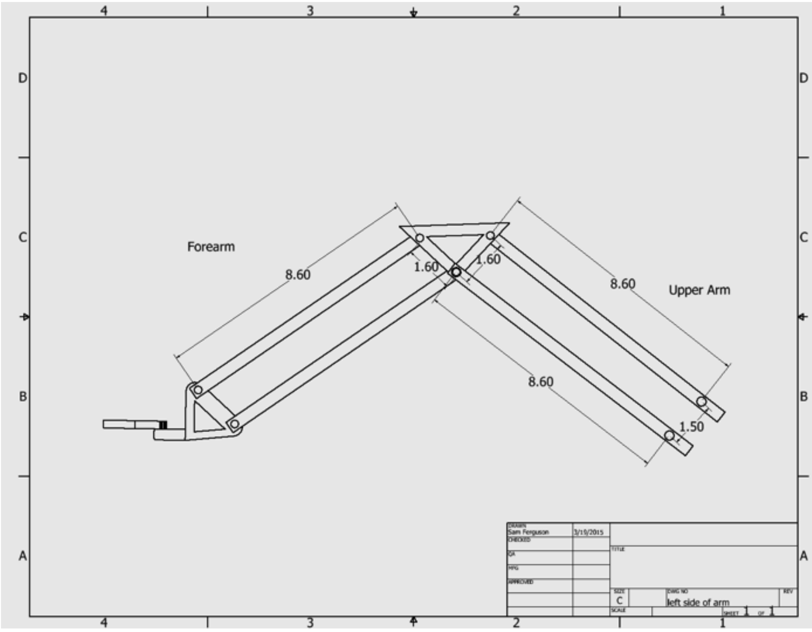

Figure 7: *The left side of the robotic arm (dimensions in inches).*

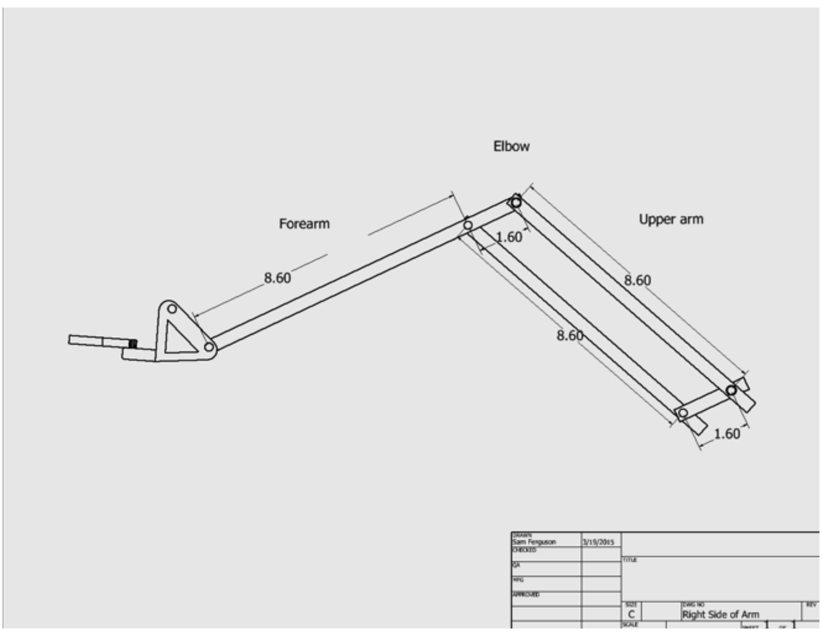

Figure 8: *The right side of the robotic arm (dimensions in inches).*

While the left side of Version 2.0 of the arm keeps the wrist level, the right side of the arm (as shown in Figure 8) is responsible for actuating the elbow joint. Unlike in Version 1.0 which had a servo built into the elbow, Version 2.0 actuates the elbow joint with a servo located at the base of the arm. Before understanding how this works, it is important to recognize that the right side of the arm is connected to the left side of the arm at the wrist and at the bottom pivot in the elbow. The right side of the arm is also pinned to the base of the arm at the bottom pivot in the lower arm. At this pivot, the small connection between the top and bottom supports in the upper arm is anchored to a servo. Because the right side of the arm is connected to the left side at the lower elbow, the lower support in the right arm is held in place by the left side of the arm. This means that when the servo connected to the right side of the arm moves, it pulls or pushes on the top pivot in the elbow. Because the forearm on this side of the arm connects to both pivots in the elbow, the pivot in the lower elbow serves as a fulcrum and moves the forearm up or down.

