

# Speed Measurement of a Moving Object Using Arduino

Jay Bahadur Singh<sup>1</sup>, Golu Pal<sup>2</sup>, Abhay Vishwakarma<sup>2</sup>, Ramniwas Yadav<sup>2</sup>, Akshay Kumar<sup>2,\*</sup>

## Abstract

*This research paper presents the design, construction, and testing of a low-cost and reliable system for measuring the speed of moving objects using an integrated Arduino microcontroller. The project employs infrared (IR) sensors as optical photogates to detect the time taken by an object to travel between two fixed points, allowing the Arduino to calculate the speed using basic timing algorithms. The design provides an affordable and educational alternative to expensive commercial instruments. The system finds applications in laboratory experiments, traffic speed monitoring, robotics, and automation. Experimental results demonstrate that the system achieves accuracy within a  $\pm 3\%$  error margin when properly calibrated. An Arduino board, IR transmitter-receiver sensor pairs, a display interface, and other electronic parts make up the hardware architecture. The software is designed to guarantee accurate timing, data processing, and speed computation in real time. Applications for the suggested device are numerous and include motion analysis systems, robotics, industrial automation, traffic speed monitoring, and physics lab studies. The system delivers dependable performance with accuracy within a  $\pm 3\%$  error margin, according to the findings of experimental testing and calibration carried out under varied operating conditions. The results verify that the suggested design is accurate, affordable, and useful for technical and educational applications.*

**Keywords:** Arduino, automation, IR sensor, microcontroller, photogate, real-time measurement, robotics, speed measurement

## INTRODUCTION

Speed measurement is a fundamental requirement in numerous engineering and scientific domains, including transportation systems, robotics, automation, and experimental physics. Although highly accurate, traditional instruments such as radar guns and Doppler velocimeters are often expensive and require specialized expertise and calibration [1–3].

With the emergence of open-source hardware platforms such as Arduino, it has become possible to design affordable and flexible instrumentation systems for various measurement applications. The proposed system uses two IR sensors placed at a fixed distance from each other. As a moving object passes both sensors sequentially, the Arduino records [4, 5] the timestamps for each event and computes the speed using the following formula:

$$v = d/\Delta t$$

Where,  $v$  is the velocity,  $d$  is the distance between the sensors, and  $(\Delta t)$  represents the time interval between detections.

The system is simple to construct, uses minimal electronic components, and delivers satisfactory accuracy for academic and practical applications [6–9].

## OBJECTIVES

To design and implement a system using Arduino Uno that accurately measures the speed of a moving

### \*Author for Correspondence

Akshay Kumar  
E-mail: [akshay40379@gmail.com](mailto:akshay40379@gmail.com)

<sup>1</sup>Assistant Professor, Department of Electrical Engineering, Bansal Institute of Engineering and Technology, Lucknow, Uttar Pradesh, India

<sup>2</sup>Student, Department of Electrical Engineering, Bansal Institute of Engineering and Technology, Lucknow, Uttar Pradesh, India

Received Date: December 22, 2025

Accepted Date: January 20, 2026

Published Date: February 03, 2026

**Citation:** Jay Bahadur Singh, Golu Pal, Abhay Vishwakarma, Ramniwas Yadav, Akshay Kumar. Speed Measurement of a Moving Object Using Arduino. Recent Trends in Sensor Research & Technology. 2026; 13(1): 1–8p.

object by calculating the time taken to travel a known distance using appropriate sensors [10].

- To study and understand the working principles of Arduino Uno and its interfacing with various sensors (e.g., IR and ultrasonic sensors).
- A setup was designed to measure the time taken by a moving object to pass between two fixed points.
- To calculate the speed of the object using the formula:  $\text{Speed} = \text{Distance}/\text{time}$
- The measured speed was displayed on an output device such as an LCD screen or serial monitor.
- To develop and test the Arduino code that reads the sensor input, calculates the time and speed, and displays the results.

