Journal of Physics: Conference Series

PAPER • OPEN ACCESS

Design & Implementation of Solar Powered Automatic Weather Station based on ESP32 and GPRS Module

Journal of Physics: Conference Series **1737** (2021) 012009 doi:10.1088/1742-6596/1737/1/012009

Design & Implementation of Solar Powered Automatic Weather Station based on ESP32 and GPRS Module

A S Priambodo[1] and A P Nugroho[2]

[1]Electronics and Informatics Engineering Education Departement, Universitas Negeri Yogyakarta, Yogyakarta, Indonesia
[2]Agricultural and Biosystems Engineering Departement, Universitas Gadjah Mada, Yogyakarta, Indonesia

E-mail: ardyseto@uny.ac.id

Abstract. The fundamental aim of this project is to develop a solar-powered automatic weather station (AWS), which can be accessed via the website. Users can find out the weather changes in an area without needing to come to the area and can do an analysis of irrigation water needs. This design uses ESP32 as main processor. The measure weather parameters include temperature and humidity using HDC1080 sensor, wind speed using an anemometer sensor, wind direction using a wind vane sensor, air pressure using BME280, rainfall using a tipping bucket sensor, and the last small solar panel for irradiance sensor. AWS has two working modes, normal and maintenance mode. During maintenance mode, sensor data will be displayed on a local website that can be accessed via a wifi network broadcasted by ESP32. In normal mode, the ESP32 will send sensor data to the cloud using SIM800L GPRS Module. The system proposed is also designed to have a feature to log sensor data locally in SD card. Watchdog timer circuit uses timer 555 implemented in this project for the recovery system from the failure process. Power sources are limited, so the system must be put into deep-sleep mode state when not processing data. Data transmission is carried out periodically with an interval of 5-6 minutes. Test results show that the sending of sensor data can be received by the cloud with an acceptance rate of up to 98%.

1. Introduction

Agriculture is one of the essential sectors of Indonesian society [1]. The villagers use large areas of land to become rice fields. Rice yields planted by farmers are influenced by many factors, including climate change, irrigation water availability, soil fertility, types of plant varieties, crop management systems, and development of pests and diseases in plants.

Plant growth and development can be disrupted due to insufficient or excessive water requirements for plants. The irrigation system is a network system with techniques and methods used to provide, regulate, and channel water to agricultural land, plantation, or other crop cultivation land. In general, farmers or irrigation system officers visit agricultural land to periodically see land conditions and regulate the flow of irrigation water by opening or closing the water-gate according to the farmer or irrigation system officer's perspective. Hence, the distribution and distribution of water are sometimes ineffective and less efficient. An irrigation system is needed that can increase effectiveness and efficiency so that water availability in the irrigation system is always maintained by applying the principle of justice.

The issue of climate change and global warming has pushed the importance of modernizing irrigation so that irrigation management becomes more effective [10]. Irrigation modernization is a change in the irrigation management and development system into a participatory irrigation system that is more effective, efficient, and sustainable in supporting food and water security by increasing the reliability of water supply, infrastructure, irrigation management, management institutions, and human resources [2].

Modernization of irrigation is related to data, so that it is necessary to use technology that can be used for data collection, data transmission, data processing, and data presentation. The use of irrigation software is the use of technology that can be used for decision making. The operation decision-making is based on the irrigation system's behaviour in computer operation and is carried out on the existing irrigation system. The use of irrigation operation software is one of the components of irrigation modernization requiring preparation both in terms of hardware and system parameters so that the behaviour of the operating software is following the whole system.

Collecting data is one of the critical stages in the development of modern irrigation system software. The need for hydrological and hydro-climatological data in irrigation areas is needed to make decisions regarding irrigation management appropriately. Weather parameter information such as temperature, humidity, and rainfall at a specific location and time must be known quickly to support irrigation system [3].

Data collection requires special equipment installed at the observation site. The solution to this problem is the implementation of the use of Internet of Things devices. Internet of Things or commonly abbreviated as IoT is a system where devices are connected & integrated [11]. This research aims to design a solar-powered automatic weather station system using an ESP32 microcontroller and a GPRS module. The system being designed must have an independent energy source because of its location far from residential areas. Also, the use of connection devices must adjust to conditions where there is no 3G/4G signal so that the use of the GPRS module is a solution. The designed system can retrieve data as needed and then send it to the cloud server to analyze irrigation modernization needs. The hope is that with the fulfillment of data needs, the modern irrigation system can be appropriately implemented to increase agricultural products' quality and quantity.

