Hindawi
Journal of Sensors
Volume 2019, Article ID 7810709, 10 pages
https://doi.org/10.1155/2019/7810709

Research Article

Implementation of Secure Communication via the RF Module for Data Acquisition

Juraj Dudak, Gabriel Gaspar, and Pavol Tanuska

Faculty of Materials Science and Technology in Trnava, Slovak University of Technology in Bratislava, Slovakia

Correspondence should be addressed to Juraj Dudak; jdudak@gmail.com

Received 28 January 2019; Revised 3 May 2019; Accepted 20 May 2019; Published 11 June 2019

Academic Editor: Mohammad Haider

Wireless monitoring systems are currently a common part of both industrial and domestic solutions. In case of industrial use, it may be a sensory system at a point where it is not possible to implement a fixed line, or it is necessary to repeatedly change the position, respectively, the location of the measuring probes. This is, in particular, the monitoring of environmental parameters (temperature, humidity) in bulk materials, biomass, or agricultural crops in storage facilities. For domestic use, local weather devices and systems are part of smart households such as lock control (entrance gate, apartment entrance), light control, and HVAC control. Devices that utilize wireless technology for communication include IoT devices as well. This article describes the use of the VirtualWire communication protocol using radio frequency waves with a carrier frequency of 433MHz-900MHz. The main topic of the article is the design and implementation of a secure one-way communication channel. Such a solution consists of a transmitting device with implemented desired sensors and a receiving device that aggregates data from multiple transmitters.

1. Introduction

The use of wireless radio link is advantageous when communication is required in an area where fixed bus networks cannot be used. In particular, free RF bands 433MHz, 866MHz, and 915MHz are used for communication. The most commonly used home appliance devices are various types of remote controls, as well as sensor providing data for smart-home solutions, monitoring air temperature, relative humidity, atmospheric pressure, and others. The data transmission is always one-way, i.e., from the sensor probe with transmitter to the receiving node. Since such a data transfer is in the free bandwidth, it is not a problem to acquire and read sent data. Possible attack on such a data transfer is trivial; data can be easily captured, modified, and resent. It is not possible to verify the origin of such data on the receiver node side. The use of RF communication is particularly advantageous where the use of wired communication buses would represent an increased financial cost of the application; the use of standard metallic wiring would be impractical or even impossible. An example of where RF communication is preferred is the monitoring of environmental parameters (temperature, humidity, light intensity, and atmospheric pressure) in larger areas (interior or exterior). The cost of wiring for large spaces often simply outweighs the cost of a sensory system solution. There are also cases where wiring is not possible, for example, when installing a sensory system directly into large volumes of material that is often handled or moved. The RF probes described in this article are designed as nonvolatile modules in a 25mm diameter and 135mm cylinder height cylindrical PET package. Such a probe can be used to monitor the temperature of materials whose internal temperature must be controlled. An alternative to the RF communication principle is utilization of Bluetooth technology. Compared to 433MHz RF technology, Bluetooth is significantly less usable in cases where the signal has to pass through obstacles such as walls and metal structures or if the sensor is placed in bulk materials deeper below its surface.

This article discusses the design and implementation of a RF communication system with secured transmission. Next the basics, design, and implementation of the transmitting and receiving part of a wireless monitoring system built on the STM32 platform with a communication module with a carrier frequency of 433.92MHz will be presented. When designing the broadcasting part, great emphasis was placed on the universality of use, the energy demandingness, and the reliability of the data transmission. The next section describes the implementation of secure communication layer for usage in embedded systems. Then, performance tests will be performed on real hardware, 32-bit ARM microcontrollers.

2. Related Works

One of the areas where communicating nodes represent single-purpose devices is the Internet-of-Things (IoT) world. Data from many sensors are sent over several network types until they arrive to the data repository, e.g., to the cloud. Therefore, it is in place to secure this communication at the point where the information is generated, in the sensors themselves. Therefore, it is appropriate to secure this communication at the point where the information is generated, in the sensors themselves. The issue of communication security in IoT networks is described in [2], where authors propose an algorithm of encryption keys exchange for smart-home systems. The enhancement of security of IoT smart devices is described in [3]. Another important category is systems that collect sensitive and private information. These are, for example, data from sensors monitoring the vital parameters of the human body. The application of the SPECK/SIMON cipher is suggested in [4], where the authors proposed faster algorithms implementation using the architecture of the 64-bit Intel AVX2 microcontroller. Comparison of the suitability of the use of cryptography algorithms in the area of healthcare is in [5]. The SPECK/SIMON encryption algorithm is covered in several publications. The original proposal is described in [6]. Implementation of these ciphers into a hardware solution is in [7]. Cryptanalysis of the cipher is in [8, 9] (differential analysis) and [10], where authors describe a cipher attack by electromagnetic analysis. The problem of using a suitable encryption algorithm for different applications, respectively, the question "What is the cost of encryption in terms of file size after performing compression?" is presented in [11]. The solution presented in this article does not fall directly into the IoT category; it is a one-way communication system based on RF, where the data is sent to a centralized data warehouse. However, the security issue of this type of communication is current.

3. RF Communication System

The primary motivation for this article was the design of a wireless communication system for monitoring environmental parameters in bulk materials, namely, wood chip. This is a situation where it is not possible to use sensors connected to a fixed bus line. The architecture of the wireless RF monitoring system is in Figure 1 [12].

