Palestine Polytechnic University

College of Information Technology and Computer Engineering

Department of Computer Systems Engineering

Graduation Project

"Object Sorting System Using Robotic Arm"

”

Team work:

Enas Ideas

Haneen Qabajah

Kifah Abdeen

Supervisor:

Dr. Liana Tamimi

A final project submitted in partial fulfillment of the requirements for the bachelor degree in Computer Systems Enginnering

Hebron - Palestine

June 2018

Chapter 1
Introduction

This chapter will discuss briefly the project overview, motivation, importance, objectives, a description of the project and a summary.
Also,we will analyze system problems, introduce system requirements, present some of the problems and risks to which we can face and how to solve it, and the results of this system.

1.1 Overview

In this fast growing industrial age every industrial unit needs speed in manufacturing. Robotics has found a wide application in industries. Automation provides far better service to customers eliminating the monotonous work by human, achieving accuracy and speed in work.

Robot is an integral part in automating the flexible manufacturing system that one greatly in demand these days." Robots are now more than a machine, as robots have become the solution of the future as cost labor payments and customers demand"[1].

In our project, we build a robot that will interact with computer to classify object based on owner commands. The purpose of this is to provide a robotic arm that sorts the objects in the most efficient way.

1.2 Motivation

The robotic arm is one of the most useful pieces of technology to be introduced in the 20th century, and quickly became a cornerstone in many areas of manufacturing. It can be used for many different jobs and functions that may be too boring, difficult or dangerous for a human to do [2].

This project also may be a solution for people who works in factories, specially sorting section to sort objects by color.

1.3 Importance:

The importance of this project is to improve process by purchasing a robotic arm to replace the workers performing this hard effort.

This robot has an arm which is used for the function of pick and place. This project presents an object sorting system solution based on color using computer vision, it's used for identification of color and information out of it. This information is processed for picking the object on the basis of color and place at particular position. The project deals with an automated material handling system.
The goal of the project is classify the object based on their color.

In addition, this project will open new prospects to other developers to enable robot to classify object based on other techniques like, size and shapes.

1.4 Objectives:

Our main objective is to build a robot that does some task that constrained to sorting objects according to their color.

Hence, the project is assumed to achieve these objectives:

1. Use camera to determine the signature for objects.
2. Send pixy camera parameters to the microcontroller (Arduino) .
3. Use arduino sorting algorithm to analyze the color of object.
4. Show the color on LCD device .
5. Build an arm system that is able to grap the object and place it.

1.5 System Description:

Our project consists of two main parts: the arm, and the camera. The first part is an arm that classifies objects according to their colors and places them in their assigned place. Camera is fixed and the arm is settled at a certain angle, then it detect the signature for objects and analyze it to determine the color, so the arm move and then pick up the object and put in a suitable place.

In addition to these main parts, there will be other components to enable the main parts work, such as arduino, stepper motors, LCD, and color detection defined in arduino-uno sorting algorithm .

1.6 Problem statement

Based on a survey conducted in a Helmet manufacturing company, there was a problem regarding the count of the total manufactured products and also, there were various assembly lines for every different colored helmet. The main idea was to provide a single conveyor for all the different colored products which would decrease the work space and labor cost but also provides the basic function of segregating different colored objects into its respective boxes. In addition, the accurate count of the manufactured products could be centralized using wireless communication[3].

In a food packaging industry huge amount of time and labor was invested in segregating raw and ripened tomatoes. An object sorting robot would decrease the time, work space and labor cost while providing the basic function .

To decrease human works in operating the mechanical machines, different functionality robotic arms are established. Different functionality arms that are used in robotics are designed and developed to handle the jobs that are repeated. Different considerations are taken care to design the automation system. To design a high strength mechanical structure, there are many important parameters to be consider, some of them load bearing capacity, optimum weight, degree of rotation and speed of movement. In the form of designing an electronics system the specification of the used electronics devices are to be considered[4].

1.7 List of requirements

The system requirements can be summarized as :

1. Detect each object by camera.
2. Algorithm to analyze the color for object
3. Control the motion of the Arm in multi direction.
4. Provide a good connectivity between the arm movement , the camera and the arduino control environment.
5. The system must be able to be expanded in the near future and new features may be added to the system.

6. Control the motion of the Arm in multi direction.

1.8 Risk management

This section includes a list of the risks that may face us in our project, and these risks are:

1. Failure in the components and the difficulty of obtaining other copies of the same type, which will be solved by buying new alternative components that will suit our project.
2. Failure in the program pieces or failure of the laptop which leads to loss of work, so we will put everything in our Google drive account.
3. Hardware components may not fit properly with each other, so we will search for alternate components and buy them.
4. Memory may not be enough to hold our project programmes.

1.9 Results

We expect to gain the following results:

1. Build the whole system as being described in this chapter.
2. Connect system components correctly as needed.
3. The robot can sort object in different colors.
4. The robot can move it's object from conveyor belt to specific colored box .
5. The robot can make it's movement decision upon color detection algorithms, with help from camera .
6. The LCD device can display the color for objects .

1.10 Summary

This chapter discusses the system in general terms, also introduces the problem statement and the state of the art solution that we suggest. Chapter 2 introduces the background of the project, and the various devices accompanied and assembled to build the whole system. Chapter 3 discusses the design of the system, it introduces the flowchart, block diagram, sequence diagram, system diagram and the main component of the systems. Chapter 4 discusses software and hardware implementation. Chapter 5 discusses the software and hardware testing. Chapter 6 highlights the future work and the conclusion of our project.

Chapter 2

Background

2.1 Overview

This chapter introduces the theoretical background of our project, short description of the hardware components that will be used in the system, short description of the software components, design specifications and constraints.

2.2 Theoretical Background

This section introduces some information about the hardware components of the project and the technologies that will be used in our project.

2.3 Related Work

Since our project is a new developed idea in the region, they have a similar project only in a sorting definition :

- Designing a Can Sorting Machine by Color

Palestine Technical University – Khadoori , Department of Mechanical Engineering.

A can sorting machine is built in this project. In this machine, used cans of beverages are categorized in two stages: material (metal versus non metal) and color (red, yellow and green) .

- Ball Collecting Robot

Al-Najah university - Department of computer system Engineering

Ball collector robot is a vehicle that analyze its surrounding area then detect balls and collect them to specific point known with predefined sign .

2.4 Design Options

In this section, we will discuss about design options that include controller options and arm options.

2.4.1 Controller Options

We used arduinouno microcontroller not the raspberry pi microcomputer, even both are used in robot controlling but we choose arduinouno since it is very easy to deal with and program it,In addition of computer vision.

The arduinouno takes the responsibility to control the whole system whether the arm or the camera. During our search about arduinomicrocontroller we faced more than one type we can use in the project, the choice fall between arduinouno and arduino mega. But we choose arduinouno to be used in our project instead of arduino mega, because of its specifications, which will be mentioned later in the next section, that we need in this project, so it is more suitable for the system.

A simple comparison between arduino mega and arduinouno could be as this:

Microcontrollers:

Arduino Mega uses an ATmega2560 microcontroller which has:

- 256K Bytes of In-System Self-Programmable Flash.
- 8K Bytes RAM.
- 4K Byte Internal SRAM[5].

Arduino Uno uses an ATmega328p microcontroller which has:

- Flash Program Memory: 32 kbytes.
- EEPROM Data Memory: 1kbytes.
- SRAM Data Memory: 2 kbytes[5].

Boards:

Arduino Mega board has:

- 54 digital I/O pins, of which 14(D0 to D13) can be used as PWM outputs.
- 16 analog input pins, which can also be used as digital I/O pins, adding to the existing 54 digital I/O pins.
- 4 serial communication lines(pins D0, D1, and from D14 to D19)[5].

The Arduino Uno board has:

- 14 digital I/O pins, of which 6 (D3, D5, D6, D9, D10, and D11) can be used as PWM outputs.
- 6 analog inputs, which can also be used as digital I/O pins, adding to the existing 14 digital I/O pins.
- Only one serial communication line (D0, D1)[5].

The reason behind our selection for arduinouno instead of arduino mega is that we do not need that much of pins, we only need 4 PWM pins and it is much smaller.

2.4.2 Arm Options

We found many choices for the arm during our search, each one has its characteristics, which differ from others and may be similar to other in some characteristics. Below some of arms that we have chosen between them.

2.4.2.1 Dobot -Robotic Arm - Magician - Basic Plan7

This one is better than what we have chosen.”It manages to maintain the industrial precision level. When performing repetitive actions, the error in precision is as small as 0.2mm.The arm can draw, write text, move, grab things following your orders, and even laser cutting, allow for 3D printing pen” [5].

The reason behind of that we have not choose it is it’s price, it needs about US $1,022.93 and we cannot afford this price.

2.4.2.2 6DOF Robot Arm Clamp Claw Mount Full Set Metal Structure without Steering wheel

The specifications of this arm are good and suitable of our project, because it's have a good stability and the price is cheap, it's need about US $350.So, we choose it .

2.5 System Components

In this section, we will discuss about components that include hardware components and software components.

2.5.1 Hardware Components

This section describes the main hardware that we will use to construct the project. In addition to the function of each one.

2.5.1.1 Arduino Uno

The Arduino Uno is an open source hardware and software a microcontroller platform based on the ATmega328. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Uno differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter[6].

It is used in this project because of it is simplicity to be used and programmed to use high-level languages such as C and C++.

Arduino uno specifications [6]**:**

● Microcontroller ATmega328
● Operating Voltage 5V
● 14 digital input/output pins (of which 6 can be used as PWM outputs)
● DC Current per I/O Pin 40 mA
● DC Current for 3.3V Pin 50 mA
● Flash Memory 32 KB (ATmega328) of which 0.5 KB used by bootloader
● Memory: Flash 32KB, SRAM 2KB, EEPROM 1KB
● Clock Speed 16 MHz

The Arduino uno has the main function in the project, which is control the whole system.

Figure 2.1: Arduino uno

2.5.1.2 6DOF Robot Arm Clamp Claw Mount Full Set Metal Structure without Steering wheel

This arm will be used as a whole package after being bought, it will be a basic component in our project. It is already assembled and tested well for its work by the manufacturers.

Its specifications are [7]:
● 6 DOF.
● Waist: 0-180 degree.
● Big arm: Big arm and small arm interaction, rotation angle range is variable.
● Small arm: Big arm and small arm interaction, rotation angle range is variable.
● Small arm rotation: 0-180 degree.
● Wrist: 0-180 degree.
● Paw 0-180 degree.

- 0~180 degree is the theoretical value, the actual angle between 5~175 degrees.

We will use the arm to classify objects and put them in their proper place, it will be connected to Arduino-uno.

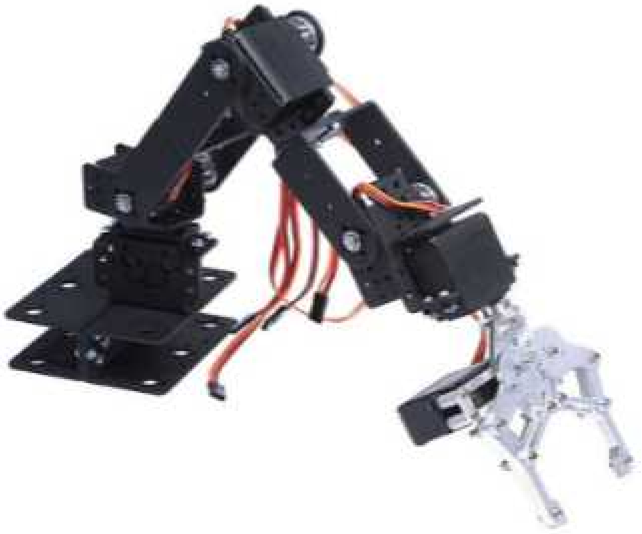


Figure 2.2 6DOF Robot Arm Clamp Claw Mount Full Set Metal Structure

2.5.1.3 Camera

We will use this camera, Pixy CMUcam5 that has a smart vision sensor and is an object-tracking camera for Arduino uno.

Its specifications [8]**:**

- RAM: 264K bytes.
- Flash: 1M bytes.
- Power consumption: 140 mA typical
- Dimensions: 2.1in x 2.0in x 1.4in
- Processor: NXP LPC4330, 204 MHz, dual core
- Image sensor: Omni vision OV9715, 1/4, 1280x800
- Lens type: standard M12 (several different types available)
- Lens field-of-view: 75 degrees horizontal, 47 degrees vertical
- Power input: USB input (5V) or unregulated input (6V to 10V)
- Available data outputs: UART serial, SPI, I2C, USB, digital, analog
- Weight: 27 grams.

This camera is connected to the arduinouno in order to reflect the color of object to the arduino. The arduinouno will interpret these information and make the arm movement decision to the appropriate location at the appropriate time

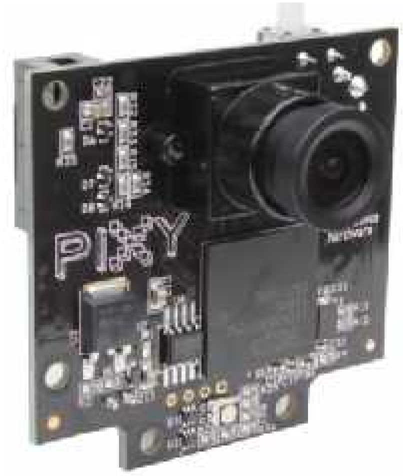


Figure 2.3: Pixy CMUcam5

2.5.1.4 Adafruit 16-Channel 12-bit PWM/Servo Driver Shield

Driving servo motors with the Arduino Servo library is easy, but each one consumes a precious pin - not to mention some Arduino processing power. The Adafruit 16-Channel 12-bit PWM/Servo Driver Shield will drive up to 16 servos over I2C with only 2 pins. The on-board PWM controller will drive all 16 channels simultaneously with no additional Arduino processing overhead. What's more, you can stack up to 62 of them to control even more servos - all with the same 2 pins!. The Adafruit PWM/Servo Driver is the perfect solution for any project that requires a lot of servos. [9]

Even if in our project we needed only 5 servos, and still we used the shield, it because of the current leakage that cause the arm to stop moving or to move randomly

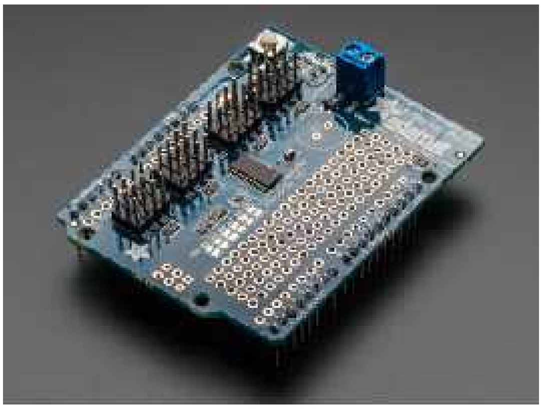


Figure 2.4: Adafruit 16-Channel 12-bit PWM/Servo Driver Shield

2.5.1.5 Conveyor

Is a common piece of mechanical handling equipment that moves objects from one location to another.

