IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 9, September 2026

DOI: 10.17148/IJIREEICE.2026.14902

IOT Based Hand-Gesture Controlled LED Blinking Effect

Siddharth S. Birajdar[1], Vaibhav K. Deshmukh[1], Gayatri P. Chandane[1],

S. T. Sanamdikar[2]

Student, Department of Electronics and Telecommunications Engineering,

Pune District Education Association's College of Engineering, Manjari (Bk.) Pune, India[1]

Associate Prof., Department of Electronics and Telecommunications Engineering,

Pune District Education Association's College of Engineering, Manjari (Bk.) Pune, India[2]

Abstract: The fast growth of human-machine interaction has created a need for simple, easy-to-use, and contactless ways to control electronic devices.

Traditional control systems usually rely on physical switches, buttons, or wired connections, which might not always be convenient or flexible. This research introduces a low-cost Hand-Gesture Controlled LED Blinking Effect system that lets users control LED lighting and blinking patterns using specific hand gestures. The system uses an Arduino Uno microcontroller, HC-05 Bluetooth module, LED lights, resistors, and a breadboard to offer wireless control of the LED circuit. In this approach, hand gestures are recognized as set control commands and sent wirelessly via the HC-05 Bluetooth module.

The Arduino Uno receives and processes these commands based on the programmed logic. Depending on the received command, the microcontroller controls the connected LEDs and creates the appropriate blinking effect. Various commands can be linked to different LED actions, such as turning the LED on or off, adjusting the blinking speed, or forming blinking patterns. This method removes the need for direct physical contact with the LED control setup and provides an easy way to show gesture-based device control.

The system emphasizes simplicity, affordability, ease of use, and dependable wireless communication.

The experimental setup shows that an Arduino-based platform can effectively handle Bluetooth-based gesture commands and control LED outputs with minimal hardware. The proposed system serves as a fundamental model for understanding gesture-based human-machine interaction, embedded systems, and wireless control technologies. Also, the idea can be expanded to control multiple LEDs and other electronic devices by adding more control commands and appropriate hardware. Possible uses include smart lighting systems, home automation, assistive technology, interactive electronic devices, educational projects, and touchless control systems. This work highlights the potential of combining gesture-based interaction with affordable microcontrollers and Bluetooth technology to build accessible and scalable electronic control systems.

Keywords: Hand Gesture Recognition, Arduino Uno, HC-05 Bluetooth Module, LED Control, Wireless Communication, Human-Machine Interaction, Embedded System, Gesture-Based Control.

I. INTRODUCTION

The fast development of embedded systems, wireless communication, and human-machine interaction has led to the creation of new ways to control electronic devices in a more convenient and intuitive manner. Traditional control systems usually depend on physical switches, push buttons, or cables. Although these methods are straightforward and dependable, they require direct contact with the control device. As the need for smart and contactless systems increases, gesture-based control has become a popular alternative that allows users to interact with electronic devices through specific hand movements or commands. Hand gestures are a natural way for humans to communicate and can be used as an input method for electronic systems. A gesture-controlled system takes a specific hand movement and interprets it as a command, then performs an action based on that command. These systems can make user interaction more convenient and reduce reliance on traditional physical interfaces. Gesture-based interaction has many applications, including smart home automation, robotics, assistive technologies, interactive screens, consumer electronics, and industrial control systems. In this research, a Hand-Gesture Controlled LED Blinking Effect is developed using an Arduino Uno, HC-05 Bluetooth module, LED lights, resistors, and a breadboard. The Arduino Uno acts as the main processor of the system and is responsible for receiving control commands and managing the connected LEDs. The HC-05 Bluetooth module enables wireless communication between the gesture-input interface and the Arduino. When a specific hand gesture is detected, it is converted into a command and sent through the Bluetooth connection. The Arduino receives the command,

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 9, September 2026

DOI: 10.17148/IJIREEICE.2026.14902

follows the programmed instructions, and applies the corresponding LED output. The LED circuit functions as the output part of the system. Different commands can be linked to different LED actions, allowing the system to produce various blinking effects. For example, one command could turn an LED on, another could turn it off, and other commands might create different blinking patterns or speeds. This provides a simple example of how human gestures can be turned into wireless control signals and then into physical outputs using a microcontroller. Using the Arduino Uno makes the system relatively easy and affordable to build. It offers digital input and output pins, programming support, and a user-friendly development environment. The HC-05 Bluetooth module also enhances the system by allowing wireless control without needing a physical connection between the input device and the LED circuit.[4]

