SMART-technologies in Irrigation Management of a Remote Land Plot

O V Matukhina¹*, L R Votyakova¹, I N Zakharova¹
¹Nizhnekamsk Institute of Chemical Technology, Nizhnekamsk, Russian Federation

E-mail: ovmatukhina@gmail.com

Abstract. The purpose of this work is to design an automatic irrigation system on a remote land plot. Along with this the objectives of the designed system were the requirement of intelligence and interactivity. In other words the automation system should not only open the solenoid valves according to the schedule, but also control the separate zones taking into account the temperature, soil humidity, weather, etc., while notifying the user of events. The result of this work is the project of an interactive irrigation management system with the possibility of remote interaction. Besides there is a developed client-server application that implements this remote interaction in a simple and familiar way through a mobile application for a smartphone running the Android operating system. It is easy to configure flexibly or to expand the system to suit any needs, to use completely different sensors and actuators, to receive reports on stable operation and planned actions set by algorithms, or on abnormal situations, such as device failure. It is worth noting as an advantage the flexibility and scalability of the developed system in terms of integration with any information management system and digitalization of agriculture.

1. Introduction

Today the intelligent systems in agriculture can help in different ways. They can be placed in a house or participate in growing plants, work on the proper functioning of the pool or with cottage lighting.

Let's analyze where one can apply automation primarily. Of course, the priorities of control without human intervention are different for everyone, let's consider suburban plantations.

The crops require regular watering. You can provide it from a conventional hose or diffuser, but only at the time of presence in the country. What to do if the weather is hot and sunny the whole week, and you will be able arrive to the cottage only by the weekend? The only way out is an intelligent irrigation system which will help either by means of a pre-established program, or depending on the temperature outside, will turn on watering.

In essence any automation systems are similar and are a simplified version of the automation of the technological process at the enterprise, although their functionality is limited only by the imagination of the developer and the requirements of the customer.

When using any kind of automation, the question of storing the information arises. If the system is quite complex, works with a relatively large amount of data, as well as to improve its reliability, or just for a more convenient display of information, there is a definitely need to store data.

The purpose of this work is to develop an automated irrigation management system on a remote agricultural site, which can be integrated with the information management system and digitalization of agriculture in the region.
2. Materials and method

Automation is a worthy alternative to manual labor, but an Autonomous system, even if it is simple and cheap, compared to an interactive one, is not able to solve a huge number of management tasks and timely information. Electronics and a "smart" system that replaces a person in his absence must have such characteristics as self-diagnosis, transition to a safe state and notification in case of malfunction of its components. For example, if any component of the Autonomous system fails, this will only be known when you arrive at the site.

And when we speak about smart technologies, the requirements for their functionality are set accordingly. Hardware failure is an unavoidable event that occurs in systems of any level of reliability.

Thus, if a critical failure of the microcontroller responsible for its work area, using Autonomous automation, the owner learns about its failure only upon arrival at the site.

It is unknown how much time will pass from the moment of equipment failure and what the consequences will be in case of prolonged inactivity. When implementing a system with remote notification, even without any self-diagnostics and additional devices that increase reliability (for example, leak protection), it will be clear that the connection to a specific control unit is lost and this is an excuse to visit the site and find out what is wrong. Notification can be configured not only for critical situations, but also, for example, for the beginning/end of control actions. This will further increase the level of reliability because in an unforeseen situation, if the next notification was not received – this is the reason for logical analysis and further analysis.

Therefore notification of the operation/incidents of the automation system by means of SMS messages, calls, email messages, push notifications in the mobile app, and so on, increases the level of reliability of the automated complex and gives less cause for concern to the person using such a system.

Thus, the main feature of the developed product will be the ability to remotely manage / receiving notifications and an easy way to interact with the user – using a mobile app.

The software product for Android OS will allow to "communicate" with a person in a more convenient, simple and familiar format that does not require studying additional information on configuring the control device.

The proposed scheme works as follows. The user interacting with the mobile app, make changes to the algorithm, for example, increases the duration of watering in the greenhouse. The new parameters are passed to the server. The server also sends the changed settings to the main controller via the Internet.

A controller with a connected GPRS module receives them, saves them, and transmits them to an auxiliary controller responsible for a specific zone, in our case, a greenhouse, via Wi-Fi via the ESP8266 module. The MC that controls the greenhouse receives the information, saves it to non-volatile memory, and makes changes to its control algorithm, then returns a response to the main control device about performing (or not performing) this operation.

This message is then sent to the server, and then to the client side of the app on the user's smartphone as a notification.

When the user sees the confirmation notification, he is assured that the changed information has been transmitted and accepted for execution. So, the automation system is being interactive with the possibility of remote interaction. At the same time nothing restricts the functionality and the entire complex can be expanded as much as you want only by adding or replacing its individual components.