In the proposed monitoring system, n environmental sensors are located in the monitored area. As the monitored area may be larger than the range of RF transmitting modules, the system contains m receiving modules. Each receiving module is capable of receiving data from a set of sensors, while these sets may overlap. For the implementation of wireless communication, technology was chosen that allows communication at a distance of min. 50m and automatic reception of data from transmitter probes without the need for further configuration. Using WiFi or Bluetooth communication would require the need to connect the transmitters to the wireless network. The use of RF signal in the free RF band 433.19MHz appears to be optimal for use in terms of implementation and energy-saving.

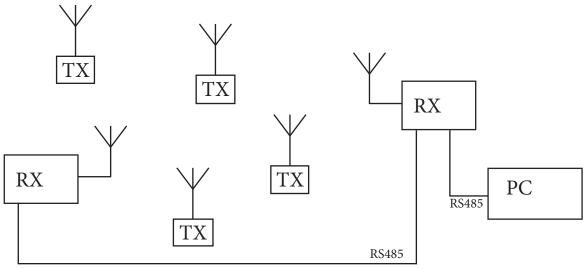

Figure 1: Architecture of the RF monitoring system.

When analyzing the problem, the requirements for the entire monitoring system were defined as follows:

(i) temperature measurement in the range -10C to 90C with accuracy of +/-0.5C and resolution of 0.1C,

(ii) autonomous operation of measuring probes for at least three months,

(iii) probe design resistant to the pressure of stored material in which the probe is placed and against mechanical stress caused by mechanisms used in processing and material handling,

(iv) operation of probes in the free RF band 433.92MHz,

(v) one-way communication protocol which allows detection of a duplicate multiprobe transmission on the receiver side,

(vi) parameterized probe transmission frequency depending on the temperature of the material,

(vii) reception of a signal covering the entire storage site, secured by multiple receivers with duplicate measured data filtering,

(viii) data model of the system—internal model of the measurement system maintaining information on the hardware configuration, active sensors, their physical location, measured data, and other supplementary information,

(ix) secure data transmission from probes to receiving modules, secure communication from the receiving modules and the control computer, and secure communication between the client software and the control computer,

(x) platform independent, simple client software,

(xi) remote access to data.

The hardware part of the monitoring system consists of the required number of measurement probes transmitting the measured data and one or more receiving modules covering the monitored site.

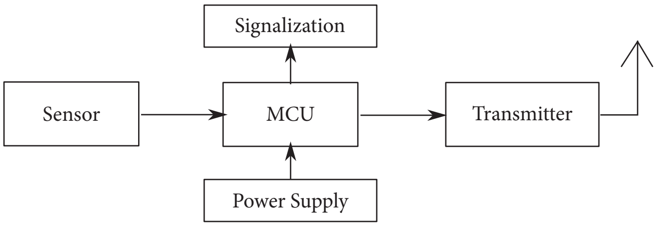

Figure 2: Block diagram of the measuring probe module.

36-bit	12-bit			
PREAMBLE	START	LEN	DATA_VW	FCS
		1B	1-27B	2B

Figure 3: Block diagram of the measuring probe module.

3.1. Measuring Probe. The task of the measuring probe is measuring the required physical quantity (temperature, humidity, pressure, etc.) at regular intervals and sending this value over the RF interface. An autonomous operation is assumed, so for this module it is necessary to contain a battery that will provide power to the entire solution. Along with the measured values, battery voltage information will also be sent over RF link.

The measuring probe module (Figure 2) is designed to measure environmental parameters such as temperature, humidity, atmospheric pressure, solar radiation intensity, and the like. For biomass temperature monitoring purposes was developed a variant containing a temperature sensor. The purpose of the measurement probe is to process the measured value, prepare telemetry data, format the data according to the entered key, and send the processed data via the RF interface. Ultra Low Power (ULP) components will be used to ensure minimum power consumption.

Measuring probe consists of the following components:

(i) control microcontroller STM32F030F4T6 with 16kB FLASH memory for the program and 4kB for data, respectively

(ii) internal A/D converter coupled through a resistive divider allowing telemetry to monitor the voltage of the battery power supply of the probe

(iii) LM75AD temperature sensor with a resolution of 0.125C and operating range of -25C to 100C connected to the control microprocessor via the I2C bus

(iv) signal LEDs indicating the state of measurement and data transmission

(v) a stabilized power supply block consisting of one AA battery and a step-up inverter with 85% efficiency and current consumption down to 30uA

(vi) transmitter, for 433MHz band with OOK modulation

As a communication channel a radio transmission was selected with a carrier frequency of 433.199MHz and OOK modulation. On-off keying (OOK) denotes the simplest form of amplitude-shift keying (ASK) modulation that represents digital data at the presence or absence of a carrier wave. In its simplest form, the presence of a carrier for a specific duration represents a binary one, while its absence for the same duration represents a binary zero [13].

