An Affordable and Sustainable Efficient Color Sorting System Using Arduino and TCS3200 Sensor

Anurag Giri[1], Sagar Sandhe[2], Rudra Joshi[3], Mr. Mayur Chavda [4], Ms. Apexa Purohit [5], Dr. Anil M. Bisen [6], Dr. Mayank Dev Singh [7], Dr. Jai Bahadur Balwanshi [8]

[1,2,3] UG Student, Department of Mechatronics Engineering, ITM Vocational University, Vadodara

[4,5]Assistant Professor, Department of Mechatronics Engineering, ITM Vocational University, Vadodara

[6] Provost, Professor, Department of Mechanical Engineering, ITM Vocational University, Vadodara

[7] Associate Professor, Department of Mechatronics Engineering, ITM Vocational University, Vadodara

[8] Professor, Department of Mechanical Engineering, ITM Vocational University, Vadodara

DOI : https://doi.org/10.51583/IJLTEMAS.2026.150100004

Received: 29 October 2025; Accepted: 06 November 2025; Published: 22 January 2026

ABSTRACT

In the era of rapid industrialization, automation plays a vital role in enhancing efficiency and accuracy in manufacturing processes. This project presents the design and implementation of an Arduino-based color sorting system that automates the sorting of objects based on their colors. The system uses an infrared (IR) sensor to detect the presence of objects on a conveyor belt and a TCS3200 color sensor to identify their color. The Arduino microcontroller processes the sensor data and controls a servo motor that directs objects to designated bins according to their detected color red, green, or blue. This automated solution offers an affordable and effective alternative to expensive industrial sorting machines, making it suitable for small-scale industries and educational purposes. The system also addresses challenges such as sensor calibration and ambient light variations to ensure accurate color detection and sorting. Testing under various conditions demonstrates the system's reliability, accuracy, and efficiency, highlighting its potential for wider adoption in automation and prototyping.

Keywords: Automation, TCS3200 color sensor, infrared sensor, servo motor, conveyor belt, microcontroller, object detection, low-cost automation

INTRODUCTION

Automation has become a cornerstone of modern industry, revolutionizing traditional manufacturing and processing techniques. The drive for higher efficiency, enhanced precision, and reduced operational costs has led industries to increasingly adopt automated systems. Among the many automation applications, color sorting systems hold a pivotal role across various sectors such as food processing, recycling, packaging, and manufacturing. These systems enable the segregation of objects based on their color attributes, which is crucial in ensuring product quality, reducing waste, and optimizing production workflows.

Manual sorting processes, which have been the norm historically, are labor-intensive, slow, and prone to human error. Such inefficiencies become especially problematic in high- volume environments where consistency and speed are critical. In contrast, automated color sorting systems offer a scalable solution, delivering consistent accuracy, rapid processing, and labor cost reduction. Nevertheless, commercially available industrial color sorters are typically expensive, bulky, and complex, often placing them beyond the reach of small-scale industries and educational institutions.

This research introduces an Arduino-based color sorting system designed to provide a cost-effective, accessible, and easy-to-implement automation solution. Leveraging the popular open-source Arduino platform combined

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue I, January 2026

with readily available sensors and actuators, the system automates the sorting of objects based on three primary colors: red, green, and blue. The system integrates a conveyor belt for continuous object movement, an infrared (IR) sensor for object detection, a TCS3200 color sensor for precise color identification, and a servo motor for the sorting actuation.

The primary objectives of this project are to design and develop a functional prototype capable of accurately sorting colored objects, evaluate its performance under varying conditions, and demonstrate its potential for educational and small industrial applications. By doing so, this project underscores how accessible technologies can empower users to harness automation without the need for expensive or complex machinery.

LITERATURE REVIEW

Color sorting systems have evolved significantly over the past decades, driven by increasing industrial demands for automation and accuracy. Initially, color sorting was performed manually, a process that was time-consuming and prone to human error. With growing industrial needs, mechanical sorting solutions emerged, yet they still required substantial human oversight and lacked the precision required by modern production lines [1].

Automation has significantly transformed industrial manufacturing by improving productivity, accuracy, and consistency while reducing dependence on manual labor. Early studies on color-based segregation relied on manual and mechanical techniques, which, although functional, were limited by low throughput, operator fatigue, and high error rates, particularly in large-scale operations [1], [11]. These limitations motivated the transition toward automated and sensor-based sorting mechanisms.

Machine vision has emerged as a powerful solution for industrial sorting applications due to its ability to analyze complex visual features. Johnson and Lee [2] highlighted advances in vision-based sorting systems, emphasizing their high accuracy and adaptability. However, such systems often involve high computational complexity, expensive cameras, and intensive calibration requirements, making them unsuitable for small-scale industries and educational setups. Similar challenges associated with vision-based approaches were also discussed by Park et al. [15], where system cost and lighting sensitivity were identified as major constraints.

To overcome these challenges, optical color sensors have been explored as a low-cost alternative for color detection. Patel and Kumar [3] provided a comprehensive overview of optical color sensors used in industrial environments, highlighting their reliability and ease of integration. Among these, the TCS3200 color sensor has gained significant attention due to its affordability and compatibility with microcontroller platforms. Chen et al. [4] demonstrated a microcontroller-based color sorting system using the TCS3200 sensor, achieving satisfactory accuracy for basic RGB classification. However, their work emphasized the need for improved calibration techniques to handle varying illumination conditions.

Sensor calibration and adaptive techniques have been further investigated to enhance sorting accuracy. Zhao et al. [5] proposed adaptive algorithms for color sensor calibration, improving robustness under environmental variations. Despite these advancements, the integration of such algorithms increases system complexity and processing requirements, which may not be ideal for entry-level automation platforms.

