Available online www.jsaer.com

***Journal of Scientific and Engineering Research*, 2025, 12(4):91-98**

Research Article

ISSN: 2394-2630
CODEN(USA): JSERBR

Development And Validation of Remote Keyless Entry (RKE) System Using Can Protocol for Secure Vehicle Access

Mr. M. Suresh Babu[1], Dr. Ch. Hariprasad[2], Mr. A. Durgaprasad[3], Mr. R. Manikanta[4], Ms. N. Jayasree[5], Ms. P. Pavani Ramya Sri[6]

[1,2]Assistant. Professor, dept. of EEE, Bapatla Engineering College, Bapatla, India
[3,4,5,6]dept. of EEE, Bapatla Engineering College, Bapatla, India
Email: [1]sureshbabu.maddina@becbapatla.ac.in, [2]hariprasad.chitte@becbapatla.ac.in,
[3]durgaprasad19627@gmail.com, [4]manikantaravuri16@gmail.com, [5]jayasreenizampatnam@gmail.com,
[6]pavaniramyasripallempati@gmail.com

Abstract: The project focuses on developing a Remote Keyless Entry (RKE) system using the CAN protocol to enable secure and wireless vehicle access. It integrates key components like the keyfob and Body Control Module (BCM), ensuring encrypted communication for functions such as locking, unlocking, and remote start. TS Master software is employed for writing CAPL scripts to simulate CAN communication and for validating system performance through Panel, Trace, and Graphics tools. The RKE system emphasizes robust security, scalability, and compatibility with a wide range of vehicles, following industry standards for future adaptability and market readiness.

Keywords: RKE, ECU, CAN communication, BCM, CAN cable, Arbitration, IVN Bed, CAN High and CAN Low, TS Master, CAPL Scripting, Panel, Trace, Graphics, Topway HMI- display.

1. Introduction

In today's rapidly advancing automotive industry, communication between various electronic systems inside a vehicle is critical. The Controller Area Network (CAN) protocol has made it possible to connect multiple Electronic Control Units (ECUs) efficiently, enabling reliable, real-time communication without the need for complex wiring. Taking advantage of this technology, our project focuses on designing and implementing a Remote Keyless Entry (RKE) system, aimed at improving vehicle security and user convenience. The RKE system allows drivers to lock, unlock, and even remotely start their vehicles without using a physical key. Instead, a keyfob transmits encrypted signals to the vehicle's Body Control Module (BCM) via the CAN bus. This secure communication ensures that only authorized users can access and control the vehicle. For the development and testing of the system, we used TS Master software, a tool that allows us to write CAPL (Communication Access Programming Language) scripts to simulate and control CAN messages. We also utilized TS Master's Panel, Trace, and Graphics features to visualize the CAN signals, monitor the message flow, and validate the system's performance during different RKE operations. In our setup, CAN High and CAN Low wires were used to establish proper CAN communication. We also focused on important aspects like Arbitration, where the priority of CAN messages is managed to avoid collisions on the network. Testing was carried out using an In-Vehicle Network (IVN) Bed, providing a real-world environment to check the reliability and response of the system.

This project not only ensures secure and wireless vehicle entry but is also scalable, meaning it can be easily adapted for newer features or integrated into future car models. By strictly following automotive industry

standards and best practices, our RKE system stands ready for real-world application, offering both flexibility and enhanced security to vehicle owners.

2. Literature Review

1. Evolution of Remote Keyless Entry Systems

The evolution of RKE systems began with simple, unencrypted RF systems, which were eventually found vulnerable to hacking techniques such as signal interception and replay attacks. Researchers have explored various methods to improve RKE security, including the use of rolling codes and encryption algorithms. Early implementations of RKE systems relied on low-frequency RF communication, but this method has shown to have security flaws, such as code grabbing and intercepting signals. The next step was to incorporate cryptographic techniques, including symmetric encryption to prevent unauthorized access. However, while these solutions improved security, they often did not provide scalability or adaptability to future vehicle architectures.

