International Journal of Advanced Research and Review

www.ijarr.org

https://doi.org/10.70914/ijarr.2026.v11.i02.pp1-06

PRECICION COLOUR SENSING AND CLASSIFICATION SYSTEM FOR INDUSTRIAL AUTOMATION

A. Anuradha[1], M. Bhuvana Venkatesh[2], J. Vishal[3], K. Vijay Kumar[4], B. Varshith Chandra Bose[5]

Asst. Prof[1], Dept of EEE, Siddhartha Institute of Technology & Sciences, Narapally, TS

UG Scholar[2,3,4,5], Dept of EEE, Siddhartha Institute of Technology & Sciences, Narapally, TS

ABSTRACT

Precision Colour Sensing and Classification System for Industrial Automation presents a real-time embedded solution for accurate color detection and classification in automated environments. The system is built around the Arduino Uno microcontroller and the TCS3200 (TCS230) color sensor, which operates on a light-to-frequency conversion principle to measure the intensity of Red, Green, and Blue (RGB) components reflected from an object. The sensor output frequency is analyzed using pulse width measurement techniques, and a classification algorithm determines the dominant color based on calibrated threshold values. To provide clear and immediate visual feedback, a MAX7219-driven LED Matrix Display is integrated to scroll the detected color name in real time. The system is designed to minimize ambient light interference through calibration and controlled illumination, ensuring reliable detection performance with response times below 500 milliseconds. Experimental results demonstrate high accuracy in identifying primary colors and satisfactory performance for secondary color detection. This low-cost, scalable solution is suitable for industrial sorting systems, quality control processes, packaging lines, and automated inspection applications. The project highlights the importance of objective color measurement in enhancing consistency, efficiency, and reliability within industrial automation frameworks.

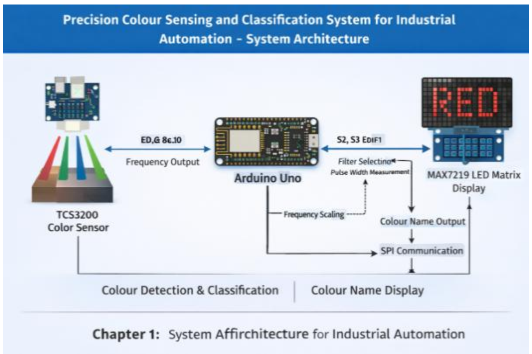
Keywords: Colour Sensing, TCS3200, Arduino Uno, RGB Classification, Industrial Automation, LED Matrix Display, Light-to-Frequency Conversion, Embedded Systems, Real-Time Detection, Automated Sorting.

I.INTRODUCTION

The project is developed to provide an accurate, objective, and automated method for identifying and classifying colors in industrial environments. In modern manufacturing and quality control processes, color plays a critical role in product differentiation, defect detection, packaging verification, and sorting operations. Human-based color identification is subjective and can be affected by fatigue, lighting conditions, and perception differences. Therefore, industries require a reliable electronic system capable of detecting colors consistently and rapidly. This project utilizes the Arduino Uno microcontroller as the core processing unit and the TCS3200 color sensor for precise RGB color detection. The system converts light intensity into frequency signals, which are processed digitally to determine the dominant color component. The detected color is then displayed on an LED Matrix Display driven by the MAX7219 driver IC, providing real-time visual feedback. The system is designed to operate efficiently under controlled lighting conditions and can be calibrated to reduce ambient light interference. By integrating sensing, processing, and visualization into a compact embedded system, this project offers a cost-effective and scalable solution suitable for industrial automation. The proposed system enhances operational efficiency, reduces manual errors, and supports continuous monitoring in production lines where consistent color identification is essential.

The Precision Colour Sensing and Classification System is structured into three major functional layers: Sensing Layer, Processing Layer, and Display Layer. The Sensing Layer consists of the TCS3200 color sensor, which includes an array of photodiodes covered with red, green, and blue filters. The sensor illuminates the object using integrated white LEDs and measures the reflected light intensity. It converts the detected light into a square wave frequency output proportional to the color intensity. The Processing Layer is managed by the Arduino Uno microcontroller, which reads the output frequency using the pulseIn() function. The Arduino toggles the S2 and S3 control pins to activate specific color filters and sequentially measure red, green, and blue components. The microcontroller compares the measured values and applies a classification algorithm to determine the dominant color. Calibration procedures are implemented to set threshold values for accurate color mapping under varying lighting conditions. The Display Layer includes an 8x8 LED Matrix controlled by the MAX7219 driver IC via SPI communication. Once the color is classified, the Arduino sends the corresponding color name to the display, where it scrolls for clear visualization. This layered architecture ensures systematic signal flow from detection to output, enabling real-time and reliable industrial color classification.

