INDUSTRIAL PRODUCT DISTINGUISHER WITH MODERN CONTROL SYSTEM

Submitted By:

MD. JOYNUL ABEDIN

ID: BME 1601008166

MD. ZAHID HASAN

ID: BME 1503007160

MD. MEHEDI HASAN

ID: BME 1502006028

MD. TAWHID ALI

ID: BME 1503007161

Supervised By:

MD. MAINOL HASAN

Lecturer

Department of Mechanical Engineering

Sonargaon University (SU)

Dhaka, Bangladesh

A thesis report submitted to the department of Mechanical Engineering, Sonargaon University in partial fulfillment of the requirements for the degree of Bachelor of Science in Mechanical Engineering.

DECLARATION

This is to certify that the thesis work entitled **‘‘Industrial Product Distinguisher with Modern Control System”** has been carried out by Md. Joynul Abedin, Md. Zahid Hasan, Mehedi Hasan, Tawhid Ali in the department of Mechanical Engineering, Sonargaon University, Bangladesh. The above thesis work or any part of this work has not been submitted anywhere for the award of any degree or diploma.

APPROVAL SHEET

This thesis entitled **‘INDUSTRIAL PRODUCT DISTINGUSHER WITH MODERN CONTROL SYSTEM’** was prepared and submitted as partial fulfillment of the requirements for the degree of B. Sc in Mechanical Engineering of Sonargaon University

Approved By:

..……………………

(Md. Mainol Hasan)

Date……………….

Lecturer

Department of Mechanical Engineering

Sonargaon University (SU)

ACKNOWLEDGEMENT

We would like to express gratefulness and humbleness to almighty Allah for his blessing upon us. We wish to express my greatest gratitude and profound respect to our supervisor Md. Mainol Hasan, Lecturer Department of Mechanical Engineering, Sonargaon University, who in spite of having a tight and busy schedule found time to go through our work, continuous guidance and has given valuable suggestions, encouragement, careful supervision, wisdom, positive criticism and invaluable advice throughout the course of this study and in the fulfillment of the work. We are fortunate enough that in spite of having a tight and busy schedule, our supervisor has managed time to help and guide us. For this again we express our gratitude to him.

We would like to thank Mostofa Hossain Sir, head of Department of Mechanical Engineering, Sonargaon University for giving the permission of using available labs and workshop facilities.

We also want to thank Abul Bashar Sir, Vice –chancellor, Sonargaon University for his assistance.

We are grateful to those teachers, staff members of related workshop and those who helped me directly and indirectly for the accomplishment of the project work.

Authors

ABSTRACT

For distinguishing object in industry optical sorting is very much convenient. Color and size are the most important features for accurate classification and distinguishing of product which can be done by using some optical sensors or analyzing their pictures. The color distinguishing machine is mainly a device that can sense the different color of the object and distinguish them in their right place. There is an unloading box to unload the product. When any object keeps on the unloading box the color sensor will sense the color of that product and then as the input devices color sensor will send signal to microcontroller where Arduino is a microcontroller will give command to stepper motor and servo motor to do action. The final result was quite satisfactory. The color detecting sensors worked well and it is able to detect blue, pink & green object quite nicely. The system performed well as programmed and detects the object according to their color.

CONTENTS

PAGE

CHAPTER I Introduction

CHAPTER II Literature Review

CHAPTER III Methodology

CHAPTER IV Hardware Analysis

CHAPTER V Software Analysis

CHAPTER VI Discussion& Conclusion

List of Figures

Figure No Description **Page**

CHAPTER I
INTRODUCTION

1.1 General

Over many years, around the world labors working in manufacturing industries have been losing their lives doing dangerous and risky jobs. Frequent injuries while doing repetitive works using big machines have been common especially in the developing countries. While analyzing and finding out reasons for cause of accidents in industries during production as well as in loading and unloading goods from production line to warehouse, experts have come up with some reasons which are: Lack of proper healthy working environments for labors, lack of proper safety precautions taken by owners of industries, lack of proper training to workers to operate machines and insufficient number of well-educated and skillful workers. Thus the present situations in industries have been remained worse and needs to be dealt properly. In many industries, such as: Soap manufacturing industries use various chemicals to prepare soaps, and bags containing these chemicals are carried by workers. This repetitive work has caused many workers to suffer from shoulder pain and also they gradually get infected by skin diseases. Hence, the realization to minimize sufferings of these workers has been the main motive to develop and build up a robot that can imitate this action of loading and unloading loads. Besides, we can see some existing system in industries that use trolleys to carry loads from one place to another. These trolleys are run by humans. However, the problem arises as these are hazardous tasks for humans, takes a lot of time to shift goods from one place to another and requires a lot of space. Such existing system can be improved by the development of technology and use fully automated system in industries replacing additional unskilled workers. It saves time taken for loading and unloading, efficiency of performing the task can be increased, and risking human lives can be reduced. Color sensor systems are increasingly being used in automated applications to detect automation errors and monitor quality at the speed of production line. They are used in assembly lines to identify and classify products by color in any industry. [1]. The objectives of their usage include to check the quality of products, to facilitate sorting and packaging, to assess the equality of products in storage, and to monitor waste products. These unpleasant boredom and tiring jobs can be easily performed by this machine as the machine is untiring. Thus cost of production can also be increased and cost of manufacturing goods can be reduced due to use of less labors.

1.2 Executive Summary

Machines can perform highly repetitive tasks better than humans. Worker fatigue on assembly lines can result in reduced performance, and cause challenges in maintaining product quality. An employee who has been performing an inspection task over and over again may eventually fail to recognize the color or shape of the product. Automating many of the tasks in the industries may help to improve the efficiency of manufacturing system. The purpose of this model is to design and implement a system which automatically distinguishes the products based on their color. This machine consists of stepper motor, lead screw, smooth rod, color sensor, and servo motor. The output and input of these parts have interfaced using Arduino microcontroller. To reduce human efforts on mechanical maneuvering different types of distinguishing machines are being developed. A common requirement in the field of color distinguishing is that of color sensing and identification.

1.3 Objectives of Thesis

Any kinds of automated program which have been designed to perform various tasks, which are complex, at a faster speed and with greater efficiency than human beings. This paper focuses on designing and developing an automatic product distinguisher which has been instructed to follow a pre-defined path starting from a particular staring point to the destination point.

- The main objective of this project is to design & construct color distinguishing system.
- To design & construct accurate color identification system.
- To reduce human error & extra man power.
- To reduce human efforts & any kinds of accident.

1.4 Significance of the project

Industrial automation presents today's manufacturers with a host of unbeatable benefits. The entire purpose behind industrial automation is to increase productivity and quality. Our Industrial automation systems allows companies to achieve faster cycle times, greater efficiency, and repeatability. Automation offer a quick return on investment via dramatic increases in productivity and efficiency. Industrial automation not only simplifies labor-intensive tasks, thereby reducing workforce costs significantly, it also minimizes the production hours. Robots and other types of industrial automation are capable of providing consistent, repeatable results. When manufacturers utilize industrial automation, they eliminate the quality control issues involved with human error. With industrial automation, processes can be carefully regulated and controlled, so the quality of the end product is not only reliable it is often vastly improved. So, whether the company is a manufacturer looking to create exact, consistent welds each and every time, or a pharmaceutical company that requires hygienic, clean room results each and every time, industrial automation provides consistent, reliable results. Industrial automation effectively improves workplace safety and protects workers from injury. Automation systems can endure extreme work environments and work around hazardous substances. [1].

CHAPTER II

LITERATURE REVIEW

2.1 General

Though we tend to think of automation as a modern phenomenon, it's history is quite extensive. Automation in the manufacturing industry has evolved from the use of basic hydraulic and pneumatic systems to today's modern robots. Most industrial operations are automated with the goal of boosting production and reducing the cost of labor. Since its inception, industrial automation has made great advances among activities that were previously carried out manually. A manufacturing organization that uses the latest technologies to fully automate its processes typically sees improved efficiency, production of high-quality products and reduced labor and production costs. Industrial automation in manufacturing entails the use of machines to carry out manufacturing processes with levels of speed, consistency, stamina, and precision beyond the capacity of a human worker. The machines can be powered using a multitude of methods including electrical, hydraulic, mechanical, pneumatic and computer.

The main benefits of manufacturing automation include reduced production costs, improved quality and reliability, and decreased waste.

