Innovative Journal for Opportunities and Development in Science & Technology,
Volume 6, Issue 3, Feb. 2026
Paper Id: – IJ2025V6I3 -3 A-F
ISSN 2582-6026

AUTOMATIC IGNITION STARTER

[1]Rushikesh kudale, [2]Aditya Jadhav, [3]Samarth Chavan, [4]Omkar Avhad, [5]Pranav Maher
[6]Sagar Gaikwad

[1]UG Scholar, Electrical Dept. JSPM's Bhivrabai Sawant polytechnic, Wagholi
[2]UG Scholar, Electrical Dept. JSPM's Bhivrabai Sawant polytechnic, Wagholi
[3]UG Scholar, Electrical Dept. JSPM's Bhivrabai Sawant polytechnic, Wagholi
[4]UG Scholar, Electrical Dept. JSPM's Bhivrabai Sawant polytechnic, Wagholi
[5]UG Scholar, Electrical Dept. JSPM's Bhivrabai Sawant polytechnic, Wagholi
[6]Asst. Prof. Electrical Dept. JSPM's Bhivrabai Sawant polytechnic, Wagholi

ABSTRACT-
By substituting an electronic fob for physical keys, push button start systems allow keyless vehic le ignition while providing increased security via encrypted, rolling-code signals.
The fob must be in close proximity for the system to work, which uses sensors and the BCM (Bo dy Control Module) to validate credentials before permitting a button press and the brake pedal t o start the engine.

I. INTRODUCTION

The automotive and industrial industries have seen substantial changes in recent years due to the quick development of automation and electronic control systems. Electronic, keyless, and push-button ignition systems are gradually replacing conventional ignition systems, which depend on physical locking mechanisms and mechanical keys.These contemporary methods provide superio r control over the ignition process, increased operational reliability, and increased user convenien ce.Electronic ignition systems extend system longevity and decrease physical wear by doing awa y with mechanical components. Mechanical ignition systems suffer from several inherent limitations, such as wear and tear of key mechanisms, accidental key loss, duplication risks, and unauthorized access. Over time, mechanical components degrade due to repeated use, leading to ignition failures and increased maintenance costs. Additionally, the use of physical keys poses security risks, as keys can be stolen or misused, compromising vehicle or equipment safety.

The “Start Ignition Without Key Using Push Button” project presents a microcontroller-based ignition control system in which a simple push button replaces the conventional ignition key. The system is developed using an Arduino UNO microcontroller, which serves as the main control unit, and an L293D motor driver IC, which safely controls a DC motor representing the engine.

This setup allows precise control of motor operation while isolating the microcontroller from high-current motor loads.

II. LITERATURE REVIEW

Previous research and development in the field of automotive electronics highlights a significant transition from traditional mechanical systems to advanced electronic and smart control mechanisms. Early vehicle ignition systems relied entirely on mechanical components, which were prone to wear, failure, and limited security. With advancements in electronics, ignition systems have evolved into electronic ignition systems that offer improved reliability, efficiency, and safety.

Several research studies and commercial implementations focus on smart ignition systems that enhance vehicle security and user convenience. Common approaches include RFID-based ignition systems, biometric authentication systems, and mobile or Bluetooth-controlled ignition mechanisms. These systems ensure that only authorized users can start the vehicle, thereby reducing theft and unauthorized usage. While such solutions provide advanced security features, they also introduce increased system complexity, higher hardware costs, and greater dependency on external devices such as smartphones, RFID cards, or biometric modules..

III. PROBLEM STATEMENT

Conventional ignition systems rely on mechanical keys and lack automation, flexibility, and electronic control. Existing low-cost electronic ignition models often fail to simulate real engine behavior and do not implement proper timing logic.

