FABRICATION AND DEVELOPMENT OF A SERVO CONTROLLED PICK AND PLACE ROBOTIC ARM

A. Johnson1

Senior Assistant Professor

KITS Singapur, Huzurabad.

Telangana, India-505468

ajs.kits@gmail.com

M.Venkatesh.2

Assistant Professor

KITS Singapur, Huzurabad.

Telangana, India-505468

ajs.kits@gmail.com

ABSTRACT

This paper deals with an automated material handling system that deals with the movement of the robotic arm to pick the objects which are moving on a vertical axis conveyor belt. This work aims the fabrication a model which detects the objects placed on the conveyor and then it picks and places the objects in their respective pre-programmed location. Thereby eliminating the monotonous work done by a human, achieving accuracy and speed in the work. The main aim of the project is to pick and place objects automatically using the robotic arm. The system also uses an IR obstacle sensor and buzzer alert. The microcontroller sends a signal to the motors of the robotic arm and which drives the various motors of the robotic arm to grip the object and place it in the specified location. Based upon the object's presence detected, the robotic arm moves to the specified location releases the object, and comes back to the original potion. The system also checks the replacement of the container using the LDR sensor.

The object is sensed using an IR obstacle sensor. The optical sensor is the combination of an infrared sensor & a phototransistor. When the object cuts the infrared lights passed to the phototransistor, we get a square wave output signal on the output stage of the sensor the digital signal is then applied to the Microcontroller. The basic firmware for the microcontroller is written in Embedded C language. The system also uses a comparator circuit to compare and detect the obstacle and alerts through a buzzer alarm system. The system uses a DC motor for the movement of the conveyer belt and a servomotor for picking the objects and placing them in the container at one place with a particular angle. The Project consists of a microcontroller, object detection using an IR obstacle sensor, Servomotors, and DC motors. The block diagram of the hardware is shown below. The system also uses a DC motor to operate the conveyer belt on a vertical axis and alerts the user when the object is detected using an IR obstacle sensor. Using the object detection IR sensor, the system can detect the presence of the object on the conveyor belt. The microcontroller is programmed & stored in `C` language.

Keywords— microcontroller, phototransistor, LDR sensor

# INTRODUCTION

## **Robotic Arm:**

Arms are types of jointed robot manipulators that allow robots to interact with their environment. Many have onboard controllers or translators to simplify communication, though they may be controlled directly or in any number of ways. Due to this fact, stand-alone arms are often classified as full robots. A robotic arm is a type of mechanical arm, usually programmable, with similar functions to a human arm; the arm may be the total of the mechanism or may be part of a more complex robot. The links of such a manipulator are connected by joints allowing either rotational motion (such as in an articulated robot) or translational (linear) displacement. The links of the manipulator can be considered to form a kinematic chain. The terminus of the kinematic chain of the manipulator is called the end effecter and it is analogous to the human hand.

## **Robotic Hand:**

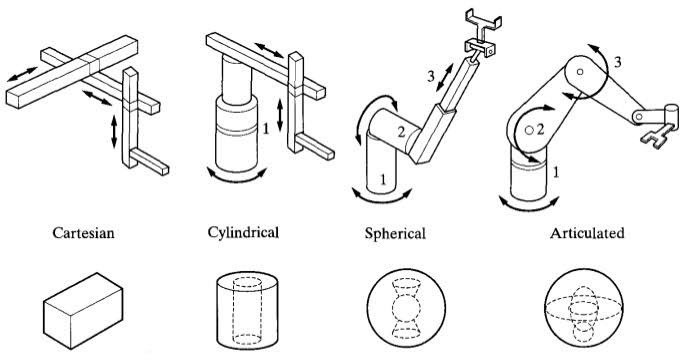
The end effecter, or robotic hand, can be designed to perform any desired task such as welding, gripping, spinning etc., depending on the application. For example, robot arms in automotive assembly perform a variety of tasks such as welding and parts rotation and placement during assembly. In some circumstances, close emulation of the human hand is desired, as in robots designed to conduct bomb disarmament and disposal.