2.2 The early years of automation

In 1913, Ford Motor Company introduced a car production assembly line which is considered one of the pioneer types of automation in the manufacturing industry. Before then, a car was built by a team of skilled and unskilled workers. Production automation improved Ford's production rates and increased its profits. The assembly line and mass car production were the first of their kind globally. It reduced the car assembly time from 12 hours per car to about one and a half hours per car.

The world automation, in the manufacturing sense, was coined by Ford Motor Co. Vice President Delmer S. Harder in 1948. However, automation can trace its roots back much further than that. 762 B.C. to be exact(ish). [2].

Since 1970, ACORD, a not-for-profit standards-setting association for the insurance industry, has been involved in automation. The association is comprised of carriers, agents, vendors, solution providers, associations and other interested parties. “We aren’t the ones who ‘built’ the automation system,” said Carolyn “Cal” Durland, managing director of Standards for ACORD. “What we did and still do is provide standards-Forms and Electronic Data Interchange (EDI) upon which the vendors or solution providers base their automation systems.” She explained that to find out how the industry became automated, one must look to the vendors and solution providers. [2].

In 1972, the first ACORD form, a property loss notice, went into use. Today, ACORD’s standards include Forms, AL3 (Automation Level 3), XML, OLife and ObjX. The Forms are point of sale, data collection vehicles, AL3 is ACORD’s EDI standard (or machine-to-machine, business-to-business, data transmission formats or components), OLife is a data integration standard and ObjX is “much more than EDI.”

In the 1980s-when the number of PCs in use increased dramatically-ACORD members asked for standardized electronic transmissions between the agents’ computers and the carriers’ computers.

2.3 Advancement of automation in the 20th century

During the 1930s, Japan was a leader in developing components that facilitated industrial manufacturing automation. One company developed the first micro-switch, protective relays, and a highly accurate electrical timer. By this time, the rest of the world had started appreciating automation and significant research and development had occurred, like the solid-state proximity switch. Between 1939 and 1945 during the Second World War, automation was highly used in fighter airplanes, landing crafts, warships, and tanks. Japan surrendered to the US and allied forces in 1945 and an industrial rebuilding program was initiated. The program relied on new and superior technologies as opposed to the old-fashioned manufacturing methods that were being used by the rest of the world. Hence, Japan became the world leader in industrial automation. Automobile companies like Honda, Toyota, and Nissan could produce numerous high quality and reliable cars. They offered standard features that were classified as extras by other car manufacturers as well as competitive prices that triggered the success of Japan’s automobile industry.

2.4 The current state of industrial automation in the manufacturing industry

The current industrial robots feature high-quality computing capabilities, improved operational degrees of freedom, and vision systems. However, they can only operate in highly structured environments and still require a certain level of human intervention. Moreover, they are quite inflexible and highly-specialized for use in small and medium-sized industries, so industrial automation is usually better suited to large manufacturers and long production runs. Automation in the manufacturing industry relies on computer and software capabilities to automate, integrate and optimize different components of manufacturing systems. As a result, it is also referred to as computer integrated manufacturing.

Thanks to the inception and evolution of industrial automation in manufacturing industries, the world enjoys high-quality products and better energy, resource, and raw materials utilization. Contrary to what most people believe, the manufacturing industry is set to create more jobs with the use of robots, which will continue to drive the operations and benefits of industrial automation. [2].

CHAPTER III

METHODOLOGY

3.1 General

Now-a-days, there is general trend of industrial automation which is nothing but the process of handling different industrial processes automatically with the help of various machines such as computers and robots i.e. with the help of microcontrollers and sensors. This approach increases efficiency, productivity and decreases the human efforts and labor cost. Food industry is also incorporating this same approach of industrial automation in processing techniques to reduce production time and labor cost for enhancing quality of the produce and profit. A common requirement in the field of color distinguishing is that of color sensing and identification. [3].

3.2 Working Process

- Creating an idea for Design and construction to distinguish the industrial product with different Color Identification.
- Then designing a block diagram & circuit diagram to know which components need to construct it.
- Collecting the all components and programming for the microcontroller to controlled the system.
- Setting all components in a PCB board & soldering. Then assembling the all block in a board and finally run the system & checking. [4].

3.3 List of Necessery Components with Price:

Sl. No	Particulars	Specification	Unit	Qty.	Unit Price (Taka)	Total Price (Taka)
1	Arduino Mega	Mega 2560	PC	1	690	690
2	Stepper Motor	4 Wire Bipolar	PC	2	1000	2000
3	Stepper Motor Driver	A4988	PC	2	200	400
4	Color Sensor	TCS3200	PC	1	500	500
5	Servo Motor	S8503 (30kg)	PC	2	2000	4000
6	SMPS Power Supply	S-250-12 12 v 20 A	PC	1	700	700
7	Lead Screw 600 mm	8 mm pitch	PC	2	600	1200
8	Flexible Coupling	5/8 mm	PC	2	300	600
9	Linear Ball Bearing	LM8UU 8 mm	PC	8	250	2000
10	Smooth Rod 8 mm dia	600 mm		4	400	1600
11	Plastic Board	(8’*4’)	PC	1	2500	2500
12	Acrylic Board	(8’*4’)	PC	1	2500	2500
13	Others					2000
					Total	20,690/=

3.4 Development of the Mechanical Structure

At first a wooden block (plastic board) was chosen to make the base of the project. A square shaped wooden block30 inches length and 30 inches width was selected for the base of the structure. One piece of acrylic board was chosen to make the stepper motor base. Size is 680 mm in length and width is 125 mm. One lead screw with stepper motor & coupling and two smooth rods have been used, same products also have been used in the acrylic board. This two-stepper motor will give the unloading box 'X' & 'Y' axis movement. This two-stepper motor installed one upon another. One stepper motor, one lead screw and two smooth rods have been installed on the plastic board base. One side of lead screw is connected with the motor via a coupling. And other side of the lead screw has been placed in a bearing. Lead screw has been installed between two smooth rods. Same design has been made for the acrylic board which is placed upon the lead screw of the base. An unloading box has been placed on the acrylic board (width 125 mm) with two servo motor which will give the unloading box the required inclined motion to deliver the products and will give a 90° rotation to take the product. And the two-stepper motor will give the unloading box left, right, forward & backward movement with the help of lead screw and smooth rod. There is a color sensor which has been placed in the unloading box to detect the color of product. After getting sense the unloading box will starts to move to its pre-decide location for unloading the product. Total project is based on the automation system.

