Vol. 8, No. 1, 2023

A COMPUTER SYSTEM FOR COLLECTING DATA ON TEMPERATURE AND HUMIDITY ON PREMISES

Andrii Kalytka, Iryna Yurchak

Lviv Polytechnic National University, 12, S. Bandery str., Lviv, 79013, Ukraine.
Authors' e-mails: andrii.kalytka.mkiks.2022@lpnu.ua, iryna.y.yurchak@lpnu.ua

https://doi.org/10.23939/acps2023.01.017

Submitted on 29.01.2023

Abstract: **Nowadays, it is impossible to do without climate control in enterprises, especially when these are enterprises that manufacture products that can become unusable under the influence of environmental factors such as high or low temperature, humidity, and dryness. Basically, in most enterprises, finding a solution to this issue is required by state standards, which cannot be ignored, because it can harm people.**

The problem of building a computer system for collecting data on the temperature and humidity of premises has been considered. The main means of indoor systems for collecting data on temperature and humidity have been defined, and their main disadvantages and advantages have been highlighted. The basic principles of operation of the new system have been developed. The main nodes required for system implementation have been given. Hardware components for the implementation of the new system have been put forward. The structural scheme of the computer system of protection of living space has been developed and considered. The practicality and expediency of developing a new system have been substantiated. The general algorithm of system operation has been developed and considered.

Index Terms: **computer system; climate control systems; sensors; Arduino; climate control for premises.**

I. INTRODUCTION

Cyber-physical systems are intelligent systems and the goal of such intelligent systems is to use sensors data that signal the environment parameters change as quickly and accurately as possible, special algorithms that involve higher-level automation to perform necessary actions. Usually, cyber-physical systems cover all known aspects of operation with information and measuring systems, complicated by the interaction of their components through networks. They unite informational technologies: from getting data from sensors with their next processing using built-in computing power or using cloud technologies, to traditional operational control and management technologies. In other words, a feature of cyber-physical systems is the combination of informational and operational technologies with some imposed time and space constraints [1]. Sensor monitoring plays a very important role in indoor environment control and energy efficiency. However, it is always a challenging task when discussing the methods and strategy for sensor deployment, considering its numbers and locations [2].

Nowadays, the popularity of the ESP32 chip grows and now both hardware variants of this chip and various branches of its software developers are developing. A wide community of developers, but also academics deal with utilizing the higher generation ESP32 chip as the successor of the ESP8266 microcontroller. The latest scientific articles prove the wide use of the ESP32 chip in various areas. The general possibilities of using a microcontroller with a recommendation for electronic projects are described in the article. The paper provides a comparative analysis of the ESP32 with some other market competitors and introduces the microcontroller specification, features, and programming details. An interesting solution to implement the ESP32 chip as a web server that works in real-time and can be effectively used for monitoring small solar energy systems is provided. The use of a microcontroller is possible with a wide range of environmental monitoring sensors whether it concerns air pollution or directly the implementation of monitoring LPG leakage [3]. In contrast, article [4] describes the use of the chip as a monitoring IoT system in the health sector. The use of the chip as a secure communication system with LoRa modules and implementation of cryptographic standards is described in the article [5], which deals with the Communication System for Remote Microgrids using AES Cryptography on ESP32. The ESP32 chip can also be used to monitor alarms of technological equipment for operator workplaces [6], vibration monitoring [7], and nowadays practically in all areas where a monitoring embedded system is needed.

At the present time, IoT, smart home automation and embedded systems are rapidly developing. This is closely related to the development of available hardware modules and processors. Development boards, a communication interface and peripherals on which are implemented together with the main processor chip, are produced. Nowadays, the popularity of the ESP32 chip grows and now both hardware variants of this chip and various branches of its software development are developing [8].

The majors have learning substance that demands mastery of microcontroller programming competencies [9].

With the improvement of living standards, people's demand for intelligent electric appliances is growing fast. Also, the widespread application of embedded systems allows us to meet this demand [10].

ESP32 is a single 2.4 GHz Wi-Fi-and-Bluetooth combo chip designed with the TSMC low-power 40 nm technology. It is designed to achieve the best power and RF performance, showing robustness, versatility, and reliability in a wide variety of applications and power scenarios [11].

Out of the specifications, it looks like this is the chip of the future, for anything connected that you want to build. Instead of using a microcontroller and add-on Wi-Fi, Bluetooth modules for building connected things, this is the only chip you might want to use [12].

People spend most of their time indoors. Whether it is at home, at work, at school, in healthcare structures, or in recreational facilities, the demand for a comfortable environment is a key driver in building research. As it was pointed out by Frontczak and Wargocki, among the aspects encompassed by the definition of human comfort (visual, acoustic, thermal, and air-quality-related), the thermal condition of the occupants is decisive in determining their level of satisfaction [13].