2. Materials and Methods

2.1. Materials

The main parts of the automatic weather station are shown in Figure 1. The system uses an ESP32 microcontroller as a data processing center and several sensors and peripherals. ESP32 is a powerful 32-bit microcontroller from Espressif Systems with integrated Wi-Fi, full TCP/IP stack for internet connection, and Bluetooth 4.2 [4]. The ESP32 has a 32bit LX6 dual-core Xtensa microprocessor, frequency clock up to 240Mhz, 520kB SRAM, 4MB flash memory, and consists of 48 GPIO pins. Due to the low cost combined with great power and the opportunity to connect the ESP32 to many other electronic devices, the microcontroller is well suited for IoT projects. In this research using the ESP32 as an Automatic Weather Station processor that handles sensors and peripherals installed on AWS. Solar panels with 20 WP capability are used as a source of electric power in the designed system. The electric power produced by the solar panels is then stabilized using a solar controller and stored in a 12V 6A battery.

Based on data requirements for irrigation system settings [5], the sensors used in this study are temperature and humidity sensors, rainfall gauge sensors, air pressure sensors, light irradiation sensors, wind speed sensor, wind direction sensor, and voltage sensor. In addition to sensor devices, there are peripherals installed on AWS such as the GPRS module, RTC module, power supply module, and ADC module. The schematic of the proposed system is shown in Figure 1.

HDC1080 is a low power, high accuracy digital temperature, and humidity sensor. This sensor's temperature measurement range is -20ºC to 85ºC, while the measurement range for relative humidity is between 0% to 100% [6]. This sensor has a temperature measurement accuracy of ± 0.2ºC and accuracy of measuring ± 2% humidity. This sensor can be accessed using I2C communication via the SDA and SCL pins on the ESP32. This sensor is a sensor with low power consumption, so it is suitable for the system proposed.

The rainfall measurement sensor uses a tipping bucket type where every bucket full of water will occur one digital signal, which is 0.279mm worth. The ESP32 then reads this signal change as an interrupt signal. The wind speed sensor used is a cup type where the 2.4km/h wind speed will activate the sensor's switch every second. Wind speed readings are carried out at a certain period so that the average wind speed is obtained. Wind direction sensors that can measure eight wind directions (N, NE, E, SE, S, SW, W, NW) based on different resistance values in each wind direction.

The BMP280 is a sensor that can measure air pressure from the Bosch Sensortec. This sensor can measure the relative air pressure of 900 to 1100 hPa [7]. The low current consumption of around 2.74uA makes this sensor excellent for use in equipment with limited electric sources. Measuring the value of solar irradiance can be done using solar panels. However, solar panels here are not meant to be solar panels used as a power source for the system. Solar irradiance is the power per area received from the sun in the form of electromagnetic waves in units of W/m2. So that by using a solar panel with a known area, the solar irradiance value can be calculated. The measurement range of this sensor is 0W / m2 to 800W/m2.