2. CAN Protocol in Automotive Communication

The CAN protocol, developed by Bosch in 1983, has become the backbone of in-vehicle communication. It connects various ECUs, including engine control units, airbags, braking systems, and infotainment modules, enabling them to exchange data in real-time. The high fault tolerance, low-latency communication, and error detection capabilities of CAN make it ideal for critical systems like RKE (Bosch, 2018). The protocol uses two wires—CAN High and CAN Low—to transmit data at high speeds, making it robust for automotive applications where reliability is paramount. Integrating CAN protocol into RKE systems offers several advantages. One of the main benefits is secure encryption, which ensures that only authorized signals from the key fob are processed by the vehicle's Body Control Module (BCM). CAN's built-in message priority and arbitration mechanisms allow for efficient communication, even in a system where multiple ECUs are involved (Bosch, 2019).

3. Security Challenges in Traditional RKE Systems

Security remains a major concern in traditional RKE systems, particularly regarding signal interception and replay attacks. In some older RKE systems, the communication between the key fob and vehicle was based on static codes, which could easily be intercepted and replayed by unauthorized individuals. A study by Smith et al. (2020) showed that attackers could bypass security systems by capturing the signal and transmitting it later to unlock vehicles. This problem led to the adoption of rolling codes, where each signal is unique and changes with each transmission. While rolling codes enhanced security, they were still vulnerable to some sophisticated attacks, such as code prediction and signal amplification. In response to these vulnerabilities, more advanced solutions have been proposed, such as advanced cryptographic protocols and the integration of the CAN protocol. These solutions focus on enhanced encryption and the use of secure keys for communication between the key fob and BCM, minimizing the risk of unauthorized access.

4. Role of TS Master Software and CAPL Scripting in RKE Systems

For developing and testing the CAN communication in RKE systems, tools like TS Master are used extensively. TS Master enables the simulation of CAN messages, the creation of test cases, and the visualization of data through its Panel, Trace, and Graphics tools. These features allow engineers to test various CAPL (CAN Access Programming Language) scripts to simulate real-world conditions and ensure the security and reliability of the system.

In particular, CAPL scripting is used to simulate the interaction between the key fob and the Body Control Module (BCM), ensuring that the system can handle various scenarios such as locking, unlocking, and remote start commands. The Trace feature is especially useful for monitoring the flow of CAN messages in real-time, which helps in detecting errors, troubleshooting, and optimizing the system's performance.

3. Proposed Methodology

1. Signal Transmission (Keyfob)

• The key fob sends encrypted signals to the vehicle's BCM. These signals will contain commands like unlock, lock, or remote start.

• The key fob uses the CAN protocol to send these signals through the CAN bus (using CAN High and CAN Low wires).

2. Signal Reception (BCM)

• The Body Control Module (BCM) on the vehicle receives the encrypted signals from the key fob through the CAN bus.

• The BCM will decrypt the received signal to verify the authenticity and perform the requested function.

3. Security Mechanisms

• The RKE system will use encryption for secure communication between the key fob and BCM.

• Authentication checks will be implemented to ensure the legitimacy of the key fob and prevent unauthorized access.

4. Testing and Validation (Using TS Master)

• The system will be tested using TS Master software, which will simulate CAN communication between the key fob and BCM.

• CAPL scripting will be used to create test cases that simulate real-world scenarios (e.g., unlocking the vehicle, starting the engine).

• Tools like Panel, Trace, and Graphics in TS Master will help monitor and visualize CAN messages in real-time, ensuring the system works correctly and securely.

5. Error Handling

• In case of any transmission errors or invalid signals, the BCM will reject the command and notify the user with appropriate feedback (e.g., lights flashing or beeping).

6. Integration and Scalability

• The system will be designed to integrate easily with existing vehicle networks using the CAN bus.

• The RKE system will be scalable, allowing for future improvements and integration with more advanced security features (like biometric authentication or mobile app integration).

4. System Architecture

1. Key Dongle (Key Fob):

• Microcontroller: A microcontroller (e.g., Arduino, ESP32) is used in the key fob to handle the signal generation and encryption. The microcontroller will process inputs from the user (e.g., pressing the unlock or start button) and send the corresponding encrypted signal to the BCM.

• Encryption Mechanism: The system will use a basic encryption method such as AES (Advanced Encryption Standard) or a rolling code technique to ensure secure communication between the key fob and the BCM. This prevents unauthorized access by hacking the signals.

• CAN Transceiver: A CAN transceiver (e.g., MCP2515) is used to send signals via the CAN bus to the BCM.