**Fig1.1: Articulated Robotic Arm**

## **Types of Robotic Arms:**

There are many different types of robotic arms, but most can be characterized into one of six major categories by their mechanical structure. Cartesian (also known as Gantry) robots have three joints that are coincident with the standard X-Y-Z Cartesian axes. Cylindrical arms have any number of joints that operate on a cylindrical axis, normally rotating about one fixed rod. Spherical (polar) arms are those with joints that allow full rotation throughout a spherical range. SCARA robots have two parallel rotary joints to allow full movement throughout a plane, typically for pick-and-place work. Articulated robots are used for complex assembly operations, and consist of three or more rotary joints. Parallel robots have three concurrent prismatic or rotary joints and allow for tilting of heavy or sensitive platforms.



**Fig1.2: Types of Robotic Arms**

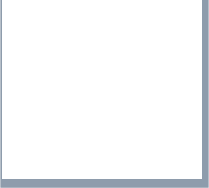
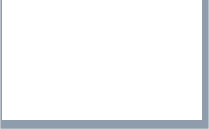
# WORKING PRINCIPLE OF PICK AND PLACE ROBOTIC ARM

## **Working of Servo-Controlled Pick and Robotic Arm:**

A pick and place robot is used to pick up an object and place it in the desired location. It can be a cylindrical robot providing movement in horizontal, vertical, and rotational axes, a spherical robot providing two rotational and one linear movement, an articulated robot or a scara robot (fixed robots with 3 vertical axes and rotary arms)

The basic function of a pick and place robot is done by its joints. Joints are analogous to human joints and are used to join the two consecutive rigid bodies in the robot. They can be rotary joints or linear joints. To add a joint to any link of a robot, we need to know about the degrees of freedom and degrees of movement for that body part. Degrees of freedom implement the linear and rotational movement of the body and Degrees of movement imply the number of axes the body can move.

A simple pick and place robot consists of two rigid bodies on a moving base, connected with a rotary joint. A rotary joint provides rotation around any one of the axes. A simple pick and place robot can be controlled by controlling the movement of its end effector. The motion can be using hydraulic motion, i.e. using hydraulic fluid under pressure to drive the robot, or using pneumatic motion, i.e. using pressurized air to cause mechanical motion. However, the most effective way is using motors to provide the required motion. The motors have to be controlled to provide the required motion to the robot and the end effector.



**START**

0

**STOP**

**ARM COMES BACK AT ORIGINAL POSITON**

**ARM PLACES THE**

**OBJECT AT**

**DESIRED POSITION**

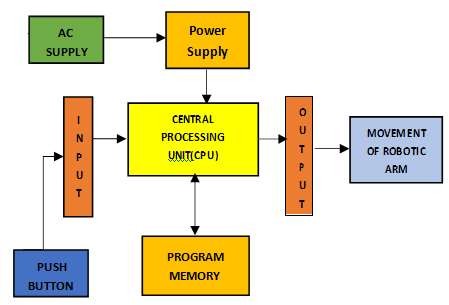
**OBJECT PICKEDUPBY ARM**

**OBJECT**

**DETECTION BYSENSOR**

**CONVEYOR**

**START**



# SPECIFICATION OF PICK AND PLACE ROBOTIC ARM HARDWARE REQUIREMENTS

**3.1 Hardware Components:**

1. Microcontroller (16F877A) 2. Reset button 3. Crystal oscillator 4. Regulated power supply (RPS) 5. LED indicator 6. IR Obstacle sensor 7. DC Motors 8. DC motor driver 9. Servo motor

10.LDR sensor 11. Buzzer.

**3.2 Software Required**

This project is implemented using the following software:

**Express PCB** – for designing circuit

**PIC C compiler** - for the compilation part

**Proteus 7 (Embedded C)** – for the simulation part

**3.2.1 Express PCB**

#### 

Breadboards are great for prototyping equipment as it allows great flexibility to modify a design when needed; however, the final product of a project, ideally should have a neat PCB, few cables, and survive a shake test. Not only is a proper PCB neater but it is also more durable as there are no cables that can yank loose.

