International Journal of Industrial Engineering and Management (IJIEM), Vol. 9 No 1, 2018, pp. 53-58
Available online at www.iim.ftn.uns.ac.rs/ijiem_journal.php
ISSN 2217-2661

UDC 621.38:007.5

A Conceptual Solution of Low-Cost Temperature Data Logger With Relatively High Accuracy

Milovan Medojevic
Teaching Assistant and Researcher, Faculty of Technical Sciences, University of Novi Sad, Novi Sad, Serbia, medojevicm@uns.ac.rs
Milana Medojevic
Researcher, Faculty of Technical Sciences, University of Novi Sad, Novi Sad, Serbia, milanaperic@gmail.com
Nikola Radakovic
Full Professor, Faculty of Technical Sciences, University of Novi Sad, Novi Sad, Serbia, nikolar@uns.ac.rs
Milovan Lazarevic
Associate Professor, Faculty of Technical Sciences, University of Novi Sad, Novi Sad, Serbia, laza@uns.ac.rs
Nemanja Sremcev
Assistant Professor, Faculty of Technical Sciences, University of Novi Sad, Novi Sad, Serbia, nextesla@uns.ac.rs

Received (11.12.2017.); Revised (09.01.2018.); Accepted (30.01.2018.)

Abstract

In this paper, an effort has been made to design and develop low-cost temperature data logger with relatively high accuracy. Initially, the conceptual solution of data logger consists of several elements where the main components are Arduino Uno R3 with ATmega328 microcontroller, 1-wire digital temperature sensor DS18B20 and pull-up resistors, connected with jumper wires through Breadboard according to the program algorithm. In addition, relevant component specifications are concisely provided upon which schematic connection diagram was generated. Lastly, the aim of this study was to build a platform for sensor with data logging ability in order to achieve excellent performance at substantially lower costs then currently available ones on the market.

Key words: *Data Logger, temperature, Arduino, DS18B20 probe.*

1. INTRODUCTION

In modern world, practically there are no systems in which some kind of temperature monitoring is not needed, while the measurement accuracy offered by today's most advanced temperature data loggers rivals the performance of many higher priced, computer-based data acquisition systems. Moreover, the need for collecting high quality data exponentially increases [1], as better information on performance can improve understanding dynamics of system energy use [2], thermal comfort [3], indoor environmental quality (IEQ) [4,5], microbiology of the built environment [6], etc. Although advanced management systems can collect large amounts of data on system operation [7], accurate characterizations of many parameters are often limited to the use of proprietary hardware/software, which adversely affects costs, flexibility and data integration in decision making and control [8]. As a result, many investigations are unable to collect widespread performance, which may negatively impact system effectiveness. Temperature data loggers are easy to deploy, and can be used as stand-alone devices. However, not all temperature data loggers are created equal, and with so many choices available today, it can be challenging to know which one is right for certain application.

2. THE PLATFORM

In the following section, relevant components of the data logger platform were described. Respectively, relevant information regarding Arduino hardware and software were given in the section 2.1, while section 2.2 provides important temperature probe specifications.

2.1 Specification of Arduino Uno microcontroller

The Arduino Uno is an electronic board based on the ATmega328 microcontroller. It has 14 digital input/output pins of which 6 can be used as PWM outputs, 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. In one word, it contains everything needed to support the microcontroller; simply connected to a computer via USB cable or powered with a AC-to-DC adapter or battery in order to get started. Main characteristics of Arduino Uno R3 microcontroller are given in the table 1.