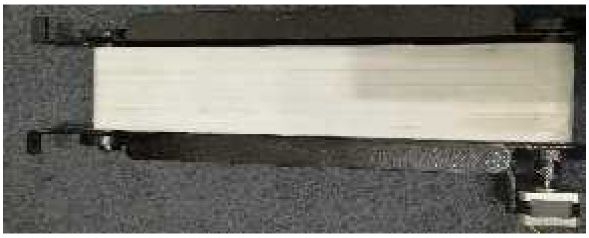


Figure 2.5: Conveyor

2.5.1.6 Stepper motor

The stepper motor is an electromagnetic device that converts digital pulses into mechanical shaft rotation. Advantages of step motors are low cost, high reliability, high torque at low speeds and a simple. The main disadvantages in using a stepper motor is the resonance effect often exhibited at low speeds and decreasing torque with increasing speed[10]

Figure 2.7:Stepper motor

2.5.1.7 TB6560 driver stepper motor

The TB6560 Stepper driver is a good option to drive bipolar stepper motors that require up to 3A of current (adjustable via dip switches).

Figure 2.6: TB6560 driver stepper motor

2.5.1.8 Object

An object to be classified like cubes that have different color specifically four colors: red, green ,yellow and blue.

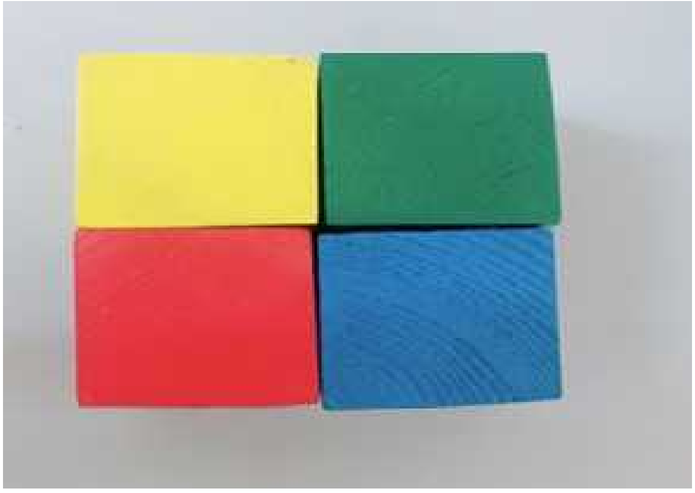


Figure 2.8:Objects

2.5.1.9 Power Supply kit

This kit was used to power the arduino servo shield since it needs separate voltage and current to operate well rather the one provided to the arduinouno . We supplied it with 5 volts -at least- and more than 1.8 ampere.

Figure 2.8:Power Supply kit

2.5.1.10 LCD

The LCD display the color, When the camera recognize the color of object .

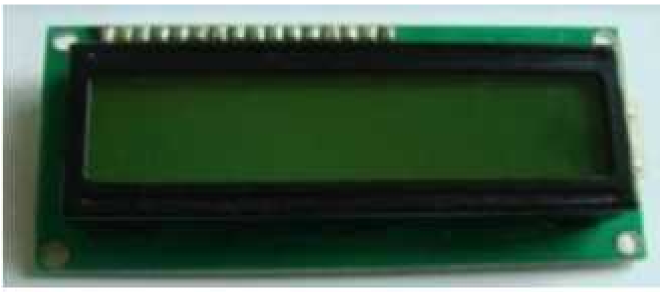


Figure 2.9:TheLCD

2.5.1.11 Breadboard

It was used to connect the lcd pins with it .

2.5.1.12 Laptop

Laptop is an important part in our project .We used it to program the Arduino-uno, also to connect the camera in order to see what it is seeing and to install the required application for it .

2.5.1.13 Wires

Wires are used to connect the component of the system to each other.

2.5.2 Software Components

This section describes software components that we will use to program the project. In addition to the function of each one.

2.5.2.1 Arduino IDE

The Arduino Integrated Development Environment - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino and Genuine hardware to upload programs and communicate with them. Simple, clear programming environment-The Arduino Software(IDE) is easy to-use for beginners, yet flexible enough for advanced users to take advantage of as well. For teachers, it’s conveniently based on the Processing programming environment, so students learning to program in that environment will be familiar with how the Arduino IDE works[11].

2.5.2.2 PixyMon

"PixyMon is an application that allows you to configure Pixy and see what it sees. It runs on several different platforms including Windows, MacOS and Linux, as well as other smaller embedded systems . It also allows you set the output port and manage color signatures. PixyMon communicates with Pixy over a standard mini USB cable"[12].

This software was used in order to give each object an signature, and to teach the pixycamera the object in order to detect it later.

2.6 Design Specifications and Constraints

In this section, we will discuss design specifications and constraints.

2.6.1 Design Specifications

The system must be able to analyze the signatureof an objects, send the informations to the Arduino and make the right decision of the arm movement synchronically depend on the analyzing.

2.6.2 Constraints

- The use of a large number of servo motors in the automatic arm, which required amount of energy that was not enough to use the microcontroller directly, forcing us to use an external power supply.

- The robotic arm is fixed in one place so it can do a specific functions.

- The amount of power supply required for the arm is fixed ,it have a voltage about (5.3-5.9) V and the current is nearly about 1.9A.

- The color of conveyer is white, to provide a good vision on camera .

2.7 Inverse kinematics equation

```
voidSpecial_Calc_Point() {

Hypoth = sqrt(sq(x1) + sq(y1)) + WristLength;
  A = ElbowLength;//4;
  B = Hypoth;//B_Value;//6.7; //check here for nan.
  C = ShoulderLength;//6;


  //*** Calculate angle A ***//
Angle_A = acos((sq(B) + sq(C) - sq(A)) / (2 * B * C)) * (180 / PI);
  //    Angle_A_Temp = acos((sq(x1) + sq(Hypoth) - sq(y1)) / (2 * x1 * Hypoth))*(180
/ PI); //Second 1/2 of Angle A from X axis upto Shoulder arm.
  //     Angle_A = Angle_A + Angle_A_Temp; //Second 1/2 of Angle A from X axis
plus Shoulder arm angle.
  // Angle_A = map(Angle_A, 0, 90, 30, 70); //must map values since using a gearbox
and limits at 30 and 65.Largest shoulder degree:109.


  //*** Calculate angle B ***//
Angle_B=  acos((sq(C) + sq(A) - sq(B)) / (2 * A * C)) * (180 / PI);

  //*** Calculate angle C ***//
Angle_C = acos((sq(A) + sq(B) - sq(C)) / (2 * A * B)) * (180 / PI);
  //  Angle_C = 180 - Angle_C; //must calc larger opposite side angle.
}
```

2.7.1 Converting PWM into degree :-

```
voidsetservopulse(unit8_t  n ,double  pulse){
double  pulselength ;
pulselength = 1000000;
pulselength   /=  60;
pulselength  /=  4096;
pulse *= 1000;
pulse /= pulselength;
pwm.setPWM (n,0,pulse );
degrees =map(pulselength, SERVOOMIN , 0 ,180) ;
}
```

Chapter 3

System Design

3.1 Overview

This chapter discusses the conceptual description of the system, it show schematic diagrams ,block diagrams, flowcharts and circuits of its hardware and software.

3.2 Brief description of the system

Our project mainly consists of the following parts, The first part presenting the automatic sorting based on color processed in arduino microcontroller, the microcontroller in turn will control the servo motors . These servomotors will control the movement of robotic arm, by controlling their angular movement with respect to the colored objects moving on conveyor belt which is also controlled by arduino.

The robotic arm will be fully controlled by servomotors. The gripper of Robotic arm will pick the object place it depending on its colors .

The other main part of this project is pixy camera for high resolution and color identification using PixyMonsoftware.The camera perform the well job of identifying objects, in addition it helps for turning of robotic arm to the next object sorting.

The sequence of system steps are: -

1. Turn on the robot system.
2. Put colored objects in conveyor belt and turn it on .
3. The pixy camera will identify the color for the current object by using signature configuration, also conveyor belt will stop automatically when the first object appear on PixyMon window.
4. Pixy Camera will send signature in file that contains pixy parameters to arduino-uno to determine the color for object in arduino sorting algorithm .
5. Display the detected color for object on LCD device .
6. Arduino will control the arm movement by angular movements for servo motors to grasp the object from conveyor belt then put it in suitable place .

   The arduino sorting algorithm will run in four cases :

   a. If the color object is red or close to red put it at the box which is the same color by moving arm servos in a sequential degrees .
   b. Else ,if the color object is green or close to green put it at the box which is the same color by moving arm servos in a sequential degrees .
   c. Else, if the color object is blue or close to blue put it at the box which is the same color by moving arm servos in a sequential degrees .
   d. Else, if the color object is yellow or close to yellow put it at the box which is the same color by moving arm servos in a sequential degrees .

7. Continue the steps from 1 to 6 while conveyor belt on and objects are available else turn off the system .

3.3 System design
3.3.1 Logical structure

3.3.2 Block Diagram

The block diagram shown in Figure 3.1 below illustrates the general overflow of the system.
In the working process it has two different area of work , one is for working with the colored objects for detection the color and another is for controlling the arm motion .

The conveyor motor receive the power form the power supply. The one pulley conveyor belt, with a circular loop of objects on the conveyor belt that rotates about them.
Forward movement of the belt bring the objects near the pixy camera lens. When objects detected by the camera the conveyor belt stop for the identification of the color for the object with the help of the smart vision sensor (pixy camera),this camera considered as smart camera provide computer vision using particular software called PixyMon .

Then the signature of the particular object color is send to the control unit in arduino for color determination which is using pixy parameters for arduino sorting algorithm , then the control unit (Arduino) sends the signal to the robotic arm that picks the object and place it in the boxes based on their colors , after placing the object, the arm of the robot is comeback to the normal place and waiting for the next object to arrive. Then the microcontroller start the conveyor to bring the next object to the sensor unit in the camera and the process continuous as explained in above in this page until the objects become not available on conveyor or when power for system turned off .

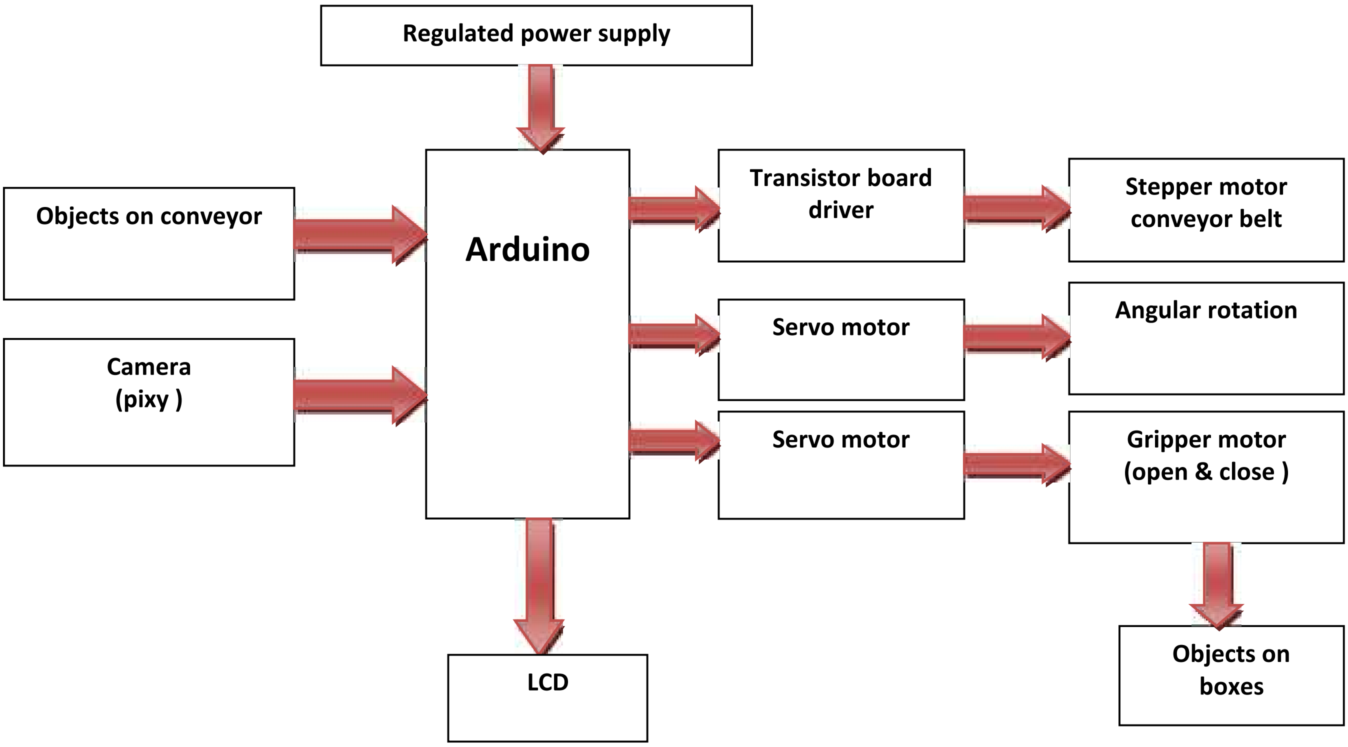


Figure 3.1 : The general block diagram

3.3.3 System Diagram

Figure 3.2 show how hardware components are connected with each other .

