

Temperature and Humidity Monitoring, Through Bluetooth HC-06

Tanalt Alina*

Department of Electronics and Computers, Transylvania University, Brasov, Romania

*Corresponding author: Tanalt Alina, Department of Electronics and Computers, Transylvania University, Brasov, Romania; E-mail: alina.tanalt@student.unitbv.ro

Received date: 15 January, 2023, Manuscript No. JEEET-23-87161;

Editor assigned date: 18 January, 2023, PreQC No. JEEET-23-87161 (PQ);

Reviewed date: 01 February, 2023, QC No. JEEET-23-87161;

Revised date: 18 April, 2023, Manuscript No. JEEET-23-87161 (R);

Published date: 25 April, 2023, DOI: 10.4172/2325-9833.1000954

ABSTRACT

The document consists in the implementation of a system of temperature and humidity monitoring, what's the matter has a sensor of temperature DHT11 controlled with an Arduino UNO R3 AT-board mega328p. Transmission of results regarding temperature and humidity is done with the help of a bluetooth module HC-06 via through an application, installed on a smartphone, having android, but and controlling the lighting of some LEDs. For task synchronization and real time data collection, s.a used FreeRTOS and the semphr library (add to work, FreeRTOS files for traffic lights).

Keywords: Bluetooth module HC-06; Temperature sensor; Microcontroller; Real-Time Operating System (RTOS); LED

Introduction

The thermostat has been widely applied to medicine, textiles, food processing and other fields. However, since there is a relatively large time gap in terms of temperature measurement, together with a large and time varying time constant for temperature control in the thermostat, the precise regulation of the temperature in the thermostat is a difficult problem [1]. Temperature is a common and important physical parameter in the manufacturing process, *i.e.*, and in scientific experiments, we should effectively control the main parameters, such as temperature, pressure, flow, speed and so on, for an efficient, high production during the industrial manufacturing process. Temperature management is essential in the manufacturing process [2]. Precise temperature measurement and control are essential prerequisites for ensuring a high quality production, a high yield, low costs and a safe product. The adoption of microelectronic technology is a significant technique to ensure the stable operation of the manufacturing process and to increase the precision the control. Because it can improve working conditions, save energy, avoid production and equipment accidents and offer economic and technological benefits [3]. As we know, some products can work, get, hold a better one quality assurance, and can have a better benefit only in the condition of constant temperature [4]. Thus, the appearance of the thermostat made these industries have a certain spatial development, and temperature control is the problem that must be solved. For this reason, projects the thermostat system, having in component Arduino Uno board, a bluetooth HC module, a sensor of temperature DHT11, two

LEDs, 2 resistors and a breadboard. The work presents the design and implementation of a system monitoring system controlled *via* bluetooth. The main contributions of this work are;

- Designing electrical circuits.
- Arduino programming.
- FreeRTOS implementation.
- Synchronization of tasks.
- The connection between the project and android bluetooth.

Current state

In the article the authors implemented a thermostat system, so that it adapts to the production requirement. Currently, the thermostat control methods are PID control and fuzzy control. The performance of PID control primarily depends on its parameters, and the same set of parameters is used to control the dynamic process from heating to heat preservation, the control effect is difficult to achieve. And the steady state error of the fuzzy control exists, it is difficult to eliminate [5]. A good control effect can be obtained if both are combined and optimized. The authors reached the following conclusions after performing several scenarios.

- When using the fuzzy PID control plan to regulate the constant temperature box, not only can it become fast in the heating stages, but the fuzzy control also has the advantage of small overshoots. It can benefit from PID control in the constant temperature stages, as well as from the elimination of steady-state errors.
- Using the fuzzy+PID control model in the system thermostat temperature control will have a value of reference, specified to increase the precision of temperature control and the quality of the product.