Express PCB is a software tool to design PCBs specifically for manufacture by the company Express PCB (no other PCB maker accepts Express PCB files). It is very easy to use, but it does have several limitations.

It can be likened to more of a toy than a professional CAD program. It has a poor part library (which we can work around)

It cannot import or export files in different formats

It cannot be used to make prepare boards for DIY production

Express PCB has been used to design many PCBs (some layered and with surface- mount parts. Print out PCB patterns and use the toner transfer method with an Etch Resistant Pen to make boards. However, Express PCB does not have a nice print layout. Here is the procedure to design in Express PCB and clean up the patterns so they print nicely.

**3.2.2 Preparing Express PCB for First Use:**

Express PCB comes with a less than an exciting list of parts. So, before any project is started head over to Audio logical and grab the additional parts by morsel, ppl, and tangent, a extract them into your Express PCB directory. At this point start the program and get ready to set up the workspace to suit your style.

Click View -> Options. In this menu, set up the units for "mm" or "in" depending on how you think, and click "see through the top copper layer" at the bottom. The standard color scheme of red and green is generally used but it is not as pleasing as red and blue.

**3.2.3** **PIC Compiler:**

PIC compiler is software used where the machine language code is written and compiled. After compilation, the machine source code is converted into hex code which is to be dumped into the microcontroller for further processing. PIC compiler also supports C language code.

You must know the C language for the microcontroller which is commonly known as Embedded C. As we are going to use PIC Compiler, hence we also call it PIC C. The PCB, PCM, and PCH are separate compilers. PCB is for 12-bit opcodes, PCM is for 14-bit opcodes, and PCH is for 16-bit opcode PIC microcontrollers. Due to many similarities, all three compilers are covered in this reference manual. Features and limitations that apply to only specific microcontrollers are indicated within. These compilers are specifically designed to meet the unique needs of the PIC microcontroller. This allows developers to quickly design application software in a more readable, high-level language. When compared to a more traditional C compiler, PCB, PCM, and PCH have some limitations. As an example of the limitations, function recursion is not allowed.

This is because the PIC has no stack to push variables onto, and also because of the way the compilers optimize the code. The compilers can efficiently implement normal C constructs, input/output operations, and bit-twiddling operations. All normal C data types are supported along with pointers to constant arrays, fixed point decimal, and arrays of bits.

PIC C is not much different from a normal C program. If you know assembly, writing a C program is not a crisis. In PIC, we will have the main function, in which all your application-specific work will be defined. In the case of embedded C, you do not have any operating system running in there. So you have to make sure that your program or main file should never exit. This can be done with the help of a simple while (1) or for (;;) loop as they are going to run infinitely.

We must add a header file for the controller you are using, otherwise, you will not be able to access registers related to peripherals.

#include <16F877A.h> // header file for PIC 16F877A**//**

* + 1. **Proteus:**

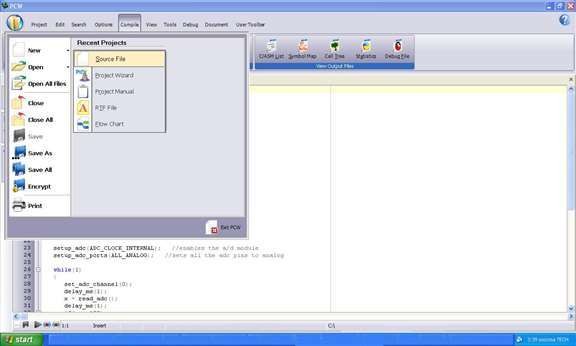
Proteus is software that accepts only hex files. Once the machine code is converted into hex code, that hex code must be dumped into the microcontroller and this is done by the Proteus. Proteus is a programmer that itself contains a microcontroller in it other than the one which is to be programmed. This microcontroller has a program in it written in such a way that it accepts the hex file from the pic compiler and dumps this hex file into the microcontroller which is to be programmed. As the Proteus programmer requires a power supply to be operated, this power supply is given from the power supply circuit designed and connected to the microcontroller in proteus. The program which is to be dumped into the microcontroller is edited in proteus and is compiled and executed to check any errors hence after the successful compilation of the program is dumped into the microcontroller using a dumper.