The adoption of open-source microcontroller platforms, particularly Arduino, has enabled rapid prototyping of automation systems. Martinez and Lopez [6] reviewed Arduino-based automation solutions, concluding that Arduino platforms provide an optimal balance between cost, flexibility, and ease of programming. Several studies have successfully implemented Arduino-based color sorting systems for educational and small-scale industrial use. Singh and Kumar [7] developed a color sorting mechanism aimed at teaching mechatronics concepts, demonstrating the effectiveness of Arduino in academic environments. Similarly, Rahman et al. [18] presented an Arduino-controlled color sorting robot, highlighting its suitability for low-budget automation projects.

Despite these successes, low-cost systems face challenges related to accuracy, scalability, and reliability. Gomez et al. [8] analyzed the limitations of budget-friendly color sorting systems, identifying sensor noise, inconsistent lighting, and mechanical misalignment as primary sources of error. Gupta and Mishra [16] further emphasized

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue I, January 2026

the importance of precise synchronization between object detection, conveyor motion, and actuation mechanisms in conveyor-based sorting systems.

Infrared (IR) sensors have been widely used for object detection and positioning in automated conveyor systems. Nguyen et al. [10] demonstrated effective integration of IR sensors in conveyor-based sorting, enabling accurate object presence detection and improved timing control. Zhou and Sun [19] explored sensor fusion techniques combining optical and proximity sensors, achieving enhanced system reliability and fault tolerance.

Actuation mechanisms such as servo motors play a critical role in sorting accuracy. Al-Jabri and Al-Habsi [17] analyzed microcontroller-driven servo systems, confirming their suitability for precise angular positioning in industrial automation. Conveyor speed control using PWM-based motor drivers has also been studied extensively, with Sharma and Jain [20] highlighting its importance in balancing throughput and detection accuracy.

Recent research has explored the integration of intelligent algorithms, including machine learning, to further improve sorting performance. Wang and Li [9] discussed the application of machine learning techniques in automated sorting systems, demonstrating improved classification accuracy. However, such approaches require higher computational resources, which may conflict with the simplicity and low-cost objectives of Arduino-based systems.

Finally, the role of Arduino-based automation in education has been strongly emphasized in literature. Brown and Green [12] and Chatterjee and Banerjee [22] noted that such systems provide hands-on learning opportunities, bridging the gap between theoretical knowledge and real-world industrial practices.

Evolution of Color Sorting Technologies

Early automated color sorting efforts utilized machine vision systems incorporating cameras and advanced image processing algorithms. These systems enhanced sorting accuracy and speed but were complex and expensive, restricting their use mainly to large-scale industrial operations [2]. The dependency on sophisticated hardware and processing power limited adoption by smaller enterprises and educational institutions.

Advances in Sensor Technologies

Recent progress in optical sensors, especially color sensors like the TCS230 and TCS3200, has reduced barriers to automation projects. These sensors operate by illuminating objects and measuring the intensity of reflected light through red, green, and blue photodiodes, outputting frequency signals correlated with color intensity [3]. Their affordability, ease of integration, and acceptable accuracy have popularized their use in low-cost color sorting systems.

Several studies have demonstrated the feasibility of these sensors in color sorting applications. For instance, Chen *et al.* developed a prototype employing a TCS3200 sensor with microcontroller control, achieving sorting accuracy above 90% in controlled conditions [4]. Nonetheless, these sensors are vulnerable to inaccuracies due to ambient lighting variations, object reflectivity, and surface texture, necessitating calibration or adaptive correction algorithms [5].

Arduino as a Microcontroller Platform in Automation

Arduino has become a widespread platform for prototyping automation solutions due to its low cost, open-source design, and strong community support [6]. Its compatibility with numerous sensors and actuators facilitates rapid development of embedded systems, including color sorting mechanisms.

Prior works have utilized Arduino to control color sensors, conveyor motors, and servo actuators for sorting applications. Singh and Kumar presented an Arduino-based sorting proto- type for educational use that effectively sorted colored objects, establishing a basis for further refinement [7]. Arduino’s ease of programming and modularity support customization and scalability, advantageous for both research and practical deployments.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue I, January 2026

Challenges in Low-Cost Color Sorting Systems

Despite advances, challenges remain in low-cost color sorting. Sensor accuracy is highly sensitive to environmental factors like lighting changes and object positioning [8]. Additionally, mechanical constraints such as conveyor speed and actuator precision affect throughput and sorting quality.

Research efforts have addressed these issues with algorithmic enhancements including adaptive thresholding, sensor fusion, and machine learning methods [9]. The use of IR sensors for object detection has also improved sorting reliability by ensuring precise timing of color detection, minimizing motion blur and misclassification [10].

Industrial and Educational Applications

Color sorting benefits industrial processes by enhancing quality control, reducing waste, and streamlining manufacturing [11]. Meanwhile, low-cost Arduino-based systems serve as valuable educational tools, providing hands-on experience in automation, sensor integration, and embedded programming [12].

From the reviewed literature, it is evident that while high-end machine vision systems offer superior performance, their cost and complexity limit accessibility. Low-cost Arduino-based color sorting systems present a viable alternative but often suffer from calibration challenges and limited scalability. This research addresses these gaps by proposing a cost-effective, modular, and easily deployable Arduino-based color sorting system that balances accuracy, simplicity, and educational as well as small-industrial applicability.

SYSTEM DESIGN AND ARCHITECTURE

The system design and architecture of the proposed Arduino-based color sorting system constitute the fundamental framework that integrates mechanical, electrical, and software components into a unified automated solution. The overall design philosophy emphasizes modularity, affordability, scalability, and operational reliability, making the system suitable for deployment in small-scale industrial environments as well as academic laboratories and training setups. This section presents a comprehensive discussion of the conceptual framework, component-level interactions, hardware selection rationale, control logic, and the complete operational workflow of the system.