Let's look at the selection and technical characteristics of the used devices. A single-Board raspberry Pi 3 Model b computer is used as the server. The Quad-Core processor operates at a frequency of 1200 MHz, which guarantees a sufficient level of performance. The device has HDMI and MIPI (CSI) interfaces for connecting external image display devices. The most popular connectors are provided for working with external devices. These are 4 USB 2.0 ports and 40 General-purpose I/o (GPIO) pins, you can connect peripherals to them: Executive devices, any sensors, and anything that runs on electricity. RAM memory of 1 GB SDRAM.
A video card of a similar production works based on the amount allocated from RAM. There is a Wi-Fi module and an Ethernet network adapter (RJ-45) that operates at a speed of 100 Mbit/s. It is managed by the Raspbian operating system based on Debian (Linux). Linux-like operating systems have become widespread due to their use as a server OS. They are safe, efficient, and easy to administer.

The control devices for direct control, collection and processing of information are the Arduino Uno and Nano. Arduino is a convenient platform for rapid development of electronic devices based on ATmega microcontrollers. The device is programmed via USB without the use of programmers.

The master controller communicates with the server using SIM800L. Communication between the main MC and the managed ones is provided by Wi-Fi modules ESP8266, forming a local network between them. GSM/GPRS communication module based on SIM800L developed by SIMCom Wireless Solutions. The standard management interface of the SIM800L component provides access to GSM/GPRS 850/900/1800/1900MHz network services for sending calls, SMS messages and exchanging digital GPRS data.

The ESP8266 chip is one of the most popular tools for organizing wireless communications in smart home projects. With the wireless controller, you can organize communication over the Wi-Fi interface, providing Arduino projects with Internet access and the ability to remotely control and collect data. Module programming is available in the Arduino IDE.

To measure the water level in the storage tank, an ultrasonic sensor HC-SR04 is used. The type of signal from the sensor is digital, this type of signal simplifies data processing, since it does not use an ADC, for example, in the case of an analog signal, it would first have to be converted to a digital value using the resources of the microcontroller and spending time, although it is not critical in this case. Installation involves installation above the water surface, excluding the direct contact with the liquid, this method will increase reliability, as it will prevent oxidation or fouling of the sensor with plaque.

The sealed temperature sensor is based on the popular DS18B20 chip. It allows to determine the ambient temperature in the range from -55°C to +125°C and receive data as a digital signal with 12-bit resolution over the 1Wire Protocol. This Protocol allows to connect a huge number of such sensors using only 1 digital port of the controller, the sensor has only 3 wires: +, - and signal.

Each sensor has a unique 64-bit code that can be used by the microcontroller to communicate with a specific sensor on a shared bus.

The code of a separate sensor can be read by a separate command. In the DS18B20's permanent memory, you can store temperature limit values, which will cause the sensor to go into alarm mode when it exits.

On a common bus of many sensors, the microcontroller can find out at a time which of them have switched to this mode. This makes it easy to identify the problem area in a controlled environment. The reading resolution is configurable and can range from 9 to 12 bits. Lower resolution means higher conversion speed.

DHT22 is responsible for measuring air temperature and humidity. This is a digital temperature and humidity measurement module based on the DHT22 sensor (AM2302), which has a higher accuracy and a wider measurement range than the DHT11. It can be used to detect ambient temperature and humidity using a standard single-wire interface. The main meteorological parameter for predicting precipitation is atmospheric pressure. It will be measured by the atmospheric pressure sensor BMP280.

Corrosion-resistant capacitive soil moisture sensor V1.2. The measuring principle is based on the capacitance change of the environment in which it is located.

In an agricultural land, electricity can be supplied with drawdowns, less often there are short-term outages, usually lasting no more than 4 hours. However, the task of the system being developed is to achieve maximum reliability, which cannot be achieved if communication with the main controller and its communication modules is lost. To ensure constant and stable operation, THE Falcon EYE FE-1220 uninterruptible power supply is used, the battery capacity of which is sufficient for several days of operation of the equipment powered by it, even in the absence of a complete voltage in the network.

Software design. The server used to store and process data as well as to ensure access to it from many different devices, shall run under appropriate software. This software consists of an operating system
(OS) and applications running by a set algorithm and performing strictly assigned functions. As specified above, the server runs under Raspbian OS (a Debian-based Linux distribution). To receive incoming data, process, send it to MySQL server for storage, and select it from the database (DB), ensure communication between users and the controllable object – all these are the server tasks of being developed software product. The server application is developed in the Java programming language, which allows you to run it with any device and any OS. The development is resulted in a console application. The software product operation does not require graphical interface. The incoming requests as well as the error messages are displayed on the console and stored in the logs. As seen in the figure 1, messages contain time of request and IP address of the sending device.

![Figure 1. The north application.](image)

The target OS for the mobile app is Android version 8.0 and higher.

When you launch the app, the main window is displayed (shown in figure 2), with buttons for accessing trends and app settings located in the header.

The settings window (shown in figure 3) allows to edit the IP address and port of the server; enable and disable notifications, select the frequency of their verification; change the description of each watering zone. The made changes take the effect immediately after you return to the main window.

When the app is launched, the latest weather information is uploaded to the corresponding field and the current irrigation status for each zone. The application implements the following watering modes: off, on, scheduled, and humidity-lowering.

