# TOWARDS SUSTAINABLE SUBSISTENCE AGRICULTURE IN ROMANIA WITH LOW-COST IoT MODULES AND SOLUTIONS – PROOF OF CONCEPT
/
# *CĂTRE AGRICULTURĂ DE SUBZISTENȚĂ SUSTENABILĂ ÎN ROMÂNIA CU MODULE ȘI SOLUȚII IoT LOW-COST – PROOF OF CONCEPT*

**Ştefania-Adriana HOŢEI[1)], Cristina Ioana MARGHESCU[2*)], Rodica Cristina NEGROIU[2)], Andrei DRUMEA[2)]**

[1)] Faculty of Electronics, Telecommunication and Information Technology, National University of Science and Technology POLITEHNICA of Bucharest;

[2)] Electronic Technology and Reliability Department, Faculty of Electronics, Telecommunication and Information Technology, National University of Science and Technology POLITEHNICA of Bucharest;

**Tel. +40 21 3169634, e-mail: cristina.marghescu@cetti.ro*

*DOI: https://doi.org/10.35633/inmateh-71-56*

***Keywords***: *smart agriculture, IoT, low cost, ESP32*

## ABSTRACT

*Climate change is having a strong impact on agriculture. Smallholder farmers and those practicing subsistence agriculture are most affected. IoT monitoring tools enable real-time monitoring of crops to make decisions and better manage resources. For such tools to be attractive to smallholder and subsistence farmers, the system must be inexpensive and easy to use, preferably with interfaces with which the user is already familiar. The goal of the research was to develop a low-cost system with commercially available components and a familiar user interface. The system has successfully passed the first tests and is now being implemented.*

## REZUMAT

*Schimbările climatice au un impact puternic asupra agriculturii, micii fermieri și cei care practică agricultura de subzistență fiind cei mai afectați. Instrumentele de monitorizare IoT permit monitorizarea în timp real a culturilor pentru a lua decizii și a gestiona mai bine resursele. Pentru ca astfel de instrumente să fie atractive pentru utilizatorii vizaţi, sistemul trebuie să fie ieftin și ușor de utilizat, de preferință cu interfețe cu care utilizatorul este deja familiarizat. Scopul cercetării a fost de a dezvolta un sistem ieftin cu componente disponibile comercial și o interfață familiară. Sistemul a trecut cu succes primele teste și este acum în curs de implementare.*

## INTRODUCTION

Romania has one of the highest percentages of rural population among the countries of the European Union, estimated at 44.51% in 2022 (*http://starea-natiunii.ro/index.php/ro/noutati/facts-figures/25-facts-figures/170-romania-si-agricultura-de-subzistenta*). At the same time, the percentage of the population working in agriculture is one of the highest, at 29.4% (as of 2014) (*https://highclere-consulting.com/agricultura-de-subzistenta-o-realitate-socio-economica-fara-viitor/*). In 2020, there were 2887 thousand agricultural holdings cultivating 12.8 million hectares of land. Of these, 54% were smaller than 1 ha. To put it another way, in 2010 (last known data), 86% of Romanian farms had an annual production of less than 4000 euros and could be considered subsistence farms (*https://highclere-consulting.com/agricultura-de-subzistenta-o-realitate-socio-economica-fara-viitor/*). Although the situation has changed since 2010 (most importantly, 1 million farms have been lost since then), there is still a significant number of subsistence farmers and small vegetable gardens next to houses. Against this backdrop and given the negative impact of climate change, it is important to find ways to improve production. Smart agriculture provides information to farmers and helps improve practices to increase production and protect crops. It is based on the IoT framework: sensors are used to collect data and the data is then processed to either guide a process or to improve understanding and enable prediction of various factors. The most commonly used sensors in agriculture are temperature, humidity, soil moisture, light dependent resistor (LDR), wind speed, RGB camera, hyperspectral camera, rainfall, NPK (Nitrogen Phosphorus Potassium) sensor (*Pathmudi et al., 2023*).

---

*Ştefania-Adriana Hoţei, MSc. Student Eng.; Cristina Ioana Marghescu, Lect.Ph.D.; Rodica Cristina Negroiu, Lect.Ph.D.; Andrei Drumea, Assoc. Prof. Ph.D.*

In the adoption of smart farming practices, the most common issues are related to initial investment costs, data volume, connectivity, and scalability (*Pathmudi et al., 2023*). Regarding the factors that facilitate the adoption of smart agriculture, the authors of Thomas and O'Hare, 2023 identified, based on a literature review, seven factors (positive impact on productivity, time and cost, information and knowledge gain, support for training, familiarity and market readiness, effective support for daily practice, accuracy and reliability, effective decision support, personalized and localized information, increased well-being) as well as six barriers (difficult or complex to use, infrastructure requirements for setup and use, lack of support, not adapted to farmers' needs, increased burden, perception of technology). In the Romanian context, initial investment costs, especially for those practicing subsistence agriculture, are perhaps the most important factor considered at the outset. The system should be easy to use and scalability is important from a sustainability perspective. With this in mind, a literature review of Low Cost Agricultural Systems was conducted. The results are summarized in Table 1 under the aspects of processing unit, sensors, communication and power supply. The keywords used were low cost & IoT& agriculture, and the databases queried were IEEE and ScienceDirect.

