An-Najah National University

Faculty of Engineering and Information Technology

Computer Engineering Department

PackSort

By:

Baker Razi Wael Yaeesh

Mohammad Maher Mohammad Himuni

Project supervisor: Dr. Abdallah Rashed

Presented in partial fulfillment of the requirements for Bachelor’s Degree in Computer Engineering

Acknowledgement

We would like to sincerely thank Dr. Abdallah Rashed, our supervisor, for his invaluable guidance and unwavering support throughout our project. We are also deeply grateful to our friends and families for their constant encouragement and assistance from the very beginning.

Table of Contents

Table of Figures

Disclaimer

This Graduation Project Report has been prepared as part of the requirements for completing the Bachelor's degree in Computer Engineering at An-Najah National University.
The content of this report, including all ideas, designs, and implementations, represents the work of the students and is intended solely for academic purposes.
While every effort has been made to ensure accuracy and reliability, the authors and the university bear no legal responsibility for any misuse, errors, or consequences resulting from applying the concepts, designs, or systems described herein.
All technologies, software, and hardware mentioned in this report are the property of their respective owners, and their use is solely for educational purposes.

Abstract

This project presents the design and implementation of an intelligent automated sorting system capable of categorizing packages based on multiple parameters including barcode data, color, and physical dimensions. The system integrates an Arduino Mega as the central processing unit with a specialized Arduino Uno for barcode processing, creating a robust distributed control architecture.

The sorting mechanism employs a conveyor belt powered by a DC wiper motor controlled through a BTS7960 driver module, with package diversion accomplished using servo-actuated pusher arms. Detection subsystems include an omnidirectional barcode scanner for package identification, TCS3200 color sensors for color recognition, ultrasonic sensors for dimensional analysis, and IR sensors for position tracking.

A significant feature of this system is its flexible multi-mode operation, allowing sorting based on seven different criteria combinations: size alone, color alone, barcode alone, color and size, barcode and color, barcode and size, or all three parameters simultaneously. The system incorporates multiple fail-safes including an emergency stop button and comprehensive error handling protocols.

For power efficiency, the system also includes an automatic shut-off feature: if no packages are detected on the conveyor for 2 minutes, the system powers down automatically until new activity is detected.

Human-machine interface is provided through a keypad and LCD display for local control, while an ESP32 microcontroller enables wireless connectivity for remote monitoring and operation via a custom-developed Flutter mobile application. This mobile interface provides real-time package statistics and system status updates, enhancing operational visibility.

This automated sorting solution demonstrates significant improvements in efficiency and accuracy over manual sorting methods, with potential applications in logistics centers, postal services, and industrial packaging facilities. The modular design allows for adaptation to various sorting criteria and operational requirements, making it a versatile solution for modern package handling challenges.

Introduction

General Background

The massive growth in e-commerce and logistics has created a critical need for efficient, automated package sorting. Manual methods are slow and error-prone. This project addresses this need by developing an integrated system that uses modern sensors and microcontrollers to automate the sorting process based on multiple criteria.

Objectives

This project aims to design and build a versatile automated sorting machine. Its key goals are to:

1. Identify packages using barcode, color, and size sensors.
2. Physically sort them using a conveyor belt and servo-controlled pushers.
3. Offer seven different sorting modes (e.g., by color, by size, or a combination).
4. Allow control via both a local keypad/LCD and a remote mobile app.
5. Ensure safety with an emergency stop feature and system diagnostics.

Significance of The Work

This work is significant because it creates a flexible and affordable automation solution for small-to-medium warehouses. It demonstrates how to effectively integrate multiple technologies (sensing, mechanics, and IoT connectivity) into a single, functional system. It also serves as a valuable educational model for the principles of mechatronics and smart logistics.

Organization of The Report

This report is structured into six comprehensive chapters that document the project's development in a logical progression.

Introduction, provides the general background and context for the research topic, outlining the core problem of manual package sorting and presenting the objectives and significance of the automated solution developed in this project.

Constraints and Earlier Coursework, details the specific challenges encountered during the project's lifecycle, including budget limitations, hardware integration issues, and software complexities. It also highlights the pivotal role of prior academic courses in electronics, programming, and mechanics, which provided the foundational knowledge necessary to devise effective solutions and strategies.

Literature Review, conducts an in-depth examination of the current research landscape in automated sorting systems. This chapter investigates similar projects and commercial solutions, analyzing their methodologies, technological stacks, and findings to establish a thorough understanding of the field and identify the unique contribution of this work.