Temperature, CO_2 concentration, and humidity are the most important factors of the greenhouse's indoor climate that should be carefully regulated in controlled environment agriculture. Controlling indoor temperature in an appropriate range can not only increase fruit production but also prevent plants from heat stress or cold damage. When the temperature is higher or lower than what plants can tolerate, plants would stop growing, and crop yield would significantly decrease [14].

According to an epidemiological survey, people in the USA and Canada on average spent more than 90 % of their time indoors. Thermal comfort of individuals is considered an important factor affecting the overall indoor experience of the occupants, as it is associated with their health, well-being, and productivity. However, a recent study has shown that only 38 % of the occupants are satisfied with the thermal environments of their workplace while 43 % of them are thermally dissatisfied [15].

Although it's not essential that you understand how microcontrollers developed to the point where they are today, it's an interesting story, which can help you understand where an AVR microcontroller fits into the overall hierarchy of information technology (IT) and electronics products [16].

The Atmel ATmega328P is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega328P achieves throughputs approaching 1 MIPS per MHz allowing the system designer to optimize power consumption versus processing speed [17].

II. FORMULATION OF THE PROBLEMS

Providing convenient conditions for storing goods becomes one of the most priority tasks, which is solved with the help of such a system. A temperature and humidity monitoring system are a stand-alone system that collects indoor temperature and humidity data, which can warn of various emergencies and help people avoid harm. It is difficult to imagine modern life without such a system since there are many different rooms, premises, etc. that need it.

Nowadays, it is impossible to imagine infrastructure projects that do not use such temperature and humidity monitoring systems. State safety standards are also an important factor, non-observance of which can bring trouble to production, so they cannot be ignored.

Modern similar systems can prevent many emergencies such as:

- fires;
- gas leakage;
- flooding;
- increasing the level of radiation.

The practical implementation of the master's qualification work is the implementation of a temperature and humidity monitoring system in the room with the subsequent sending of data using the WIFI module on the ESP32 microcontroller to the AWS cloud environment, after which the processing server will send the data to the web application to monitor the situation in the room from large distances A user can install one sensor in a room, but more sensors can be installed depending on the area of the room itself. Sensors are the basis of any monitoring system. Room temperature and humidity sensors are devices that can detect or measure anything. So, sensors can measure the magnitude of a certain phenomenon or action, it can be room temperature, smoke level, pressure or speed, voltage, etc.

This is not the entire list of existing sensors that can be used in monitoring systems. Manufacturers of monitoring systems constantly create new and modify existing models of sensors with improved characteristics.

Firstly, manufacturers of monitoring sensors focus on the needs and goals of users of such equipment. It can be assumed that most problems are solved in one way or another with the development of specific sensors to solve them.

An important advantage of sensors is that they can work stably around the clock, as they do not need rest, they do not need food, and can work anywhere and anytime.

Sensors can also be used in microcontroller systems. By creating such a system, you can influence the indicators of the room without human intervention. In such a system, you can connect various devices that can start working when the indicators decrease or increase. These devices include air conditioners, heating batteries, humidifiers, etc. It is very economical and convenient for automating work.

To summarize, it is worth noting that the topic discussed in the literary sources described in the introduction is of great relevance and interest for the research. Additionally, it should be stated that modern technologies enable easy implementation of the project.

III. OBJECTIVES

So, the main objective of the system for collecting data on temperature and humidity is to prevent overheating, dampness, and dryness of production products or to prevent other emergencies with the possibility of controlling the situation remotely from the other end of the enterprise. This system can provide better data measurement and data acquisition speed than most analogs and other measuring devices such as thermometers, hygrometers, etc. For example, thermometers and hygrometers are already the worst options, since these devices can have certain errors, it is easier to put them out of service due to careless handling, and there is no question of the speed of obtaining data at all since to view the received values, it is necessary to send a person to collect the data. Therefore, obtaining data using this method takes from 20 minutes to an hour, depending on how far the room with the device is located.

Analogs on the market may be devices that can perform the functions of the devices described above, but they may not have the functionality that provides the ability to send data to the user. In addition, they are significantly more expensive than the system proposed in this article. Therefore, data acquisition will be the same as in simple devices, that is, from 20 minutes to an hour.

Of course, there are analogs on the market that can send data to the user using various methods, such as WIFI technology, GSM technology, etc. They can have a high speed of sending data (about 1–2 seconds), but they lose due to their cost because they cost from 2.5 thousand to 10 thousand hryvnias.

The proposed system will cost less since the components are much cheaper (up to 1 thousand hryvnias), and the data transfer speed is high (from 1–3 seconds), but everything depends on the quality of the connection to the Wi-Fi router, but in defense, I can say that more expensive analogs that can transfer data using Internet technologies also depend on it, and in some rooms, there may be no connection to the mobile network for data transfer to the phone.