**Table 1**

**Low cost monitoring systems**

| Reference | Sensors | Processing Unit | Communication | Power |
|---|---|---|---|---|
| Vijayaraja et al., 2022 | Soil moisture | Node MCU | ADAFRUIT server, MQTT protocol | photo voltaic panels, battery |
| Mondal et al., 2017 | Soil moisture | ATMEGA328P, RichDuino Board | Xbee | |
| Bhattacherjee et al., 2020 | Soil moisture | Arduino pro mini | LoRA | 3:7V Li-Ion 6V solar panel |
| Gill and Albaadani, 2021 | Relative humidity and temperature, Soil Moisture Sensor, Soil Temperature Sensor, Light Intensity Sensor | NodeMCU | MQTT protocol | Lithium ion battery and solar panel |
| Kasama et al., 2019 | illuminance, temperature, humidity, pressure, and concentration of carbon dioxide | a modified Atmega328P 8-bit 8-MHz RISC microcontroller board | RF24L01+, 2.4GHz ISM RF transceiver | Ultracapacitor and solar panel |
| Valenzuela et. al, 2018 | Measure temperature, humidity, air pressure | tpanStamp AVR 2 module wireless abstract protocol (SWAP) | free ISM frequency . | AAA or AA batteries |
| John, 2016 | Soil temperature and moisture | PIC16F877A | RF module | battery |
| Heble et al., 2018 | Soil moisture Soil temperature Light intensity, relative humidity, and ambient temperature. Carbon dioxide Total solar radiation (TSR) | ATmega1281 | 802.15.4 compliant wireless mote which is developed in-house at the Wireless Networks (WiNet), LoRa | Solar panel, battery |
| Math and Dharwadkar, 2017 | temperature and humidity – DHT22 | ATmega328, Arduino Uno R3 Board | XBee | 2 AA batteries |
| Mishra et al., 2023 | Humidity and Temperature (DHT11) IR Sensor (LM393) PIR (ZRD09) Capacitive Soil Moisture | | LoRa | |
| Dai and Sugano, 2019 | photovoltaic temperature sensor STM431J provided by EnOcean | EnOcean sensor | LPWAN Sigfox | Energy harvesting |
| Kökten et al., 2020 | SCD30 sensor | STM32 microcontroller | LoRa | |

The most used sensors are soil humidity (*Vijayaraja et al., 2022; Mondal et al., 2017; Bhattacherjee et al., 2020; Gill and Albaadani, 2021; John, 2016; Heble et al., 2018; Math and Dharwadkar, 2017; Mishra et al., 2023*) and air temperature and humidity (*Gill and Albaadani, 2021; Kasama et al.,2019; Valenzuela et. al, 2018; Heble et al., 2018; Math and Dharwadkar, 2017; Mishra et al., 2023; Dai and Sugano, 2019*). XBee and LoRa are the most used communication protocols. Based on these findings a low cost IoT system for agriculture was developed.

## MATERIALS AND METHODS

### *System description*

A greenhouse or field is to be monitored using a network of sensors. The implementation of the system will consist of a number *n* of physical modules or nodes. The following requirements for the system have been defined: 1) It should measure soil moisture, air temperature, air humidity, and light intensity; 2) It should be powered either by easily replaceable batteries or be self-sufficient; 3) Data should be available online; 4) The user should be notified when certain thresholds are reached.

As stated, the proposed system targets home users who need to monitor key environmental factors by simplifying the access and use modalities. In simpler terms, in order to convince a user (a small to middle scale farmer) of the usefulness of such a solution, the resulting economic impact should be highlighted: fewer resources are consumed and the environmental impact is smaller by reducing water and thermal energy waste, maximization of production by extending the growing period, ease of automatic monitoring, which brings accurate and continuous data, and the possibility of continuous monitoring of the lot by reporting anomalies. The operation and implementation are simple, the maintenance is not very costly, only the battery needs to be changed, but with minimal cost and consumption. Light, air temperature, humidity and soil moisture sensors will be placed at key points, providing meaningful and general data about the condition of the area in question. The project will be modular so that additional sensors can be added for monitoring other environmental factors of interest, such as: soil pH, $CO_2$ concentration, solar radiation, etc. Low cost sensors have been chosen. The sensors used are presented in table 2.

**Table 2**

**Sensors used**

| To be measured | Chosen sensor | |
|---|---|---|
| Air temperature and humidity | DHT11 | Temperature range 0 °C and 50 °C |
| | | Precision ± 2 °C |
| | | Humidity range 20-80 % |
| | | Precision ± 5 % |
| | | Resolution 8 bits |
| | | Sampling rate 1 Hz |
| | DHT22 | Temperature range -40 °C and 80 °C |
| | | Precision ± 0,5 °C |
| | | Humidity range 0-100 % |
| | | Precision ± 2 % and 5 % |
| | | Resolution 16 bits |
| | | Sampling rate 0.5 Hz |
| Soil humidity | Capacitive | Accuracy 2 |
| | Resistive | Accuracy 2%, can corrode in time |
| Light | LDR | |

From the point of view of the communication protocol, in order to make it easily accessible, aspects related to the reduction of complexity in use were considered. Romania has a population of **19.18 million** as of January 2021, and **15.49 million** internet users. Internet penetration in Romania stood at **80.7%** in January 2021. Here were **26.00 million** mobile connections in Romania in January 2021 (+0.9% vs January 2020). Smart phone ownership stands at 97% (*https://datareportal.com/reports/digital-2021-romania*).

