

Estimation of the carbon footprint of IoT devices based on ESP8266 microcontrollers

Alexander Lukyanov, Danila Donskoy, Miroslav Vernezi and Dmitry Karev*

Don State Technical University, Rostov-on-Don, 344000, Russia

Abstract. The development of the Internet of Things contributes to improving network protocols and increasing the requirements for energy efficiency of devices. In the field of the Internet of Things and automation systems, one of the most popular microcontrollers is the ESP8266. This article discusses the leading Internet of Things connection protocols based on ESP8266, such as ESP-NOW, HTTP, and ESP-MESH. The study of the power consumption of this microcontroller in various situations and describes the optimal applications of IoT based on ESP8266. The correct choice of communication means of the ESP8266 microcontroller allows you to reduce its power consumption by more than 10% relative to energy-intensive communication algorithms. Compared to the power-intensive MESH network, the reduction in power consumption when using the HTTP protocol is 3.34%, and the percentage of energy-consuming events drops by 50.85%. When using ESP-NOW, energy efficiency increases by 5.35%, and the number of energy-consuming events decreases by 83.05%. The value of the carbon footprint generated by the device during the year was, for the three communication technologies used, 2 kg 500 g, 2 kg 320 g, and 2 kg 290 g of CO₂, respectively.

1 Introduction

The increasing and widespread use of internet of things (IoT) devices make us think about the extent of their impact on forming the carbon footprint and considering ways to reduce it. On the one hand, a significant number of researchers show that the use of devices contributes to reducing the carbon footprint by optimizing the operating modes of various household, industrial or agricultural equipment [1-4]. On the other hand, few people think about the size of the carbon footprint generated by the IoT devices themselves. In a modern apartment, the number of IoT devices can be dozens, and in a cottage or country house – up to a hundred. Thus, the negligible power consumption of a single device scales, and in general, can have a noticeable impact on the formation of a personal carbon footprint. On the scale of a manufacturing enterprise operating hundreds and thousands of IoT devices [3], the amount of carbon footprint about them can become significant.

* Corresponding author: dand22@bk.ru

The study aims to assess the impact of the software and operating modes of IoT microcontrollers on the value of energy consumption and, accordingly, the formation of the carbon footprint of the device.

2 Methods and materials

2.1 Physical organization of measurements

The object of the study is the Wemos D1 mini board with the ESP8266EX microcontroller. [5] The most common modification of this board contains a 4-8 megabyte flash memory model (W25Q80/25Q32ASS1G) with a rated consumption current of 4 mA. Modules with smaller memory (for ex., ECT25S80) are not considered in this article, but their power consumption will be similar and will not exceed 10 mA.

The experimental measurements were carried out in the normal operation mode of the ESP8266EX controller, without the use of power-saving modes. The measurement scheme is shown in Fig. 1.

A module based on the INA219 chip performed the task of converting the voltage drop on the shunt resistor into the current consumed. The data was collected by measuring the voltage drop on the shunt resistor. The shunt resistor was built into the sensor board, with a resistance of 0.1 Ohms and an accuracy of 1%. The resistor is included in the power supply circuit of the ESP8266 microcontroller. The sensor-polling period is 400 microseconds. The measured and digitized data is transmitted via the I2C bus to the Arduino Nano microcontroller and then via the serial port to the computer. The Keysight E3649A certified laboratory power supply is used as the power source.

The INA219 chip has a configurable range of electric current measurements. Based on the maximum current consumption of the ESP8266 microcontroller [20, p.20], to improve the measurement accuracy, a range of 400mA was selected. The total measurement error is about 2% (8mA).