Methodology, presents the systematic, step-by-step plan devised to build the sorting system. It encompasses the detailed process of constructing the robot, including the selection and integration of specific software tools (Arduino IDE, Flutter) and hardware components (sensors, motors, microcontrollers), serving as a comprehensive replicable guide for the project.

Results and Discussion, summarizes the data gathered from system testing and performance trials. The findings from different sorting modes are subjected to analysis, comparing and contrasting their efficiency and accuracy to facilitate a deeper understanding of the project's outcomes and encourage meaningful interpretation of the results.

Conclusion and Recommendation, encapsulates the final project summary and the valuable lessons learned throughout the development journey. It offers concrete recommendations for further improvement, suggesting potential enhancements to features, scalability, and subsystems to guide future work.

In essence, the report structure follows a logical progression, starting with an introduction and background, addressing constraints and earlier coursework, delving into a literature review, outlining the methodology, presenting results and facilitating discussions, and culminating in a conclusive summary and recommendations for future work. The foundational knowledge gained from earlier coursework was instrumental in applying theory to practical applications and successfully executing this project.

Literature Review

Introduction

Automated sorting systems increase efficiency and reduce errors in manufacturing and logistics. Multi-criteria sorting, based on size, color, barcode, or destination, improves accuracy and flexibility.

Automated Sorting Systems

Modern systems use sensors, actuators, and conveyors. Traditional single-criterion systems are being replaced by multi-criteria approaches for better sorting performance.

Sensor Technologies

1. **IR Sensors:** Detect object presence quickly for real-time sorting.
2. **Ultrasonic Sensors:** Measure object dimensions without contact.
3. **Color Sensors:** Classify items based on color for accurate sorting.
4. **Barcode Scanners:** Provide product identification to guide sorting decisions.

Actuators

1. **DC Motors & Drivers:** Control conveyor movement and speed.
2. **Servo Motors:** Operate gates or pushers with precise angles.
3. **DVD-RW Drives:** Cost-effective linear pushers for small items.

Microcontroller Integration

Arduino boards coordinate sensors and actuators. The system manages sorting logic and can communicate with IoT-enabled devices.

Mobile Integration & IoT

ESP32 allows remote monitoring and mobile app control. Users can select sorting modes and track packages in real time.

Multi-Criteria Sorting

Combining multiple sensors reduces errors and increases throughput. The system can handle size-only, color-only, barcode-only, or combined sorting modes.

Safety Features

Emergency stops and LCD feedback ensure operational safety and prevent damage to packages or equipment.

Conclusion

Integrating sensors, actuators, Arduino control, and IoT makes a flexible, accurate, and cost-effective multi-criteria sorting system suitable for small to medium-scale operations.

Methodology

Overview

The methodology of this project focuses on designing and implementing a multi-criteria automated sorting system. The system combines sensors, actuators, and microcontroller-based control to detect, classify, and sort objects based on size, color, and barcode. It integrates real-time data processing, decision-making logic, and mechanical actuation to ensure accurate and efficient sorting. The methodology is structured to include system design, hardware selection, sensor calibration, software programming, and optional mobile app integration, ensuring the system is flexible, scalable, and easy to operate.

Mechanical Components

Wood Sticks (on the Servo Motors)

Wood sticks are used as actuating arms attached to servo motors to push or redirect objects into the correct sorting lanes. Their lightweight and rigid nature make them suitable for fast, repetitive movements without adding significant load on the motors. The length and placement of the sticks are carefully chosen to ensure precise interaction with objects of different sizes.

Figure 1 Wood Sticks

Conveyor Belt

The conveyor belt serves as the primary transportation mechanism for moving objects through the sorting system. It provides a continuous, controlled movement that allows sensors to detect object characteristics (size, color, barcode) before they are sorted. The belt material and speed are selected to ensure smooth transport and prevent objects from slipping or colliding.

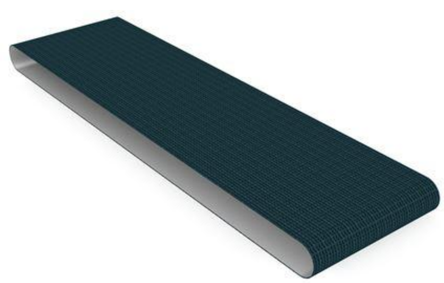


Figure 2 Conveyor Belt

Bearings

Bearings are used in rotating parts such as rollers of the conveyor belt or any pivoting axes. They reduce friction between moving components, enabling smooth rotation and increasing the efficiency and lifespan of the mechanical system. Properly positioned bearings ensure stability, prevent wear, and allow precise control of moving parts.