For these reasons WiFI has been chosen. Another advantage is that its high data rates would allow the addition of a webcam. After the communication solution the ESP32-DevKit1 was chosen since it had integrated WiFi communication. ESP has been used before. A study (*Tao et al., 2021*) on communication technologies in agriculture for the period 2007-2021 identified 26 papers that used WiFi, 7 of which used this solution, published after 2020. The main components of the system can be seen in figure 1. To demonstrate the functionality, four types of nodes were designed: two simple nodes using batteries, and two complex nodes based on renewable energy.

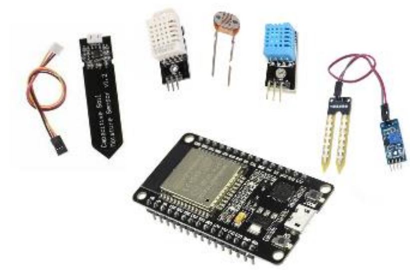


Fig. 1 - Main parts of the system

Since one problem of WiFi is its high energy consumption compared to other technologies (*Visconti et al., 2020*), it was decided to also test a renewable energy solution. There are three types of power blocks: power from a module with four AA alkaline batteries, power from photovoltaic panels, and energy stored in the Li-ion battery or in supercapacitors (ultracapacitors).

The collected data will be stored in an online Google Sheets file to facilitate its review and monitoring, avoiding mediation through a third-party cloud. Processing will consist of displaying the evolution of the collected data in the form of graphs to detect and report any anomalies.

When choosing the data storage method different IoT platforms were researched: ThingSpeak, Blynk, ThingsBoard MQTT and InfluxDB were considered but all had restriction in the free version. The best solution in the end was considered to be Google Sheets. In the case of the Google Sheets application, no additional account is required, only the Google account that most owners of a cell phone, tablet, laptop or PC with Internet access usually have. All the user has to do is connect directly to the spreadsheet where the data is to be stored from their phone or other device. From there, the code already implemented in the Google Script application for the spreadsheet takes over the entire process of receiving, retrieving, storing and displaying the data from the microcontroller. The account is free, keeping such a register is also free. And probably one of the most advantageous aspects is the fact that most people who own devices such as laptops or PCs are familiar with Microsoft's Office package, which includes Excel, which is similar to Google Sheets in its functionality and usage. The configuration possibilities of the functions are very extensive thanks to their scripting application, in which a Java Script-like code can be implemented to customize for each user.

The basic architecture of the system is presented in figure 2 and the diagram given in the figure 3 was created to illustrate the general operation of a node. It succinctly contains the main phases that a node continuously goes through in the system. To achieve effective data monitoring, the following steps must be performed: data acquisition from the sensors; data transmission via WiFi wireless communication protocol; storage of the collected data in a user-friendly app, in this case the free online application Google Sheets; putting the microcontroller into deep sleep to save energy until the next data acquisition from the sensors; and waking up the microcontroller via the internal RTC counter and restarting the cycle.

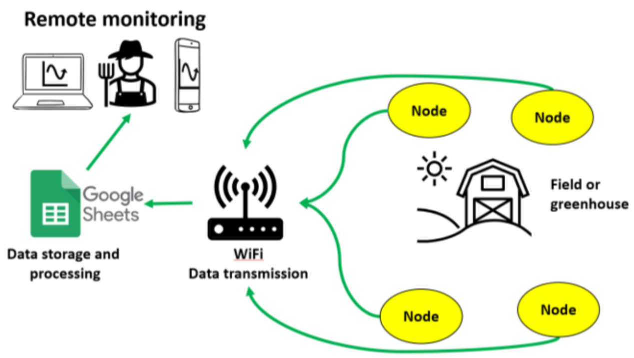



Fig. 2 - System architecture

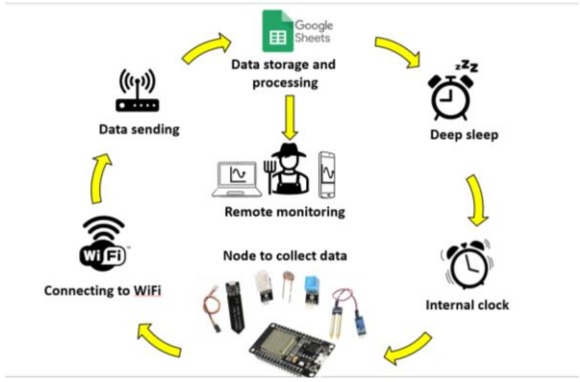
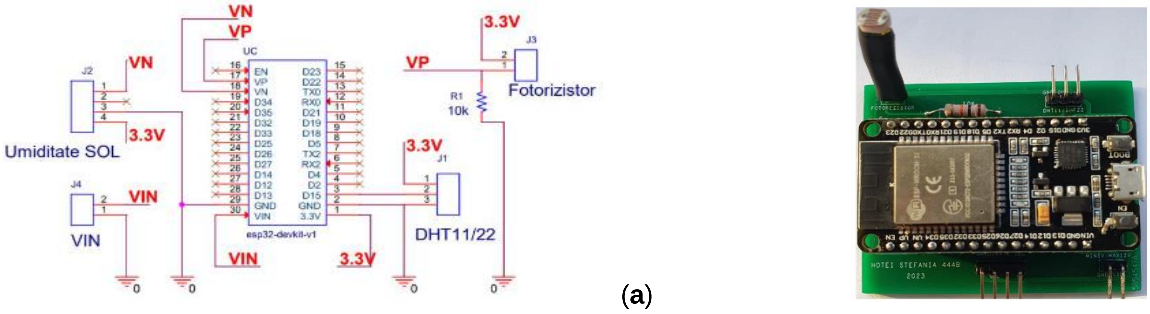