The authors of the article "autonomous sensor system with power harvesting for telemetric temperature measurements of pipes" have developed an autonomous sensor system with an electronic system with low energy consumption for radio frequency communication, which incorporates a thermoelectric energy capture module for unattended operation. It consists of an app, first for measuring the temperature aturis, ages with wall, *i.e.*, when the autonomous sensor is placed on the heat source, a thermoelectric module collects energy, feeds the autonomous sensor. In this scenario, an external power source is not required, the temperature is measured, and the data is stored in a non-volatile memory. The electromagnetic field is used to power the autonomous sensor system and to transfer data when the external reading unit is connected. An experimental setup was established and characterized by measuring the temperature along the pipe, the voltage that the thermoelectric generators can create and the effect of different materials on the RF transmission. The temperature data collected by the autonomous sensor from the heat source are compared with those of a reference thermistor [6].

In this article, the authors developed an active, ionized device voice that controls the set temperature of a thermostat residential [7]. The designed device is voice activated, which allows to avoid disability, a voice activated chip is used in the design and is mounted on a standard room thermostat. The device is dependent on the speaker. Available safety features are also incorporated, allowing it to be deactivated and return to normal manual control. The user speaks the desired temperature in a voice sensor Built-in (microphone), the signal is processed, recognized and sent to the thermostat, which, in turn,

adjusts according to the instructions given. The device takes 15 sentences, spoken with a maximum duration of 3.2 seconds. As a result, the device's voice activated temperature range will be between 65°F and 79°F. To activate the device, the user will say "temperature 75", which the device will recognize and use to adjust the thermostat. In order to avoid accidentally turning on the gadget while the user is engaged in a conversation, usually at night, the term "temperature" must be included in each instruction. In addition, the microphone detects the user's voice only when it is close enough to it. The displayed device shows the current temperature of the room, as well as I requested temperature [8].

Materials and Methods

The designed system

Detailed analysis of real time performances

Adding FreeRTOS improved the response time of the sensor, but not only. The following changes have occurred:

- Simplification of tasks, better organization of the code.
- Synchronization of tasks through semaphore.
- Faster response time. Following the addition of FreeRTOS, the data collection time has improved, it. The response time has improved, from 1.5 seconds to 0.5 seconds.
- Adding the task priority term, in our case temperature has priority 1, and humidity has priority 2.
- System management.

The implemented system can be part of the category of real-time hard systems, but also of the category of real-time soft systems, depending on how it was used and what exactly it was used for.

Components used

The UNO R3 ATmega328p development board has evolved from simple 8 bit boards to applications with complex products, I/O, 3D printing, wearables, and integrated environments. All Arduino boards are open source so that users can work individually and thus build their applications freely and ultimately adapt to each idea [9]. The software is also open source, in full development, and a main factor that determined this growth is represented by the users of this board who, following the ingenious developments, contributed massively increasing the quality of this platform. The microcontroller represents a microcircuit that encapsulates a central unit and a memory together with resources that allow it to interact with the external environment. A microcontroller has a processor [10]. The processor is a component importance which has the role of to execute certain instructions, such as:

Technical characteristics

- **Microcontroller type:** ATmega328p.
- **USB:** CH340G.
- **Operating voltage:** 5V.
- **Input voltage:** 7 V-9 V.
- **Output voltage:** 6 V-12V.
- **Input/output pins, digital inputs:** 14.
- **Analog input pins:** 6.

The DHT11b temperature sensor is an easy to use temperature and humidity sensor with good reliability and stability. The temperature is measured with a surface mounted NTC (thermistor), integrated into the component [11].

Technical specifications

- **Operating voltage:** 3.5V to 5.5V.
- **Operating current, ionization:** 0.3 mA (measuring) 60uA (standby).
- **Ies, ire:** Serial data.
- **Temperature:** 0°C to 50°C.
- **Humidity:** 20.
- **Accuracy:** $\pm 1^\circ\text{C}$ and ± 1 .

Bluetooth module HC-06, a bluetooth connection module was used for the remote transmission of the measurements made by the sensor. Communication is carried out *via* serial RX(10) and wireless TX(9) at 2.4 GHz.

