

Methods of Organizing Communication Between Microcontrollers in the System of Monitoring Energy Consumption

Sergiy Novoselov

ORCID 0000-0002-3190-0592

Department of Computer-Integrated Technologies, Automation
and Mechatronics

Kharkiv National University of Radioelectronics

Kharkiv, Ukraine

sergiy.novoselov@nure.ua

Oksana Sychova

ORCID 0000-0002-0651-557X

Department of Computer-Integrated Technologies, Automation
and Mechatronics

Kharkiv National University of Radioelectronics

Kharkiv, Ukraine

oksana.sychova @nure.ua

Abstract—This paper discusses the principles of building networks based on modern LoRaWAN modems. The basics of implementation of data exchange in these networks are given. Experimental studies have been carried out which showed that in the case of "floating window" it is possible to increase the capacity of the network up to 10...15 packets per second without loss of data transmission speed.

Keywords—LoRaWAN modem, electricity accounting, automated control system, ASC, ASCAE

I. INTRODUCTION

Modern devices on microcontrollers are widely used in various spheres of human life. One of the promising areas of such equipment development is autonomous energy accounting systems. The use of microcontrollers combined with transmitters makes it possible to create built-in devices for monitoring energy consumption with low energy consumption from autonomous power sources. The solution of this problem is relevant for the construction of a distributed network of data collection from energy meters.

The paper proposes an analysis of different methods of communication between remote devices for guaranteed message delivery while using the minimum amount of energy consumed. The structure of the microcontroller tool for working with modern data transmission protocols in the LoRaWAN network is also considered.

II. STRUCTURE OF A MICROCONTROLLER DATA COLLECTION DEVICE FROM ENERGY METERING DEVICES

The network of energy metering devices includes the following types of devices: base module or gateway; wireless energy metering modules (meters); suitable sensors for measuring voltage, current, liquid flow, gas.

The block diagram of the basic module is shown in the Fig. 1. The main purpose of this module is the transmission of requests for energy meters to obtain from them data on the current state based on information from the sensor.

The base module is connected to the network, and then to a personal computer (control panel) through interface

converter. The interface can be a serial RS-232 port or Ethernet.

The request is sent to the microcontroller, where a packet is formed for sending over a wireless network. Each packet receives a unique identifier. The transmitter sends the packet to the network and the device switches to receive mode, listening to information from network devices. The received data comes through the transmitter back to the microcontroller. Here they are checked for possible errors. After identification of the received data and their processing, the received information is sent also through the converter of interfaces to a local area network.

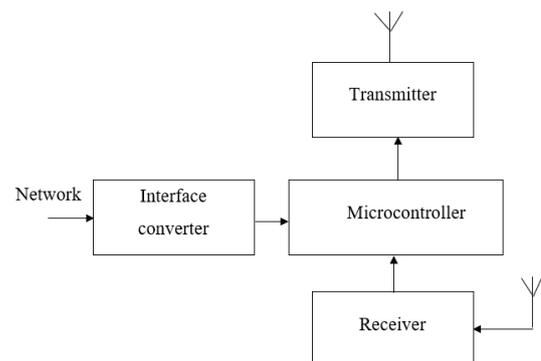


Fig. 1. Block diagram of the base module (gateway).

The block diagram of the energy metering device is shown in Fig. 2. The module includes: sensor; matching unit; analog to digital converter; microcontroller module; internal memory; status indicator; battery; power supervisor; transmitter; receiver.

The sensor is connected to the microcontroller via the matching unit. The sensor can be a digital or analog temperature sensor, current sensor, light sensor, humidity sensor, pressure sensor, etc. From the ADC output, the data is get to the input of the microcontroller through the I2C serial interface. The microcontroller module processes the commands received from the transmitter, and determines their affiliation to this sensor. After parsing the command, the data is read from memory and a packet is formed to be sent to the base module.

The transmission is performed by programmatically entering data that will be transmitted to the frame buffer Tx / Rx together with parameters such as destination address and number of iterations, programming one of the protocol timers to indicate the time at which the frame should be sent.