* 1. **Procedural Steps for Compilation, Simulation and Dumping:**

For the PIC microcontroller, the PIC C compiler is used for compilation. The compilation steps are as follows:

Open PIC C compiler.

You will be prompted to choose a name for the new project, so create a separate folder where all the files of your project will be stored, choose a name and click save.



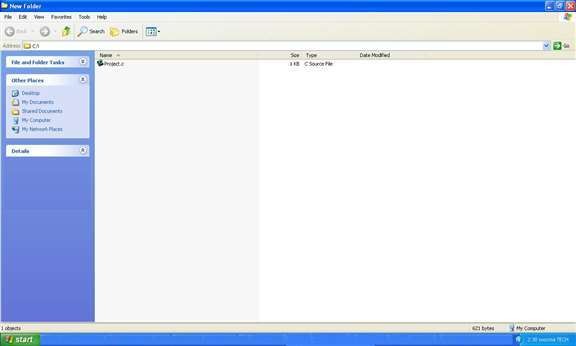
**Fig 3.3.1: Picture of opening a new file using the PIC C compiler**

Click Project, New, and something in the box named 'Text1' is where your code should be written later.

Now you must click 'File, save as' and choose a file name for your source code ending with the letter '.c'. You can name as 'projects' for example and click save. Then you have to add this file to your project work.

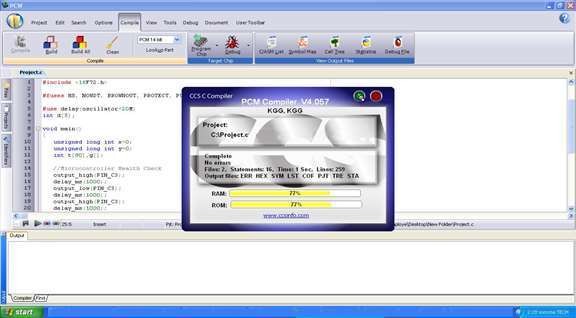


**Fig 3.3.2: Picture of compiling a new file using the PIC C compiler**



**Fig 3.3.3: Picture of compiling a project's file using the PIC C compiler**

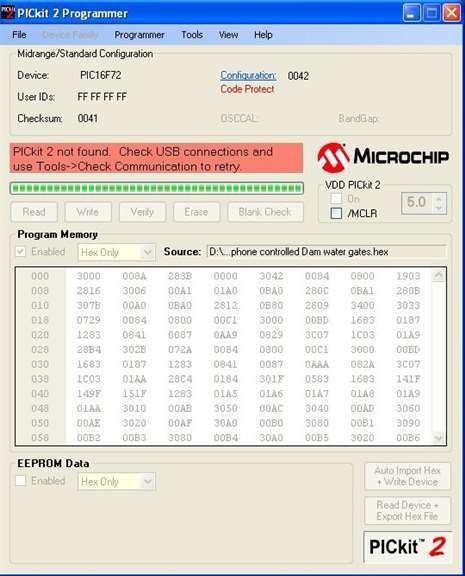
You can then start to write the source code in the window titled 'projects' then before testing your source code; you have to compile your source code and correct eventual syntax errors.



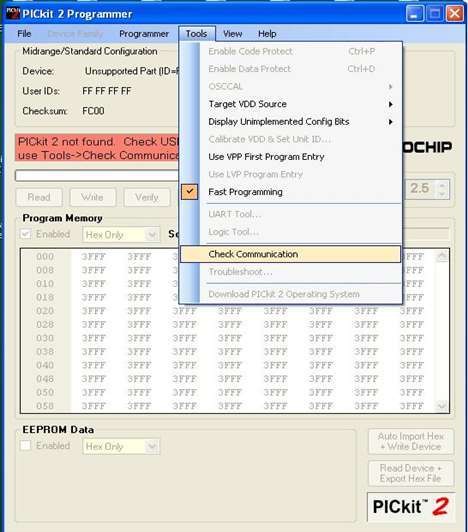
**Fig 3.3.4: Picture of checking errors and warnings using the PIC C compiler**

* 1. **Dumping steps:** The steps involved in dumping the program edited in proteus 7 to the microcontroller are shown below:

Initially, before connecting the program dumper to the microcontroller kit the window appeared as shown below.