So, the system proposed in this article has several advantages over other methods of obtaining data, such as price, speed, and compactness. But there is also a drawback, it is its strength since such a system can be easily broken by affecting it physically (in the future, this will be corrected by adding a case) and dependence on the quality of the connection to the Internet network.

Also, the most important parameter of system this type is a throughput (T). For calculations we can use this formula:

$$T = \frac{DR \times Ef \times 1 - Ov}{8} =$$
$$= \frac{54 \times 0.8 \times 1 - 0.2}{8} = 4.32\ \frac{Mbyte}{s}.$$

DR (Data Rate) – data transfer rate in Mbit/s;

Ef (Efficiency) is the ratio of the time of transmission of useful information to the total time of transmission.

Ov (Overhead) is additional information that is added to the payload to ensure correct transmission and reception. These can be checksums, recipient address and other metadata.

The value of Data Rate depends on the Wi-Fi standard used and the connection configuration. For example, when using the 802.11n standard with a frequency of 2.4 GHz, the data transfer rate can be up to 150 Mbit/s. Other standards, such as 802.11ac or 802.11ax, can have Data Rate values between 433 Mbps and 6 Gbps.

In the proposed system, 802.11g Wi-Fi standard is used. It supports data transfer rates up to 54 Mbps at 2.4 GHz. So, for calculation of throughput, Data Rate will equal to 54 Mbps.

To measure efficiency, you can use the formula below. For example, if you send 100 bytes of data, of which 80 bytes are useful information and 20 bytes are used for transmission control, then the efficiency is:

$$Efficiency = \frac{useful\ data\ transmitted}{total\ data\ transmitted} =$$
$$= \frac{80}{100} == 0.8\ or\ 80\ \%.$$

To measure overhead, we use the formula below. For example, if you send 100 bytes of data, of which 80 bytes are useful information and 20 bytes are used for the checksum, the redundancy would be:

$$Overhead = \frac{total\ data - usefull\ data}{total\ data} =$$
$$= \frac{(100 - 80)}{100} = 0.2\ or\ 20\ \%$$

Considering the calculations above, we can assume that the proposed system shows the same data transfer speed indicators or even better than analogues on the market.

IV. OVERVIEW OF THE ESP32 PLATFORM

ESP32 is a powerful SoC (System on Chip) microcontroller with integrated Wi-Fi 802.11 b/g/n, dual-mode Bluetooth version 4.2, and a variety of peripherals. It is an advanced successor of 8266 chips primarily in the implementation of two cores clocked in different versions up to 240 MHz. Compared to its predecessor, except for these features, it also extends the number of GPIO pins from 17 to 36, the number of PWM channels per 16, and is equipped with 4 MB of flash memory.

The ESP32 chip has been developed by the Espressif Systems company, which currently offers several ESP32 versions of the SoC in the form of the ESP32 Developer Kit, the ESP32 Wrover Kit, which also includes an SD card and 3.2’’ LCD, and finally the ESP32 Azure IoT kit with USB Bridge and other built-in sensors. In addition to Espressif Systems, other producers are devoted to these chips – SparkFun with ESP32 Thing DB, WeMoS with its TTGO, D1, Lolin32, and Lolin D32, Adafruid (with Huzzah32), DF Robot (ESP32 FireBeeatle) and many other manufacturers sometimes offer good and sometimes bad clones.

ESP32 includes two cores (Xtensa LX6 processor made with 40 nm technology). CPU cores can be individually controlled. There is 520 KB of on-chip SRAM for data and instructions available. Some SoC modules such as ESP32-Wrover feature 4 MB of external SPI flash and an additional 8 MB of SPI PSRAM (Pseudo static RAM). We can use SPI, I2S, I2C, CAN, UART, Ethernet MAC, and IR in various quantities, depending on the type of board. Standard equipment also includes a Hall Effect sensor, temperature sensor, and touch sensor, other built-in sensors are implemented in Azure IoT and Developer kit. SoC also provides Cryptographic hardware acceleration: AES, SHA-2, RSA, Elliptic Curve Cryptography (ECC), and random number generator (RNG).

ESP32 boards are produced in prototype designs that can be used in smart home applications, automation, wearables, audio applications, cloud-based IoT applications, and more. It is possible to choose a specific development kit or to design a custom embedded system built on the ESP32 microcontroller [3].

V. DEVELOPMENT ENVIRONMENT

Arduino IDE – cross-platform environment (for Windows, Linux, MacOS), developed in C and C ++. It is used to write downloadable software on Arduino-compatible boards as well as third-party boards. The source code for the IDE is released under the GNU Public License 2. The Arduino IDE supports C and C ++ languages using special code structuring rules.

