Design and Implementation of a Vehicle Data Transmission Protocol Using the PRESENT Algorithm

[1]Chitti Rasanya, [2]PankajRangaree

[1]M. Tech VLSI, Dept. of ECE, Vaagdevi College of Engineering, Warangal, Telangana, India.
[2]Assistant Professor, Dept. of ECE, Vaagdevi College of Engineering, Warangal, Telangana, India

Abstract- *It is essential to capture and analyse real time vehicle operating data using IoT devices to evaluate the operational state of transportation vehicles. The collected data, on the other hand, includes multiple-source divergent properties, restricted network resources, poor server performance. Real-time data processing may be challenging to successfully execute. Furthermore, data have to be sent via network; it is vital to make certain the data transmission security. Given the aforementioned challenges, it is critical to organize data and create standard data format for data transfer and evaluation. Concurrently, improve server communication and concurrent processing capabilities. Moreover, because data must be communicated across a network, PRESENT lightweight data encoding method will be used to assure security of the information transfer. This technique has substantially lower hardware requirements than encryption algorithms like AES. The dynamic key version created by this work is a once-only pad that significantly increases data security. It incorporates the project's characteristics and makes use of the rate of interaction between the device and servers.*

Keywords: ***Monitoring of a Vehicle, Present Algorithm, Vehicle Data Transmission Protocol***

1. INTRODUCTION

Swift expansion of mobile networks as well as the progressive drop in prices of IoT devices laid solid base to the growth of IoT industry. IoT industry growth supports existing companies along with giving rise to new ones as well. While, the IoT device number has grown rapidly, so has the amount of data, and growing demands have been put on processing, transferring, storing, and protecting data. The IoT's technology is now broadly employed in a variety of companies, but it has also introduced a number of issues, such as device connectivity, system access, and device administration. When dealing with the problem of diverse information drawn from multi-sources, it is important to standardize data-format, streamline its structure, and, ideally, make it accessible with current technologies. In addition to making data collecting, processing, and transmission simpler, this will also make database storage and data analysis simpler. Additionally, the issue of data security during transmission must be taken into account. Therefore, some protections must be provided to guarantee that the data is not taken or altered. However, specific hardware resources are needed to encrypt the data. We must meet the low-power criteria for low-power Internet of Things devices while simultaneously ensuring that the encryption method utilized can offer some level of data security. The benefit of the lightweight encryption technique is that it utilizes less CPU power and has substantially lower hardware performance requirements than other encryption algorithms like AES. The outcome is a large increase in use time of IoT devices and a reduction in the requirement for battery replacement. From an engineering perspective, this work develops a vehicle tracking system that efficiently receives, processes, and realizes services including vehicle placement, personnel administration, authorities' management, asset management, and condition tracking.

2. LITERATURE REVIEW

Automatic gearbox systems, electronic fuel injection, airbag systems, anti-lock brake systems, and other innovations will be common in compressors with control units. These are the fundamental components of a modern automobile's gadgetry. It is dependent on the stability and protection of the whole system and is time-sensitive. Each subsystem's real-time needs must be met by a general information exchange that includes speed of an engine, wheel, and position of throttle must be developed. Information update fee is primarily used to determine the real-time requirement management module and control period differs. The material covers a variety of topics, such as cadence-gauge finish, gas dimension, A/D conversion, calculating circumstances, control trigger, and more. They transmit and receive data in electrical fuel control must be finished in less than one millisecond in order to obtain the requirements in real-time. The statistic trading community should therefore be competitive and have a quick-witted conversation style.

IoT devices have limited hardware resources and low power consumption, so a lightweight symmetric packet cypher set of rules that requires fewer resources is a popular area for research. PRESENT, CLEFIA, MIBS, and L Block are the foundational algorithms for lightweight block cyphers.