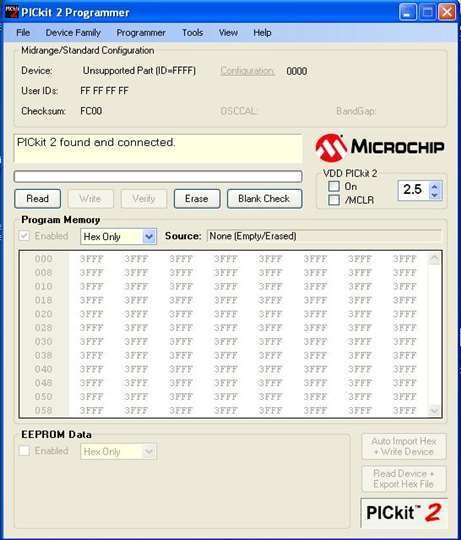


**Fig 3.4.1: Picture of program dumper window**

select the Tools option and click on Check Communication for establishing a connection as shown in the below window

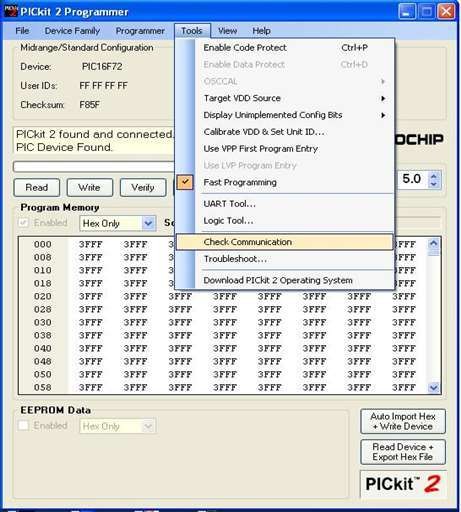
**Fig 3.4.2: Picture of checking communications before dumping the program into the**

**microcontroller**

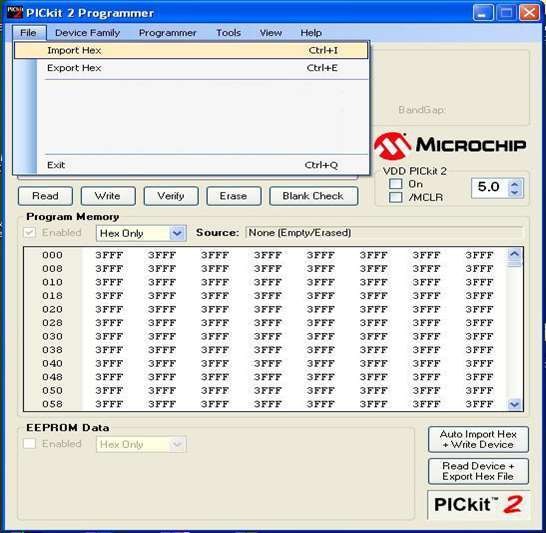
After connecting the dumper properly to the microcontroller kit the window appeared as shown below.

**Fig 3.4.3: Picture after connecting the dumper to the microcontroller**

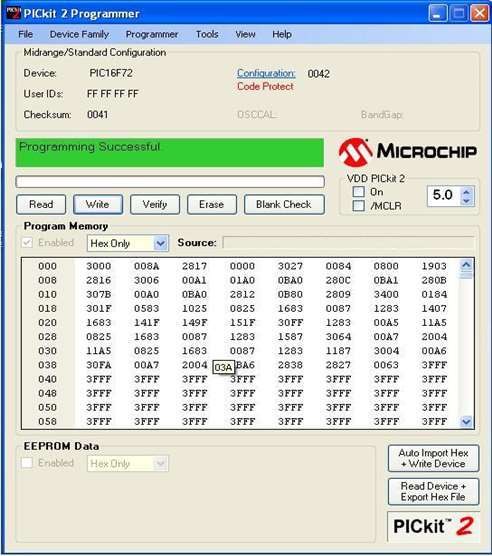
Again, by selecting the Tools option and clicking on Check Communication, the microcontroller gets recognized by the dumper and hence the window is as shown below.