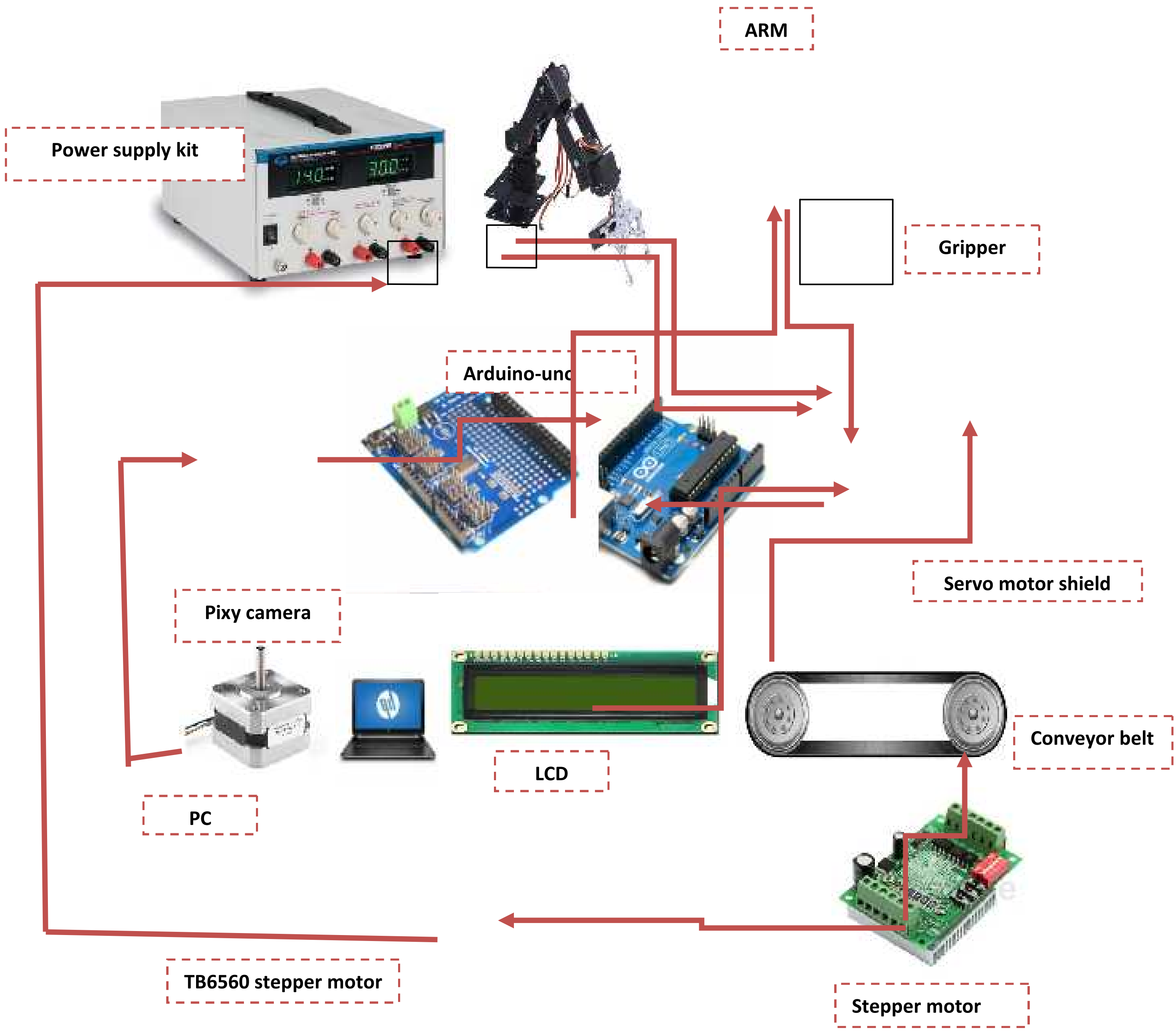


Figure 3.2: The system diagram

3.3.4 Flow Charts

Figure 3.3, 3.4 and 3.5 represent the flowcharts of system activity following the steps mentioned in section 3.2.

Flowchart : Methodology

In our project, we used a simple method for sorting the objects of a different colored , it is detecting the color of the object and sort out the different colored objects. With the help of a robotic arm that is controlled by a controller unit in Arduino, we ensure that place the same colored objects in the same boxes .

The general flowchartdiagram :-

The system will start when we press turn on button .

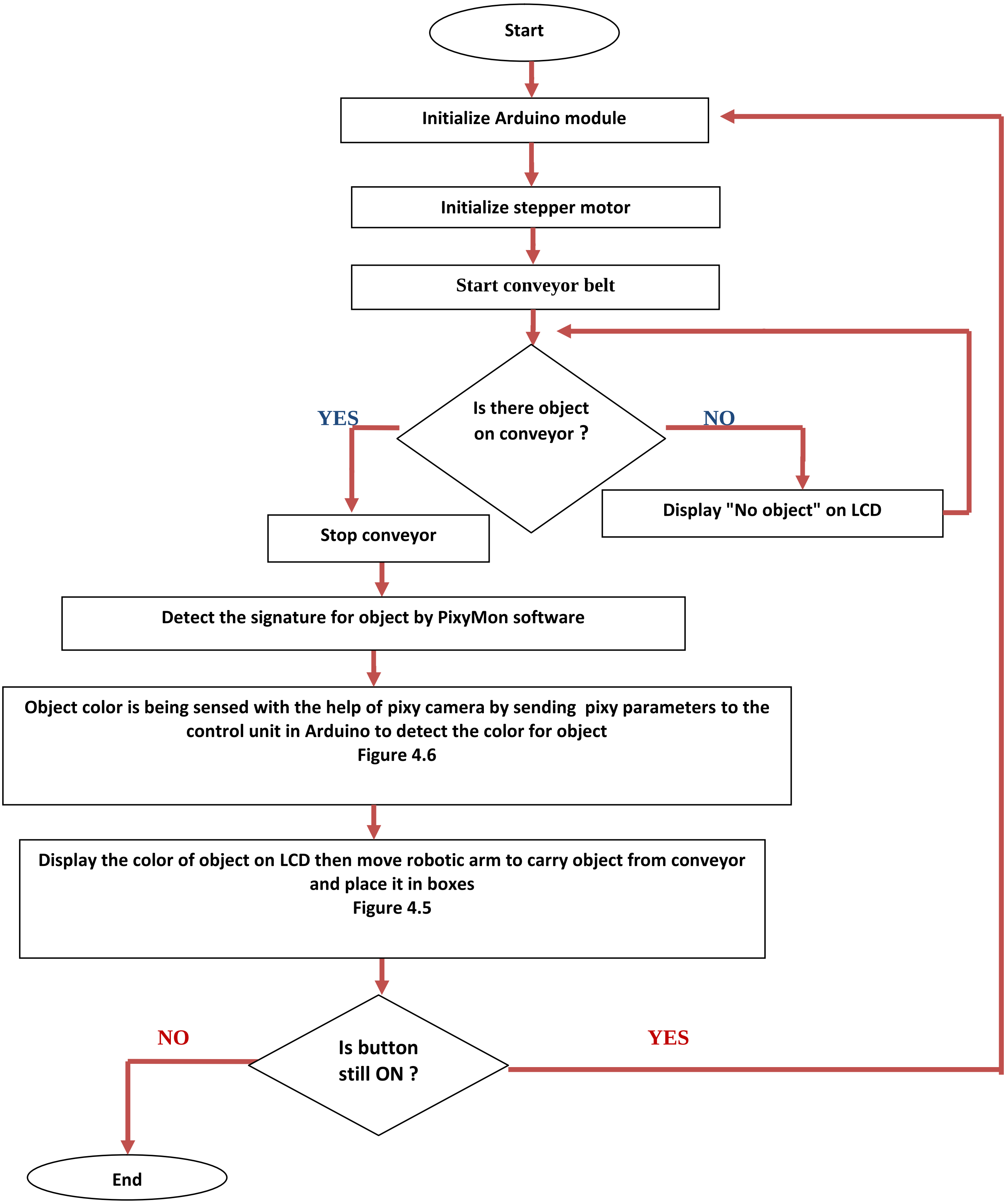
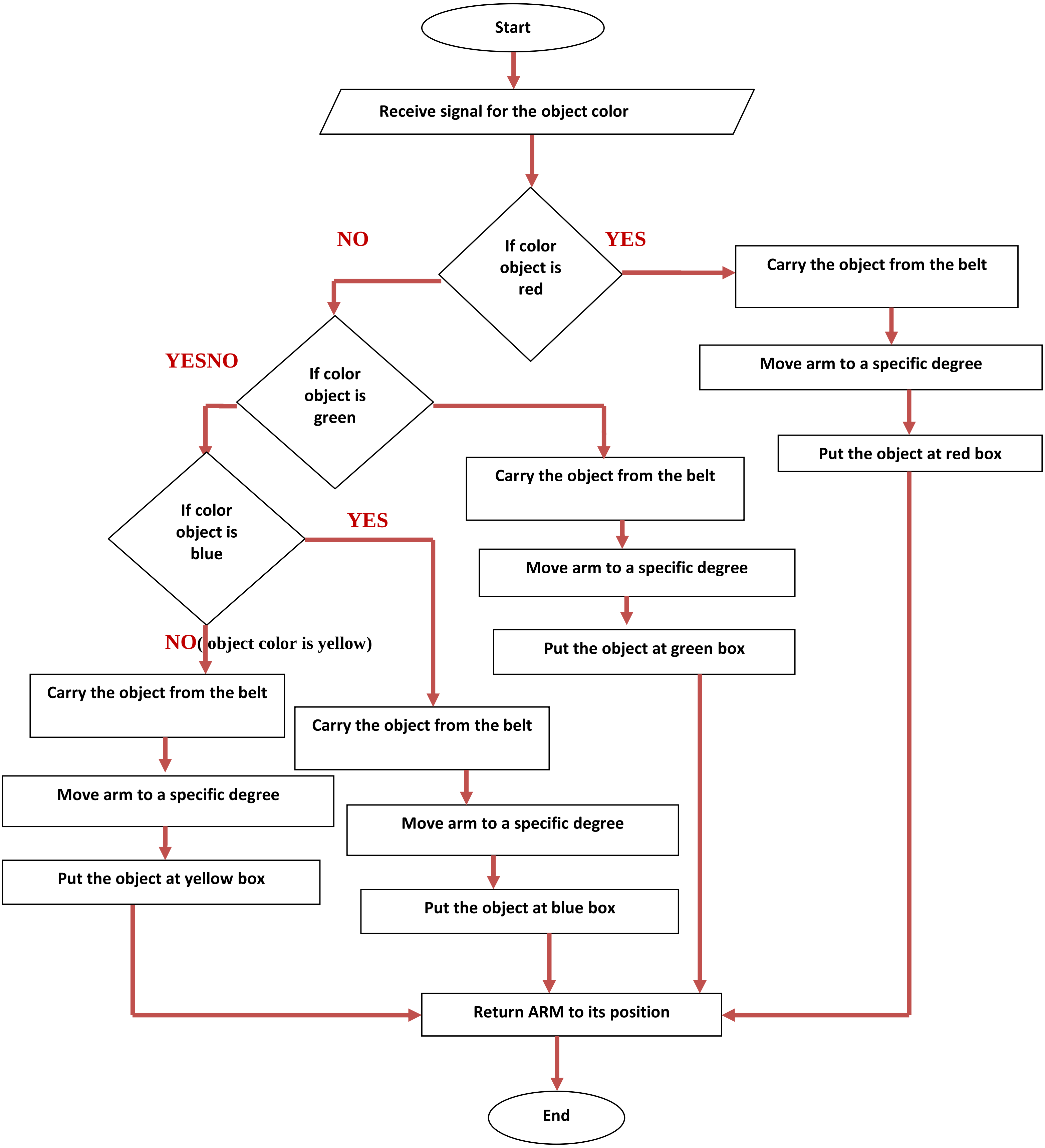

Start
Initialize Arduino module
Initialize stepper motor
Start conveyor belt
Is there object on conveyor ?
YES
NO
Display "No object" on LCD
Stop conveyor
Detect the signature for object by PixyMon software
Object color is being sensed with the help of pixy camera by sending pixy parameters to the control unit in Arduino to detect the color for object
Figure 4.6
Display the color of object on LCD then move robotic arm to carry object from conveyor and place it in boxes
Figure 4.5
Is button still ON ?
NO
YES
End

Figure 3.3: General flowchart diagram

Figure 3.3 show the flowchart of the procedure followed for sorting of objects based on their color. Initialize the arduino module. Then the stepper motor is initialized to rotate the belt and then the objects are placed which move along the belt. Camera is activated and conveyor belt will stop If any object is detected, after that the camera determine the signature of that object and detect the color of it, then make arm moving by a specific angle to grasp the object from conveyor belt and put it on the correct box then arduino make conveyor belt on , the process will repeat infinite number of times until the power turned off or the system does not have objects.