While the physical design of Version 2.0 of the robotic arm is more complex, it provides a functionally more simple and efficient system. Rather than having two simple rotary joints as Version 1.0 had, Version 2.0 uses a system of levers to create a smoother design. The left side of the robot takes advantage of the geometry of parallelograms to keep the wrist parallel to the z plane at all times, and the right side of the arm uses leverage to actuate the elbow from the base of the arm.

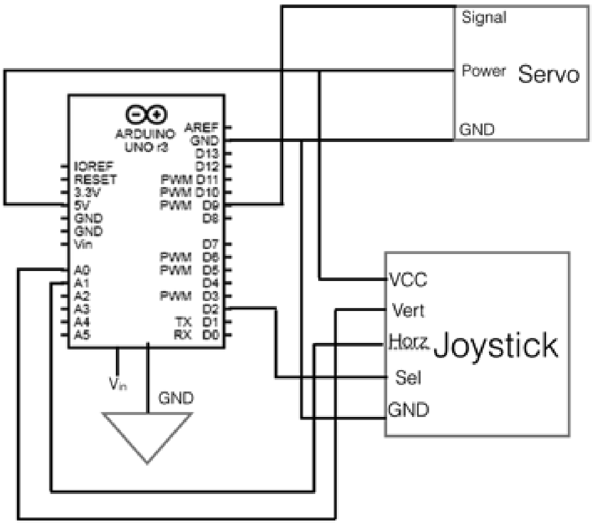

Figure 9: *A sample circuit diagram for a single servo controlled by an Arduino Uno and a joystick. See Appendix for code.*

5.1 Servos, Joysticks & Circuit Diagram

At the onset of this project, the arm was intended to be controlled by using a mouse. However, after research into this method, using a mouse as a control was scrapped and replaced with using joysticks. This decision was made for a variety of reasons, but ultimately it came down to mouse input being overly complicated. Controlling this arm essentially means controlling four servos. Mice are convenient for performing different functions at any point in a single plane. This project works through all three dimensions, making it already a little awkward to use a mouse. Using two joysticks has turned out to work exceptionally well. Because the arm uses exactly four servo motors and each joystick has two potentiometers, the combination of two joysticks has proven ideal for controlling the arm.

The above circuit shows how an example servo is connected. In the final circuit four servos are attached in this manner on Pins 10-7, and a second joystick is added.

6 Theory

6.1 Servos

Servo motors are relied on to move each joint in the robot. Servos are ideal for this project because they control rotation down to precise increments and allow a micro controller to specify the angle at which they should be performed. This precision, as well as the torque and speed of servos, is why they are commonly used in many applications and are an essential feature of robotics.

Part of what makes servos so widely used is that they are so durable. This durability stems from the simplicity of their construction. There are four main parts that make up a servo, all housed within the rectangular shell of the device. The bulk of the device is found in the DC motor which provides the heart of the servo. The DC motor is generally small, low torque, and high RPM. The high frequency of rotations from the DC motor's shaft is converted into torque through the next main part of the servo—the gears. The gears slow down the rotations of the motor shaft and add torque.

Attached to the last gear in the servo is a potentiometer. The potentiometer communicates the servo's output pin angle to a small circuit board. The circuit board, which is connected to a micro controller, then turns the DC motor on or off based on commands sent out by the micro controller.

While all servos have this same basic setup, servos range in torque and rotational ability. The main servo that has been used in the first two phases of this project is the SpringRC SM-S4303R Continuous Rotation Servo. These servos are contained within a 41.3 mm x 20.7 mm x 40.2 mm box and have a mass of 41 g. These servos also feature .500 N-m of torque, and can continually rotate.