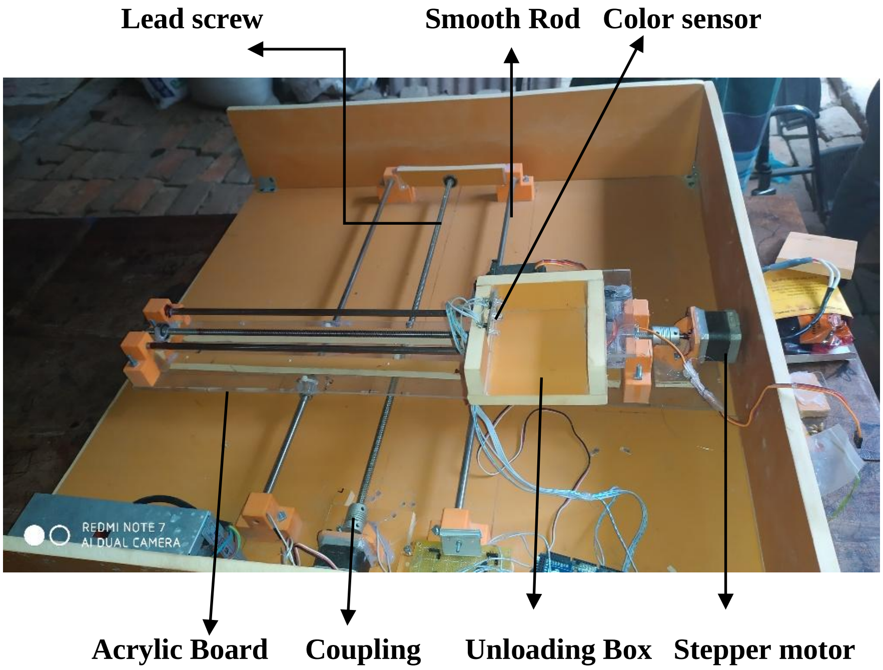


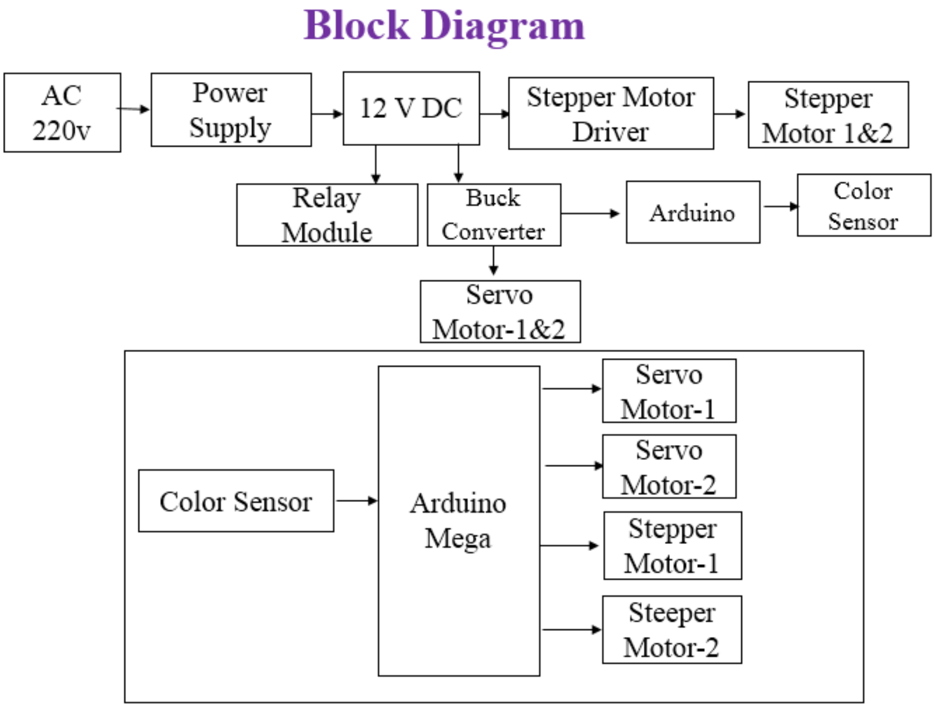
Fig3.1: Acrylic Board for X & Y axis movement and Unloading box

3.5 Development of the Electrical System

Electrical system is the heart of the sensing and distinguishing process. Three sensor circuits were made to control this automation system. The Arduino circuit, motor driver circuit & color sensor circuit. Arduino mega 2560 has been used as the microcontroller which controls the whole system. It has 54 input and output pin.

TCS3200 has been used as color sensor. It has 4 points S0, S1, S2 & S3. The output of color sensor connected to Arduino no 13 pin. S0 is connected to no 8 pin S1 is to 9, S2 is to 12& S3 is to 11 no pin. Stepper Motor 1 stepin X is connected to pin no 17, Stepper motor 2 stepin Y is connected to 15. Stepper Motor 1 dirpin X is connected to pin no 16 & Stepper motor 2 dirpin Y is connected to 14. When the color sensor detects the color of the product it sends signal to Arduino and then Arduino sends signal to stepper motor driver. If the product is pink then the stepper motor 1 will step forward 2000 step and stepper motor 2 will step forward 9500 and the unloading servo will give an inclined motion of 30-degree angle to unload the product. Then it will step back to its previous position& bottom servo will give a 90˚ circular rotation to unloading box for taking the product from outside. If the product is green then it will step forward to 6000 steps and same as before the inclination angle of servo will give for unloading. Then it will step back 6000 steps. And if the product is blue then it will move forward to 9000 steps to unload the product & same angle will get the servo for unloading. Then it will step back 9000 steps into its initial position. For unloading the product it will get 10 seconds. The whole system will get power by a SMPS power supply. It will convert the AC 220 v to DC 12 V. The Arduino will get power from normally a PC by USB cable or power supply. Bottom servo is connected to the pin of Arduino 6 and the top servo is connected to the pin of Arduino 7. Servo will be operated by getting power from buck converter which will convert the 12 v DC to 6 V DC to operate both servo motor. We have used a delay relay module to give a delay time at the very beginning of the project. We have also used a limit switch to control the movement of the stepper motor.

3.6 Block Diagram



Block diagram of product distinguishing system

]

3.7: Electrical Drawing

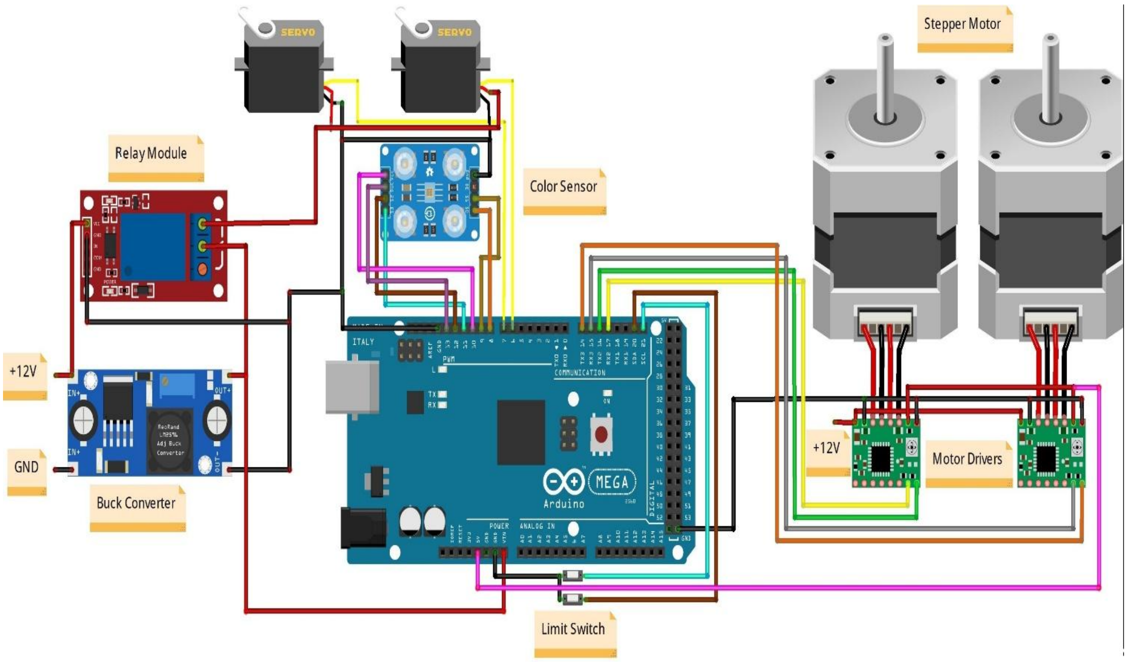


Fig 3.2: Electrical Drawing

3.8 Working Principle

For distinguishing the object in industry optical sorting is very much convenient. Color and size are the most important features for accurate classification and distinguishing of product which can be done by using some optical sensors or analyzing their pictures. This machine is mainly sense the different color of the object and assert them into their desired location. When an object keeps on the unloading box color sensor senses the color of the object, as the input devices will send signal to microcontroller where Arduino as the microcontroller will give command to the stepper motor to move its right place and then Arduino will give command to servo motor for unloading the product. The final result was quite satisfactory. The color detecting sensor works well and it is able to detect pink blue & green object quite nicely. The stepper motor, lead screw and smooth rod gives 'X' & 'Y' axis movement of unloading box and servo motor gives the unloading rotation & another servo motor gives 90˚ rotation for taking the products. The system performed well as programmed and detects the object according to their color.

CHAPTER IV

HARDWARE ANALYSIS

4.1 Arduino Mega 2560

Arduino Mega 2560 is a microcontroller board based on ATmega 2560 (Datasheet). Arduino Boards have revitalized the automation industry with their easy to use platform where everyone with little or no technical background can get started with learning some basic skills to program and run the board.

There are many other Arduino board like Arduino Uno, Arduino Nano, and Arduino Pro Mini. All these boards function similarly in one way or the other. There are some basic features like PCB layout design, size, number of analog pins and breadboard friendly nature that make them different from each other. In terms of coding, all these boards are programmed in Arduino IDE software and you don’t need to attach extra components or devices to put them in the running condition. Everything is already built in the board that makes this device readily available. Just plug and play with the board as per your requirement.

All the boards mentioned above work perfectly for a number of Arduino Projects when you require a simple task to be completed with less number of I/O pins and memory. However, when the nature of project goes complex, a board with less memory fails to complete the task. This is where Arduino Mega 2560 comes handy. This board comes with 54 pins and 16 analog pins with more memory to store the code. Sounds crazy, isn’t it? Thanks to technology that keep your covered in every aspect and provides support in any way when it comes to fulfilling your technical needs.