Flowchart for sorting :-



Start
Receive signal for the object color
If color object is red
NO
YES
Carry the object from the belt
Move arm to a specific degree
Put the object at red box
If color object is green
YESNO
Carry the object from the belt
Move arm to a specific degree
Put the object at green box
If color object is blue
YES
NO(object color is yellow)
Carry the object from the belt
Move arm to a specific degree
Put the object at yellow box
Carry the object from the belt
Move arm to a specific degree
Put the object at blue box
Return ARM to its position
End

Figure 3.4 : The flowchart diagram for sorting

Flow chart for color detection :-

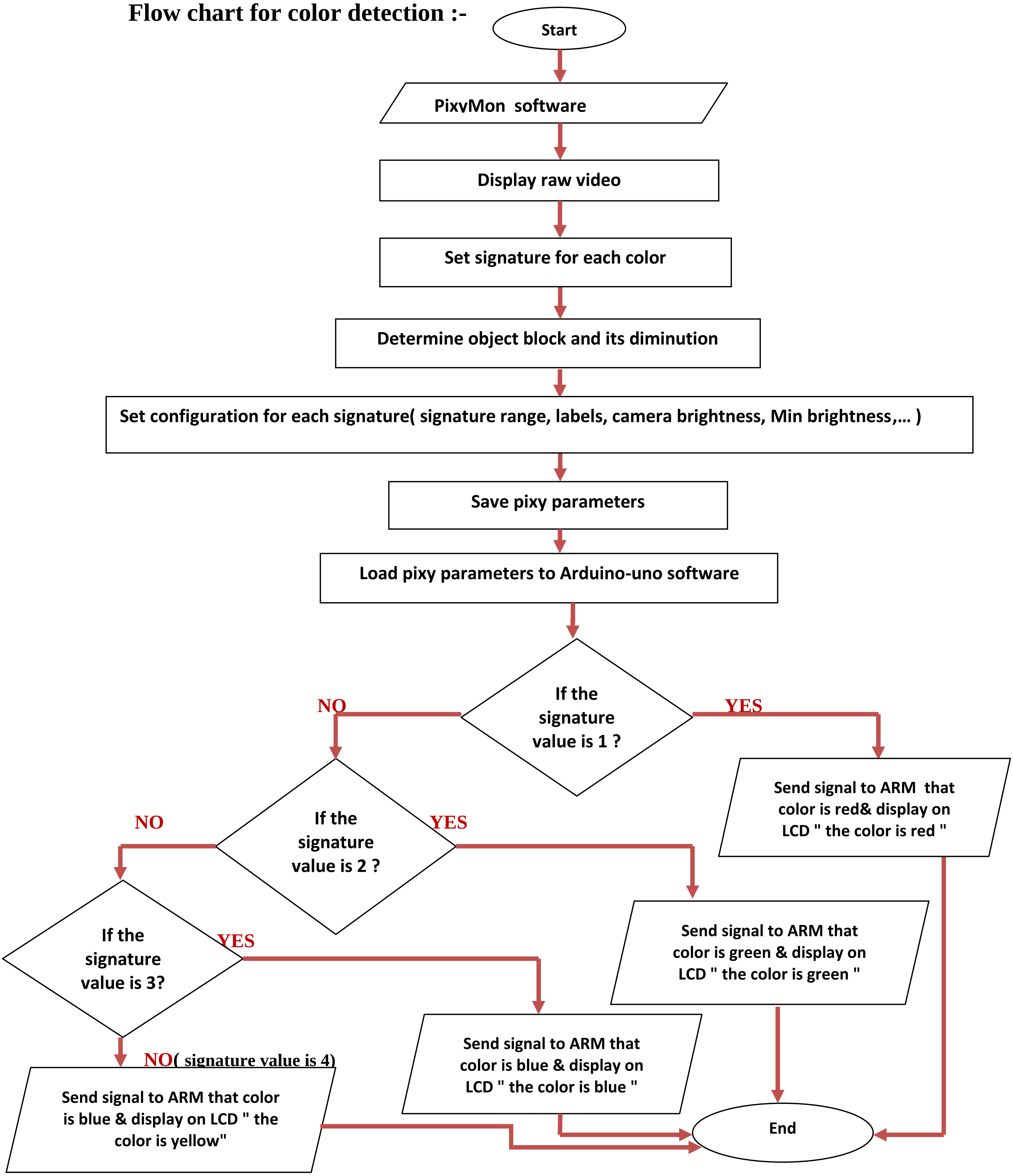

Start
PixyMon software
Display raw video
Set signature for each color
Determine object block and its diminution
Set configuration for each signature(signature range, labels, camera brightness, Min brightness,...)
Save pixy parameters
Load pixy parameters to Arduino-uno software
If the signature value is 1 ?
NO
YES
Send signal to ARM that color is red& display on LCD " the color is red "
If the signature value is 2 ?
NO
YES
Send signal to ARM that color is green & display on LCD " the color is green "
If the signature value is 3?
YES
Send signal to ARM that color is blue & display on LCD " the color is blue "
NO(signature value is 4)
Send signal to ARM that color is blue & display on LCD " the color is yellow"
End

Figure 3.5 : The flowchart diagram for color detection

3.3.5 Sequence Diagram

Figure 3.6 shows the sequence diagram of the whole system. As the figure shows, the operation starts when the conveyor belt start rotation around color object to reach it near the camera .

The camera will detect the signature of the object and send it to the arduino.

The arduinouno will determine the color of each object depending to the signature received. If the system have objects and the camera still detect block for object, arduinouno will send the movement command to the arm and the arm will grasp the object from conveyor then place it on the boxes and display the color on LCD until the first object sorted .The sorting syatem will end if objects not available, or the power turned off .

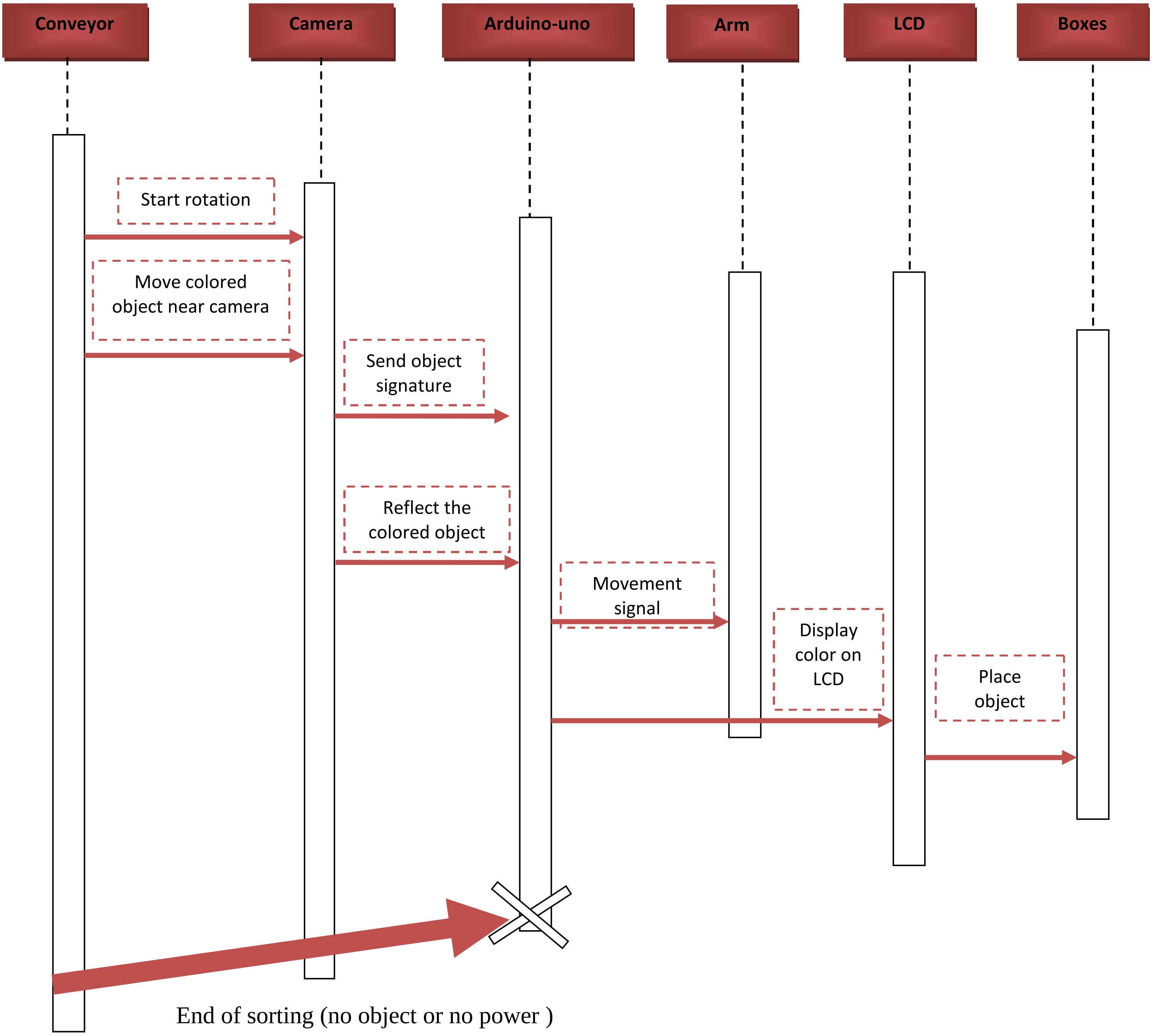


Figure 3.6 : The sequence diagram

3.3.6 Sub-Circuits

Figure 4.7 shows how each servo of the robot arm is connected with the servo motor shield. We connect ground of servo with ground pin of shield, Vcc of servo with 5V pin of shield, and the servo data pin with one of the output pins of shield.

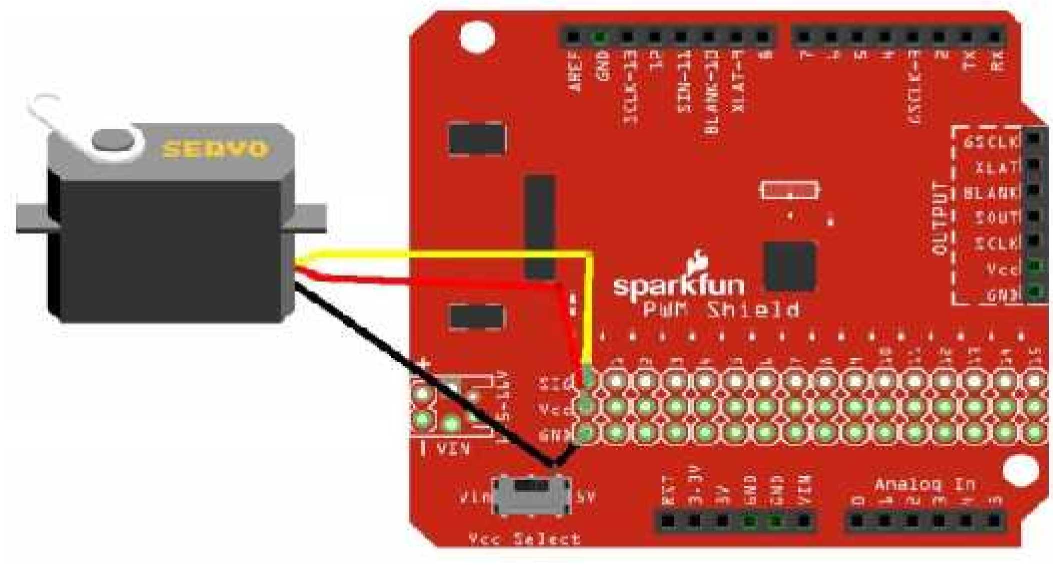


Figure 3.7 : The servo motor-PWM shield circuit.

● **LCD circuit:**

Figure 4.8 shows how LCD is connected with arduino UNO. We connect ground of LCD with ground pin of arduino, Vcc of LCD with 5V pin of arduino, data pins of LCD from D4 to D7 with arduino pins from pin 5 to pin 8, RS with pin 10, E with pin 9.

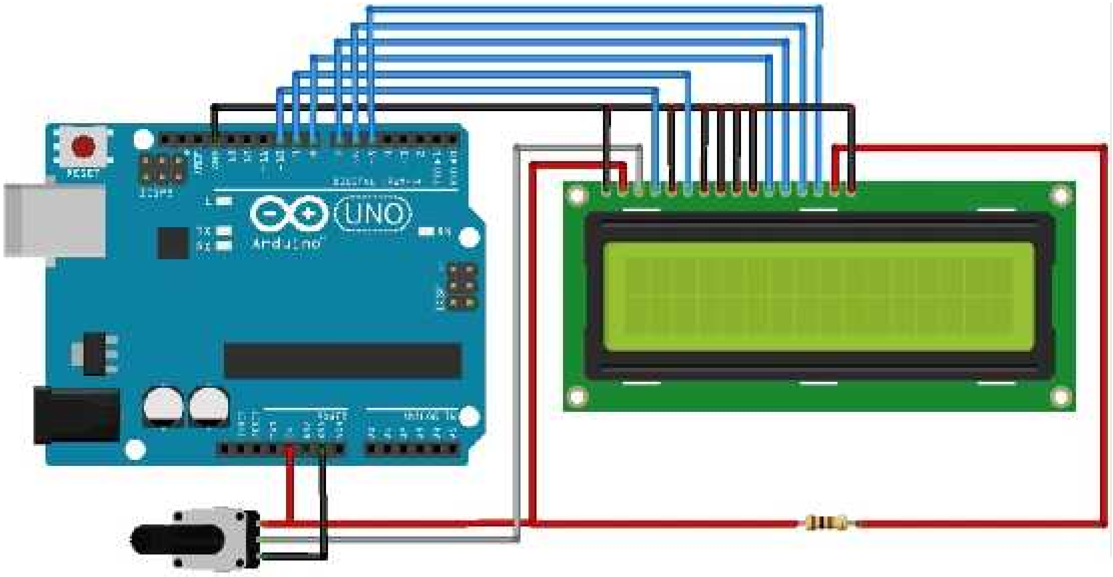


Figure 3.8 : The LCD circuit

- **Stepper motor circuit**

Figure 4.8 shows how stepper motor is connected with TB6560 driver. We connect ground and Vcc of driver with power supply,{A-,A+,B-,B+} of driver with stepper motor , step pins of driver with pin 5 of arduino, direction pins of driver with pin 2 of arduino ,enable pins of driver with pin 8 of arduino ,and others on GND of arduino.