A simple block cypher is the PRESENT algorithm [1]. It was put forth in 2007 and is now covered by ISO/IEC 29192-2. The method requires a key of size eighty-bits or one twenty-eight bits and sixty-four bits data length block. Ordinarily, the eighty-bit key satisfies all of the prerequisites. The nonlinear layer of the method depends on a 4-bit S-box, there are 31 iterations, and hardware optimizations are taken into consideration. The technique utilizes a SP network topology. Since PRESENT-80's hardware implementation requires approximately 1570 gate equivalents, it can be employed, when low power consumption and excellent efficiency are needed. For smart homes, [3] has created a brand-new, highly-advanced chat protocol in order to conserve energy, protect privacy, and offer security to guarantee data integrity and dependability. [4] proposed a set of 64-bit block-length lightweight SIT cyphers. The method, which can be used with a cheap eight-bit microcontroller, requires 5 times to perform the encoding to offer high level of protection. [5] presents the current algorithm. The current set of rules uses a simple block cypher. Literature [6] proposes a data storage system that can analyse both structured and unstructured data in addition to effectively storing enormous amounts of IoT data. [7] assessed popular cloud IoT platforms, identified important cloud IoT service suppliers, and examined both positive and negative aspects of every IoT cloud provider.

3. EXISTING METHOD

Overall design of the vehicle control system is shown in Figure 1. The layers of perception, network, platform and application make up the architecture. The layer of perception is necessary for direct data flow, IoT, and information collection. This layer is made up of a network of sensors, including RFID network and sensor systems, as well as fundamental sensor components like tags with RFID and readers. It also includes a range of sensors, such as recording devices, GPS technology, and other sensing elements. The network layer, which is made up of many private networks, regional networks, Web, mobile phone networks, and sensor networks that are wireless, is primarily responsible for data transfer and reliable delivery. Among these capabilities are messaging servers, storage space for files servers, internet and app servers, and administration services for support like control of access and app installation, app performance management, and usage monitoring and billing, among others. The application layer is at the top of the Internet of Things' organizational structure. It was created to "process information" using cloud computing framework.

The perception layer collects statistics that are especially used by the application layer in the physical world for dynamic surveillance, real-time manipulation, proprietary administration, and clinical choice-making. Primary functions of IoTs' public service layer are to end information control and processing while integrating this information with diverse industry applications. Various sensors in the vehicle management system capture real-time records on vehicle popularity, such as temperature, time, latitude, longitude, and personnel statistics. Statistics are encrypted and analysed by a single-chip STM32 microcontroller before being transmitted to the wireless voice interchange network using BC-28 voice interchange module. Software layer provides the basis for dynamic tracking, real-time processing, accuracy processing, and data extraction, which complements control of information and data processing. Businesses might benefit from its precision control and analytical selecting capabilities. The application layer also implements people control, equipment management, proximity management, and device temperature monitoring. The MVC design structure, which helps eliminate coupling, is mostly used in the platform's construction. between buildings, making future maintenance simpler and enhancing usefulness. Ajax enables fact interaction, Spring, SpringMVC, and Mybatis frameworks create business feature separation, the MySQL database achieves data survival strength, and the JSON format unifies records. Fact rendering and car location rendering are implemented using Baidu Map API in conjunction with the HTML5 framework, CSS, JavaScript, Bootstrap, and styling plug-in integration, including Charts.

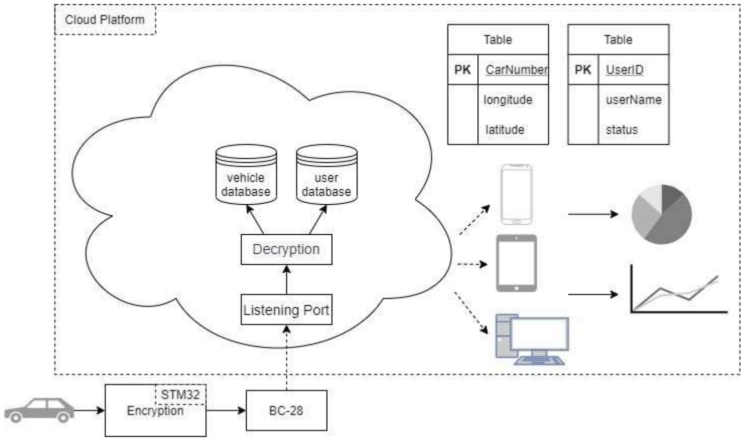


Figure 1. Vehicle Management System

4. PROPOSED SYSTEM

The STM32 single-chip microcontroller encrypts the data gathered by the vehicle's sensors using PRESENT algorithm before sending it to the cloud platform's servers via the BC-28 module, where server's software decodes the incoming ciphertext. These data will be transferred to the application level for comprehensive administration and processing, establishing the framework for real-time, precise, dynamic monitoring, and data mining, which may help businesses make informed management decisions. Additionally, functions like the administration of users, device management, location management, and device temperature tracking are integrated at application's layer, providing a more thorough and methodical management approach.