Figure 3 Bearings

Hardware Components

Arduino Mega

The Arduino Mega acts as the main microcontroller of the project, coordinating all the sensors, motors, and communication between devices. It processes input from barcode scanners, color sensors, IR sensors, and other peripherals, then sends commands to actuators like motors and servos for sorting packages. Its numerous input/output pins make it suitable for handling multiple devices simultaneously.

Figure 4 Arduino Mega

Arduino Uno

The Arduino Uno is dedicated to handling the barcode scanning process. It receives barcode data from the USB Host Shield and transmits it via serial communication to the Arduino Mega. This separation of tasks allows the Mega to focus on the overall control of the sorting system.

Figure 5 Arduino Uno

Omnidirectional Barcode Scanner

The omnidirectional barcode scanner simplifies barcode reading by allowing the scanner to read codes from any orientation. This ensures accurate and fast identification of packages, reducing the need for precise alignment.

Figure 6 Omnidirectional Barcode Scanner

BTS7960 Driver

The BTS7960 motor driver controls the DC wiper motor that drives the conveyor belt. It enables precise speed and direction control, allowing the system to transport packages smoothly and reliably.

Figure 7 BTS7960 Driver

Car Wiper DC Motor

The DC wiper motor is used to move the conveyor belt. Its torque and durability make it suitable for carrying packages of varying sizes and weights along the conveyor path.

Figure 8 Car Wiper DC Motor

DVDRW Drive (Old 2010)

An old DVDRW drive is repurposed as a mechanical pusher to physically push packages onto the conveyor or into their designated boxes. Its precise motion ensures consistent package handling.

Figure 9 DVDRW Drive (Old 2010)

Push Button

The push button functions as an emergency stop, immediately halting all operations when pressed to ensure safety and prevent damage to packages or the system.

Figure 10 Push Button

ESP32

The ESP32 allows wireless control and monitoring of the system through a mobile app built in Flutter. Users can select operational modes, deactivate them, and view real-time reports of package counts for each category.

Figure 11 ESP32

IR Sensors

Infrared sensors detect the presence and position of packages. They determine when a package has reached a particular point on the conveyor or the position of the mechanical stick or microservo motor for sorting.

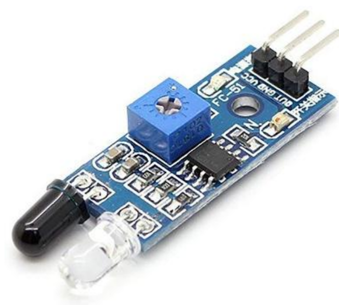


Figure 12 IR Sensors

Keypad

The keypad is used to enter passwords for system security and to select operational modes via the LCD interface.

Figure 13 Keypad

LCD 16×2 I2C

The LCD display provides real-time information to the user, including password prompts, mode selection, operational messages, and emergency alerts.

Figure 14 LCD 16×2 I2C

DC Power Supply(ATX-P4)

The DC power supply provides the necessary voltage and current to all components, ensuring stable and reliable operation.

Figure 15 DC Power Supply(ATX-P4)

Relay Module (1 Channel)

The relay module controls the DVDRW drive, allowing it to turn on/off and return to its original position after pushing a package.

Figure 16 Relay Module (1 Channel)

Microservo Motors

Microservo motors carry the wooden sticks that push packages into designated boxes. Their precise angular motion ensures accurate sorting.

Figure 17 Microservo

TCS3200 Color Sensor

The TCS3200 sensor detects package colors (red, green, blue, yellow), allowing the system to sort packages based on color criteria.

Figure 18 TCS3200 Color Sensor

Ultrasonic Sensors

Two ultrasonic sensors measure the width of packages to help categorize them by size before sorting.

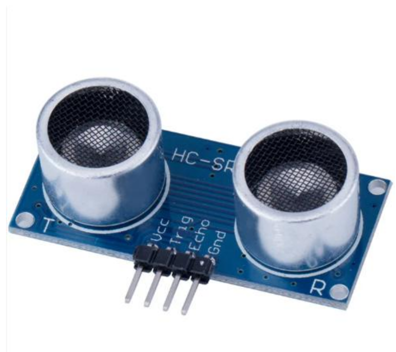


Figure 19 Ultrasonic Sensors

Arduino USB Host Shield

The USB Host Shield allows the Arduino Uno to interface with the barcode scanner, facilitating barcode data transfer to the Mega.