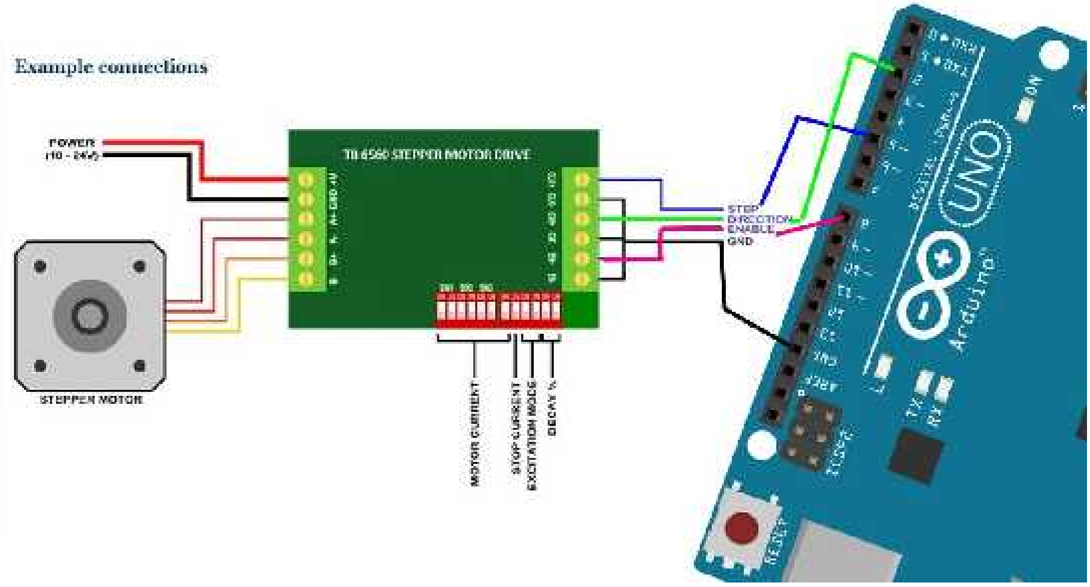


Figure 3.9 : The Stepper motor circuit

3.3.7 Schematic diagram

● Servo motor with PWM Shield schematic diagram

Figure 4.10 shows how each servo of the robot arm is connected with the servo motor shield. We connect ground of each servo with ground pin of shield, Vcc of each servo with 5V pin of shield, and the each servo data pin with one of the output pins of shield

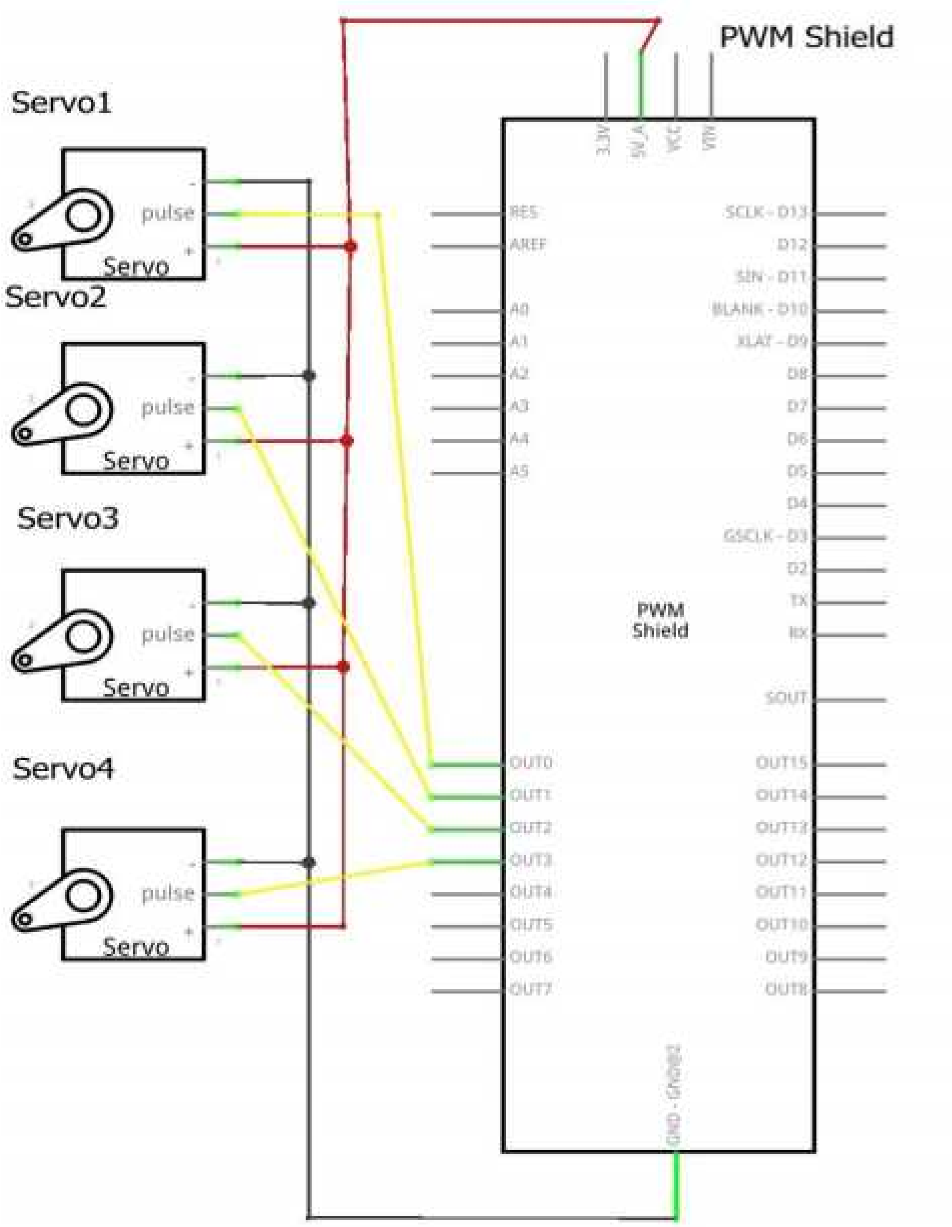


Figure 3.10 : The servo motor- PWM shield schematic diagram

● LCD schematic diagram

Figure 4.11 shows how LCD screen is connected to arduinouno. We do this by connecting the following pins: We connect ground of LCD with ground pin of arduino, Vcc of LCD with 5V pin of arduino, data pins of LCD from D4 to D7 with arduino pins from pin 5 to pin 8, RS with pin 10, E with pin 9.

Additionally, wire a 10k pot to +5V and GND, with it's wiper (output) to LCD screens VO pin (pin 3). A 100E is used to power the backlight of the display, usually on pin 15 and 16 of the LCD connector.

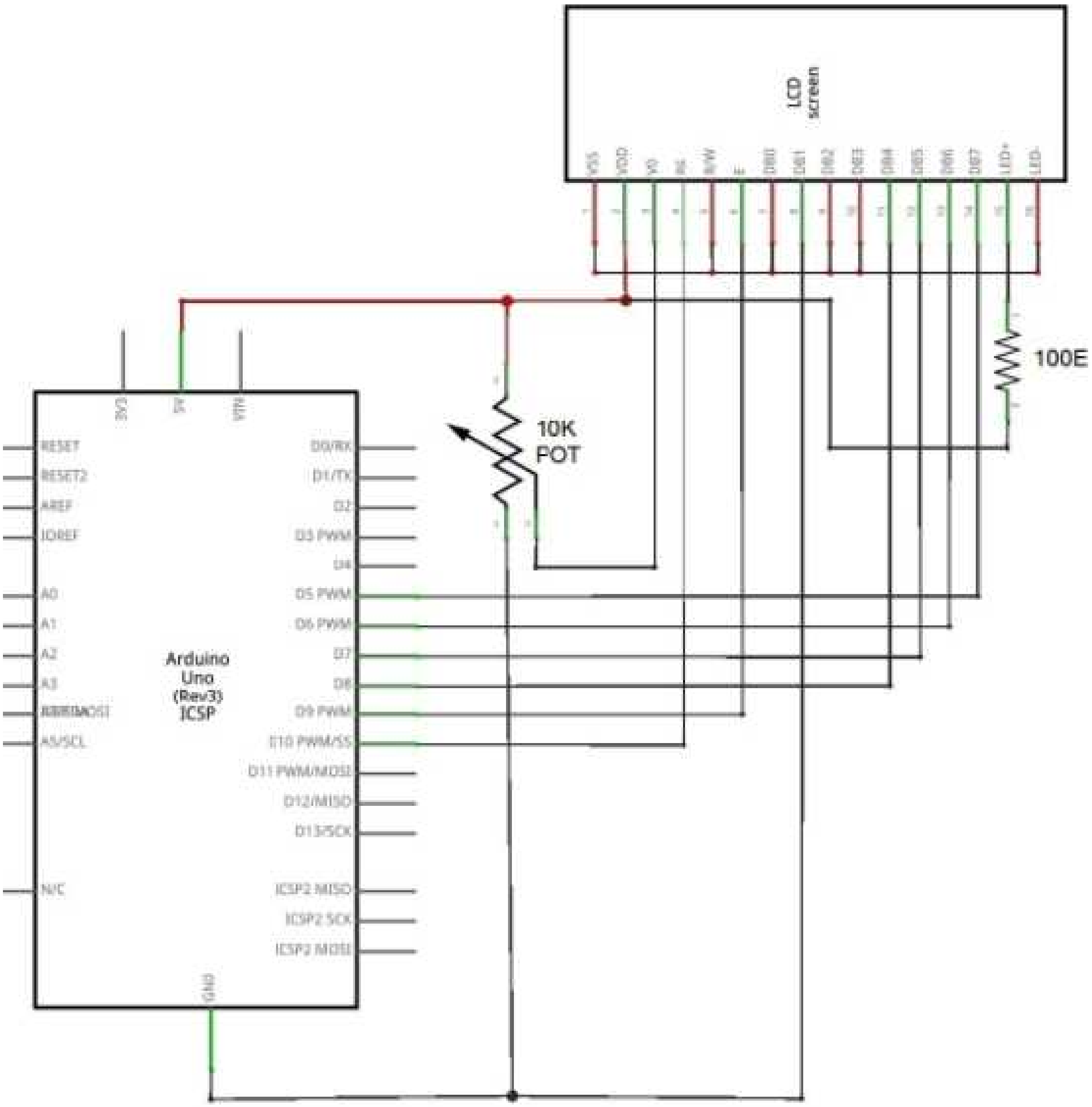


Figure 3.11 : The LCD-Arduino-uno schematic diagram.

- **Stepper motor schematic diagram**

Figure 4.8 shows how stepper motor is connected with TB6560 driver. We connect ground and Vcc of driver with power supply,{A-,A+,B-,B+} of driver with stepper motor.

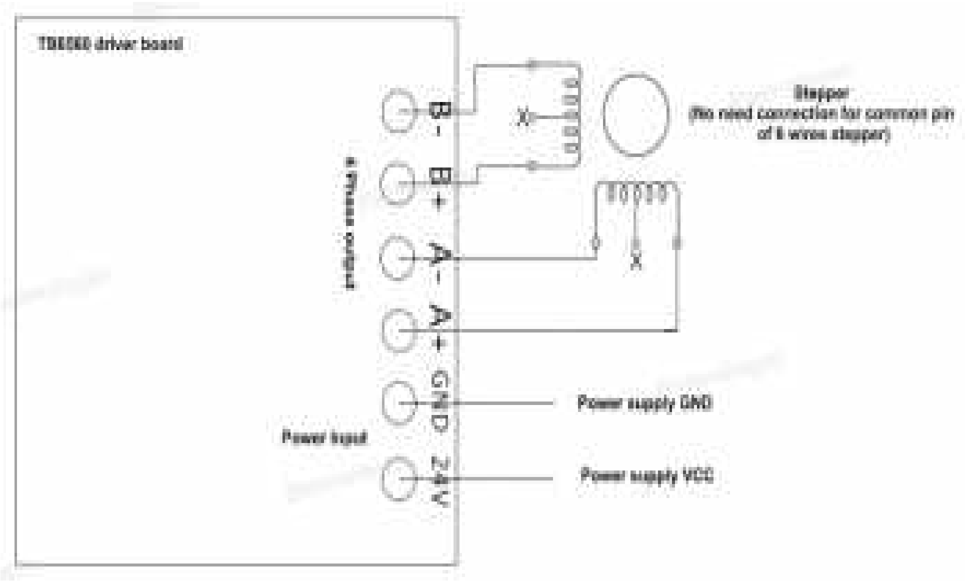


Figure 3.12 : Stepper motor schematic diagram

Chapter 4

Software & Hardware Implementation

4.1 Overview

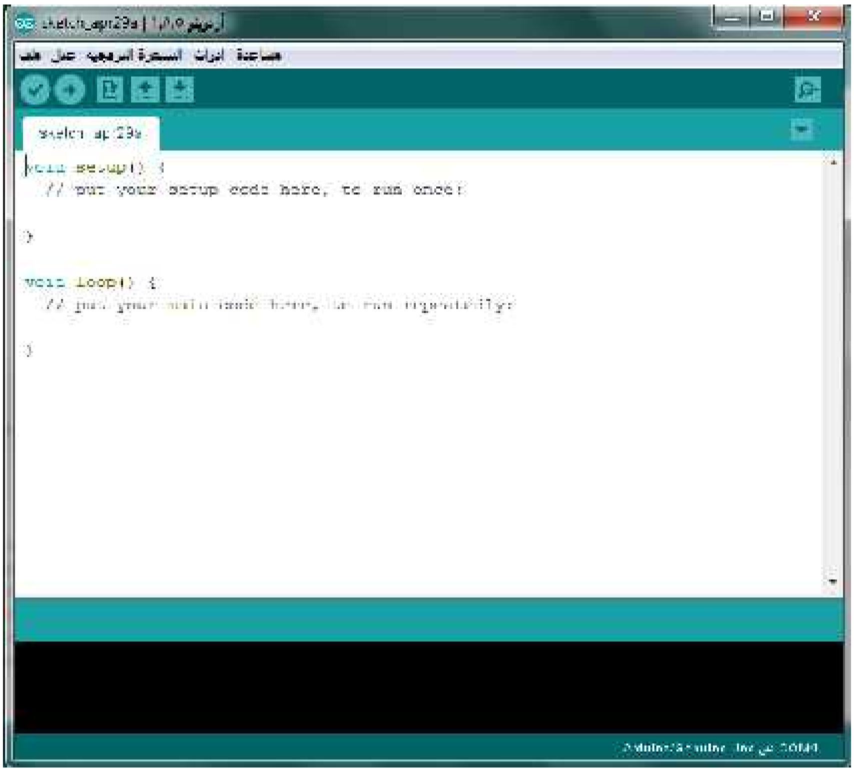
In this chapter, we described the implementation of software and hardware components that are used in our project.

