UNITECH 2017 GABROVO
INTERNATIONAL SCIENTIFIC CONFERENCE
17-18 November 2017, GABROVO

SMART CARTESIAN ROBOTIC ARM

Assoc. Prof. Hilmi KUSCU
Trakya University, Mechanical Engineering Department, Edirne/ Turkey

Kaan Ertugrul
Trakya University, Mechanical Engineering Department, Edirne/ Turkey

Abstract
Nowadays, robotic arms became an essential part of industry. They are so useful for too many fields and helping human life. In this study, a smart robotic arm which is cartesian type is examined with its missions and logical working. It has motions based on cartesian coordinates (three axis) which are provided by step motors, magnet gripper and programmed with Arduino microcontroller. The robotic arm scans a certain area and find locations of five parts which has variable lengths and finally makes descending or ascending sort based on their lengths.

Keywords: Robotic, arm, smart, cartesian, Arduino

INTRODUCTION

Day by day, importance of robotic industry is increasing fast also too many robotic applications are being made throughout the world. Robotic applications consist of mechanical and electrical design and program which constitutes working logic. Robotic arm field is one of most common robotic application. Today, robotic arms are being used for welding, painting, assembling, carriage, sorting, etc. In this study, a smart robotic arm which is cartesian type is examined with its missions and logical working.

EXPOSITION

The cartesian robotic arm which is designed and run in this study has three axis linear motion that three step motors caused with TB6560 drivers. TB6560 drivers makes system work with low heat and get needed steps. Robotic arm has a screen that shows situation of robotic arm like "scanning, resetting, a part has been found" also parts lengths in millimeter when it found a part. At the bottom of the screen, there is a button which causes to start robotic arm working. First, robotic arm starts scanning to find locations of parts that placed randomly on robotic arm plane. While scanning, if robotic arm finds a part it stops and measures length of part and store locations (two coordinates) and length of the part and continues scanning. When scanning is done, robotic arm asks to user for descending sort or ascending sort. According to answer, robotic arm arrays all parts on its mind based on bubble sort algorithm.

```
void ascendingsort(){
for(int g=0; g<4; g++){
if(L[0]>L[1]){
  int value=L[0];
  L[0]=L[1];
  L[1]=value;
}
else{
}
if(L[1]>L[2]){
  int value=L[1];
  L[1]=L[2];
  L[2]=value;
}
else{
}
if(L[2]>L[3]){
  int value=L[2];
  L[2]=L[3];
```

```
 L[3]=value;
}
else{
}
}
}
```

All information such as length and coordinates are stored in arrays. After ascending sort, the longest part becomes a first element of length array. Thus, robotic arm remembers coordinates of first element of length array. Analog displacement sensor has been used for measuring part lengths and to grip parts electromagnet has been used in this study. Inductive sensors are used for every single axis to make robotic arm comes to origin (resetting).

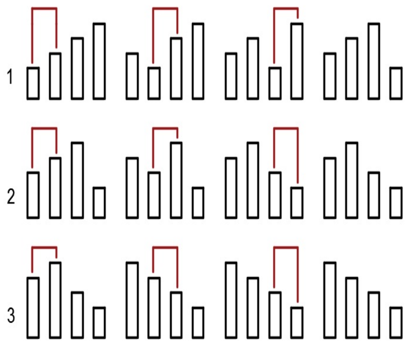

Fig. 1. *Descending sort*

Cartesian robotic arm has three stepper motors for three axes. However, dc motors may be used to achieve this linear motion, stepper motors are more useful than dc motors because of motion accuracy and motion accuracy is one of most important parameter in cartesian robotic arm applications. The robotic arm which is made in this project is examined with three titles: mechanic, hardware and software.

Mechanic

Cartesian robotic arms have a linear motion for each axis. There are variable mechanisms as ball screw and timing belt to supply this linear motion. Also, these mechanisms need to be formed a bearing. There are variable mechanisms as linear guideway & blocks and linear bearings to form a bearing linear motion supplier. In this project, the ball screw mechanism is used to make linear motion for each axis and the guideway & blocks system is used to form a bearing for each axis.