Table 1. Characteristics of Arduino Uno R3 microcontroller

Microcontroller	ATmega328
Operating Voltage	5V
Input Voltage	7-9V
Input Voltage (limits)	6-20V
Digital I/O Pins	14 (6 provide PWM output)
Analog Input Pins	6
DC Current per I/O Pin	40 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328) (0.5 KB used by bootloader)
SRAM	2 KB (ATmega328)
EEPROM	1 KB (ATmega328)
Clock Speed	16 MHz

More importantly, a large number of high-grade sensors and devices have custom Arduino libraries while the equipment manufacturers provide active support constantly. Subsequently, the Arduino platform has been used successfully in several other similar data collection research studies such as wireless sensor networks for temperature and humidity monitoring [12], monitoring human activity and integration via Wi-Fi Networks [13], and balancing envelope and heating system parameters for retrofit analyses using building sensor data [14]. The validation of these studies and many other projects demonstrated the reliability of Arduino platform for data collection, making it a viable choice for developing sensor based data logger. In the figure 1 physical appearance of Arduino Uno microcontroller is shown.

Figure 1. Arduino Uno physical appearance

Each of previously mentioned 14 digital pins on the Uno can be used as an input or output, using pinMode(), digitalWrite() and digitalRead() functions. They all operate at 5V. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor of 20-50 kΩ which is disconnected by default. In addition, some pins have specialized functions such as:

- Serial: 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 USB-to-TTL Serial chip.
- External Interrupts: 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value.
- PWM: 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the analogWrite() function.
- SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication using the SPI library.
- LED: 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

Also, Arduino Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values). By default these inputs can measure from ground to 5 volts, though it is possible to change the upper end of their range using the AREF pin and the analogReference() function. Similar to digital inputs, some pins have specialized functionality:

- TWI: A4 or SDA pin and A5 or SCL pin. Support TWI communication using the Wire library.

There are a couple of other pins on the board:

- AREF. Reference voltage for the analog inputs. Used with analogReference().
- Reset. Brings LOW line to reset the microcontroller, typically used to add a reset button to shields which block the one on the board [15].

Subsequently, code uploaded to an Arduino (called a "sketch") stays in memory until it is replaced with something else, even when the power is off. Thus, it is possible to develop and upload code when Arduino board is connected to a computer and then run that code with some other power supply when it is no longer connected to a computer. This is an essential feature for outdoor systems like weather stations [16].

2.2 **Specification of DS18B20 temperature probe**

The DS18B20 digital thermometer provides 9-bit to 12-bit Celsius temperature measurements and has an alarm function with non-volatile user-programmable upper and lower trigger points. The DS18B20 communicates over a 1-Wire bus that by definition requires only one data line (and ground) for communication with a central microprocessor. In addition, the DS18B20 can derive power directly from the data line ("*parasite power*"), eliminating the need for an external power supply. Each DS18B20 has a unique 64-bit serial code, which allows multiple DS18B20s to function on the same 1-Wire bus. Thus, it is simple to use one microprocessor to control many DS18B20s distributed over a large area. Figure 2 provides closer insight regarding pin configurations.