## LITERATURE REVIEW

Speed measurement systems have been developed for diverse applications such as traffic control, robotics, and automation. Researchers have explored various approaches to achieve low-cost, contactless velocity measurements. The measurement of the speed of moving objects is a fundamental requirement in various fields such as transportation systems, robotics, sports, and physics experiments. Traditional speed measurement methods involve high-end radar guns or laser sensors, which are often expensive and complicated. In contrast, microcontroller-based solutions, such as those using Arduino Uno, offer low-cost and flexible alternatives suitable for educational and small-scale applications [11]. Several studies and projects have explored the use of Arduino Uno for speed measurements using different types of sensors. The most common method involves placing two sensors at a known distance. When a moving object crosses the first sensor, the timestamp is recorded. When it crossed the second sensor, another timestamp was recorded. The Arduino then calculates the speed using the time difference and known distance [12].

- A microcontroller-based IR sensor system for vehicle speed detection to enhance road safety [13].
- A contactless conveyor speed measurement technique using photodiode sensors [14].
- The flexibility and cost efficiency of Arduino-based prototyping for sensor applications [15] is well established.
- The Ultrasonic and IR sensor systems demonstrate that IR sensors perform better for short-range detection [16].

## SYSTEM COMPONENTS

- *Arduino Uno*: Acts as the primary controller for data acquisition and computation (Figure 1).
- *Infrared (IR) sensors (two units)*: Function as photogates to detect object motion (Figure 2).
- IR obstacle detection sensor module used for proximity sensing.
- *16×2 LCD*: Displays real-time speed readings (Figure 3).
- *Breadboard and jumper wires*: Facilitate circuit prototyping and connections (Figure 4).
- *Power source*: 5 V DC or USB power supply (Figure 5).
- *Moving object*: A wheeled car used for testing and calibration (Figure 6).



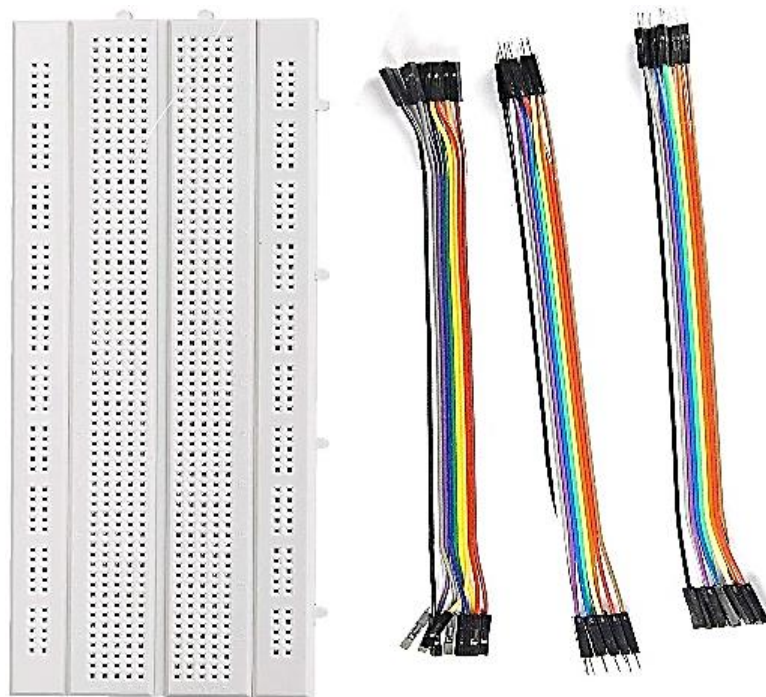
**Figure 1.** Arduino Uno.



**Figure 2.** IR sensors (two units).



**Figure 3.** 16×2 LCD.



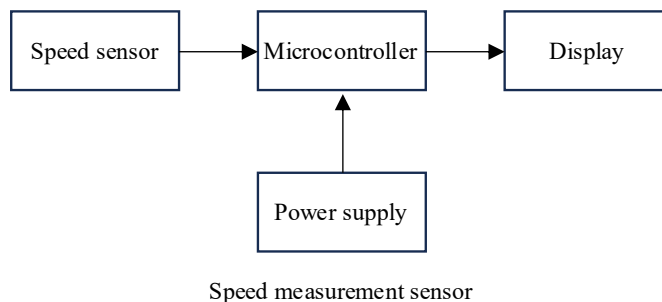
**Figure 4.** Breadboard and jumper wires.