There is a need for a simple, reliable, and low-cost keyless ignition system that:

- Eliminates the mechanical key
- Uses a push-button interface
- Simulates real engine starting behavior
- Is suitable for educational and prototype applications

IV. OBJECTIVES

The objectives of this project are:

- To design a keyless ignition system using a push button
- To use Arduino UNO for timing and control logic
- To control a DC motor using L293D motor driver IC
- To implement a 5-second button press validation

Innovative Journal for Opportunities and Development in Science & Technology,
Volume 6, Issue 3, Feb. 2026
Paper Id: – IJ2025V6I3 -3 A-F
ISSN 2582-6026

- To simulate engine cranking using ON–OFF motor cycles
- To run the motor continuously after successful ignition
- To improve understanding of embedded control systems

V. SYSTEM OVERVIEW

System Architecture and Working Principle

Each functional block in the suggested ignition system simulation's modular architecture plays a distinct role in the system's overall operation.Clarity, scalability, and implementation ease are all enhanced by this blockbased design.The input unit, control unit, driver unit, and output unit are t he four main components of the system.Input Unit – Push ButtonThe input unit of the system is a push button, which acts as the ignition trigger. This button represents the ignition key or start button used in real-world vehicles. Instead of initiating the motor immediately upon a short press, the system requires the button to be pressed continuously for a predefined duration of 5 seconds.

This time-based button validation serves multiple purposes:

- It prevents accidental motor activation due to brief or unintended presses.
- It ensures that the ignition command is intentional.
- It mimics real ignition behavior where the key or button must be held for a short period to start the engine.

The push button is connected to one of the digital input pins of the Arduino UNO, allowing the microcontroller to continuously monitor its state.

Control Unit – Arduino UNO

The Arduino UNO acts as the central control unit of the system. It is responsible for reading input signals from the push button, processing logic conditions, executing timing sequences, and generating control signals for the motor driver.

Once the Arduino detects that the push button has been pressed continuously for 5 seconds, it initiates the ignition sequence. The sequence is implemented using precise software delays and conditional logic, allowing the Arduino to simulate different ignition stages.

The Arduino ensures that:

- The ignition sequence only starts after successful validation of the input.
- The motor control follows the exact predefined ON–OFF timing pattern.

- After completion of the ignition cycle, the motor remains in continuous running mode until power is turned OFF or reset.

By handling all timing and logic operations, the Arduino UNO provides flexibility and accuracy in simulating real-world ignition behavior.

Driver Unit – L293D Motor Driver

The L293D motor driver IC serves as the interface between the low-power Arduino control signals and the high-power DC motor. Since the Arduino cannot directly drive a motor due to current and voltage limitations, the L293D is used to safely amplify the control signals.

Key functions of the L293D include:

- Providing sufficient current to drive the DC motor
- Protecting the Arduino from back electromotive force (EMF)
- Allowing bidirectional motor control if required
- Ensuring electrical isolation between control and power circuits

The Arduino sends digital control signals to the L293D, which in turn switches the motor ON or OFF according to the ignition sequence.

Output Unit – DC Motor

The DC motor represents the vehicle engine in this system. Although a real engine involves complex mechanical and electrical components, the motor is used as a simplified and effective simulation tool.

The motor behavior follows a predefined ignition sequence, which closely resembles real engine startup dynamics:

1. Motor ON for 5 seconds – Simulates initial engine cranking
2. Motor OFF for 5 seconds – Represents stabilization or ignition pause
3. Motor ON for 5 seconds – Simulates secondary ignition attempt
4. Motor OFF for 5 seconds – Represents final stabilization
5. Motor ON continuously – Represents successful engine start and running condition

This staged ON–OFF pattern replicates the gradual process involved in starting an internal combustion engine rather than an instant start.