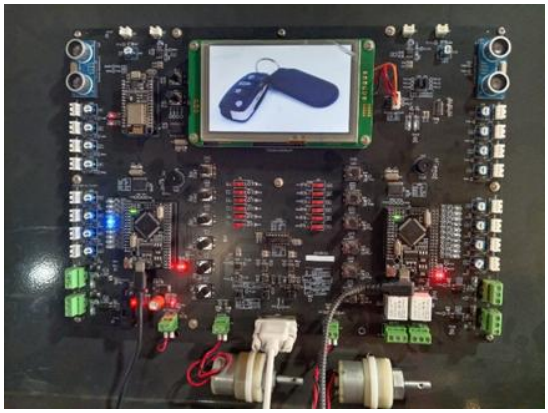

Figure 1: IN-VEHICLE TEST(IVN) BEDS

2. Body Control Module (BCM):

• Microcontroller: The BCM uses a microcontroller (e.g., ESP32) to manage the control logic of the vehicle, process incoming signals from the key fob, and execute vehicle functions such as locking/unlocking the doors and starting the engine.

• CAN Transceiver: The BCM uses a CAN transceiver to receive the encrypted signals from the key fob.

• Vehicle Subsystems: The BCM is connected to the door lock systems and engine control units via other ECUs. These ECUs receive commands from the BCM to perform actions like locking/unlocking the doors and starting the engine

3. Communication Bus (CAN Bus):

• the Communication Bus, specifically the CAN Bus (Controller Area Network), plays a vital role in transmitting keyfob-related commands across different Electronic Control Units (ECUs). When a keyfob button is pressed (such as lock, unlock, or trunk release), the signal is first received by the Remote Keyless Entry (RKE) ECU, which validates and interprets the wireless command. The RKE ECU then communicates this information to the Body Control Module (BCM). From there, the BCM generates a CAN frame and transmits it over the CAN Bus, a robust, high-speed serial communication network designed to allow multiple ECUs to exchange information efficiently. Other ECUs, such as the Instrument Cluster ECU and the Immobilizer ECU, listen to the CAN Bus. Based on the CAN message, the instrument cluster may update the HMI display (showing "Doors Locked" or "Trunk Opened"), while the immobilizer prepares for engine start authorization. Each message on the CAN Bus typically includes an identifier (CAN ID), data length (DLC), and a payload (actual command data), enabling synchronized and secure vehicle operations triggered by the keyfob.CAN High (CAN_H) and CAN Low (CAN_L) wires are used for communication. The CAN protocol allows for the exchange of small data frames between ECUs.

Figure 2: CAN CABLE

4.TS Master and Topway Tools

This tool plays a central role in monitoring, controlling, and analyzing the interactions between different components in the test bench. TS Master is used for generating CAN signals, simulating different vehicle states, and automating test cases. Through scripting and panel creation, the system is able to simulate complex real-world scenarios and validate the behavior of the BCM and IC under these conditions.

5.CAPL Scripting for RKE Monitoring

CAPL (CAN Application Programming Language) is widely used to simulate, test, and automate communication in automotive networks, particularly for features like keyfob operations. In keyfob communication testing, CAPL scripts are developed to simulate button press actions such as lock, unlock, or trunk release by generating and transmitting specific CAN messages onto the network. A typical CAPL program for this application involves defining a message structure with a specific CAN ID, setting the appropriate data bytes to represent the keyfob command, and sending the message when the simulation starts. Additionally, CAPL can be used to listen for response messages from other ECUs, such as lock status confirmations from the Body Control Module (BCM) or display updates from the Instrument Cluster. This helps in verifying whether the vehicle reacts correctly to simulated keyfob inputs. By using CAPL, keyfob functionalities can be efficiently validated in a controlled, repeatable, and automated environment, which significantly improves the development and testing process for in-vehicle network systems.

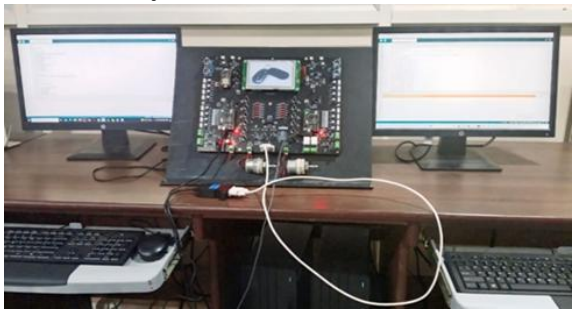

Figure 3: Connecting two IVN test bed using can cable.