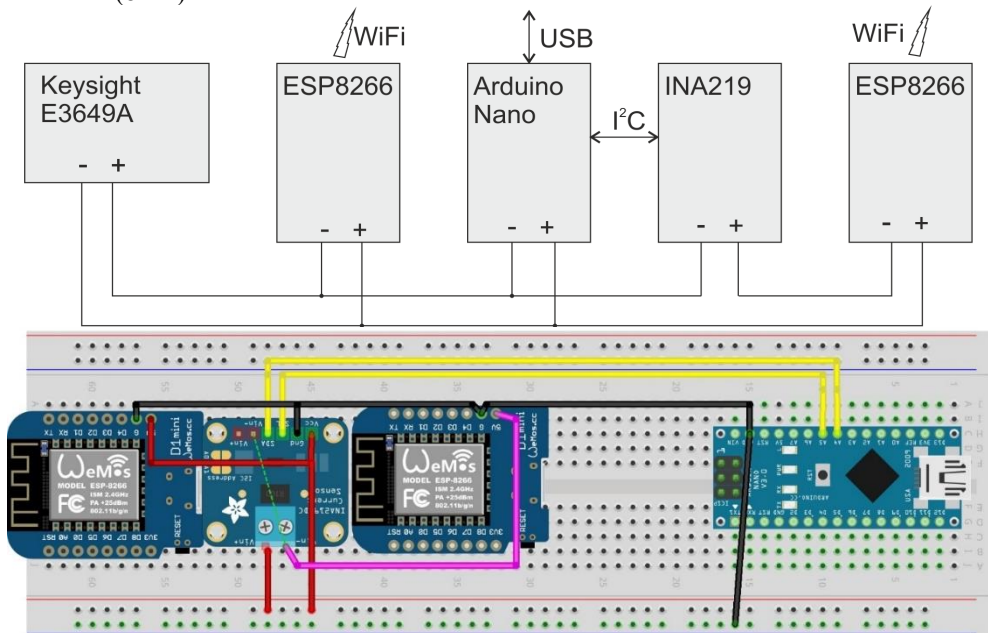


Fig. 1. Circuit for measuring current consumption by the controller.

2.2 Network communication technologies used in the study

2.2.1 ESP-MESH

When organizing an extensive network with homogeneous devices and high requirements for the degree of autonomy of maintaining a network connection, it is necessary to turn to the ESP-MESH technology. [6] The most complex algorithm organizes the communication of devices with this protocol. The standard architecture of a Wi-Fi network is a multipoint network in which there is a central node (access point) and stations. In MESH networks, there is no need for a single node, and the load is distributed evenly over the entire network. Thus, this technology allows you to organize communication with the maximum possible number of network devices in one network. The main parameters required for organizing an autonomous network are: RSSI; the preferred parent node parameter; the routing table; network propagation restriction; point tracking in the network; the loopback exclusion parameter; and the connection recovery algorithm.[7-8].

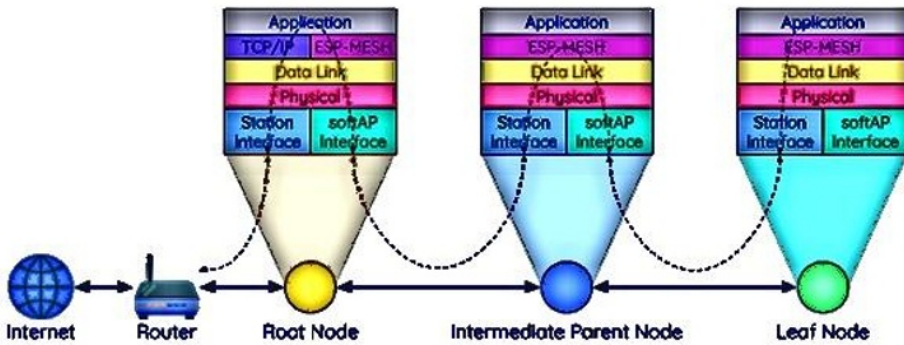


Fig. 2. Description of communication between devices with ESP-MESH. Based on the materials of [6].

Thus, the system needs to analyze a large amount of data about the state of the network and constantly access the network part of the chip, which is the most energy-consuming.

2.2.2 ESP-NOW

ESP controller communication technology was developed and released by Espressif Systems in 2016 [9]. This technology eliminates the procedures associated with Wi-Fi support and speeds up the process of exchanging data packets. The data transfer rate is not more than 1 MB per second. To start communication, only the first connection is required, after which the link is not broken. It is the simplest data exchange protocol on this controller and has the most extended stable connection range. Packets have a small size, up to 250 bytes of helpful information. There is also a limited number of devices in the network, up to 20 pairs. The ESP-NOW network is very flexible in configuring and allows organizing both unidirectional and bidirectional connections between pairs of devices.



Fig. 3. Description of the types of communication between devices with ESP-NOW. Based on the materials of [9].

2.2.3 HTTP

The last microcontroller communication protocol under consideration will be HTTP. This protocol involves the configuration of the network in the form of a client-server. During the operation of this type of device connection, all processes related to Wi-Fi are turned on. The microcontrollers work alternately or simultaneously in two modes: an access point and a station for connecting to the server or working with clients, respectively. When organizing two-way communication, the GET request method was used as the simplest. Thus, the devices used need to process requests and maintain a local network with all Wi-Fi procedures.

2.3 Logical organization of measurements

2.3.1 Bidirectional MESH connection

For this experiment, the painlessness library was used [link]. For the purity of the investigation, we simplified the task as much as possible. The protocol once a second transmitted packets between devices with a payload in the form of a string: "123.1 35". All debugging functionality of the program is disabled for accurate measurements of power consumption [6].