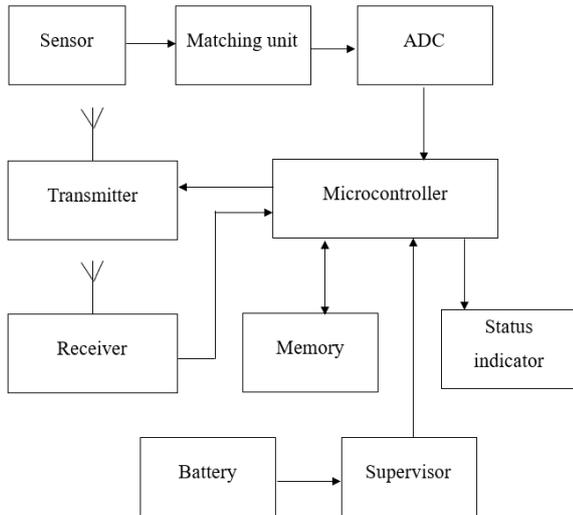


Fig. 2. Block diagram of the device of the metering of energy resources.

This time will be determined by higher-level software, such as the timing of the superframe and the interval. As soon as the packet is prepared and the protocol timer is set, the supervisor unit controls the transmission. When the scheduled time comes, the transmitter controls the sequence of operations of the radio unit and modem to perform the required transmission. The transmitter can execute all algorithms required by IEEE802.15.4, such as CSMA / CA and GTS without CPU intervention, including iterations and random returns.

When the transmission starts, the frame header is created from the parameters programmed by the software and sent with the frame data to the transmitter. At this time, the radio unit is ready for transmission. On the way the data flows to the modem, it enters the checksum generator, which calculates the checksum and attaches it to the end of the frame.

When receiving, the radio unit is configured to receive data on a specific channel. After receiving data from the modem, the frame is sent to the frame buffer Tx / Rx, where both the header and the frame data can be read by the protocol software. An interrupt may be generated when the frame header is received. The frame received from the receiver is transmitted to the checksum generator; at the end of the reception the result of the checksum is compared with the checksum at the end of the frame to control the correctness of the frame.

The network energy meter is powered by a built-in battery, so the program and design provide a mode of saving electricity. To do this, the device uses a supervisor. Its main task is to control the voltage on the battery.

In the case of its reduction to a critical value, the supervisor issues a signal to the microcontroller and it decides to transmit the signal to the network (warning about the transition from sleep mode) and switch to low power consumption mode.

III. ORGANIZATION OF THE MESSAGE EXCHANGE PROCESS

Two approaches to the organization of the confirmation exchange process are proposed: with downtime and with the organization of a floating window. Methods of communication are shown in Fig. 1.

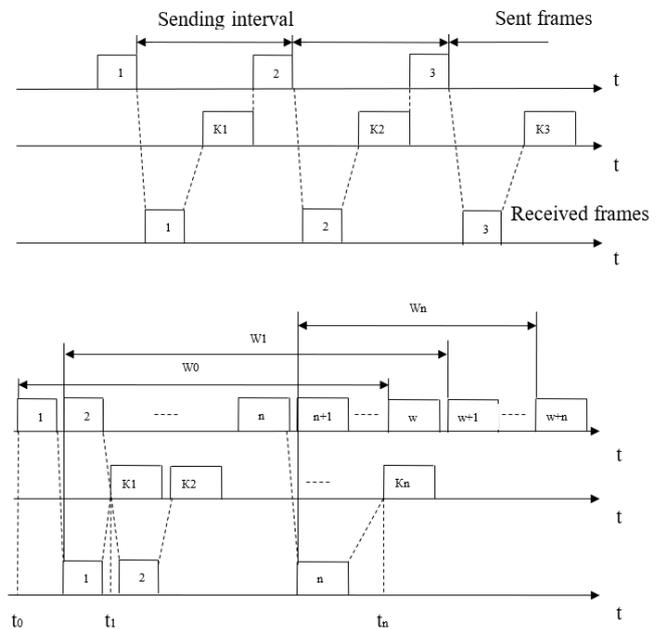


Fig. 3. Methods of organizing messaging.