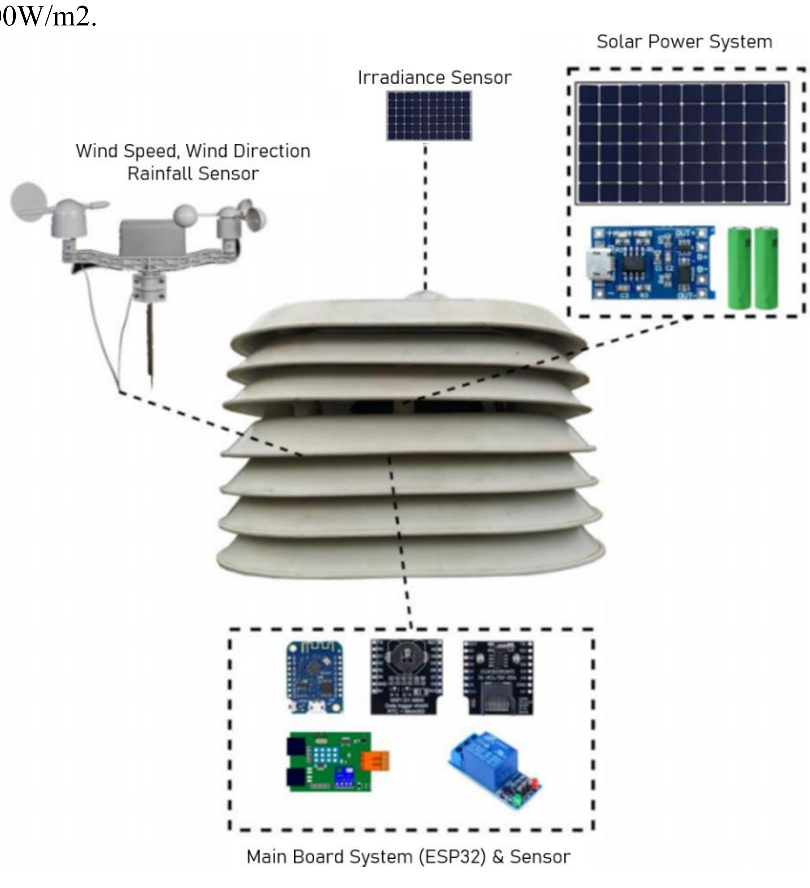

Figure 1. Automatic Weather Station Part & Scheme

DS3231 is a real-time clock module that can store time. a CR2302 battery is installed as a backup power so that the set time is not lost. The RTC module is essential because it is used to know when sensor data is taken. The DS3231 is accessed using the same I2C communication as the sensors previously mentioned. The voltage sensor is used to read the battery voltage by using a voltage divider circuit. This circuit consists of 2 resistors arranged in parallel and read using analog pins on the ADC module. This circuit is needed because of the limited voltage reading on the ADC module with a maximum value of 5V. The result of the analog signal reading is then converted to the measured battery voltage value.

The last component is the SIM800L GPRS Module which functions as a connectivity medium with the cloud server. SIM800L accessed by using AT Command via serial communication connected to the ESP32. All components are installed on a single PCB to ease the installation and maintenance process.

2.2. Methods
This system development has an independent energy source using solar panels because the installation location is far from residential areas. The system proposed can read sensor data in real-time and log sensor data to the SD card and send it to the cloud server so that users can easily monitor changes that occur. The working principle of the system created is shown in the flow chart in figure 2.

The proposed system has two working modes, maintenance mode, and normal mode. Maintenance mode is a mode where users can configure, such as changing the API key used on the server, changing the parameters of some sensor values, and testing data delivery to the server. All activities in maintenance mode are carried out on local websites broadcast by ESP32. In this condition, the user can also check the sensor's value, making it easier to check the condition of the sensor.

Normal mode is used when the system is configured and ready to run. In this mode, the ESP32 will retrieve sensor data every 5 minutes, do logging to the SD card, and send sensor data to the cloud server. If the data is sent successfully, the ESP32 will enter deep-sleeping mode so that power consumption decreases, and the battery's energy source does not run out quickly.