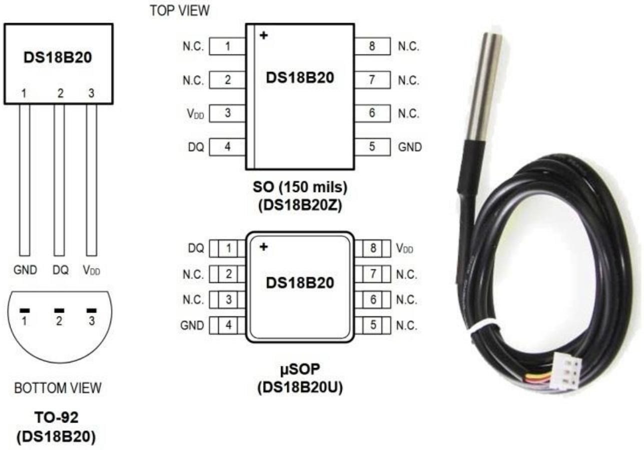

Figure 2. DS18B20 pin configurations and appearance

The core functionality of the DS18B20 is its direct-to-digital temperature sensor. The resolution of the temperature sensor is user-configurable to 9, 10, 11, or 12 bits, corresponding to increments of 0.5°C, 0.25°C, 0.125°C, and 0.0625°C, respectively. The default resolution at power-up is 12-bit. The DS18B20 powers up in a low-power idle state. To initiate a temperature measurement and A-to-D conversion, the master must issue a Convert T [44h] command. Following the conversion, the resulting thermal data is stored in the 2-byte temperature register in the scratchpad memory and the DS18B20 returns to its idle state. If the DS18B20 is powered by an external supply, the master can issue "read time slots" (see the 1-Wire Bus System section) after the Convert T command and the DS18B20 will respond by transmitting 0 while the temperature conversion is in progress and 1 when the conversion is done. If the DS18B20 is powered with parasite power, this notification technique cannot be used since the bus must be pulled high by a strong pull-up during the entire temperature conversion [17]. On the other hand, featuring strong anti-interference ability and high accuracy, DS18B20 is covered with waterproof rubber hose outside and able to measure temperature in the range -55°C ~+125°C. Typical performance curve and temperature reading error diagram for DS18B20 sensor is given in figure 3.

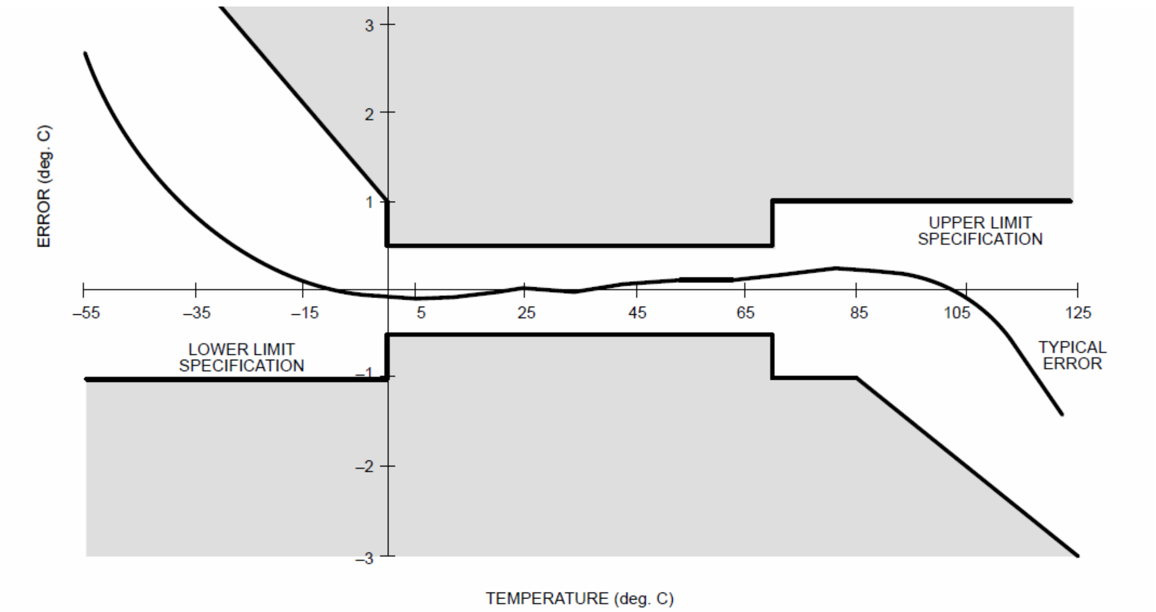

Figure 3. Typical performance curve and temperature reading error diagram for DS18B20 sensor [17]

3. CONCEPTUAL SOLUTION

The conceptual solution of Arduino-based data logger consists of several elements as already stated, where the main components are Arduino Uno, DS18B20 temperature sensor in the form of probe, pull-up resistors and breadboard. The connection of the sensor to Arduino Uno was performed through a breadboard which is used to test the device operation in order to determine possible shortcomings (Figure 4).