Fig. 3 - General operation

### *Hardware implementation*

The nodes were built either on printed circuit board or perfboard. An overview is given below. The nodes were powered by battery (Node 1), solar panel and EDLC (Node 2), and LiPo Accumulator (Node 3).

The schematic and PCB for Node 1 is given in figure 4.




(a) (b)

**Fig. 4 - Node 1, battery powered: schematic (a) and PCB (b)**

For the renewable energy node that uses solar panels and supercapacitors the block diagram can be observed in figure 5 the schematic and PCB in figure 6. This node is also equipped with GPS and can be placed where it is needed.

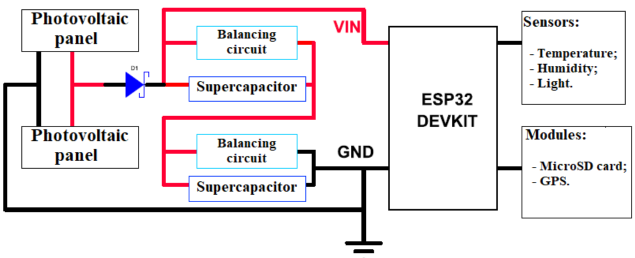
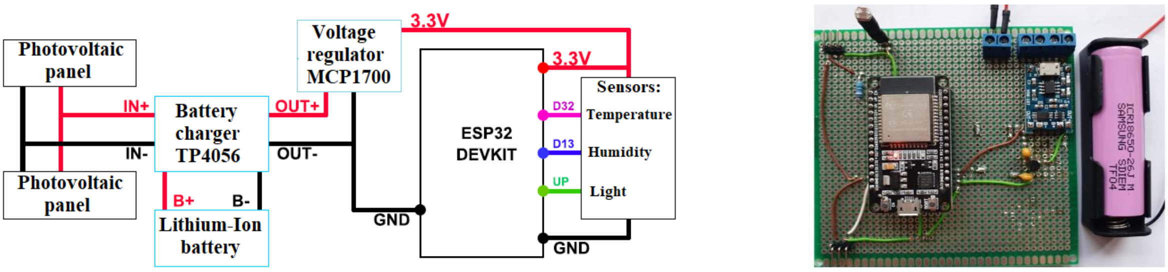



**Fig. 5 - Node 3, photovoltaic panel and supercapacitors: block diagram**

**Fig. 6 - Node 3, photovoltaic panel and supercapacitors: PCB**

The supercapacitor circuit was based on an earlier one researched by the authors (*Ionescu et al., 2016*). The last node (Node 4) used Li-Ion Accumulators.




(a) (b)

**Fig. 7 - Node 4, block diagram (a) and perfboard (b)**

An overview of the prices of the different components is given in Table 3.

**Price (when purchased) overview** **Table 3**

| Part | Price in USD per piece |
|---|---|
| ESP DevKit | 11.30 |
| DHT11<br>DHT22 | 2.10<br>5.93 |
| LDR | 0.36 |
| Soil humidity capacitive<br>Soil humidity resistive | 3.62<br>2.26 |
| Solar panel | 5.99 |
| Supercapacitors | 12.23 |
| TP4056 | 1.77 |
| MCP1700 | 1.15 |
| Li-Ion | 9.29 |
| Other components | 5.10 |
| Perfboard | 1.36 |

*Node 1 had an approximate price of 17 USD, Node 2 21.21, Node 3 35.44 USD.*

The power consumption per component is concentrated in Table 4.

**Table 4**

**Power consumption**

| Part | Operating current |
|---|---|
| ESP DevKit<br>- Deep sleep ultra-low power<br>- RTC counter + RTC memory | 240 mA<br>150 µA<br>10 µA |
| DHT11, DHT22 | 2.5 mA |
| Soil humidity<br>- Resistive<br>- Capacitive | <br>< 7 mA (maxim - 35 mA)<br>5 mA |
| SD Card | 0,2 mA – 200 mA |
| GPS | 45 mA |

***Source code and Google Sheets***

The source code implemented on the microcontroller deals only with the data acquisition part from the sensors (figure 8), the management of the node activity, interpretation and implementation of alarms in case of anomalies is handled by the source code implemented in the Google Script application for several reasons. A first reason is the limited memory of the microcontroller, so only what is necessary is processed. Another reason is the large number of possibilities to implement the desired functions through the Java Script language adapted to Google Script.