3.2. Link Layer Communication Protocol. As the lowest-level communication protocol VirtualWire was selected, respectively, RH_ASK. VirtualWire is an Arduino library that provides features to send short messages, without addressing, retransmit, or acknowledgment, a bit like UDP over wireless, using OOK, supporting a number of inexpensive radio transmitters and receivers. All that is required is to transmit data, receive data, and (for transmitters, optionally) enable PTT transmitter [14]. The maximum length of the data part is defined by VW_MAX_PAYLOAD (27 bytes). Each byte in the sent packet is encoded using 4-to-6 encoding for two 6-bit words. The purpose of 4-to-6 encoding is to achieve the state in which the same number of ones and zeroes will be transmitted in the broadcast. The format of the data packet is shown in Figure 3.

The data packet parts are:

(i) PREAMBLE - 36-bit training preamble consisting of 0-1 bit pairs

(ii) START - 12-bit start symbol 0xB38

(iii) LEN - 1 byte of message length byte count (4 to 30), count includes byte count and FCS bytes

(iv) DATA_VW - n message bytes, maximum n is VW_MAX_PAYLOAD (27)

(v) FCS - 2 bytes of Frame Check Sequence

The measured sensor data is transmitted in the data portion of the packet. The structure of the data part is described in the following section.

3.3. Application Communication Protocol - nSoric RF Packet. The proposed monitoring system will work with one-way communication. For the data packet, the following requirements were defined:

(i) sufficiently large address space for sensors

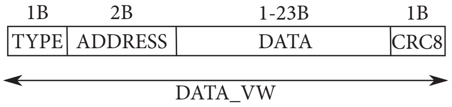

Figure 4: Application packet format.

(ii) variable packet items

(iii) content of the data part: telemetry data and data from connected sensors

(iv) packet check

(v) minimalistic demands for overhead communication

Application packet format, Figure 4: The address space for sensor probes was defined in the width of 2^{16} of unique addresses. To describe the structure and format of the sent data, 1Byte (TYPE) was reserved.

The packet data has a maximum length of 23B. The last item is the CRC8 checksum using the $X^8 + X^5 + X^4 + 1$ polynomial, which is used in 1-wire standard [15].

TYPE. The first byte in the packet (Table 1) defines the format of the entire packet, as well as the list of quantities contained in the packet.

TYPE byte is divided into two parts: content and version. The three bits (R1R2R3) defines a revision, respectively, meaning of content. The individual bits Vi indicate whether there is a part in the packet that contains the measured value of a defined quantity. Data frame format breakdown by revision (R1R2R3) is as follows:

(i) 000 environmental quantities

(ii) 001 electrical quantities

(iii) 010 universal A/D converter

Revision 000. For the probe measuring environmental variables, revision value (0,0,0) was defined. For this version, content part has the following meaning:

(i) V1: Temperature - 2B, range: -127 . . .+127, resolution 0.0125, unit C

(ii) V2: humidity - 1B, range 0 100, resolution 0.5, unit

(iii) V3: atmospheric pressure - 2B, range 0.. 8192, resolution 0.25, unit hPa

(iv) V4: light intensity - 2B, range 0.. 32768, resolution 0.5, unit lx

(v) V5: UV index - 1B, 0 32, resolution 0.25

Revision 001. Data frame format for the measuring electrical quantities is as follows:

(i) V1V2 - number of sets of measurements

(ii) V3 - electric resistance, 2B

(iii) V4 - electric voltage, 2B

(iv) V5 - electric current, 2B

Revision 010. Data frame format for values from n-channel analog-to-digital converter is as follows:

(i) V1V2 - number of converter bytes

(ii) V3V4V5 - number of converter channels

ADDRESS. 2 bytes is reserved for addressing. The address space is from addresses 1 to 65535. There are no additional restrictions for the addresses defined.

DATA. The length of DATA part is given by the byte TYPE, in which the DATA content is defined. The first byte in the DATA section is always the telemetry data, respectively, battery level. All other values are optional. In Table 2 are examples of the application frame for different revisions.

Table 2(a) example 1: Revision 000, data frame contains all values TELE - telemetry, TEMP - temperature, HUM - relative air humidity, PRES - atmospheric pressure, LIGHT - light intensity, and UV - UV index.

Table 2(b) example 2: Revision 000, content value in the TYPE configuration byte is 101000. DATA will contain temperature data (TEMP) and atmospheric pressure (PRES) only.

Table 2(c): Revision 001, data frame for electrical quantities values: The number of sets of measurements is given by V1V2 = 01; in the data part there will be 2 series of measurements. The measured values are determined by the triplet V3V4V5. Value 110 means that each set will contain the value of electrical resistance and electric voltage.

Table 2(d): Revision 010; data frame for AD converter values: The word width of the AD converter is defined by V1V2 = 10. The width of the word will then be 3B = 24 bits. The number of channels of the converter is determined by the triplet V3V4V5=001. Thus, there will be 2 values in the data section, each with the width of 3B.

4. RF Transmission Security

When using RF communication in the free bandwidth, the weakest point is the unencrypted transmission of measured data from the probes transmitter to the receiver. Such transmission may be acquired, modified, and further transmitted, which represents a security risk. Since it is a one- way transfer, it is not possible to exchange the encryption keys before the communication itself. To implement secure transmission, the following requirements were defined:

(1) utilization of a symmetric cipher

(2) computational light-weight encryption algorithm

(3) encryption algorithm suitable for small source message lengths

(4) use of multiple encryption keys

To implement encryption into measuring probes, the cryptographic algorithm needs to be simple because it is necessary to conserve electrical energy when it comes to autonomous measuring probes. The methods of reducing power consumption are lowering the core frequency of the microcontroller, use of microcontroller sleep modes, and minimization of the time when the microcontroller works in RUN mode.