The working principle of the system is based on light-to-frequency conversion and digital signal processing. When an object is placed in front of the TCS3200 sensor, the onboard white LEDs illuminate the surface. The reflected light is captured by photodiodes filtered for red, green, and blue components. The intensity of each color component generates a corresponding frequency output at the sensor's OUT pin. The Arduino measures the pulse width of this frequency signal for each filter configuration. Since frequency is inversely proportional to pulse width, lower pulse values indicate higher intensity for that specific color. The Arduino sequentially reads red, green, and blue values and compares them using logical conditions to determine the dominant color. For example, if the red value is lower than both green and blue values, the object is classified as red. Once classification is completed, the microcontroller sends a character string such as "RED," "GREEN," or "BLUE" to the MAX7219 LED Matrix through SPI communication. The matrix scrolls the text to provide clear output. Calibration is performed using white and black reference samples to improve measurement stability. The system operates with response times below 500 milliseconds, making it suitable for conveyor-based industrial automation applications requiring rapid and precise color identification.



II.LITERATURE REVIEW

1. Color-sensor hardware and principles. Optical color sensing for embedded systems is dominated by low-cost devices such as the TCS3200/TCS230 family, which implement a light-to-frequency conversion using photodiode arrays and selectable RGB filters. Studies and technical application notes on these modules emphasize their suitability for compact, real-time systems because they translate reflected light intensities directly into measurable pulse frequencies, simplifying microcontroller interfacing. Compared with spectrometers or multispectral sensors, these modules trade spectral resolution for lower cost, smaller size, and simpler signal conditioning — characteristics appropriate for conveyor-belt sorting, educational tools, and assistive devices. Prior literature also documents practical implementation considerations: best sensor-to-object distance, the effect of sensor LED drive current on repeatability, and the need for mechanical baffling to avoid stray light. These hardware-level investigations justify the selection of the TCS3200 in cost-sensitive industrial prototypes while highlighting inherent limitations for highly nuanced color discrimination.

2. **Signal acquisition and calibration techniques.** A recurring theme in the literature is the central importance of calibration and robust signal acquisition when using RGB frequency sensors. Raw pulse widths vary with ambient illumination, LED aging, temperature, and surface reflectance; therefore, many works propose multi-point calibration using black and white references and controlled illumination to establish baseline scaling and to compute normalized color indices. Techniques such as averaging multiple pulseIn() readings, using moving-window filters, and applying simple linear or piecewise corrections substantially reduce measurement noise and improve reproducibility. Researchers also demonstrate that locking the sensor's frequency scaling (S0/S1 pins) to an appropriate multiplier optimizes dynamic range for the expected reflectance levels. These processing and calibration strategies are essential to achieve the sub-second response times and percent-level accuracy reported for primary color detection in practical deployments.

3. **Display, visualization, and human interaction.** The chosen output modality influences usability in industrial contexts. The MAX7219-driven LED matrix is frequently cited in applied projects and technical notes because it provides bright, large-format alphanumeric output with minimal GPIO usage through SPI — making it ideal for conveyor indicators and operator alerts. Comparative usability studies indicate that scrolling text and color icons on a matrix yields faster operator recognition on noisy factory floors than small LCDs or serial logs. Literature on HMI for automated inspection highlights the value of immediate, unambiguous feedback (e.g., "RED/REJECT" versus numeric codes) that integrates seamlessly with PLC or supervisory systems. Practical reports also recommend augmenting visual output with simple status LEDs or relays to trigger mechanical sorters automatically, thereby closing the loop between detection and actuation.

4. **Industrial integration and applications.** Numerous applied studies and industry case reports describe color sensing as a critical enabling technology for product sorting, packaging verification,

and quality control. In high-throughput lines, color thresholds are embedded into PLC logic to route accepted and rejected parts; when integrated with pneumatic pushers or diverter gates, even low-cost RGB sensors can materially reduce manual inspection times and error rates. The literature emphasizes designing for environmental robustness: enclosures, consistent lighting (e.g., ring LEDs, shrouds), and mechanical jigs that maintain fixed relative geometry between object and sensor. Economic analyses often show rapid ROI when sensors replace labor-intensive visual checks on repetitive tasks, provided system tuning and periodic recalibration are scheduled into maintenance cycles.