The Arduino IDE is a software library from the Wiring project that contains many common I/O procedures. The user-written code requires only two basic functions to run the sketch and the main cycle of the program, which are compiled and linked from the main () into an executable cyclic program with the GNU toolchain [9].

The Arduino IDE uses Avrdude to convert the source code to a text file in hexadecimal, which is loaded into the Arduino board by a bootloader in the firmware of the board. By default, Avrdude is used as a tool to download user code to official Arduino boards.

With the growing popularity of the Arduino, other developers as a software platform have begun to implement custom compilers and opensource tools (cores) that can create and load thumbnails into other microcontrollers that are not supported by the official line of Arduino microcontrollers [16].

VI. IMPLEMENTATION OF THE DEVICE

You should start installing the system with certain instructions on how to properly power the device. The alarm is based on a microcontroller that can relate to a USB 2.0 cable (Type A/B), which is compatible with Arduino Uno, Arduino Mega, and many other microcontrollers. Alternatively, the device can be connected to a computer or a USB port that should be on the premises. The power supply is plugged into a standard outlet. Power supply constitutes 5 V, 2 A, 10 W. The STANDART socket has a power of 12 W. The output current of this power supply is 1 A, and the DC output voltage is 5 V (it should be assumed that when using a power supply with a DC output voltage greater than 5 V, you can simply burn the device). The next step when installing the system will be to check the functionality of the microcontroller. Many factors can damage the device, such as transportation or careless connection to the power source. When the microcontroller is connected to the power supply, there are certain indicators on the board, if everything works then the indicator glows red. This is one of the easiest ways to check the microcontroller. If the previous stage was successful, then you can proceed to connect the most important component of the system – the sensor. This sensor is responsible for taking readings in the room. Next, you can consider the element of the system interface. It is represented by a small LCD screen of 16×2 pixels. With its help, you can view the temperature and humidity in real-time. Based on the received data, the consumer will be able to independently decide what to do next and act further.

VII. GENERALIZED STRUCTURE OF A COMPUTER SYSTEM

All operations on all nodes of the system are controlled by a microcontroller. The interaction between all nodes of the system is depicted on the block diagram in Fig. 1. Also, this diagram reflects the general principles of the system.

Based on the results of the analysis of the requirements and the technical task, the structure of the computer system for collecting temperature and humidity data in the room was developed. The general management of the system is carried out by a programmed board with the corresponding application software.

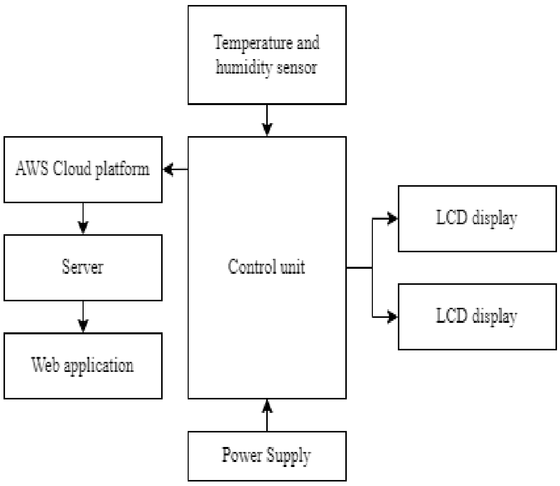

Fig. 1. Block diagram of a computer system for collecting data on temperature and humidity on premises

VIII. SOFTWARE MODEL

The microcontroller, in contrast to the microprocessor, usually has a relatively small bit (8–16 bits) and a rich set of commands for manipulating individual bits. Bit commands provide the ability to control discrete equipment (raise/lower the barrier, turn on/off the lamp or heater, start/stop the engine, open/close the valve, etc.). The ability to operate with individual bits, input, and output discrete signals is called a "bit processor". To achieve the required performance and flexibility, the registry file must support the following I/O schemes:

- One 8-bit output operand and one 8-bit input.
- Two 8-bit output operands and one 8-bit input.
- Two 8-bit output operands and one 16-bit input.
- One 16-bit output operand and one 16-bit input.

Almost all instructions that work with a register file must have direct access to the registers, and many of them are single-cycle instructions. The data memory address is assigned in each register, which reproduces them directly to the first 32 locations of the user's data space. This memory organization is not physically implemented as an SRAM location, it provides more flexibility in accessing the registers, because the registers of the X-, Y-, and Z-pointers can be set to index any register in the file. Registers R26.R31 also have additional functions to their general use. These registers are 16-bit address pointers for indirect addressing of the data space [13]. The three registers of indirect addresses X, Y, and Z are defined as described in Fig. 2. In different addressing modes, these address registers have the functions of fixed movement, automatic increase, and automatic decrease (for details, see the reference to the instruction set).