Together, these components make the system suitable for educational and experimental use in areas like embedded systems, Bluetooth communication, and human-machine interaction. The main goal of this research is to show a practical and cost-effective way to control LED blinking using hand gestures. The work also explores how wireless command transmission and microcontroller-based processing can be used to implement gesture-controlled outputs. The proposed system serves as a foundation that can be expanded to control multiple LEDs, appliances, motors, or other devices by increasing the number of commands and adjusting the control logic. The importance of this work is in showing how simple and accessible electronic components can be combined to create a contactless control system. Even though the current implementation is focused on LED blinking, the core idea can be applied to other areas such as smart lighting, home automation, assistive devices, interactive systems, robotics, and other human-machine interfaces. Therefore, the proposed Hand-Gesture Controlled LED Blinking Effect system offers a basic yet effective example of wireless gesture control using an Arduino Uno and Bluetooth technology.[5]

II. WORKING COMPONENTS

The main components of the Hand-Gesture Controlled LED Blinking Effect are:

a) Arduino-Uno
b) HC-05 Bluetooth Module
c) LED Lights
d) Resistors
e) Breadboard

The proposed Hand-Gesture Controlled LED Blinking Effect system is implemented using a simple and low-cost electronic setup. The major components used are LED lights, Arduino Uno, resistors, breadboard, and HC-05 Bluetooth module. Each component performs a specific function in receiving commands, processing them, and producing the desired LED blinking output.[7]

2.1 Arduino Uno

Figure No 1. Arduino Uno

The Arduino Uno is the main microcontroller platform used in the proposed system. It is based on the ATmega328P microcontroller and provides digital and analog input/output pins that can be used to interface with external electronic components. In this project, the Arduino Uno acts as the central control unit. It receives commands from the HC-05 Bluetooth module through its serial communication pins. The received command is then processed according to the program stored in the microcontroller. Based on the command, the Arduino generates appropriate signals at its digital output pins to control the LEDs.

The Arduino Uno is suitable for this application because of its simple programming environment, sufficient digital I/O pins, low cost, and compatibility with the HC-05 Bluetooth module.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 9, September 2026

DOI: 10.17148/IJIREEICE.2026.14902

Important features:

- Microcontroller: ATmega328P
- Operating voltage: 5 V
- Digital I/O pins: 14
- Analog input pins: 6
- PWM-capable digital pins: 6
- USB interface for programming and serial communication
- Simple programming using Arduino IDE

2.2 HC-05 Bluetooth Module

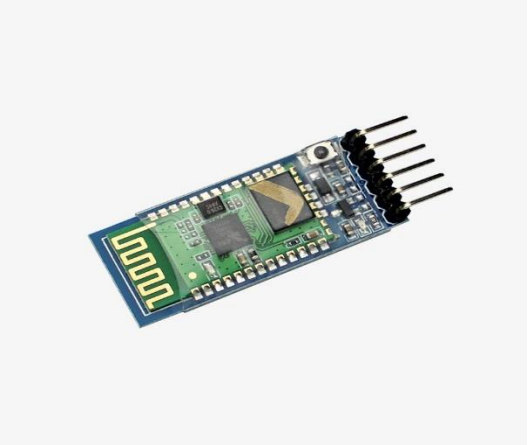


Figure No 2. HC-05 Bluetooth Module

The HC-05 Bluetooth module provides wireless serial communication between the command-generating device and the Arduino Uno. It is commonly used in embedded-system projects where data needs to be transmitted wirelessly over a short distance. The module communicates with the Arduino using UART (Universal Asynchronous Receiver/Transmitter) serial communication. Commands generated from the gesture-control interface are transmitted wirelessly to the HC-05. The module converts the received Bluetooth data into serial data that can be read by the Arduino. The HC-05 generally operates using a 3.3 V logic level on its serial interface, although its breakout board can commonly be powered from 5 V depending on the specific board design. Care should be taken with the Arduino-to-HC-05 TX connection because the HC-05 RX input should be protected from a 5 V logic signal, typically using a voltage divider. [3]

Typical pins include:

Table no 1: Functions of HC-05 Bluetooth Module

Pin	Function
VCC	Power supply
GND	Ground
TXD	Transmits serial data
RXD	Receives serial data
EN/KEY	Configuration/AT-mode control

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 9, September 2026

DOI: 10.17148/IJIREEICE.2026.14902

2.3 LED Lights

Figure No 3. LED light

A Light Emitting Diode (LED) is a semiconductor device that produces visible light when current flows through it in the forward direction. In the proposed system, LEDs act as the output devices and visually indicate the command received by the Arduino.