**Figure 5.** Power source.



**Figure 6.** A wheeled car used for testing.



**Figure 7.** Block diagram of the speed measurement system.

### Working Principle

Each IR sensor comprised an infrared LED and a photodiode. When an object passes through the beam, it interrupts the infrared signal, triggering a change in the sensor output (Figures 2–6). When the first sensor detects an object, the Arduino records the first time ( $t_1$ ). When the second sensor is triggered, it records the second time ( $t_2$ ). The Arduino calculates the speed as

$$V = \text{distance/time difference}$$

$$V = D/\Delta t \text{ (from Figure 7).}$$

Where,  $d = 0.1$  m (distance between the sensors). The results were displayed on an LCD screen and sent to a serial monitor for analysis (Figure 2).

### Arduino Program Code

```
#include <LiquidCrystal.h>
```

```
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
```

```
volatile unsigned long t1 = 0, t2 = 0;
volatile bool first = false;
```

```
const float distance = 0.50; // 50 cm = 0.5 m
```

```
void setup() {
  lcd.begin(16, 2);
```

```
Serial.begin(9600);

pinMode(7, INPUT);
pinMode(8, INPUT);
}

void loop() {
  if (digitalRead(7) == LOW && first == false) {
    t1 = micros();
    first = true;

    while (digitalRead(7) == LOW);
  }

  if (digitalRead(8) == LOW && first == true) {
    t2 = micros();

    unsigned long timeDiff = t2 - t1;
    float speed = distance / (timeDiff / 1000000.0);

    lcd.clear();
    lcd.setCursor(0, 0);
    lcd.print("Speed:");

    lcd.setCursor(0, 1);
    lcd.print(speed);
    lcd.print(" m/s");

    Serial.print("Speed (m/s): ");
    Serial.println(speed);

    first = false;
    delay(1000);
  }
}
```

## EXPERIMENTAL SETUP

A small wheeled cart was used to move through two IR sensors placed 0.5 m apart. The Arduino measured the time difference between the sensor triggers to calculate the speed [17–20].

For validation, manual speed measurements using a stopwatch were compared with the Arduino readings. The results showed that the Arduino-based measurements deviated by less than  $\pm 3\%$  from the manual calculations, confirming good accuracy and repeatability. The experimental setup for measuring the speed of a moving object using an Arduino typically uses two sensors (such as infrared or ultrasonic) placed at a fixed distance apart to measure the time it takes for an object to travel between them.

## APPLICATIONS

- Traffic speed detection systems.
- Robotic motion control and feedback systems.
- Educational physics laboratories for kinematic studies.
- Industrial automation and conveyor belt speed monitoring.
- Sports performance analysis (e.g., measuring object or athlete velocity).

- Robotics and automation.
- Speed measurement systems (e.g., moving objects, toy cars).
- Temperature, humidity, and environmental monitoring.
- Internet of Things (IoT) projects and data logging.
- Educational kits for electronics and programming learning.

### ADVANTAGES

- Simple, portable, and low-cost design.
- Real-time and accurate measurements.
- Easy assembly and calibration.
- Suitable for beginners and academic experiments.
- Expandable for automation and IoT applications.
- Easy to interface with microcontrollers (Arduino, PIC, etc.).
- Readable alphanumeric display.
- Backlit display allows usage in low-light conditions.
- Characters remain visible without constant refresh.
- Cost-effective and widely available.

### LIMITATIONS

Limited range due to IR sensor constraints.

- Susceptible to interference from ambient light.
- Requires precise sensor alignment.
- Unsuitable for high-speed or long-range measurements.
- Needs calibration for different object sizes and materials.
- Sensitivity to strong sunlight (affecting IR sensors).
- Reduced precision for very high-speed objects due to sensor response time.
- Ambient light interfering with IR sensors.
- Irregular movement of the object (bouncing or vibration).
- Suitable only for small distances.

### RESULTS

The system demonstrated reliable performance at speeds ranging from 0.2 to 5.0 m/s, maintaining an average accuracy of  $\pm 3\%$ . The accuracy of this method largely depends on the sensor placement, environmental lighting, and timing resolution.