A. Conceptual Overview

The primary objective of the proposed system is to automate the sorting of objects based on their color attributes using sensor-driven decision-making and microcontroller-based control. The conceptual model follows a sequential material-handling process, where objects are transported via a conveyor belt, detected at a predefined sensing zone, analyzed for color characteristics, and finally diverted to designated output bins using a mechanical sorting mechanism.

At the core of the system lies the Arduino Uno microcontroller, which functions as the central control unit responsible for coordinating sensor data acquisition, executing color classification algorithms, and controlling actuators in real time. The Arduino platform was selected due to its open-source nature, ease of programming, wide community support, and compatibility with various sensors and actuators.

An infrared (IR) proximity sensor is employed to detect the presence of objects on the conveyor belt, enabling precise synchronization between object detection and color sensing. For accurate color identification, the TCS3200 color sensor is utilized, which provides reliable RGB color intensity measurements through frequency-based outputs. A servo motor acts as the mechanical actuator that physically directs the object into the appropriate bin based on the detected color. Together, these components form a closed-loop automation system that operates with minimal human intervention.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue I, January 2026

B. Detailed Component Interaction and System Workflow

The proposed system operates in a structured, closed-loop manner, ensuring continuous feedback between sensors, the microcontroller, and actuators. The complete operational workflow is described as follows:

Object Loading and Transportation

Objects to be sorted are placed manually or automatically onto a conveyor belt driven by a DC motor. The conveyor facilitates continuous and orderly movement of objects through the system. Motor speed is regulated using a motor driver module controlled via Pulse Width Modulation (PWM) signals generated by the Arduino. Precise speed control is essential to ensure that objects arrive at the detection zone at a manageable and consistent rate, preventing overlapping or missed detections.

Object Detection Using IR Sensor

An IR sensor module, consisting of an infrared emitter and receiver pair, is positioned near the conveyor's entry to the sensing zone. When an object interrupts the IR beam, the sensor outputs a digital signal indicating object presence. This signal serves as a trigger for the Arduino to initiate the sorting cycle. The use of an IR sensor ensures reliable object detection regardless of object color or surface texture.

Conveyor Halt for Sensor Stability

Upon receiving the detection signal, the Arduino momentarily stops the conveyor motor. Halting the conveyor during the sensing phase minimizes vibrations and motion-induced errors, allowing the color sensor to obtain stable and repeatable readings. This step significantly enhances measurement accuracy and reduces misclassification.

Color Detection Using TCS3200 Sensor

The TCS3200 color sensor detects color by converting the intensity of reflected red, green, and blue light into corresponding frequency signals. The sensor contains an array of photodiodes with integrated color filters. The Arduino sequentially activates these filters by controlling the sensor's selection pins and measures the output frequency for each color channel. These frequency values are inversely proportional to the intensity of reflected light.

Data Processing and Color Classification

The Arduino processes the acquired frequency signals using an embedded algorithm. The algorithm normalizes the RGB values to compensate for ambient lighting variations and sensor noise. The dominant color is identified by comparing the relative intensity levels of the red, green, and blue channels. Predefined threshold values and hysteresis margins are incorporated to avoid unstable decisions caused by borderline or fluctuating sensor readings, thereby improving classification robustness.

Servo Motor Actuation for Sorting

Once the color classification is complete, the Arduino generates PWM control signals to position the servo motor at specific angular positions corresponding to predefined sorting bins. The servo motor operates a mechanical deflector or gate that redirects the object into the appropriate bin. The high positional accuracy of the servo ensures consistent and repeatable sorting performance.

Conveyor Restart and Cycle Continuation

After the object is successfully sorted, the servo motor returns to its neutral position. The Arduino then restarts the conveyor motor, allowing the next object to enter the detection zone. This cyclic process continues until all objects are sorted.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue I, January 2026

C. Design Considerations and System Optimization

Several critical design considerations were addressed to ensure reliable and efficient system operation:

Sensor Placement and Alignment:

The IR sensor is strategically positioned upstream of the color sensor to provide adequate response time for stopping the conveyor. The distance between the IR and color sensors is calibrated to ensure precise object positioning during color detection. The TCS3200 sensor is mounted perpendicular to the conveyor surface at a fixed height to maintain consistent reflectance measurements [25].

Lighting Control and Ambient Interference:

To mitigate the impact of ambient lighting variations, the color sensing region is enclosed within a light-shielded housing. Integrated white LEDs provide uniform illumination, ensuring consistent sensor readings across different environmental conditions.

Conveyor Speed Calibration:

Conveyor speed is optimized through experimental testing to balance throughput and detection accuracy. Lower speeds improve sensing accuracy, while higher speeds increase productivity. An optimal operating speed is selected to meet both requirements.

Servo Motor Selection:

The servo motor is chosen based on torque capacity, angular resolution, and response time. Adequate torque ensures reliable movement of the sorting mechanism without stalling, while fast response time maintains efficient system cycle times.

Power Supply Stability:

A regulated Switched-Mode Power Supply (SMPS) is employed to provide stable voltage and current to all system components. Power stability is crucial to prevent sensor drift, communication errors, and actuator malfunctions.

D. Control Logic and Algorithmic Implementation

The control software implemented on the Arduino follows a finite state machine (FSM) architecture to ensure deterministic and structured system behavior. The primary operational states include:

Idle State: Conveyor operates continuously while monitoring IR sensor input.

Detection State: Conveyor halts upon object detection.

Sensing State: RGB color data acquisition and processing.

Sorting State: Servo motor actuated based on classification result.

Reset State: Servo returns to neutral position and conveyor restarts.

This state-based approach ensures precise synchronization between sensing, decision-making, and actuation, preventing overlapping operations and enhancing system reliability.

