Autommated Terrarium Implementation

Tomáš Bača, Ján Šumský, Michal Kubaščík, Peter Šarafín

Abstract— The automated terrariums represent an interesting application for automatization. The goal of the work presented in this article was the development of an automated terrarium that could control the temperature and humidity inside the terrarium. The automation also consisted of automatic watering of the animals inside the terrarium. The work was later on improved by the addition of the Wi-Fi module and the creation of a web page through which the user could set the required and monitor the current temperature and humidity.

Keywords— automatization, atmega328P, electromagnetic valve, heating resistors, LCD display, bang-bang regulation, PID regulation

I. INTRODUCTION

The goal of this article is to present the result of the work which was focused on the creation of such an automated terrarium. The automatization consisted of the automatic temperature and humidity control system. And the automatic watering of the animals living in the terrarium. The work was extended by the addition of the wireless communication module, which was running a web page through which the user could interact with the automated terrarium.

II. DEVELOPMENT OF AUTOMATED TERRARIUM

This paper is focused on the development of small automated terrariums that used as box and aquariums. The main goals from the point of view of automatization were the automatic humidity and temperature control to the values set by the user, the automatic watering of the animal inside of the terrarium and the creation of a simple user interface with the use of buttons, rotary encoders and an LED display capable of showing to the user 32 symbols at the same time.

In this work was also designed the printed circuit board (PCB) at which were placed all the SMD components and to which were connected by wires certain sensors and action elements.

Basic functionality was extended by connecting the originally used printed circuit board with the microcontroller ATMega328P to a module containing the microcontroller ESP32-S1 which supported wireless communication. And creating a web page through which the temperature and humidity could be set and controlled. This web page served as a graphical user interface and it turned the developed automated terrarium into an IoT application.

A. Sensors

In the automated terrarium development two different sensors were used.

The first sensor was the DHT11 [1], this sensor measured the temperature and humidity inside of the terrarium. This sensor was communicating with the microcontroller ATMega328P through a proprietary communication protocol.

The second sensor was measuring the water level, this sensor was placed inside of the bowl with water from which the animal was being watered. The sensor had an analog output which was processed by the analog-to-digital converter inside of the microcontroller ATMega328P used in this application.

Tomáš Bača, University of Žilina, Slovak Republic (e-mail: tomas.baca@fri.uniza.sk)
Ján Šumský, University of Žilina, Slovak Republic (e-mail: jan.sumsky@fri.uniza.sk)
Michal Kubaščík, University of Žilina, Slovak Republic (e-mail: michal.kubascik@fri.uniza.sk)
Peter Šarafín, University of Žilina, Slovak Republic (e-mail: peter.sarafin@fri.uniza.sk)

B. Action elements

In the automated terrarium developed during the bachelor theses were used five types of action elements.

The first action element was two electromagnetic valves. These valves were used in the vertical position and they were fixed to the two walls of the terrarium. Both of these valves had on their upper sides bottles with water which served as the water reservoir. Underneath these valves were placed two bowls into which the water could flow. One of the bowls was used for watering the animal and the other bowl was used for increasing the humidity in the terrarium.
The second action element was two heating resistors [2]. The first heating resistor was used to control the temperature in the terrarium during its operation it heated the air inside of the terrarium. The second heating resistor was placed under one of the bowls with the water. When this second heating resistor is turned on it will start heating the water. The heated water will start to evaporate and as a result of this evaporation of the water the humidity in the terrarium will be increased.

Both the electromagnetic valves and the heating resistors were using as a power supply 12V voltage source. All four of these elements were controlled by the output pins. The microcontroller ATMega328P had on its output pins when set to logic one voltage 5V. Because of this, the N MOSFETs were used as switches. When the output pin is set to logic one the N MOSFET will turn on the power to the electromagnetic valve or a heating resistor will be supplied. And when the output pin is set to logic zero the N MOSFET will be turned off and no power is being supplied.

The third action element was the LCD display. During the normal operation of the terrarium on this display were shown the last measured values of the temperature and humidity. When the user was setting the wanted temperature and the humidity in the terrarium these values were also shown on this LCD display. The LCD display was capable of displaying the letters of the Latin alphabet, the number and certain special symbols. The LCD display contained two lines, and each of them was capable of displaying 16 symbols, therefore the total number of symbols that could be shown at the same time was 32. This display was communicating with the microcontroller ATMega328P through the communication interface I2C.
The fourth action element was the electromagnetic buzzer. This buzzer was controlled by the PWM pin of the microcontroller ATMega328P. This buzzer was used as an audible alarm in error situations. One example of an error situation was when sensors measured higher values of temperature or humidity than the values set by the user. This situation could occur naturally by the influence of the temperature and humidity of the surrounding environment in which the automatic terrarium was placed. Or by some error in the control or operation of the heating and moisturizing elements.