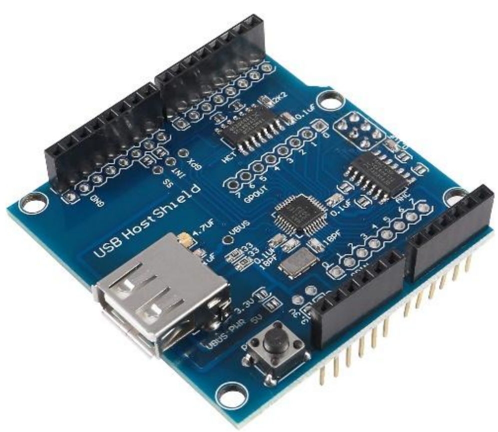


Figure 20 Arduino USB Host Shield

UTP Wires

UTP wires are used for electrical connections between sensors, motors, and controllers, ensuring organized and reliable communication throughout the system.

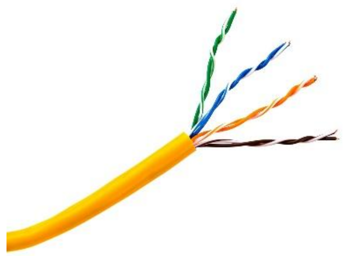


Figure 21 UTP Wires

Process of Work

Initialization

1. When the system powers on via the DC power supply, the Arduino Mega initializes all sensors, motors, and modules.
2. The LCD 16×2 display shows a welcome message and prompts the user to enter a password using the keypad.
3. Upon correct password entry, the LCD displays mode selection options (e.g., sort by color, size, barcode, or mixed mode).
4. The user selects the desired mode via the keypad, which is then sent to the Arduino Mega for processing.
5. The ESP32 connects to the mobile app to allow remote monitoring, control, and viewing package statistics.

Package Detection

1. Packages are manually or automatically placed on the conveyor belt powered by the DC wiper motor and controlled via the BTS7960 motor driver.
2. As packages move along the conveyor, IR sensors detect their presence and position.
    a) Sensors at the start detect the package arrival.
    b) Sensors near the sorting station detect the exact position for mechanical sorting.

Barcode Scanning

1. If the sorting mode involves barcode recognition, the omnidirectional barcode scanner reads the barcode of the package.
2. The scanner passes the barcode data through the USB Host Shield to the Arduino Uno, which then sends it via serial communication to the Arduino Mega.
3. The Mega uses this data to determine the package's destination based on predefined rules.

Color Detection

1. If sorting by color, the package passes under the TCS3200 color sensor.
2. The sensor detects the color of the package (red, green, blue, yellow) and sends the information to the Arduino Mega.
3. The Mega decides which microservo motor and wooden stick will push the package into the correct bin.

Size Detection

1. Two ultrasonic sensors measure the width of the package as it passes through the conveyor.
2. The Arduino Mega uses these measurements to classify packages by size (small, medium, large).
3. Based on size, the Mega selects the corresponding servo motor and stick to sort the package into the appropriate container.

Mechanical Sorting

1. When the package reaches the correct sorting point, IR sensors trigger the microservo motor, which carries a wooden stick.
2. The stick pushes the package into the designated box accurately.
3. If the DVDRW drive is used for pushing larger packages, the relay module controls the on/off movement of the drive.
4. After the push, the stick or DVDRW returns to its original position, ready for the next package.

Conveyor Operation

1. The conveyor continues moving packages under the detection sensors.
2. The Arduino Mega ensures smooth operation by controlling the DC wiper motor via the BTS7960 driver.
3. If multiple packages are detected, the Mega sequences the operations to avoid collisions and ensure proper sorting.

Emergency Handling

1. If the push button (emergency stop) is pressed:
    - All motors (conveyor and servos) immediately stop.
    - The LCD displays an emergency message.
    - The system waits for manual reset before resuming operation.

Monitoring & Reporting

1. All package counts, categories, and sorting actions are logged by the Arduino Mega.
2. The ESP32 sends real-time reports to the mobile app, displaying:
    - Package counts by color, size, or barcode category.
    - Operational status of the system.
    - Mode selection and any alerts or errors.