The Arduino controls the LED by applying a HIGH or LOW signal to the corresponding digital output pin.

- HIGH signal: LED turns ON.
- LOW signal: LED turns OFF.
- Rapid alternation between HIGH and LOW: LED produces a blinking effect.

By changing the time interval between the HIGH and LOW states, different blinking patterns can be produced.[6]

2.4 Resistors

Figure No 4. Resistor

Resistors are passive electronic components used to limit the flow of electrical current in the circuit. In this project, resistors are connected in series with the LEDs.

An LED should not normally be connected directly between an Arduino output pin and ground because excessive current can damage the LED and potentially affect the microcontroller pin. A suitable current-limiting resistor is therefore used. The resistance required can be estimated using Ohm's law:

$$R = \frac{V_{OUT} - V_F}{I}$$

where:

- R = required resistance
- V_{OUT} = Arduino output voltage
- V_F = forward voltage of the LED
- I = desired LED current

For a typical 5 V Arduino output and a standard LED, a resistor such as 220 Ω or 330 Ω is commonly used, depending on the LED characteristics and desired brightness.[9]

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 9, September 2026

DOI: 10.17148/IJIREEICE.2026.14902

2.5 Breadboard

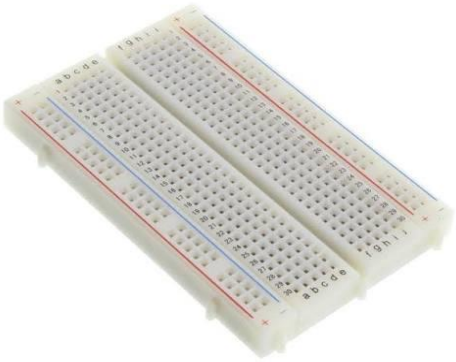


Figure No 5. Breadboard

A breadboard is a reusable prototyping board used to construct electronic circuits without soldering. It contains interconnected conductive contacts that allow components and jumper wires to be connected easily.[12]
In this project, the breadboard is used to connect:

- Arduino Uno
- HC-05 Bluetooth module
- LEDs
- Current-limiting resistors
- Power and ground connections

The breadboard makes it easy to modify the circuit during testing and debugging. Components can be inserted and removed without permanently altering them.

III. WORKING PRINCIPLE

The proposed Hand-Gesture Controlled LED Blinking Effect system operates on the principle of converting human hand gestures into digital control commands and using wireless Bluetooth communication to control an LED circuit. The system is designed to provide a simple, intuitive, and contactless method of controlling LED illumination without requiring the user to operate a conventional physical switch. When the user performs a predefined hand gesture, the gesture recognition mechanism identifies the gesture and associates it with a specific control command. Each gesture is assigned a unique command so that different hand movements can be used to perform different operations, such as switching an LED ON or OFF or producing a particular blinking pattern. This approach establishes a direct relationship between human interaction and the operation of an electronic device.

After the hand gesture is recognized, the corresponding command is generated in the form of digital data. This command is transmitted wirelessly using the Bluetooth communication system. The HC-05 Bluetooth module serves as the communication interface for transferring the generated command to the Arduino Uno. Bluetooth communication eliminates the need for a physical wired connection between the command-generating device and the microcontroller, thereby improving the flexibility and convenience of the system. The HC-05 receives the transmitted data and communicates it to the Arduino Uno through its serial communication interface. The received information is therefore converted into a form that can be interpreted and processed by the microcontroller.