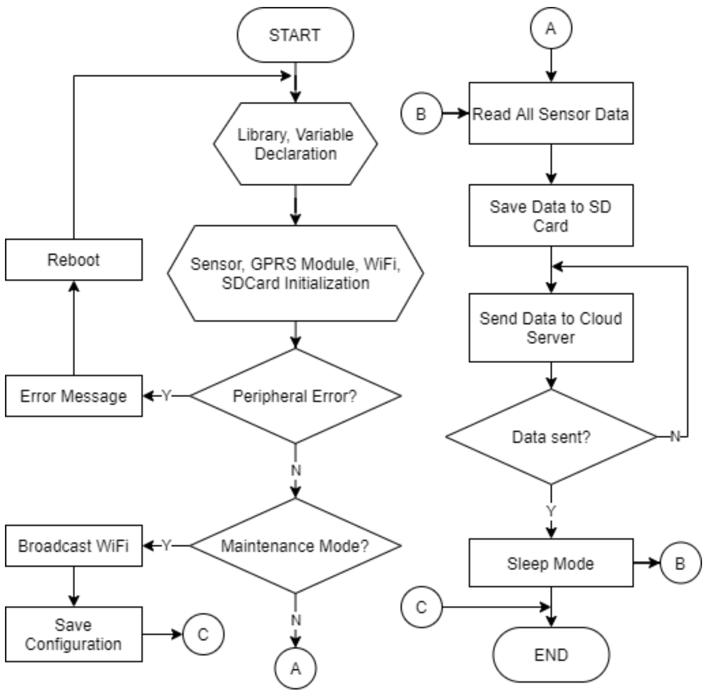

Figure 2. Flowchart for Automatic Weather Station

A watchdog timer is installed in the system to increase the robustness of the proposed system. The watchdog timer is hardware (or hardware emulation), which can automatically reset the system in software fault [8]. The watchdog timer circuit on the AWS system uses the 555 circuits configured as an astable oscillator. The component resistor and capacitor determine the frequency, and thus a period of the oscillator. In its current configuration, the time until reset is about 69 seconds. At the end of that period, the circuit will generate a LOW pulse that will reset the attached microprocessor. On the server, a program with a RESTful API is designed so that it can receive data via HTTP requests via the

POST command. The use of this RESTful API makes device development easier because each device has own API key. The cloud server provides database slots from C0 to C15 so that the cloud server can accommodate 16 data slots. weather station proposed uses slots C0 to C9 and C14 in the order in table 1.

Tabel 1. Database slots and weather station parameters

Database	Parameter	Sensor	Unit
C0	Battery Voltage	Resistor Voltage Regulator & ADS1115	Voltage
C1	Solar Irradiance	Solar Panel	W/m^2
C2	Air Pressure	BMP280	hPa
C3	Temperature In	BMP280	^oC
C4	Temperature In	DS3231	^oC
C5	Temperature Out	HDC1080	^oC
C6	Humidity	HDC1080	%
C7	Rainfall	Tipping Bucket Rainfall	mm
C8	Wind Speed	Anemometer	Km/h
C9	Wind Direction	Wind Vane	-
C14	Unixtime	DS3231	-

3. Result and Discussion

The system designed is developed into a prototype and tested in the laboratory before being tested in the field. All parameters (battery voltage, solar irradiance, temperature, humidity, rainfall, wind speed and wind direction) can be read correctly, and the data is received by the server smoothly. Besides, the deep-sleep mode test works well, and it is proven that when the deep-sleep condition the current consumption is measured in the ampere meter, it shows about 257uA and during normal conditions it is measured around 74mA. The watchdog timer circuit also works well, proven when there is no trigger given by the ESP32 for more than 69 seconds, and the ESP32 will reset.

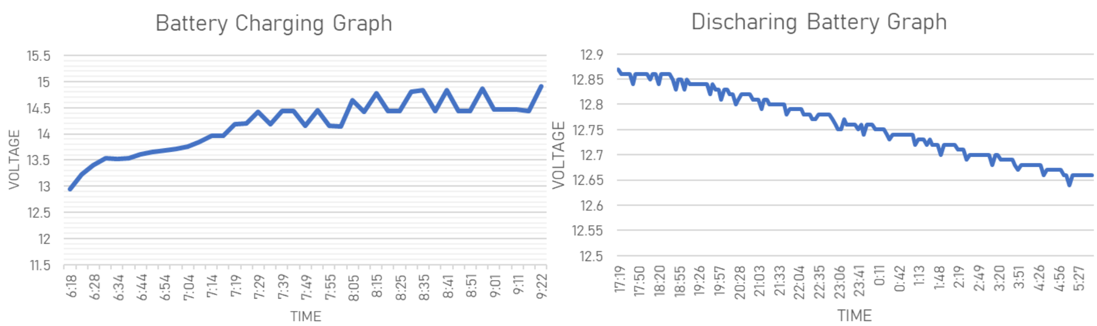

Figure 3. (a) Graph of Battery Charging (b) Graph of Battery Discharging

After the two functions mentioned above are running well, then the system is installed in the field and from the battery voltage data it can be seen that the battery used can handle the needs of a power source for one day without any problems. Seen in Figure 3, the charging process from a voltage of 12.8V to 14.8V takes about 3 hours with conditions not cloudy/rainy. In the same picture, it can be seen that the discharging process starts when it enters the afternoon where there is no sunlight, and the next morning the measured voltage shows a voltage of 12.66V.

AWS is designed to support two work modes, normal and maintenance mode. There is a switch to activate this mode before AWS is turned on. Maintenance mode is made to make it easier for users to configure before installation. This mode is activated by changing the switch to the OFF position. Figure 4 shows the appearance of a local website broadcast by ESP32 in maintenance mode accessed via Android. In this mode, there is also a button to test delivery via the GPRS module. The existing AWS mostly uses LCDs to display data [9][12], but the system proposed in this research does not use LCDs due to power consumption considerations.