The downtime method requires that the source that sent the frame expect to receive a confirmation (positive or negative) from the receiver and only then send the next frame (or repeat the distorted). If the confirmation does not fall during the timeout, the frame (or confirmation) is considered lost and its transmission is repeated.

In Fig. 3, *a* shows that in this case the performance of data exchange is significantly reduced. Although the transmitter could send the next frame immediately after sending the previous one, it must wait for the confirmation. The decrease in the productivity of this method of correction is especially noticeable on low-speed communication channels, i.e. in local area networks.

The second method is called the floating window method. In this method, to increase the use of the line, the source is allowed to transmit a number of frames in a continuous mode, i.e. at the maximum possible rate for the source, without receiving positive frames for these frames. The number of frames that can be transmitted in this way is called the window size. Fig. 3, *b* illustrates this method for a window the size of W frames.

The floating window method is more difficult to implement than the downtime method, because the transmitter must store in the buffer all frames for which positive confirmations have not yet been received. In addition, you need to track several parameters of the algorithm: the size of the window W , the frame number for which the confirmation was received, the frame number that can still be transferred before receiving a new confirmation.

The device may not send confirmations for each received correct frame. If several frames came almost together, the receiver can send a confirmation only for the last frame. This meant that all the previous frame and reached safely.

The floating window method has two parameters that can significantly affect the efficiency of data transmission between transmitter and receiver. This is the size of the window and the size of the timeout waiting for the confirmation. In reliability networks, the size of the window needs to be increased to increase the speed of data exchange. Because the transmitter will send frames with smaller pauses.

In unreliable networks, the window size should be reduced. With frequent losses and distortions of frames, their volume increases, so network bandwidth will be spent idle. Useful network bandwidth will be reduced. The choice of timeout does not depend on the network reliability, but on the delays in the frames transmission by the network. In many implementations of the floating window method, the window size and timeout are selected adaptively, depending on the current state of the network.

Thus, the method of data transmission using a floating window is advantageous at low speed information exchange and a reliable communication channel. If there is a problem area of the network, the method of sequential confirmation by speed is close to the method with a floating window due to the large number of lost frames and the need to resend the message.

IV. EXPERIMENTAL RESEARCH

The LoRaWAN protocol provides full two-way communication between network nodes and has special encryption methods to ensure the reliability and security of the system. A typical LoRaWAN network can be represented as end devices (points, nodes), the data of which is transmitted in encrypted form to the gateways, then to the provider's network server and then to the provider's application server, where it comes to the end user.

In a LoRaWAN network, gateways are also called hubs, endpoints, points, or nodes. LoRaWAN nodes can perform various functions. It is measurement, management and control. Typically, such nodes are located at a distance from each other, and are powered by batteries. Using the LoRaWAN protocol, these nodes (points) are configured to communicate with the gateway (hub) LoRa.

Data from nodes is transmitted in both directions, from the node to the server and back. The nodes operate in the transmission mode only for short periods of time, then a temporary window for receiving data opens. The rest of the time the nodes are either in a dormant state or in a receiving state, which depends on the class of the device.

We will use the AnyLogic tool as a modeling tool. AnyLogic contains a graphical modeling language and allows the user to extend the created models using the Java language.

The experiment will be to simulate the behavior of a wireless network. The network gateway is selected as the simulation object. Let this device receive packets from energy meters and send them with a LoRaWan modem. The speed of sending messages will take 9600 B/s. Processing one frame will take a random time from 0.5 to 1.5 ms. In the

experiment, the time scale will be increased to illustrate the passage of time. Therefore, one millisecond in real time will be replaced by one second in the simulation system.

In Fig. 4 shows the window for setting the "Delay" element, which delays the agents for a specified period of time.