E. System Block Diagram and Signal Flow

The system block diagram, illustrated in Figure 1, represents the interaction among all components:

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue I, January 2026

Sensors:

The IR sensor provides digital presence signals, while the TCS3200 outputs frequency-based RGB data.

Microcontroller:

The Arduino Uno processes sensor inputs, executes classification algorithms, and generates PWM signals.

Actuators:

The DC motor drives the conveyor via a motor driver module, and the servo motor performs sorting actions.

Power Supply:

The SMPS delivers regulated power to sensors, controllers, and actuators.

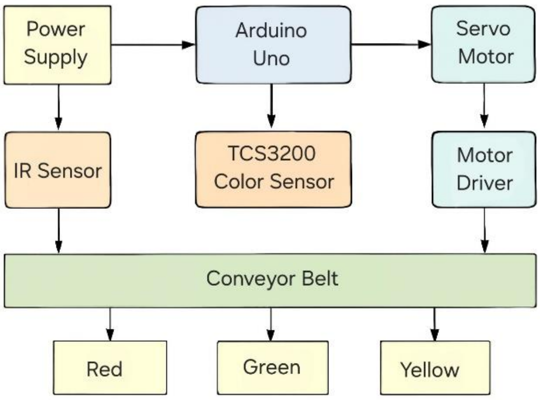

Fig. 1. System Block Diagram and Signal Flow

The diagram highlights a closed-loop architecture where sensor feedback continuously influences actuator behavior, resulting in an efficient and autonomous color sorting system.

F. Scalability and Modularity

The modular design supports future system expansion:

- Adding additional color detection bins is achievable by integrating extra servo motors and extending the control algorithm.
- Sensor upgrades to higher resolution or multi-spectral devices can be incorporated with minimal changes to the control logic.

Integration with communication modules (e.g., Bluetooth, Wi-Fi) can enable remote monitoring and data logging for industrial IoT applications.

Hardware Components

The overall performance, accuracy, and operational reliability of the proposed Arduino-based color sorting system are strongly influenced by the appropriate selection and seamless integration of hardware components. Each module plays a critical role in ensuring accurate sensing, precise control, and stable mechanical operation. The hardware architecture has been carefully designed to balance cost-effectiveness, functional robustness, and ease of implementation, making the system well suited for small-scale industrial automation as well as educational and research applications. This section provides a detailed description of the primary hardware components, including their specifications, operating principles, and functional roles within the system.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue I, January 2026

A. Arduino Uno Microcontroller

The Arduino Uno serves as the central processing and control unit of the color sorting system. It is based on the ATmega328P, an 8-bit AVR microcontroller widely used in embedded control applications due to its reliability, low power consumption, and extensive peripheral support.

The Arduino Uno operates at a clock frequency of 16 MHz and provides 32 KB of flash memory for program storage, 2 KB of SRAM for dynamic data handling, and 1 KB of EEPROM for non-volatile data retention. The board operates at 5 V, which ensures compatibility with most commonly available sensors and actuators. It offers 14 digital input/output pins, of which six support PWM functionality, and six analog input pins with a 10-bit analog-to-digital converter (ADC) resolution.

In addition to general-purpose I/O, the Arduino Uno supports multiple communication protocols, including UART (serial communication), SPI, and I2C, enabling integration with external modules such as LCD displays, data loggers, and additional sensors. Within the proposed system, the Arduino processes digital trigger signals from the IR sensor and frequency-based outputs from the TCS3200 color sensor. Based on the processed data, it generates PWM signals to control the conveyor belt motor and servo motor for sorting actuation. The simplicity of Arduino programming using the Arduino IDE further enhances the system's adaptability and maintainability.

B. TCS3200 Color Sensor Module

The TCS3200 color sensor is a key sensing component responsible for accurate color detection. It is an optical sensor capable of measuring the intensity of reflected light across red, green, and blue color channels. The sensor consists of an array of photodiodes arranged in an 8×8 matrix, where each photodiode is filtered to respond to red, green, blue, or clear (unfiltered) light.

The TCS3200 operates within a voltage range of 2.7 V to 5.5 V, making it directly compatible with the Arduino Uno. The sensor outputs a square-wave frequency signal whose frequency is inversely proportional to the intensity of the incident light for the selected color channel. Color selection is achieved using control pins S2 and S3, while output frequency scaling is controlled using S0 and S1, allowing sensitivity adjustment at 2%, 20%, or 100% of the full-scale output.

Integrated white LEDs provide consistent illumination of the object surface, reducing dependency on ambient lighting conditions. During operation, the Arduino sequentially activates red, green, and blue filters, measures the corresponding output frequencies, and computes relative color intensities. This frequency-based sensing mechanism offers improved noise immunity and repeatability compared to purely analog color sensors, making it suitable for real-time industrial and educational applications.

C. Infrared (IR) Sensor Module

The infrared (IR) sensor module is employed for reliable object detection on the conveyor belt. It consists of an IR LED emitter and a photodiode or phototransistor receiver arranged to detect reflected infrared radiation from nearby objects.

The sensor operates at an input voltage range of 3.3 V to 5 V and typically provides an adjustable detection range from 2 cm to 30 cm, which can be fine-tuned using an onboard potentiometer. The module outputs a digital signal that transitions to a HIGH logic level when an object is detected and remains LOW otherwise.

This simple yet effective detection mechanism enables precise synchronization between object arrival and color sensing operations. By providing a trigger to temporarily halt the conveyor belt, the IR sensor ensures that the object remains stationary during color measurement, significantly improving sensing accuracy and minimizing errors caused by motion or misalignment.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue I, January 2026

D. Servo Motor

A servo motor is utilized as the primary actuation device for the sorting mechanism. Servo motors are preferred in such applications due to their precise angular control, built-in feedback mechanisms, and ease of interfacing with microcontrollers.