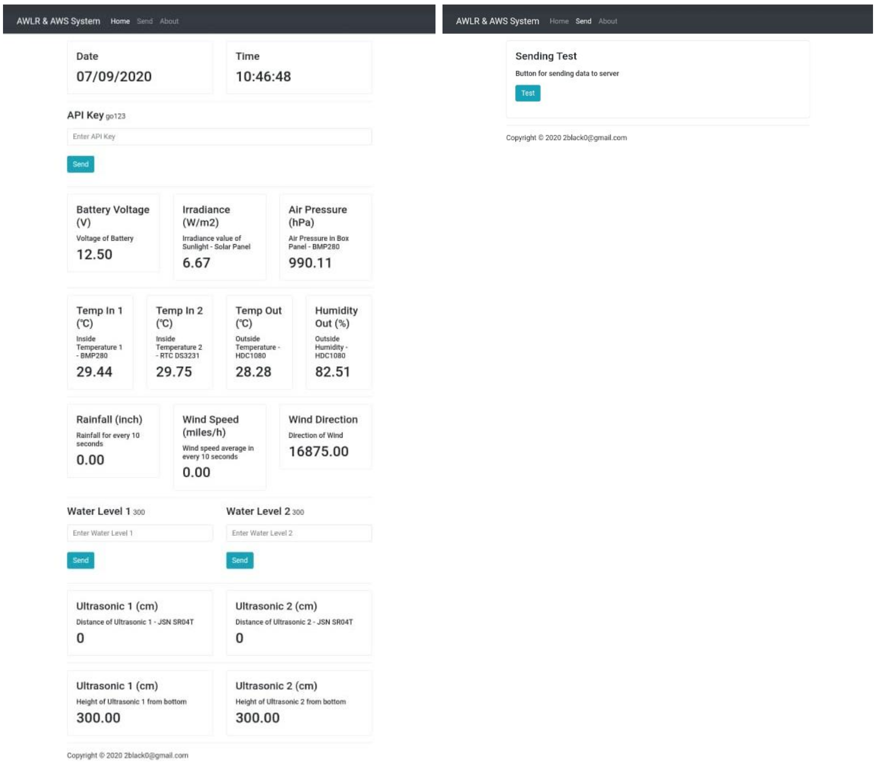

Figure 4. (a) Main tab for sensor display in maintenance mode (b) Sending test tab for GPRS module testing

The results of collecting sensor data and sending data to the server after AWS are installed in the irrigation area are shown in Figure 5. There are several columns in the database such as id which is the data sequence number, name is the name of the AWS device, time shows the time when data is received by the server, and C0 to C15 is the database slot for storing sensor data. C10 to C13 is a slot for other sensor requirements that are not included in the proposed system. Three temperatures measured at C3, C4, and C5, where the values of C3 and C4 are not much different, while C5 has a difference due to the location of the sensor installation. The value of C8 is the wind direction (N, NE, E, SE, S, SW, W, NW), which is converted to numbers 1-8, which means number 6 in the direction of SW. Time sent to the database is in unix time format, so it needs conversion so that it is easy to read in the date & time format. The Unix time value on C14 at id 312252 is 1601673064, which converted to 10/02/2020 @ 9:11 pm (UTC). All sensor data can be read, and the transmission runs smoothly as expected. Out of 100 times, only 2 data were not sent. This means the accuracy using the GPRS module in this design is 98%.

id	name	time	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12	C13	C14	C15
312252	Aws re	2020-10-02 21:11:50	13.05	0.02	1012.19	28.39	28.25	26.77	98.79	0	7.2	6	0	300	0	300	1601673064	
312239	Aws re	2020-10-02 21:05:57	13.05	0.03	1012.12	28.43	28.25	26.81	98.49	0	7.2	6	0	300	0	300	1601672712	
312226	Aws re	2020-10-02 21:00:02	13.05	0.03	1012.08	28.45	28.25	26.8	98.59	0	7.2	6	0	300	0	300	1601672355	
312213	Aws re	2020-10-02 20:54:08	13.05	0.02	1012.04	28.48	28.5	26.8	99.58	0	7.2	6	0	300	0	300	1601672002	
312201	Aws re	2020-10-02 20:48:15	13.05	0.02	1012.08	28.5	28.5	26.86	99.93	0	2.4	6	0	300	0	300	1601671649	
312188	Aws re	2020-10-02 20:42:22	13.06	0.01	1012.07	28.56	28.5	26.9	99.04	0	2.4	8	0	300	0	300	1601671295	
312175	Aws re	2020-10-02 20:36:28	13.06	0.02	1012	28.51	28.5	26.9	98.44	0	4.8	6	0	300	0	300	1601670941	
312162	Aws re	2020-10-02 20:30:33	13.06	0.01	1011.88	28.55	28.5	26.89	98.65	0	2.4	6	0	300	0	300	1601670588	