The Arduino Uno acts as the central processing and control unit of the proposed system. A predefined control program is stored in the Arduino, containing instructions corresponding to the different commands that can be received from the Bluetooth module. When a command arrives, the Arduino continuously monitors the serial communication interface and identifies the received data. The microcontroller compares the received command with the predefined instructions in the program and determines the appropriate operation to be performed. For example, a particular command may instruct the Arduino to switch the LED ON, another command may switch it OFF, while other commands may activate different blinking patterns. This command-processing mechanism allows the system to respond according to the user's selected hand gesture.[15]

Once the command has been interpreted, the Arduino generates a corresponding digital signal through one of its digital output pins. The output signal is used to control the connected LED circuit. When the digital output is set to a HIGH state, electrical current flows through the LED and causes it to emit light. When the output is set to a LOW state, the current flow is stopped and the LED turns OFF. A resistor is connected in series with the LED to limit the current flowing through the device. This current-limiting resistor protects the LED from excessive current and helps protect the Arduino's

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 9, September 2026

DOI: 10.17148/IJIREEICE.2026.14902

digital output pin. Thus, the Arduino can safely control the LED according to the command received through the Bluetooth communication link.[18]

The blinking effect is generated by repeatedly switching the Arduino's digital output between HIGH and LOW states at a predetermined time interval. When the output alternates between these two states, the LED repeatedly turns ON and OFF, producing a visible blinking effect. The speed and pattern of blinking depend on the timing parameters programmed into the Arduino. A longer time interval between the ON and OFF states produces a slower blinking effect, whereas a shorter interval produces faster blinking. By modifying the timing sequence, different visual effects can be generated. Multiple LEDs can also be controlled independently or simultaneously to produce more complex blinking patterns, provided that sufficient Arduino output pins and appropriate current-limiting resistors are used.

The breadboard is used to establish the electrical connections between the Arduino Uno, HC-05 Bluetooth module, resistors, and LEDs during the development and testing of the system. It provides a convenient platform for assembling and modifying the circuit without soldering. The Arduino supplies the required control signals to the LED circuit, while the HC-05 provides the wireless communication interface. Together, these components form a simple embedded control system in which the input is represented by a hand gesture, the communication medium is Bluetooth, the processing unit is the Arduino Uno, and the output is the LED blinking effect.[8]

The complete principle of operation can therefore be described as a sequence of sensing, command generation, wireless transmission, command processing, and output control. The user's hand gesture is first recognized and converted into a digital command. The command is then transmitted wirelessly through Bluetooth and received by the HC-05 module. The HC-05 forwards the command to the Arduino Uno through serial communication, where it is processed according to the programmed logic. The Arduino subsequently generates the required digital output signal, which controls the LED through a current-limiting resistor. Depending on the received command, the LED can remain continuously ON, remain OFF, or repeatedly switch between the ON and OFF states to create the desired blinking pattern.

The proposed principle demonstrates the integration of human-machine interaction with embedded control and wireless communication technologies. Although the present system is focused on controlling LED blinking effects, the same operating principle can be extended to other electronic applications. By modifying the Arduino program and replacing or adding suitable output devices, the system can potentially be used for smart lighting, home automation, interactive electronic systems, assistive technologies, robotics, and other wireless control applications.

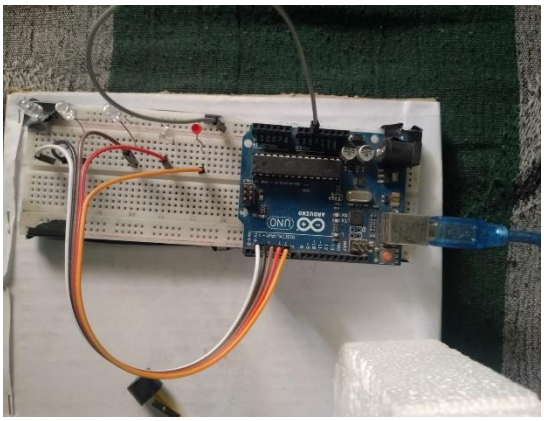


Figure no 6. Main Circuit

IV. WORIKING OF HAND GESTURE CONTROLLED LED BLINKING EFFECT

The proposed Hand-Gesture Controlled LED Blinking Effect system operates by converting a user's predefined hand gesture into a control command and transmitting that command wirelessly to an Arduino Uno through an HC-05 Bluetooth module. The Arduino processes the received command and controls one or more LEDs to produce the required blinking effect. The overall principle combines gesture-based human-machine interaction, Bluetooth wireless communication, and microcontroller-based LED control.[10]