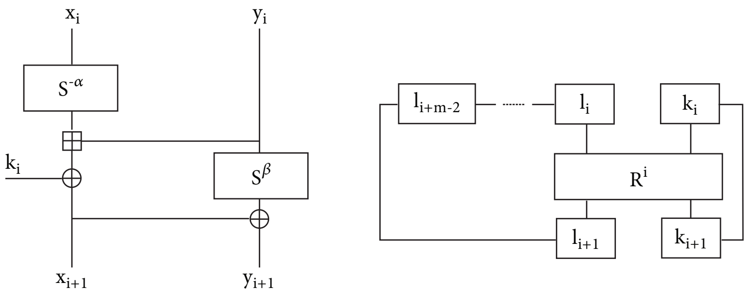

Figure 5: The round function and the key schedule of SPECK [1].

Another important parameter when selecting a suitable encryption algorithm is the length of the message that is being sent. The resulting encrypted frame must not exceed 27B. The data frame used (DATA_VW) is in Figure 4. The maximum frame length is 27B, of which 4B is the overhead and 23B is the data itself. The minimum possible frame length is 5B: 4B overhead and 1B telemetry data. When using a probe with a temperature sensor, the communication frame will have a length of 7B. An important parameter in choosing encryption algorithm is the length of the encrypted block and the key size. Due to the minimum and maximum length of the communication frame, the block length 8B is suitable. For this length, the data frame may have lengths 8B, 16B, and 24B.

When implementing the cryptographic algorithm for data acquisition systems [16, 17], the means of communication must be taken into account. In a one-way communication, the use of symmetric encryption is the optimal choice. Based on the analysis of the efficiency and complexity of the block cipher algorithms, we chose the SPECK algorithm.

4.1. Block Cipher SPECK and SIMON. With the upcoming era of Internet of Things and the Pervasive Computing, there is a need to develop block ciphers with tight constraints such as area, power, memory, performance, throughput, and others. These are so called the lightweight block ciphers which are specifically intended for resource constrained platforms. Lined up in the line is SIMON, a lightweight block cipher proposed by NSA after the prompting from the U.S. Government in the year 2013 along with SPECK lightweight block cipher. SIMON implementation on hardware has excellent results in terms of area and has been found to be a very strong alternative to the existing AES [18]. SPECK supports a variety of block and key sizes. A block is always two words, but the words may be 16, 24, 32, 48, or 64 bits in size. The corresponding key is 2, 3, or 4 words. The round function consists of two rotations, adding the right word to the left word, xoring the key into the left word, and then xoring the left word to the right word. The number of rounds depends on the parameters selected, as shown in Table 3 [1].

SPECK has been optimized for performance in software implementations, while its sister algorithm, Simon, has been optimized for hardware implementations. SPECK is an add-rotate-xor (ARX) cipher. The SPECK2n encryption maps a plaintext of two n-bit words (x0, y0) into a ciphertext (xT, yT), using a sequence of T rounds. The key-dependent round function is defined as [1]

Table 1: The meaning of the bits in the TYPE byte.

Byte	0							
Group	Content					Revision		
Bit	7	6	5	4	3	2	1	0
Info	V_1	V_2	V_3	V_4	V_5	R_1	R_2	R_3

$$R_k\left(x,y\right) = \left(\left(S^{-\alpha}x + y\right) \oplus k, \left(S^{\beta}x \oplus y\right) \oplus k\right) \tag{1}$$

where k is the round key and rotation constants α and β are given in Table 3. Used operations are

(i) $\oplus$ bitwise XOR

(ii) left circular shift, S^j, by j bits

(iii) right circular shift, S^{-j}, by j bits

The decryption rule is

$$R_k^{-1}\left(x,y\right) = \left(\left(S^{\alpha}x \oplus k\right) - y, \left(S^{-\beta}x \oplus y\right)\right) \tag{2}$$

The SPECK key schedule reuses the round function to generate the round keys $k_0, \ldots, k_T$. The m-word master key $K = (l_{m-2}, .., l_0, k_0)$ is used as follows [8]:

$$l_{i+m-1} = \left(k_i + S^{-\alpha}l_i\right) \oplus i \tag{3}$$

$$k_{i+1} = S^{\beta}k_i \oplus l_{i+m+1} \tag{4}$$

Figure 5 provides a schematic view on the round function and the key schedule of SPECK. Ri is the SPECK round function with i acting as the round key.

Differential cryptoanalysis can break 25 of 34 rounds of Speck128/256 with $2^{253.35}$ time complexity using $2^{125.35}$ chosen plaintexts and $2^2$2 bytes memory or 23 of 32 rounds of Speck128/128 with 2125.35 time complexity and $2^{125.35}$ chosen plaintexts [8]. Distinguishers for reduced-round versions of Speck32,48,64 have been found by automated means [19], and it is suspected that the same would happen to Speck128/256, given more computer power. According to European Network of Excellence in Cryptology stream cipher benchmarks (eBASC) [20], SPECK is one of the fastest ciphers available, both for long and short messages, and is comparable in

Table 2: Application frame examples.

