Interface for Data Protection and Integrity in IoT Equipment for Industry

N. Belu¹, L.M. Ionescu² and A.G. Mazare²
1 University of Pitesti, Manufacturing and Industrial Management Department, Targul din Vale Street No.1, Romania
2 University of Pitesti, Electronics, Computer and Electrical Engineering Department, Targul din Vale Street No.1, Romania

E-mail: nadia_belu2001@yahoo.com

Abstract. Presently, industrial processes are increasingly based on the equipment that communicates with each other. IoT is present in all branches of the manufacturing process, from the flows of the equipment that work together to achieve the finished product, to the equipment for checking the quality of the process, to the existing logistics infrastructure inside the factory, to the coordination and administration side. One of the problems raised by using IoT is that related to data security and integrity. We are talking about data protection for unauthorized access: maintaining confidentiality at the level of the different departments in the production flow, protecting the data provided by equipment and preventing the injection of unauthorized data flows and we are talking about ensuring data integrity: the data from the equipment must be resistant to any disturbances that occur in the industrial environment on the communication channels (wireless or wired), be assured the correctness of the transmission of the commands from the operator to the equipment. The paper proposes a specialized IoT interface that ensures the protection and integrity of the data transmitted between endpoints. The protection is provided by a light encryption algorithm based on two keys stored at the source and the recipient and on a very simple but efficient coding / decoding scheme. Data integrity is ensured by an 8-bit code word algorithm that allows for the detection of 2 wrong bits and the correction of a wrong bit for a 4-bit packet. All these features are provided by a computer system integrated in the IoT equipment. The efficiency of the solution is illustrated by a case study where in the manufacturing process of pistons it was desired to introduce specific elements Industry 4.0: communication between equipment in order to detect problems in the flow of manufacture (a Poka Yoke system). On the one hand, the system is presented without the protection solution in which there are several communication errors due to the disturbances induced on the communication channels on the other side to the system, the interface proposed in this article is added (at minimum cost) and the performances will be presented in the communication introduced by it.

1. Introduction
A specific feature of Industry 4.0 is the ability of systems to communicate with each other and to communicate with a central entity. Industrial networks connect sensors and actuators from various industrial installations. The Internet of Things (IoT) is a paradigm that involves a network of physical objects containing embedded technologies to collect data, communicate and interact with other systems or the external environment via wireless or wired connections providing information [1]. The paradigm
raises several problems regarding ensuring the protection and integrity of the data that appear in all communications networks. These problems are common to communication systems, but in this case, they have a particularity regarding the limited resources available to the IoT equipment and the limited power supply. The paper [2] presents a block chain solution combined with IoT in order to ensure data security and efficient management. The relationship between data processing and energy consumption to IoT is presented in the paper [3]. Here is proposed a lightweight encryption algorithm that generates less effort on the processor and therefore less power consumption. The solution proposed in our paper has another approach: the use of a dedicated, low power circuit, which takes over the entire encryption and data integrity task. The paper [4] also presents a solution to implement block chain on IoT that ensures data security. The scheme proposes a redesign of the computing effort to be more efficient on IoT equipment. The implemented solution is tested on a pilot with Raspberry PI embedded PC used in a section of a factory. The reduction of the effort on the equipment is done through a centralized architecture that consists of 3 layers: one composed of sensors, another composed of gateway and intermediate processors and the third composed of a central processor - solution proposed in the work [5]. The solution allows the connection of many sensors and big-data analysis.

In the area of error detection, it is work paper [6] in which a solution for error detection is presented, which also aims at reducing electricity consumption in order to be implemented on IoT devices used in the industrial area. The paper [7] analyzes implementation methods for more complex solutions that are based not only on detection / correction but also on prediction of error occurrence - solutions that are also implemented at the IoT level used in industrial environment.

The solution proposed in our paper ensures data protection and integrity. This is possible by implementing two modules: one that ensures data protection by encryption (to transmitter) / decryption (to receiver) and the second that ensures data integrity by encoding the message and by implementing a mechanism with error detection and error codes – coding to transmitter and decoding to receiver. Correction of 1 bits of error and detection of 2 bits of error is ensured. Both ideas are not new: at present there are distinct fields of research and practical solutions for data encryption and protection of their integrity. The novelty elements brought to this paper are the portability of the proposed solutions on IoT devices and the use of these solutions in the area of industrial applications. The first element of novelty will be highlighted by evaluating the FPGA resources used to implement the two modules. For data protection a symmetric key encryption algorithm inspired by the modern method of quantum photon encryption will be used. The solution uses a minimum of resources consisting of two synchronized pseudo-random sequence generators and 2 inverters. For the part of detection and correction errors will be used associative memories. The encoder will perform the 8-bit (encrypted) message conversion into the 12 bits code word. The decoder will retrieve the code word and through the associative memory it will convert it into the message. Further the message will be decrypted. So, for the two modules, it will work on minimal architectures, implementable on low-cost FPGA circuits.

