

WEATHER MONITORING SYSTEM

Satyabrata Mishra, Abhisek Bera, Jayanta Kumar Behera

Jyostnamayee Behera, Assistant Professor,

Department of Electronics and Communication Engineering,

Gandhi Institute For Technology, Bhubaneswar.

(Affiliated to All India Council For Technical Education (AICTE))

ABSTRACT

We propose an IoT and cloud-based Weather Monitoring System here that aims, building a system to detect, record, and display different weather data like temperature, humidity, pressure and rain. This system employs sensors to detect and monitor meteorological factors, after which the collected data is transferred to Arduino cloud that can be accessed over the internet. It is possible to examine and report the data displayed as an output on a dashboard. A NodeMCU board, sensors, and a inbuilt esp8266 module are used in the system, which feeds data to Arduino cloud platform. In addition, a dashboard is constructed that presents the data to users.

INTRODUCTION

Weather Monitoring Systems are used to keep track of the ever-changing weather conditions. The information obtained by such sensors is used to report weather and maintain track of environmental changes in a given location. Such information is particularly useful in the study of the earth as well as the analysis of changing climatic and environmental conditions in a given location. Furthermore, the collected data and analytics can be used in a range of applications, including agriculture, geology, mining, and building weather forecasting models. A simple weather monitoring system is built in this project, which can monitor the temperature, humidity, barometric pressure and rain condition of a location.

This project's weather monitoring system is an IoT gadget based on NodeMCU. NodeMCU is an IoT board that works with Arduino. To write the program code for any NodeMCU board, we can use Arduino IDE or we can first create an account on the Arduino Cloud Platform and link his user account to the NodeMCU board. The program code can then be developed on the Arduino website's Web IDE and sent over the internet to a registered IOT board. If the selected board, in this case NodeMCU, is turned on and connected to the Arduino cloud service, the code is burned to the board over the air via an internet connection, and the board begins to operate according to the transferred code.

In our project we have interfaced four sensors for temperature, humidity, barometric pressure, rain and light intensity measurement those are DHT11, BMP180, FC-37, LDR module respectively.

OBJECTIVE OF THIS PROJECT

The design and deployment of a low-cost IoT device that can be used to monitor environmental changes at the location where it is installed is the primary goal of this project. The device should upload them to any cloud service where the information can be analyzed and processed,

The goal of a weather monitoring system is to detect, record, and display weather characteristics like temperature, humidity, barometric pressure and light intensity.

Another goal of this project is to make it more tailored, cost-effective, and personable. so that anyone may simply create their own personalized gadgets for personal use at a very low cost, and then alter them to fit their needs.

REVIEW OF LITERATURE

In [1], the author describes an IoT-based weather monitoring system. The environmental parameter can be collected by sensors in this study. The author employs a variety of sensors to scale various parameters such as humidity, temperature, pressure, and rain value, including the LDR sensor. The temperature prototype is also used to compute the dew point value. The temperature sensor can be used to determine the temperature of a certain region, room, or location. The light intensity can be employed as described by the author with the help of the LDR sensor. The author employed an additional functionality of weather monitoring in this as an SMS alert system depending on the value of sensing parameters such as temperature, humidity, pressure, light intensity, and rain exceeding the value of the sensing parameters.

The author of this work [2] depicts a low-cost live weather monitoring system using an OLED display. To measure the weather conditions, the author solely employs two devices: Wemos and OLED. Following the connection, the data will be stored in the cloud, and the data will be shown on the Thingspeak website. The data is shown on an OLED screen and in the cloud by the system. The author's goal is to obtain real-time weather information on an OLED display.

The author of the work [3] also depicts a similar NodeMCU based weather reporting system where a android application is being used to display the stored data in the Thingspeak cloud. The android application uses APIs to collect the data from the Thingspeak server and display the same in the dashboard. but in the system only a temperature and rain sensor is interfaced.