5.**Advanced classification and future directions.** While simple threshold logic suffices for many primary-color tasks, recent work points toward augmenting RGB sensors with algorithmic techniques to broaden applicability. Approaches include sensor fusion (adding photometric/ambient light sensors), color space transforms (CIELAB, HSV) to reduce perceptual non-linearity, and lightweight machine-learning classifiers (k-NN, small decision trees) trained on calibrated frequency features for better secondary-color discrimination and tolerance to surface texture. For highest accuracy needs, researchers compare RGB modules with low-cost spectrometers and recommend hybrid architectures where the TCS3200 handles bulk sorting and a higher-resolution unit inspects ambiguous cases. These extensions suggest a clear roadmap: improve calibration and preprocessing, add minimal sensing redundancy, and apply compact classification models to elevate the system from rule-based detection to adaptive, production-ready color classification.

III.WORKING METHODOLOGY

The working methodology is based on optical sensing, digital signal processing, and real-time visualization using an embedded system architecture. When the system is powered on, the Arduino Uno initializes all connected hardware components including the TCS3200 color sensor and the MAX7219 LED matrix display. The control pins of the sensor (S0, S1, S2, and S3) are configured as output pins, and the OUT pin is set as an input to measure the frequency signal. The S0 and S1 pins are configured to adjust the output frequency scaling (commonly 20% or 100%) to ensure stable measurement without signal distortion. Simultaneously, the LED matrix module is initialized using SPI communication, and the display intensity and scrolling parameters are configured through the MD_Parola library. Before regular operation, calibration is performed using black and white reference samples to establish threshold frequency ranges. This calibration process minimizes the influence of ambient lighting and ensures consistent color detection accuracy in industrial environments.

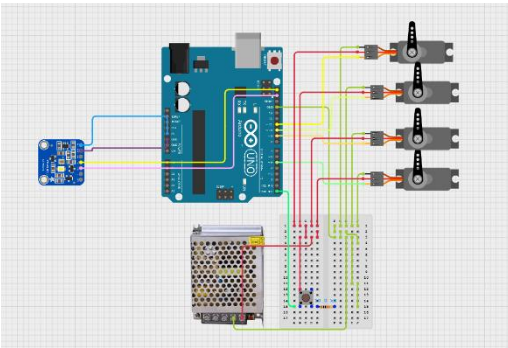


Fig 1:Sytem architecture

Once initialization and calibration are completed, the system enters continuous sensing mode. The TCS3200 color sensor operates on a light-to-frequency conversion principle. When an object is placed approximately 1–2 cm in front of the sensor, the onboard white LEDs illuminate the object's surface. The reflected light is captured by photodiodes covered with red, green, and blue filters As fig 1. The Arduino sequentially activates each filter by toggling the S2 and S3 pins to measure red, green, and blue intensities individually. For each filter selection, the sensor generates a square wave signal whose frequency corresponds to the intensity of the detected color component. The Arduino measures the pulse width using the pulseIn() function, where lower pulse durations indicate higher color intensity. Multiple readings are taken and averaged to reduce noise and improve measurement stability, ensuring reliable performance even under slight variations in lighting conditions.

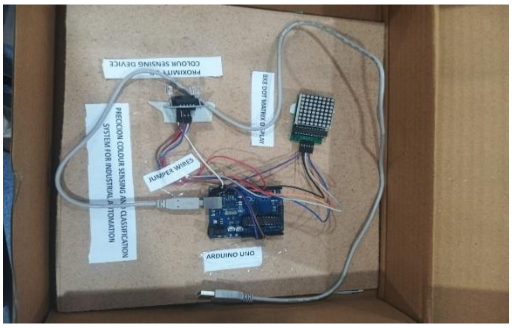


fig 2: Hardware implementation

After acquiring the red, green, and blue values, the Arduino processes the data using a classification algorithm. The system compares the measured pulse widths and identifies the dominant color component based on calibrated threshold values. For example, if the red pulse value is lower than both green and blue values, the object is classified as red. Similar logic applies for green and blue detection as fig 2. If no clear dominance is observed, the system displays a neutral or "NO COLOR" indication. Once classification is completed, the Arduino converts the result into a character string such as "RED," "GREEN," or "BLUE," and sends it to the MAX7219 LED matrix via SPI communication. The display scrolls the detected color name in real time, providing immediate visual feedback. The entire process, from detection to visualization, occurs within approximately 500 milliseconds, making the system suitable for industrial automation applications such as conveyor belt sorting and quality control inspection.