The second element of novelty is the application of the methods proposed in the paper to ensure the security and integrity of data on IoT equipment used in industry. The proposed pilot is implemented on a system composed of current sensors used to determine the operating regime of industrial machines. As will be seen from the results presented from the implementation of this pilot, the encryption solution does not affect the speed of communication. In the error detection / correction solution the size of the code words is larger than the messages, but overall, the proposed solution has a very good message/code ratio compared to other methods. Therefore, we have an IoT system where the data are protected, and their identity is ensured in exchange for a low implementation cost and very good communication parameters.

2. Presentation of the solution
The solution is based on two main modules, one that provides data encryption and the other that provides encoding / decoding for error detection and correction. A block diagram of the system, to the transmission part, is shown in the figure 1.
Figure 1. Transmitter module block diagram.

The key is generated by a pseudo random sequence generator. It is synchronized with the decoder in the sense that the sequences from both are the same. The solution does not require the synchronization of clock signals between the encoder and the decoder but does not exclude it. The message bit is compared to the key, if they are the same one will be transmitted 1 if they are different it will be transmitted 0. The encrypted bits, thus resulting, are then assembled into message words (there are 8 bits in us). The message words are converted, through a coder, into code words. The code words are then transformed into bits that are transmitted on the communication channel.

At the reception we have the following block diagram, figure 2:

Figure 2. Receiver module block diagram.

The receiver module starts from the code word bits, received on the communication channel, which are immediately converted to message (parallel format). The code words are transformed into the message through the decoder. If at the encoder level we have a 1:1 message conversion - code word (implemented through conventional memory) at the decoder there is a possibility that the code word may not be identical to the expected one - an error has occurred on the communication channel. In this case, the message must be reconstructed from partially identical code words (similar) to the original ones. For this reason, an associative memory was used. At this level, a comparison of the identity of the entry is not made (as in conventional memories), but a comparison is made of the similarity of the entry with what is expected. Specifically, a comparison is made in the space of Hamming distances -
the number of different bits between the received and expected code word. If the received and expected code words are close (1 or 2 Hamming distance) then the message is reconstructed using the expected code word. In this mode, it is possible to make a correction/detection of 1/2 erroneous bits without requesting the retransmission of the message.

The code word is converted to serial format and then, bit by bit, it passes through the decryption module. Here is a circuit like the one from the transmission module. If the encrypted message bit is 1 then the key will be transmitted if not, then the reverse of the key will be transmitted. From a certain point of view, we do not have message bits on the communication channel but bits indicating whether the message is in the key. Therefore, we have stated that at the base of our communication encryption lies an algorithm inspired somewhat by the method of quantum encryption.

3. Implementation and analysis of resources
The proposed solutions are simple as architecture - this is a main objective proposed for our solution. Specifically, the proposed encryption-detection / error correction system is used for IoT systems. It will be attached to IoT equipment as an additional device. This means that the circuits used must be small and inexpensive.

We have implemented the proposed structure on a Xilinx Spartan 3 FPGA circuit from the family of widely used circuits on the market of applications with dedicated integrated circuits, which allows the implementation and testing of integrated hardware modules. The resources used are presented in the tables 1 and 2.

| Resource                  | Value | Occupation percent (XC3S400) |
|---------------------------|-------|-----------------------------|
| Number of Slices          | 68    | 1%                          |
| Number of Slice Flip Flops| 60    | 0%                          |
| Number of 4 input LUTs    | 115   | 1%                          |
| Number of IOs             | 3     | 1%                          |
| Number of GCLKs           | 1     | 12%                         |

| Resource                  | Value | Occupation percent (XC3S1000) |
|---------------------------|-------|-------------------------------|
| Number of Slices          | 4787  | 62%                           |
| Number of Slice Flip Flops| 72    | 0%                           |
| Number of 4 input LUTs    | 8299  | 54%                           |
| Number of IOs             | 3     | 0%                           |
| Number of GCLKs           | 1     | 12%                           |

As can be seen in the tables 1 and 2, the volume of resources consumed is a small one at the transmitter module. The tests were performed with an XC3S400 member of the Spartan 3 family, which, as can be seen in table 1, remains largely unused - so you can go with the smallest member, XC3S50, of the Spartan 3 family. This is of course also the cheapest (~ $ 10). On the other hand, at the receiver, the resources from the XC3S400 are not sufficient - the implementation was done on the XC3S1000. At its level, over 50% of the resources are used. The explanation comes from the implementation of the associative memory which consists of 256 cells, each having a register for the code word, a comparator and a minimum determination circuit starting from the Hamming distance between the stored code word and the received code word and the Hamming distance for the nearest code word stored at previous locations.
The analysis done at the Xilinx synthesis and implementation tool shows a propagation time ratio, presented in table 3.