Fig. 2. *Guideway & Blocks*

Fig. 3. *Ball Screw*

In mechanic, one of most important step is forming a bearing shafts. It must be well to make motors work easily and safely and make shafts turn without vibrations. Couplings are so useful to form a bearing shafts. It is not very safe to assemble motor shaft and ball screw with coupling directly because ball screw has no smooth surface. Best way to assemble them is using cylindrical part. Respectively, assembling motor shaft to coupling, coupling to cylindrical part, cylindrical part to ball screw gives us perfect results. To assemble cylindrical part to ball screw, welding is the best choice.

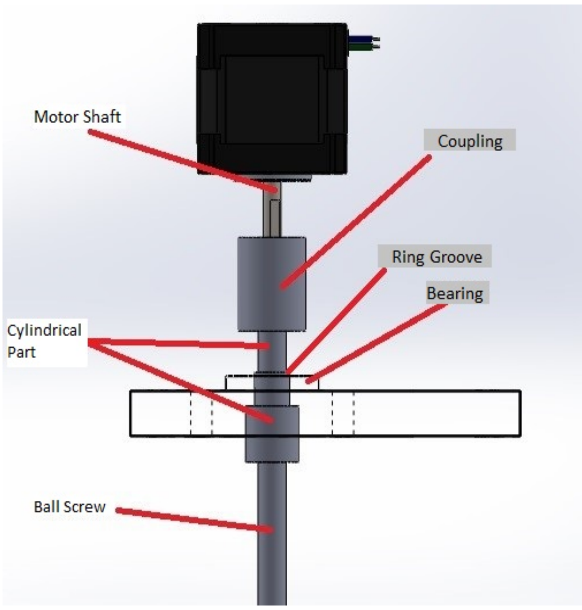

Fig.4. *Form a bearing to shaft.*

Hardware

Hardware components of robotic arm which is examined in this project are given below.

- Stepper Motors
- Stepper Motor Drivers
- Inductive sensors
- Sharp distance sensors
- Electromagnet
- LCD Screen

As mentioned, stepper motors are the best choice for motion accuracy. They change their angular position by steps. The steps can be controlled by sending signals to pairs of motor. Step angle is the quantity that equals to rotary motion of every single signal. Speed of stepper motor can be controlled by changing frequency of signals. Also, rotation direction can be controlled by changing order of signals however Arduino has a stepper library to do all of them easily. Stepper motors can be drove by 1, ½, ¼, 1/8 and 1/16 steps based on using driver. There are two kind of stepper motors as bipolar and unipolar. Bipolar stepper motors have four cables, unipolar stepper motors have six cables however bipolar stepper motors are more common than unipolar stepper motors in robotic applications. In this study, NEMA 17 HS44017 stepper motors are used. Stepper motors cannot be connected to microprocessor directly. Stepper motor driver is needed. There are some important parameters to consider for selection of stepper motor driver. The driver that will be selected, must supply needed current to stepper motors. For an example; if the stepper motor works with 2A, the driver must be able to supply current that equals to 2A or greater than 2A. In this study, TB6560 stepper motor driver is used.

Fig. 5. *TB6560 Stepper Motor Driver*

In this study, LCD screen that has 2 rows and 16 columns is used. Most important thing when using LCD screen, is identifying LCD screen to Arduino. Pin order must be like that "RS, ENABLE, D4, D5, D6, D7. Also, LiquidCrystal library is used for programming LCD screen.

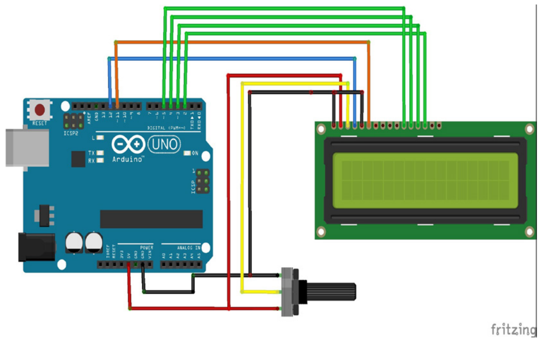

Fig. 6. *LCD Screen Connection*

Inductive sensors sense metal parts in certain range and create a signal. In this study, three inductive sensors are used for three axes to take robotic arm to origin (0,0) and their range distance is 8mm.