I’ll try to cover each and everything related to Arduino Mega 2560, what is this about, main features, working, technical specifications and everything you need to know. Let’s jump right in. [5].

- **Arduino Mega** 2560 is a Microcontroller board based on Atmega2560. It comes with more memory space and I/O pins as compared to other boards available in the market.
- There are 54 digital I/O pins and 16 analog pins incorporated on the board that make this device unique and stand out from others.
- Out of 54 digital I/O, 15 are used for PWM (pulse width modulation).
- A crystal oscillator of 16MHz frequency is added on the board.
- This board comes with USB cable port that is used to connect and transfer code from computer to the board.
- DC power jack is coupled with the board that is used to power the board. Some version of Arduino board lacks this feature like Arduino Pro Mini doesn't come with DC power jack.
- ICSP header is a remarkable addition to Arduino Mega which is used for programming the Arduino and uploading the code from the computer.

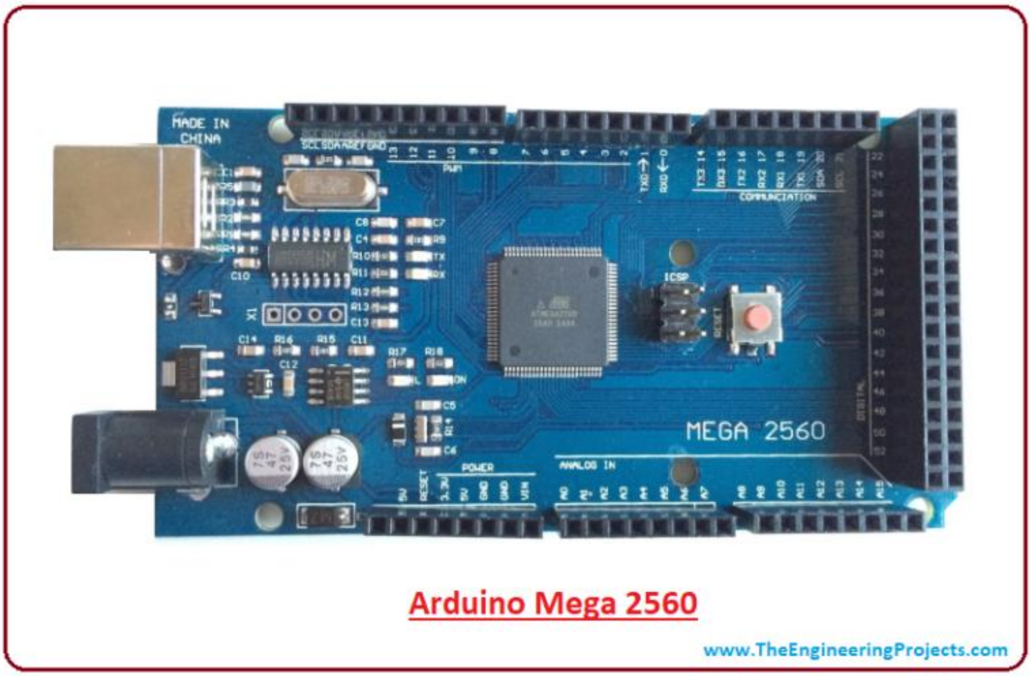
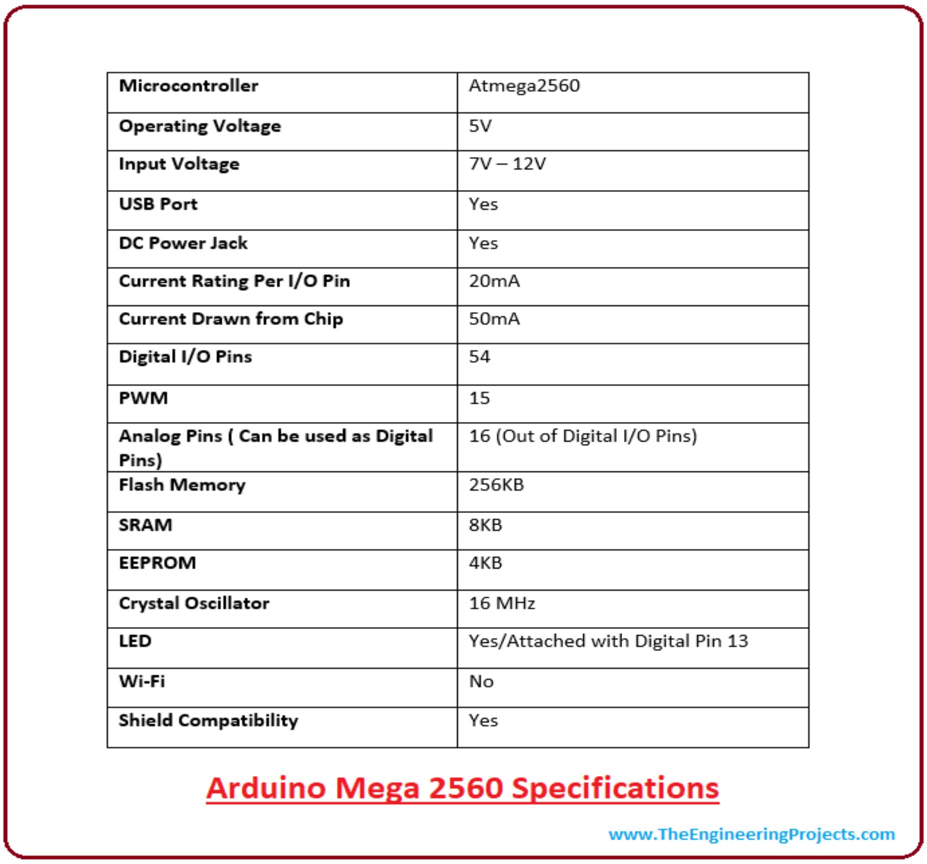


Fig 4.1: Arduino Mega 2560

- This board comes with two voltage regulator i.e. 5V and 3.3V which provides the flexibility to regulate the voltage as per requirements as compared to Arduino Pro Mini which comes with only one voltage regulator.
- There is no much difference between Arduino Uno and Arduino Mega except later comes with more memory space, bigger size and more I/O pins.
- Arduino software called Arduino IDE is used to program the board which is a common software used for all boards belonged to Arduino family.
- Availability of Atmega16 on the board makes it different than Arduino Pro Mini which uses USB to serial converter to program the board.
- There is a reset button and 4 hardware serial port called USART which produces a maximum speed for setting up communication. [5,7].



Microcontroller	Atmega2560
Operating Voltage	5V
Input Voltage	7V – 12V
USB Port	Yes
DC Power Jack	Yes
Current Rating Per I/O Pin	20mA
Current Drawn from Chip	50mA
Digital I/O Pins	54
PWM	15
Analog Pins (Can be used as Digital Pins)	16 (Out of Digital I/O Pins)
Flash Memory	256KB
SRAM	8KB
EEPROM	4KB
Crystal Oscillator	16 MHz
LED	Yes/Attached with Digital Pin 13
Wi-Fi	No
Shield Compatibility	Yes

Arduino Mega 2560 Specifications

www.TheEngineeringProjects.com

Fig 4.2: Arduino Mega 2560 Specification

- Arduino Mega is specially designed for the projects requiring complex circuitry and more memory space. Most of the electronic projects can be done pretty well by other boards available in the market which make Arduino Mega.
- However, there are some projects that are solely done by Arduino Mega like making of 3D printers or controlling more than one motors, because of its ability to store more instructions in the code memory and a number of I/O digital and analog pins.
- There are three ways to power the board. You can either use a USB cable to power the board and transfer code to the board or you can power it up using Vin of the board or through Power jack or batter.
- Last two sources to power the board are required once you already built and compile code into the board through USB cable.
- This board comes with resettable polyfused that prevents the USB port of your computer from overheating in the presence of high current flowing through the board. Most of the computers come with an ability to protect themselves from such devices, however, the addition of fuse provides an extra layer of protection.
- It can be used either way i.e. for creating stand-alone projects or in combination with other Arduino boards. Most complex projects can be created using this board. [5].

Arduino Mega 2560 Pinout

Following figure shows the pinout of Arduino Mega 2560.