You can also say a few words for the status registers. The status register contains information about the result of the last executed arithmetic instruction. This information can be used to change the program flow to perform conditional operations. Note that the status register is updated after all ALU operations, as it is indicated in the instruction set link. This, in many cases, eliminates the need to use specialized comparison instructions, which will lead to faster and more compact code. The status register is not saved automatically when the interrupt procedure is entered and is restored when you return from the interrupt. You need to deal with this software [17].

The predecessor of ESP32, the ESP8266 has a built-in processor. However, due to the multitasking involved in updating the Wi-Fi stack, most of the applications use a separate micro-controller for data processing, interfacing sensors, and digital Input Output. With the ESP32 you may not want to use an additional microcontroller. ESP32 has Xtensa® Dual-Core 32-bit LX6 microprocessors, which run up to 600 DMIPS. The ESP32 will run on breakout boards and modules from 160 Mhz up to 240 MHz. That is a very good speed for anything that requires a microcontroller with connectivity options.

The two cores are named Protocol CPU (PRO_CPU) and Application CPU (APP_CPU). That basically means the PRO_CPU processor handles the WiFi, Bluetooth, and other internal peripherals like SPI, I2C, ADC, etc. The APP_CPU is left out for the application code. This differentiation is done in the Espressif Internet Development Framework (ESP-IDF). ESP-IDF is the official software development framework for the chip. Arduino and other implementations for the development will be based on ESP-IDF.

ESP-IDF uses freeRTOS for switching between the processors and data exchange between them. We have done numerous tutorials on freeRTOS and with all the bare-metal programming tutorials for ESP32 we will try and cover this aspect in detail. Although the feature set is great at the price at which the chip is being sold, the complexity is enormous. For the chip to get widely adopted, it will require huge efforts from Espressif as well as the community.

Most of the modules like ESP32 Wroom use external Flash-W25Q32 (4 Mbytes) for storing the application code. The chip supports 4×16 Mbytes of external QSPI flash and SRAM with hardware encryption based on AES.

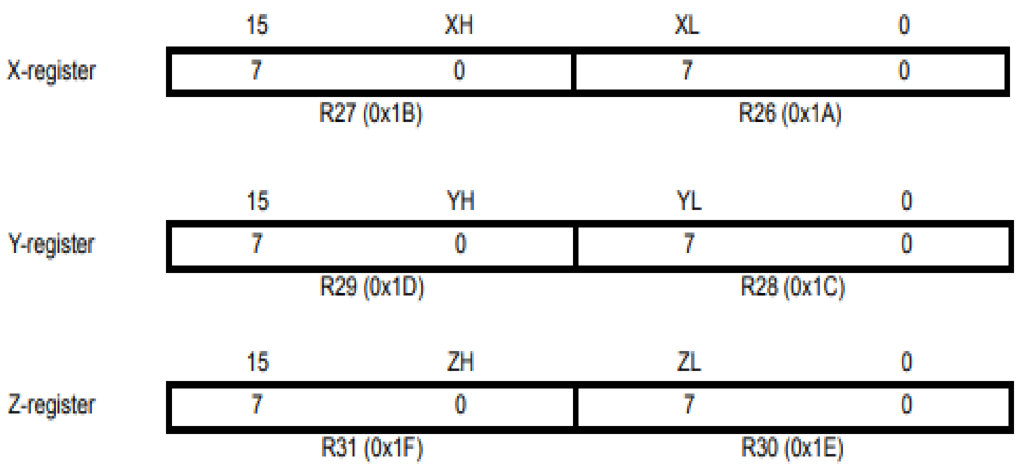

Fig. 2. Format X, Y, and Z registers of indirect addresses

ESP32 accesses the external QSPI flash and SRAM through high-speed caches.

- Up to 16 Mbytes of external flash are memory-mapped onto the CPU code space, supporting 8, 16, and 32-bit access. Code execution is supported.
- Up to 8 Mbytes of external SRAM are memory-mapped onto the CPU data space, supporting 8, 16 and 32-bit access. Data-read is supported on the flash and SRAM. Data-write is supported on the SRAM.

The predecessor of ESP32, the ESP8266 has a built-in processor. However, due to the multitasking involved in updating the Wi-Fi stack, most of the applications use a separate microcontroller for data processing, interfacing sensors, and digital Input Output. With the ESP32 you may not want to use an additional microcontroller. ESP32 has Xtensa® Dual-Core 32-bit LX6 microprocessors, which run up to 60 DMIPS. The ESP32 will run on breakout boards and modules from 160 MHz up to 240 MHz. That is a very good speed for anything that requires a microcontroller with connectivity options.