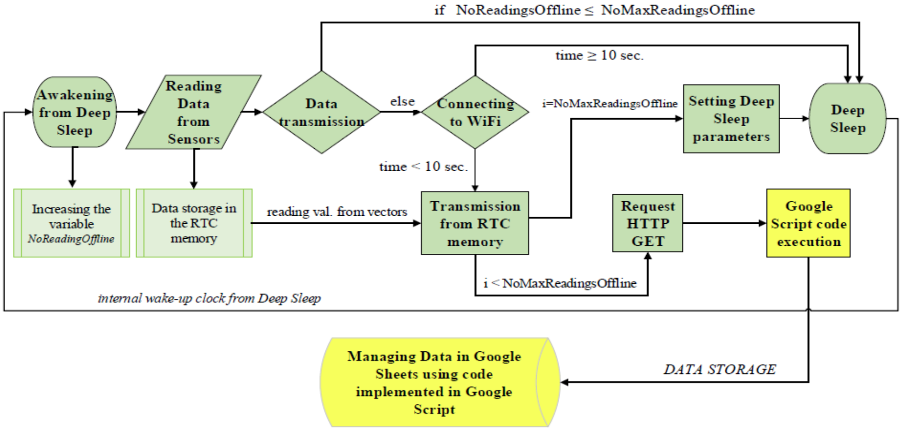



**Fig. 8 - Microcontroller source code organigram**

The process of retrieving data from the microcontroller is handled by the source code implemented in Google Script (figure 9). The language is based on JavaScript and provides a way to write custom scripts that can interact with Google services and supports event-driven programming, where scripts can be triggered automatically based on events such as form submissions, time-based triggers or changes in data spreadsheets.

The request made by the microcontroller, activates the doGet() method that takes the event e as a "string" of elements. In order to be able to enter the data to be monitored into a single computation register, but into spreadsheets for each node, the request must also contain a distinguishing feature for that node. So the first step after triggering the doGet() method is to check the source of the request. This is done in the "Verification no. Node" decision block, where, depending on the value entered in the "Node No=" parameter, the "sheetName" variable is assigned the name of the spreadsheet in which the values associated with the node in question are entered.

This procedure was necessary in order not to limit the user to the use of the network of nodes in a given space. Therefore, correlation of the obtained values should preferably be done in a separate spreadsheet. After deciding to which spreadsheet the values should be assigned the event containing the data is repeated and the values of interest are stored in a value buffer, each in the position it will physically occupy in the spreadsheet. After the "Extract values" action has been performed, the "Assignment in spreadsheet" step of these values follows.

Separately implemented methods were "Mark values below/above threshold" and "Format background colors". The first method takes the current vector of values and checks to see if each cell falls within the range of minimum and maximum thresholds that the user has selected in the table provided. If it does, nothing happens; if at least one of the values is out of range, the text "Attention!" is displayed in the last column of the table. The second method uses the same checking logic, but in case of mismatched values, this time of the whole worksheet, it colors the background of the mismatched cells bright red, so that the user can recognize them more easily and then make a decision.

The third implemented method deals with the part that sends a warning to the user when one of the examined parameters is out of the allowed range. The logic used only considers the number of selected transmissions. For example: the system has a complete cycle of 2 hours. This means that the microcontroller is brought out of deep sleep every 30 minutes to retrieve the data and transmits the data to the server on the fourth wake-up. This aspect drastically reduces the consumption, because the biggest consumer is the WiFi connection and the data transfer. Since the data in this example is sent in four sets, the method designed for "Send Alert!" does not check the first three sets of data, but only when the fourth is entered, it checks all the four datasets. If it detects irregularities, it sends the user an email with a warning that includes: the name of the node, the line(s) with the relevant parameter(s), the link to the node's worksheet, the graphics included in the worksheet. The process of data storage is performed in the block "Assignment in Spreadsheet" of the source code organization chart (Figure 9). At the end of the data transfer cycle, at least one new row is displayed in the spreadsheet on the node that transferred the data. After this block is executed, the data can be displayed in the spreadsheet.

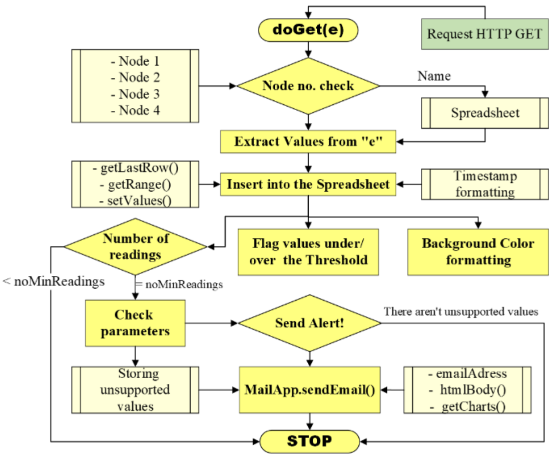



**Fig. 9 - Google Script source code organigram**

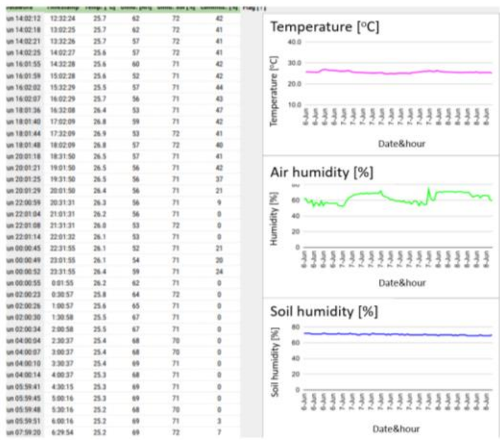



**Fig. 10 - Received data displayed in table format as well as charts**

The charts are displayed in real time and updated with each further iteration of the entered data and are available to the user at any time. They are displayed as seen in Figure 10. Each collected data set is assigned the exact time of collection. This is possible thanks to the RTC (Real Time Counter) module. There is also a column with the date and time when the iteration was inserted into the worksheet, to also track the process of automatic value insertion. Conveniently, monitoring from this point on can be done either by viewing the collected data or by using the worksheet that has the task of summarizing the data collected from all the nodes involved in the process; in the test case, all four nodes are monitored simultaneously in the "Total" worksheet (Fig.11).