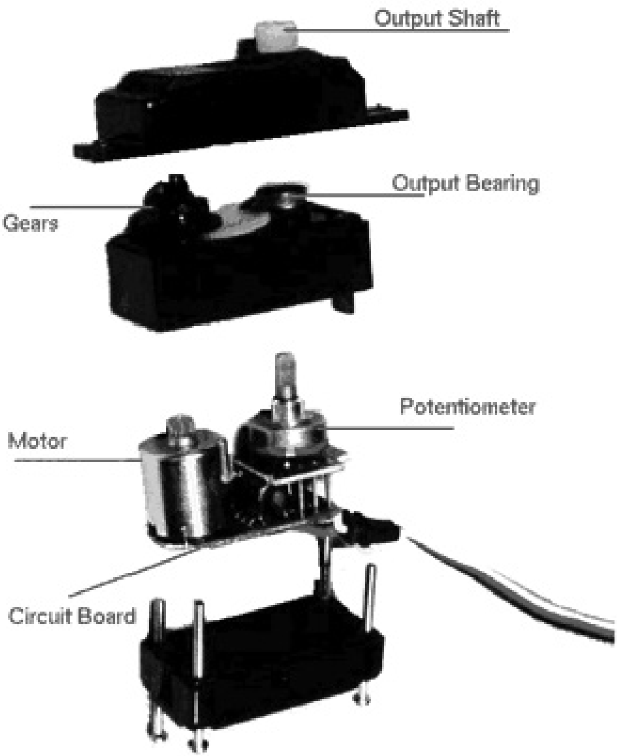

Figure 10: *The inside of a servo motor [7].*

6.2 Joysticks

While servos are what physically operate the joints, this project employs joysticks to take in user inputs for how the servos should be oriented. The joysticks in use have a relatively simple structure. There are three main devices employed to send user input to the micro controller: a small switch, a horizontal potentiometer, and a vertical potentiometer. The switch works quite simply. When the joystick is pushed down, a small lever is pushed into a button. When the button is pressed, a circuit is closed, and the micro controller receives a signal voltage in its respective pin.

The potentiometers are slightly more complicated. Potentiometers are made of a large arc of resistive material (often lead or graphite), separated by a wiper. The wiper essentially separates the resistive material into two variable resistors that are in series. Because of Ohm's Law, the larger the amount of resistive material between the first pin and the wiper, the greater the voltage drop. This size of this voltage drop is what changes the output from the potentiometer.

The joysticks in use have two potentiometers aligned on adjacent sides of the base. This leaves one potentiometer to receive horizontal input and the other to receive vertical output. The joystick has a small gimbal mechanism inside it that converts movement in one direction into rotational movement on the input pin of the respective potentiometer. Through this mechanism, the joystick is able to turn user movement over the x and z planes into voltage outputs which it then sends to the micro controller.

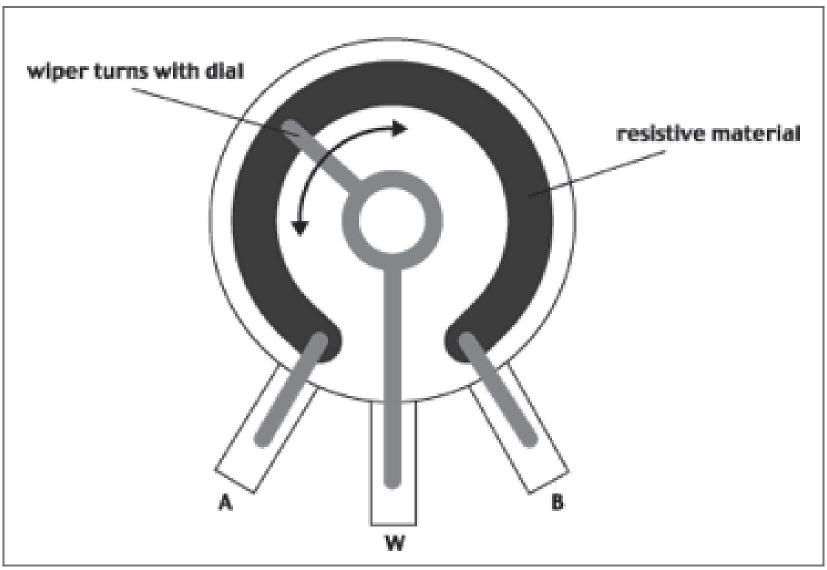

Figure 11: *The inside of a potentiometer [8].*

6.3 Torques & Levers

An important part of this project involves torques, leverage, and mechanical advantage. The entire point of building a robotic arm is to be able to use it to move things, and it is important to know that it actually has the power to lift things. While there are four servos in the robot, the servos located in the hand and wrist have minimal force exerted on them, meaning that they have plenty of torque to carry out their purposes. However, the servos controlling the elbow and shoulder joints are more interesting.

Torque, also known as moment of force, is the force that rotates an object about an axis. The basic equation for calculating the torque of a moment arm is *Torque = Force * Moment Arm*. In this project, the force acting on the arm is gravity, which is given at 9.81m/s^2. The moment arm is considered to be the distance to the center of mass in the arm. Taking this into consideration, torque is strongest when the force is acting perpendicularly to the moment arm, because in this situation, the force is acting over a greater area of the moment arm.