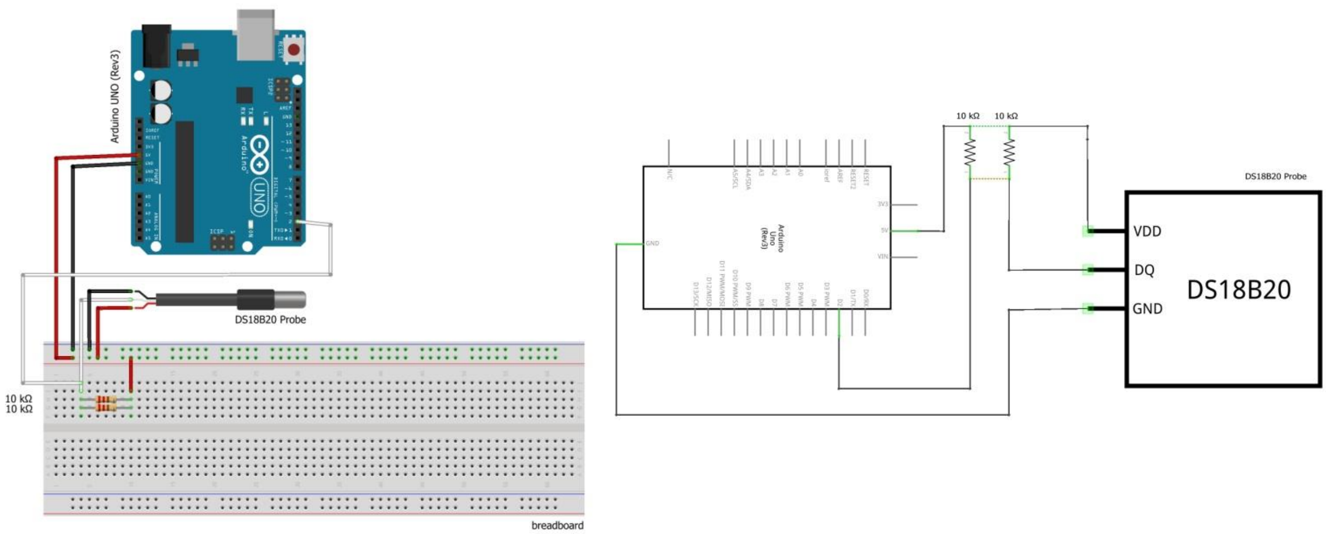

Figure 4. The connection of sensor to Arduino Uno via bradboard (left) and its electrical scheme (right)

Subsequently, an adequate program algorithm was determined and transferred to Arduino Uno microcontroller in the form of code. Key elements of program algorithm are given in the figure 5.