The basic operation can be represented as:
Hand Gesture → Gesture Recognition → Command Generation → Bluetooth Transmission → HC-05 → Arduino Uno → Command Processing → LED Control → Blinking Effect

3.1 Gesture Detection and Command Generation

The operation begins when the user performs a predefined hand gesture. Each gesture is assigned a particular command

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 9, September 2026

DOI: 10.17148/IJIREEICE.2026.14902

corresponding to an LED operation. For example, one gesture may represent LED ON, another may represent LED OFF, while other gestures may be assigned to different blinking patterns.[11]
The gesture recognition section identifies the performed gesture and converts it into a digital command. Instead of transmitting the physical movement of the hand directly, the system transmits a corresponding character or command through the Bluetooth communication link.
For example:

Table no 1: Gesture Commands

Hand Gesture	Generated Command	Expected Output
Gesture 1	A	LED ON
Gesture 2	B	LED OFF
Gesture 3	C	Slow blinking
Gesture 4	D	Fast blinking

3.2 Bluetooth Communication
Once the gesture has been converted into a command, the command is transmitted wirelessly using Bluetooth. The HC-05 Bluetooth module acts as the wireless communication interface between the gesture-control device and the Arduino Uno. The HC-05 uses serial communication to exchange data with the Arduino. When a Bluetooth connection is established, the command generated by the input device is transmitted to the HC-05.
For example, if the user performs the gesture associated with the command C, the character C is transmitted through the Bluetooth connection.[1]
The wireless communication can therefore be represented as:
Gesture → C → Bluetooth → HC-05 → Arduino

3.3 Receiving the Command
The HC-05 receives the transmitted Bluetooth data and makes it available to the Arduino through its UART serial interface. The Arduino continuously monitors its serial communication channel. When data is received, the microcontroller reads the incoming character and stores it temporarily for processing.[2]
For example:
Received data = "C"
The Arduino then compares the received data with the commands defined in its program.

3.4 Command Processing by Arduino Uno
The Arduino Uno acts as the central processing unit of the system. A control program is uploaded to the Arduino using the Arduino IDE.
The program contains predefined conditions corresponding to different commands. When a command is received, the Arduino determines which operation should be performed.

Conceptually, the program operates as follows: Flow Chart

Receive command
↓
Identify command
↓
Is command = A?
↓ ↓
Yes No

3.5 LED Control
The LEDs are connected to the digital output pins of the Arduino through current-limiting resistors.
The Arduino controls the LEDs by changing the voltage level at its digital output pins.
When the output pin is set to HIGH:

$$V_{OUT} \approx 5V$$

and current flows through the resistor and LED, causing the LED to illuminate.
When the output pin is set to LOW:

$$V_{OUT} \approx 0V$$

and the LED turns OFF.
Therefore:
HIGH → LED ON

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 9, September 2026

DOI: 10.17148/IJIREEICE.2026.14902

LOW → LED OFF
The resistor connected in series with the LED limits the current and protects the LED and Arduino output pin from excessive current.[13]

3.6 Generation of Blinking Effect
The blinking effect is produced by repeatedly changing the LED output between HIGH and LOW states.
For example, a slow blinking sequence can be represented as:
HIGH → LOW → HIGH → LOW → HIGH → LOW
ON OFF ON OFF ON OFF
The duration for which the LED remains ON or OFF determines the blinking frequency.
For example, if the Arduino uses:

$$T_{ON} = 500ms$$

and

$$T_{OFF} = 500ms$$

then the LED completes one complete ON-OFF cycle every:

$$T = T_{ON} + T_{OFF}$$

$$T = 500 + 500 = 1000ms$$

Therefore:

$$T = 1s$$

and the approximate blinking frequency is:

$$f = \frac{1}{T} = 1Hz$$

Changing these time intervals allows different blinking speeds to be produced.

3.7 Execution of Different Blinking Patterns
The system can be programmed to provide different LED effects depending on the received gesture command. For example:

Slow blinking:
ON ———— OFF ———— ON ———— OFF

Fast blinking:
ON ─ OFF ─ ON ─ OFF ─ ON ─ OFF

Multiple LED pattern:
LED 1: ON → OFF → ON → OFF
LED 2: OFF → ON → OFF → ON
Thus, the gesture does not directly control the LED. Instead, it selects a particular control instruction, and the Arduino executes the corresponding LED operation.