**Table 3.** Delay time for transmitter and receiver modules.

| Module     | Minimum period | Total number of paths/destination ports |
|------------|----------------|----------------------------------------|
| Transmitter| 12.200ns       | 17990 / 100                            |
| Receiver   | 12.200ns       | 18001 / 111                            |

It can be seen in table 3 that the proposed system has a propagation time of ~ 12 ns at the transmitter and receiver which means it works at a frequency of 81,968 MHz (bps). The features in time are similar - the operating mode is similar even if the architectures differ. The 82Mbps frequency expresses the maximum transmission speed when encrypting data and adding additional sequences for error detection and correction.

The conclusion is that from the point of view of the resources used but also from the point of view of the performance (propagation time) our solution can be implemented on low cost FPGA circuits. A solution is given by the circuits in the Xilinx Spartan 3 family that are used in commercial applications. Therefore, at a low cost on equipment, data protection and integrity are ensured - important features especially in the industrial environment.

4. Testing in real environment, results obtained

The IoT system in which the encryption and errors detection/correction module was implemented is a system used to read the current (power) consumption values from the power cables of some electrical equipment for industrial use. The power consumption is an indicator of the operating regime of the equipment. For example, in an electric motor, the power consumption indicates the load the motor is pulling - whether it is higher or lower. At a welding station the power consumption indicates whether the operator is welding.

A block diagram of the implemented system is shown in the figure 3.

![Figure 3](image_url)

**Figure 3.** Transmission/receiver modules proposed in this paper are inserted between the data acquisition module from the sensors (coils) and the transceiver.

It can be seen in red the module proposed in this paper used for transmission and reception. It is placed between the data acquisition module from the sensors: in our case coils placed on isolated conductors carrying electricity and the wireless transceiver. So, the data from the sensors are retrieved, they are encrypted, they are encoded for detection / correction errors and then they reach the...
transceiver where they are then transmitted, in our case, over a wireless communication channel. This is the functionality provided at IoT equipment level (illustrated in figure 3 with Eq.IoT).

The receiver is a central server station that has a Gateway for connecting to the wireless IoT network (Gat.IoT). The gateway contains a transceiver - like those found in IoT equipment but which contains a network coordination part - and the transmission / reception module proposed in this paper. The receiver that will receive data from the transmit / receive module is the server station. In our case, the port to which the gateway is connected is a USB one.

The system is installed on 4 industrial equipment. It performs data protection and ensures data integrity for wireless (2.4GHz) communications. From the point of view of data protection, the analysis of the efficiency of the system implementation is done by analyzing the speed of the communication to the solution in which the module proposed by us is introduced and in case it would not be introduced. It is obvious that in the absence of the module and in the absence of other data encryption methods, the data will be exposed and can be captured / modified by an attacker. Along with the level of data protection, an important criterion is to change the speed of communication under the conditions in which encryption solutions are introduced. In our case, the encryption solution does not introduce any additional information transmitted on the communication channel - so it has no effect on decreasing system performance. Instead, the coding / decoding module for data integrity introduces additional bits in the code words, which have an impact on the speed of communication, as can be seen in the figure 4.
Figure 4. Results from 100 tests with 100 messages on 8 bits for each of 4 IoT equipment used for power consumption transmission. Comparative analysis for transmission with data protection module and without data protection module. The analysis was made for error rates of small (up to 10%), medium (10% -30%) and high (50%). (a) Data volume for 100 code words – detection/correction insert 4 additional bits to each message, (b) Left – data volume for 100 messages words without detection/correction, Right – comparisons between total data volume with (left) and without detection/correction. Error rate 0.1 / 8 bits message, (c) Left – data volume for 100 messages words without detection/correction, Right – comparisons between total data volume with (left) and without detection/correction. Error rate 0.3 / 8 bits message, (d) Left – data volume for 100 messages words without detection/correction, Right – comparisons between total data volume with (left) and without detection/correction. Error rate 0.5 / 8 bits message.