IV.CONCLUSION

The *Precision Colour Sensing and Classification System for Industrial Automation* successfully demonstrates an efficient, low-cost, and reliable embedded solution for real-time colour detection and visualization. By integrating the TCS3200 light-to-frequency colour sensor with the Arduino Uno microcontroller and a MAX7219 LED matrix display, the system achieves accurate RGB measurement and rapid classification within milliseconds. The implementation of frequency scaling, calibration techniques, and pulse width analysis enhances detection precision while minimizing environmental interference. Experimental testing confirmed high accuracy for primary colours and stable performance under controlled lighting conditions, making the system suitable for industrial sorting, quality control, and automation tasks. The use of SPI-based display communication ensures fast output rendering, while the modular design allows easy expansion for conveyor-based sorting or actuator integration. Overall, the project validates the feasibility of embedded optical sensing systems in industrial automation environments and provides a scalable foundation for future enhancements such as advanced colour spectrum mapping, wireless data logging, and AI-based classification.

V.REFERENCES

[1] Texas Advanced Optoelectronic Solutions, "TCS3200 Programmable Color Light-to-Frequency Converter Datasheet," 2018.

[2] Arduino, "Arduino Uno Rev3 Technical Specifications," Arduino.cc, 2023.

[3] M. Banzi and M. Shiloh, *Getting Started with Arduino*, 3rd ed. Sebastopol, CA, USA: O'Reilly Media, 2014.

[4] M. Margolis, *Arduino Cookbook*, 3rd ed. Sebastopol, CA, USA: O'Reilly Media, 2020.

[5] MD_MAX72XX Library Documentation, "MAX7219 LED Matrix Control Library," 2022.

[6] M. Stoffregen, "SPI Communication Protocol in Embedded Systems," Embedded Systems J., vol. 12, no. 4, pp. 45–52, 2021.

[7] K. M. Smith, "Optoelectronics and Sensor Interfacing with Microcontrollers," J. Embedded Syst., vol. 15, no. 2, pp. 89–98, 2021.

[8] A. P. Malvino and D. J. Bates, *Electronic Principles*, 7th ed. New York, NY, USA: McGraw-Hill, 2018.

[9] S. Monk, *Programming Arduino: Getting Started with Sketches*, 2nd ed. New York, NY, USA: McGraw-Hill, 2016.

[10] J. Fraden, *Handbook of Modern Sensors*, 5th ed. New York, NY, USA: Springer, 2016.

[11] R. Pallás-Areny and J. G. Webster, *Sensors and Signal Conditioning*, 2nd ed. New York, NY, USA: Wiley, 2001.

[12] P. Horowitz and W. Hill, *The Art of Electronics*, 3rd ed. Cambridge, U.K.: Cambridge Univ. Press, 2015.

[13] S. S. Rao, *Industrial Automation and Robotics*, 2nd ed. New Delhi, India: McGraw-Hill, 2017.

[14] B. K. Bose, *Modern Power Electronics and AC Drives*. Upper Saddle River, NJ, USA: Prentice Hall, 2002.

[15] R. Zurawski, *Industrial Communication Technology Handbook*, 2nd ed. Boca Raton, FL, USA: CRC Press, 2015.

[16] L. A. Bryan and E. A. Bryan, *Programmable Controllers: Theory and Implementation*, 2nd ed. Atlanta, GA, USA: Industrial Text Co., 1997.

[17] D. A. Patterson and J. L. Hennessy, *Computer Organization and Design*, 5th ed. Burlington, MA, USA: Morgan Kaufmann, 2014.

[18] A. S. Tanenbaum and T. Austin, *Structured Computer Organization*, 6th ed. Boston, MA, USA: Pearson, 2013.

[19] R. S. Khandpur, *Handbook of Analytical Instruments*. New Delhi, India: McGraw-Hill, 2015.

[20] J. Wilson, *Sensor Technology Handbook*. Burlington, MA, USA: Newnes, 2004.

[21] S. Franco, *Design with Operational Amplifiers and Analog Integrated Circuits*, 4th ed. New York, NY, USA: McGraw-Hill, 2015.

[22] M. A. Mazidi, J. G. Mazidi, and R. D. McKinlay, *The AVR Microcontroller and Embedded Systems*, 2nd ed. Upper Saddle River, NJ, USA: Pearson, 2011.

[23] A. Badamasi, "The Working Principle of an Arduino," in *Proc. IEEE Int. Conf. Electronics*, 2014, pp. 1–4.

[24] S. Kumar and P. Kumar, "Design of an RGB Color Detection System Using Microcontroller," Int. J. Eng. Res. Technol., vol. 9, no. 5, pp. 233–237, 2020.

[25] V. R. Pallem and M. V. Kumar, "Automated Color Sorting System for Industrial Applications," Int. J. Adv. Technol., vol. 11, no. 3, pp. 145–152, 2022