3.8 Complete Working Sequence
The complete working of the proposed system can be explained in the following sequence:
Step 1: The user performs a predefined hand gesture.
Step 2: The gesture recognition mechanism identifies the gesture.
Step 3: The recognized gesture is converted into a predefined digital command.
Step 4: The command is transmitted wirelessly through Bluetooth.
Step 5: The HC-05 Bluetooth module receives the transmitted command.
Step 6: The HC-05 transfers the received command to the Arduino Uno through serial communication.
Step 7: The Arduino reads and identifies the received command.
Step 8: The microcontroller executes the corresponding instruction stored in its program.
Step 9: The Arduino generates an appropriate digital output signal.
Step 10: The signal passes through the current-limiting resistor and controls the connected LED.
Step 11: Depending on the command, the LED remains ON/OFF or repeatedly switches between ON and OFF states to generate the desired blinking effect.
Step 12: The process continues, allowing the user to perform another gesture and change the LED operation.[14]

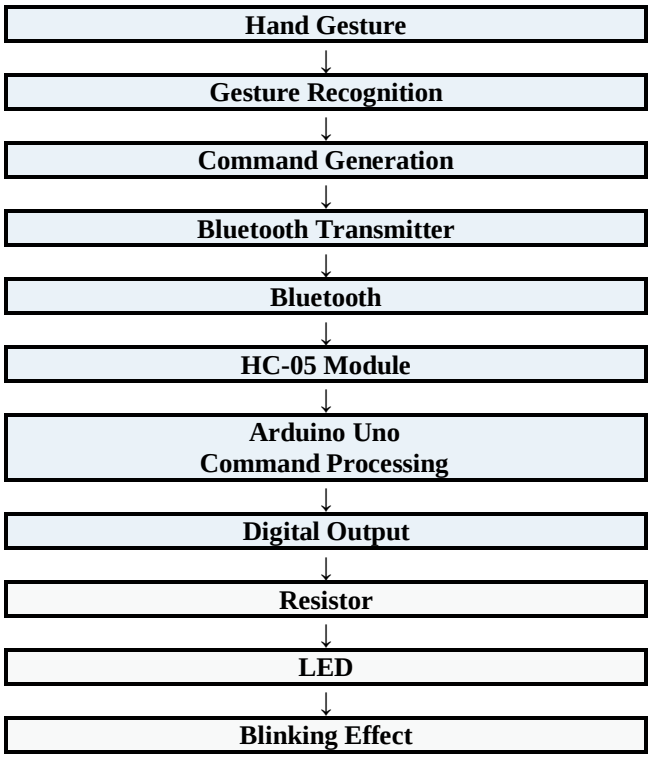
IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 9, September 2026

DOI: 10.17148/IJIREEICE.2026.14902

3.9 Overall Working Diagram:



V. RESULTS

The proposed Hand-Gesture Controlled LED Blinking Effect system was successfully implemented using an Arduino Uno, HC-05 Bluetooth module, LED lights, resistors, and a breadboard. The experimental setup demonstrated that hand-gesture-based commands could be converted into control instructions and transmitted wirelessly to the Arduino through the Bluetooth communication interface. After receiving the commands, the Arduino correctly processed the corresponding instructions and controlled the connected LEDs according to the programmed logic.

During the experimental operation, different predefined commands were used to control the LED output. The system was able to perform basic operations such as switching the LED ON and OFF and generating programmed blinking effects. The blinking operation was achieved by repeatedly changing the digital output of the Arduino between HIGH and LOW states for predefined time intervals. The change in the timing intervals produced different blinking speeds, demonstrating that the blinking behaviour could be controlled through software without requiring changes to the physical circuit.

The HC-05 Bluetooth module provided the wireless communication link between the command-generating device and the Arduino Uno. The received commands were transferred to the Arduino through serial communication and were interpreted according to the programmed instructions. The successful execution of the corresponding LED operations confirmed that the Bluetooth module and Arduino could work together effectively for wireless control. The system also demonstrated that the use of wireless communication can reduce the requirement for physical interaction with the electronic circuit.[20]

The resistors connected in series with the LEDs provided current limitation and allowed the LEDs to operate safely during testing. The breadboard-based implementation provided a convenient platform for assembling and modifying the circuit during experimentation. The overall hardware configuration was simple and required relatively few components, demonstrating the feasibility of implementing the proposed system using inexpensive and readily available hardware.