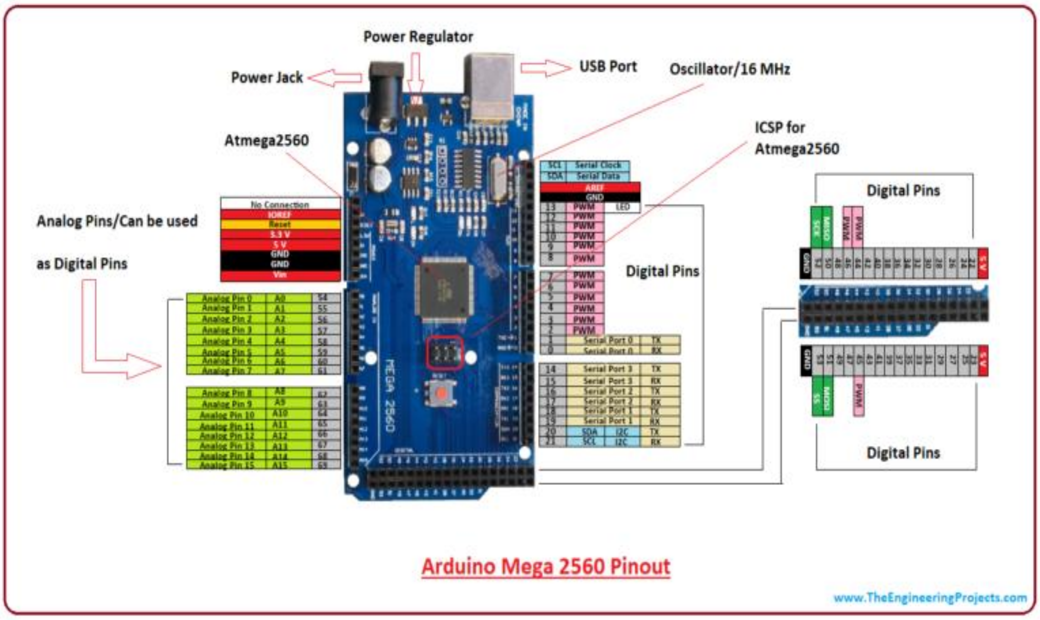


Fig 4.3: Arduino Mega 2560 Pinout

Pin Description

5V & 3.3V. This pin is used to provide output regulated voltage around 5V. This regulated power supply powers up the controller and other components on the board. It can be obtained from Vin of the board or USB cable or another regulated 5V voltage supply. While another voltage regulation is provided by 3.3V pin. Maximum power it can draw is 50mA.

GND. There are 5 ground pins available on the board which makes it useful when more than one ground pins are required for the project.

Reset. This pin is used to reset the board. Setting this pin to LOW will reset the board.

Vin. It is the input voltage supplied to the board which ranges from 7V to 20V. The voltage provided by the power jack can be accessed through this pin. However, the output voltage through this pin to the board will be automatically set up to 5V.

Serial Communication. RXD and TXD are the serial pins used to transmit and receive serial data i.e. Rx represents the transmission of data while Tx used to receive data. There are four combinations of these serial pins are used where Serail 0 contains RX(0) and TX(1), Serial 1 contains TX(18) and RX(19), Serial 2 contains TX(16) and RX(17), and Serial 3 contains TX(14) and RX(15).

External Interrupts. Six pins are used for creating external interrupts i.e interrupt 0(0), interrupt 1(3), interrupt 2(21), interrupt 3(20), interrupt 4(19), interrupt 5(18). These pins produce interrupts by a number of ways i.e. providing LOW value, rising or falling edge or changing value to the interrupt pins.

LED. This board comes with built-in LED connected to digital pin 13. HIGH value at this pin will turn the LED on and LOW value will turn it off. This gives you the change of nursing your programming skills in real time.

AREF. AREF stands for Analog Reference Voltage which is a reference voltage for analog inputs.

Analog Pins. There are 16 analog pins incorporated on the board labeled as A0 to A15. It is important to note that all these analog pins can be used as digital I/O pins. Each analog pin comes with 10-bit resolution. These pins can measure from ground to 5V. However, the upper value can be changed using AREF and analog Reference () function.

I2C. Two pins 20 and 21 support I2C communication where 20 represents SDA (Serial Data Line mainly used for holding the data) and 21 represents SCL(Serial Clock Line mainly used for providing data synchronization between the devices)

SPI Communication. SPI stands for Serial Peripheral Interface used for the transmission of data between the controller and other peripherals components. Four pins i.e. 50 (MISO), 51 (MOSI), 52 (SCK), 53 (SS) are used for SPI communication. [5,7].

4.2 Stepper Motor

The stepper motor is a professional and high precision motor with 1.8 deg. step angle (200 steps/revolution), 2 phase 4 wires, high holding torque, and each phase draws current 1A at 6V [6].

Specifications

- Size: 42.3 mm square × 48 mm, not including the shaft (NEMA 17)
- Weight: 350 g (13 oz)
- Shaft diameter: 5 mm “D”
- Steps per revolution: 200
- Current rating: 1.2 A per coil
- Voltage rating: 4 V
- Resistance: 3.3 Ω per coil
- Holding torque: 3.2 kg-cm (44 oz-in)
- Inductance: 2.8 mH per coil
- Lead length: 30 cm (12″)
- Output shaft supported by two ball bearings

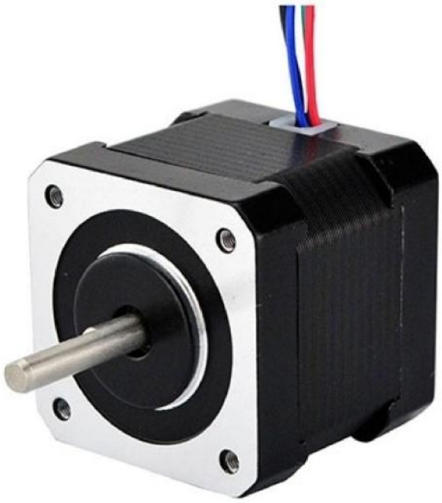


Fig4.4: Stepper Motor

4.3 Servo Motor S8503

Description:

The CYS S8503 is a modified design and high quality servo which is digital, fitted with metal gears for strength and reliability. And it is made of high quality material, which may make your car more remarkable. The specifications are very impressive, masses of torque, excellent high speed, perfect for 1/5 RC car.

Features:

- All aluminum alloy case torque
- Large stall torque of 28Kg
- High precision metal gear
- Light weight and easy to take
- Modified design and high quality
- Ideal choice for 1/51 Redact HPI Baja 5B SS RC car

Specifications:

- Brand: CYS
- Item name: S8503 Digital metal gear servo
- Material: Metal
- Signal: Digital
- Connector wire: About 310mm
- Operating voltage: 6.0-7.4V
- Operating Speed: 0.22 sec / 60°@ 6.0V; 0.20 sec / 60° @ 7.4V
- Stall torque: 26kg.cm @ 6.0V; 28kg.cm @ 7.4V
- Item size: 74 * 60 * 28mm (artificial measurements error allowed)
- Item weight: 154g (wires included

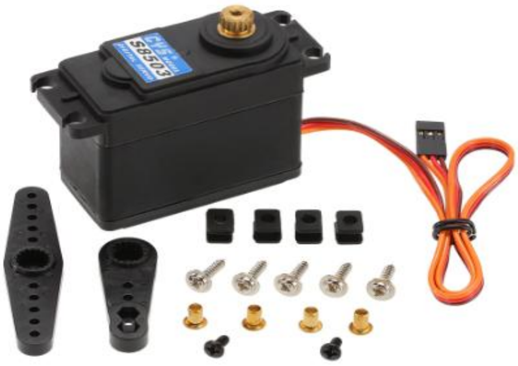


Fig 4.5: Servo Motor

4.4 SMPS Power Supply

A switched-mode power supply is an electronic power supply that incorporates a switching regulator to convert electrical power efficiently. Like other power supplies, an SMPS transfers power from a DC or AC source to DC loads, such as a personal computer, while converting voltage and current characteristics.