Continuous Operation

1. The system continuously monitors incoming packages.
2. The process repeats for each package until the conveyor is empty or the emergency button is pressed.
3. Users can change sorting modes anytime via the keypad or mobile app.

Controller Design

Based on the previous steps, we have decided to select the components shown in the figure below(Block Diagram):

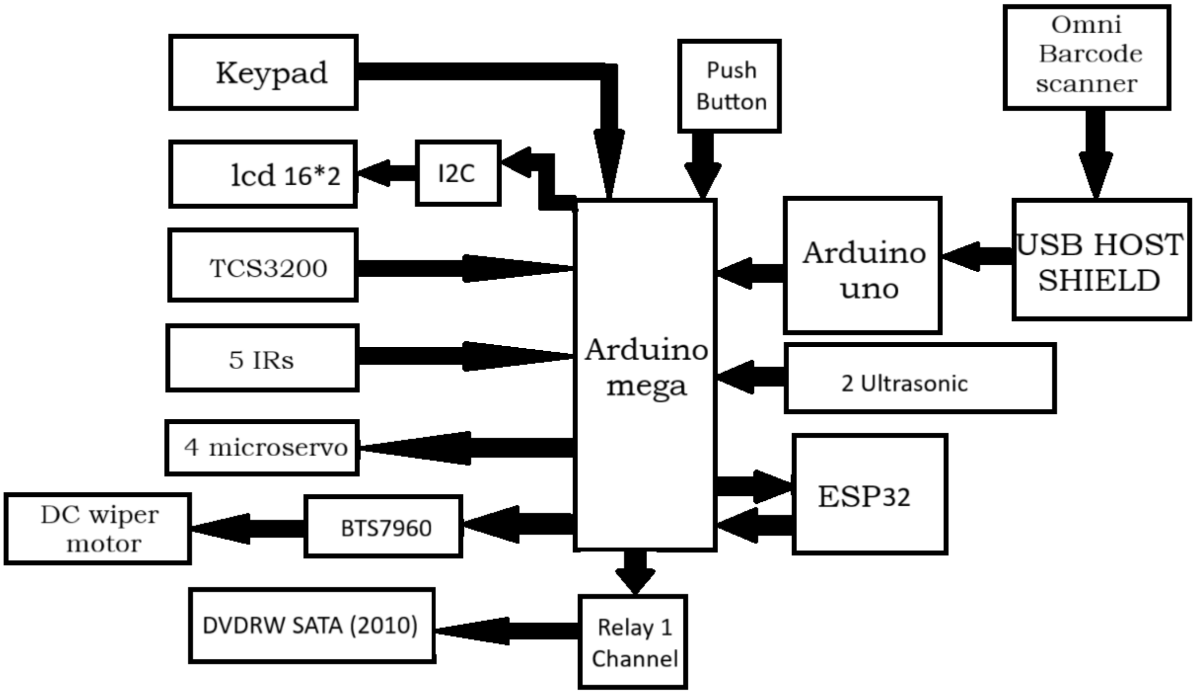
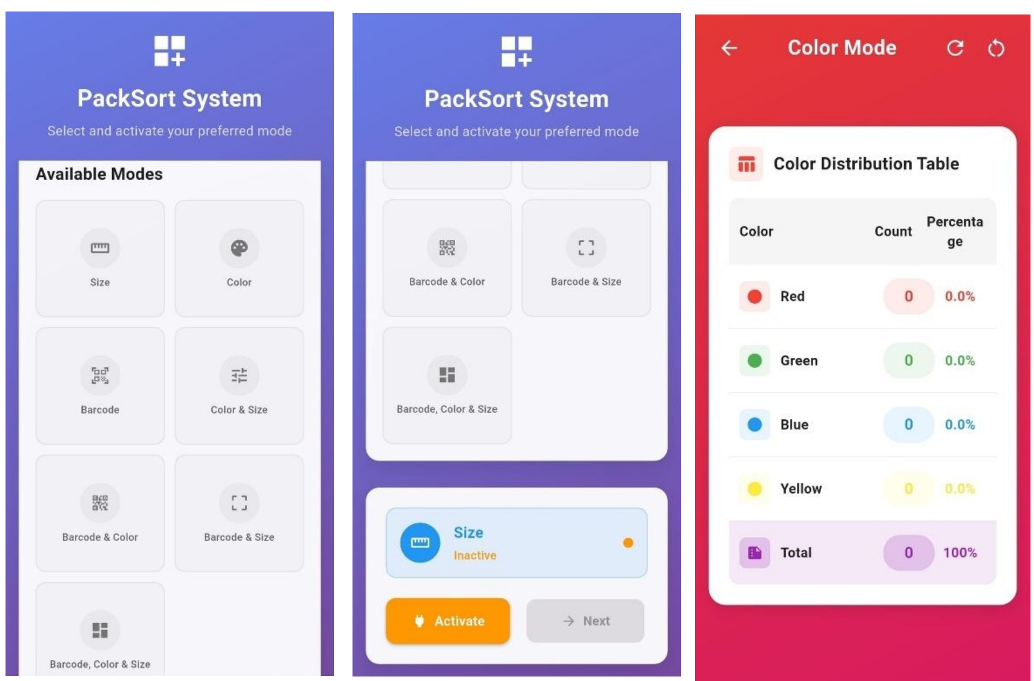