**Fig 3.4.4: Picture of dumper recognition to the microcontroller**

Import the program which is the '. hex' file from the saved location by selecting the File option and clicking on 'Import Hex' as shown below the window.

**Fig 3.4.5:** **Picture of program importing into the microcontroller**

After clicking on the 'Import Hex' option we need to browse the location of our program and click the 'prog hex' and click on ‘open' for dumping the program into the microcontroller. After the successful dumping of the program the window is as shown below.

**Fig 3.4.6: Picture after program dumped into the microcontroller**

**IV: HARDWARE TESTING**

### **: Continuity Test:**

In electronics, a continuity test is the checking of an electric circuit to see if current flows (that it is a complete circuit). A continuity test is performed by placing a small voltage (wired in series with an LED or noise-producing component such as a piezoelectric speaker) across the chosen path. If electron flow is inhibited by broken conductors, damaged components, or excessive resistance, the circuit is "open".

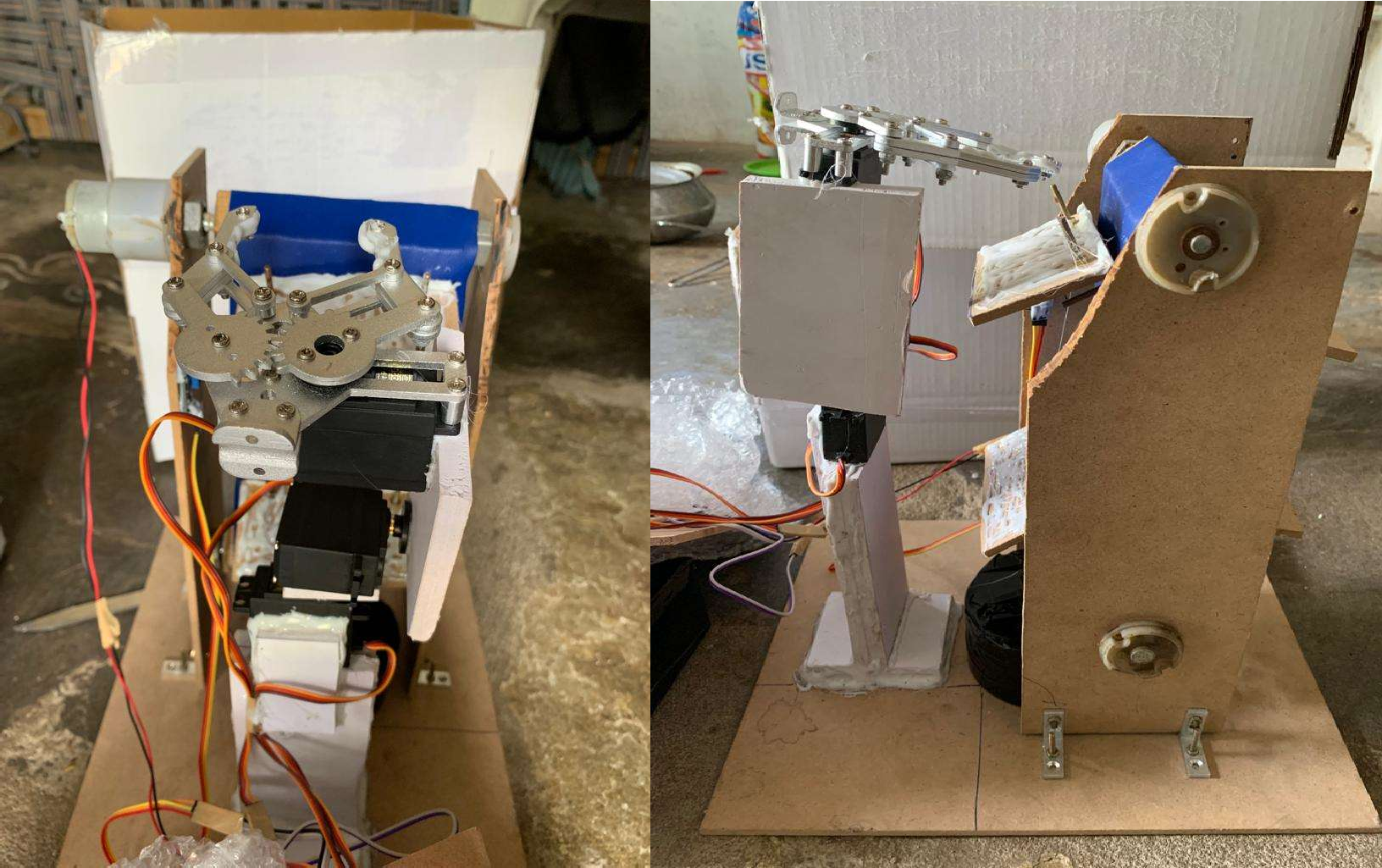
Devices that can be used to perform continuity tests include multimeters which measure current and specialized continuity testers which are cheaper, more basic devices, generally with a simple light bulb that lights up when current flows.