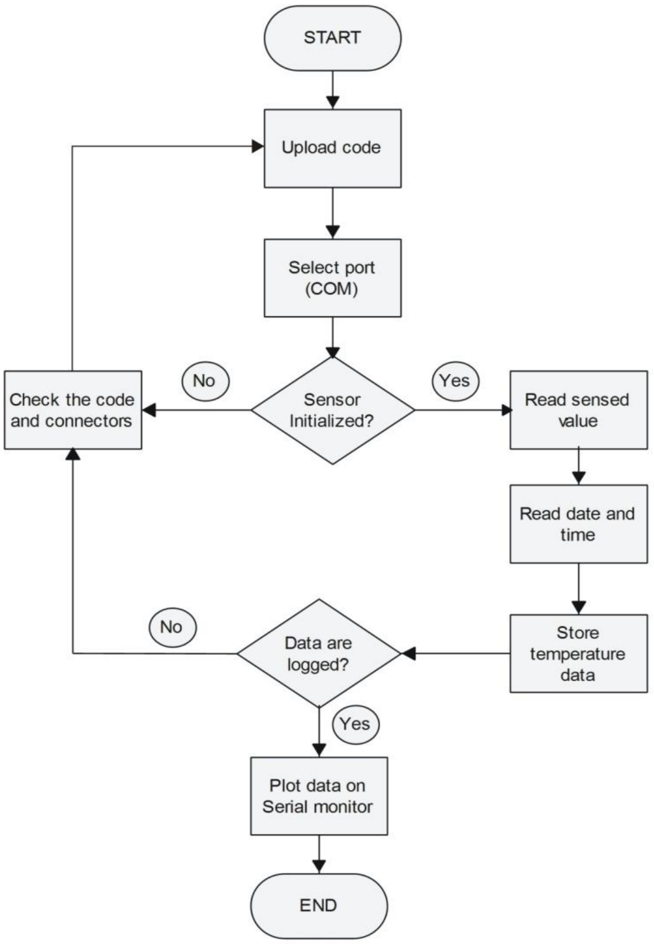

Figure 5. Simplified Program Algorithm

The first step is the code uploading after which it is necessary to determine and select the communication (COM) port program. After that, program performs the initialization of sensor. Here the pins on which the sensors are located must be defined in order microcontroller can recognize them. Upon initialization the recoded values are sent and read. In this version, 6 readouts from the sensor are recorded for a period of one minute, while the data are plotted on the Serial monitor.

4. DEVICE TEST

Device testing was carried out in controlled climatic conditions where a reference temperature of a room was 22°C. The experiment setting assumes the pouring of boiling water into a 2 dl glass in which the sensor is located. The temperature reading is done in an interval of 10 seconds. In this fairly simple setting, water was naturally cooled down to the approximate temperature of the environment in which the glass is located. The decreasing temperatures were plotted on serial monitor, upon which the generated data were processed and the graph given in the figure 6 was build.

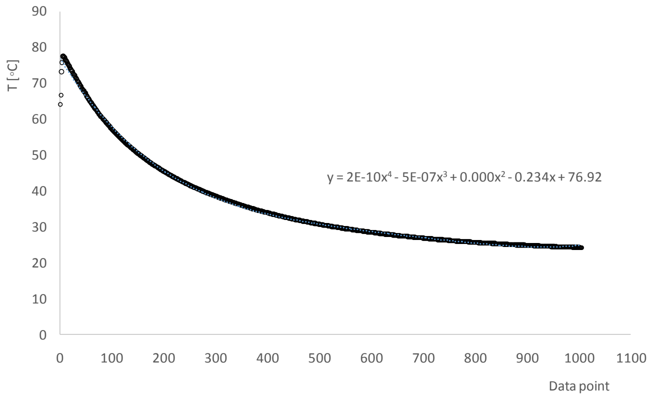

Figure 6. Simple test results

The decrease of the water temperature in the glass can be described by the fourth-order polynomial according to the expression:

$$y = 2E\text{-}10x^4 - 5E\text{-}07x^3 + 0.001x^2 - 0.234x + 76.92 \quad (1)$$

4.1 Cost Structure

Having in mind that some temperature data loggers can reach prices up to several thousands of euros, depending on the accuracy levels they can provide, it is important to point out the cost structure of proposed, Arduino based conceptual model for temperature data logging. Table 2 specifies the exact cost structure of aforementioned logger, while the prices were shown in RSD and €.

Table 2. The cost structure of shown temperature data logger

Nº	Hardware item	Quantity [pcs]	Cost [RSD]	Cost [€]
1	Arduino Uno R3	1	1000	8.47
2	DS180B20 probe	1	300	2.55
3	10 kΩ Pull-up resistor	2	2	0.03
4	Jumper wires	4	35	0.30
5	Breadbord	1	500	4.25
∑			**1837**	**15.6**

Bering in mind the performance curve of DS18B20 sensor given in the figure 3, temperature measurement range and the cost of the device envisaged by this conceptual solution it is more than obvious that for a low cost it is possible to log temperature at a very high level of accuracy.