Figure 3 shows the connection diagram of the total hardware.

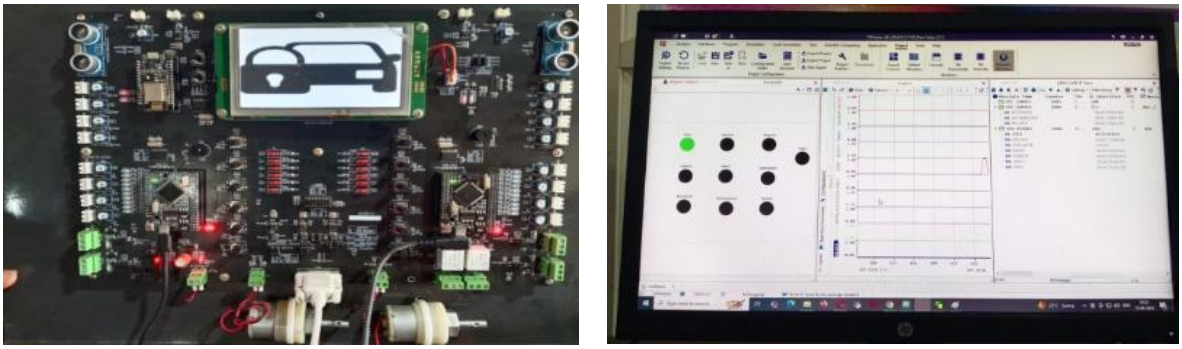

Figure 4

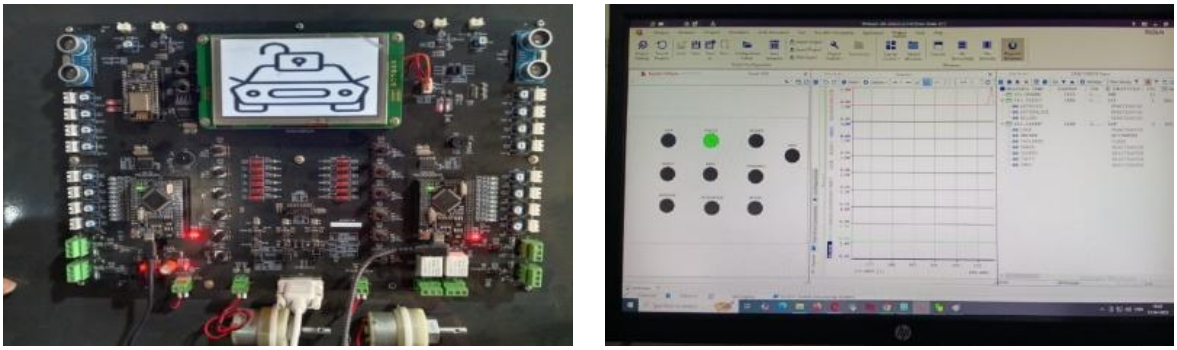

Figure 5

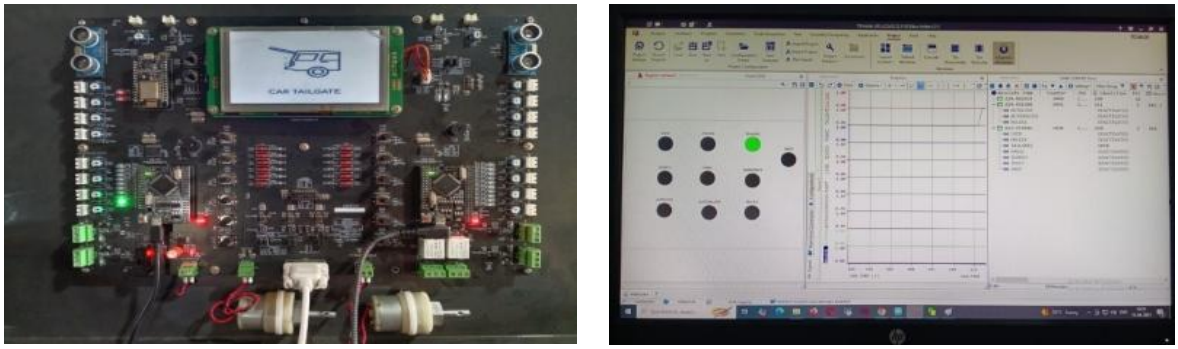

Figure 6

Figure 7

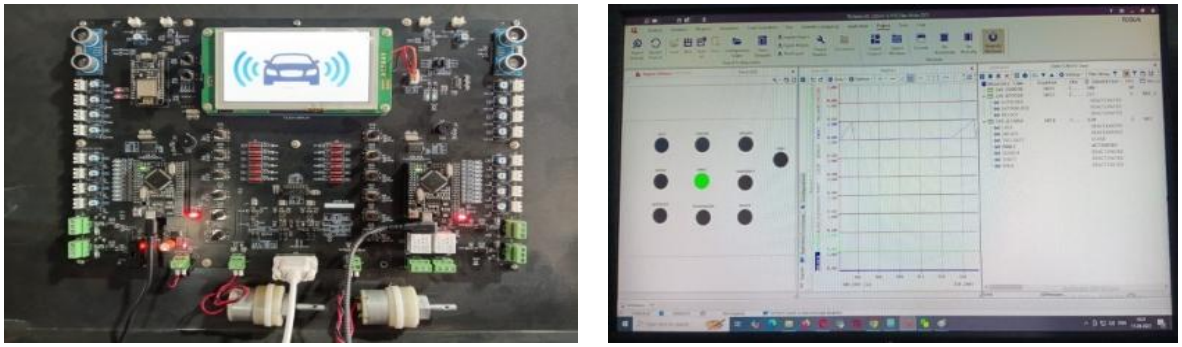

Figure 8

Figure 9

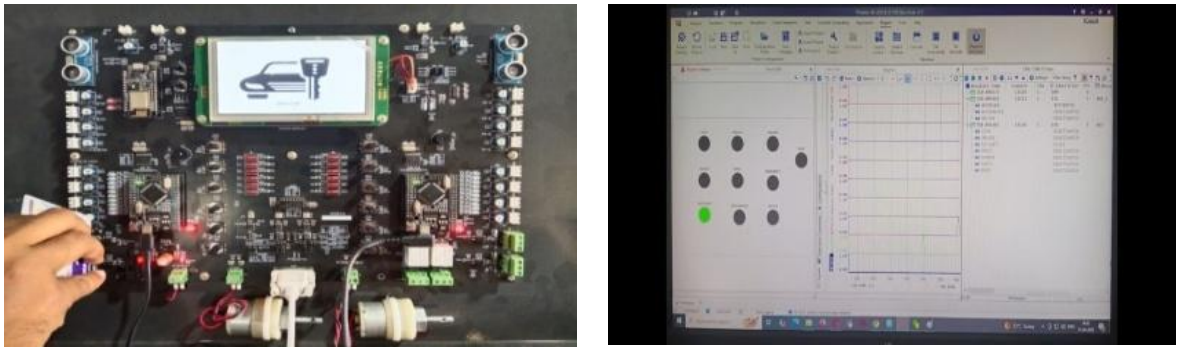

Figure 10

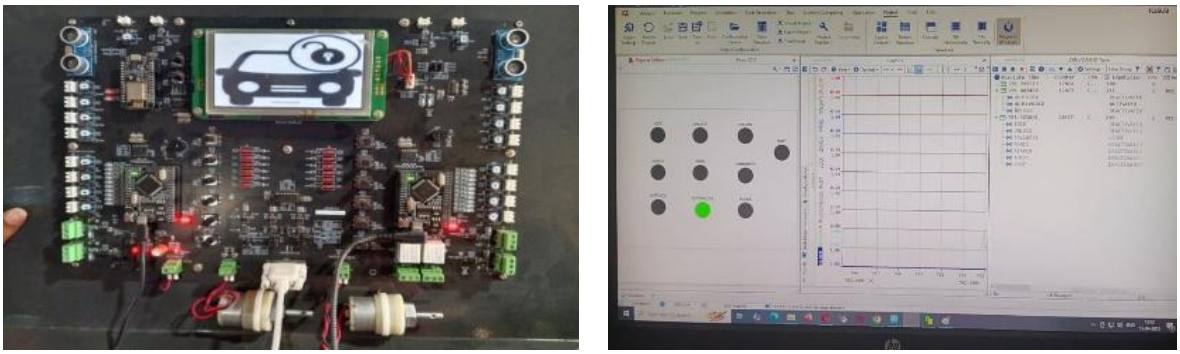

Figure 11

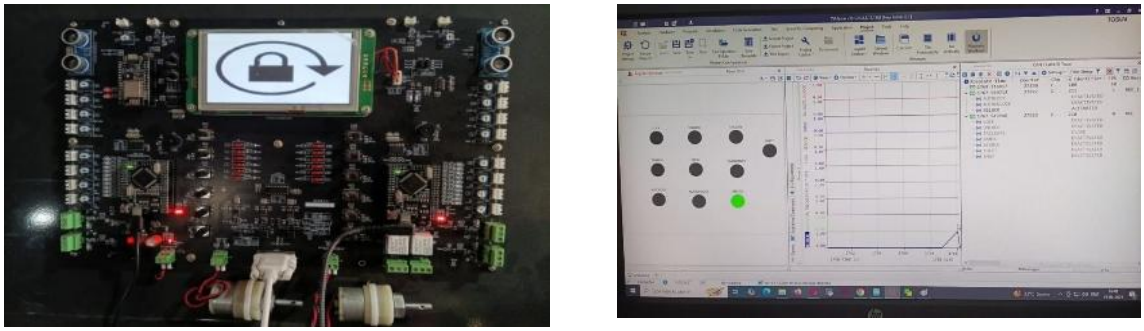

Figure 12

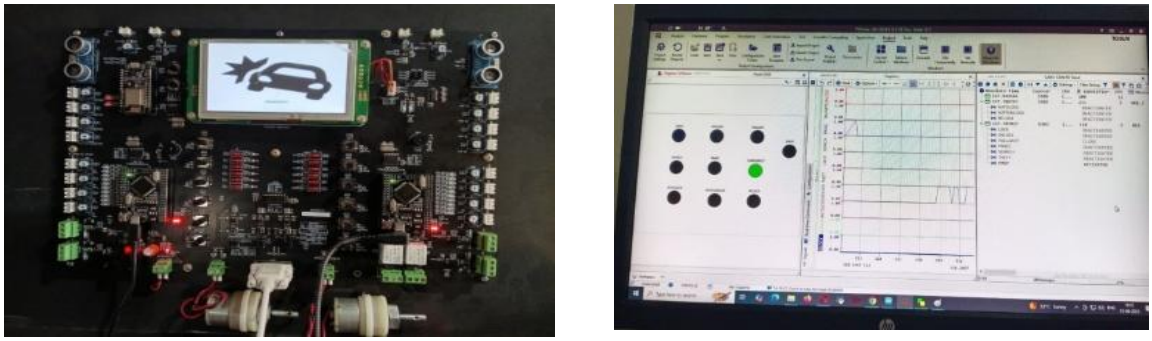

Figure 13

Figure 4 shows the output of lock function, whenever the lock pushbutton is pressed in the transmitter ecu (RKE) the receiver will receive the lock request from RKE and checks all the conditions mentioned if all the conditions are satisfied lock function will be enabled with a buzzer sound and led on indication and output is observed in TSMASTER by visual and graphical manner.

Figure 5 shows the output of unlock function, whenever the unlock pushbutton is pressed in the transmitter ecu (RKE) the receiver will receive the unlock request from RKE and checks all the conditions mentioned if all the conditions are satisfied unlock function will be enabled with a buzzer sound and led on indication and output is observed in TSMASTER by visual and graphical manner.

Figure 6 shows the output of tailgate open function, whenever the Tailgate open pushbutton is pressed in the transmitter ecu (RKE) the receiver will receive the Tailgate open request from RKE and checks all the conditions mentioned if all the conditions are satisfied Tailgate open function will be enabled with a buzzer sound and led on indication and output in TSMASTER in visual and graphical manner and output is observed in TSMASTER by visual and graphical manner.

Like that figure 7,8,9,10,11,12,13 shows the search,panic, theft, autolock, autounlock, autorelock, emergency functions are performed based upon condition respectively and related output is observed in TSMASTER by visual and graphical manner.

FUTURE SCOPE:

1. Passive Keyless Entry (PKE)

It is the advanced version of Remote Keyless Entry (RKE). In PKE, you don't even need to press a button to unlock the car. The car automatically detects the proximity of the key fob (or smartphone) and unlocks the doors as soon as you approach. Similarly, it locks itself when you walk away.

