23

by Rina Mardiati

Early Warning System of Flood Disaster Using JSN-SR04 and Rainfall Sensor Based on Internet of Things

Irfan
Department of Electrical Engineering
UIN Sunan Gunung Djati Bandung
Bandung, Indonesia
1187070032@student.uinsgd.ac.id

Rina Mardiati
Department of Electrical Engineering
UIN Sunan Gunung Djati Bandung
Bandung, Indonesia
r_mardiati@uinsgd.ac.id

Mufid Ridlo Effendi
Department of Electrical Engineering
UIN Sunan Gunung Djati Bandung
Bandung, Indonesia
Mufid.ridlo@uinsgd.ac.id

Abstract— Flood disasters are a problem that is difficult to detect because of the sudden arrival. Flooding is caused by the high intensity of rainfall with a long duration until river water overflows and it will make some property losses and even casualties. Therefore, the development of technology that provide early warning system is needed for reduce the risk of losses caused by floods. This study will combine the JSN-SR04 sensor with rainfall equipped with the Internet of Things (IoT) technology. This proposed system can detect the changes of water level, rainfall intensity and the system also can predict when flooding will occur. With the support of IoT technology, the system can send an information about water level and rainfall data to mobile phones installed with AWS applications. In this study, the communication system was designed into two parts, namely the sender module and the receiver module. In the sender module we used the programmable Atmega328P microcontroller for processing the data from the JSN-SR04. This sensor will read into water level values and rainfall sensors into rainfall intensity values, then the data will be sent to the receiving module using the Long Range (LoRa) module. In the receiver module using the programmable ESP32 microcontroller, the microcontroller and LoRa in this module will process the data that has been sent by the sending module and classified into flood levels and precipitation levels and then the data is sent to the AWS application installed on the mobile phone. In addition, in this receiver module there is an OLED that functions to display data, relays and sirens to provide warnings in the form of sounds when the flood level or water level is at level 3. Based on the results of implementation and testing at the ciyasana dam, data on flood and rainfall levels are at level 1 by following the provisions in the ciyasana dam.

Keywords—Internet of Things, Sender Module, Receiver Module, Flood Detection System

I. INTRODUCTION

Flood disasters are a problem that continues to occur throughout the world, one of which is in Indonesia. This disaster is an event due to nature or human habitual behavior. Climate change, land, increasing population in Indonesia and lack of rainwater infiltration resulting in a greater threat of disasters in Indonesia, one of which is flooding [1][2].

From data obtained by the National Disaster Management Agency (BNPB) during the July 2021 period there have been 130 natural disasters, with the highest disaster events, namely floods 53 times, forest fires 42, tornadoes 22, landslides 11, earthquakes 1 and droughts 1 [3]. Until now, flooding is still a problem that often occurs and must be overcome immediately to minimize casualties, environmental damage, and property. Flooding is caused by the high intensity of rainfall with a long enough duration, so that river water can potentially overflow onto land [4][5]. Several studies on flood disasters have certainly been carried out by several researchers, one of which is by Choo Kam Khuen and Alireza Zourmand about an internet of things-based flood disaster early warning system, in the study explained the flood early warning system uses ultrasonic sensors to detect changes in water levels and rainfall sensors to calculate rainfall intensity. The monitoring application in this study used the Blynk application, so that the data from the sensor readings can be see on the Blynk application installed on the cellphone [6].

Therefore, with the development of technology based on internet of Things, it is necessary to have a flood disaster early warning system [7]. With several references related to research on flood disaster early warning systems, this study made a flood disaster early warning system using JSN-SR04 sensors and Rainfall Based on the Internet of Things. This system is divided into two modules, namely the sender module and the receiver module. The sending module uses the Atmega328P microcontroller, the JSN-SR04 ultrasonic sensor which is used to detect changes in Water Level Height, the Rainfall sensor which serves to calculate the intensity of precipitation, the voltage sensor to find out the remaining battery and the Long Range (LoRa) is used to transmit data to the receiving module, this sending module uses a power source from the Solar Power Plant. In the receiver module, there is only LoRa which functions to receive data from the sender module, ESP32 as the main microcontroller and siren to provide warnings in the form of sound.

This JSN-SR04 ultrasonic sensor has several advantages over ordinary ultrasonic sensors, including waterproof and long-distance detection range with an accuracy of +0.5 cm and – 0.5 cm. [8]. The results readings of the two sensors are classified into flood levels and then sent to the cloud monitoring application (AWS). So that this tool can provide early warning to the public regarding potential floods. [8][9].

II. DESIGN AND IMPLEMENTATION

A. Block Diagram System

The design of the flood disaster early warning system is divided into two parts, namely the sender module and the receiver module. In the sending module, there are sensors, namely the JSN-SR04 ultrasonic sensor, rainfall, voltage

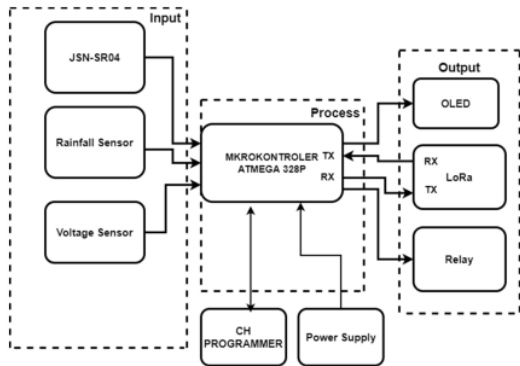sensor, and LoRa, while the receiver module uses esp32 microcontrollers, LoRa, and sirens.

