Global Scientific JOURNALS

GSJ: Volume 12, Issue 6, June 2024, Online: ISSN 2320-9186
www.globalscientificjournal.com

DEVELOPMENT OF DIGITAL THERMOMETER WITH A MICROCONTROLLER FOR TEMPERATURE DETECTION

Aballa A. N.,[1] Oluka I. S.,[2] Nwobodo-Nzeribe N. H.,[3] Eneh O. A.,[4] Ojo A. O.,[5] Nnadi J. C.,[6] Aniude V. E.[7]

1, 4,6,7 Department of Biomedical Engineering, Federal University of Allied Health Sciences, Enugu, Nigeria.
2 Department of Agricultural and Bio-Resources Engineering, Enugu State University of Science and Technology, Enugu, Nigeria
3 Department of Computer Engineering, Enugu State University of Science and Technology, Enugu, Nigeria
5 Department of Pure and Applied Sciences, Federal University of Allied Health Sciences, Enugu, Nigeria
Corresponding Author Email: agathaaballan@gmail.com, Phone No.: +2347038772345

KeyWords
Arduino-nano, Digital, Liquid Crystal Display, Microcontroller, Multisim, Simulation, Temperature-Detection, Thermometer.

ABSTRACT

Measuring body temperature is a major technique used in the hospitals to get the vital signs of the patient to help detect clinical signs of systemic diseases. This study details the development of a low-cost digital thermometer to provide continuous and accurate real-time temperature of patients. The technology was developed to monitor the body temperature and to raise an alarm if the temperature exceeds a predefined threshold. The study was implemented using the Arduino nano Microcontroller (ATmega 328), DS 18B20 temperature sensor, Liquid Crystal Display (LCD), rocker switch, 9V battery, resistors, capacitors, and connecting wires as hardware requirements while Tinker CAD and Multisim 14.2 were used as simulation software. C++ programming language was used for sketch development. The system was soldered in the Veroboard, and was packaged carefully. When tested on five subjects represented by male (geriatric male and female), teenage male and female and one infant), satisfactory results were obtained with the following temperature readings (34.9oc, 34.8oc, 35.7oc, 35.8oc, and 36.3oc) respectively. The device triggered an alarm once the highest temperature of the body was reached. In conclusion, a digital thermometer for temperature detection has been developed with locally available resources without compromising standards.

INTRODUCTION

Thermometer has been a trusted instrument for measuring temperature ranging from the mercury filled type to the sleek digital models of today. Thermometers have played a major role in science and medicine despite their ubiquity. Temperature measuring sensors when built into thermometers measure temperature by sensing changes the body's physical characteristics. Traditional thermometers usually lack functionality, accuracy, and real-time data. A microcontroller is an integrated circuit device used to control other portions of an electronic system through a micro processing unit, memory and some peripherals. When connected with temperature sensor, microcontroller works as a tiny brain capable of transforming a thermometer into a sophisticated instrument by combining the precision of temperature sensors with the computational power of the microcontroller and can be used for temperature measurement. Microcontrollers have in-built analog-digital converters (ADC) which converts analog sensor signals into digital. Temperature sensors on the other hand work with a specific range of temperature to ensure accuracy while picking up the temperature reading. A programming language such as C++ is used to program the microcontroller to carry out its required task of converting the raw values from the sensor into meaningful temperature readings and also display the measured temperature on the liquid crystal display (LCD) while providing a power supply to provide stable voltage to the microcontroller and temperature sensor. This digital thermometer is developed to encourage temperature detection of patients at home by care givers, thereby reducing the constant visits of patients to hospitals for vital sign checkups. The developed digital thermometer has the unique features of enhancing safety by eliminating the use of mercury and breakable glass materials, improving functionality of existing thermometers by giving the temperature reading in both Celsius and Fahrenheit and Increasing versatility by programming it to alert users once the temperature reading is completed. These features are lacking in conventional thermometers.

[1] designed and constructed a digital thermometer based on ATmega 16 microcontroller using LM35 temperature sensor. The result of their system test showed a mean deviation of ± 0.5℃ from the reading when compared to a standard digital thermometer.