(a) Revision = 000, example 1

Byte	0	1..2	3	4..5	6	7..8	9..10	11	12
Value	11111**000**	ADRR	TELE	TEMP	HUM	PRES	LIGHT	UV	CRC8

(b) Revision = 000, example 2

Byte	0	1..2	3	4..5	6..7	8
Value	10100**000**	ADRR	TELE	TEMP	PRES	CRC8

(c) Revision = 001

Byte	0	1..2	3	4..5	6..7	8..9	10..11	12
Value	01110**001**	ADRR	TELE	R1	U1	R2	U2	CRC8

(d) Revision = 010

Byte	0	1..2	3	4..6	7..9	10
Value	10001**010**	ADRR	TELE	ADC1	ADC2	CRC8

Table 3: SPECK algorithm variants of block size and key size.

Variant	Block size (bits)	Key size (bits)	Rounds	(α, β)
1	2x16 = 32	4x16 = 64	22	(2,7)
2	2x24 = 48	3x24 = 72	22	(3,8)
3		4x24 = 96	23	(3,8)
4	2x32 = 64	3x32 = 96	26	(3,8)
5		4x32 = 128	27	(3,8)
6	2x48 = 96	2x48 = 96	28	(3,8)
7		3x48 = 144	29	(3,8)
8	2x64 = 128	2x64 = 128	32	(3,8)
9		3x64 = 192	33	(3,8)
10		4x64 = 256	34	(3,8)

speed to the stream cipher Salsa20. When implemented on 8-bit AVR microcontroller, SPECK encryption with 64b blocks and 128b key consumes 192 bytes of Flash; temporary variables consume 112 bytes of RAM and take 164 cycles to encrypt each byte in the block [21].

4.2. Implementation of Cryptographic Algorithms. For the implementation of the cryptographic algorithm, we chose the ARM platform, namely, the 32-bit STM32L0xx microcontroller. This microcontroller includes a 32-bit CORTEX M0+ core; the core clock frequency is from 65kHz to 80MHz. The main reason for choosing this microcontroller was the ability to set the clock frequency, core architecture, and the fact that microcontrollers of this class are very often used for embedded solutions due to their low power consumption. The Cortex-M0+ processor is built on a high area and power optimized 32-bit processor core, with a 2-stage pipeline von Neumann architecture. The processor delivers exceptional energy efficiency through a small but powerful instruction set and extensively optimized design, providing high-end processing hardware. On the basis of relations 1 to 4, a library in the C language for the implementation of the SPECK cipher was created. From the available variants of the key length and the length of the encrypted block of the SPECK, variants were selected so that the means of architecture of the microcontroller used were effectively used in their implementation. The STM32L0xx family of microcontrollers are 32-bit microcontrollers; native word length is 32 bits. Variant 1 and Variant 5 were selected (Table 3). Variant 5 uses 128-bit (4 words) key length and 64-bit (2 words) block size, therefore 8 bytes. 8-byte size is sufficient for a packet that contains measured temperature and humidity data (see Section 3.3). For packets longer than 8 bytes, Variant 5 can be applied for each block of 8 bytes independently.

4.2.1. Implementation of Secure Transmission in the Communication Protocol. The secure communication layer using the SPECK cipher was implemented into the RF communication system described in Section 3. Before the implementation itself, the requirements for the transmitting and receiving sides were defined.

(A) Transmitting Side

(1) It will contain a set of nonrepeating encryption keys.

(2) When sending data, encryption key different from the previous transmission is used.

(3) Message encryption time must be minimal.

(B) Receiving Side

(1) The receiving side does not know the encryption key but contains the set of keys from which the particular encryption key was selected.

(2) Based on the known format of the received decrypted message, it will be able to find the correct decryption key.

When implementing the cipher on the ARM platform the gcc compiler in version 7 was used. After SPECK cipher implementation, the size of occupied FLASH memory increased by 1304 bytes compared to the version without encryption, of which there were 1024 bytes for 252 encryption keys and 280 bytes for the algorithm itself. RAM occupancy increased by 104 bytes.

Transmitting Side. The transmitting device will contain a set of keys, of which one key will be selected by a quasirandom selection. In order to minimize the use of the limited memory capacity of the microcontroller, a key selection procedure has been proposed. This applies to Variant 5 (Table 3). For Variant 1, it will vary by word length: instead of 32-bit values, 16-bit values are used, and the key size will be a pair of subkeys.

Encryption Key Selection Algorithm

(1) n random 32-bit numbers are generated and stored in the array with size n. This is a set of partial keys: $K = k_1 k_2 k_3 \ldots k_n$

(2) The random number j is selected; $0 <= j <= n - 4$

(3) The key used for the encryption will be $K_j = k_j k_{j+1} k_{j+2} k_{j+3}$

With the set size n = 256, we get 252 keys, each with a 128-bit size. The memory size that will be used to store these keys is 256 words = 1024 bytes.