2. Integration with Smartphones

The future of RKE systems is moving beyond traditional key fobs. Smartphones and wearable devices like smartwatches will soon replace physical keys completely. Through dedicated mobile apps, users will be able to lock, unlock, start, and monitor their vehicles remotely, offering greater convenience and flexibility

3. Integration of Biometric Authentication

Biometric technologies such as fingerprint scanning, facial recognition, voice recognition, and iris scanning will enhance RKE security. Only the authorized driver will be able to access the vehicle, providing an extra layer of protection against theft, even if the smartphone or wearable device is lost or stolen.

5. Conclusion

The integration of Remote Keyless Entry (RKE) systems with the Controller Area Net work (CAN) protocol represents a significant leap forward in enhancing vehicle security, operational efficiency, and user convenience. By leveraging the robust and standardized communication infrastructure offered by CAN, RKE systems can be seamlessly embedded within modern automotive architectures while maintaining high levels of data integrity and system reliability. This research highlights the importance of carefully designing message formats, incorporating secure encryption methods, and ad dressing real-world implementation challenges to ensure optimal performance.

The incorporation of TSMaster in the prototype phase has proven invaluable for monitoring CAN bus activity, validating system behavior, and facilitating efficient de bugging. Overall, the study demonstrates that a CAN-based RKE solution is not only technically viable but also highly scalable, paving the way for future innovations in connected and intelligent vehicle access systems.

References

[1]. A. I. Alrabady and S. M. Mahmud, "Analysis of attacks against the security of keyless-entry systems for vehicles and suggestions for improved designs," IEEE trans actions on vehicular technology, vol. 54, no. 1, pp. 41–50, 2005.

[2]. F. Bersani and H. Tschofenig, "The eap-psk protocol: A pre-shared key extensible authentication protocol (eap) method," Tech. Rep., 2007. item

[3]. C. Böhm, M. Hofer, and W. Pribyl, "A microcontroller sram-puf," in 2011 5th International Conference on Network and System Security. IEEE, 2011, pp. 269–273.

[4]. N. T. Courtois, G. V. Bard, and D. Wagner, "Algebraic and slide attacks on keeloq," in Fast Software Encryption: 15th International Workshop, FSE 2008, Lausanne, Switzer land, February 10-13, 2008, Revised Selected Papers 15. Springer, 2008. pp. 97–115.

[5]. T. Eisenbarth, T. Kasper, A. Moradi, C. Paar, M. Salmasizadeh, and M. T. M. Shal mani, "On the power of power analysis in the real world: A complete break of the keeloq code hopping scheme," in Advances in Cryptology–CRYPTO 2008: 28th An nual International Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2008. Proceedings 28. Springer, 2008, pp. 203–220.

[6]. F. D. Garcia, D. Oswald, T. Kasper, and P. Pavlidès, "Lock it and still lose it—on the (In) Security of automotive remote keyless entry systems," in 25th USENIX security symposium (USENIX Security 16), 2016.

[7]. J.-R. Lin, T. Talty, and O. K. Tonguz, "On the potential of bluetooth low energy technology for vehicular applications," IEEE Communications Magazine, vol. 53, no. 1, pp. 267–275, 2015.

[8]. S. Ravi, A. Raghunathan, P. Kocher, and S. Hattangady, "Security in embedded systems: Design challenges," ACM Transactions on Embedded Computing Sys- tems (TECS), vol. 3, no. 3, pp. 461–491, 2004.

[9]. N. Semiconductor, "Nrf52832 product specification," Nordic Semiconductor, 2017.

[10]. P. Smith, "Comparing low-power wireless technologies," Tech Zone, Digikey On line Magazine, Digi-Key Corporation, vol. 701, 2011.

[11]. P. Štembera and M. Novotny, "Breaking hitag2 with reconfigurable hardware," in 2011 14th Euromicro Conference on Digital System Design. IEEE, 2011, pp. 558–563.

[12]. R. Verdult, F. D. Garcia, and B. Ege, "Dismantling megamos crypto: Wirelessly lockpicking a vehicle immobilizer," in Supplement to the Proceedings of 22nd USENIX Security Symposium (Supplement to USENIX Security 15), 2015, pp. 703–718.