An important application is the continuity test of a bundle of wires to find the two ends belonging to a particular one of these wires; there will be a negligible resistance between the "right" ends, and only between the "right" ends.

This test is performed just after the hardware soldering and configuration have been completed. This test aims at finding any electrical open paths in the circuit after the soldering. Many times, the electrical continuity in the circuit is lost due to improper soldering, wrong and rough handling of the PCB, improper usage of the soldering iron, component failures and the presence of bugs in the circuit diagram. We use a multi-meter to perform this test. We keep the multimeter in buzzer mode and connect the ground terminal of the multimeter to the ground. We connect both the terminals across the path that needs to be checked. If there is continuation then you will hear the beep sound.

### **: Power On Test**:

This test is performed to check whether the voltage at different terminals is according to the requirement or not. We take a multimeter and check the voltages. First, check the voltage across the terminal whether the power is passing or not.



**Fig 4.1: Fabricated Model of servo-controlled pick and place robotic arm**

### **: Subsequent**: **Work:**

This work can be extended using the sorting of objects through color sensors the future advancements can be done by increasing the efficiency of the color sensor. The sensor is a key component of the project which aids in distinguishing the objects. Failing may result in wrong material handling. Thus, it becomes vital that the sensor had a very high sense of sensitivity and ability to distinguish between colors. The image processing software system is then used to develop a statistical characterization of the reflectance for each information class. Genetic algorithm has the merits of plentiful coding, and decoding, conveying complex knowledge flexibly. An advantage of the Genetic Algorithm is that it works well during global optimization, especially with poorly behaved objective functions such as those that are discontinuous or with many local minima. MATLAB genetic algorithm toolbox is easy to use, does not need to write long codes, the run time is very fast and the results can be visual. This work aimed to realize the image classification using Mat lab software.

MATLAB is a widely used software environment for research and teaching applications on robotics and automation, mainly because it is a powerful linear algebra tool, with a very good collection of toolboxes that extend Mat lab basic functionality, and because it is an interactive open environment.

The paper presents a toolbox that enables access to real robotic and automation (R&A) equipment from the Mat lab shell. If used in conjunction with a robotics toolbox it will extend significantly their application, i.e., besides robotic simulation and data analysis the user can interact online with the equipment. Personal experience with this tool shows its usefulness for research applications, but also teaching projects.

55

For students, using MATLAB means taking advantage of the reduced training required to start using it, if compare with other programming environments and languages that can also be used (Microsoft Visual C++or Visual Basic).

An innovative approach for quality sorting of objects such as apples sorting in an agricultural factory, using an image processing algorithm. The objective of the approach is; firstly, to sort the objects by their colors precisely; secondly to detect any irregularity of the colors surrounding the apples efficiently. An experiment has been conducted and the results have been obtained and compared with those that have been performed by the human sorting process and by color sensor sorting devices. The existing sorting method uses a set of inductive, capacitive and optical sensors to differentiate object color.