[2] developed a microcontroller based automatic temperature controller with PIC16F877A microcontroller interfaced with LM35DZ temperature sensor. Their design considered the predetermined temperatures of 26℃ and 29℃ as minimum and maximum temperatures respectively. The test results indicated that their developed system can be very useful in environments requiring temperature regulations.

[3] developed a digital thermometer using AT89C51 microcontroller and LM35 temperature sensor and analog to digital converter IC (ADC 0804), MAX-232IC chip for serial communication with the microcontroller. Their result revealed linear temperature output indicating that the timing and the system temperature calculations were successful and accurate.

[4] developed a digital thermometer using Arduino UNO based microcontroller. The result of their experimental test with their system revealed a highly accurate digital thermometer that nullified the effects of external parameters.

[5] developed an automated alarmed wearable thermometer using Arduino Uno Microcontroller (ATMega328P), LM35 Temperature sensor, and GSM SIM module 800L with the readings displayed on an LCD screen. Satisfactory results were obtained with their system when tested on eight subjects. The device gave an alarm at the reading of 38.05℃ with an SMS of the reading sent and received simultaneously by the physician's phone number programed into the device.

[6] designed and constructed microcontroller-based pulse rate and temperature monitor with GSM module reading device using DSI8B20 pulse sensor and MAX30162 temperature sensor, the device was able to take accurate readings of the pulse and temperature and also sent a short message service to the user's phone.

[7] developed a working prototype system for real-time health-monitoring temperature system using DS18B20 temperature sensor and Arduino Nano microcontroller and Zigbee module was used for wireless communication. The outcome of the system test was satisfactory after comparing it with the temperature obtained from a conventional medical thermometer. It worked properly the way it was designed for reading body temperature, transmitting over wireless communication and analyzed data can be saved and used for future references after receiving them.

MATERIALS AND METHODS

Both hardware and software were used for the system development. The hardware components include: Arduino nano (ATmega 328), temperature sensor (DS18B20), Liquid Crystal Display (LCD), alarm system, 9V battery, Resistors (4.7kΩ), capacitors, switch, soldering lead, Veroboard, breadboard and connecting wires. Software components include: Multisim 14.2 and Tinker Cad. C++ programming Language was used for sketch development.

Designing and Assemblage of the components: The casing chamber of the temperature detection device is designed to a size of 110cm in length, 50cm in width and 117cm in height giving a rectangular shaped box with an area of 5500cm^2 and volume of 643,500cm^3 on which are attached the Arduino board and other encased regulatory knobs. The LCD is soldered on to the center top of the rectangular casing measuring 90cm x30cm and an area of 2700cm^2. This is structured to accommodate the board and all connecting wires. Following the designing and assemblage of all the components, the casings were soldered and the casing components were screwed together.

Modelling and Simulation: In this project, the software used were Multisim 14.2 and tinker cad. The circuit diagram of the system was first designed and drawn with Multisim. An oscillator of 16mHz was used to generate the internal signal of the microcontroller while the internal clock frequency of the microcontroller was measured and maintained at 16mHz using an Oscilloscope. For simulation, the Tinker CAD software was used, the microcontroller was programmed by writing a simplified code snippet simulating the logic for reading the sensor data from the ADC, converting the raw sensor data to temperature units (Celsius and Fahrenheit) and preparing the temperature data for display on the chosen output device. The written code snippet was imported into the simulator environment and the simulator software to display the calculated temperature value in the chosen format on the virtual output device, executing the simulation within the software. This simulates the microcontroller showing the sensor data virtually, processing it based on the code, and displaying the calculated temperature. Different functions were created to handle specific tasks to help understand the codes and easily debug it. To debug the codes, serial print (value) was imputed with values to be evaluated at specific points in the code block. The codes were written in the Arduino integrated development environment (IDE) which were then uploaded into the microcontroller to execute them by interacting with other components of the device. During the simulation process, the system was broken down into separate simple blocks because the simulation software cannot simulate the whole system at once. These separate simple blocks were then tested in units. All the components were simulated and tested with the microcontroller.