VI. WORKING PRINCIPLE

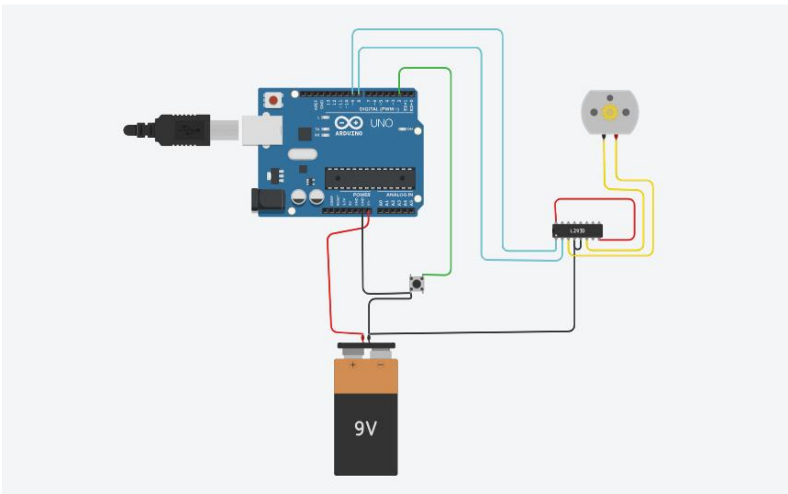
Innovative Journal for Opportunities and Development in Science & Technology,
Volume 6, Issue 3, Feb. 2026
Paper Id: – IJ2025V6I3 -3 A-F
ISSN 2582-6026

The Arduino keeps an eye on the push button status while the system is turned on.Nothing happens if you press and hold the button for less than five seconds.The Arduino verifies the input and initiates the ignition sequence when the button has been held steadily for five seconds.The DC motor is then turned on in accordance with the preset timing sequence by the L293D motor driver, which receives control signals from the Arduino.

A successful ignition is indicated by the motor running continuously after the ON–OFF cycles are finished.This approach provides a realistic, safe, and educational simulation of an automotive ignition system. It enhances traditional motor control projects by introducing timing validation, sequential control, and continuous operation, making it more aligned with real-world automotive electronics.

Circuit Diagram Explanation

The given circuit diagram represents a button-controlled DC motor ignition system using Arduino UNO and an L293D motor driver IC. The setup simulates a real-world engine starting process by controlling a DC motor through a predefined sequence.



1. Arduino UNO (Control Unit)

The Arduino UNO is the central controller of the system. It performs the following functions:

- Reads the input from the push button
- Executes the ignition logic and timing sequence
- Sends control signals to the L293D motor driver
- Controls the ON and OFF states of the DC motor

The Arduino is powered through the USB cable or external supply and shares a common ground with the motor driver circuit.

2. Push Button (Input Unit)

- The push button is connected to a digital input pin of the Arduino.
- One terminal of the button is connected to Arduino input pin, and the other terminal is connected to GND.
- When the button is pressed and held, Arduino detects a LOW/HIGH signal (depending on logic used).

3. L293D Motor Driver IC (Driver Unit)

The L293D IC is used to drive the DC motor safely because:

- Arduino cannot supply sufficient current to drive a motor directly
- Motors generate back EMF which can damage microcontrollers

4. DC Motor (Output Unit)

- The DC motor represents the vehicle engine
- It is connected to the output pins of the L293D
- The motor operates according to signals received from Arduino via the motor driver

Motor Behavior:

- ON and OFF cycles simulate engine cranking
- After successful ignition sequence, motor runs continuously

VII. HARDWARE REQUIREMENTS

7.1 Components Used

- Arduino UNO
- L293D Motor Driver IC
- DC Motor
- Push Button
- Power Supply (5V & 12V)
- Connecting Wires
- Breadboard

7.2 Arduino UNO

Innovative Journal for Opportunities and Development in Science & Technology,
Volume 6, Issue 3, Feb. 2026
Paper Id: – IJ2025V6I3 -3 A-F
ISSN 2582-6026

The Arduino UNO is based on the ATmega328P microcontroller and provides digital I/O pins, timers, and easy programming support.

7.3 L293D Motor Driver

The L293D IC allows bidirectional control of DC motors and protects the microcontroller from high current loads.