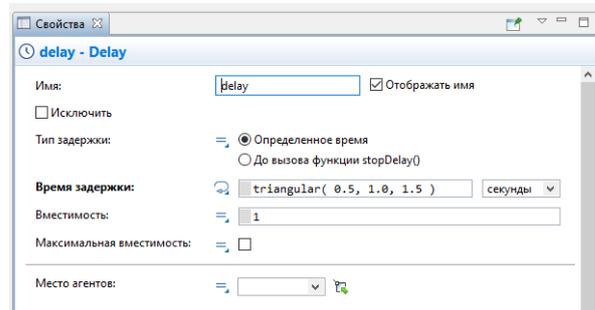


Fig. 4. Setting item "Delay".

The delay time is calculated dynamically, may be random, depending on the current agent or any other conditions. This time is given by the triangular function, which is described by the formula:

$$f(x) = \begin{cases} \frac{2(x - \min)}{(\max - \min)(\text{mod } e - \min)}, & \min < x \leq \text{mod } e \\ \frac{2(\max - x)}{(\max - \min)(\max - \text{mod } e)}, & \text{mod } e < x \leq \max \end{cases}$$

where

$\min = \text{minimum}(x)$; $\max = \text{maximum}(x)$; $\text{mode} = \text{most likely}$.

The triangular distribution is often used in the absence of sufficient information or its complete absence. He can rarely accurately describe the value. Nevertheless, due to the ease of use, the triangular distribution is used as a functional form of representation of areas with blurred logic.

This function automatically checks whether the most probable value of x (mode) belongs to a certain interval (\min , \max). If the specified maximum value is exceeded, the function considers it as the maximum. Conversely, \max is considered as mode , i.e. the appeal to the triangular function (1, 5, 10) is equivalent to triangular (1, 10, 5). This generates a variant of the triangular distribution with a minimum value of 1, a maximum value of 10 and the most probable value of 5. The result was the following scheme of the simulated process, as shown in Fig. 5.

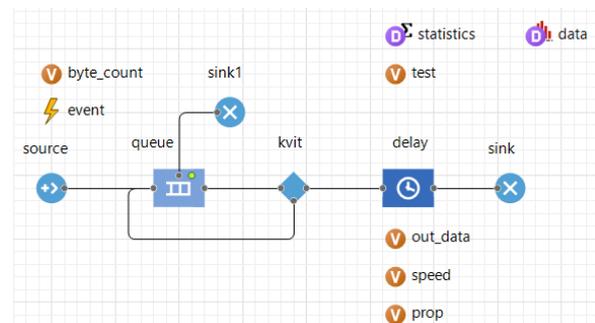


Fig. 5. Scheme of the simulated process.

In this scheme, each packet sent is waiting to be sent and then confirmed by the receiver. The diagram uses the "Kvit" element to model this behavior. This element can be configured so that the confirmation of packets will take place with a certain probability.

In this experiment, a value of 0.2 was chosen as the most appropriate to the actual values in the network. In fig. 6 shows the result of the experiment.

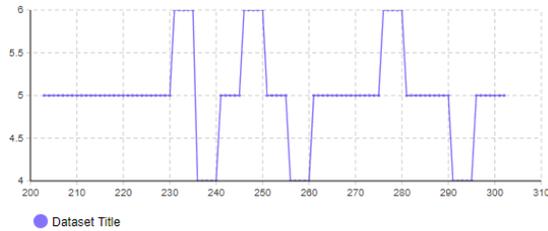


Fig. 6. The result of the experiment.

The result shows that the average network bandwidth is five packets per second.

The second experiment will be to implement a floating window to transmit a set (sequence) of message packets. In our case, the number of messages in the sequence is fixed and equal to five. After receiving the sequence of messages, their integrity is checked and a return message is sent, which confirms the correctness of the data.

The "Batch" component was selected to simulate a floating window. The Batch object converts a specified number of agents that enter the object to a single batch agent. The party can be permanent or temporary. With the creation of a permanent batch, the agents stored in the queue are destroyed (and the properties of the batch agent may depend on the properties of these agents). With the creation of a temporary batch, all these agents are added to the contents of the batch agent and can be subsequently removed from there using the Unbatch object.