6.4 Calculating the Torque on the Shoulder Joint

Because torque is strongest when it is perpendicular to the moment arm, it is important to find the torque of gravity acting on the shoulder joint when the arm is fully extended and parallel to the ground. In this situation, the length of robotic arm is equal to 18.2 inches. If the arm was a constant weight all the way across, this would make the center of mass 9.1 inches away, but the arm's mass is focused more towards the wrist, making the moment arm 14.5 inches long, or .368 meters. An object in the hand causes the most torque on the robotic arm. This both shifts the center of gravity out and increases the force of gravity acting on the arm. However, the following calculation assumes the robotic arm is not holding anything in order to see if there is enough torque to move the shoulder joint alone.

With the robotic arm not holding anything, the mass of the arm is 269 g and its weight 2.64 Newtons. This means that the torque acting on the shoulder joint = 2.64 Newtons * .368 Meters = .972 Newton-meters. This is appropriate because the Hitec 35645S HS-5645MG Digital Hi Torque Metal Gear BB servo has sufficient torque of 1.19N-m. This difference means that when fully extended, the arm can lift objects with a mass up to 329 grams.

6.5 Calculating the Torque on the Forearm Joint

Unlike the upper arm, the forearm joint is a lever with the fulcrum located closer to the middle of the support. This slightly changes the way to calculate the torque acting on the pivot. In this instance the equation *Length1 * Force1 = Length2 * Force2* is used, where Force and Length 1 are the main part of the forearm, and Length and Force 2 refer to the part of the forearm that extends past the elbow pivot.

Because the elbow is actuated from the base of the arm, there is also torque exerted on the top support of the upper arm. When the arm is fully extended, the gravitational torque acting on the arm is at its greatest. At this point, the forearm has a weight of .628 N, and extends out .218 m. On the other side of the pivot the lever extends .041 m. Therefore, the equation's values are as follows:

Length1 = .218m, Force 1 = .628N, Length 2 = .041m, and Force 2 = ?

yielding the equation:

*.218m * .628N = .041m * F2*

Solving for F2, the force needed to bring the lever into static equilibrium is 3.34N.

This force of 3.34N is needed to pull down on the top of the elbow. Because the arm is actuated from the base, there is an additional lever that connects the servo and the elbow to provide this force. This system of levers works at an angle. This means that the torque equation needs to take into account the angle at which the gravitational force is applied to the arm. The equation *sin(x)*r*F = torque* is used, where x is the angle of elevation of the top support in the upper arm. The force needed to move the arm was calculated above to = 3.34 Newtons. The robotic arm has a radius of .218 m but the center of mass is closer to the base, providing a moment arm of Length 1.4 m. As mentioned in the servo section above, the servo provides a torque of 1.19 N-m. This information yields the equation:

$$1.19\text{N-m} = 1.4\text{m} * \sin(x)*3.34\text{N}$$

Isolating for x, the equation indicates that x = 014rad. This means that the elbow will have the necessary power to operate all the way down to 6.1 degrees from the fully extended position.

7 Results

In its final form, the arm has proven to be effective at picking up and moving small objects. The above calculations would indicate that the arm could lift a mass of 329 grams when fully extended. Testing has shown that the arm is capable of exceeding this calculation and lifting 400 g. Meanwhile, testing closer in to the base of the arm has shown that the arm can lift significantly larger amounts. The main limiting factor has been gripping larger weights, but testing has shown that the arm has the power to lift objects over 600 g.

While the force and lifting ability of the robot are impressive, what makes these numbers even more striking is that the arm did not have to sacrifice any of its reach. When fully extended, the arm can reach out an impressive 20.6 inches from the base to the end of the pinchers. Because of the rotational base, the arm can cover an area of 9.26 square feet. The arm also features four degrees of freedom.

The long reach and lifting ability of the arm did contribute to one unforeseen problem. When fully extended, there is a large force pulling on the base of the arm. This force initially caused the rotating servo at the base of the arm to lean several degrees. Not only did this cause the hand to no longer be level with the ground, but it also caused the servo at the base of the arm to move more sporadically. Instead of responding smoothly to joystick commands, the base would jerk, and when it did move it would move more than desired. In order to fix this problem, a 500g counterweight was added to the back of the base. This counterweight helped keep the full arm more sturdy and balanced.