The experimental results indicate that the proposed system can provide a responsive and convenient method of controlling LED outputs through predefined gesture commands. The system successfully demonstrates the basic concept of gesture-based human-machine interaction combined with Bluetooth wireless communication and microcontroller-based control. The observed LED responses corresponded to the commands programmed into the Arduino, confirming the proper operation of the control logic.

IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁜ Peer-reviewed & Refereed journal ⁜ Vol. 14, Issue 9, September 2026

DOI: 10.17148/IJIREEICE.2026.14902

The results can be summarized as follows:

Parameter	Observed Result
Gesture command recognition	Successfully converted into predefined commands
Wireless communication	Successfully transmitted through HC-05
Arduino command processing	Successfully executed programmed instructions
LED ON/OFF control	Successfully achieved
LED blinking effect	Successfully generated
Blinking speed variation	Achieved by modifying programmed timing
Circuit implementation	Successfully assembled on breadboard
System operation	Successfully demonstrated

Overall, the experimental implementation confirms that the proposed system can control LED blinking effects using wireless gesture-based commands. The results demonstrate the potential of the developed approach as a basic platform for contactless electronic control. The system can be further improved by incorporating additional gestures, multiple LEDs, variable blinking patterns, and other output devices to increase its functionality and practical applications.[16]

VI. CONCLUSION

The proposed Hand-Gesture Controlled LED Blinking Effect system successfully demonstrates the implementation of a simple, low-cost, and contactless method for controlling LED illumination using hand gestures. The system combines an Arduino Uno, HC-05 Bluetooth module, LEDs, current-limiting resistors, and a breadboard to establish communication between the user and the electronic output. By converting predefined hand gestures into digital control commands and transmitting these commands wirelessly through Bluetooth, the system eliminates the need for conventional physical switches and provides a more intuitive method of interaction.[17]

The Arduino Uno serves as the central processing unit, receiving the commands through the HC-05 Bluetooth module and executing the corresponding instructions stored in its program. Based on the received command, the Arduino controls the connected LEDs by generating appropriate digital output signals. The LEDs can be switched ON or OFF and can also produce different blinking effects by varying the timing of the HIGH and LOW output states. The use of resistors ensures safe operation by limiting the current flowing through the LEDs, while the breadboard provides a simple and flexible platform for circuit assembly and testing.[19]

The developed system demonstrates the practical integration of gesture-based interaction, Bluetooth wireless communication, and microcontroller-based control using readily available electronic components. Its simple architecture, low cost, ease of implementation, and wireless operation make it suitable for educational and experimental applications. The system also demonstrates that gesture-based commands can be effectively used to control electronic outputs without requiring direct physical contact with the control circuit.

Although the present research focuses on LED blinking, the proposed concept has significant potential for further development. Additional gestures and control commands can be incorporated to operate multiple LEDs with more complex patterns. The same control principle can also be extended to devices such as relays, motors, displays, and other electronic appliances, making it applicable to smart lighting, home automation, assistive technology, robotics, interactive systems, and other human-machine interaction applications.

ACKNOWLEDGMENT

We would like to express our sincere gratitude to everyone who contributed to the successful completion of this research work entitled “Hand-Gesture Controlled LED Blinking Effect.” This work would not have been possible without the valuable guidance, support, and encouragement received throughout the development and documentation of the project.

We would like to express our heartfelt gratitude to our Head of Department and faculty members for providing us with the necessary guidance, technical knowledge, laboratory facilities, and resources required to carry out this research successfully. Their valuable suggestions and continuous encouragement helped us understand the practical aspects of embedded systems, Bluetooth communication, and microcontroller-based control. We would also like to thank our institution and laboratory staff for providing the required facilities and equipment for assembling and testing the proposed system. The availability of components such as the Arduino Uno, HC-05 Bluetooth module, LEDs, resistors, and breadboard made it possible to successfully implement and evaluate the proposed system.

DOI: 10.17148/IJIREEICE.2026.14902

Finally, we express our sincere thanks to our friends, classmates, and family members for their encouragement, cooperation, and support throughout the completion of this work. Their motivation and assistance played an important role in successfully completing this research project.