The fifth action element was the RGB LED diode, on this diode were shown different colours based on the specific actions that the system was performing at that time. The RGB LED diode was controlled by the three pins of the microcontroller ATMega328P.

C. Control elements

In the automated terrarium development two different control elements were used.

The first control element was the rotary encoder. With this control element, the user could set the required value of temperature and humidity in the terrarium. This rotary encoder was sending data into the two input pins of the microcontroller ATMega328P.

The second control element was two buttons connected to two input pins of the microcontroller ATMega328P. After the user pressed the first button he went into a mode in which he could change the required values of the temperature and humidity by the rotary encoder, in this mode were also shown on the LCD display the currently set values of temperature and humidity. The second button was not used in the final version of the application, it was used just during the testing of the application.

D. Microcontroller

In the automated terrarium used the microcontroller ATMega328P [3] as a control unit. It is a low-power AVR 8-bit microcontroller with advanced RISC architecture. The microcontroller was programmed in the programming language C++.

This microcontroller has 32 eight-bit universal registers. 32 KB of programmable flash memory, 1 KB of EEPROM memory and 2 KB of internal SRAM memory.

This microcontroller was chosen because of its low power consumption and easy internal architecture that allowed it to be programmed through the registers without bigger complications.

The microcontroller was using as the clock signal its internal RC oscillator which is working at the frequency of 1 MHz.

E. Control algorithms

In the automated terrarium three different control algorithms was implemented.

The first control algorithm was responsible for sustaining certain levels of water in the bowl used for watering the animal. If the sensor in the bowl detected water level under a certain minimal threshold, the electromagnetic valve on top of the bowl would open for a short time and let a small amount of water flow into the bowl. This was repeated until the sensor detected a water level above the maximal threshold. If this process was repeated 10 times but the water level in the bowl didn't increase the system would throw out an error and start the audible alarm to get the attention of the user.

The second control algorithm was responsible for sustaining the required temperature in the terrarium. If the temperature sensor detects a lower temperature then the heating resistor will be turned on. During experiments were used two types of control algorithms for determining the intensity of the heat supplied by the heating resistor.

The third control algorithm was responsible for sustaining the required humidity in the terrarium. If the humidity sensor detected a lower humidity then the electromagnetic valve on top of the bowl would open for a short time and let a small amount of water flow into the bowl. In the next step, the heating resistor under this bowl was turned on causing the water to evaporate and as a result, increase the humidity in the terrarium.

In the end, both the temperature and humidity were controlled by the bang-bang regulation. The bang-bang regulation doesn't take into account the size of the gradient between the required and actual temperature or humidity. If the system detects a lower actual value than the predefined it just simply activates the action element in our case the heating resistors for a predefined amount of time. After this time passes the sensor will again take a measurement and if the actual value is still lower than the required the action element will get again activated for the predefined time.

The bang-bang regulation represents a very simple form of regulation, in most cases even this simple regulation can, in the end, achieve the required value of the affected parameter, however, the bang-bang regulators can take a long time before they will achieve this required value.

This regulation could be improved by the use of the P, PI or PID regulators [4] which take into account the size of the gradient between the required and actual temperature or humidity. The P, PI and PID regulators will be explained in the temperature control.

The final intensity of the heat supplied by the heating resistor (the duty cycle of the used PWM signal) is determined by the final value from the output of the P, PI or PID controller.

In the simplest control algorithm, the intensity of the heating resistor was proportional to the thermal gradient between the actual and required temperature multiplied by a constant. This is called simple proportional control and the used type of controller is called the P controller.

The time required for heating up the terrarium from room temperature to the required temperature could be improved by the use of the second type of heat control algorithm, the proportional integral control and this type of controller is called the PI controller.

The integral components sum the previously measured thermal gradients both positive and negative. The proportional component just takes the current thermal gradient.

Both the proportional and integral elements are then multiplied by different constants. These two elements after the multiplication are summed together and we get the final value from the output of the PI controller.

The final value from the output of the PI controller depends on the chosen constants with which are multiplied the proportional and integral elements.

However, if the constant for the integral element is chosen too high then the overshoot will occur. The overshoot is basically an output temperature that is higher than the required temperature.

The regulation could be further improved by the use of a PID regulator which has an additional derivation element. The final value from the output of the PID controller is calculated as the sum of the three elements multiplied by three different constants.

The derivation element is calculated as the difference between the two successive measured values. The derivation element could help improve the resistance of the controller system (temperature in the terrarium) to sudden external changes in temperature.

The problem with the PI regulation is that if a sudden change in the external temperature occurs such as when someone opens a window close to the terrarium in the middle of the winter, the temperature in the terrarium will quickly drop down. The proportional and the integral elements are not capable of quickly responding to this quick change in the temperature and it would take them some time to increase the temperature to the desired temperature.

However, the derivation element will detect this sudden change in temperature between the two successive measured values. Then it will quickly increase the final value from the output of the PI controller. This will result in a quick increase in the intensity of the temperature of the heating resistor which will counteract the lowering of temperature caused by the external factors.