4.1 Encryption Algorithm Design

The current round is one of 31. The SP structure is repeated for each round. AddRoundKey, sBoxLayer, and pLayer are the three operations that make up each round. The permitted key lengths be 80 and 128 bits, while the block length being 64 bits. However, version of 80-bit keys are used in this project's implementation. This Version offers more than enough protection for the low-security applications that are typically required in tag-based installations [1], For each of the 31 rounds, a round key Ki for 1 i 32 is generated using a xor operation. K32 is then used for post-whitening, which is a bit-wise permutation, and not a linear substitution layer. Only one 4-bit S-box S is utilized 16 times simultaneously throughout each round in the non-linear layer. The cipher's pseudo-code description is shown in Figure 2.

4.1.1 Add Round Key

Given that the round key is Ki = Ki63... Ki0 for 1 ≤ i ≤ 32 and the current

Add Round Key is comprised of operations for state $0 \leq j \leq 63$, $bj \rightarrow bj \oplus \kappa ij$

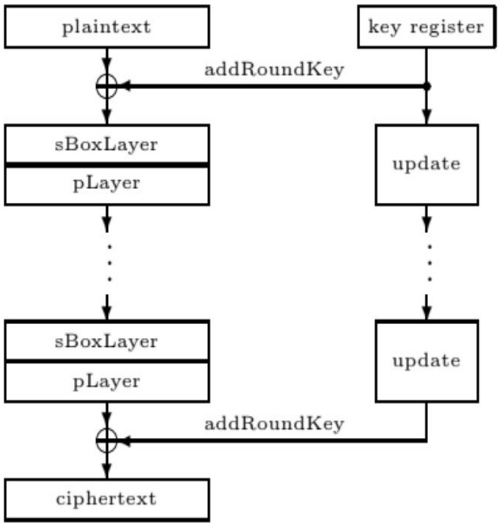


Figure 2. Block diagram of present Encryption algorithm

4.2 Decryption Algorithm Design

The reverse order of encryption is used in the decryption. The XOR gate 1 is provided with the cipher text and key value. The output of XOR gate 1 and XOR gate 2 is supplied to the decryption input for rounding. At the end of the 31st round, the original data will be retrieved.

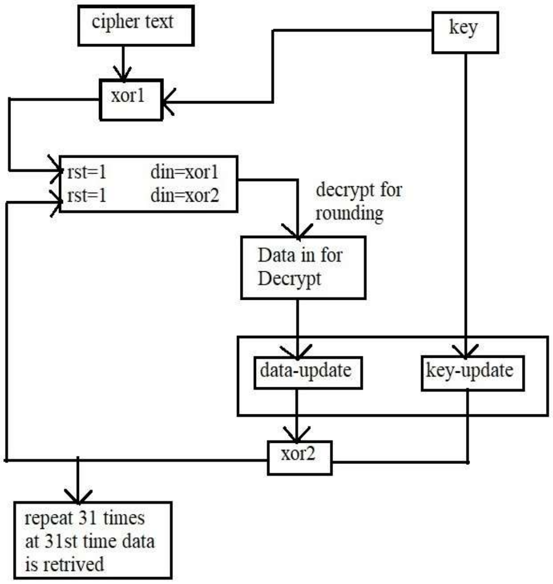


Figure 3. Block diagram of present Decryption algorithm

5. RESULTS

The below results show the encryption and decryption using PRESENT algorithm. The below figure shows the Encryption output. After 31 rounds of iteration the plain text is converted into a cipher text.

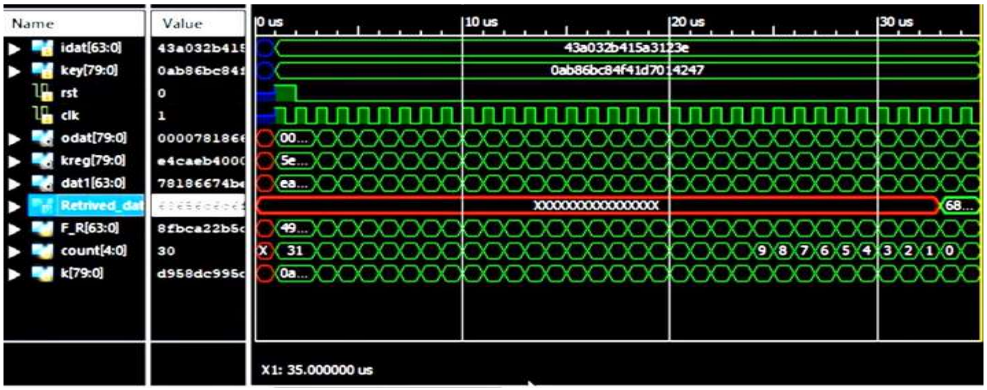


Figure 4. Successful encrypt waveform

The below fig shows the decryption waveform after 31 rounds of iteration the cipher text is converted into the plain text.