Obtained results

| Node no. 1 | | | | Node no. 2 | | | | Node no. 3 | | | | Node no. 4 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Parameter | Min. | Max. | Mean | Parameter | Min. | Max. | Mean | Parameter | Min. | Max. | Mean | Parameter | Min. | Max. | Mean |
| Temperature [°C] | 24.5 | 26.7 | 25.45 | Temperature [°C] | 24.7 | 26.9 | 25.61 | Temperature [°C] | 24.8 | 27.1 | 25.24 | Temperature [°C] | 24.5 | 27.1 | 25.3 |
| Air humidity [RH] | 40 | 62 | 29.29 | Air humidity [RH] | 52 | 74 | 62.63 | Air humidity [RH] | 42 | 66 | 51.11 | Air humidity [RH] | 42 | 59 | 48.11 |
| Soil humidity [%] | 70 | 90 | 73.16 | Soil humidity [%] | 69 | 72 | 70.37 | Soil humidity [%] | 28 | 74 | 69.14 | Soil humidity [%] | 60 | 72 | 68.04 |
| Light intensity [%] | 0 | 57 | 26.24 | Light intensity [%] | 0 | 51 | 25.57 | Light intensity [%] | 0 | 54 | 18.61 | Light intensity [%] | 0 | 56 | 12.89 |
| Alert no. [!] | 1 | | | Alert no. [!] | 12 | | | Alert no. [!] | 2 | | | Alert no. [!] | 4 | | |

**Fig. 11 - Visualization of the summarized data of all nodes**

In this worksheet, some Google Sheet-specific calculation formulas are implemented to take data from other worksheets and determine the minimum, maximum and average values of each parameter.

The downloaded file can also be stored in the cloud, which is also offered through Google services, but with the disadvantage that the corresponding storage space is limited in terms of memory size and cannot be the only option for long-term data storage. In this way, reports can be generated as often as the user wishes:

daily, weekly, monthly or yearly, emailed in the form of a .pdf file, and alerts are sent when the values exceed thresholds chosen by the user. This type of report is useful and necessary for long-term data monitoring. These data are related to the period of the agricultural season, the general environmental factors in the country, the type of crop and so on. Manual downloading of files can be done in several ways, depending on the needs of the user: Microsoft Excel, PDF, web page, CSV, TSV. They can be imported back into the online book at any time to correlate values or continue observations. A separate spreadsheet has been created in which the threshold values for each node are entered, so that the user has the possibility to change these parameters in real time according to his needs. Every node has a specific color to be easily distinguishable from other nodes and to complement the graphical interface the user interacts with.

## RESULTS

In figure 12 the thresholds set can be seen, and non-compliant values were found for node number one when the values were inserted into the spreadsheet. This represents the worst case: The entire data collection and transmission cycle takes two hours, with data retrieved every half hour and all four captures transmitted, and the warning message is not sent until 90 minutes after the first deviation in values. This scenario was chosen to illustrate that a gradual decline in soil moisture values over the chosen time interval is not destructive. Certainly, a shorter cycle is required for "demanding" crops, and a longer cycle is possible for crops that are less sensitive to gradual changes over time.

THRESHOLDS SET BY THE USER

e-maila address @gmail.com

| Node no. 1 | | | Node no. 2 | | | Node no. 3 | | | Node no. 4 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Parameter | Min. | Max. | Parameter | Min. | Max. | Parameter | Min. | Max. | Parameter | Min. | Max. |
| Temperature [°C] | 20 | 30 | Temperature [°C] | 21 | 30 | Temperature [°C] | 18 | 30 | Temperature [°C] | 18 | 30 |
| Air humidity [RH] | 30 | 70 | Air humidity [RH] | 30 | 80 | Air humidity [RH] | 30 | 80 | Air humidity [RH] | 30 | 70 |
| Soil humidity [%] | 40 | 90 | Soil humidity [%] | 40 | 90 | Soil humidity [%] | 30 | 90 | Soil humidity [%] | 30 | 90 |
| Light intensity [%] | 0 | 100 | Light intensity [%] | 0 | 100 | Light intensity [%] | 0 | 100 | Light intensity [%] | 0 | 100 |
| Transmission cycle | 4 | | Transmission cycle | 4 | | Transmission cycle | 4 | | Transmission cycle | 4 | |