AC input 110/220 v and output DC 12 V 20A

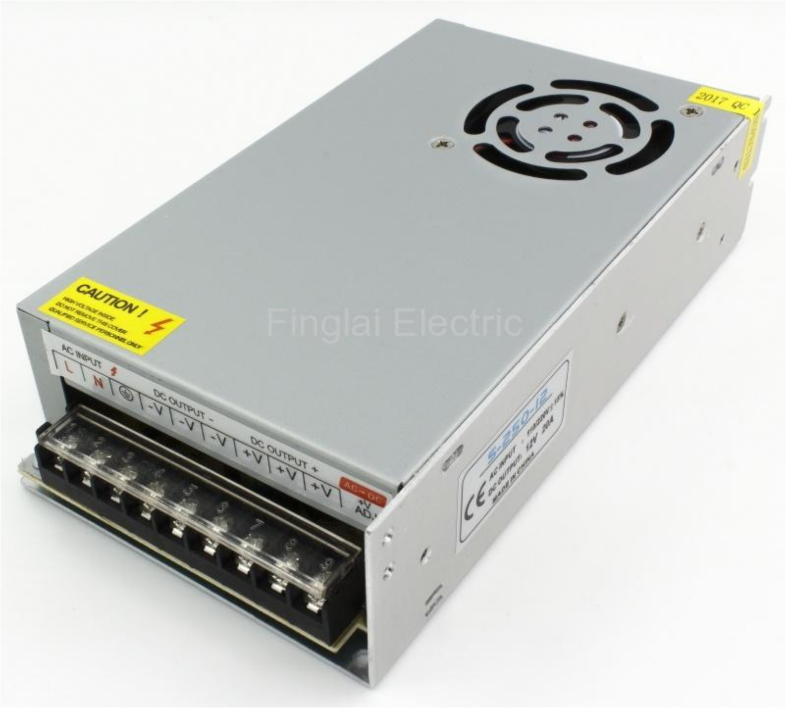


Fig 4.6: SMPS Power Supply

A switched-mode power supply (switching-mode power supply, switch-mode power supply, switched power supply, SMPS, or switcher) is an electronic power supply that incorporates a switching regulator to convert electrical power efficiently.

Like other power supplies, an SMPS transfers power from a DC or AC source (often mains power) to DC loads, such as personal computer, while converting voltage and current characteristics. Unlike a linear power supply, the pass transistor of a switching-mode supply continually switches between low-dissipation, full-on and full-off states, and spends very little time in the high dissipation transitions, which minimizes wasted energy. Ideally, a switched-mode power supply dissipates no power. Voltage regulation is achieved by varying the ratio of on-to-off time. In contrast, a linear power supply regulates the output voltage by continually dissipating power in the pass transistor. This higher power conversion efficiency is an important advantage of a switched-mode power supply. Switched-mode power supplies may also be substantially smaller and lighter than a linear supply due to the smaller transformer size and weight.

Switching regulators are used as replacements for linear regulators when higher efficiency, smaller size or lighter weight are required. They are, however, more complicated; their switching currents can cause electrical noise problems if not carefully suppressed, and simple designs may have a poor power factor.

Specification:

- Input voltage range: AC 110~220V
- Over load protection: 105~200% (rated load), auto recovery
- Start inrush current: 20A @AC 120V (cold start)
- Output adjusting range: +/-10% (rated output voltage)
- Over-voltage protection: 115~135% (rated output voltage)
- Short protection: auto recovery
- Output start time: <=2s (AC 120V), 0s (AC 230V)
- Output keeping time: >=10ms (AC 120V), >=20ms (AC 230V)
- Insulation: A 1500V, 1min- Input: AC 1500V, 1min, Output: DC 500V, 1min

4.5 TCS3200 Color Sensor

The TCS3200 color sensor – shown in the figure below – uses a TAOS TCS3200 RGB sensor chip to detect color. It also contains four white LEDs that light up the object in front of it. This chip can sense a wide variety of colors and it gives the output in the form of corresponding frequency. This module can be used for making color sorting robots, test strip reading, color matching tests etc. [12].

Power: 2.7V to 5.5V

Size: 28.4 x 28.4mm (1.12 x 1.12″)

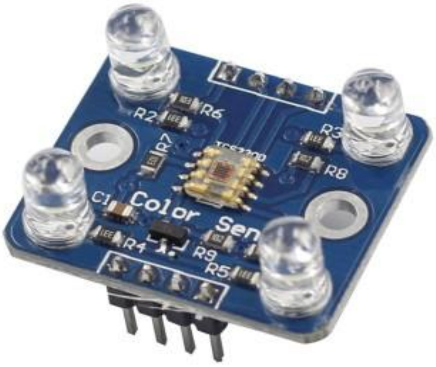


Fig 4.7: TCS 3200 Color Sensor

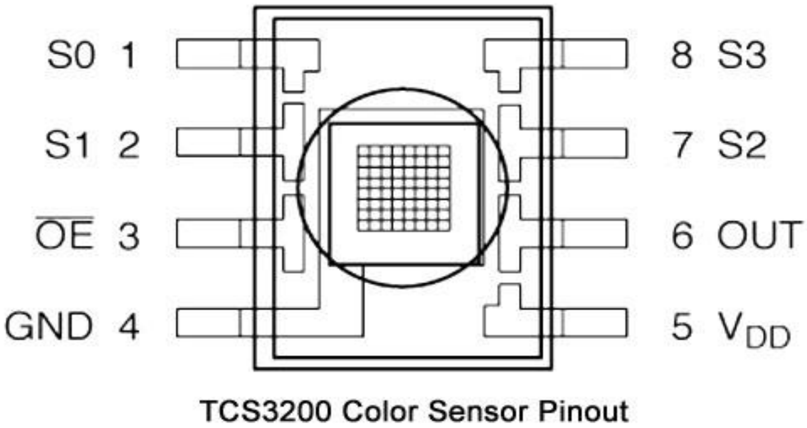


Fig 4.8: TCS 3200 Color Sensor Pin out

4.6 A4988 Stepper Motor Driver

This breakout board for Allegro’s A4988 micro stepping bipolar stepper motor driver features adjustable current limiting, over-current and over-temperature protection, and five different micro step resolutions (down to 1/16-step). It operates from 8 V to 35 V and can deliver up to approximately 1 A per phase without a heat sink or forced air flow (it is rated for 2 A per coil with sufficient additional cooling). This board ships with 0.1″ male header pins included but not soldered in.

This product is a carrier board or breakout board for Allegro’s A4988 DMOS Micro Stepping Driver with Translator and Over current Protection; we therefore recommend careful reading of the A4988 datasheet before using this product. This stepper motor driver lets you control one bipolar stepper motor at up to 2 A output current per coil

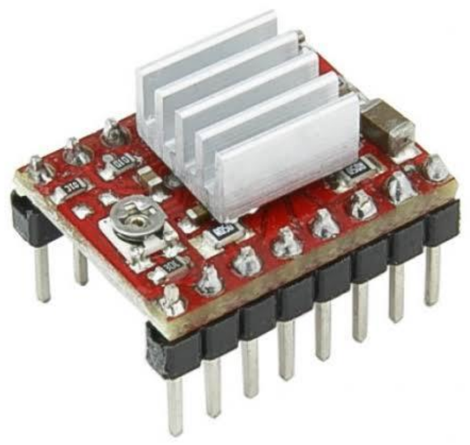


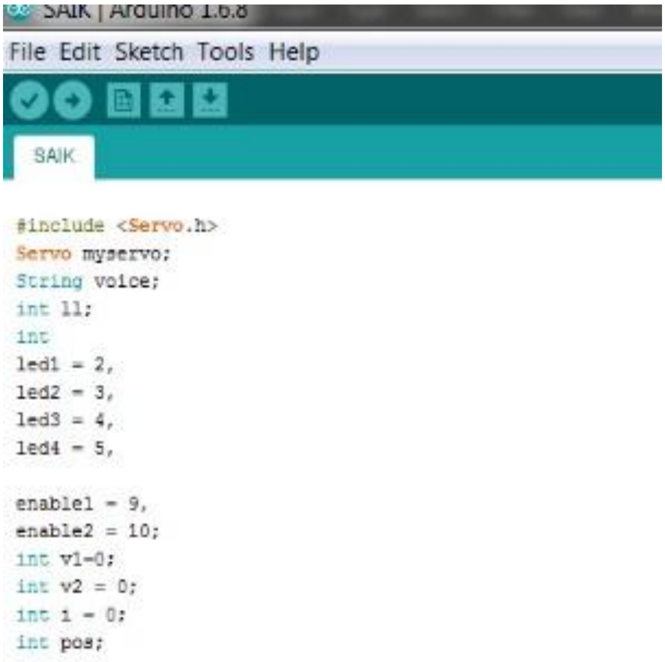
Fig 4.9: A4988 Stepper Motor Driver

CHAPTER V

SOFTWARE ANALYSIS

5.1Arduino software:

The smart microcontroller unit named as Arduino Uno can be programmed with the Arduino software. There is no any requirement for installing other software rather than Arduino. Firstly, Select "Arduino Uno from the Tools , Board menu (according to the microcontroller on your board).The IC used named as ATmega328 on the Arduino Uno comes pre burned with a boot loader that allows you to upload new code to it without the use of an external hardware programmer.



```
#include <Servo.h>
Servo myservo;
String voice;
int l1;
int
led1 = 2,
led2 - 3,
led3 = 4,
led4 - 5,

enable1 - 9,
enable2 = 10;
int v1-0;
int v2 = 0;
int i - 0;
int pos;
```

Figure 5.1: Arduino Software Interface IDE.