Another requirement for the transmitting part was changing the encryption key in subsequent communication. Due to the nature of the original application, it is not possible to simply move the key to the next position in the K set. The RF sensor solution was designed with significant energy savings when the probe is inactive. The microcontroller switches to the STOP mode after sending the measured data, when all parts including FLASH and RAM are disconnected from the supply voltage. Only real-time clock remains active to ensure the wake-up of the microcontroller. After the microcontroller passes from STOP to RUN mode, all data stored in the RAM is lost. Therefore, it is impossible to remember the order of the last used encryption key. Encryption key selection is described in relationship (5).

$$index\,(key) = \sigma\,(P) \bmod \left(N - |key|\right) \tag{5}$$

where $\sigma(P)$ is a function that based on the contents of the ready-to-send communication frame calculates the key index to be used for encryption. N is the number of keys in the set K and —key— is the key size in the blocks. The last criterion was minimizing the encrypted message computation time. The microcontroller is in the RUN mode clocked at 8MHz. At this frequency, time required to apply the SPECK cipher on an 8-byte communication frame is 55us. For comparison, the calculation of the CRC (Table 1) for the same communication frame lasts 60us. Compared to the summary time since the microcontroller switches to RUN state, over to the sensor measurement to RF transmission, the time needed to encrypt the message is negligible. When using the LM75 temperature sensor, the time slots are the following: time needed to initiate and get the sensor value $t_{init} = 200ms$ and the time to send the measured value over the RF interface at a baud rate of 1000 baud: $t_{transmit} = 30ms$.

Receiving Side. The receiving part uses the same microcontroller. Due to the nature of the application, only RUN mode will be used; there is no need to switch the receiving module into low power mode. In order to be able to decrypt the received message, a decryption key must be available. However, according to the transmitting module specification, one encryption key from the set K (5) is randomly selected for transmission. The receiving module must know this set of keys and find a key that decrypts the frame correctly. The algorithm was designed to determine the decryption key and decrypt the received packet, Figure 6.

Figure 6 uses the labeling: P' - encrypted data packet, P - decrypted data packet, and K - set of available keys. The SPECK_setup function prepares the sequence of the partial keys used for the decryption according to the selected encryption key. When determining the correct decryption key, the following conditions must be met:

(1) CRC8 of the received frame must be the same as the last byte in the received frame

(2) The length and format of the data part must correspond to the definition of the data part contained in the first frame of the communication frame

For the N key set, it is necessary to run a decryption function maximum N-times and check the validity of the decrypted frame. When using the STM32L053 microcontroller with a maximum clock frequency 32MHz, the length of the encoded word 8 bytes takes the calculation of key schedule value 131us and decryption itself takes 19us. At the maximum number of 252 iterations is the decryption boundary time 44.6ms.

In order to select encryption key from the K set, it is necessary to ensure that the algorithm (5) has a distribution similar to a probability distribution. The test communication framework, which included telemetry data (battery voltage level in the range of 0.8V - 1.4V) and measured quantity (temperature in the range of 15°C - 30°C), was used to test the algorithm. Figure 7 illustrates the likelihood of selecting the encryption key index in 768 different communication packets according to algorithm (5).

5. Conclusions

This article presented an RF communication system designed for collecting data from environmental sensors. The size of the communication packet ranges from 6B to 16B. The secure communication layer using symmetric encryption was implemented into the existing communication system.