Standard hobby-grade servo motors operate within a voltage range of 4.8 V to 6 V and offer angular motion typically between 0° and 180°. Torque ratings vary depending on the model, generally ranging from 2.5 kg-cm to 10 kg-cm, which is sufficient for actuating lightweight sorting gates or deflectors.

The servo motor is controlled using PWM signals generated by the Arduino, where pulse widths of approximately 1 ms to 2 ms correspond to angular positions from 0° to 180°. Based on the classified object color, the servo motor is positioned to direct the object into the appropriate bin. The internal feedback mechanism ensures accurate positioning, repeatability, and minimal overshoot, contributing to consistent sorting performance.

E. Conveyor Belt and DC Motor

The conveyor belt assembly provides a mechanical platform for transporting objects through the sensing and sorting stages. It consists of a smooth, durable belt mounted over rollers and driven by a DC motor.

The DC motor converts electrical energy into rotational motion, driving the conveyor belt at a controllable speed. Speed regulation is achieved using PWM signals supplied through a motor driver module. The physical dimensions of the conveyor belt are selected based on the size, weight, and shape of the objects being sorted, ensuring smooth movement and minimizing slippage or misalignment.

The controlled and uniform movement of objects along the conveyor is critical for ensuring sufficient time for object detection, color sensing, and mechanical sorting without compromising throughput.

F. Motor Speed Controller

The motor speed controller module enables precise control of the conveyor belt speed using Pulse Width Modulation (PWM) techniques. PWM varies the effective voltage supplied to the motor by rapidly switching the power signal on and off, with the duty cycle determining the motor speed.

Typically designed to operate with input voltages ranging from 12 V to 24 V DC, the motor controller includes built-in protection features such as overcurrent, thermal shutdown, and short-circuit protection. These safeguards enhance system reliability and prolong component lifespan.

The Arduino dynamically adjusts PWM duty cycles to slow down the conveyor during sensing operations and speed it up during idle transport, optimizing both accuracy and productivity.

G. Switched-Mode Power Supply (SMPS)

A Switched-Mode Power Supply (SMPS) is used to provide regulated DC power to all system components. The SMPS converts AC mains voltage (110 V / 220 V) into stable DC outputs such as 5 V, 12 V, or 24 V, depending on system requirements.

SMPS units are characterized by high efficiency, compact size, and minimal heat dissipation compared to linear power supplies. Additionally, they incorporate protection mechanisms against overvoltage, overload, and short-circuit conditions. Stable power delivery is crucial for preventing sensor drift, microcontroller resets, and actuator malfunctions, particularly in environments with fluctuating power conditions.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue I, January 2026

H. I2C LCD Display Module

An optional I2C-based LCD display is integrated to enhance user interaction and system monitoring. The display typically supports 16×2 or 20×4 character formats and communicates with the Arduino using the I2C protocol, which requires only two data lines (SDA and SCL), thereby reducing wiring complexity.

The LCD provides real-time feedback on system status, including detected color, sorting decisions, conveyor operation status, and error notifications. Adjustable contrast and backlighting ensure readability under various lighting conditions. This feature is particularly useful for debugging, demonstrations, and educational use, eliminating the need for continuous connection to an external computer.

Integration and Assembly

The successful operation of the Arduino-based color sorting system relies not only on the selection of individual hardware components and implementation of software logic but also critically on their precise integration and physical assembly. This section details the comprehensive process of combining all components into a functional prototype, encompassing mechanical assembly, electrical wiring, component placement, system calibration, troubleshooting, and safety considerations to ensure optimal performance and reliability.

A. Mechanical Assembly

The mechanical framework forms the structural foundation of the system, supporting the conveyor belt, sensors, actuators, and wiring while ensuring alignment and operational stability.

- **Conveyor Belt Setup:** The conveyor belt is mounted on a rigid frame constructed from aluminum profiles or wooden panels to minimize vibrations and maintain mechanical integrity. The belt's dimensions are selected based on object size and desired throughput.
- **Motor Mounting:** The DC motor responsible for driving the conveyor belt is securely affixed to the frame, with a coupling mechanism connecting the motor shaft to the conveyor roller to enable smooth motion transmission. The motor speed controller is positioned nearby to facil- itate efficient wiring and heat dissipation.
- **Sensor Placement:** The IR sensor is installed upstream of the color sensor and aligned perpendicularly to the conveyor surface. Its height and angle are adjusted to maximize detection accuracy. The TCS3200 color sensor is mounted above the conveyor within a light-blocking enclosure to mitigate ambient light interference. The sensor-to-belt distance is experimentally optimized to ensure stable and accurate color measurements.
- **Servo Motor and Sorting Mechanism:** The servo motor is mounted adjacent to sorting bins, with a lightweight but robust arm or flap attached to deflect objects into the appropriate bins. The design minimizes servo load while ensuring reliable sorting action.
- **Bins and Object Collection:** Multiple collection bins are arranged at fixed positions corresponding to the servo motor's angular settings. Each bin is clearly labeled with its associated color category to facilitate verification and maintenance.

B. Electrical Wiring and Connections

Robust electrical wiring practices are essential for maintaining signal integrity, ensuring power stability, and safeguarding against hazards.

- **Power Distribution:** Regulated voltages are distributed to all components from a central power supply. Power lines are routed to minimize electromagnetic interference, and overcurrent protection is implemented using fuses or circuit breakers.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue I, January 2026

- **Signal Wiring:** Sensor output lines, including digital and frequency signals, connect to designated Arduino input pins, with shielded cables employed as necessary to reduce electromagnetic interference. PWM outputs from the Arduino control the servo motor and motor driver modules, with attention paid to preserving signal quality.
- **Grounding:** A common ground reference is maintained across all components to prevent ground loops and signal distortion.
- **I2C Communication:** For systems incorporating an I2C LCD display, the SDA and SCL lines are connected to the Arduino with appropriate pull-up resistors to ensure reliable communication.
- **Cable Management:** Wiring is organized and secured us- ing cable ties, routed away from moving parts to prevent mechanical wear, damage, or accidental disconnections.