The two cores are named Protocol CPU (PRO_CPU) and Application CPU (APP_CPU). That means the PRO_CPU processor handles the Wi-Fi, Bluetooth, and other internal peripherals like SPI, I2C, ADC, etc. The APP_CPU is left out for the application code. This differentiation is done in the Espressif Internet Development Framework (ESP-IDF). ESP-IDF is the official software development framework for the chip. Arduino and other implementations for the development will be based on ESP-IDF.

ESP-IDF uses freeRTOS for switching between the processors and data exchange between the processors and data exchange between them. We have done numerous tutorials on freeRTOS and with all the bare-metal programming tutorials for ESP32, we will try and cover this aspect in detail. Although the feature set is great at the price at which the chip is being sold, the complexity is enormous. For the chip to be widely adopted, it will require huge efforts from Espressif and the community.

The processors have closely tied internal memory for the following usage:

- 448 Kbytes ROM for booting and core functions.
- 520 Kbytes on-chip SRAM for data and instruction.
- 8 Kbytes SRAM in RTC, which is called RTC SLOW Memory and can be accessed by the co-processor during the Deep-sleep mode.
- 8 Kbytes SRAM in RTC, which is called RTC FAST Memory and can be used for data storage; it is accessed by the main CPU during RTC Boot from the Deep-sleep mode.
- 1 Kbit of EFUSE, 256 bits of which are used for the system (MAC address and chip configuration), and the remaining 768 bits are reserved for customer applications, including Flash-Encryption and Chip-ID.

Up to 16 Mbytes of external SRAM are memory-mapped onto the CPU data space, supporting 8, 16, and 32-bit access. Data-read is supported on the flash and SRAM. Data writing is supported on the SRAM. Since the processor architecture is 32-bit. The internal peripherals, the Wi-Fi, Bluetooth, External Memories, etc. are mapped to 2^32 (4 GB) address space [11], [12].

Also, one interesting thing to note is that both processors are mapped symmetrically to this address space. It means a register for example can be accessed from the same address location from both the CPUs as it is shown in Fig. 3 below.

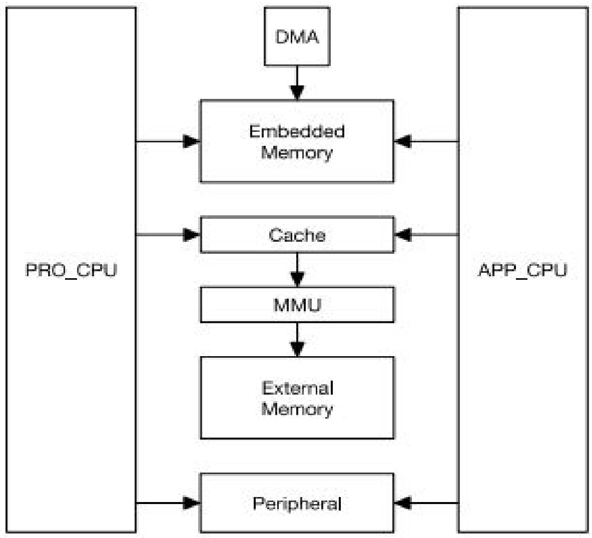

Fig. 3. Processors Mapping Structure of ESP32

IX. ALGORITHM OF SYSTEM FUNCTIONING

After initializing the microcontroller, it performs all the necessary requests to the outputs of the sensors and performs certain calculations. To ensure this, you can look at the indicator on the microcontroller. If it is lit, it means that the initialization was successful. Accordingly, this will be the start of the algorithm, which will mean that the sensors have also started working. Firstly, the sensor reads temperature and humidity data and transmits this data to the microcontroller, which at this point sends it to the cloud environment. The cloud environment in turn sends it to the server. The server receives this data and transmits it to the web application. When a web application receives a packet of data, it graphs it and displays it in a separate widget. In general, at the request of the client, the microcontroller can be modified in any way possible, everything depends on its needs and resources. This system is very “flexible”, so it can be modified with different sensors and interfaces. You will not be able to disconnect the system to turn it off as it will be connected to the battery. It will turn off after 2–3 hours.

The structural scheme of the algorithm is created, which is shown in Fig. 4.