Technical specifications

- **Bluetooth protocol:** Bluetooth V2.0 type.
- **Power level:** Class 2.
- **Frequency band:** 2.40 GHz-2.48 GHz.
- **Receiver sensitivity:** -85dBm.
- **USB:** USB v1.1/2.0.
- **Modulation mode:** Frequent, gauss shift keying.
- **Security function:** Authentication and encryption.
- **Operating voltage range:** +3.3 V to +6 V.
- **Operating temperature:** -20°C to +55°C.
- **Operating current, ionization:** 40 mA.

The last components are 2 LEDs and 2 resistors, its 1 k Ω , the LEDs serve to check the process of switching on/off an LED with the help of an application, pre-installed on the mobile phone.

Functional description

The DHT11 is a basic, very cheap digital temperature and humidity sensor. It is able to detect the temperature and also the relative humidity, which is the amount of water vapor in the air compared to the saturation point of the water vapor in the air. The DHT11 humidity and temperature sensor consists of 3 main components. A resistive humidity sensor, an NTC (negative temperature coefficient) thermistor (for temperature measurement) and an 8 bit microcontroller, which converts the analog signals from both sensors and sends a single digital signal.

The DHT11 controlled by the development board measures the ambient temperature, and with the help of the connection to the application *via* the bluetooth module, upon entering certain keywords established in the code, the established functions will be performed (Figure 1).

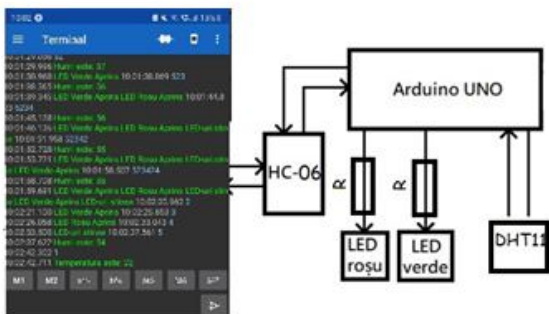


Figure 1: Application project connection.

Explained

In order to understand, as best as possible, the way the project works, several functions and syntaxes are exemplified and explained;

Libraries used

- Arduino-FreRTOS.
- Software serial (makes the connection with the application).
- dht11 (to use the DHT11 temperature sensor).
- Semphr (add FreeRTOS functions for semaphores). Syntaxes used from the "semphr" library.

Syntaxes used from the "semphr" library

- XSemaphoreCreateBinary each required binary semaphore a small amount of RAM that is being used to save the state of the traffic light. If a binary semaphore is created using xSemaphoreCreateBinary, then the required RAM is automatically allocated from the FreeRTOS heap.
- XSemaphoreGiv (xBinarySemaphore) macro for a r lease a traffic light. It is obtained, holds pdTRUE if the semaphore He was released. PdFALSE if an error occurred.
- XSemaphoreTake (xBinarySemaphore 5) macro to get, holds a semaphore. 5 represents time, expressed in ticks, in which it sits, waiting for the traffic light to become available. It get, hold pdTRUE if obtained, hold the semaphore. PdFALSE if xTicksToWait timed out without the semaphore being available.

The semaphore is created in state 0, which means that the semaphore must first be given, calling the function, xSemaphore-give, before it can be retrieved (obtinut) later using the function, xSemaphoreTake.

The critical section is executed only when xSemaphore-take returned pdTRUE, otherwise everything inside the semaphore is not executed (if (xSemaphore-Take (xSerialSemaphore, (ticktype t) 5) =pdTRUE)).

Wiring diagram

The components used in the realization of the project are presented in Figure 2, this block diagram was created through an online open source circuit maker platform.

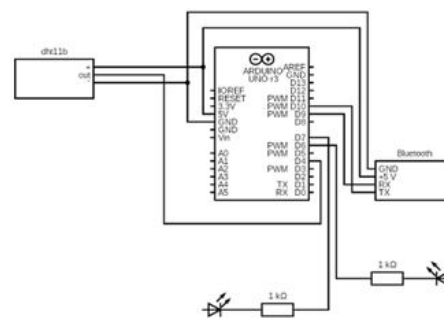


Figure 2: Wiring diagram.