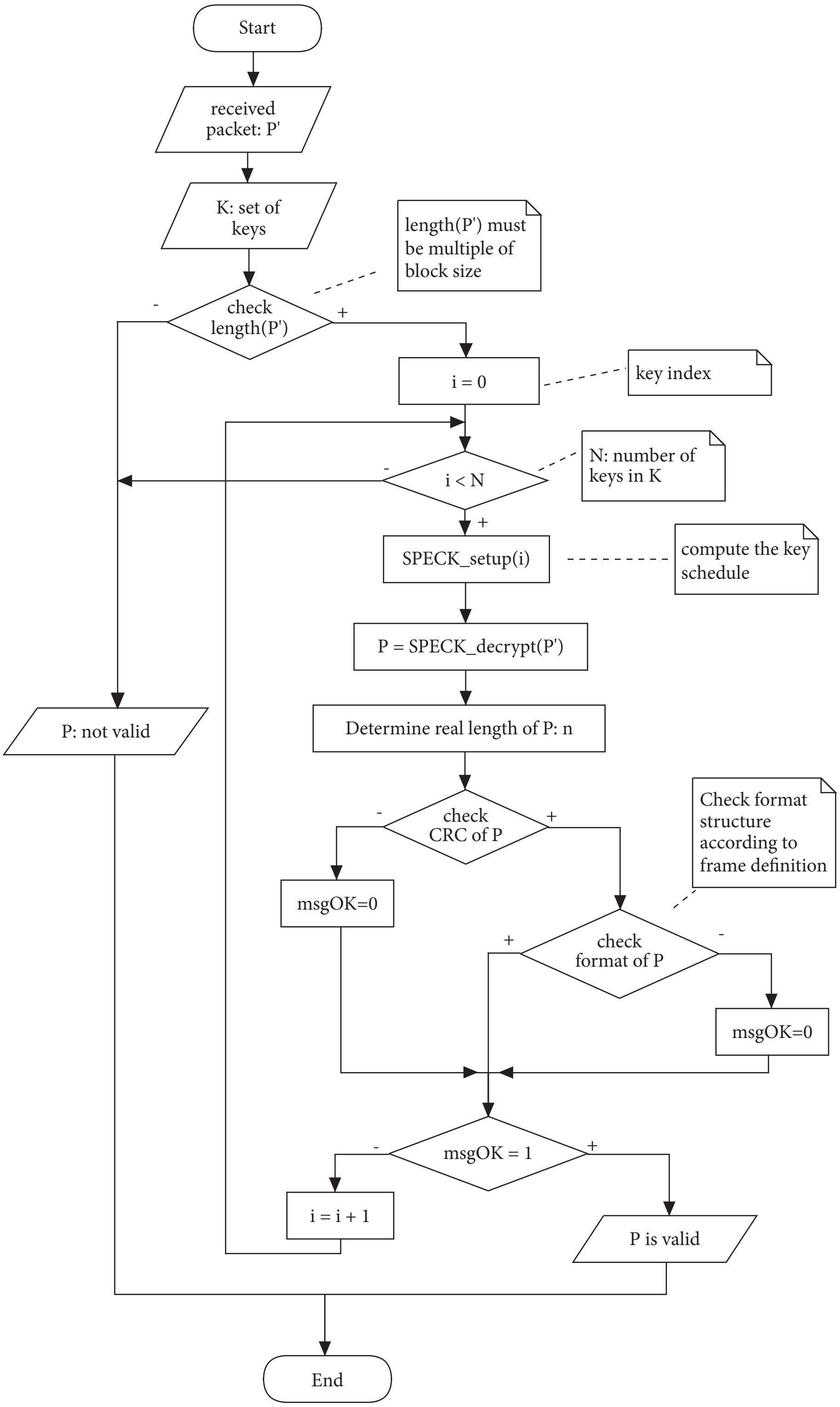

Figure 6: Algorithm for determining the decryption key.

When selecting a suitable encryption algorithm, the size of the original data packets and the maximum size of the data portion in the VirtualWire communication protocol, where the maximum length of the useful part is 27B, were taken into account. Also, consideration was given to the size of the data block that is the output of the encryption algorithm and the size of the encryption key. Based on these criteria, a SPECK encryption algorithm with a 64-bit block size and a

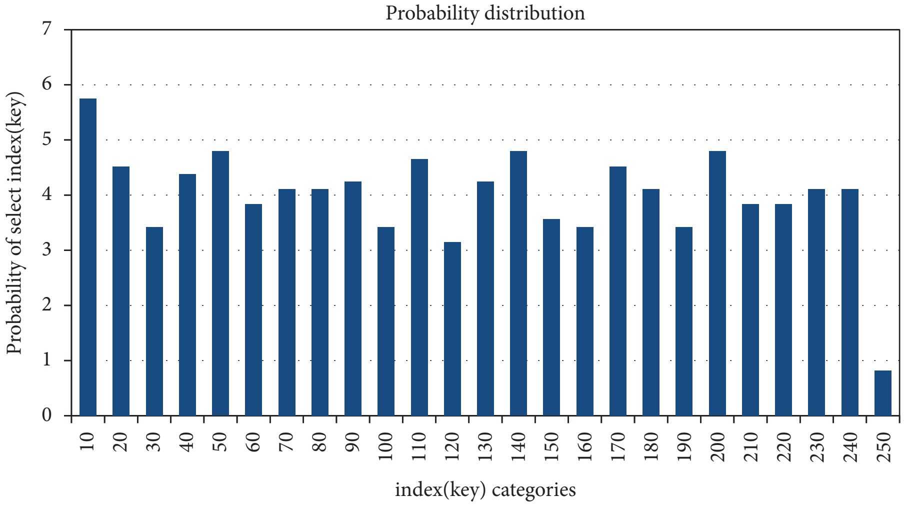

FIGURE 7: Probability of k_i key selection.

128-bit encryption key size was selected. Encryption key (5) selection algorithm was used for the measurement probes. An algorithm for determining the correct encryption key was proposed for the receiving modules (Figure 6).

In implementing the SPECK encryption algorithm on the 32-bit STM32L3xx microcontroller with the Cortex M3 core, 440 of the processor cycles were required to encrypt a 64-bit (8-byte) data block (according to formula (1)). For the calculation of the cryptographic code (according to formula (4)), 2873 of the processor cycles were needed. At the 8MHz core clock frequency this is a time of 360us for key schedule and 55us for encryption, which is a negligible value with respect to the time of measurement and transmitting. For the LM75 sensor used, the conversion time of the measured value is $t_{conversion} = 100ms$, and the time of sending 8-byte data packet at a baud rate of 1000 baud is $t_{transmit} = 30ms$. Using the maximum core frequency of the microcontroller f=32MHz, this time is reduced to 104us.

Presented sensory system has been used to monitor the temperature of the wood chip used for heat production. An undesirable feature of wood chip mass is that the temperature of the wood chip increases spontaneously when stored due to microbiological processes. Uncontrolled self-ignition may occur in such a storage. In our case, 10 RF probes were used to monitor the 50m x 100m space, which were placed directly into the wood chip meter, circa 1,5-2m below its surface. The software part of the measuring system notifies the operator of exceeding temperature limits of the monitored environment. Selected advantages of using the proposed monitoring system with implemented encryption are as follows:

(i) The RF probe is made of a nontoxic PET material, allowing use in agricultural materials and food industry.

(ii) Probes do not need to be manually removed from the material. When transporting the wood chips on a conveyor, they can be easily captured by a magnetic separator and reused.

(iii) The probe is energetically independent. It is powered by a single cell AA battery (1.5V) that can provide the probe with power for approximately 1 year.