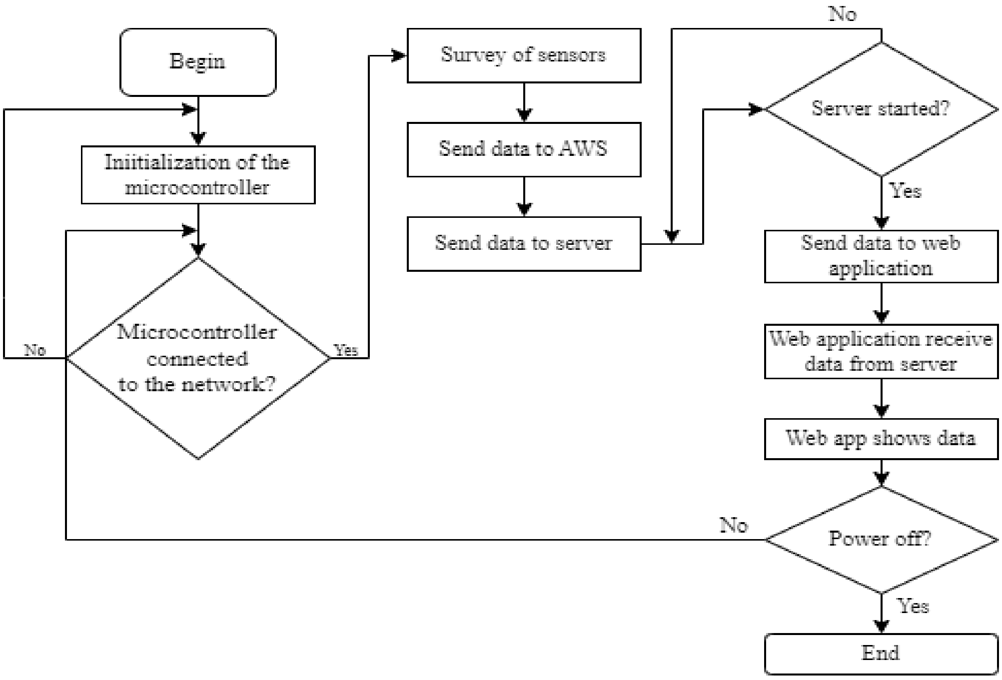

Fig. 4. Generalized system operation algorithm

X. PLACEMENT OF MONITORING SYSTEM

Installing a monitoring system on the premises requires attention to many factors, such as the size of the room or area, the location of power sources, and the possibility of additional wiring (if the customer intends to improve the device by adding several special sensors).

Despite everything, you do not need to plan the room, since this system does not require any mounting in the walls, etc. You just need to connect the device to the outlet and put it in any place convenient for you.

When installing in garages, it is necessary to consider not only the absence of heating but also long-term power outages, without which monitoring systems cannot work. There are still few models that can work in autonomous mode, but some of them can continuously cope with their tasks on one set of batteries for a whole year. In addition, if the autonomous power is exhausted, a notification will be displayed on the web page of the application to let you know about it. Car batteries can be used to power the systems for a long time.

In general, there are no requirements for installing the components, since they have been already installed and do not need to be installed, but there will be problems with flashing the microcontroller itself since a specialist cannot be dispensed with. To flash the microcontroller itself, it is necessary to specify the router data in the operating code of the microcontroller. These data can be entered at the installation site or the customer will send the data to the specialist in advance.

XI. CONCLUSION

According to the results of the research, the basic means of premises monitoring temperature and humidity systems were determined. Comparisons and experiments showed that the method using a thermometer and hygrometer indicated the time of obtaining data in the interval from 20 minutes to 1 hour. Analogs on the market without the functionality of remote data transfer also showed that the time of receiving data was long. Analogs with the functionality of remote data transmission showed high performance, approximately 1–2 seconds, but they significantly lost in price (from 2.5 thousand to 10 thousand hryvnias). The proposed system won in price (up to 1 thousand hryvnias) and was in no way inferior in data transfer speed, approximately from 1–3 seconds. Their main disadvantages and advantages were highlighted. Approaches to building a system of monitoring were considered. The main nodes required for system implementation were given. Hardware components for the implementation of the new system were proposed.

A generalized block diagram of a computer system for monitoring temperature and humidity on the premises was developed. A general algorithm of the system operation was developed and described. It monitored the state of the object in real-time. The prospects of application of the developed structural and algorithmic solutions were shown.

Using such a system at home, premises, and storage you can significantly increase the control of products and reduce the risk of product deterioration. If you look at the statistics, any monitoring system reduces the risk of deterioration of a product, but the main advantage of the development is that it is much cheaper than analogs on the market. The developed monitoring system has high reliability, speed, autonomy, and easy installation in a smart home.

REFERENCES

[1] Bielik V., Morozov Y., Morozov M. (2021). Sensors in Cyber-physical Systems Based on Android Operating System, *Advances in Cyber-Physical Systems*, Lviv, pp. 83–89. DOI: https://doi.org/10.23939/acps2021. 02.083

[2] Cao S., Ding J., Ren C. (2020). Sensor deployment strategy using cluster analysis of Fuzzy C-Means Algorithm: Towards online control of indoor environment's safety and health, pp. 35–41. DOI: https://doi.org/ 10.1016/j.scs.2020.102190

[3] Babiuch M., (2019), Using the ESP32 Microcontroller for Data Processing. *Conference: 2019 20th International Carpathian Control Conference (ICCC)*, pp. 1–6. DOI: 10.1109/CarpathianCC.2019.8765944.