Figure 22 Block Diagram

In this project, the Arduino Mega serves as the main microcontroller, responsible for controlling the conveyor, reading sensor inputs, managing actuators, and coordinating the entire sorting process. It is chosen because of its large number of digital and analog pins, multiple serial ports, and sufficient memory to handle complex system logic. The Arduino Uno is used specifically for barcode handling; it receives scanned barcode data through a USB Host Shield, processes it, and sends the information to the Mega via serial communication. The ESP32 adds wireless capabilities to the system, enabling remote control and monitoring through a mobile app. It allows users to select operation modes, activate or deactivate the system, and view real-time package counts. Together, these three boards integrate seamlessly: the Uno handles barcode input, the Mega manages system operations, and the ESP32 facilitates mobile connectivity and user interaction.



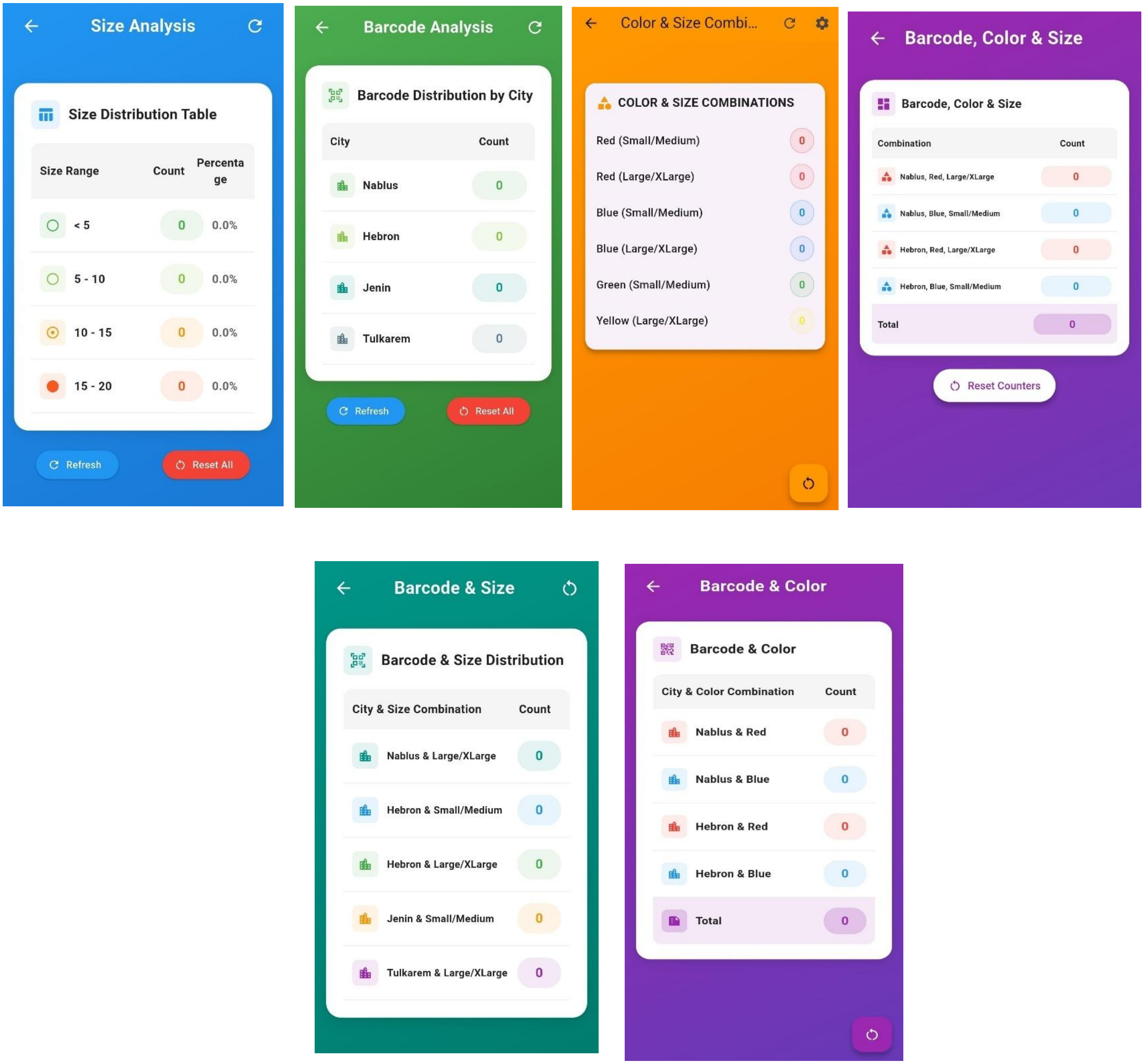


Figure 23 Mobile App Screens

Mechanical Design

The mechanical system of the automated sorting project integrates multiple actuators, sensors, and moving mechanisms to ensure smooth package handling, measurement, and sorting. The setup is designed for reliability, precision, and modularity, allowing packages to move through different inspection and sorting stages automatically.

Package Loading Mechanism

Packages are initially placed in a wooden vertical tower that serves as the package loading area. At the base of this tower, an IR sensor is installed to detect when a package is ready to be deployed. Once detected, a DVD-RW drive mechanism (repurposed from an old DVD drive) acts as a mechanical pusher, extending outward to push the package from the tower onto the motorized conveyor belt. The pusher then retracts automatically, preparing for the next package.

Motorized Conveyor System

The conveyor belt is powered by a DC car wiper motor, driven by a BTS7960 motor driver, and is responsible for transporting packages through the different inspection stations. The conveyor is designed for controlled movement:

1. It starts moving when a package is pushed onto it.