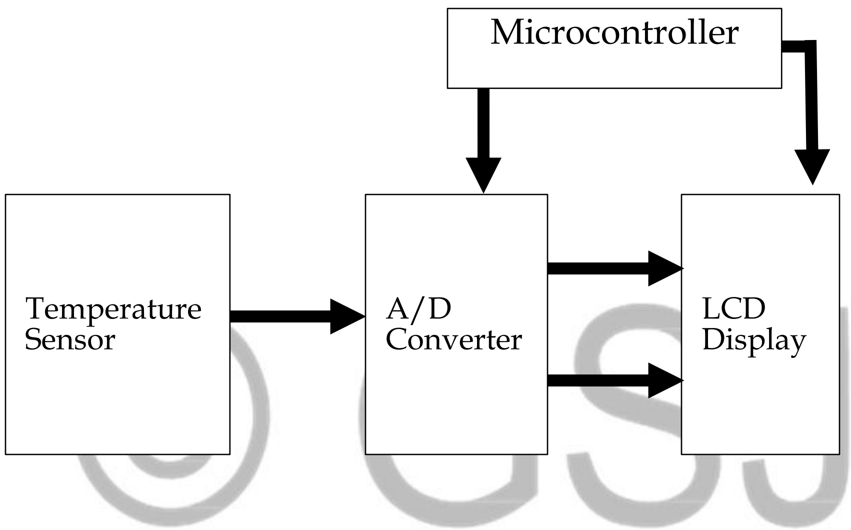

Fig 1a: Showing the block diagram of the temperature detection device

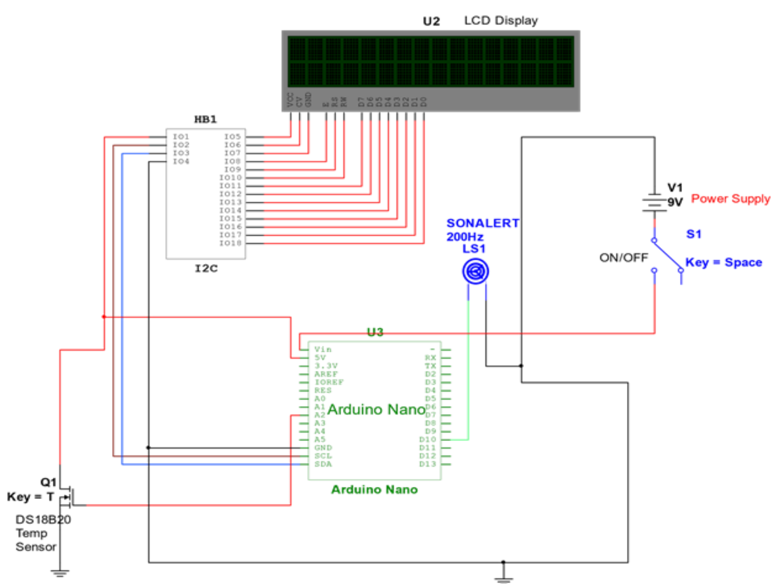

Fig 1b: Schematic diagram of the Microcontroller based thermometer

The block diagram depicts a simplified view of a temperature measurement system. The sensor detects temperature and converts it into an electrical signal. The microcontroller receives this signal, converts it to a readable temperature value using the analog to digital converter to both centigrade and Fahrenheit before transmitting to the LCD display for visualization. Figure 1b is the schematic diagram of the microcontroller-based thermometer. The temperature sensor was connected to the analog input pin of the Arduino microcontroller, this converts the voltage value from the sensor to digital format and determines whether the temperature reading is within the acceptable range. If the reading is within the range, it will be displayed on the screen but if not, a signal will be sent by the microcontroller through pin D10 to activate the alarm system.

Prototyping and Testing on breadboard:

The prototype was done using the design drawing and the model already created in the modelling and simulation stage. All the important mechanized parts from the design were incorporated to function as a whole. The bread board was used as a prototyping tool. This prototyping was carried out with codes written specifically for each component to be connected. The microcontroller unit was first connected to the bread board and making it light a simple LED on and off or continuously or subjecting it to a simple blink test. A pass on this test indicates that the microcontroller unit is working. The temperature sensor was connected and interfaced with the microcontroller on the bread board. The codes were written to read the temperature using a serial monitor to display the temperature reading on the integrated development environment of the microcontroller unit. A multimeter set to a voltage mode which changed as heat was applied was also used to test the temperature sensor. The readings were then compared to actual temperature values to check the workability. The screen was also monitored for output of the temperature readings by interfacing the I2C display module with the LCD and the microcontroller on the bread board. The alarm/ buzzer was linked with the microcontroller and then connected to the bread board. It was observed that as soon as the temperature became high up to 39℃ the alarm system was activated. The circuit wiring was done and all sections of the device were connected together to get a prototype.