4.2 Software Implementation Tools

This section will provide some information about the main programs and software technologies used in our project.

4.2.1 Arduino Software (Arduino IDE)

The Arduino Integrated Development Environment, or Arduino-IDE (Figure 4.1 show Arduino IDE) contains a text editor to write the code. Through this symbol we can say Arduino What should I do. After writing the code, we make sure there are no errors until the upload is done Program in the Arduino Council. In our project, we need Arduino code to send the arm Motion commands and store information about objects as you can see the camera, to check From the color of the object and move it to suitable place.



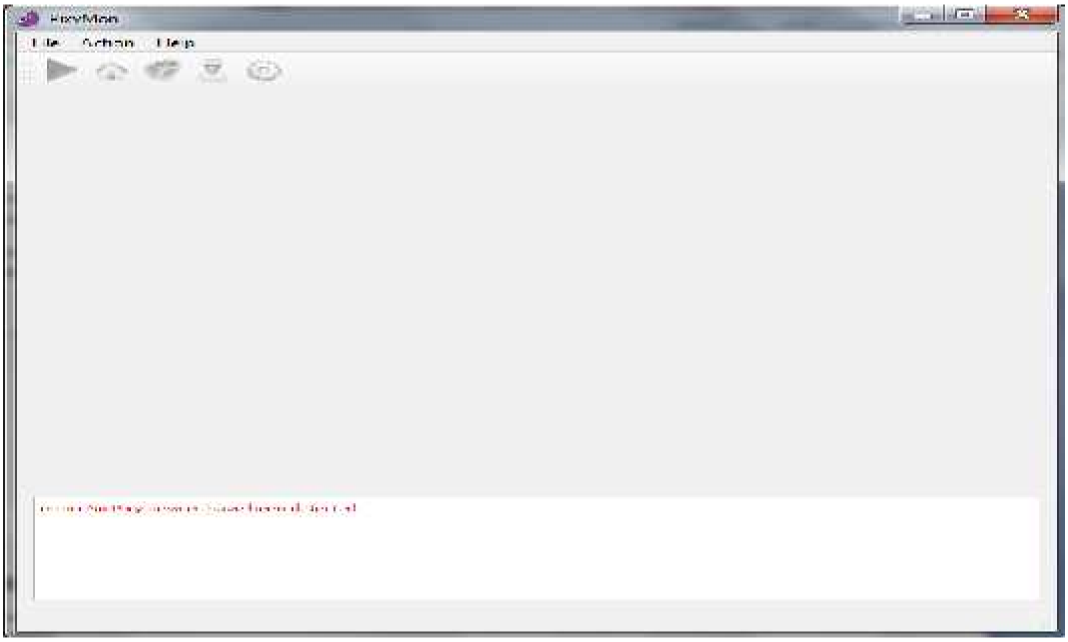
```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

Figure 4.1: Arduino IDE

4.2.2 PixyMon

“PixyMon is an application that allows you to configure Pixy and see what it sees. It runs on several different platforms including Windows, Mac OS and Linux, as well as other smaller embedded systems . It also allows you set the output port and manage color signatures. PixyMon communicates with Pixy over a standard mini USB cable”

This software was used in order to give each object a signature, and to teach the pixy camera the object in order to detect it later.



[illegible]

Figure 4.2: PixyMon

4.3 Hardware Implementation

This section provides some information about the hardware implementation done through our project .

4.3.1 Pixy CMU cam5 configuration

We used the camera in the project to identify the color by placing a signature for each color. Assisted by the smart color sensor.

The figure 4.3 shows a descriptive image of the connection of the camera with the arduino- Uno.

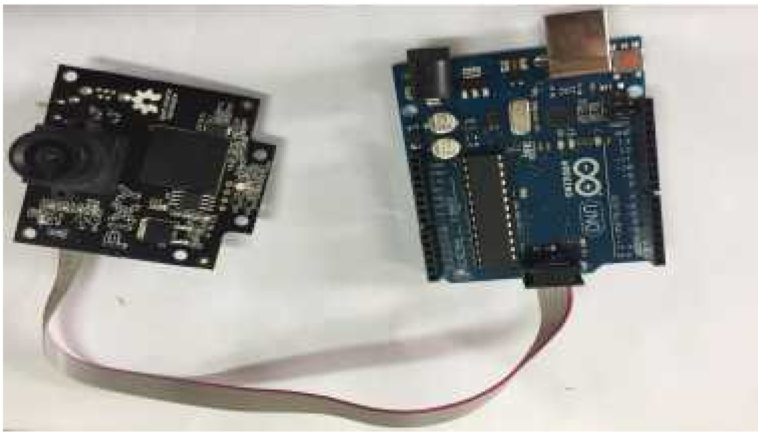


Figure 4.3: The connection of the camera with the arduinouno.

4.3.2 Robotic Arm configuration

The arm is one of the main parts of the system and is used to move the object from the conveyer to a suitable location at a certain angle after being categorized by color using the camera . The figure 4.4 shows a descriptive image of the connection of the arm servos with the arduinouno via arduino servo shield. We used only four servos from six servos because it is enough to move the arm and in smooth movement.

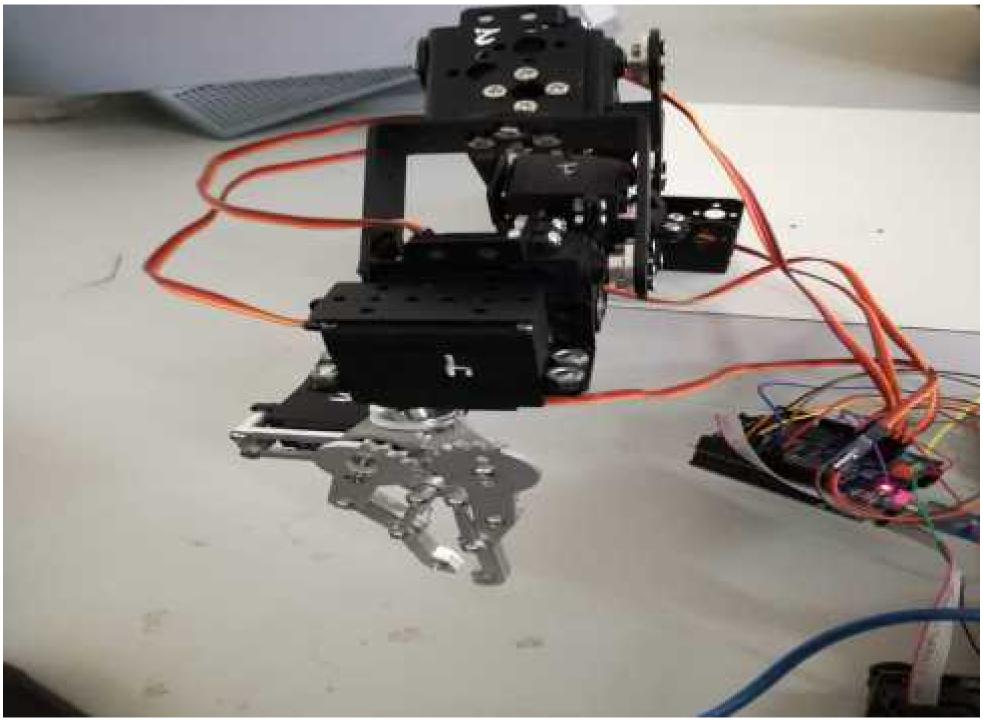


Figure 4.4: The connection of Arm servo motors with servo motor shield.

4.3.3 Conveyer configuration

The conveyer is one of parts of the system and is used to move the object from one place to another by stepper motor. The figure 4.5 show a descriptive image of moving the conveyer.

The figure 4.5 moving the conveyer for object.

4.3.4 LCD configuration

LCD is a useful part of the project where it was used to display color.

The figure 4.6 show a descriptive image of the connection of the LCD with the arduino-uno

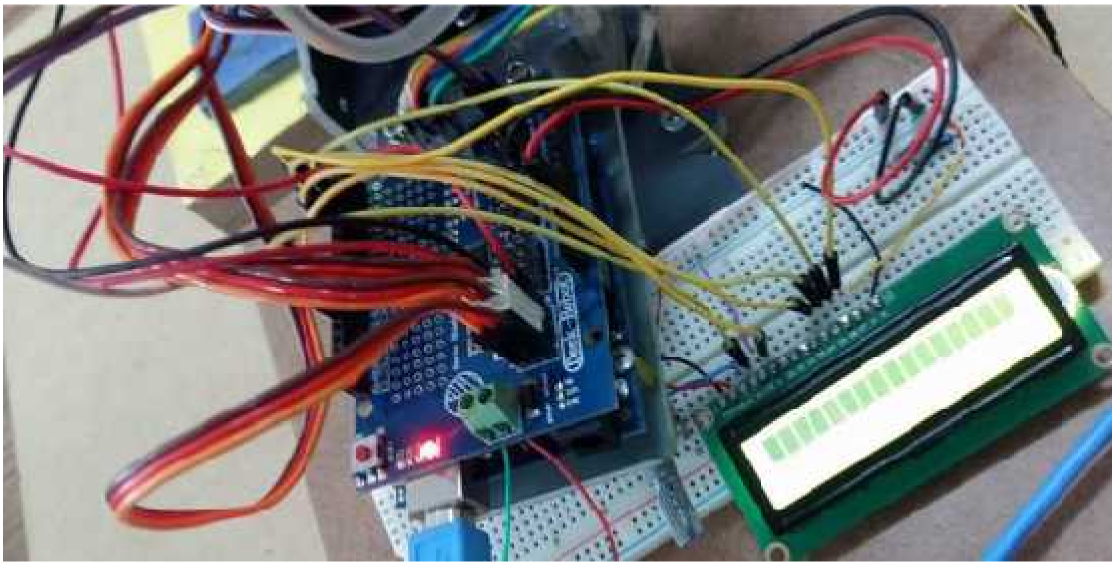


Figure 4.6: The connection of the lcd with the arduinouno.

4.3.5 All the system connected together

The figure 4.7 below shows the connection of all the components of the system together.

ع

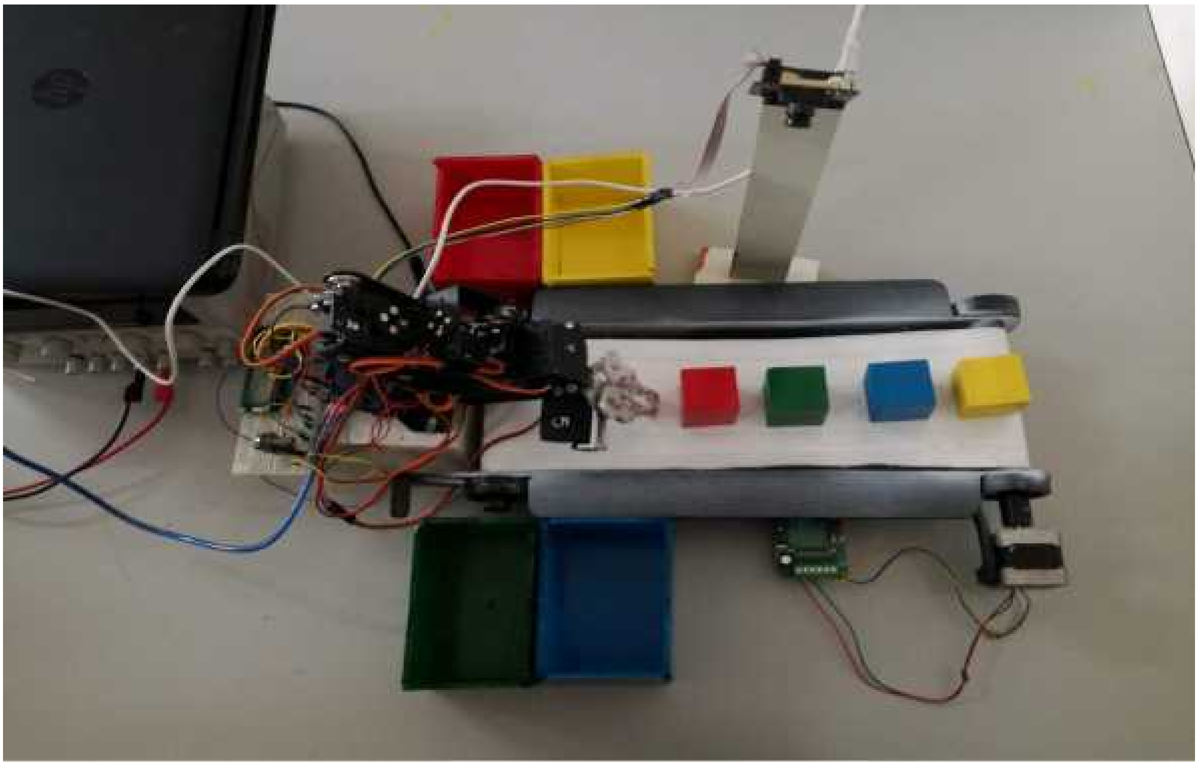


Figure 4.7: The connection of all the components of the system together.

Chapter 5
Testing

5.1 Overview

In this chapter we described and discussed the steps, results of the implementation and the testing of the system.

5.2 Units Testing

We started by testing all the parts to ensure that all of the functions works perfectly and without errors.

5.2.1 Hardware Testing

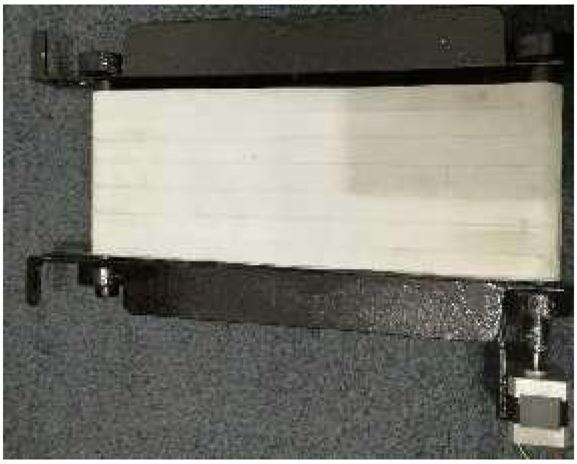
- Test connection of the stepper motor with the arduinouno
- Test connection of the arm with the arduinouno.
- Test connection of the camera with the arduinouno.
- Test connection of the lcd with the arduinouno.

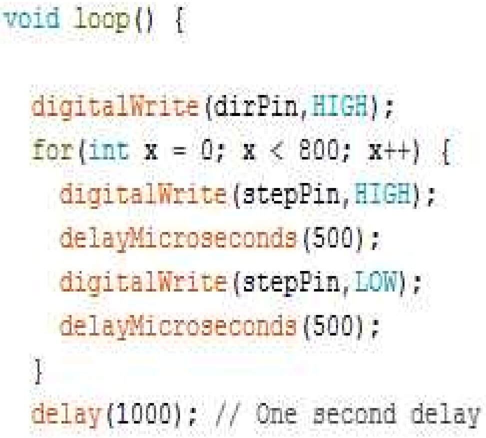
5.3 Integration Testing

After ensuring that all the parts are working well, we started assembling and integrating the parts with each other to make the system ready to operate.

5.3.1 Stepper motor test

The stepper motor was used to move the conveyer to move objects from one place to another. We used Stepper motor instead of DC motor because it is more accurate to stop in the right place.