2.3.2 Bidirectional connection over the ESP-NOW network

Like the previous experiment, we are working with a payload in the form of a string: "123.1 35". All debugging functions and third-party procedures are disabled [9].

2.3.3 Bidirectional connection with HTTP protocol

The data is received and sent by the GET method. The payload is the same, but now the microcontrollers need to use resources to work with Wi-Fi. The program structure is simplified as much as possible [10].

3 Results and discussion

The experiments were conducted on the Laboratory of Local Automation and Embedded Systems of the Department "Automation of Production Processes", Don State Technical University.

3.1. Bidirectional MESH connection

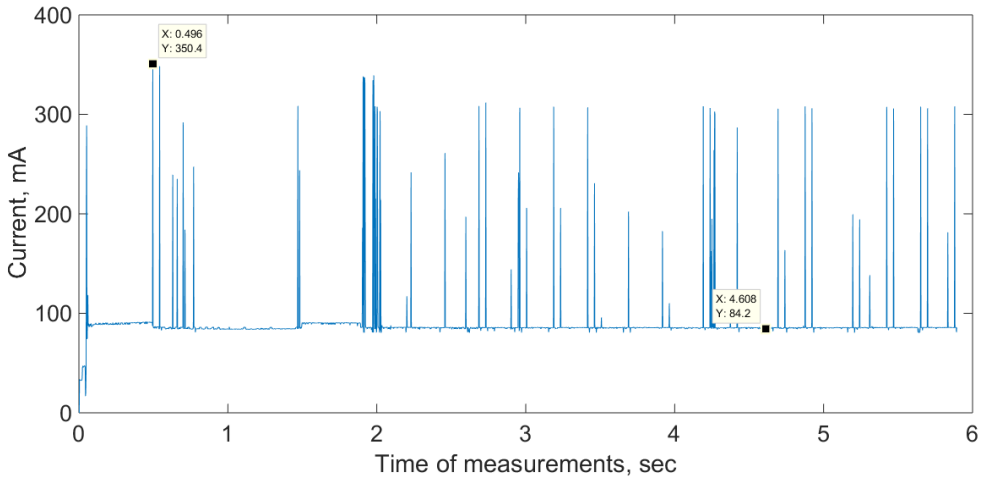


Fig. 4. Current consumption with ESP-MESH.

Figure 4 shows the experimental results of measuring the current consumed by the microcontroller with ESP-MESH technology. The measurement time was about 6 seconds. The current is consumed unevenly, and the overall picture of current consumption in standby mode is overlaid with bursts (events) associated with the activation of the module. At such moments of high consumption, the most energy-consuming procedures occur: connecting to Wi-Fi, starting the server or connecting to it, updating the connection, and so on [6].

For the analysis, these events were converted into a binary sequence at the level of 30% of the maximum current, as shown in Figure 5. The number of events during the measurement period is taken as one of the characteristics of the technology activity. During the experiment, 59 events were detected.

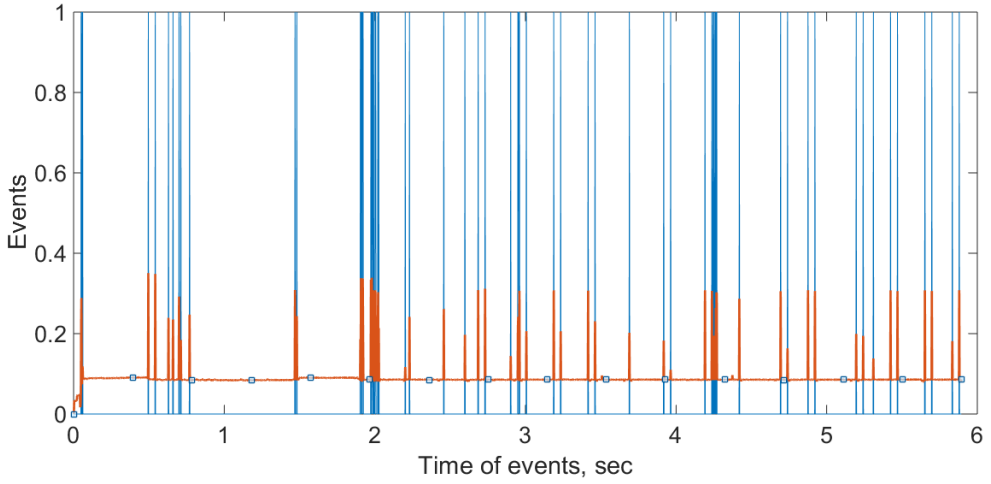


Fig. 5. Current graph with superimposed events, ESP-MESH technology.

Analysis of the amount of current consumed showed that the current is consumed evenly over time. Considering the supply voltage of 4.95 V, the average power consumed by the microcontroller is 2.19 mW.