C. *System Calibration*

Calibration is critical to optimizing system performance under practical operating conditions.

- **Sensor Calibration:** The IR sensor's detection range and sensitivity are tuned via its onboard potentiometer to reliably detect objects of varying dimensions and sur- face properties. The TCS3200 color sensor is calibrated using standardized color samples to establish frequency- to-color mapping thresholds, compensating for ambient lighting and surface reflectivity variations.
- **Servo Motor Calibration:** Angular positions corre- sponding to sorting bins are fine-tuned to ensure the sorting arm aligns precisely with each bin, preventing misdirection or spillage.
- **Conveyor Speed Tuning:** The conveyor belt speed is adjusted via the motor speed controller and Arduino PWM parameters to balance throughput requirements with sensor reading accuracy.
- **System Testing:** Comprehensive testing with a variety of sample objects under diverse lighting conditions verifies sorting accuracy and identifies areas for mechanical or electrical refinement.

D. *Troubleshooting and Optimization*

Systematic troubleshooting during integration addresses common operational challenges:

- **Sensor Signal Noise:** Implementation of shielded wiring and modifications to sensor enclosures reduce electrical noise and improve signal fidelity.
- **Servo Motor Jitter:** Stabilization of power supply lines and refinement of PWM signals mitigate erratic servo movements.
- **Conveyor Belt Slippage:** Adjustments to belt tension and motor torque ensure consistent and reliable object transport.
- **Synchronization Delays:** Optimization of code execution and strategic sensor placement minimize latency between object detection and sorting actuation.

E. *Safety Considerations*

The assembled system incorporates multiple safety features to protect users and equipment:

- Emergency stop switches enable immediate shutdown of the conveyor and motors in the event of a malfunction or hazard.
- Electrical wiring and components are insulated to prevent shocks and short circuits.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue I, January 2026

- Moving parts are enclosed or shielded where feasible to prevent accidental contact and injury.

Experimental Setup and Testing

The experimental phase is critical for validating the design assumptions of the Arduino-based color sorting system, assessing its operational performance, and identifying opportunities for further improvement. This section describes the test environment, objectives, procedures, performance metrics, and results analysis methods employed to evaluate the system's functionality and robustness across varied conditions.

A. ***Test Environment***

Experiments were conducted within a controlled indoor laboratory environment to reduce the influence of extraneous variables. The test setup comprised:

- The fully assembled color sorting system as outlined in previous sections.
- Adjustable lighting sources enabling simulation of di- verse ambient conditions, including natural daylight, fluorescent illumination, and a darkened enclosure with integrated white LEDs housed within the color sensor enclosure.
- Sample objects of varied sizes, shapes, surface textures, and primary colors (red, green, and blue), either fabricated or sourced to assess system versatility.
- Measurement instruments such as a digital timer, multi- meter, and oscilloscope for precise timing and electrical signal verification.

B. ***Test Objectives***

The testing phase was designed to fulfill the following primary objectives:

- **Accuracy Assessment:** Quantify the system's ability to correctly classify and sort objects based on color.
- **Throughput Measurement:** Determine the sorting speed and volume of objects processed per unit time.
- **Robustness Evaluation:** Examine performance consistency under variable lighting conditions and object characteristics.
- **Reliability and Repeatability:** Assess the frequency of sorting errors and system stability during prolonged operation.

C. ***Test Procedure***

The experimental methodology was conducted in sequential stages:

- **Baseline Calibration:** Initial calibration of sensors and actuators using standardized color samples under controlled lighting conditions.
- **Object Sorting Trials:** Random batches of color-coded objects were introduced onto the conveyor belt. Sorting outcomes were recorded manually and via Arduino's internal counters.
- **Lighting Variation Tests:** System accuracy was evaluated under different ambient lighting scenarios to assess sensitivity and calibration robustness.
- **Object Variation Tests:** Objects differing in size, shape, and surface reflectivity were tested to evaluate detection and sorting consistency.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue I, January 2026

Continuous Operation Testing: Extended runtime tests were conducted to observe system stability, sensor drift, mechanical wear, and error accumulation.

Fig. 2. Prototype

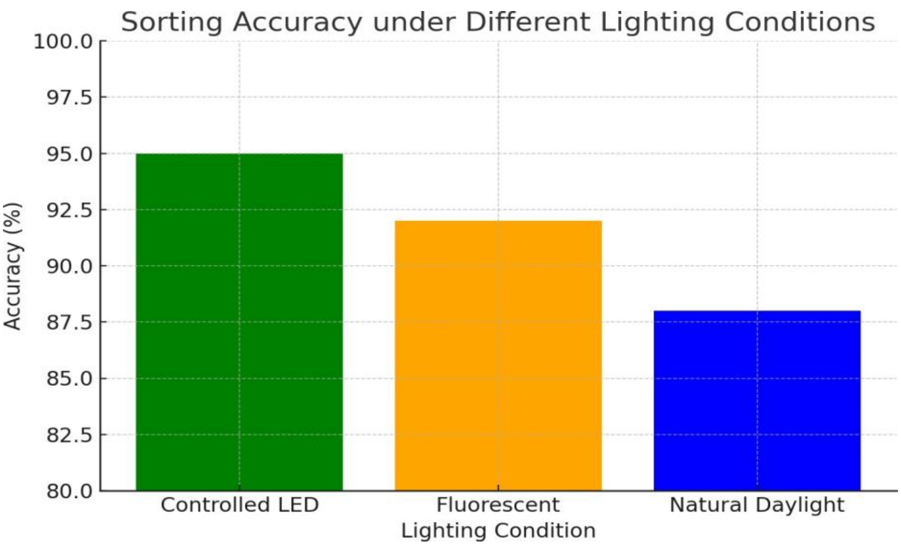

Fig. 3. Sorting accuracy under different ambient lighting conditions, showing highest accuracy with controlled LED lighting.