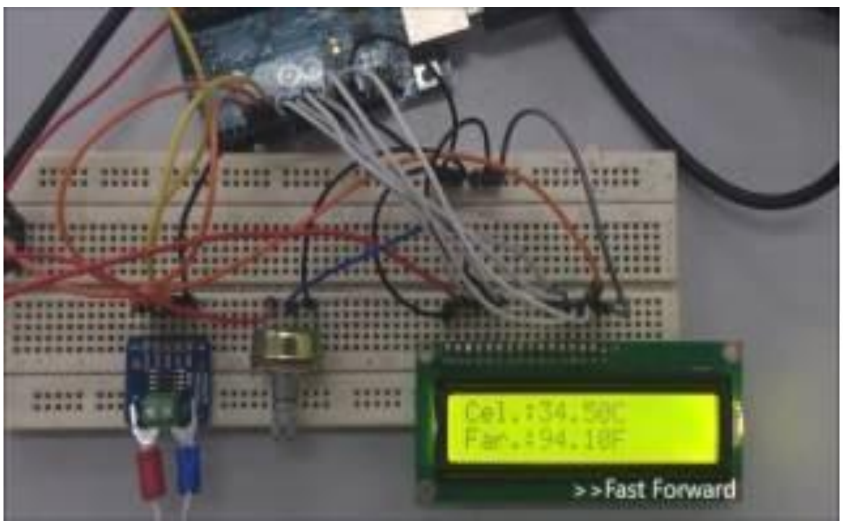

Fig 2: Prototyping and Testing on Breadboard

Prototyping and Testing on Veroboard: The next stage was prototyping and testing on a Veroboard. The Veroboard is a printed circuit board designed with rows of copper tracts with holes drilled in them for electronic components to be soldered to construct electric circuits. It is used for permanent prototyping. The layout of the circuit design was made, which showed how the components will be placed on the Veroboard. Components were then placed accordingly on the designed circuit layout. Soldering of component parts were done. Lead soldering was used because it has a lower melting point and causes fewer joint quality problems and poses a lower risk of adverse thermal effects on soldered components. However, precautions were taken to ensure that too much high temperature was not used to prevent damage to the components. Each soldered component was tested for workability. The prototype was formed after all the components were soldered together to form a whole.

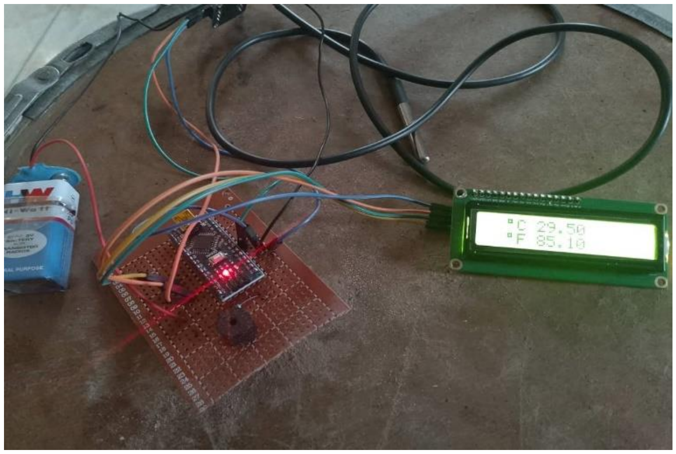

Fig 3: Prototyping and Testing on Veroboard

Troubleshooting and Repair: The prototype was inspected for defects and tested for continuity using the continuity tester. A digital multimeter was also used to test the prototype to determine the components that produce values significantly lower than the expected values and those with much greater or higher values above their expected outputs which indicated a problem. These components with problems were carefully isolated and were repaired or replaced completely and were carefully resoldered. The device was also tested for performance accuracy, functionality, safety and cost evaluation with satisfactory results observed.
Final packaging: Following the tests, the components were package and encased in the PVC rectangular box designed for the device. Again, a performance test was carried out finally on the device.