Communication is using the original STK500 protocol (reference, C header files). We can also bypass the boot loader and programs the microcontroller through the ICSP (In Circuit Serial Programming) header. The ATmega16U2 (or 8U2 in the rev1 and rev2 boards) firmware source code is available. [7]. The ATmega16U2/8U2 is loaded with a DFU boot loader, which can be activated by:

On Rev1 boards: connecting the solder jumper on the back of the board (near the map of Italy) and then resetting the 8U2.On Rev2 or later boards: there is a resistor that pulling the 8U2/16U2 HWB line to ground, making it easier to put into DFU mode.

The Arduino Uno is one of the latest smart microcontroller units and has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL at (5V) with serial communication, which is available on digital pins 0 -(RX) for receive the data and pin no.1 (TX) for transmit the data. An ATmega16U2 on the board channels this serial communication over USB and appears as a virtual com port to software on the computer. The '16U2 firmware uses the standard USB COM drivers, and no external driver is needed. However, on Windows, an .inf file is required. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. [7].

The RX and TX LEDs on the board will flash when data is being transmitted via the USB-to-serial chip and USB connection to the computer (but not for serialCommunication on pins 0 and 1). A Software Serial library allows for serial communication on any of the Uno's digital pins. The ATmega328 also supports I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus. Arduino programs are written in C or C++ and the program code written for Arduino is called sketch. The Arduino IDE uses the GNU tool chain and AVR Linc to compile programs, and for uploading the programs it uses avrdude. As the Arduino platform uses Atmel microcontrollers, Atmel's development environment, AVR Studio or the newer Atmel Studio, may also be used to develop software for the Arduino.

5.2 Proteus:

The Proteus Design Suite is a proprietary software tool suite used primarily for electronic design automation. The software is used mainly by electronic design engineers and technicians to create schematics and electronic prints for manufacturing printed circuit boards.

The first version of what is now the Proteus Design Suite was called PC-B and was written by the company chairman, John Jameson, for DOS in 1988. Schematic Capture support followed in 1990, with a port to the Windows environment shortly thereafter. Mixed mode SPICE Simulation was first integrated into Proteus in 1996 and microcontroller simulation then arrived in Proteus in 1998. Shape based auto routing was added in 2002 and 2006 saw another major product update with 3D Board Visualization. More recently, a dedicated IDE for simulation was added in 2011 and MCAD import/export was included in 2015. Support for high speed design was added in 2017.Feature led product releases are typically biannual, while maintenance based service packs are released as required.

5.3 Microcontroller Simulation:

The micro-controller simulation in Proteus works by applying either a hex file or a debug file to the microcontroller part on the schematic. It is then co-simulated along with any analog and digital electronics connected to it. This enables its use in a broad spectrum of project prototyping in areas such as motor control, temperature control and user interface design. It also finds use in the general hobbyist community and, since no hardware is required, is convenient to use as a training or teaching tool. Support is available for co-simulation of:

- Microchip Technologies PIC10, PIC12, PIC16, PIC18, PIC24, dsPIC33 Microcontrollers.

- Atmel AVR (and Arduino), 8051 and ARM Cortex-M3 Microcontrollers
- NXP 8051, ARM7, ARM Cortex-M0 and ARM Cortex-M3 Microcontrollers.
- Texas Instruments MSP430, PICCOLO DSP and ARM Cortex-M3 Microcontrollers.
- Parallax Basic Stamp, Free scale HC11, 8086 Microcontrollers. [5,7].

5.4 PCB Design:

The PCB Layout module is automatically given connectivity information in the form of a net list from the schematic capture module. It applies this information, together with the user specified design rules and various design automation tools, to assist with error free board design. PCB's of up to 16 copper layers can be produced with design size limited by product configuration.

5.5 3D Verification:

The 3D Viewer module allows the board under development to be viewed in 3D together with a semi-transparent height plane that represents the board’s enclosure. STEP output can then be used to transfer to mechanical CAD software such as Solid works or Autodesk for accurate mounting and positioning of the board. [7].

5.6 Results:

We have developed an automatic distinguishing machine using color sensor for automatic product distinguishing, taking in to consideration three colors namely green, pink and blue. We have consumed twelve months to produce the prototype with the expense of TK.20000.

A simple comparison between a manual unloading & automatic unloading:

Comparison Points	Manual Unloading	Automatic Unloading
Power	Excessive man power	240 W electric power
Cost	50 tk per hour	3 tk per hour
Time	Average 10 mints	Average 1.5 mints
Continuous Process	Is not possible to continue	Possible to continue

CHAPTER VI

DISCUSSION & CONCLUSION

6.1 Advantages:

- It will save time by its automatic process.
- Very Cost Effective.
- Very fast distinguishing can be done.
- Physical power has not waste.
- More accurate than human working process.

6.2 Applications:

Main applications can be for:

- Distinguishing the product according to their color.
- Ensuring quality control in mass production.
- By some modification it can be used to measure the dimension of a product.
- By increasing its sensing capacity, it can be used in airport.
- It is also very useful in laboratories and workshops.
- It can be widely used in any productive industry

6.3 Future Scope

The model can be improved by making some changes in the program and components. Some suggestions are given below.

- We can add a load cell for measurement and control of weight of the product.
- We can also add a counter for counting the number of products.
- We can add automatic gripper to load the product into unloading box.
- Speed of the system can be increased accounting to the speed of production.
- The system can be used as a quality controller by adding more sensors.
- The sensor can be changed according to the type of product.
- The Arduino can be replaced with PLC.

6.4 Conclusion

Now-a-days in highly competitive industrial manufacturing, the management of the integrity of supply of a product from raw material to finished product through quality manufacturing is of paramount importance. Bickman, et al [13] described in the article about automated color-sorting using optical technology that has evolved from early designs intended to remove ceramic contaminants.

For the declaration of a product bearing high-Quality and dimensional accuracy is mandatory. So, this project of automatic color sorting is an excellent one because of its working principle and wide implementation [15]. By applying the idea of this project an industry can easily distinguish the required product according to its color. Though it has some limitations, but by having done some modification this concept can be implemented in wide range of application.

References:

[1]A.K. Gupta, S.K. Arora , Jean Riescher Westcott “Industrial Automation and Robotics: An Introduction ”

[2] [Forces of Production: A Social History of Industrial Automation] By Noble, David F (Author) [2011) [Hardcover] Hardcover – March 14, 2011

[3] M. Frank, Nobert Kaiser, Wolfgang Buss, Ramona Eberhardt, “High-speed industrial color and position sensors”, Electronic
Imaging’99, pp 50-57, 1999.

[4] A. Pacquit, J. Frisby, Danny Diamond, King Tong Lau, Alan Farrell, Brid Quilty, Dermot Diamond, “Development of a smartpackaging for the monitoring of fish spoilage”, Food Chem, vol 102, pp 466-470, 2007.

[5] J.M Hughes “Arduino A Technical Reference”

[6] D.J. Lee and R. S. Anbalagan, “High-speed automated color sorting vision system”, in Optical Engineering Midwest’95, pp
573-579, 1995.

[7] John wiley & Sons“Exploring Arduino: Tools and Techniques for Engineering Wizardry”

[8] J. V. Popov-Ralji and J. G. Lalii-Petronijevi, “Sensory properties and color measurements of dietary chocolates with different
compositions during storage for up to 360 days”, Sensors, vol 9, pp 1996-2016, 2009.

[9] R. Bogue, “Optical chemical sensors for industrial applications”, Sensor Review, vol 27, pp 86-90, 2007.

[10] R. Baribeau, “Color reflectance modeling using a polychromatic laser range sensor”, IEEE T Pattern. Anal., vol. 14, pp. 263-
269, 1992.

[11] H. Escid, et al., “0.35 mm CMOS optical sensor for an integrated transimpedance circuit”, the International Journal on Smart
Sensing and Intelligent Systems, vol. 4, no. 3, pp. 467481, September 2011.

[12] Norfazlinda Binti Daud,“Application of colors sensor in an automated system”, Technical University Malaysia, May 2007.

[13] Bickman, Josh, “Automated Color-Sorting uses optical technology”, vol. 13, 1996.

[14] Bozma and Yal-cin, “Visual processing and classification of items on a moving conveyor: a selective perception approach”,
vol. 18, issue 2, 2002.