8 Conclusion

At the outset of this project the goal was to make a miniature robotic arm capable of picking up and moving small objects. This goal was fully achieved; the arm is quite functional and moves effectively in response to user joystick commands. There were, however, other nuances of the initial goal that were not achieved due to a lack of time. The first shortcoming was that there were no other attachments created for the end effector of the arm. The pincher works to pick up many objects, but initially it was intended that there would also be a vacuum pump and electromagnet added which would add to the number of objects that could be moved. The other main shortcoming was that there was not enough time to make the arm move more fluidly. This is the more troubling of the two shortcomings. The arm works and can pick up and move objects, but it is not easy to use. Getting the arm into a position where it can actually pick up a small object is difficult.

Despite these shortcomings, the project still was a worthy endeavor. Building the robotic arm helped reveal many aspects of robotics, design, and electronics. One aspect that seems to have particularly succeeded is the redesign of the main arm from Version 1.0 to 2.0. This upgrade was important in helping optimize the flow and usability of the arm. However, it required scrapping most of the work done through the first stage and starting over. In the end this cost paid off in increased functionality of the arm. ●

9 Appendix

Arduino Code

```
#include <Servo.h>
Servo myservo;
Servo myservo2;
Servo myservo3;
Servo myservo4;

const int VERT = 0; // analog
const int HORIZ = 1; // analog
const int SEL = 2; // digital

const int VERT2 = 3;
const int HORIZ2 = 4;
const int SEL2 = 5;

int pos = 90;
int servoVal = 0;

int pos2 = 90;
int servoVal2 = 0;

int pos3 = 90;
int servoVal3 = 0;

int pos4 = 45;
int servoVal4 = 0;
//int speed = 1;


// Also connect the joystick VCC to Arduino 5V, and joystick GND to
Arduino GND.

// This sketch outputs serial data at 9600 baud (open Serial Monitor to
view).
```

```
void setup()
{
 // make the SEL line an input
 pinMode(SEL,INPUT);
 // turn on the pull-up resistor for the SEL line (see http://arduino.cc/
en/Tutorial/DigitalPins)
 digitalWrite(SEL,HIGH);

 // set up serial port for output
 Serial.begin(9600);

 //set up the servo
 myservo.attach(9);
 myservo2.attach(10);
 myservo3.attach(8);
 myservo4.attach(7);
}

void loop()
{
//  int delay = 10;
 int vertical, horizontal, select;
 int vertical2, horizontal2, select2;


 // read all values from the joystick

 vertical = analogRead(VERT); // will be 0-1023
 horizontal = analogRead(HORIZ); // will be 0-1023
  select = digitalRead(SEL); // will be HIGH (1) if not pressed, and
LOW (0) if pressed


 vertical2 = analogRead(VERT2);
 horizontal2 = analogRead(HORIZ2);
 select2 = digitalRead(SEL2);
 // print out the values
```

```
// Serial.print("vertical: ");
// Serial.print(vertical,DEC);
// Serial.print(" horizontal: ");
// Serial.print(horizontal,DEC);
// Serial.print(" select: ");
// if(select == HIGH)
//   Serial.println("not pressed");
// else
//   Serial.println("PRESSED!");

 if(select == HIGH)
 {
  servoVal = map(horizontal, 0, 1023, 0, 179);    // scale it to use it with
the servo (value between 0 and 180)
   servoVal2 = map(vertical, 0, 1023, 0, 179);
 }

   servoVal3 = map(horizontal2, 0, 1023, 0, 179);
   servoVal4 = map(vertical2, 0, 1023, 0, 179);


/////strong servo
if (servoVal > 110)
 {
  if (pos < 180)
  {
   pos = pos + 1;
   delay(15);
  }
 }
 else if(servoVal < 70)
 {
  if (pos > 0)
  {
   pos = pos - 1;
   delay(15);
   }
 }
```

```
 Serial.print("servoVal1: ");
Serial.println(servoVal);

////strong servo
if (servoVal2 > 110)
{
  if (pos2 < 180)
  {
    pos2 = pos2 + 1;
    delay(15);
  }
}
else if(servoVal2 < 70)
{
  if (pos2 > 0)
  {
    pos2 = pos2 - 1;
    delay(15);
   }
}
Serial.print("servoVal2: ");
Serial.println(servoVal2);
Serial.println(pos2);

 //base rotating servo
//    Serial.println(servoVal3);

  if (servoVal3 > 100)
  {
    pos3 = pos3 + 1;
    delay(40);
  }
  else if(servoVal3 < 80)
  {
    pos3 = pos3 - 1;
    delay(40);
  }
```

```
    else
    {
      pos3 = 90;
//    delay(5);
    }
     Serial.print("servoVal3: ");
  Serial.println(servoVal3);



    //pincher micro servo
if (servoVal4 > 110)
  {
    if (pos4 < 90)
    {
      pos4 = pos4 + 1;
      delay(10);
    }
  }
  else if(servoVal4 >10)
  {
    if (pos4 > 0)
    {
      pos4 = pos4 - 1;
      delay(10);
     }
  }


    myservo2.write(pos2);
    myservo.write(pos);
    myservo3.write(pos3);
    myservo4.write(pos4);

  }
```