A comparative analysis with manual stopwatch readings confirmed the consistency and repeatability of the results. The project validates Arduino's potential for low-cost, real-time instrumentation and educational applications using IR sensors to measure the time taken by an object to travel a set distance.

### CONCLUSION

In this study, we successfully designed and implemented an Arduino-based speed measurement system using IR sensors. This project proved that such a system is both affordable and accurate for small-scale applications. The experimental results confirmed that the system provided reliable data within a reasonable margin of error. Its simplicity, flexibility, and educational value make it a valuable tool for engineering students, educators, and automation developers alike.

### Feasibility and Effectiveness

The project successfully demonstrated the use of an Arduino microcontroller and sensors (such as IR or ultrasonic) to detect a moving object and calculate its speed in real time.

### Principle Application

Speed is typically calculated by measuring the time taken for an object to travel a fixed, known

distance between two points (using two IR sensors) or by measuring changes in distance over a fixed time interval (using a single ultrasonic sensor).

### **Accuracy and Limitations**

The system's accuracy can be high under ideal conditions (e.g., proper sensor calibration, consistent environment, and appropriate object size/shape) but may decrease owing to factors such as the distance from the sensor, ambient temperature (for ultrasonic), or the object's surface characteristics.

### **Automation and Monitoring**

The system provides an automated solution for continuous monitoring, thereby reducing the need for manual measurements. The calculated speed can be displayed on an LCD screen, and optional alerts (such as a buzzer or LED) can be integrated for over-speeding conditions.

### **Educational Value and Versatility**

This project offers a practical, hands-on introduction to the integration of hardware and software using an embedded system platform. It is a versatile tool with applications ranging from hobbyist projects and scientific experiments to basic traffic-management systems.

### **Future Scope**

The system can be extended with features such as data logging, camera integration to capture number plates, or wireless communication (IoT) for remote monitoring.

- Integration with wireless modules (Bluetooth, Wi-Fi) for remote monitoring.
- Use of laser or ultrasonic sensors for extended measurement range.
- Implementation of data logging and graphical interfaces.
- Deployment of multi-sensor arrays for trajectory tracking.
- Use of machine learning for predictive motion analysis.

### **REFERENCES**

1. Seth D, Satputaley SS, Rehman MAA, Bhende AR, editors. Technological innovations & applications in Industry 4.0. London: CRC Press; 2025. doi:10.1201/9781003567653.
2. Singh T, Mohapatra T, Mohapatra H. Sensor revolution: unveiling the transformative impact on society. In: Pathak D, Mohapatra H, Mishra SR, editors. Role of Emerging Technologies in Social Science. United Kingdom: Cambridge Scholars Publishing; 2024. p. 156-185.
3. Manjari K, Verma M, Singal G. A travel aid for visually impaired: R-Cane. In: Wang G, El Saddik A, Lai X, Martinez Perez G, Choo KK, editors. Smart City and Informatization: 7th International Conference, iSCI 2019, Guangzhou, China, November 12-15, 2019, Proceedings. Communications in Computer and Information Science. Vol. 1122. Singapore: Springer; 2019. doi:10.1007/978-981-15-1301-5\_32.
4. Sharma H, Sharma S, Kumar H, Yadav M, Singh K, Moreira F. Object sensing robot and distance measurement using Arduino UNO. In: Rocha A, Ferrás C, Calvo H, editors. Information Technology and Systems. ICITS 2025. Lecture Notes in Networks and Systems. Vol. 1448. Cham: Springer; 2025. doi:10.1007/978-3-031-93106-2\_4.
5. Shelke D, Pawar P, Tate P, Shinde P, Mali AS. Trends & technologies in obstacle avoidance systems based on Arduino. Int J Adv Res Sci Commun Technol. 2025;5(4):440-455. doi:10.48175/IJAR SCT-29427.
6. Ahmmed F, Rahman A, Islam A, Alaly A, Mehnaj S, Saha P, et al. Arduino-controlled multi-function robot with Bluetooth and nRF24L01+ communication. Int J Robot Control Syst. 2024;4(3):1353-1381. doi:10.31763/ijrcs.v4i3.1517.
7. Korunur Engiz BK, Bashir R. Implementation of a speed control system using Arduino. 2019 6th International Conference on Electrical and Electronics Engineering (ICEEE), Istanbul, Turkey. 2019, p. 294–297. doi:10.1109/ICEEE2019.2019.00063.
8. Ziouzios D, Chatzisavvas A, Fragulis G, Dasygenis M. An advanced robotic system utilizing