[15] C Boukouvalas, J Kittler, R Marik, M Mirmehdi and Petrou, “Ceramic tile inspection for color and structural deffects”,
University of Surrey, 1995.

APPENDICS

```
#include <Servo.h>
//#include <LiquidCrystal.h>
//LiquidCrystal lcd(7, 6, 5, 4, 3, 2);//RS,EN,D4,D5,D6,D7

// SENSOR
const byte S0 = 8;
const byte S1 = 9;
const byte S2 = 12;
const byte S3 = 11;
const byte sensorOut = 13;

//MOTOR
const byte stepPinX = 17;
const byte dirPinX = 16;

const byte stepPinY = 15;
const byte dirPinY = 14;

//Limit Switch
const byte SwitchTopX = 20;
const byte SwitchBottomX = 21;
const byte SwitchTopY = 18;
const byte SwitchBottomY = 19;

// LED pins connected to Arduino
const byte redLed = 22;
const byte greenLed = 23;
const byte blueLed = 24;

Servo servoTop;  // create servo object to control a servo
Servo servoBottom;  // create servo object to control a servo

// Variables
// Stores frequency read by the photodiodes
int redFrequency = 0;
int greenFrequency = 0;
int blueFrequency = 0;

int posTop = 0;    // variable to store the servo position
int posBottom = 5;    // variable to store the servo position
int TopX = 0;
int BottomX = 0;
int TopY = 0;
int BottomY = 0;
int Speed = 700;

void setup()
{
  pinMode(6, INPUT_PULLUP);
  pinMode(7, INPUT_PULLUP);
  digitalWrite(6,LOW);
  digitalWrite(7,LOW);
  Serial.begin(9600);
  //lcd.begin(16, 2);
  pinMode(S0, OUTPUT);
  pinMode(S1, OUTPUT);
  pinMode(S2, OUTPUT);
  pinMode(S3, OUTPUT);
  pinMode(sensorOut, INPUT);

  // Setting frequency scaling to 20%
  digitalWrite(S0,HIGH);
  digitalWrite(S1,LOW);

  // Power on Color sensor
  pinMode(10, OUTPUT);
  digitalWrite(10,HIGH);

  pinMode(redLed, OUTPUT);
  pinMode(greenLed, OUTPUT);
  pinMode(blueLed, OUTPUT);
  pinMode(stepPinX,OUTPUT);
  pinMode(dirPinX,OUTPUT);
  pinMode(stepPinY,OUTPUT);
  pinMode(dirPinY,OUTPUT);
  pinMode(SwitchTopX, INPUT_PULLUP);
  pinMode(SwitchBottomX, INPUT_PULLUP);
  pinMode(SwitchTopY, INPUT_PULLUP);
  pinMode(SwitchBottomY, INPUT_PULLUP);

  attachInterrupt(digitalPinToInterrupt(SwitchTopX), limittopx,
RISING);
  attachInterrupt(digitalPinToInterrupt(SwitchBottomX), limitbottomx,
RISING);
  attachInterrupt(digitalPinToInterrupt(SwitchTopY), limittopy,
RISING);
  attachInterrupt(digitalPinToInterrupt(SwitchBottomY), limitbottomy,
RISING);
  TopX = 0;
  BottomX = 0;
  TopY = 0;
  BottomY = 0;
  delay(2000);
  //SERVO
  servoTop.write(25);
  servoBottom.write(5);
  servoTop.attach(7);
  servoBottom.attach(6);
  servoTop.write(25);
  servoBottom.write(5);
  delay(2000);
}

void loop()
{

  /*Serial.print("TopX = ");
  Serial.print(TopX);
  Serial.print("BottomX = ");
  Serial.print(BottomX);
  Serial.print("TopY = ");
  Serial.print(TopY);
  Serial.print("BottomY = ");
  Serial.println(BottomY);
```

```
delay(1000);*/
// Setting RED (R) filtered photodiodes to be read
digitalWrite(S2,LOW);
digitalWrite(S3,LOW);

// Reading the output frequency
redFrequency = pulseIn(sensorOut, LOW);
Serial.print("R = ");
Serial.println(redFrequency);
if(redFrequency < 140){
  delay(3000);
  digitalWrite(S2,LOW);
  digitalWrite(S3,LOW);

  // Reading the output frequency
  redFrequency = pulseIn(sensorOut, LOW);
   // Printing the RED (R) value
  Serial.print("R = ");
  Serial.print(redFrequency);
  delay(100);

  // Setting GREEN (G) filtered photodiodes to be read
  digitalWrite(S2,HIGH);
  digitalWrite(S3,HIGH);
  // Reading the output frequency
  greenFrequency = pulseIn(sensorOut, LOW);

  // Printing the GREEN (G) value
  Serial.print(" G = ");
  Serial.print(greenFrequency);
  delay(100);

  // Setting BLUE (B) filtered photodiodes to be read
  digitalWrite(S2,LOW);
  digitalWrite(S3,HIGH);

  // Reading the output frequency
  blueFrequency = pulseIn(sensorOut, LOW);

  // Printing the BLUE (B) value
  Serial.print(" B = ");
  Serial.println(blueFrequency);
  delay(100);
  int lt[3] = {redFrequency, greenFrequency, blueFrequency};
  int minIndex = 0;
  int min = lt[minIndex];
  for (int i=1; i<3; i++){
    if (min>lt[i]){
      minIndex = i;
      min = lt[i];
    }
  }

  Serial.println();
  Serial.print(min);
  Serial.print( "=");
  Serial.println(minIndex);
  int cycleX = 0;
```

```
    if(minIndex == 0){
      cycleX = 2000;
    }
    else if(minIndex == 1){
      cycleX = 6000;
    }
    else if(minIndex == 2){
      cycleX = 10000;
    }
    for (posBottom = 5; posBottom <= 90; posBottom += 1) { // goes from
5 degrees to 90 degrees
      servoBottom.write(posBottom);              // tell servo to go to
position in variable 'pos'
      delay(30);                       // waits 15ms for the servo to
reach the position
    }
    digitalWrite(dirPinY,HIGH);
    for(int y = 0; y < 9500; y++) {
        digitalWrite(stepPinY,HIGH);
        delayMicroseconds(Speed);
        digitalWrite(stepPinY,LOW);
        delayMicroseconds(Speed);
    }
    digitalWrite(dirPinX,LOW); //Changes the rotations direction
//   Makes 400 pulses for making two full cycle rotation
    for(int x = 0; x < cycleX; x++) {
      if(TopX == 0){
        digitalWrite(stepPinX,HIGH);
        delayMicroseconds(Speed);
        digitalWrite(stepPinX,LOW);
        delayMicroseconds(Speed);
        BottomX = 0;
      }
      else{
        break;
      }
    }
    for (posTop = 25; posTop >= 0; posTop -= 1) { // goes from 180
degrees to 0 degrees
      servoTop.write(posTop);              // tell servo to go to
position in variable 'pos'
      delay(10);                       // waits 15ms for the servo to
reach the position
    }
    delay(3000);
    for (posTop = 0; posTop <= 25; posTop += 1) { // goes from 0
degrees to 180 degrees
      servoTop.write(posTop);              // tell servo to go to
position in variable 'pos'
      delay(30);                       // waits 15ms for the servo to
reach the position
    }

    digitalWrite(dirPinX,HIGH);
    for(int x = 0; x < 90000; x++) {
      if(BottomX == 0){
        digitalWrite(stepPinX,HIGH);
        delayMicroseconds(Speed);
        digitalWrite(stepPinX,LOW);
        delayMicroseconds(Speed);
        TopX = 0;
      }
      else{
        break;
      }
    }
    digitalWrite(dirPinY,LOW); //Changes the rotations direction
    for(int y = 0; y < 9500; y++) {
        digitalWrite(stepPinY,HIGH);
        delayMicroseconds(Speed);
        digitalWrite(stepPinY,LOW);
        delayMicroseconds(Speed);
    }
    for (posBottom = 90; posBottom >= 5; posBottom -= 1) { // goes from
180 degrees to 0 degrees
      servoBottom.write(posBottom);              // tell servo to go to
position in variable 'pos'
      delay(30);                       // waits 15ms for the servo to
reach the position
    }
    delay(1000);

  }
  delay(1000);
 }

void limittopx(){
  TopX = 1;
}

void limitbottomx(){
  BottomX = 1;
}

void limittopy(){
  TopY = 1;
}

void limitbottomy(){
  BottomY = 1;
}
```