For a better one presentation of the electric circuit, a more graphic model was made, a user friendly model, with an ability to aim, better understanding. This can be seen in Figure 3.

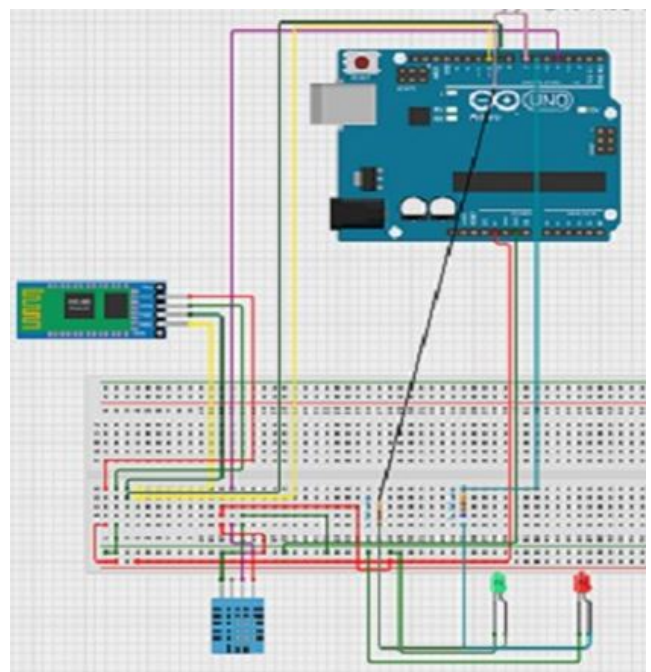


Figure 3: Circuit electric, fritzing.

For a better friendly visualization of the electrical scheme trice, this one scheme the electricity was done with help to fritzing. After as can be seen, in this they have a figure all the components listed in the "components" subsection have been designed.

The components used together with the price of each can be seen in Table 1.

No.	Component	Quantity price	Price/unit	Price
1	Arduino Uno	1	\$30	\$30
2	1kΩ rezisten	2	\$0.25	\$0.5
3	Modul BT HC-06	1	\$5	\$5
4	Senzor DHT11	1	\$3	\$3
5	LED	2	\$0.05	\$0.10
6	Breadboard	1	\$5	\$5
7	Fire Jumper	1	\$1	\$1
	Total cost			\$44.6

Table 1: List of materials for the project components.

Results and Discussion

In Figure 4 shows the values transmitted by the sensor DHT11, in a graphic form. The blue line represents the data collected by DHT, for humidity. The pink line represents the data collected by DHT, for temperature. It can be seen from the graph that the temperature collection task had a faster running time, due to the set priority level (in the case of temperature, the set priority was "1"), unlike the task where the humidity data is collected, where the established priority is "2".



Figure 4: Serial plotter Graphs.

In Figure 5 shows the real values transmitted by the sensor, in text form. In this Figure they were implemented display, temperature area, humidity, but also of a preset command in the code (Table 2).

```

Timpul executarii task-ului TaskCitireHumi este de: 177.33ms
Timpul executarii task-ului TaskCitireTemperatura este de: 173.93ms
Waiting for command...
Temperatura:21
Humi:58
Timpul executarii task-ului TaskCitireHumi este de: 177.29ms
Timpul executarii task-ului TaskCitireTemperatura este de: 177.29ms
Waiting for command...
Temperatura:21
Humi:57
Timpul executarii task-ului TaskCitireHumi este de: 177.24ms
Timpul executarii task-ului TaskCitireTemperatura este de: 177.25ms
Waiting for command...
Temperatura:21
Humi:57
Timpul executarii task-ului TaskCitireHumi este de: 177.21ms
Timpul executarii task-ului TaskCitireTemperatura este de: 177.22ms
Waiting for command...
Temperatura:21
Humi:58
Timpul executarii task-ului TaskCitireHumi este de: 177.24ms
Timpul executarii task-ului TaskCitireTemperatura este de: 177.24ms
Waiting for command...
Temperatura:21
Humi:58
Timpul executarii task-ului TaskCitireHumi este de: 177.22ms
Timpul executarii task-ului TaskCitireTemperatura este de: 180.64ms
Waiting for command...
Temperatura:21
    
```

Figure 5: Serial monitor tasks.