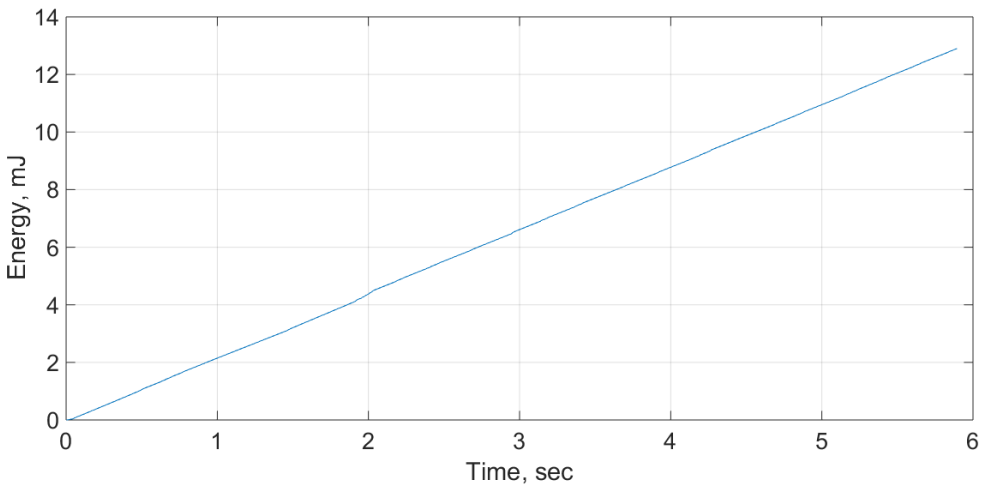


Fig. 6. Power consumption with ESP-MESH.

3.2 Bidirectional connection over the ESP-NOW network

The following graph shows the time characteristic of the current:

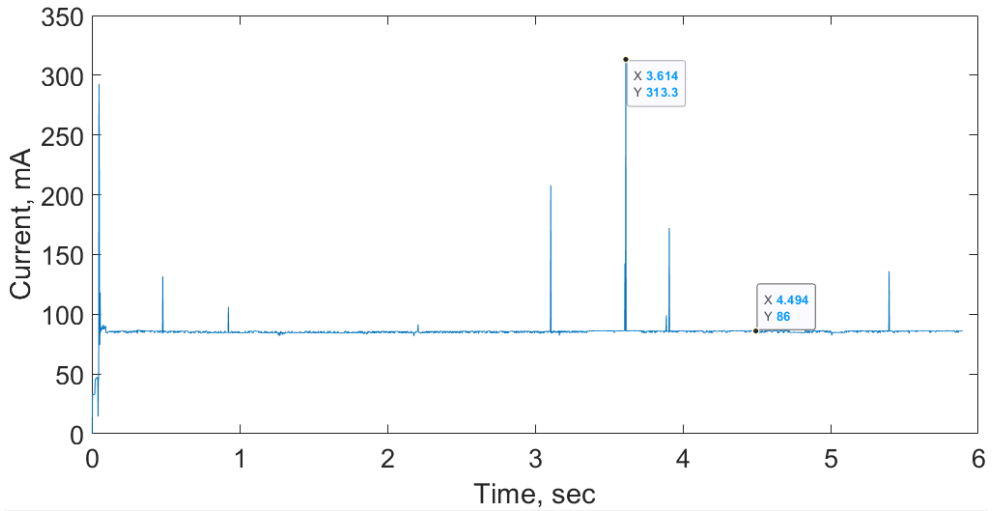


Fig. 7. Current consumption with ESP-NOW.

The characteristics of the events are as follows.

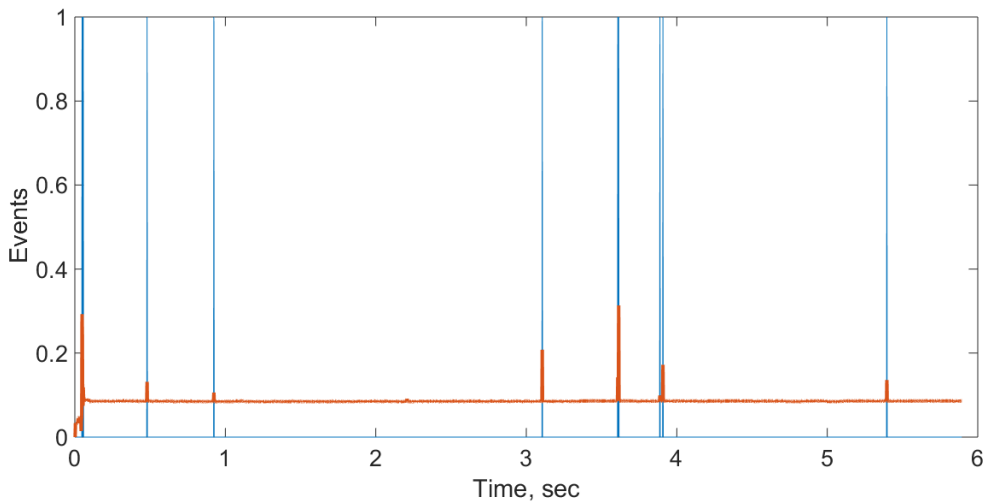


Fig. 8. Characteristics of events with ESP-NOW.