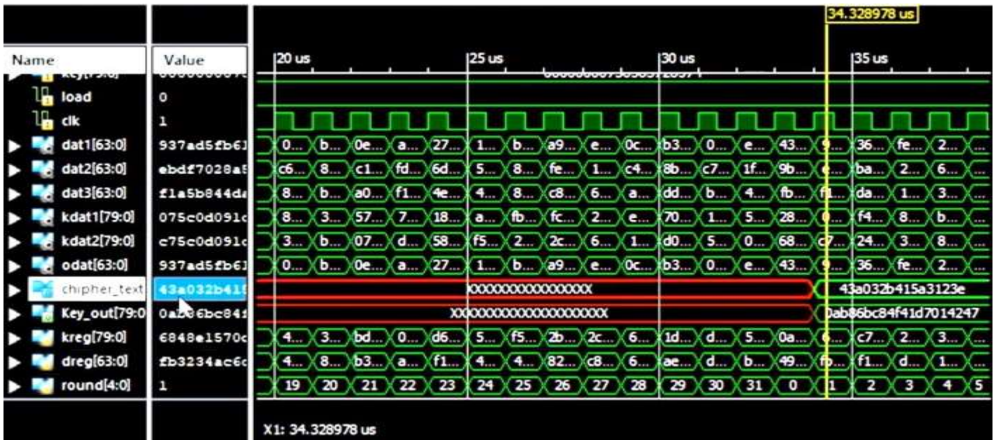


Figure 5. Successful decrypt example

5.1 Power Consumption Concerns

We first estimate the power after the synthesis of various ciphers in comparison to our lightweight PRESENT-80 cipher implementation. The estimated table below shows that our 80-bit block cipher has a higher throughput of 200Kbps amongst others.

Table 1. Key performance indicators of different cipher algorithms

	Key size	Block size	Cycles per block	Throughput at 100KHz (Kbps)	Logic process	Area GE	Area rel.
Block ciphers							
PRESENT-80	80	64	32	200	0.18μm	1570	1
AES-128 [16]	128	128	1032	12.4	0.35μm	3400	2.17
HIGHT [22]	128	64	34	188.2	0.25μm	3048	1.65
mCrypton [30]	96	64	13	492.3	0.13μm	2681	1.71
Camellia [1]	128	128	20	640	0.35μm	11350	7.23
DES [37]	56	64	144	44.4	0.18μm	2309	1.47
DESXL [37]	184	64	144	44.4	0.18μm	2168	1.38
Stream ciphers							
Trivium [18]	80	1	1	100	0.13μm	2599	1.66
Grain [18]	80	1	1	100	0.13μm	1294	0.82

After successfully completing the implementation stage, Vivado estimation of the power consumption was available. It was 2 orders of magnitude lower than the initial estimation (done based upon the synthesis stage) of more than 5 W of used power. This means that, in this way the device can work with 2AA batteries for around 9 months with the total on-chip power of 0.068W.