The "Off" state corresponds to the complete shutdown of automatic watering. "On" - the so-called manual mode, is equivalent to a simple opening of the valve, but with the difference that this happens remotely. This mode is convenient to be used for checking the system health and debugging automatic modes. "On schedule" means that watering starts every day at the selected (fixed) time and with a set duration. The window with the trends updates the chart data every time it is opened, thus showing the most up-to-date information at the moment. At the user's request, each chart can display either one or several curved lines that visually show the dynamics of the corresponding measurement parameter. All application windows work in both portrait and landscape orientation.
Figure 2. The main window of the mobile application

Figure 3. Settings window

The application works in different display orientations as shown in figures 4 and 5- portrait and landscape, respectively, on the example of the trends window.

Figure 4. Trends window in portrait orientation

Figure 5. Trend window in landscape orientation

3. The results

The main purpose of this work was to design an automatic irrigation system on a remote land plot. At the same time the objectives of the designed system were the requirement of intelligence and interactivity. In other words the automation system should not only open the solenoid valves according to the schedule, but also control the separate zones taking into account the temperature, soil humidity, weather, etc., while notifying the user of events.

After conducting a market research of irrigation automation systems, it was concluded that the most part of the control devices are offered in a functionally simple non-extensible standalone version, which does not meet the goal.

The result of this work is the project of an interactive irrigation management system with the possibility of remote interaction. And also a developed client-server application that implements this remote interaction in a simple and familiar way through a mobile application for a smartphone running the Android operating system.
It is easy to configure flexibly or expand the system to suit any needs, use completely different sensors and actuators, receive reports on stable operation and planned actions set by algorithms, or on abnormal situations, such as device failure.

It is worth noting as an advantage the flexibility and scalability of the developed system in terms of integration with any information management system and digitalization of agriculture.

4. References
[1] Igo T 2015 Arduino, sensors and networks for device communication (Saint Petersburg: BHV-Petersburg) 544 p
[2] Rechinsky A V, Sergeev S F 2012 Development of user interfaces. Usability testing of interfaces of information systems (Saint Petersburg: Politehn Publishing house. UN-TA) 145 p
[3] Sergeev S F 2013 Methods of testing and optimization of information system interfaces (Saint Petersburg: ITMO UNIVERSITY) 117 p
[4] Horstmann K S 2014 Java SE 8 Introductory course (Moscow: Williams) 203 p
[5] Monk S 2012 Programming Arduino: Getting Started with Sketches (McGrawHill) 177 p
[6] Blum J 2015 Studying Arduino. Tools and Techniques of Technical Wizardry (Saint Petersburg: BVKh - Peterburg Publ.) 336 p
[7] GOST 7.1-2003 2010 SIBID Bibliographic record Bibliographic description General requirements and rules for compilation (Moscow: Standardinform)
[8] GOST 2.105-95 2011 Unified system of design documentation (ESKD) General requirements for text documents (Moscow: standardinform)
[9] Sanitary and epidemiological rules and regulations of the SanPiN 2.2.2/2.4.1340-03 Hygienic requirements for personal electronic computers and work organization (approved by the Chief state sanitary doctor of the Russian Federation on may 30, 2003 with changes and additions from: April 25, 2007, September 3, 2010) Mode of access: http://base.garant.ru/4179328/
[10] Habr Russian-language web site with elements of a news site, analytical articles, thoughts related to information technologies [Electronic data] Mode of access: https://habr.com/
[11] Wikipedia-free encyclopedia [Electronic data] Mode of access: https://ru.wikipedia.org/
[12] Guidelines for the hygienic assessment of working environment and labor process factors. Criteria and classification of working conditions: Manual P 2.2.2006-05 [Electronic data] Mode of access: http://base.consultant.ru/cons/cgi/online.cgi?req=doc;base=LAW;n=8553 7
[13] Pros and cons of Java programming [Electronic data] Mode of access: https://nuancesprog.ru/p/2234
[14] Bakanov A S 2011 Ergonomics of the user interface (Moscow: Institute of psychology of the Russian Academy of Sciences) 176 p
[15] Arduino Nano V 3.0 Board: Pinout, schematics, driver [Electronic data] Mode of access: https://arduinoconfiguration.ru/platy-arduino/plata-arduino-nano/
[16] Automatic irrigation system: how to make automatic watering [Electronic data] Mode of access: https://dachaklub.ru/sistema-avtopoliva-foto-video-kak-sdelat-avtopoliv-na-dache.html/
[17] How to deploy a smart automatic watering system at the dacha [Electronic data] Mode of access: https://zoom.cnews.ru/publication/item/62642/
[18] Smart farming: Overview of leading manufacturers and technologies [Electronic data] Mode of access: https://geoline-tech.com/smartfarm/
[19] Overview of digital technologies for the agro-industrial complex: from GIS to the Internet of things [Electronic data] Mode of access: http://integral-russia.ru/2020/03/20/tsifrovaya-platforma-razvitiya-agropromyshlennogo-kompleksa-konteptsiya-i-osnovnye-tezisy/
[20] Automatic watering of lawn and crop plants [Electronic data] Mode of access: https://smart-poliv.ru

Acknowledgment
The work was carried out with the financial support of the RFBR, project no. 19-08-00261.