During the experiment, only ten complex events in the operation of the microcontroller were detected.

Analyzing the current consumption with the ESP-NOW protocol, we determine that the average power consumption is 2.07 mW.

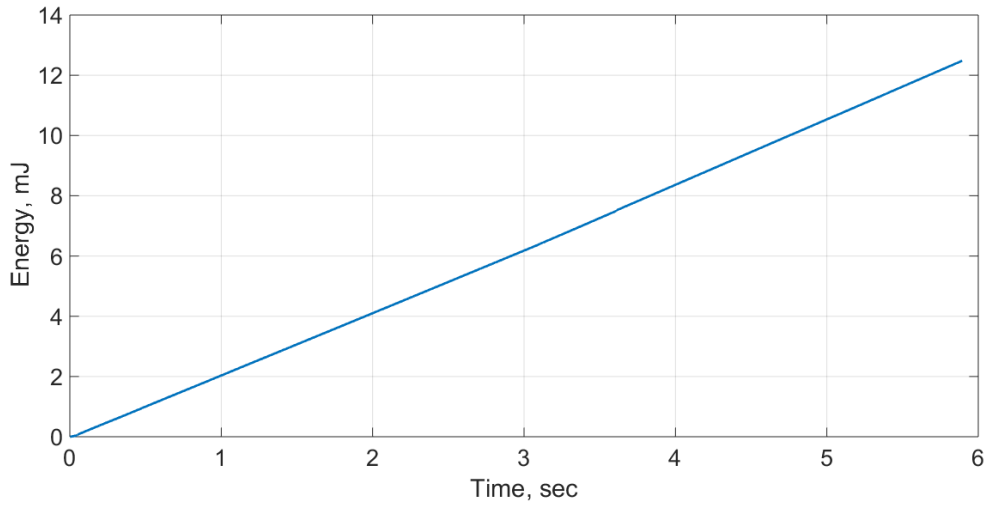


Fig. 9. Energy consumption with ESP-NOW.

3.3 Bidirectional connection with the HTTP protocol

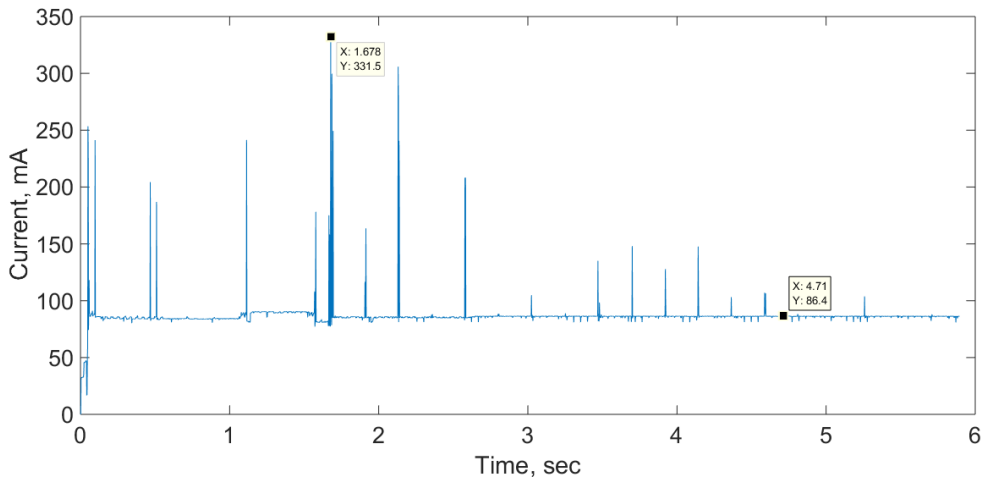


Fig. 10. Current consumption with HTTP.

Consumption increased with the increase in the number of events, which amounted to 29 events during the measurement period.