This object contains a queue (Queue object) inside which stores the incoming agents. With the accumulation of the number of agents equal to a given batch size, one new agent (agent-batch) is created, which instantly leaves the object.

After receiving five message packets, the "Batch" component generates one final packet, which arrives at the block "Kvit_1". This unit also generates packets that have not been validated with a probability of 0.2. Information that has not been validated is returned to the queue by the "Split" component. In fig. 7 shows the properties of the component "Split".

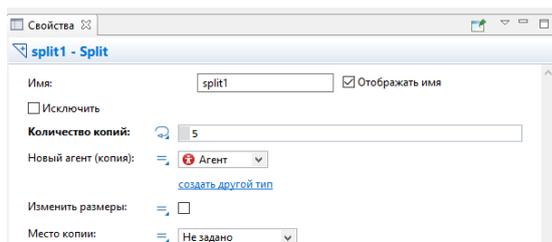


Fig. 7. Properties of the component "Split".

In Fig. 8 shows a diagram of the second experiment with a "floating window".

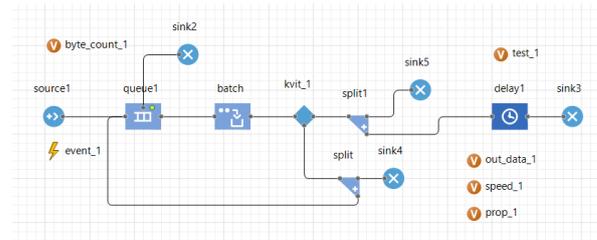


Fig. 8. Scheme of the second experiment with a "floating window".

In Fig. 9 shows the result of the experiment.

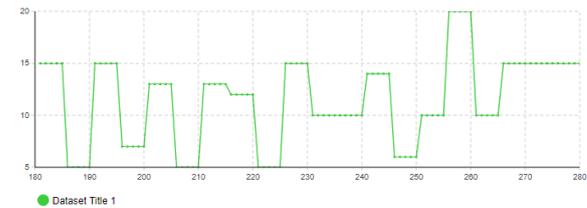


Fig. 9. The result of the experiment.

As can be seen from the experiments, in the case of using a "floating window" was able to solve the problem of increasing network bandwidth with the same data rate. In the second experiment, it was 10...15 packets per second.

V. CONCLUSIONS

Thus, the paper proposes two methods to increase the reliability of data transmission: the method of transmission with downtime and with the organization of a "floating window". The downtime method requires that the source that sent the frame expect to receive a confirmation (positive or negative) from the receiver and only then send the next frame (or repeat the distorted). According to another method, to increase the use of the line, the source is allowed to transfer a certain number of frames in a continuous mode, i.e. at the maximum possible rate for the source, without receiving positive frames for these frames.

An experimental study was conducted using the AnyLogic tool, which confirmed that the "floating window" method is best for these types of wireless networks.

REFERENCES

- [1] Nevludov I.Sh. Programming technology of industrial controllers in an integrated CODESYS environment: Study manual / I.Sh. Nevludov, S.P. Novoselov, O.V. Sichova. - Kharkiv: KNURE, 2019, 264 p. DOI: 10.30837/978-966-659-265-4.
- [2] Al-Juboori G., Tsimbalo E., Doufexi A. [et al.]. A comparison of OFDM and GFDM-Based MFSK modulation schemes for robust IoT applications // IEEE 85th Vehicular Technology Conference (VTC Spring). 2017. P. 1–5.
- [3] Lwanitz F., Lange J. OPC Fundamentals, Implementation, and Application. 2 rev. Heidelberg: Hating, 2002, 225 p.
- [4] Long-distance data networks LoRaWAN [Electronic resource] / Access mode: www / URL: <http://gamma.spb.ru/media/pdf/masters2015/LORA.pdf> – Title from the screen.