**Fig. 12 - Thresholds chosen by the user (interface in Romanian with translation of the relevant parts)**

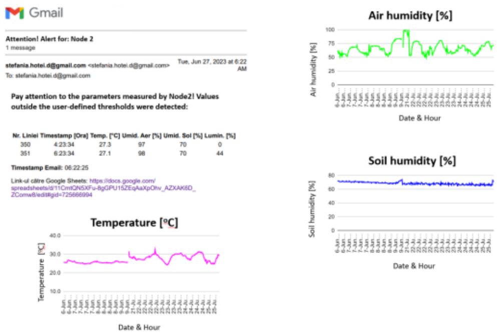



**Fig. 13 - Example of alert mail received by the user**

When the user receives the email on their phone, they can access the spreadsheet for that node to review the values and make decisions on how to proceed. An example of displaying non-compliant data can be the one that is described by the results shown in figure 13. If several parameters exceed the threshold at the same time, they are displayed in the same way: A warning "flag" is set in the last column of the table (regardless of whether only one or more values require a flag, only one is set), the background of the cell(s) is colored to highlight the out-of-bounds value(s), and the same email is sent with one or more rows, depending on the number of affected values. In this way, it can be seen the impact achieved by reducing quantity of the transmitted data to be stored in the database for monitoring.

A comparison of the nodes whose individual power block consists of four alkaline batteries of 1.5 V each is given in table 5.

**Table 5**

**The comparison of the nodes**

| Cycle of continuous transmissions | Cycle of 4 transmissions |
|---|---|
| - once every 30 minutes. Data acquisition occurs every 30 minutes, and transmission occurs immediately thereafter, skipping the step of storing the data in RTC memory. Approximately 115 hours, 4.5 days of continuous operation. | - time between captures of 30 minutes, so data transmission occurs every 120 minutes, when the last iteration of data is captured, WiFi connection is established, and data packets are sent to Google Script. Approximately 240 hours, 10 days of continuous operation - node 1 second type of batteries approximately 120 hours, 5 days of continuous operation. |

In this way, it can be seen the impact achieved with the help of reducing the number of connections to the access point and thus sending data to the stored database for monitoring.

As part of the analysis of the node powered by the supercapacitor battery, it was investigated how long the power block could withstand a long period of use, whether it was suitable to power the node at night or on less sunny days, and finally whether it was a supported choice in the current project.

The circuit was tested under shade charging conditions, where charging was slower, for a few hours over the course of a day. In the sun, the total charge requires about 0.5 hours of direct sunlight, with the two photovoltaic cells connected in parallel. The module built and tested is enough to cover the user's needs during a whole sunny or at least bright day. When it gets dark, the supercapacitors gradually and relatively quickly lose their stored energy. In one of the tests, collecting data every 30 minutes and transferring data every 2 hours (on the 4th transfer), only one data transfer was successful, but probably maxed out until the second, when reaching the operating threshold voltage of 4.4 V at the development board terminals after 3 hours. In another iteration of the tests, a continuous transmission of 75 minutes was achieved by immediate fetching and sending at 15 second intervals.

## CONCLUSIONS

As mentioned at the outset, the system is aimed at subsistence farmers. In the testing process, different cases and scenarios of using the designed system were implemented. The general conclusion about the first two types of nodes implemented is that they offer a wide range of possible applications depending on the needs of the user. The system can be programmed to collect data at short intervals, which has the disadvantage of consuming batteries faster, but on the other hand, it can be scheduled to transmit as infrequently as possible. The overall operation of the two nodes is correct. However, in order not to invest too much money in batteries all the time, it was decided to test the operation of the system with renewable energy. This method did not prove to be particularly useful in terms of nighttime operation. In the case of the supercapacitor battery node, power cannot be maintained for more than 2 hours after the solar cells are turned off after dark. Therefore, this type of node is best suited for use outside the greenhouse, somewhere in the field or even in the garden outside the greenhouse, that is, in a place where monitoring is necessary during the day and not necessarily at night. The node works well under these conditions. It needs about 30 minutes of direct sunlight in the morning and then if it is a cloudy day to remain functional throughout the day. Transmission would occur each time it wakes from deep sleep so as not to risk losing power until the next iteration. In this way, only the time of day would be monitored, which in Romania is between 12 and 14 hours of light per day for the growing phase of most local crops.

The collected data can be analyzed at any time and using on any device and can be stored locally for future checks. For example, if the user has a record from the previous season, he can keep it locally and release the calculation book for the current season. Another method is to create a new online register to maintain consistency so that data obtained between seasons and physical harvest results can be automatically compared. This kind of automation of the process helps the user to better understand what they need for their crop, or even to generally adopt the data suggested by other studies and keep the environmental factors in the same register of values. To increase the covered area and if the nodes need to be placed further away from an Internet access point, it is possible to use a mesh network.