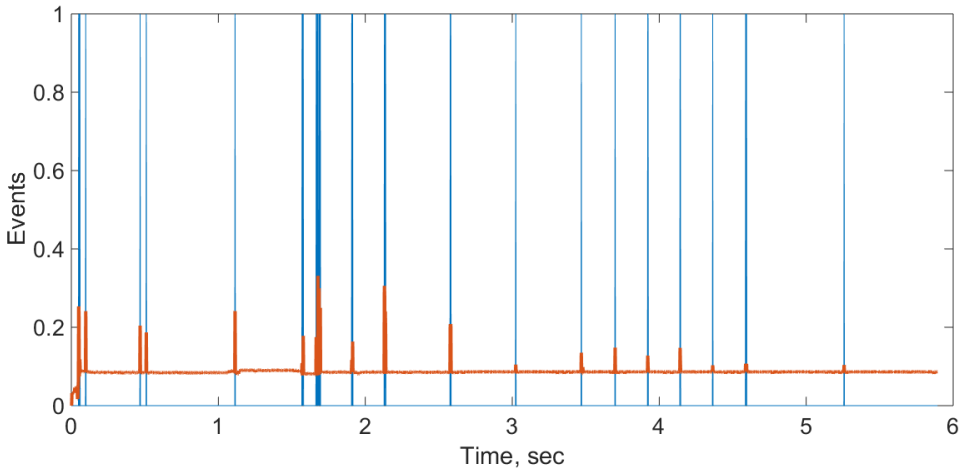


Fig. 11. Characteristics of events with HTTP.

Analyzing the current consumption using the HTTP protocol, we determine that the average power consumption is 2.11 mW.

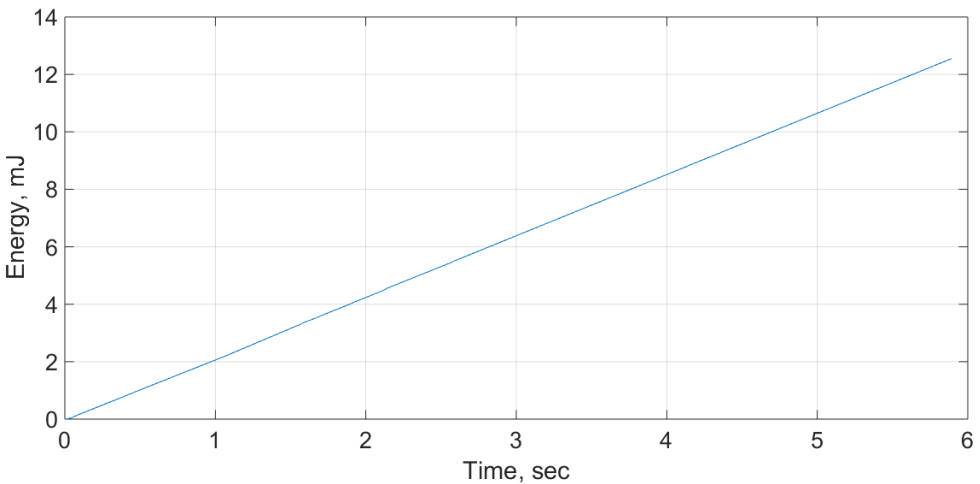


Fig. 12. Characteristics of energy consumption with HTTP.

3.4 The analysis of the obtained experimental results

Immediately it is worth noting that the controllers worked most effectively with the ESP-NOW protocol. This protocol caused the least number of complex events and had stability with a predominant average consumption of 86 mA. This protocol is ideal for continuous monitoring and management in the field of the Internet of Things with maximum energy efficiency [11-13]. What is most important is the ability to use the HTTP protocol in parallel with ESP-NOW. The main scope of this protocol is small automation systems and smart home systems, where the number of devices is no more than 50.

The HTTP protocol is an excellent alternative to ESP-NOW due to its high popularity and good energy efficiency compared to complex MESH network protocols. However, the network structure is becoming standard for Wi-Fi networks, which limits the number of devices. With a high network

load, the power consumption of microcontrollers increases, and the risks of connection loss and network destruction increase [7,9,10].

The ESP-MESH protocol uses almost all the resources of the ESP8266. A complex network topology requires constant monitoring of the surrounding devices, which leads to high power consumption [14-15]. However, the reliability of such a network is very high. Nodes are self-healing and can form large networks of 100 or more devices. But this all affects the performance of the controller.