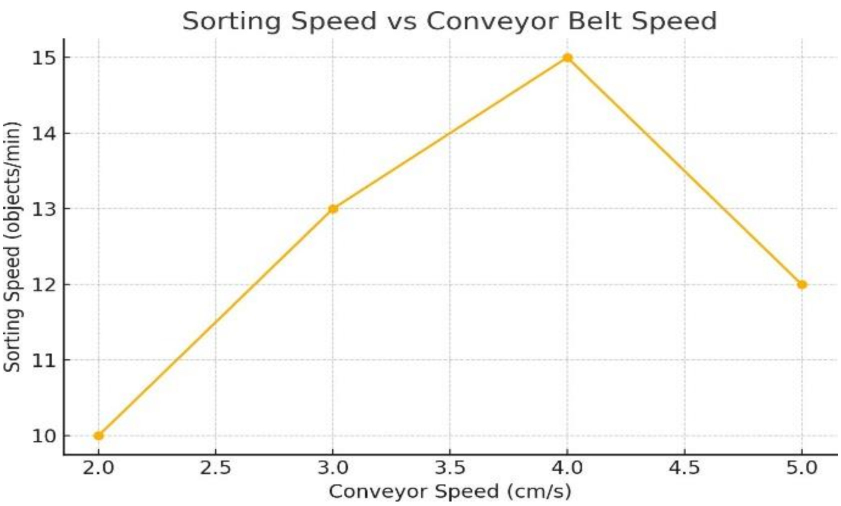

Fig. 4. Sorting speed (objects per minute) as a function of conveyor belt

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue I, January 2026

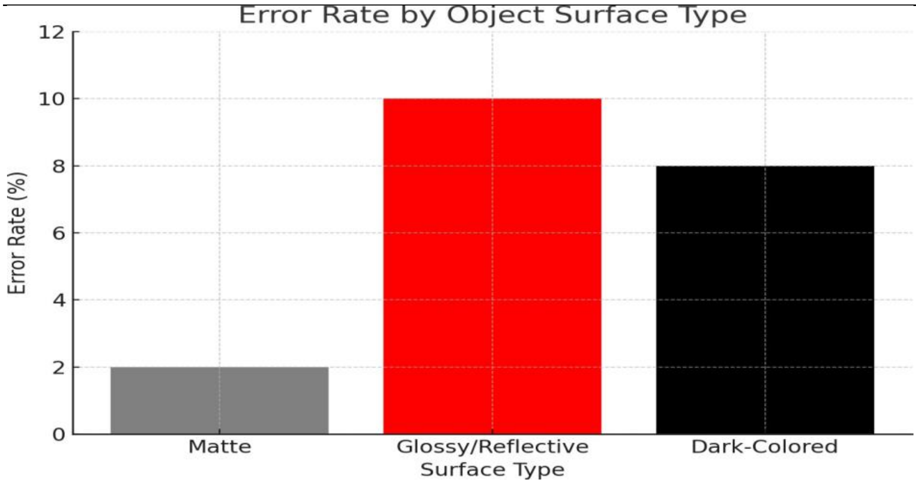

Fig. 5. Error rate based on object surface type, showing increased errors for glossy/reflective and dark-coloured objects.

D. *Performance Metrics*

The following quantitative metrics were defined to objectively evaluate system performance:

- **Sorting Speed:** Number of objects processed and sorted per minute.
- **False Positives/Negatives:** Count of incorrect sorting decisions or missed detections.
- **Sensor Response Time:** Time elapsed between object detection by the IR sensor and completion of sorting action.
- **System Downtime:** Duration of operational interruptions or failures during continuous operation.
- **Accuracy:** The system achieved an average sorting ac- curacy of approximately 93–95% under standard indoor lighting. The majority of misclassifications occurred with objects exhibiting reflective surfaces or irregular shapes.
- **Throughput:** Sorting speed averaged 12–15 objects per minute, primarily constrained by conveyor motor velocity and servo motor actuation time.
- **Lighting Sensitivity:** Utilization of integrated white LEDs within the sensor housing improved sorting ac- curacy by roughly 5% compared to ambient fluorescent lighting, underscoring the importance of consistent illumination.
- **Robustness:** The system effectively detected and sorted objects ranging from 2 cm to 8 cm in size. However, objects with dark matte finishes occasionally resulted in IR sensor detection failures, leading to missed sorting events.
- **Reliability:** During continuous operation over a 4-hour period, the system exhibited stable performance with negligible sensor drift and no mechanical failures observed.

CONCLUSION

This research successfully demonstrated the design, development, and evaluation of an Arduino-based automated color sorting system, providing a practical and affordable alternative to expensive industrial sorters. By integrating a TCS3200 color sensor, infrared object detection, a servo motor actuator, and a conveyor mechanism controlled via an Arduino Uno microcontroller, the system effectively sorts objects based on their color into designated bins.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue I, January 2026

Experimental results show that the system achieves a sorting accuracy of approximately 93–95% under controlled lighting conditions, with throughput suitable for educational and small- scale industrial applications. The modular design, open-source platform, and cost-effective components make this system accessible for hobbyists, researchers, and small enterprises seeking automation solutions without substantial financial investment.