Simulation number	Earth/temp no FreeRTOS	Humi/temp cu FreeRTOS
1	5.31/5.11 sec	177.29/177.29 ms
2	5.21/4.99 sec	177.24/177.25 ms
3	5.12/5.01 sec	177.24/177.24 ms
4	5.17/4.89 sec	177.21/177.22 ms
5	4.66/4.02 sec	177.22/180.64 ms
6	4.98/5.01 sec	177.23/180.51 ms
7	5.11/4.95 sec	177.29/177.88 ms
Media total	5.08/4.85 sec	177.24/178.29/ ms

Table 2: Comparison of execution times, i.e., with FreeRTOS/fara FreeRTOS.

Conclusion

In conclusion, this document presents improved execution time due to the grouping of information in the form of tasks, but also their synchronization with the help of traffic lights.

This paper presents the creation of a simple thermostat for measuring temperature and humidity, as well as the lighting of some LEDs with preset commands. Therefore, the experimental verification of the functioning of the project had several stages and adaptations, which successfully led to the fulfillment of the proposed objectives. The system is good for both humidity and air detection as well as temperature detection. It is a cheap device, easy to use and with great applicability. As a future activity, a battery can be added to power the Arduino board, a system for sending the data collected by the sensor to the phone, a sound warning system for temperatures too high/low, or humidity too high/low a poster, LCD for the poster, temperature area turii/humidity.

References

1. Barroca N, Borges LM, Velez FJ, Monteiro F, Gorski M, et al. (2013) Wireless sensor networks for temperature and humidity monitoring within concrete structures. *Constr Build Mater* 40:1156-1166.
2. Ribeiro RR, Bauer E, Lameiras R (2021) Hicroterm: An open source and low cost temperature and humidity monitoring system for laboratory applications. *Inventions* 6:84.
3. Xia Y, Hao H, Zanardo G, Deeks A (2006) Long term vibration monitoring of an RC slab: Temperature and humidity effect. *Eng Struct* 28:441-452.
4. Massaroni C, Caponero MA, D'Amato R, Lo Presti D, Schena E (2017) Fiber Bragg grating measuring system for simultaneous monitoring of temperature and humidity in mechanical ventilation. *Sensors* 17:749.
5. Wei P, Ning Z, Ye S, Sun L, Yang F, et al. (2018) Impact analysis of temperature and humidity conditions on electrochemical sensor response in ambient air quality monitoring. *Sensors* 18:59.
6. Zonno G, Aguilar R, Boroschek R, Lourenço PB (2019) Analysis of the long and short term effects of temperature and humidity on the structural properties of adobe buildings using continuous monitoring. *Eng Struct* 196:109299.
7. Huerta R, Mosqueiro T, Fonollosa J, Rulkov NF, Lujan IR (2016) Online decorrelation of humidity and temperature in chemical sensors for continuous monitoring. *Chemometr Intell Lab Syst* 157:169-176.
8. Gawade DR, Ziemann S, Kumar S, Iacopino D, Belcastro M, et al. (2021) A smart archive box for museum artifact monitoring using battery less temperature and humidity sensing. *Sensors* 21:4903.
9. Danao MG, Zandonadi RS, Gates RS (2015) Development of a grain monitoring probe to measure temperature, relative humidity, carbon dioxide levels and logistical information during handling and transportation of soybeans. *Comput Electron Agric* 119:74-82.
10. Polsky LB, Madureira AM, Filho ELD, Soriano S, Sica AF, et al. (2017) Association between ambient temperature and humidity, vaginal temperature, and automatic activity monitoring on induced estrus in lactating cows. *J Dairy Sci* 100:8590-8601.
11. Lee CY, Fan WY, Chang CP (2011) A novel method for in-situ monitoring of local voltage, temperature and humidity distributions in fuel cells using flexible multi-functional micro sensors. *Sensors* 11:1418-1432.