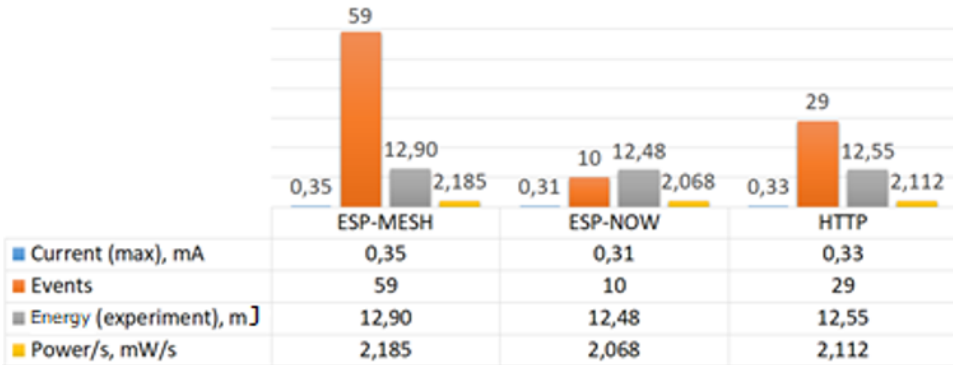


Fig. 13. Comparison of the results obtained for the regular operation of the microcontroller.

3.5 The approximation of experimental results to energy-saving operating modes

To estimate the energy gain and reduce the carbon footprint, we calculate the possible reduction in energy consumption due to the use of energy-saving operating modes of the ESP8266EX microcontroller. According to [16], the controller supports three power-saving modes (Modem sleep, Light sleep, and Deep sleep), during which the quiescent current consumption is reduced to 15 mA, 0.4 mA, and 15 μ A, respectively.

For the experimental data presented in Fig. 4, 7, 10, while maintaining the current values during the events, the quiescent current in the normal mode was replaced by the corresponding current in the energy-saving mode, and the average power consumption was calculated (Table 1). The results show both a radical decrease in the total power consumption during the transition from normal mode to deep sleep mode and an increase in the share of the influence of the number of events on the total power consumption.

Table 1. Estimation of the power consumed by the microcontroller in different operating modes.

		Normal mode	Modem sleep	Light sleep	Deep sleep
Current on wait, mA		70	15	0.4	0.015
Power, mW	<i>ESP-MESH</i>	0.8783	0.0326	0.0042	0.0034
	<i>ESP-NOW</i>	0.8142	0.0293	0.0014	0.0007
	<i>HTTP</i>	0.8038	0.0293	0.0018	0.0010

For a visual assessment of the carbon footprint that forms each of the IoT devices in different operating modes, the energy consumed and the equivalent amount of CO₂ during continuous operation of the device during the year are calculated (Table 2). [17-18] The energy equivalent of CO₂ is taken as 0.325 kg/kWh [1]. A slash in the Carbon eq graph indicates the percentage of the maximum carbon footprint generated in regular operation with ESP-MESH technology.

Table 2. Estimation of the value of the carbon footprint generated by the controller in different modes during the year

		Normal mode	Modem sleep	Light sleep	Deep sleep
ESP-MESH	Energy, kWh	7.69	0.29	0.036	0.030
	Carbon eq, g	2500 / 100%	93.0 / 3.71%	12.0 / 0.47%	9.7 / 0.39%
ESP-NOW	Energy, kWh	7.13	0.26	0.012	0.0060
	Carbon eq, g	2320 / 92.7%	83.0 / 3.33%	4.0 / 0.16%	1.9 / 0.08%
HTTP	Energy, kWh	7.040	0.26	0.015	0.0090
	Carbon eq, g	2290 / 91.5%	80.0 / 3.34%	5.0 / 0.20%	2.9 / 0.12%

Based on the results obtained, it follows that due to the use of deep energy-saving modes, and optimal communication technology, it is possible to reduce the carbon footprint of an IoT device based on ESP8266SX controllers to 0.08% of the nominal modes, that is, more than 1000 times.

4 Conclusion

This study's analysis helps determine the most effective communication tools for devices when organizing the Internet of Things or when creating embedded automation systems with network support. We get the maximum energy efficiency when organizing ESP-NOW networks, and their combination with the HTTP protocol allows us to expand the network area covering [10]. The use of the MESH network protocol on the ESP8266 microcontrollers has many disadvantages. The main area of application of these systems is a small area mesh network covering up to 500 m². This corresponds to about 100 microcontrollers, and a larger number of them leads to a greater load on the network and individual nodes. At the same time, the high power consumption of the ESP8266 with the ESP-MESH protocol requires more capacious autonomous power sources.

When using the HTTP protocol, the reduction in power consumption is 3.34%, and the percentage of energy-consuming events drops by 50.85%. Also, with HTTP, the peak values of the consumed current are reduced by 5.39%. When using ESP-NOW, energy efficiency increases by 5.35%, and the number of energy-consuming events decreases by 83.05%. At the same time, the peak values of the consumed current decreased by 10.59%. The value of the carbon footprint generated by the device during the year was, for the three communication technologies used, 2 kg 500 g, 2 kg 320 g, and 2 kg 290 g of CO₂, respectively.