Fig. 7. *Inductive Sensor*

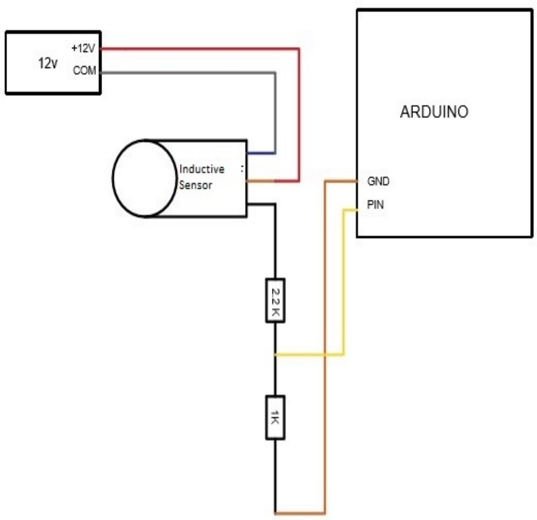

Fig. 8. *Inductive Sensor Connection*

In this study, Sharp distance sensor is used for measure length of parts. The sensor is analog sensor and some formulas are needed to convert the value that gotten to mm or cm. To get more stable values capacitor may be needed.
Electromagnet is used for gripping the parts in this study. The electromagnet that we used in our project can handle 5kg. There is a situation. Electromagnet's needed power supply is 12V so 5V signals from Arduino is not enough to make it work. Then, we use relay for electromagnet. Arduino signal goes to relay then relay allows 12V to go electromagnet.

Software

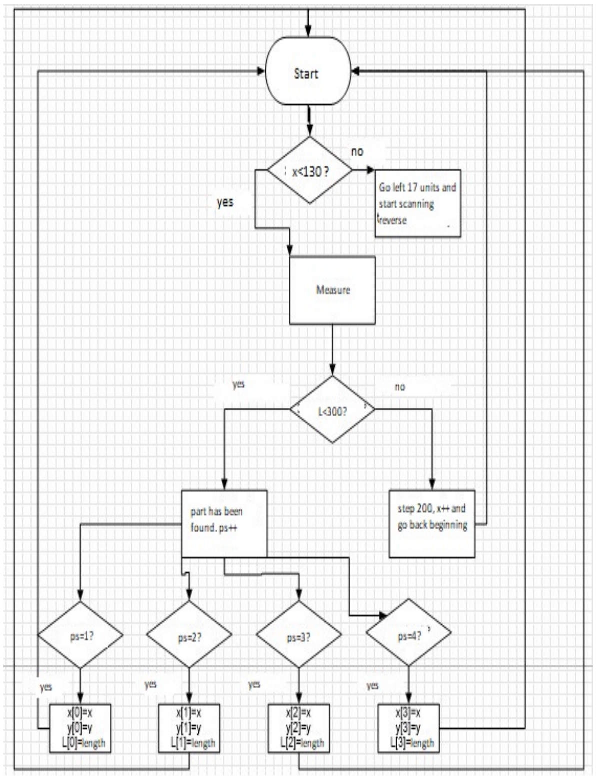

Fig. 9. *Flow Diagram*

Scanning, finding part, storing lengths and locations of parts algorithm is examined. As mentioned, the robotic arm which is made for this project, finds four parts that have variable lengths by scanning certain plane then stores their locations and lengths. After bubble sort algorithm working, the robotic arm starts to make ascending sort or descending sort. First process is scanning process. A coordinates system is developed to store location. After robotic arm comes to origin; x, y and z variables that identified are set to zero then every 200 steps of stepper motor are identified as one-unit motion. One circle ball screw creates 8mm of motion and one circle means 800 steps of stepper motor. So, after calculation, one-unit motion that we identified means 2mm linear motion. After robotic arm came to origin, z axis comes to measuring position and scanning starts.