- 
- convolutional neural networks for recycling. In: Farmanbar M, Tzamtzi M, Verma AK, Chakravorty A, editors. *Frontiers of Artificial Intelligence, Ethics, and Multidisciplinary Applications*. FAIEMA 2023. Singapore: Springer; 2024. doi:10.1007/978-981-99-9836-4\_14.
9. Shah D, Dave J, Dave D, Soni S, Patel K. Design and development of a quadruped ladder climbing robot using optimized PID parameters using genetic algorithm. *Int J Interact Des Manuf*. 2025;19(3):2159-2181. doi:10.1007/s12008-024-02167-5.
  10. Sesa E, Feriyono A, Djamal M, Musa MDT, Ulum MS, Farhamsa D. The design and implementation of an instrument for converting angular velocity to linear velocity based on Arduino ATmega 2560. *J Phys Conf Ser*. 2020;1434(1):012001. doi:10.1088/1742-6596/1434/1/012001.
  11. Teoh YX, Othmani A, Goh SL, Usman J, Lai KW. Deciphering knee osteoarthritis diagnostic features with explainable artificial intelligence: a systematic review. *IEEE Access*. 2024;12:109080-109108. doi:10.1109/ACCESS.2024.3439096.
  12. Ramu K, Ramachandran M, Selvam M. Microcontroller based sensor interface and its investigation. *Electr Autom Eng*. 2022;1(2):92-97. doi:10.46632/eae/1/2/4.
  13. Khan TA, Shaikh FA, Khan S, Siddiqui MF. Real-time wireless monitoring for three phase motors in industry: a cost-effective solution using IoT. 2019 IEEE International Conference on Smart Instrumentation, Measurement and Application (ICSIMA), Kuala Lumpur, Malaysia. 2019. p.1-5. doi:10.1109/ICSIMA47653.2019.9057343.
  14. Saini J, Kumar U, Gupta AK, Mishra P, Dhairiyasamy R, Varshney D, et al. Geospatial monitoring of periurban zones in Chandigarh through multi-year Landsat imagery. *Urban Ecosyst*. 2025;28(5):189. doi:10.1007/s11252-025-01799-z.
  15. Lyu C, Xiao Y, Deng Y, Chang X, Yang B, Tian J, et al. Tactile recognition technology based on multichannel fiber optical sensing system. *Measurement*. 2023;216:112906. doi:10.1016/j.measurement.2023.112906.
  16. Gupta N, Agarwal AK. Object identification using super sonic sensor: Arduino object radar. 2018 International Conference on System Modeling & Advancement in Research Trends (SMART), Moradabad, India. 2018. p. 92-96. doi:10.1109/SYSMART.2018.8746951.
  17. Latha KA, Mahesh K, Varadarajulu T, Dharani Kumar P, Bharat Kumar C, Sadik K. Smart water quality monitoring system using Arduino based sensors: a practical approach. *Appl Mech Mater*. 2024;924:101. doi:10.4028/p-daIn7n.
  18. Samal T, Panda A, Kabat MR, Awad AI, Jayasingh SK, Tosh DK, editors. *Intelligent Computing Techniques and Applications*. Boca Raton: CRC Press; 2025. doi:10.1201/9781003658221.
  19. Li H, Ao L, Guo H, Yan X. Indoor multi-sensor fusion positioning based on federated filtering. *Measurement*. 2020;154:107506. doi:10.1016/j.measurement.2020.107506.
  20. Gupta A, Singh A, Ahmad N, Singh TP, Sharma S, Sharma P. Experimental techniques to study protein dynamics and conformations. In: Tripathi T, Dubey VK, editors. *Advances in Protein Molecular and Structural Biology Methods*. Amsterdam: Academic Press; 2022. p. 181-197. doi:10.1016/B978-0-323-90264-9.00012-X.
-