Advanced mechatronics color sorting system solution with the application of image processing. Supported by Open CV, the image processing procedure senses the circular objects in an image captured in real-time by a webcam and then extracts color and position information out of it. This information is passed as a sequence of sorting commands to the manipulator that does the pick-and-place mechanism.

Extensive testing proves that this color-based object sorting system works 100%accurate under ideal conditions in terms of adequate illumination, circular objects' shape and color. The circular objects tested for sorting are silver, red and black. For non-ideal conditions, such as unspecified color the accuracy reduces to 80%.

**V: RESULT AND CONCLUSION**

**5.1: Result**

The work “Servo motor-controlled pick and place robotic arm with vertical axis conveyor belt and plate detection” was fabricated and developed successfully to pick and place objects. This model uses an IR obstacle sensor with a comparator and buzzer alert for object presence detection. The system also senses the container replacement using an LDR sensor for placing the objects out of the container.

**5.2: Conclusion**

Integrating features of all the hardware components used have been developed in it. The presence of every module has been reasoned out and placed carefully, thus contributing to the best working of the unit. Secondly, using highly advanced ICs with the help of growing technology, the project has been successfully implemented. Thus, the project has been successfully designed and tested.

**5.3: Future Scope**

This work “Servo motor-controlled pick and place robotic arm with vertical axis conveyor belt and plate detection” is mainly intended to design an IR obstacle sensor-based object detection and pick the object using a robotic arm from the vertical axis conveyer belt.

The microcontroller sends a signal to the motors of the robotic arm and which drives the various motors of the robotic arm to grip the object and place it in the specified location. Based upon the object's presence detected, the robotic arm moves to the specified location releases the object and comes back to the original potion. The system also checks the replacement of the container using the LDR sensor.

The object is sensed using an IR obstacle sensor. The optical sensor is the combination of an infrared sensor & a phototransistor. When the object cuts the infrared lights passed to the phototransistor, we get a square wave output signal on the output stage of the sensor the digital signal is then applied to the Microcontroller. The basic firmware for the microcontroller is written in Embedded C language. The system also uses a comparator circuit to compare and detect the obstacle and alerts through a buzzer alarm system. The system uses a DC motor for the movement of the conveyer belt and a servomotor for picking the objects and placing them in the container at one place with a particular angle.

This work can be extended by using which eliminates the line-of-sight problem and also distance of operation is increased. Also, we can get feedback about the status of the conveyer belt is not present

In the future, we can use this project in several applications by adding additional components. The project can be extended using a GSM modem which displays the conveyer belt operation through SMS alert messages. The system also can be extended using a GPS module which can alert the user to the location of the belt operation.

##### REFERENCES

1. Tsalidis, S., S., Dentsoras, A., J., “Application of design parameters space search for belt conveyor design” Journal of Plant Science, Vol. 10, No. 6, pp. 617-629, 2010.
2. Huang, T, Wang, P.F., Mei, J.P., Zhao, X.M., “Time Minimum Trajectory Planning of a 2- DOF Translational Parallel Robot for Pick-and-place Operations”, IEEE Computer Magazine, Vol. 56, No. 10, pp. 365-368, 2007.
3. Sahu, S., Lenka, P.; Kumari, S.; Sahu, K.B.; Mallick, B.; “Design a color sensor: Application to robot handling radiation work”, Industrial. Engineering, Vol. 11, No. 3, pp. 77-78, 2010.
4. Khojastehnazhand, M., Omid, M., and Tabatabaeefar, A., “Development of a lemon sorting system based on color and size”, Journal of Plant Science, Vol. 4, No. 4, pp. 122-127, 2010.
5. Dogan Ibrahim “Microcontroller Based Applied Digital Control”, International Journal of Science, Vol. 23, No. 5, pp.1000- 1010, 2007.
6. Raj Kamal –Microcontrollers Architecture, Programming, Interfacing and System Design.
7. Mazidi and Mazidi –Embedded Systems.
8. PCB Design Tutorial –David. Jones.
9. PIC Microcontroller Manual – Microchip.
10. Pyroelectric Sensor Module- Murata.