```
void loop() {

  digitalWrite(dirPin,HIGH);
  for(int x = 0; x < 800; x++) {
    digitalWrite(stepPin,HIGH);
    delayMicroseconds(500);
    digitalWrite(stepPin,LOW);
    delayMicroseconds(500);
  }
  delay(1000); // One second delay
```

Figure 5.1: conveyer with stepper motor

After applying the code on the stepper motor, the conveyer was move as showen on figure 5.2

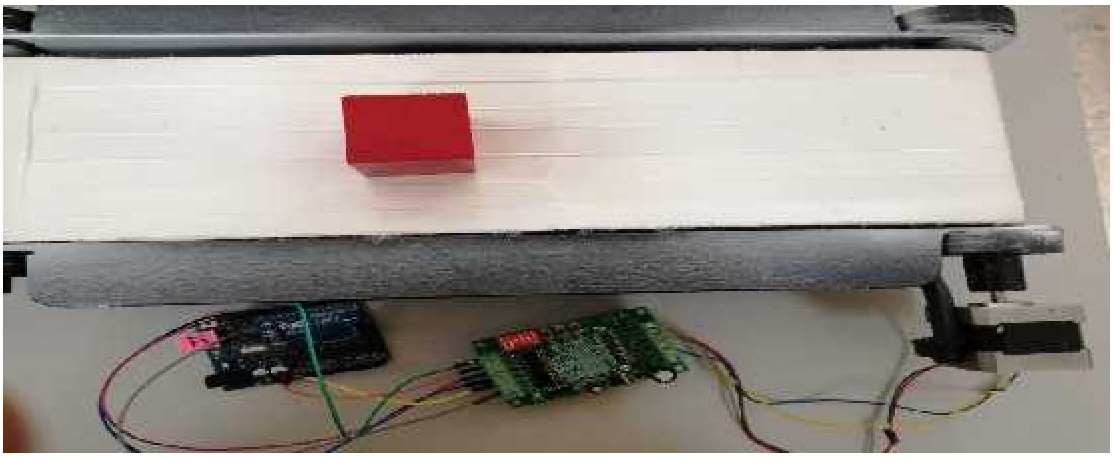


Figure 5.2: moving the conveyer for object.

5.3.2 Arm -Arduino UNO test

The figures 5.3 & 5.4 below show the gripper while it's picking the object from its location on the conveyer, and placing it on the suitable place on box consecutively. Figure 5.1 shows how the arm moved to object location, go down to enable its end effector from picking the object using gripper it close while the object holds by the gripper in order to move the object to specific box.

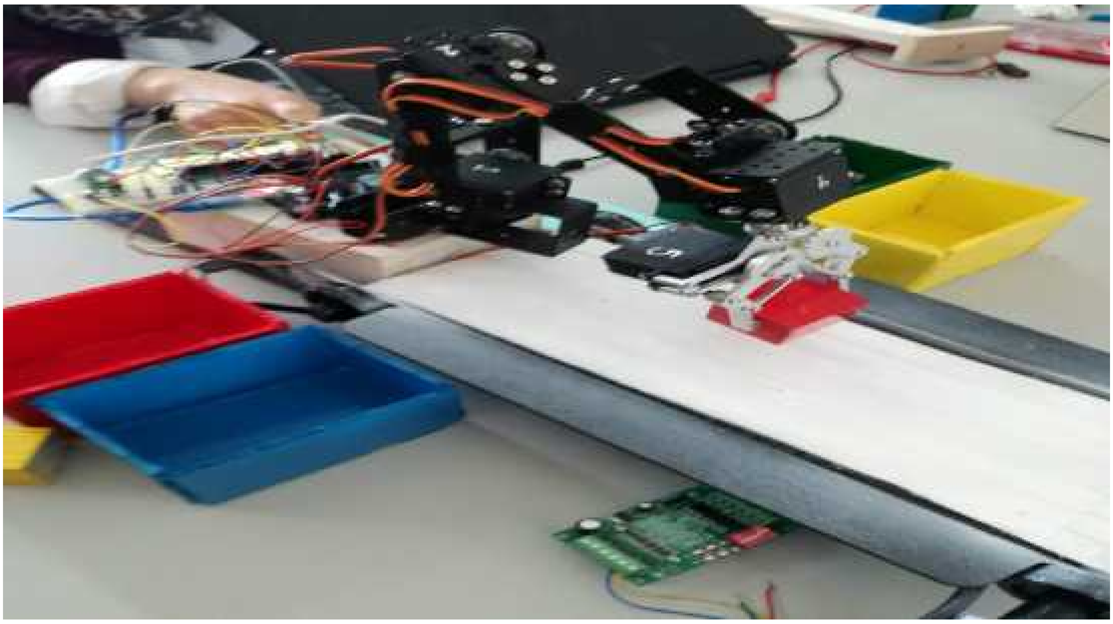


Figure 5.3 shows how the arm moved to specific box and put the object on this place.

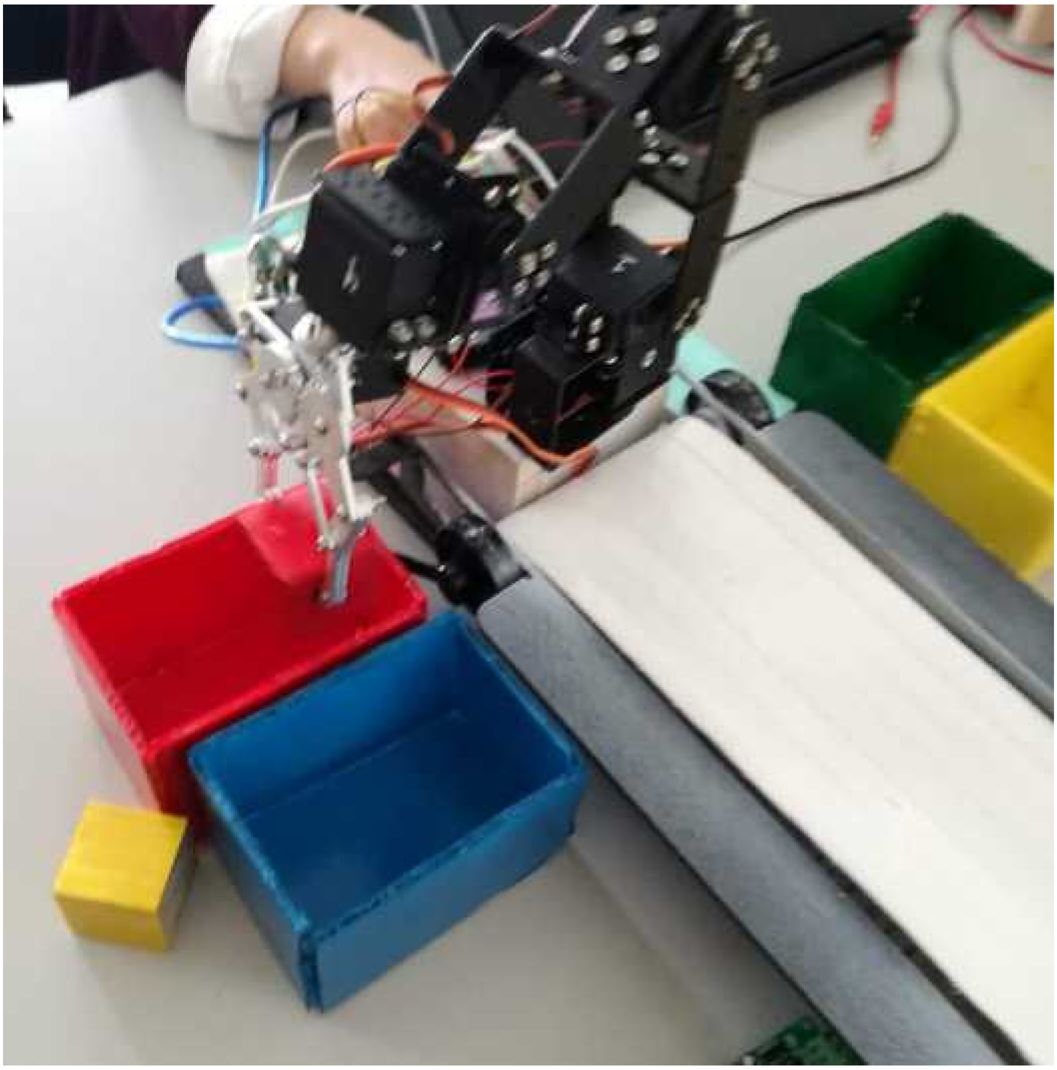


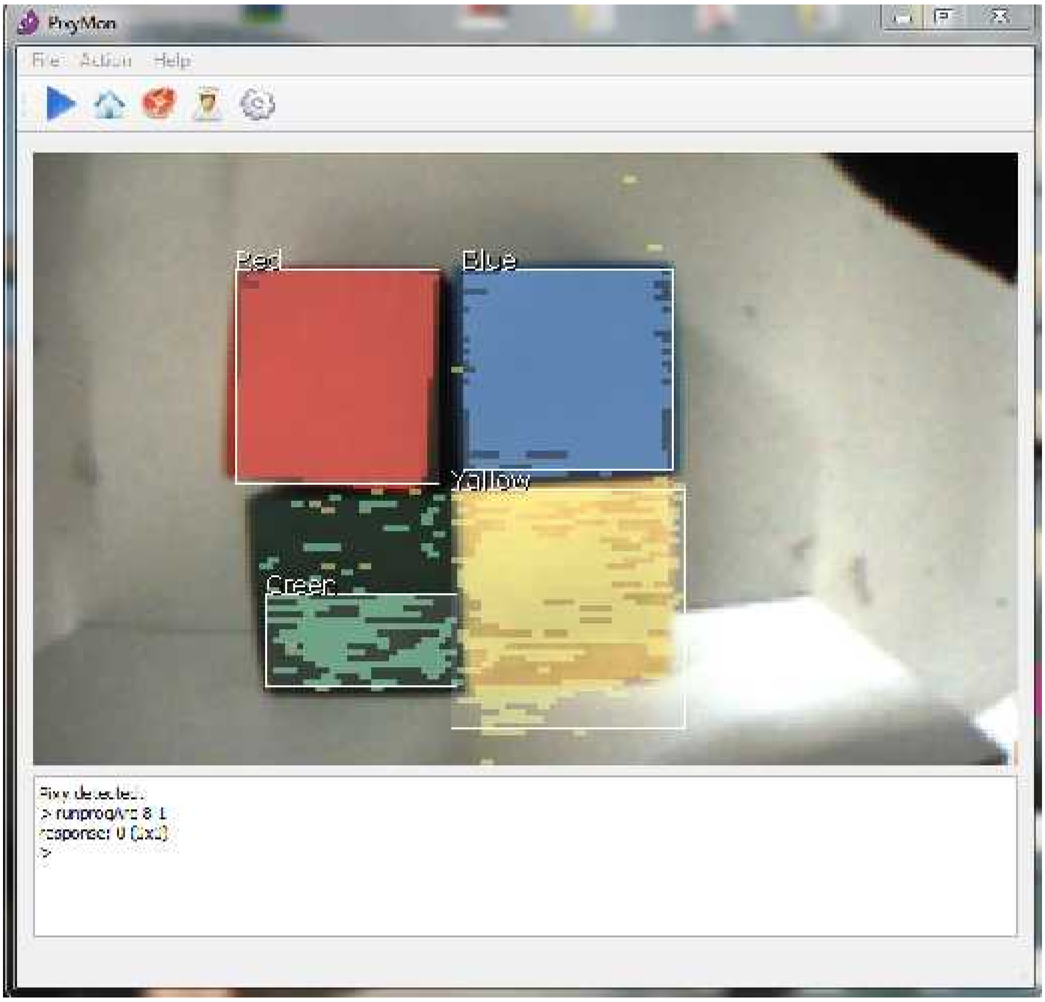
Figure 5.4 :The gripper of arm pick the object into specific box

5.3.3 Camera-Arduino UNO test

After the camera has been connected to the arduinouno, we tested it if it could detect the objects on the board.

The figures 5.5 below show the detected object by the camera, and the camera calibration process by finding the color and ignoring everything outside these coordinates

The figures 5.5 and 5.6 show the operation of determine in which square each object is placed and its signature in order to determine what is on each square of the board (red or green or blue and yellow objects), and in which square does this object exist.

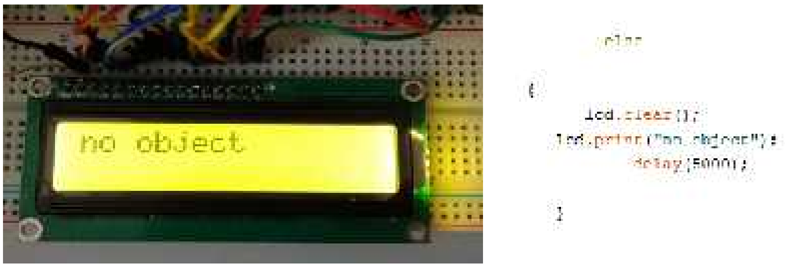


```
Pixy detected...
> runprogArg 8 1
response: 0 (0x0)
>
```

5.5: The detected objects by the cameraFigure

5.3.4 LCD- Arduino UNO test

These figures below show the lcd different tests. Figures 5.6 shows the status of conveyer if no object is moved.