The importance of microcontroller sensor outputs for data storage and acquisition is overlooked in many research articles and studies in this field. The data acquired from the device can be processed and charted in real time with a weather station monitoring system, as seen in this article. That is, the details can be exhibited and observed in two ways: directly and indirectly. Direct methodology ensures that weather patterns are recorded and saved in a computer as long as the sensors calculate climatic conditions, whereas indirect methodology ensures that weather patterns are recorded and stored in a computer as long as the sensors calculate climatic conditions. The key challenge in this study is demonstrating and validating that microcontroller and their sensors can be coupled to a data collecting network to create a database system.

BLOCK DIAGRAM

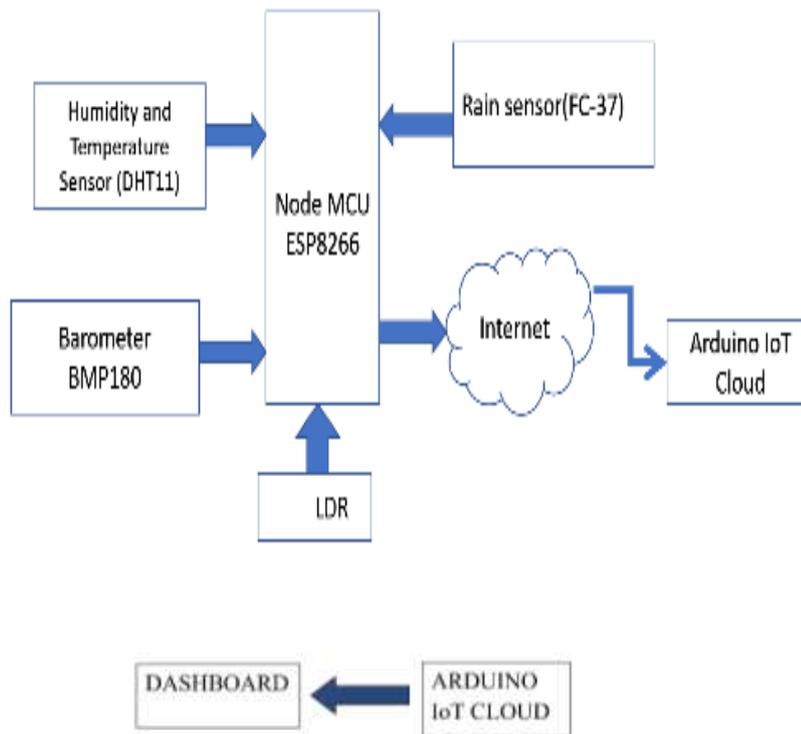


Figure 1 Block Diagram

WORKING PRINCIPLE

The device has four different sensor modules in it, those are DHT11, BMP180, FC-37 and LDR. All the sensors collect surrounding environment parameters and those values are sent to the IoT cloud by using the internet from the Wi-Fi module ESP8266 built inside the device. Then when we want to see the environment parameters value we go to the respective dashboard.

Talking in-depth, first when the whole system is powered on using the SMPS adapter, the first thing the device does is to try to connect to Wi-Fi nearby. The details such as the SSID and the Password for the nearby Wi-Fi is already provided in the code. The NodeMCU board has an ESP8266 chip on it which can connect to any nearby Wi-Fi.

After the connection is successfully established, individual sensors start to collect the surrounding environmental parameters that has been programmed to be measured. How those sensors measure the parameters that we have already discussed in the previous chapters on hardware required in the project. After that the updated data is being sent to the Arduino IoT cloud server as programmed. The respective variable data is being shown on the dashboard.

But before we want to see the dashboard, we need to login to our account in which the device and dashboard is being created. We login to our account either on the website or we can download the Arduino IoT cloud remote application on our mobile or desktop then login to see the data that is being transmitted to the cloud.