Figure 5. Table mode data display on the web cloud server

4. Conclusion

The development and implementation of a solar-powered automatic weather station for the data needs of irrigation system management using the ESP32 and the GPRS module are capable of providing the user with weather situation and conditions prevailing in and around the agriculture field. The use of the ESP32 as the processor in this design is also a very appropriate choice because of the availability of a wifi feature that can be used for maintenance mode which makes it very easy for users to perform the stages before installation. Developing the system proposed is the first step in implementing the modernization of irrigation in the agriculture sector. As a future enhancement, along with the increasing number of weather stations, using Lora technology will save development costs. Also, by using Lora, the data loss rate will be reduced.

References

[1] Bashir, A. and Susetyo, D., 2018. The relationship between economic growth, human capital, and agriculture sector: Empirical evidence from Indonesia. International Journal of Food and Agricultural Economics (IJFAEC), 6(1128-2019-554), pp.35-52.

[2] Castelli, G., Bresci, E., Castelli, F., Hagos, E.Y. and Mehari, A., 2018. A participatory design approach for modernization of spate irrigation systems. *Agricultural Water Management*, *210*, pp.286-295.

[3] Bonfante, A., Monaco, E., Manna, P., De Mascellis, R., Basile, A., Buonanno, M., Cantilena, G., Esposito, A., Tedeschi, A., De Michele, C. and Belfiore, O., 2019. LCIS DSS—An irrigation supporting system for water use efficiency improvement in precision agriculture: A maize case study. *Agricultural Systems*, *176*, p.102646.

[4] Maier, A., Sharp, A. and Vagapov, Y., 2017, September. Comparative analysis and practical implementation of the ESP32 microcontroller module for the internet of things. In *2017 Internet Technologies and Applications (ITA)* (pp. 143-148). IEEE.

[5] Lorite, I.J., Ramírez-Cuesta, J.M., Cruz-Blanco, M. and Santos, C., 2015. Using weather forecast data for irrigation scheduling under semi-arid conditions. *Irrigation science*, *33*(6), pp.411-427.

[6] Tzortzakis, K., Papafotis, K. and Sotiriadis, P.P., 2017, November. Wireless self powered environmental monitoring system for smart cities based on LoRa. In *2017 Panhellenic Conference on Electronics and Telecommunications (PACET)* (pp. 1-4). IEEE.

[7] Bonilla, J.E.O. and Saleem, A., 2020. Low cost real-time sensor monitoring through three different IoT's Plataforms.

[8] Johnston, S.J., Apetroaie-Cristea, M., Scott, M. and Cox, S.J., 2016, December. Applicability of commodity, low cost, single board computers for Internet of Things devices. In *2016 IEEE 3rd World Forum on Internet of Things (WF-IoT)* (pp. 141-146). IEEE.

[9] Mahmood, S.N. and Hasan, F.F., 2017. Design of weather monitoring system using Arduino based database implementation. *Journal of Multidisciplinary Engineering Science and Technology (JMEST)*, *4*(4), p.7109.

[10] Fader, M., Shi, S., Bloh, W.V., Bondeau, A. and Cramer, W., 2016. Mediterranean irrigation under climate change: more efficient irrigation needed to compensate for increases in irrigation water requirements. *Hydrology and Earth System Sciences*, *20*(2), pp.953-973.

[11] Javed, A., 2016. Building Arduino Projects for the Internet of Things. Experiments with Real-World Applications. United States of America: Apress Media, LLC, pp.15-34.

[12] JingXin, G.A.O., Hao, M.A. and Hao, L.I.U., 2016, May. The intelligent weather station system based on Arduino. In *2016 International Conference on Engineering and Advanced Technology (ICEAT 2016)*. Atlantis Press.