(iv) The encryption layer ensures safe transmission of the measured data between the RF probe and the RF receiver. The possible modification of the data sent by a third party and thus misinterpretation of this data are thus eliminated.

(v) As a presentation layer, a PC (Linux, Windows, Mac) and Android applications are available.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported in part by VEGA through the Holistic Approach of Knowledge Discovery from Production Data in Compliance with Industry 4.0 Concept Project under Grant 1/0272/18.

References

[1] R. Beaulieu, S. Treatman-Clark, D. Shors, B. Weeks, J. Smith, and L. Wingers, “The simon and speck lightweight block ciphers,” in *Proceedings of the 52nd ACM/EDAC/IEEE Design Automation Conference (DAC)*, pp. 1–6, 2015.

[2] C. Wang, J. Shen, Q. Liu, Y. Ren, and T. Li, “A Novel Security Scheme Based on Instant Encrypted Transmission for Internet of Things,” *Security and Communication Networks*, vol. 2018, Article ID 3680851, 2018.

[3] X. Su, Z. Wang, X. Liu, C. Choi, and D. Choi, "Study to improve security for IoT smart device controller: drawbacks and countermeasures," *Security and Communication Networks*, vol. 2018, Article ID 4296934, 14 pages, 2018.

[4] T. Park, H. Seo, S. Lee, and H. Kim, "Secure data encryption for cloud-based human care services," *Journal of Sensors*, vol. 2018, Article ID 6492592, 10 pages, 2018.

[5] B. Vinoth, M. Ramaswami, and P. Swathika, "Data security on patient monitoring for future healthcare application," *International Journal of Computer Applications*, vol. 163, no. 6, pp. 20–23, 2017.

[6] R. Beaulieu, S. Treatman-Clark, D. Shors, B. Weeks, J. Smith, and L. Wingers, "Simon and speck: block ciphers for the internet of things," *IACR Cryptology*, vol. 585, 2013.

[7] G. Yang, B. Zhu, V. Suder, M. D. Aagaard, and G. Gong, "The simeck family of lightweight block ciphers," in *Proceedings of the International Workshop on Cryptographic Hardware and Embedded Systems*, T. Gneysu and H. Handschuh, Eds., vol. 9293, Springer, Berlin, Heidelberg, Germany, 2015.

[8] Q. Yang, L. Song, and Z. Huang, "Automatic differential analysis of arx block ciphers with application to speck and lea," in *Information Security and Privacy*, J. Liu and R. Steinfeld, Eds., vol. 9723 of *Lecture Notes in Computer Science*, p. 50, Springer, Cham, 2016.

[9] K. Zhang, J. Guan, B. Hu, and D. Lin, "Security evaluation on Simeck against zero-correlation linear cryptanalysis," *IET Information Security*, vol. 12, no. 1, pp. 87–93, 2018.

[10] Y. Nozaki, Y. Ikezaki, and M. Yoshikawa, "Double-rounds–driven electromagnetic analysis attack for a lightweight block cipher simeck and its evaluation," *Electronics and Communications in Japan*, vol. 100, no. 12, pp. 29–38, 2017.

[11] B. Carpentieri, "Efficient compression and encryption for digital data transmission," *Security and Communication Networks*, vol. 2018, Article ID 9591768, 9 pages, 2018.

[12] M. Skovajsa, P. Fabo, L'. Pepucha, and I. Sládek, "Proposal of a wireless measurement system for temperature monitoring of biological active materials," in *Advanced Mechatronics Solutions*, R. Jablonski and T. Brezina, Eds., pp. 367–372, Springer International Publishing, 2016.

[13] J. Lesurf, "Digital modulation one bit a time. [Technical report]," University of St. Andrews, 2005.

[14] M. Mccauley, "Documentation for the VirtualWire communications library for Arduino".

[15] H. Shinde, "Understanding and Using Cyclic Redundancy Checks with Maxim 1-Wire and iButton Products," 2009.

[16] J. Dudak, G. Gaspar, S. Sedivy, P. Fabo, L. Pepucha, and P. Tanuska, "Serial Communication Protocol with Enhanced Properties -Securing Communication Layer for Smart Sensors Applications," *IEEE Sensors Journal*, vol. 19, no. 1, pp. 378–390, 2018.

[17] J. Dudak, M. Skovajsa, and I. Sládek, "Proposal of a communication protocol for smart sensory systems," in *Proceedings of the 16th International Conference on Mechatronics, Mechatronika*, pp. 107–112, 2014.

[18] R. Nithya and D. S. Kumar, "Where AES is for Internet, SIMON could be for IoT," *Procedia Technology*, vol. 25, pp. 302–309, 2016.

[19] Y. Liu, G. D. Witte, A. Ranea, and T. Ashur, "Rotational-xor cryptanalysis of reduced-round speck," *IACR Cryptology*, vol. 1036, 2017.

[20] EBACS, "ECRYPT Benchmarking of Cryptographic Systems: ENCRYPT II," Measurements of stream ciphers, indexed by machine, 2017.

[21] R. Beaulieu, D. Shors, J. Smith, S. Treatman-Clark, B. Weeks, and L. Wingers, "The simon and speck block ciphers on avr 8-bit microcontrollers," in *Lightweight Cryptography for Security and Privacy*, T. Eisenbarth and E. Ztrk, Eds., vol. 8898 of *LightSec*, Springer, Cham, 2014.