[4] Ghosh D., Agrawal A., Prakash N. (2018). Goyal Smart Saline Level Monitoring System Using ESP32 And MQTT-S. *20th International Conference on e-Health Networking, Applications and Services, Healthcom 2018*, pp. 1–5. DOI: 10.1109/HealthCom.2018.8531172.

[5] Iqbal A. l, Iqbal T. (2018). Low-cost and Secure Communication System for Remote Micro-grids using AES Cryptography on ESP32 with LoR a Module. 2018 IEE E Electrical Power and Energy Conference (EPEC), pp. 1–5. DOI: 10.1109/EPEC.2018.8598380.

[6] Bipasha Biswas S., Tariq Iqbal M. (2018). Solar Water Pumping System Control Using a Low-Cost ESP3 2 Microcontroller. *Canadian Conference on Electrical and Computer Engineering, CCECE 2018*, pp. 1–5. DOI: 10.1109/CCECE.2018.8447749.

[7] Urban P., Landryova L., (2017), Collaborative Operations Using Process Alarm Monitoring. *IFIP WG 5.7 International Conference on Advances in Production Management Systems*, APMS, pp. 441–448. DOI: 10.1007/978-3-319-66923-6_52.

[8] Takacs G., Vachálek J., Rohal'-Ilkiv B. (2015). Online Structural Health Monitoring and Parameter Estimation for Vibrating Active Cantilever Beams Using Low-Priced Microcontrollers. *Shock and Vibration, Volume*, pp. 34–45. DOI: https://doi.org/10.1155/2015/506430.

[9] Irawan D., Puji Astutik R. (2020). Design and Programming Atmega Microcontroller. *Kontribusia: Research Dissemination for Community Development*, [S.l.], Vol. 4, No. 1, pp. 356–359. ISSN 2614-1590. DOI: http://dx.doi.org/10.30587/kontribusia.v4i1.1554.

[10] Weihong Wan, Liegang Xia (2016). Design, and implementation of a control system using AVR microcontroller. *2016 International Conference on Image Analysis and Signal Processing*, pp. 320–323. DOI: https:// ieeexplore.ieee.org/document/5054583

Espressif Systems, ESP32 Series Datasheet v4.2., (2023), Available at: https://www.espressif.com/sites/default/ iles/documentation/esp32_datasheet_en.pdf

[12] ExploreEmbedded.com (2019). *Overview of ESP32 features. What do they practically mean?* Available at: https://exploreembedded.com/wiki/Overview_of_ESP32 _features._What_do_they_practically_mean%3F (Accessed: 01/30/2023)

[13] Grassi B., Piana E., Lezzi A., Pilotelli M. (2022). A Review of Recent Literature on Systems and Methods for the Control of Thermal Comfort in Buildings, pp. 220–251. DOI: https://doi.org/10.3390/app12115473

[14] Chen W.-H. and You F. (2022). Semi closed Greenhouse Climate Control Under Uncertainty via Machine Learning and Data-Driven Robust Model Predictive Control in IEEE Transactions *on Control Systems Technology*, Vol. 30, No. 3, pp. 1186–1197. DOI: 10.1109/TCST. 2021.3094999.

[15] Deng M., Fu B., and Menassa C. C. (2021). Room Match: Achieving Thermal Comfort Through Smart Space Allocation and Environmental Control in Buildings, *2021 Winter Simulation Conference (WSC), Phoenix, AZ, USA*, pp. 1–11. DOI: 10.1109/WSC52266. 2021.9715438.

[16] Trevennor, A. (2012). A Brief History of Microcontrollers. Practical AVR Microcontrollers. *Apress, Berkeley*, pp. 3–11 *CA*. DOI: https://doi.org/10.1007/978-1-4302-4447-9_1

[17] Microchip.com 2021. Information letter ATmega328p (2021). Available at: http://ww1.microchip. com/downloads/ en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf\ (Accessed: 01/30/2023).

Andrii Kalytka is a fifth-year student at Lviv Polytechnic University. He graduated from high school in 2018 and the same year entered the Computer Engineering Department at the Institute of Computer Technology, Automation, and Metrology. He is currently studying for a Master's degree at Computer Engineering at Lviv Polytechnic National University. Research interests – computer systems and networks, IOT systems.

Iryna Yurchak, from 1987 to 1992, was a postgraduate at Lviv Polytechnic National University. Her research interests include Artificial intelligence systems, intelligent computing systems, neural networks, genetic algorithms, fuzzy logic, recognition systems, prediction problems, computer graphics, computer modeling, and web design.