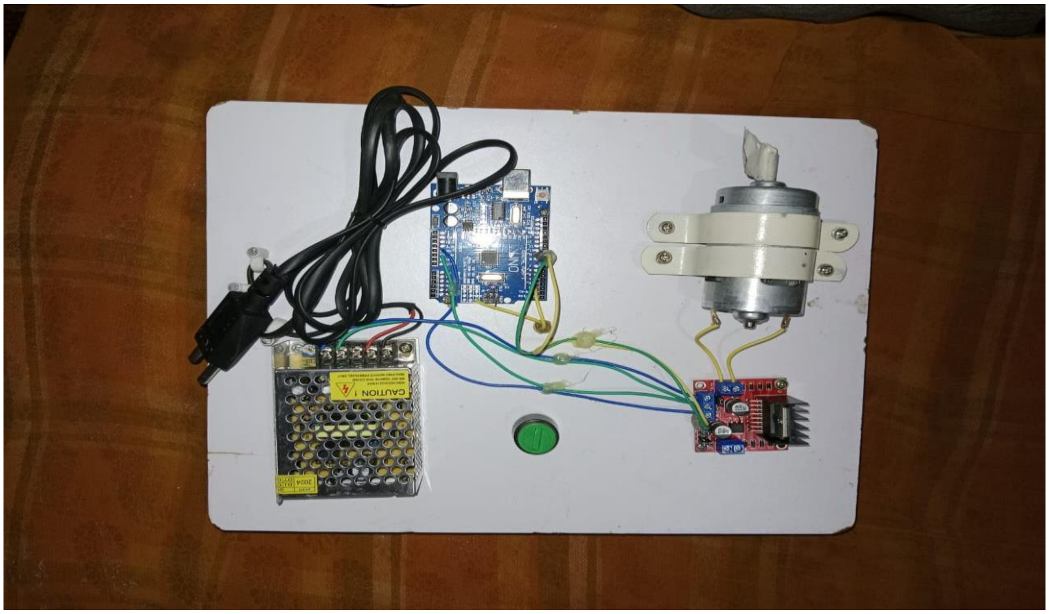
7.4 Push Button

Used as a user input device to initiate the ignition sequence.

Working Principle

1. The system remains idle until the push button is pressed.
2. If the button is pressed continuously for 5 seconds, the ignition sequence starts.
3. The Arduino controls the motor through the L293D driver.
4. The motor runs in ON–OFF cycles to simulate engine cranking.
5. After completing the cycles, the motor runs continuously.
6. If the button is released early, the system resets.
1. Motor ON continuously
2. Stop

VIII. RESULT



IX. CONCLUSION

The "Start Ignition Without Key Using Push Button" project effectively illustrates how to use the Arduino UNO microcontroller to create a straightforward, dependable, and reasonably priced ke yless ignition system.The device successfully replicates the actual enginestarting sequence, inclu ding timed ONOFF cycles followed by continuous motor running, by combining a pushbutton int erface, an L293D motor driver, and a DC motor.Understanding ignition control mechanisms requ ires a realistic depiction of engine cranking behavior, which this method offers.The project demo nstrates the useful use of embedded system ideas, such as logicbased automation, timing control, sensor input handling, and actuator interface.It provides insightful information about how conven tional mechanical systems might be replaced by microcontrollers to increase convenience, effecti veness, and dependability in industrial and automotive applications.Due to its simplicity, low-cost components, and modular design, the system is highly suitable for educational purposes, laboratory experiments, and prototype-level development. Moreover, the modularity of the design allows for future enhancements, such as incorporating security authentication, wireless control, or integration with actual vehicle engines, making it a scalable platform for advanced research and development in the field of electronic ignition systems.

Overall, the project serves as an effective educational tool that bridges the gap between theoretical knowledge and practical implementation of keyless ignition systems, while also laying the foundation for more sophisticated and real-world automotive applications.

.