2. It pauses at specific checkpoints where IR sensors are installed, allowing measurements or sorting actions to take place.

Measurement and Sorting Stations

The conveyor carries the packages to various sorting stations based on the selected mode (size, color, or barcode).

1. Size Measurement Station:
   At this checkpoint, two ultrasonic sensors are installed opposite each other to measure the package width (size). The conveyor halts during this process for accurate measurement.
2. Color Detection Station:
   If color-based sorting is selected, the package passes under a TCS3200 color sensor that detects its dominant color (green, blue, red, or yellow).
3. Barcode Station:
   In barcode mode, the package is scanned using an omnidirectional barcode scanner connected through an Arduino USB Host Shield.

Servo Arm Sorting Mechanism

Once a package's size, color, or barcode is identified, the system determines its sorting destination.

1. A micro servo motor with an attached stick or arm is positioned near each sorting bin.
2. When a package reaches its designated position, an IR sensor detects its arrival, triggering the servo arm to push it into the corresponding box.
3. After pushing, the arm retracts, and the conveyor resumes movement.

Emergency Stop and Safety Features

For operational safety, a push button acts as an emergency stop, immediately halting all motion when pressed. This ensures that the operator can intervene in case of mechanical jams or unexpected behavior.

Electrical and Wiring Integration

All actuators and sensors are powered via a DC power supply. The wiring system, primarily using UTP cables, is neatly routed to connect the sensors, motors, servos, and control boards (Arduino Mega, Uno, ESP32). A relay module is used to control the DVD-RW pusher's on/off operation.

System Workflow

1. A package is detected at the tower base by the IR sensor.
2. The DVD drive pusher pushes the package onto the conveyor.
3. The conveyor starts moving and stops at checkpoints (based on mode).
4. Size, color, or barcode data is collected.
5. The servo arm corresponding to the sorting bin pushes the package into the box.
6. The system resets for the next package.