```
            else

{
        lcd.clear();
    lcd.print("no object");
            delay(5000);

    }
```

Figure 5.6 The status of the conveyer if no object is moved.

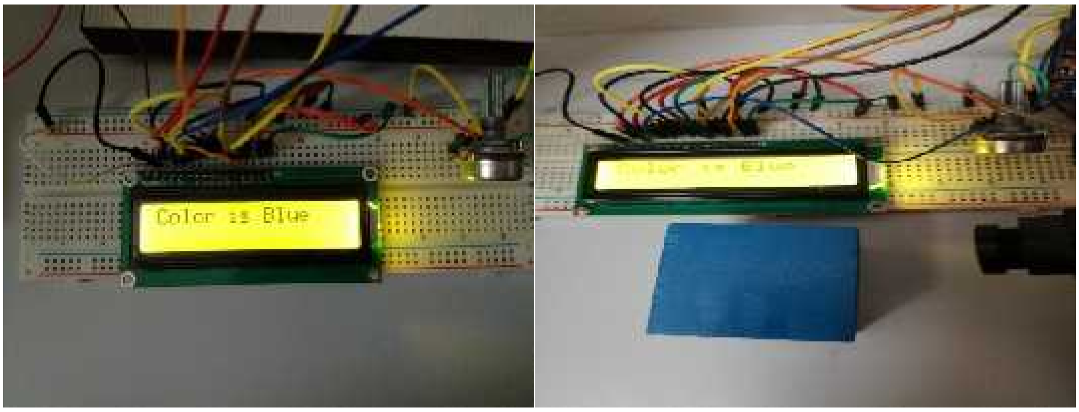
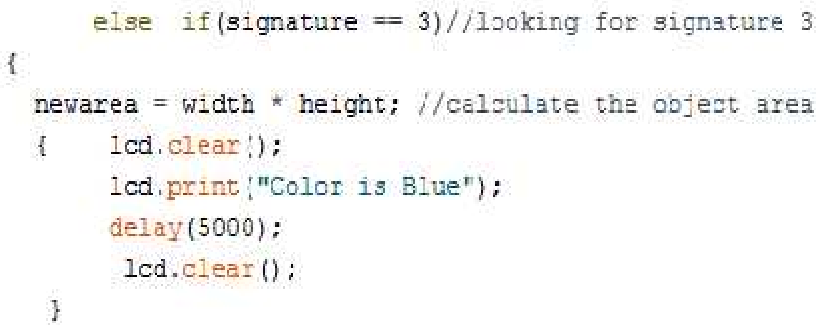


Figure 5.7 shows that the status of conveyer that have an objects blue color.



```
      else  if(signature == 3)//looking for signature 3
{
  newarea = width * height; //calculate the object area
  {     lcd.clear();
        lcd.print("Color is Blue");
        delay(5000);
         lcd.clear();
   }
```

Figure 5.8 code of blue color

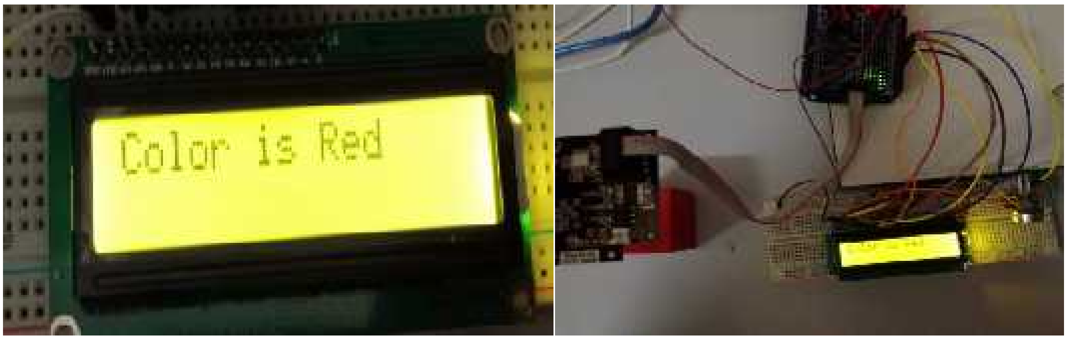
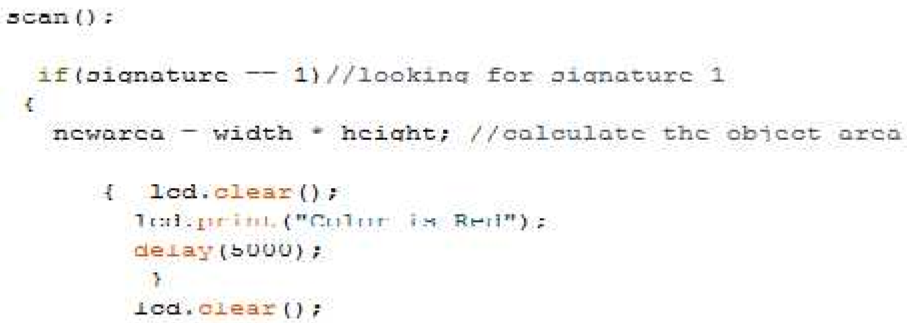


Figure 5.9 The status of the conveyer if the object is red.



```
scan();

  if(signature == 1)//looking for signature 1
 {
   newarea = width * height; //calculate the object area

      {  lcd.clear();
        lcd.print("Color is Red");
        delay(5000);
         }
        lcd.clear();
```

Figure 5.10 code of red color

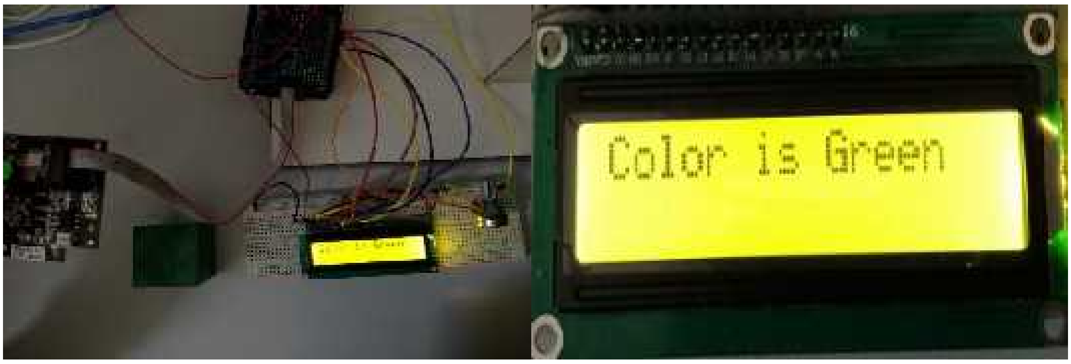
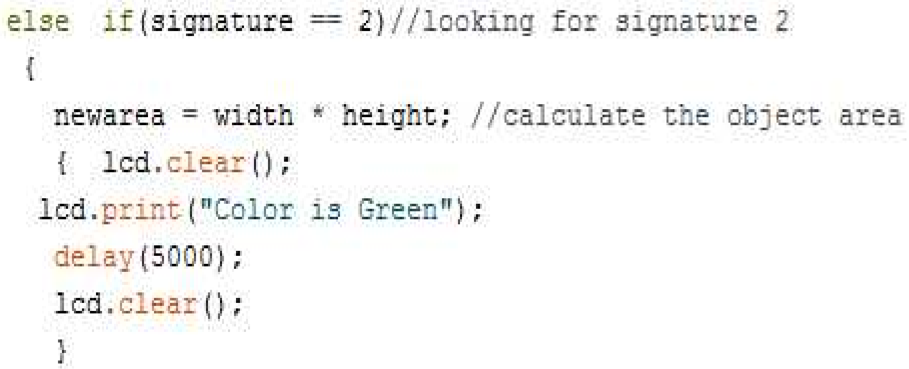


Figure 5.11 The status of the conveyer if the object is green.



```
else  if(signature == 2)//looking for signature 2
 {
   newarea = width * height; //calculate the object area
   {  lcd.clear();
  lcd.print("Color is Green");
   delay(5000);
   lcd.clear();
   }
```

Figure 5.12 code of green color

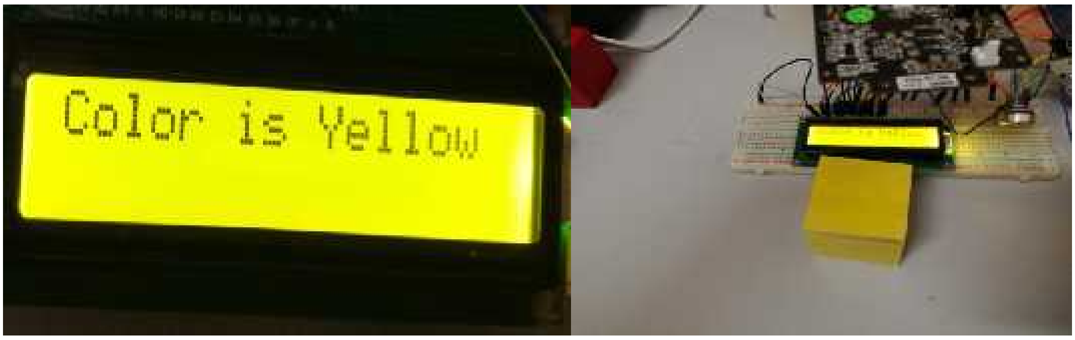
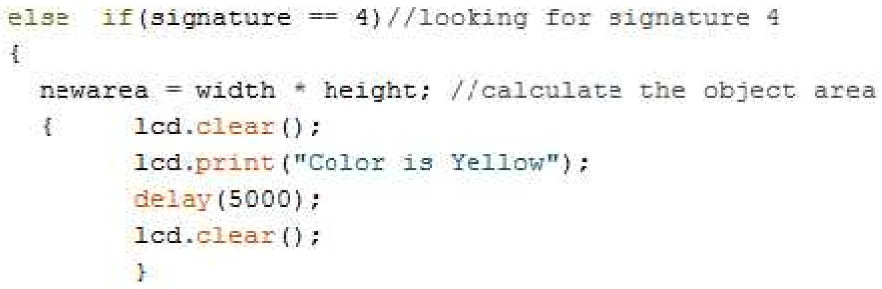


Figure 5.13 The status of the conveyer if the object is yellow



```
else  if(signature == 4)//looking for signature 4
{
  newarea = width * height; //calculate the object area
  {      lcd.clear();
         lcd.print("Color is Yellow");
         delay(5000);
         lcd.clear();
         }
```

Figure 5.14 code of yellow color

Chapter 6

Conclusion & Future Work

6.1 Overview

In this chapter, we will conclude the challenges, final results and future works of our project.

6.2 Implementation challenges and issues

We faced several problems while working on the project either on Software or Hardware.

6.2.1 Hardware Issues

We faced some hardware issues:
A: Problems when using the arm:-
1. We needed a servo shield to connect the four servos and use them at the same time. The shield had current leakage most of the time and that disturb our work.
2. Movement of arm not stable all the time, some time run successfully but another's time dose not run at the same source code .
3. Problem with suitable current and voltage for arm

B: Problems when using stepper motor for conveyor :-
1. We needed driver for stepper motor
2. Stepper motor some time does not bear conveyor load, this cause the conveyor stop rotating

6.2.2 Software Issues

1. We faced a problem with the camera, it detect more than one object , And filter pixels but we solve this problem in PixyMonsoftware .
2. Problem with ports in arduino-ide .

6.3 Final Results

This robot involves sorting of objects through camera , the future advancements can be done by increasing the efficiency of the camera . This method is verified to be highly beneficial for automated industries. The camera is key component of project which aides in distinguishing the objects. Failing of which may result in wrong material handling. So it becomes vital that the sensor in camera had a very high sense of sensitivity and ability to distinguish between colors. There is a reset button provided on the base which can used to restart the system from the beginning when there is stuck or hang situation. There are two main steps in color sensing part, objects detection and color recognition. This project is depicting the prototype of sorting systems which are used in industries.

6.4 Future Works

- Use an android application to control the sequence of the program
- Develop our robot to classified objects by using another's features in object like shape, size and metal .
- Make an application to send a message to smart phone devices showing the color and pronounce its name for people of special needs namely colorblind people.

References :-

[1]https://www.ijareeie.com/upload/2013/july/69_OBJECT

[2] Technology of Robotics, URL:

https://nocloudinthesky.wordpress.com/2013/01/17/technology-of-the-robotic-arm/

[3]Colour sensor based object sorting robot

https://www.irjet.net/archives/V4/i8/IRJET-V4I891.pdf

[4] Object Sorting by Robotic Arm Using Image Processing

https://www.irjet.net/archives/V3/i4/IRJET-V3I4291.pdf

[5] Dobot Robotic Arm - Magician - Basic Plan7, URL :
http://www.ebay.com/itm/Dobot-Robotic-Arm-Magician-Basic-Plan-/112288063310?hash=item 1a24e3df4e:g:MfsAAOSwU8hY55mu, **Nov 20,2017**

[6]Arduino Uno,URL:

https://www.arduino.cc/en/Main/ArduinoBoardUnoSMD

[7] Arm, URL:
http://www.ebay.com/itm/6DOF-Robot-Mechanical-Arm-Alloy-Robotics-Arm-Rack-with-Servo s-Arduino-Board-Kit-/172451205623, **Nov 12,2017**

[8] Camera, URL:
http://cmucam.org/projects/cmucam5,**Mar 22,2018**

[9] Adafruit 16-Channel 12-bit PWM/Servo Driver ShielD ,URL :

https://learn.adafruit.com/16-channel-pwm-servo-driver?view=all

[10]Stepper motor , URL :
https://en.wikipedia.org/wiki/Stepper_motor

[11] Arduino-IDE , URL :
https://www.arduino.cc/en/Guide/Environment

[**12**] PixyMon, URL:
https://media.digikey.com/pdf/Data%20Sheets/Seeed%20Technology/101990056_Web.pdf ,**April 2,2018**