At the same time, the transition to the use of energy-saving modes in IoT devices potentially reduces energy consumption and the value of the carbon footprint by more than 1000 times.

References

1. How much carbon dioxide is produced per kilowatthour of U.S. electricity generation? // U.S. Energy Information Administration. URL: <https://www.eia.gov/tools/faqs/faq.php?id=74>
2. P. Asopa, P. Purohit, R. R. Nadikattu and P. Whig, "Reducing Carbon Footprint for Sustainable development of Smart Cities using IoT," 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), 2021, pp. 361-367, doi: 10.1109/ICICV50876.2021.9388466.
3. A. Parsa, T. A. Najafabadi and F. R. Salmasi, "Implementation of smart optimal and automatic control of electrical home appliances (IoT)," *2017 Smart Grid Conference (SGC)*, 2017, pp. 1-6, doi: 10.1109/SGC.2017.8308861.
4. A D Lukyanov et al. *Development of methods for analyzing patterns of current consumption in a system for wireless monitoring the effectiveness of metalworking production* // 2020 IOP Conf. Ser.: Mater. Sci. Eng. **900** 012017. Doi <https://doi.org/10.1088/1757-899X/900/1/012017>
5. ESP8266EX Datasheet Version 6.6 // Espressif Systems, 2020, 26 p. URL: https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf
6. Espressif Systems *Documentation for Espressif Mesh Development Framework* (May, 2021) <https://readthedocs.com/projects/espressif-esp-mdf/downloads/pdf/latest/>
7. Abhijeet, S. Basu, and D. Goyal, "Single-Gateway Multi-Node Mesh Network for a Connected Electric Vehicle Ecosystem," 2020 5th International Conference on Communication and Electronics Systems (ICCES), 2020, pp. 259-263, doi: 10.1109/ICCES48766.2020.9137958.
8. Yoppy, R. H. Arjadi, H. Candra, H. D. Prananto and T. A. W. Wijanarko, "RSSI Comparison of ESP8266 Modules," 2018 Electrical Power, Electronics, Communications, Controls and Informatics Seminar (EECCIS), 2018, pp. 150-153, doi: 10.1109/EECCIS.2018.8692892.
9. Espressif Systems *ESP-NOW User Guide* (2016) https://www.espressif.com/sites/default/files/documentation/esp-now_user_guide_en.pdf
10. R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, *Hypertext Transfer Protocol -- HTTP/1.1*
11. P. Srivastava, M. Bajaj and A. S. Rana, "IOT based controlling of hybrid energy system using ESP8266," 2018 IEEMA Engineer Infinite Conference (eTechNxT), 2018, pp. 1-5, doi: 10.1109/ETECHNXT.2018.8385294.
12. Monika Kashyap, Vidushi Sharma, Neeti Gupta, *Taking MQTT and NodeMcu to IoT: Communication in Internet of Things*, <https://doi.org/10.1016/j.procs.2018.05.126> (2018)
13. J. Mesquita, D. Guimarães, C. Pereira, F. Santos and L. Almeida, "Assessing the ESP8266 WiFi module for the Internet of Things," 2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA), 2018, pp. 784-791, doi: 10.1109/ETFA.2018.8502562.

14. A. E. Alves, H. V. Barreto Câmara, J. V. da Silva and R. Araújo de Souza, "Development of an automatic shutdown system for lighting and air conditioning by using Esp8266 to meet energy efficiency requirements in buildings," 2019 IEEE PES Innovative Smart Grid Technologies Conference - Latin America (ISGT Latin America), 2019, pp. 1-5, doi: 10.1109/ISGT-LA.2019.8894993.
15. R. K. Kodali and S. Yerroju, "Energy Efficient Home Automation Using IoT," 2018 International Conference on Communication, Computing and Internet of Things (IC3IoT), 2018, pp. 151-154, doi: 10.1109/IC3IoT.2018.8668155.
16. Espressif Systems *Low Power Solutions* (2016)
https://www.espressif.com/sites/default/files/9b-esp8266-low_power_solutions_en_0.pdf
17. Ilya Borisenko et al. Information support system for management decisions in the agricultural sector // 2019 IOP Conf. Ser.: Earth Environ. Sci. 403 012083. Doi <https://doi.org/10.1088/1755-1315/403/1/012083>
18. Ivushkina, E.B., Alieva, N.Z., Kushnir, I.B., Kalmykova, O.M. The internet of things as a precondition of development of the ICT global infrastructure // Studies in Computational Intelligence. Springer Verlag. Volume 826, 2019, Pages 1003-1009 DOI: 10.1007/978-3-030-13397-9_104