Figure 24 Exterior Design

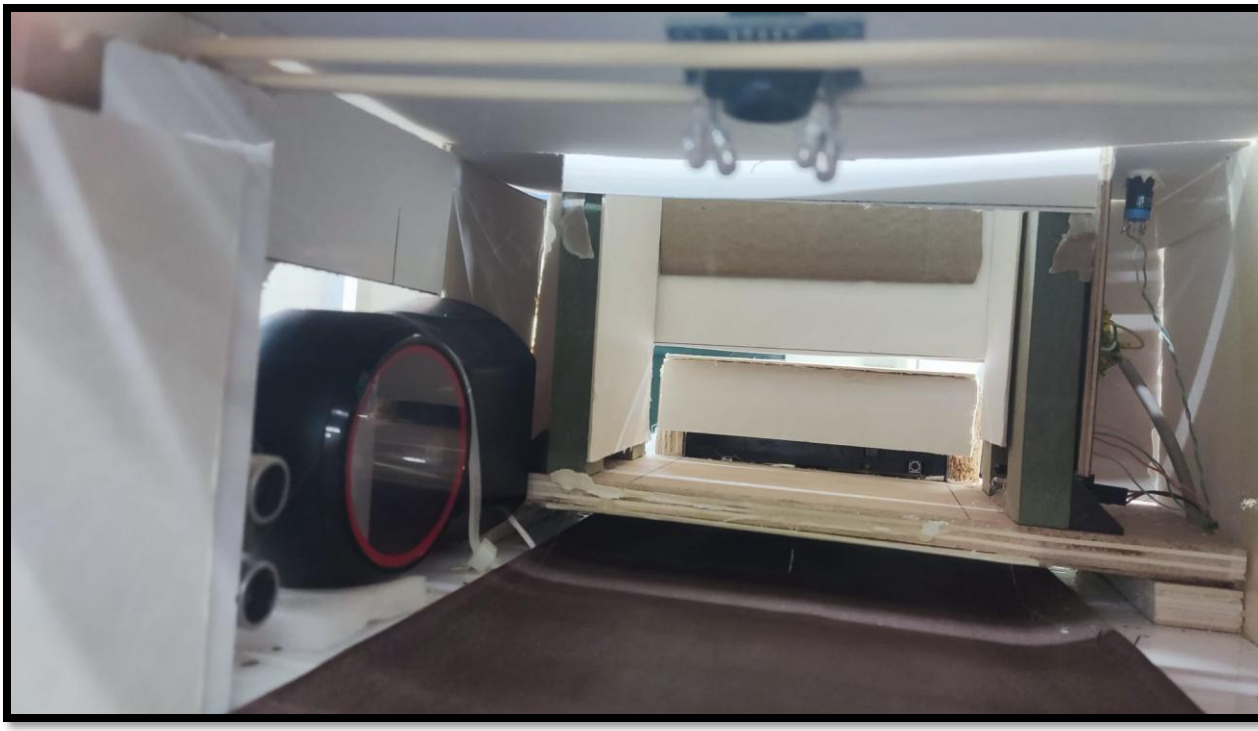


Figure 25 Interior Design

Future Work

1.Weight Classification: Add a load cell or force-sensitive resistor (FSR) under the conveyor to sort packages based on weight categories in addition to size and color.

2.Advanced Barcode/QR Reading: Upgrade to a camera-based scanner (like an ESP32-CAM) to read QR codes and 2D barcodes, which can store significantly more information (e.g., destination address, contents, handling instructions).

3.Enhanced Safety Features: Add more emergency stop buttons along the conveyor and implement light curtains or laser scanners to immediately halt operations if a hand is detected in a dangerous area.

Conclusions and Recommendations

Conclusions

This project successfully created a smart machine that automatically sorts packages in different ways, like by their size, color, or barcode. It cleverly uses old parts like a car motor and DVD drive to push items, making it a cost-effective solution. We also made a mobile app to control the machine and see a report of what was sorted, making it easy to use from anywhere.

Recommendations

In this project, we learned how to connect many different components like sensors, motors, and two Arduino boards to work together as one system.
Therefore, we strongly advise other students to first learn about how serial communication works between microcontrollers before starting a similar complex project. We also urge students to plan their mechanical design very carefully, as making parts work together in the real world is often the biggest challenge.

References

https://projecthub.arduino.cc/

https://www.espressif.com/en/products/socs/esp32

https://www.youtube.com/watch?v=BLrHTHUjPuw

https://www.youtube.com/watch?v=DPqiIzK97K0&t=36521s

https://docs.arduino.cc/retired/shields/arduino-usb-host-shield/

https://www.youtube.com/watch?v=EpXxh4k4NW8

https://www.youtube.com/watch?v=MwdANEcTiPY