Despite its promising performance, the system faces challenges including sensitivity to ambient lighting, limitations of low-cost color sensors, and mechanical constraints that limit sorting speed and robustness. Frequent calibration and environmental control are required to maintain accuracy, and the current sorting range is restricted to three primaries.

REFERENCES

1. Johnson, P., & Lee, K. (2015). Machine vision for industrial sorting: Advances and challenges. IEEE Transactions on Automation Science and Engineering, 12(4), 1340–1350. https://doi.org/10.1109/TASE.2015.2441234
2. Patel, S., & Kumar, R. (2017). Optical color sensors for industrial applications. Optical Engineering, 56(6), 066101. https://doi.org/10.1117/1.OE.56.6.066101
3. Zhao, L., Sun, H., & Chen, J. (2020). Adaptive algorithms for color sensor calibration in automated systems. Sensors and Actuators A: Physical, 305, 111962. https://doi.org/10.1016/j.sna.2020.111962
4. Wang, Y., & Li, Z. (2022). Machine learning in automated sorting: Techniques and applications. Journal of Intelligent Manufacturing, 33(7), 2001–2015. https://doi.org/10.1007/s10845-021-01812-7
5. Brown, T., & Green, J. (2018). Arduino-based automation systems in engineering education. Journal of Engineering Education, 107(2), 123–130. https://doi.org/10.1002/jee.2021
6. Alrushidy, I., Alshammari, A., & Alshehri, M. (2025). Design and fabrication of a color sorting machine based on computer vision. Journal of Science and Technology, 30(6), 1–10. https://doi.org/10.20428/jst.v30i6.2954
7. Ong, Z. C. (2024). Towards Industry 4.0: Color-based object sorting using a robotic arm and real-time detection. Industrial Management Advances, 1(1), 1–12. https://doi.org/10.59429/ima.v1i1.125
8. Sanwar, S., & Ahmed, M. I. (2023). Automated object sorting system using real-time image processing and robotic gripper control. Journal of Engineering Advancements, 3(3), 45–53. https://doi.org/10.38032/jea.2023.03.003
9. Hou, L., Zhang, Y., Liu, Q., & Wang, J. (2024). Tomato sorting system based on machine vision technology. Electronics, 13(11), 2114. https://doi.org/10.3390/electronics13112114
10. Aryeni, I., Maulidiah, H. M., & Nugroho, A. (2023). Real-time fruit color and size detection using computer vision for sorting systems. Journal of Applied Electrical Engineering, 7(2), 61–66. https://doi.org/10.30871/jaee.v7i2.6740
11. Umaru, K., Abdullahi, M., & Musa, S. (2025). PLC-based speed control in automated color sorting systems. Journal of Engineering, Technology, and Applied Science, 7(1), 37–51. https://doi.org/10.36079/lamintang.jetas-0701.828
12. Kumar, N., Sharma, P., & Verma, R. (2025). Development of a smart sorting and counting machine for industrial applications. Journal of Advanced Zoology, 44(4), 3270–3278. https://doi.org/10.53555/jaz.v44i4.3270
13. Singh, A., & Mishra, R. (2019). Conveyor-based material handling systems: Design and analysis. International Journal of Mechanical Engineering, 6(2), 85–92. https://doi.org/10.1016/j.ijme.2019.02.004
14. Lee, S., & Park, J. (2020). Vision-based object recognition for industrial automation. International Journal of Advanced Manufacturing Technology, 108(1), 237–248. https://doi.org/10.1007/s00170-020-05422-1
15. Chen, X., & Huang, Y. (2018). Color feature extraction techniques for real-time vision systems. Pattern Recognition Letters, 112, 76–83. https://doi.org/10.1016/j.patrec.2018.07.014
16. Rahman, M., & Islam, T. (2021). Design of low-cost automated sorting systems using embedded controllers. International Journal of Automation Technology, 15(3), 402–410. https://doi.org/10.20965/ijat.2021.p0402

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue I, January 2026

17. Gupta, V., & Bansal, P. (2022). Embedded systems for industrial automation and robotics. Microsystem Technologies, 28(9), 3057–3066. https://doi.org/10.1007/s00542-022-05431-9
18. Kim, H., & Lee, D. (2019). Real-time image processing for object classification in manufacturing. Journal of Manufacturing Systems, 52, 13–21. https://doi.org/10.1016/j.jmsy.2019.04.003
19. Rodrigues, J., & Silva, F. (2020). Automated inspection and sorting using computer vision systems. Measurement, 152, 107317. https://doi.org/10.1016/j.measurement.2019.107317
20. Mehta, R., & Patel, D. (2021). Design and implementation of color-based sorting robots. International Journal of Robotics and Automation, 36(4), 215–223. https://doi.org/10.2316/J.2021.206-0385
21. Liu, Y., & Zhou, X. (2018). Industrial applications of machine vision-based sorting. Sensors, 18(9), 3051. https://doi.org/10.3390/s18093051
22. Verma, S., & Jain, A. (2023). Smart automation systems using vision sensors for manufacturing. Journal of Manufacturing Processes, 86, 124–132. https://doi.org/10.1016/j.jmapro.2023.01.012
23. Torres, M., & Alvarez, J. (2022). Design of intelligent material sorting systems using artificial intelligence. Applied Sciences, 12(14), 7064. https://doi.org/10.3390/app12147064
24. Design and Development of a Radio-Controlled Aircraft and Concept of Electric Vertical Takeoff and Landing (eVTOL). (2025). International Journal of Latest Technology in Engineering Management & Applied Science, 14(10), 386-391. https://doi.org/10.51583/IJLTEMAS.2025.1410000049
25. Development and Evaluation of a Cost-Effective Desktop 3D Printing System for Rapid Prototyping. (2025). International Journal of Latest Technology in Engineering Management & Applied Science, 14(6), 500-508. https://doi.org/10.51583/IJLTEMAS.2025.140600054