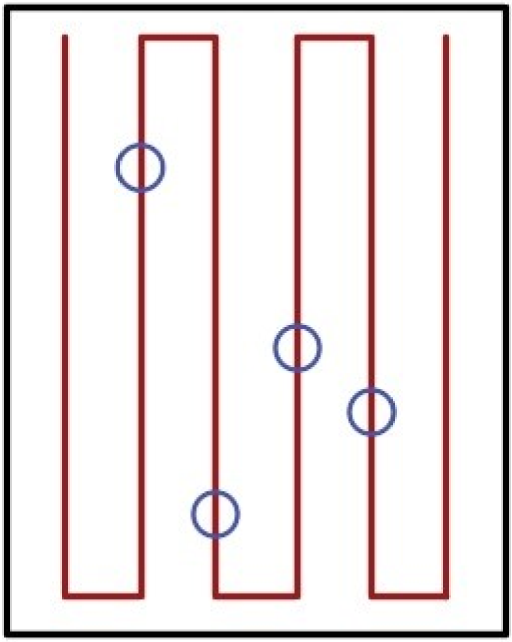

Fig. 10. *Scanning Path*

How can robotic arm understand that it finds part? To do it, we developed finding algorithm. When robotic arm measure to ground, we get 300mm of value. If robotic arm measure less than 300mm, it means there is a part and store the length and coordinates at this moment. Robotic arm applies this algorithm in every one-unit.

Counting units of motion algorithm starts from beginning (origin). When robotic arm is at the origin sets zero all axes and after every unit motion coordinates values increases one.
We store all the information as lengths and coordinates in arrays. After scanning process, bubble sort algorithm works and creates the order of parts which are going to be aligned. After bubble sort worked, first element of array becomes longest or tallest based on ascending sort or descending sort. Robotic arm uses the algorithm given below to go there.

```
if(x<X[2]){
        for(x; x<X[2]; x++){
                motorx.step(200);
                }
else{
        for(x; x>X[2]; x--){
                motorx.step(-200);
                }
```

CONCLUSION

The robotic arm which is made in this project, has been run successfully. All robotic arm parts have been made from aluminum. However, it causes robotic arm to be strong and rigid, at the same time it causes robotic arm to be too heavy so we cannot get high speeds. It caused vibrations when robotic arm stops and starts to move. All parts might be made with 3d printer or plastics. Also, ball screw systems create strong mechanisms but low speed. When we think about three axes, we need power for only z axis because of gravity. Timing belt might be used for x and y axes. Analog Sharp distance sensor could not give stable results however we connected capacitor. It is better to use digital distance sensors which give more stable values. Three stepper motor drivers have been used for three stepper motors. A card that can drive all stepper motors might be used but it might be expensive. Inductive sensors have been used for only take robotic arm to origin and set zero all coordinates. They might be used for security either by using Interrupt function.

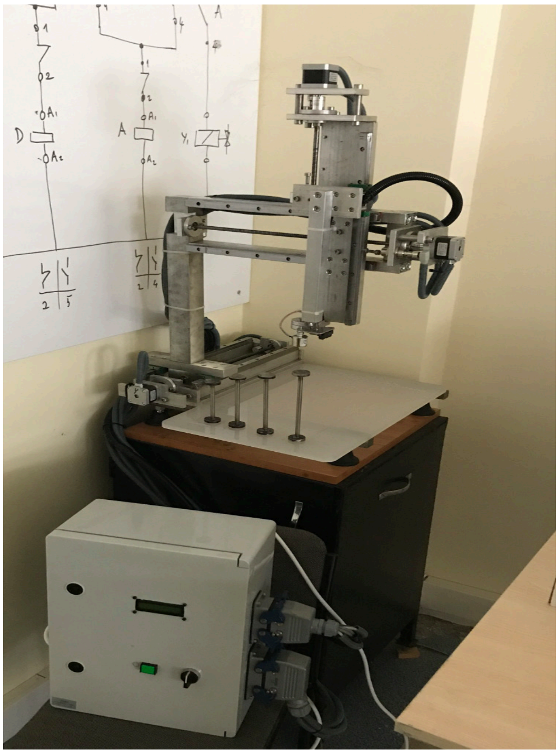

Fig. 11. *Robotic Arm*

REFERENCE

[1] Arduino- Using Sharp IR Sensor For Distance Calculation, http://luckylarry.co.uk/arduino-projects/arduino-using-a-sharp-ir-sensor-for-distance-calculation/

[2] Netflix, The Secret Rules of Modern Living: Algorithms Movie, Marcus du Sautoy

[3] ARDUINO UNO + TB6560 Stepper Motor Driver, http://www.instructables.com/id/ARDUINO-UNO-TB6560-Stepper-motor-driver/

[4] Linux Robotics: Programming Smarter Robots, D. Jay Newman