The analysis considered the transmission of 100 packets of data, with the size of 8 bits (so 256 symbols). For each rate of occurrence errors were performed 100 tests. At a low error rate (up to 10% an erroneous bit in an 8-bit package - the top of figure 4), a subunit ratio can be observed between the total of transmitted bits without correction algorithm and that of transmitted bits with correction. It is about adding a further 4 bits to each packet (useful message). This ratio, which is called communication efficiency or throughput, is 0.7309. It is normal for this report to be a subunit, which shows that the volume of data transmitted has increased (in our case by 27%) by introducing the additional bits for correction. However, there is a substantial improvement in response time. If in the
solution without correction of errors, the occurrence of an error implies, first of all, its detection at a higher level of application - so the implementation of control sums that have not been taken into account here (without these control sums the detection of errors is impossible). On the other hand, when detecting errors, it will be requested to retransmit the data packet - which implies a request retransmission sequence. The solution proposed by us solves the problem of detection and correction without retransmission. All this under the conditions where, as we have shown, data encryption also takes place.

Interestingly, at an average error rate (20% - 30% error rate on an 8-bit packet) The communication efficiency increased: 0.8654. So, the total volume of data increased by 14% by adding correction bits. The explanation comes from the fact that more errors occur and the retransmission, if there is no local correction algorithm, adds an additional volume of data. It should be noted, again, that only the volume of data involved in the retransmission was considered in the event of errors occurring without the necessary data for checks or the request for retransmission.

At a high error rate (50% an erroneous bit in an 8-bit package), the rate reached quite often in the very noisy industrial environment, the efficiency of communication (the total volume of data in the solution without correction compared to the volume of data at the corrected solution) is 1.0023. So, the total volume of data under the conditions in which each package is added 4 bits for detection and correction is smaller than the one without correction - the cause is the size of the retransmitted packets. By adding the extra bits in the code words, a self-correction is made with a significant impact on the communication efficiency. The volume of data conveyed on the communication channel are almost like a communication algorithm with the request to relay data to the proposed solution instead the proposed solution has a shorter response time (through local auto-correction). All this are provided with the data protection to unauthorized access.

5. Conclusions
The paper presents a solution that ensures the integrity and security of data from IoT equipment. On the one hand, the solution allows the protection of unauthorized access data - a typical problem for IoT equipment, on which less attention was paid to communication security and, by the way they are implemented, they do not allow the application of algorithms very sophisticated encryption.

On the other hand, in the industrial environment there are sources of multiple electromagnetic disturbances, coming from the industrial electrical equipment that work at high voltages and currents. These affect communications by introducing errors. The proposed solution solves the problem of data integrity by applying an error detection / correction method to the message packets.

The novelty elements brought by the work are: they provide a practical solution, implementable to IoT equipment, which involves the use of reduced resources so that they can be implemented on small circuits in terms of size and price; the solution is implemented and tested for IoT equipment used in the industrial environment - in this respect a pilot has been made for 4 working points consisting of 2 welding points and 2 assembly and packaging. The solution has proved its efficiency, at a reduced price, it guarantees the protection and integrity of the data, provided that the speed of communication and the efficiency of the communication are kept in parameters comparable to those of the system without protection.

As future directions of development, starting from this paper, we have the implementation of the system on several industrial equipment to prove its efficiency under the conditions in which the security and integrity of the data are ensured.

6. References
[1] Gidlund M, Han S, Sisinni E and Saifullah A, 2018 Guest editorial from industrial wireless sensor networks to industrial internet of things, *IEEE Transactions on Industrial Informatics* 14(5) 2194-2198
[2] Seok B, Park J and Park JH 2019 A Lightweight Hash-Based Blockchain Architecture for Industrial IoT, *Appl. Sci.* 9(18) p. 3740
[3] Zhou L, Wu D, Chen J and Dong Z 2018 When computation hugs intelligence: Content-aware data processing for industrial IoT IEEE Internet of Things Journal 5(3) 1657-1666
[4] Huang J, Kong L, Chen G and Wu MY 2019 Towards secure industrial IoT: Blockchain system with credit-based consensus IEEE Transactions on Industrial Informatics 15(6) 3680-3689
[5] Rehman MH, Ahmed E and Yaqoo I 2018 Big data analytics in industrial IoT using a concentric computing model IEEE Communications Magazine 56(2) 37-43
[6] Gao XC, Zhang JK, Chen H and Dong Z 2019 Energy-efficient and low-latency massive simo using noncoherent ml detection for industrial IoT communications IEEE Internet of Things Journal 6(4) 6247-6261
[7] Wu D, Jiang Z, Xie X, Wei X and Yu W 2019 LSTM Learning with Bayesian and Gaussian Processing for Anomaly Detection in Industrial IoT IEEE Transactions on Industrial Informatics p. 1