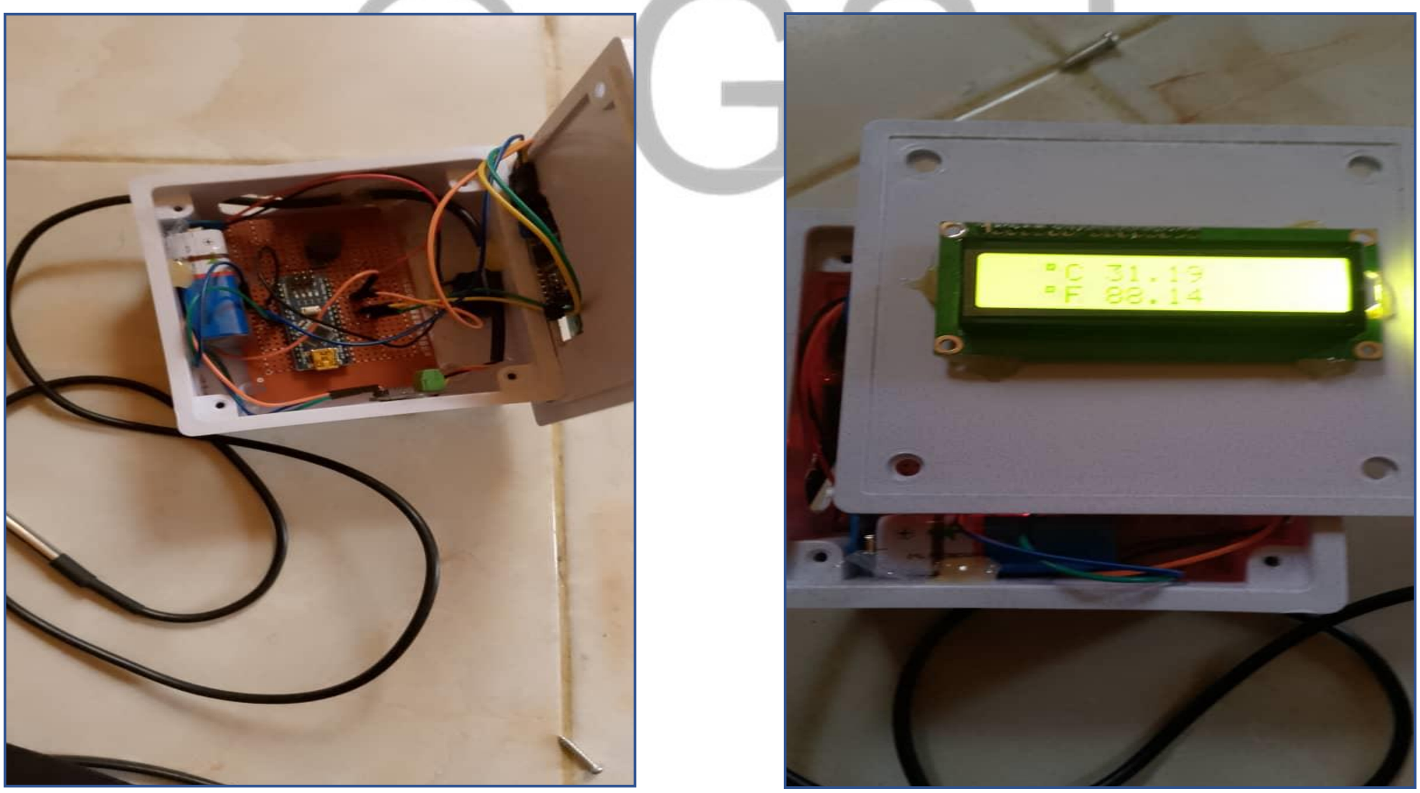

Fig 4: Packaged Prototype of the Developed Device

RESULTS AND DISCUSSION

Performance test: A performance test was carried out on the prototype to ensure its reliability and efficiency; the test was carried out on various patients and the result was satisfactory as shown below:

Table 1: Performance test result with the developed digital thermometer

S/N	Experiment	Outcome (℃)	Expected outcome (℃)	Error Difference (℃)
1	Male	36.4	35.0-36.5	+/-0.1
2	Female	36.3	35.0-36.5	+/-0.2
3	Teenage male	36.6	35.9-36.8	+/-0.2
4	Teenage female	36.7	35.9-36.8	+/-0.1
5	Infant	37.2	36.4-37.3	+/-0.1

Table 2: Result of the standard compliance test of the developed device compared to that of the clinical thermometer and Requirement accuracy under ASTM E1112---00 (2016)

Temperature range (℃)	Standard maximum error clinical thermometer	Error of developed wearable device
< 35.8	± 0.3	± 0.25
35.8 – 37.0	±0.2	± 0.2
37.0 -39.0	± 0.1	±0.1
39.0 -41.0	±0.2	±0.2

The prototype was tested on various individuals ranging from adults to infants for performance accuracy and its results were satisfactory. When the results were compared with standard values according to Anderson's loop, the result differed slightly but the values were still within the same acceptable range. The result of the standard compliance test is in agreement with the result of the study carried out by [4] as found in literature.

CONCLUSION

A digital thermometer with a microcontroller base for temperature detection to help physicians in monitoring patient's temperature in order to arrive at accurate diagnosis and treatment plan has been developed. This device has a programmable and customized functionality which differentiates it from other conventional thermometers. It is user friendly, portable and very efficient.

Acknowledgment

The authors wish to thank the Department of Biomedical Engineering, Federal University of Allied Health Sciences, Enugu, Nigeria, the Department of Agricultural and Bio-Resources Engineering, and the Department of Computer Engineering Enugu State University of Science and Technology, Enugu, Nigeria for their contributions and support throughout the period of the research.

References

[1] I.G. Saidu, M. Momoh, H.N. Yahaya, D.O. Akpootu, I. Z. Yaroko, S. A. Fagemi, and A. A. Sifawa, "Design and Implementation of a Microcontroller Based Digital Thermometer," *IOSR Journal of Environmental Science, Toxicology and Food Technology (IOSR- JESTET)*, vol. 8, Issue 2, pp. 119-125, Mar.- Apr. 2014.

[2] M.A. Abubakar, T. M. Adepoju, M.L. Rabiu, and U. A. Muslim, "Microcontroller Based Automatic Temperature Controller," *Proc. International Conf. of Science, Engineering and Environmental Technology (ICONSEET '17)*, 2(16): pp. 127-131. 2017.

[3] D. Rajdip, and K. Soumen, "Development of Digital Thermometer Using AT89C51 Microcontroller," *EPRA International Journal of Research and Development (IJRD)*, vol. 5, Issue 9, pp. 261-265, Sept. 2020.

[4] C. Bhavin, P. Krunal, D. Bhautik, S. Krushi and S. Kruti, "Digital Thermometer: Design and Implementation Using Arduino UNO Based Microcontroller," *International Journal for Scientific Research and Development*, vol. 4, Issue 3, pp. 840-843, Mar. 2016.

[5] N. H. Nwobodo-Nzeribe, A. N. Aballa, and C. C. Okafor, "Development of Automated Alarmed Wearable Thermometer," *Journal of Engineering and Computing (IJSAR-JEC)*, vol. 6, Issues 2,3,4, pp. 69-74, Dec. 2022.

[6] A. E. Amoran, A. S. Oluwole, and E. Olubakinde, "Design and Construction of a Microcontroller Based Pulse Rate and Temperature Monitor with GSM Module," *International Journal of Innovative Science and Research Technology (IJISRT)*, vol. 6, Issue 5, pp. 561-565, May. 2021.

[7] S. Ramesh, B. Suparna, S. Satyajit, K. Sushanta, and D. Pransh, "A Working Prototype Using DS18B20 Temperature Sensor and Arduino for Health Monitoring", *A Springer Nature Journal, SN Computer Science,* vol. 2:33, pp. 1-21, Jan. 2021.