RESULT AND DISCUSSION:

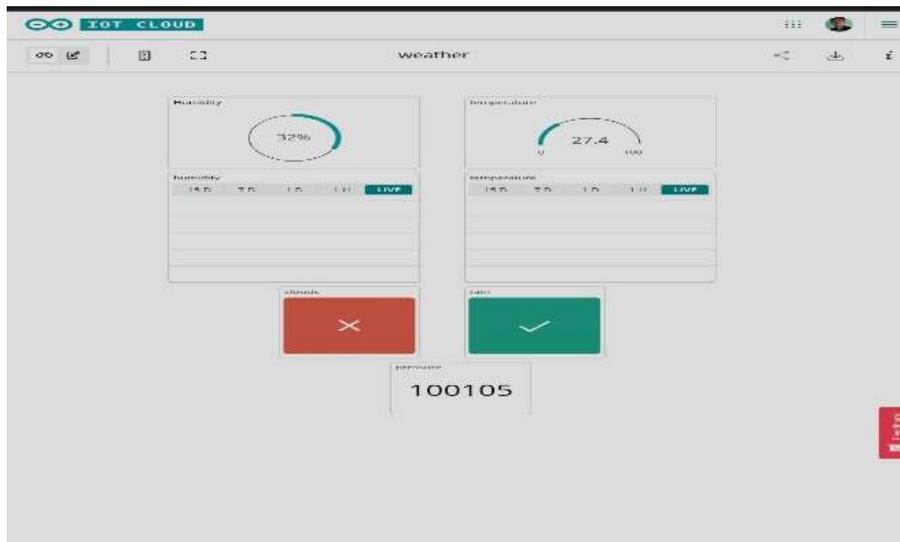


Figure 2 Desktop View



Figure 3 Mobile view

- we can say the sensors used for monitoring weather depends on the type application we going to deploy it for.
- Like we can say for a farmer soil moisture is important but for a pilot wind speed is more crucial.

CONCLUSION

- This is a smart way to monitor environment and an efficient, low-cost embedded system.
- We have successfully created and deployed a IoT based device that monitors the basic environmental parameters around us and sends the data to the server successfully.
- This data has successfully been displayed on the dashboard and saved for future uses.
- The dashboard successfully works both in mobile application and its website on desktop.

REFERENCES

- [1] Ravi Kishore Kodali and Snehashish Mandal “IoT Based Weather Station” 2016 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICT) 978-1-5090-5240-0/16/\$31.00, IEEE, (2016)
- [2] Ravi Kishore Kodali and Archana Sahu “An IoT based Weather Information Prototype Using WeMos” 2016 2nd International Conference on Contemporary Computing and Informatics (ic3i), 978-1-5090-5256-1/16/\$31.00, IEEE, (2016)
- [3] Kavya Ladi, A V S N Manoj, G V N Deepak, “IOT Based Weather Reporting System to Find Dynamic Climatic Parameters”, International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS-2017)
- [4] P. Susmitha, G. Sowmyabala “Design and Implementation of Weather Monitoring and Controlling System”, International Journal of Computer Applications (0975 – 8887) Volume 97– No.3, (July 2014)
- [5] Tanmay Parashar¹, Shobhit Gahlot², Akash Godbole³, Y.B. Thakare⁴ “Weather Monitoring System Using Wi-Fi”, (IJSR) ISSN (Online): 2319-7064 Index Copernicus Value (2015): 78.96, 2015
- [6] <http://download.arduino.org/products / UNOWIFI /0A-ESP8266-Datasheet-ENv4.3.pdf>
- [7] Kumar, Abhijeet, and Apoorva Sharma. "Internet of Life (IOL)." (2015). ISBN 978-93- 5156-328-0
- [8] Zeroday. "A lua based firmware for wifi-soc esp8266". Github. Retrieved 2 April 2015.
- [9] Brian Benchoff. "a DEV board for the ESP LUA interpreter". Hackaday. Retrieved 2 April 2015
- [10] <https://espressif.com/en/products/hardware / esp8266ex/overview>