4.2 Possible applications

In many industries it's essential to know to what temperature product has been exposed and for how long. Whether sterilizing surgical instruments or medical devices in an autoclave or heating food before canning, there should be evidence of temperature and time. In facility management, especially where precision measurement or temperature-sensitive processing is carried out, a record of temperature and humidity variation can be useful in identifying inefficiencies. A particular concern for a facilities manager is the protection against legionella which can cause a potentially fatal form of pneumonia. In applications like these a combination of temperature and humidity data loggers provides a time-stamped record of the experienced conditions over an extended period. Another situation where a time-stamped record is useful is the transportation of artworks. Here just the knowledge that conditions are being monitoring may provide an added incentive for careful treatment. In addition, temperature measurement is important in applications ranging from monitoring the health of rivers and streams to verifying that sterilization procedures have been performed correctly. In some cases, measurements are needed over extended periods as a way of determining long term trends. In others the concern is to know the maximum or minimum temperature attained and the duration of exposure [18]. Moreover, many foods and some pharmaceuticals must be shipped under carefully controlled conditions to avoid spoilage. Including a compact temperature data logger placed in with the transported materials to provide a record of the conditions experienced. This helps to ensure the integrity of the products and provides evidence should any claim of mishandling arise. Last but not the least, various types of aquariums benefit from monitoring water temperature in order to ensure a continued healthy environment for their fish. In similar vein, scientists use the temperature of rivers and streams as an indicator of the health of those ecosystems. Although both are good applications for a water temperature data logger, they place differing demands on the equipment. In an aquarium, the logger will be readily accessible so there is fewer requirements for extensive memory capacity and the data may be recovered either through direct PC connection or by Bluetooth. Conversely, gathering useful data about river temperatures may mean leaving the logger in-place for many months. Under such circumstances memory capacity, battery life and possibly wireless capabilities all take on a greater importance. These and many similar processes can be monitored, controlled and verified by using temperature data loggers.

5. CONCLUSION

In this article a conceptual solution and realization of a Arduino-based Temperature data logger was successfully carried out. The indoor testing results showed that the developed device satisfactorily performed the required measurements and data logging, while all components worked properly. This low-cost portable data logger meets the accuracy requirements of majority of monitoring systems. Subsequently, the system design features, easy-to-obtain hardware and open access software, makes it accessible to any researcher or user for the development of systems of their own design and use. Moreover, this flexibility makes the system more suitable for each intended application, such as the monitoring of manufacturing plants, households and buildings, as well as the collection of data at remote locations. The cost of the proposed system is considerably lower than commercially available devices, with negligible loss of accuracy and precision. Last but not the least, the data logger developed in this study achieves a cost of approximately 16 €. Lastly, the device can be improved by introducing additional sensors, for example sensors for humidity, CO, CH4, wind speed and direction, etc., which could stimulate a wider range of application. However, subsequent improvements are subject of further research.

6. REFERENCES

[1] Guerra-Santin, O. and Tweed, C.A., (2015), "*In-use monitoring of buildings: An overview of data collection methods*", Energy and Buildings, Vol. 93, April, pp.189-207.

[2] Babaei, T., Abdi, H., Lim, C.P. and Nahavandi, S., (2015), "*A study and a directory of energy consumption data sets of buildings*", Energy and Buildings, Vol. 94, May, pp. 91-99.

[3] Langevin, J., Wen, J. and Gurian, P.L., (2015), "*Simulating the human-building interaction: Development and validation of an agent-based model of office occupant behaviors*", Building and Environment, Vol. 88, Jun, pp. 27-45.

[4] Ramos, T., Dedesko, S., Siegel, J.A., Gilbert, J.A. and Stephens, B., (2015), "*Spatial and Temporal Variations in Indoor Environmental Conditions*", Human Occupancy, and Operational Characteristics in a New Hospital Building, PLoS ONE 10 (3) (Mar.2015) p. e0118207.

[5] Heinzerling, D., Schiavon, S., Webster, T. and Arens, E., (2013), "*Indoor environmental quality assessment models: A literature review and a proposed weighting and classification scheme*", Building and Environment, Vol. 70, December, pp. 210-222.

[6] Ramos, T. and Stephens, B., (2014), "*Tools to improve built environment data collection for indoor microbial ecology investigations*", Building and Environment, Vol. 81, November, pp. 243-257.