10 Notes

1. "uArm: Put a Miniature Industrial Robot Arm on Your Desk," Kickstarter, accessed February 9, 2015, https://www.kickstarter.com/projects/ufactory/uarm-put-a-miniature-industrial-robot-arm-on-your.

2. "What is an Articulated Robot?," Wise Geek, accessed February 8, 2015, http://www.wisegeek.com/what-is-an-articulated-robot.htm.

3. Delta Robot, photograph, Medical Design Technology, accessed February 8, 2015, http://www.mdtmag.com/product-releases/2013/08/strong-improved-delta-robot-handles-100-cycles-minute

4. SCARA Robot, photograph, Softpedia, accessed February 8, 2015, http://linux.softpedia.com/get/Science/SCARA-robot-100789.shtml.

5. "What is a SCARA Robot,?" Wise Geek, accessed February 8, 2015, http://www.wisegeek.com/what-is-a-scara-robot.htm.

6. "Cartesian Robots," All On Robots, accessed February 8, 2015, http://www.allonrobots.com/cartesian-robots.html.

7. *Inside of a Servo*, photograph, Society of Robots, accessed March 20, 2015, http://www.societyofrobots.com/robotforum/index.php?topic=4594.0.

8. "Robot Statics," Society of Robots, accessed March 20, 2015, http://www.societyofrobots.com/mechanics_statics.shtml.

11 Bibliography

Bouchard, Samuel. "What are the Different Types of Industrial Robots?" *RobotIQ* (blog). Entry posted March 13, 2013. Accessed February 8, 2015. http://blog.robotiq.com/bid/63528/What-are-the-different-types-of-industrial-robots.

Cartesian Robot. Photograph. Machine Design. Accessed February 8, 2015. http://machinedesign.com/motion-control/difference-between-cartesian-six-axis-and-scara-robots.

"Cartesian Robots." All On Robots. Accessed February 8, 2015. http://www.allonrobots.com/cartesian-robots.html.

Delta Robot. Photograph. Medical Design Technology. Accessed February 8, 2015. http://www.mdtmag.com/product-releases/2013/08/strong-improved-delta-robot-handles-100-cycles-minute.

Fedderson, Jeff. "Inside the Potentiometer." Accessed March 20, 2015. http://fddrsn.net/pcomp/examples/potentiometers.html.

Inside of a Servo. Photograph. Society of Robots. Accessed March 20, 2015. http://www.societyofrobots.com/robotforum/index.php?topic=4594.0.

MeArm. Photograph. Thingiverse. January 11, 2014. Accessed February 8, 2015. http://www.thingiverse.com/thing:360108.

"Robot Statics." Society of Robots. Accessed March 20, 2015. http://www.societyofrobots.com/mechanics_statics.shtml.

SCARA Robot. Photograph. Softpedia. Accessed February 8, 2015. http://linux.softpedia.com/get/Science/SCARA-robot-100789.shtml.

"SpringRC SM-S4303R Continuous Rotation Servo." Polalu. Accessed March 20, 2015. https://www.pololu.com/product/1248.

"uArm: Put a Miniature Industrial Robot Arm on Your Desk." Kickstarter. Accessed February 9, 2015. https://www.kickstarter.com/projects/ufactory/uarm-put-a-miniature-industrial-robot-arm-on-your.

Vaughn, Richard. "The Difference between Cartesian, Six-axis, and SCARA Robots." Machine Design. Accessed February 8, 2015. http://machinedesign.com/motion-control/difference-between-cartesian-six-axis-and-scara-robots.

"What is an Articulated Robot?" Wise Geek. Accessed February 8, 2015. http://www.wisegeek.com/what-is-an-articulated-robot.htm.

"What is a SCARA Robot?" Wise Geek. Accessed February 8, 2015. http://www.wisegeek.com/what-is-a-scara-robot.htm.