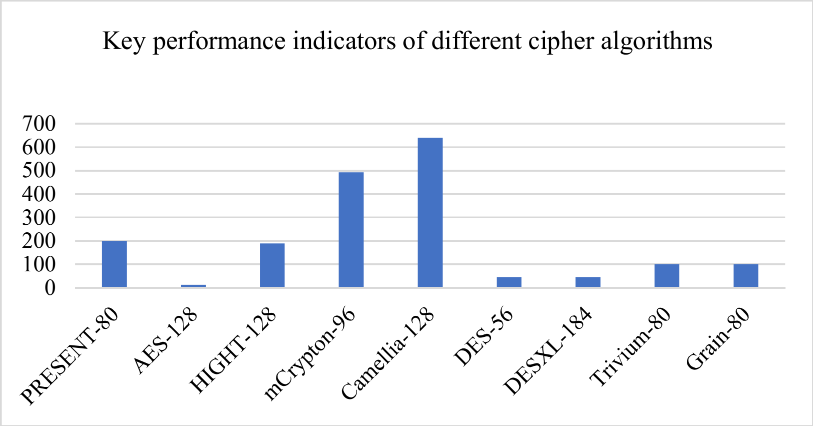
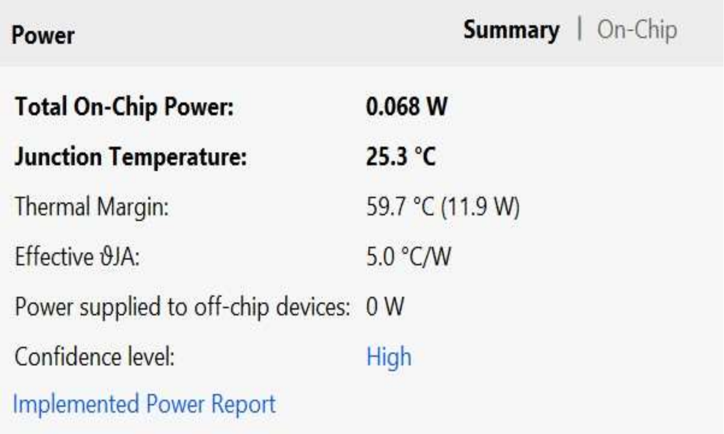


Figure 6. Key performance indicators of different cipher algorithms

Table 2. Vivado implementation stage estimates of a power usage



Power **Summary** | On-Chip

Total On-Chip Power:	**0.068 W**
Junction Temperature:	**25.3 °C**
Thermal Margin:	59.7 °C (11.9 W)
Effective ϑJA:	5.0 °C/W
Power supplied to off-chip devices:	0 W
Confidence level:	High

Implemented Power Report

6. CONCLUSION

Based on the CURRENT algorithm, this study puts forward and executes an entirely auto-logger security transmission protocol. It uses GPS and other sensors to gather pertinent logs, an ST32 single-chip microcomputer to encrypt the data by calling the preexisting rule set, and BC-module 28 to connect to the server. The utility layer, which serves the actual demands of the business, is constructed using cloud platform. The collected data is processed by the perception layer to implement the tasks of user control, instrument management, proximity monitoring, and abnormal alarm. The number of exchanges between the computer and human is determined by the present algorithm, and dynamic key updates are accomplished using the server. It is a special classifier that significantly improves data transfer security and produces fruitful results.

REFERENCES

[1] Bogdanov, A., Knudsen, L. R., Leander, G., Paar, C., Poschmann, A., Robshaw, M. J., ... & Vikkelsoe, C. (2007). PRESENT: An ultra-lightweight block cipher. In Cryptographic Hardware and Embedded

Systems-CHES 2007: 9th International Workshop, Vienna, Austria, September 10-13, 2007. Proceedings 9 (pp. 450-466). Springer Berlin Heidelberg.

[2] Jiang, L., Da Xu, L., Cai, H., Jiang, Z., Bu, F., & Xu, B. (2014). An IoT-oriented data storage framework in cloud computing platform. IEEE transactions on industrial informatics, 10(2), 1443-1451.

[3] Ray, P. P. (2016). A survey of IoT cloud platforms. Future Computing and Informatics Journal, 1(1-2), 35-46.

[4] Singh, S., Sharma, P. K., Moon, S. Y., & Park, J. H. (2017). Advanced lightweight encryption algorithms for IoT devices: survey, challenges and solutions. Journal of Ambient Intelligence and Humanized Computing, 1-18.

[5] Song, T., Li, R., Mei, B., Yu, J., Xing, X., & Cheng, X. (2017). A privacy preserving communication protocol for IoT applications in smart homes. IEEE Internet of Things Journal, 4(6), 1844-1852.

[6] Usman, M., Ahmed, I., Aslam, M. I., Khan, S., & Shah, U. A. (2017). SIT: a lightweight encryption algorithm for secure internet of things. arXiv preprint arXiv:1704.08688.

[7] Wu, W., & Zhang, L. (2011). LBlock: a lightweight block cipher. In Applied Cryptography and Network Security: 9th International Conference, ACNS 2011, Nerja, Spain, June 7-10, 2011. Proceedings 9 (pp. 327-344). Springer Berlin Heidelberg.