[7] Xiao, F. and Fan, C., (2014), "*Data mining in building automation system for improving building operational performance*", Energy and Buildings, Vol. 75, Jun, pp. 109-118.

[8] Tse, W.L. and Chan, W.L., (2008), "*A distributed sensor network for measurement of human thermal comfort feelings*", Sensors and Actuators, Vol. 144 (2), Jun, pp. 394-402.

[9] Jang, W.S., Healy, W.M. and Skibniewski, M.J., (2008), "*Wireless sensor networks as part of a web-based building environmental monitoring system*", Automation in Construction, Vol. 17(6), August, pp. 729-736.

[10] International Organization for Standardization, BS EN ISO 7726:2001, "*Ergonomics of the Thermal Environment. Instruments for Measuring Physical Quantities*", 2001st ed., International Organization for Standardization, 2001.

[11] Ali, A.S., *Low-cost Arduinobased Platform for Long-term Data Collection in Indoor Environments*", Master's thesis, Illinois Institute of Technology, Chicago, IL.

[12] Barroca, N., Borges, L.M., Velez, F.J., Monteiro, F., Gorski, M. and Castro-Gomes, J. (2013) "*Wireless sensor networks for temperature and humidity monitoring within concrete structures*", Construction and Building Materials, Vol. 40, March, pp. 1156-1166.

[13] Lian, K.Y., Hsiao, S.J.and Sung, W.T., (2013) "*Intelligent multi-sensor control system based on innovative technology integration via ZigBee and Wi-Fi networks*", Journal of Network and Computer Applications, Vol. 36 (2), March, pp. 756-767.

[14] Nagy, Z., Rossi, D., Hersberger, C., Irigoyen, S.D., Miller, C. and Schlueter, A., (2014) "*Balancing envelope and heating system parameters for zero emissions retrofit using building sensor data*", Applied Energy, Vol. 13, October, pp. 56-66.

[15] Arduino technical sheet. available at: https://www.farnell.com/datasheets/1682209.pdf

[16] David R. Brooks, D.R. (2017) "*An Arduino-Based Weather Station*", Institute for Earth Science Research and Education, available at: http://www.instesre.org/ArduinoWeatherStation.pdf

[17] DS18B20 - Programmable Resolution 1-Wire Digital Thermometer data sheet. Available at: https://datasheets.maximintegrated.com/en/ds/DS18B20.pdf

[18] OMEGA, Introduction to Temperature Data Logging, White Paper, Available at: https://www.omega.com/technical-learning/pdfs/IntroTempDataLogging_WhitePaper.pdf

Konceptualno rešenje ekonomičnog sistema relativno visoke preciznosti za akviziciju podataka o temperturi

Milovan Medojevic, Milana Medojevic, Nikola Radakovic, Milovan Lazarevic, Nemanja Sremcev

Primljen (11.12.2017.); Recenziran (09.01.2018.); Prihvaćen (30.01.2018.)

Apstrakt

Cilj i svrha ovog rada jeste dizajn i razvoj niskobudžetnog sistema za akviziciju podataka o temperaturi relativno visoke preciznosti. Inicijalno, konceptualno rešenje sistema za akviziciju podataka sastoji se od nekoliko elemenata u kojima su glavne komponente Arduino Uno R3 sa ATmega328 mikrokontrolerom, digitalni senzor temperature DS18B20 i otpornici, povezani sa preko Breadboard-a prema programskom algoritmu. Pored toga, prikazane su relevantne specifikacije komponenti koje su prikazane šematski. Na kraju, cilj ove studije bio je izgradnja platforme za senzor sa mogućnošću evidentiranja podataka kako bi se postigla izvanredna funkcionalnost po značajno niskim troškovima, u odnosu na uređaje istog tipa koji su trenutno dostupni na tržištu.

Ključne reči: *Sistem za akviziciju podataka, temperatura, Arduino, DS18B20 sonda.*