REFERENCES

[1].M. Oudah, A. Al-Naji, and J. Chahl, "Hand Gesture Recognition Based on Computer Vision: A Review of Techniques," *Journal of Imaging*, vol. 6, no. 8, p. 73, 2020. doi: **10.3390/jimaging6080073**. DOI

[2].P. K. Pisharady and M. Saerbeck, "Recent methods and databases in vision-based hand gesture recognition: A review," *Computer Vision and Image Understanding*, vol. 141, pp. 152–165, 2015.doi:**10.1016/j.cviu.2015.08.004**.Experts@Minnesota

[3].K. Gao, H. Zhang, X. Liu, X. Wang, L. Xie, B. Ji, Y. Yan, and E. Yin, "Challenges and solutions for vision-based hand gesture interpretation: A review," *Computer Vision and Image Understanding*, vol. 248, p. 104095, 2024.doi: **10.1016/j.cviu.2024.104095**. ScienceDirect

[4]."A Systematic Review of Hand Gesture Recognition: An Update From 2018 to 2024," *IEEE Access*, 2024.doi:**10.1109/ACCESS.2024.3421992**. DOI

[5].J. Shin, A. S. M. Miah, M. H. Kabir, M. A. Rahim, and A. Al Shiam, "A Methodological and Structural Review of Hand Gesture Recognition Across Diverse Data Modalities," 2024. arXiv

[6].K. Foteinos, J. Cani, M. Linardakis, P. Radoglou-Grammatikis, V. Argyriou, P. Sarigiannidis, I. Varlamis, and G. T. Papadopoulos, "Visual Hand Gesture Recognition with Deep Learning: A Comprehensive Review of Methods, Datasets, Challenges and Future Research Directions," 2025. arXiv

[7].E. Indasyah and K. Yoshida, "Vision-language models meet hand gesture recognition: Towards multimodal and intelligent human-computer interaction," *The Visual Computer*, vol. 42, art. 377, 2026. doi: **10.1007/s00371-026-04518-7**. Springer

[8].S. Wang, J. Laforet, I. Rida, B. Druart, A. Becheler, D. Istrate, and S. Boudaoud, "A comprehensive survey of multi-modal manipulative hand gesture recognition: Sensors, datasets, and models," *Journal of Visual Communication and Image Representation*, vol. 119, p. 104840, 2026. doi: **10.1016/j.jvcir.2026.104840**. ScienceDirect

[9].R. A. Bolt, "Put-That-There: Voice and Gesture at the Graphics Interface," *ACM SIGGRAPH Computer Graphics*, vol. 14, no. 3, pp. 262–270, 1980.

[10].J. Rekimoto, "GestureWrist and GesturePad: Gesture-based interaction techniques," in *Proceedings of the 5th International Conference on Human-Computer Interaction*, 1993.

[11].V. I. Pavlovic, R. Sharma, and T. S. Huang, "Visual interpretation of hand gestures for human-computer interaction: A review," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 677–695, 1997.

[12].R. Cipolla and A. Pentland, Eds., *Computer Vision for Human-Machine Interaction*. Cambridge, U.K.: Cambridge University Press, 1998.

[13].S. Mitra and T. Acharya, "Gesture recognition: A survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, vol. 37, no. 3, pp. 311–324, 2007.

[14].A. D. Wilson and A. F. Bobick, "Parametric hidden Markov models for gesture recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 9, pp. 884–900, 1999.

[15].T. Starner and A. Pentland, "Real-time American Sign Language recognition from video using hidden Markov models," in *Proceedings of IEEE International Conference on Computer Vision*, 1995, pp. 265–270.

[16].S. S. Rautaray and A. Agrawal, "Vision based hand gesture recognition for human computer interaction: A survey," *Artificial Intelligence Review*, vol. 43, pp. 1–54, 2015.

[17].A. Erol, G. Bebis, M. Nicolescu, R. D. Boyle, and X. Twombly, "Vision-based hand pose estimation: A review," *Computer Vision and Image Understanding*, vol. 108, no. 1–2, pp. 52–73, 2007.

[18].S. S. Rautaray and A. Agrawal, "A vision based tracking and recognition of hand gestures using artificial neural network," in *Proceedings of the International Conference on Image Information Processing*, 2011.

[19].M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp.71–86, 1991.

[20].R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 4th ed. Pearson, 2018.-131, 2021.