However, just like with the integral element if the constant for the derivation element is chosen too high then the overshoot will occur.

The other problem connected with the P, PI and PID regulations in this application is the propagation delay caused by the fact that the heating resistor is placed on one side of the terrarium and the temperature sensor is placed on the other side of the terrarium, therefore we have to use in the calculations also the fact that even though the sensor is measuring smaller temperature than the required, the temperature on the side of the terrarium where the heating element is located could be already higher than the required temperature, and if we will don't decrease the intensity of the heating element before the measured temperature equal the required temperature we could actually create a big overshoot.

III. Results of the performed experiments

The experiments were performed with the bang-bang regulation. Throughout the experiments, different times during which the action element (heating resistor) was active were tested.

Later on through the experiments, the bang-bang regulation was adjusted by monitoring the current temperature/humidity during the time that the active element was active and the termination of the heating process ahead of the planned schedule if the required temperature or humidity was achieved before the heating cycle ended.

The last adjustment to the bang-bang regulation was the decrease in the temperature and humidity at which the regulator stopped powering the heating element. This was done since this system has in it the propagation delay.

During the experiments, the system was capable of achieving the required temperature and humidity with this adjusted bang-bang regulation algorithm.

In the graphs showing the result of the final experiment the required temperature was 26O C and the required humidity was 48%.

Through the experiment, the humidity of the air in the terrarium was artificially increased by the vaporizer which blew water steam into the terrarium and thus increased the humidity. From the results can be seen that the regulation was capable of dealing with this problem and achieved the desired humidity.

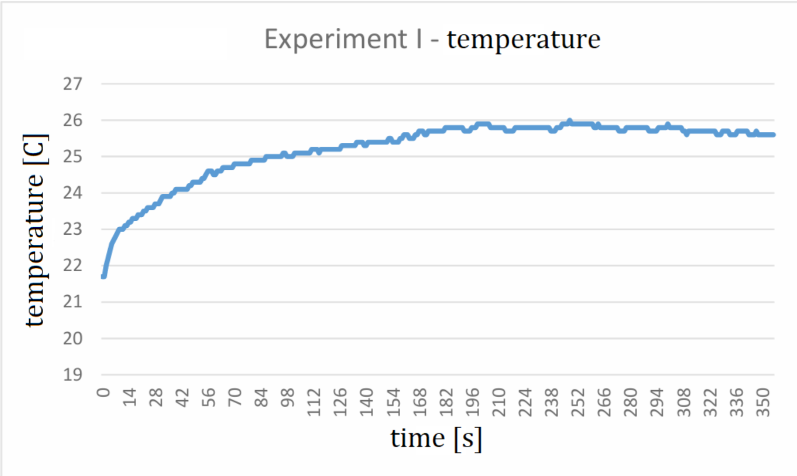

Figure 1. Experiment graph temperature

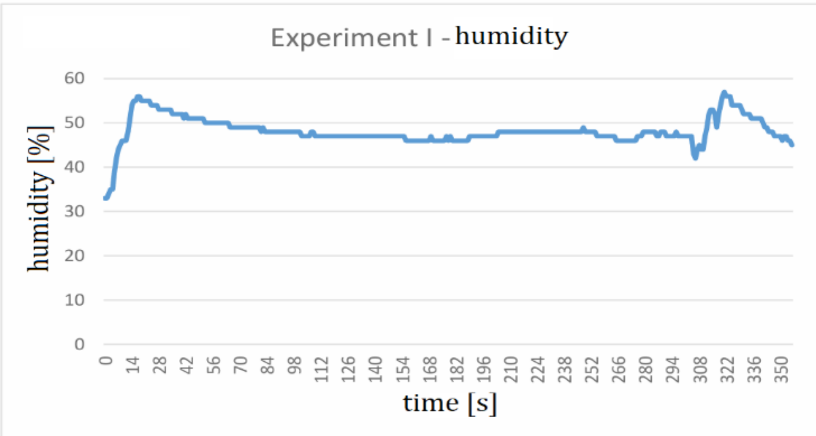

Figure 2. Experiment graph humidity

References

[1] "DHT11," *ElectronicWings*. [Online]. Available: https://www.electronicwings.com/sensors-modules/dht11. [Accessed: Dec. 6, 2024].

[2] "Heater resistor applications," *EE Power*. [Online]. Available: https://eepower.com/resistor-guide/resistor-applications/heater-resistor/#. [Accessed: Dec. 6, 2024].

[3] "ATmega328P datasheet," *Microchip Technology*, 2019. [Online]. Available: https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf. [Accessed: Dec. 6, 2024].

[4] M. R. James, "Design and implementation of proportional-integral-derivative (PID) controllers," *IEEE Transactions on Control Systems Technology*, vol. 16, no. 6, pp. 1375–1381, Nov. 2008. doi: 10.1109/TCST.2008.2000560.