## REFERENCES


[1] Bhattacherjee S. S., Shreeshan S., Priyanka G., Jadhav A. R., Rajalakshmi P., Kholova, J. (2020, March), Cloud based Low-Power Long-Range IoT Network for Soil Moisture monitoring in Agriculture, Proceedings of the *2020 IEEE Sensors Applications Symposium (SAS),* IEEE, Kuala Lumpur, Malaysia, pp. 1-5. doi: 10.1109/SAS48726.2020.9220017

[2] Dai J., Sugano, M. (2019, October), Low-Cost Sensor Network for Collecting Real-Time Data for Agriculture by Combining Energy Harvesting and LPWA Technology, Proceedings of the *2019 IEEE Global Humanitarian Technology Conference (GHTC),* IEEE, Seattle, WA, USA, pp. 1-4. doi: 10.1109/GHTC46095.2019.9033114

[3] Gill R., Albaadani A. N. H., (2021, October), Developing a Low Cost Sensor Node using IoT Technology with Energy Harvester for Precision Agriculture, Proceedings of the *2021 2nd International Conference on Smart Electronics and Communication (ICOSEC),* IEEE, Trichy, India, pp. 187-194. doi: 10.1109/ICOSEC51865.2021.9591849

[4] Heble S., Kumar A., Prasad K. V. D., Samirana S., Rajalakshmi P., Desai U. B., (2018, February), A low power IoT network for smart agriculture, Proceedings of the *2018 IEEE 4th World Forum on Internet of Things (WF-IoT),* IEEE, Singapore, pp. 609-614. doi: 10.1109/WF-IoT.2018.8355152.

[5] Ionescu C., Vasile A., Negroiu R., (2016, May), Investigations on balancing circuits for supercapacitor modules, Proceedings of the *2016 39th International Spring Seminar on Electronics Technology (ISSE)*, Pilsen, Czech Republic, pp. 521-526. doi: 10.1109/ISSE.2016.7563253

[6] John G. E., (2016, December), A low cost wireless sensor network for precision agriculture, Proceedings of the *2016 Sixth International Symposium on Embedded Computing and System Design (ISED),* IEEE, Patna, India, pp. 24-27. doi: 10.1109/ISED.2016.7977048

[7] Kasama T., Koide T., Bula W. P., Yaji Y., Endo Y., Miyake R., (2019, March), Low Cost and Robust Field-Deployable Environmental Sensor for Smart Agriculture, Proceedings of the *2019 2nd International Symposium on Devices, Circuits and Systems (ISDCS)*, IEEE, pp. 1-4. doi: 10.1109/ISDCS.2019.8719262

[8] Kökten E., Çalışkan B. C., Karamzadeh S., Soyak, E. G. (2020, October), Low-powered agriculture IoT systems with LoRa, Proceedings of the *2020 IEEE Microwave Theory and Techniques in Wireless Communications (MTTW)*, Vol. 1, IEEE, Riga, Latvia, pp. 178-183. doi: 10.1109/MTTW51045.2020.9244927

[9] Math R. K., Dharwadkar N. V., (2017, January), A wireless sensor network based low cost and energy efficient frame work for precision agriculture, Proceedings of the *2017 International Conference on Nascent Technologies in Engineering (ICNTE),* IEEE, Vashi, India, pp. 1-6. doi: 10.1109/ICNTE.2017.7947883

[10] Mishra S., Nayak S., Yadav R., (2023, January), An Energy Efficient LoRa-based Multi-Sensor IoT Network for Smart Sensor Agriculture System, Proceedings of the *2023 IEEE Topical Conference on Wireless Sensors and Sensor Networks*, IEEE, Las Vegas, NV, USA, pp. 28-31. doi: 10.1109/WiSNeT56959.2023.10046242

[11] Mondal A., Misra I. S.,Bose, S., (2017, April), Building a low cost solution using wireless sensor network for agriculture application, Proceedings of the *2017 International Conference on Innovations in Electronics, Signal Processing and Communication (IESC)*, IEEE, Shillong, India, pp. 61-65. doi: 10.1109/IESPC.2017.8071865

[12] Pathmudi V. R., Khatri N., Kumar S., Abdul-Qawy A. S. H., Vyas A. K. (2023), A systematic review of IoT technologies and their constituents for smart and sustainable agriculture applications, *Scientific African*. https://doi.org/10.1016/j.sciaf.2023.e01577

[13] Tao W., Zhao L., Wang G., Liang R., (2021), Review of the internet of things communication technologies in smart agriculture and challenges, *Computers and Electronics in Agriculture*, *189*, 106352, Vol. 189, 106352, ISSN 0168-1699. https://doi.org/10.1016/j.compag.2021.106352

[14] Thomas R. J., O'Hare G., Coyle, D., (2023) Understanding technology acceptance in smart agriculture: A systematic review of empirical research in crop production. *Technological Forecasting and Social Change*, 189, 122374. DOI: 10.1016/j.techfore.2023.122374

[15] Valenzuela A. A., Schwab M., Silnik A. A., Debattista A. F., Kiessling R. A., (2018, October), Low Power Wireless Sensor Node Platform for Agriculture Monitoring in Argentina, Proceedings of the *2018 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC)*, IEEE, Zhengzhou, China, pp. 101-1014. doi: 10.1109/CyberC.2018.00029

[16] Vijayaraja L., Dhanasekar R., Kesavan R., Tamizhmalar D., Premkumar R., Saravanan N., (2022, April), A Cost Effective Agriculture System based on IoT using Sustainable Energy, Proceedings of the *2022 6th International Conference on Trends in Electronics and Informatics (ICOEI)*, IEEE, Tirunelveli, India, pp. 546-549. doi: 10.1109/ICOEI53556.2022.9776726

[17] Visconti P., de Fazio R., Velazquez R., Del-Valle-Soto C., Giannoccaro N.I., (2020), Development of Sensors-Based Agri-Food Traceability System Remotely Managed by a Software Platform for Optimized Farm Management, *Sensors 20 (2020)* 3632. https://doi.org/10.3390/s20133632