Fig. 1. Block diagram module sender.

The CH Block is used to connect the microcontroller with the computer to be able to upload programs from the computer to the sender module and power supply serves to lower the voltage and supply it to all components connected to the microcontroller [4].

In this sender module, there are three main parts, namely input, process, and output.

1. Input

The components used are JSN-SR04 sensor which is used to detect water level, rainfall sensor to measure rainfall intensity and voltage sensor which is used to determine battery voltage [8][10].

2. The process

This section uses the Atmega 328P microcontroller. This section serves as the main control for processing measurement data from sensors. Atmega 328P is an 8-bit AVR family microcontroller, it provides 13 digital pins and 6 analog pins. This microcontroller has a flash memory capacity of 32Kb, 1Kb EEPROM, and 2Kb internal SRAM [4][11].

3. The Output

In this part, the module uses three components, namely OLED, LoRa, and Relay. Relays are used to turn on or off the sirens, OLED are used to display data that has been processed by Atmega328P [4][11]. But the sirens and OLEDs on the sender module are only optional because this sender module is used to process the data of the sensor measurement results and send the data to the receiver module using LoRa.

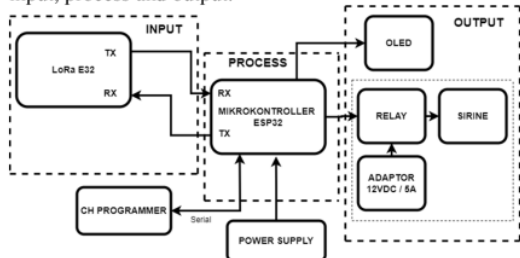The receiver module is divided into three main parts, namely input, process and output.

Fig. 2. Block diagram module receiver.

The CH Programmer block in this input section is used to upload programs from the computer to the ESP32 microcontroller or to connect the computer with the ESP3 receiver module. The Power supply block is the main voltage source to supply all components connected to the ESP32 [4].

1. The input

In this input section there is 1 module, namely LoRa. Lora is a transceiver module (which can receive or transmit radio waves. The LoRa used in this study is LoRa E32-433T20D, this LoRa uses Integrated Circuit (IC) SX1278 made by SEMTECH. This LoRa can send or receive data with a maximum distance between the sender and the receiver three KM receiver without any obstacles such as tall buildings or trees, LoRa in this receiver module is used to receive data sent by the sender module [12][13].

2. The process

This section uses the ESP32 microcontroller. This microcontroller is a System on Chip (SoC) equipped with WiFi 802.11 b/g/h, Bluetooth V4.2, and other hardware. This ESP32 is used to process data sent by the sender module using LoRa and is calcified into flood levels. When the flood is at level one, the ESP32 will not trigger the relay until the siren will off and will send the classification data to the Firebase database. Then when the flood is at level two, the ESP32 will also not trigger the relay so that the siren will die and will send the classification data to the Firebase database. Meanwhile, when the flood is at level three, the ESP32 will trigger the relay so that the siren will be on and send the classified data to the real-time Firebase database [14][15].

3. The output

In this section there are three components, namely OLED and relay. OLED is a type of LCD (Liquid Crystal Display) that has a small size but produces brighter and sharper images. OLED is used to display data that has been processed by ESP32. The OLED used is 128x64 pixels in size and white. In addition, OLED also consumes very little power compared to ordinary types of LCD. Then the relay is used to control (turn on or off) the device programmatically with a HIGH or LOW trigger [4][11].

B. Hardware Design

In designing a flood disaster early warning system using JSN-SR04 sensors and rainfall based on the internet of things, there are several supporting components used in the manufacture of Printed Circuit Boards (PCB). PCB is a printed circuit board that serves to connect every component such as capacitors, resistors, transistors, Integrated Circuits (IC) etc [9]. In this hardware design, there are 2 PCBs, namely the module sender and the receiver module PCB. You can see the name and a lot of components needed to design the sender module in Table I and the receiver module in Table II.

TABLE I. Sender Module Components.

No	Component	Lots
1	Resistor SMD 0805 470R, 10K, 1K, 30K, 7K5	10
2	Capacitor SMD 0805 22pF, 100nF	10
3	Elco 1uF/50V, 470uF/16V	3

5	Diode M7 SMD	3
6	Diode 1N4007 and schottky Diode 5A	1
7	Transistor BC547	1
8	Relay, Push Button, USB female type B connector	1
9	Led	2
10	Optocoupler PC817 and Fuse 5A/2A + Soket	1
11	IC 7809 7805, AMS 3.3V, Crystal 16 Mhz and 12Mhz	1
13	Pin Header, JST 3 pin, block terminal 3 pin & 2 pin	To taste
14	IC Atmega 328P DIP + Socket dan CH340G	1
15	Rainfall sensor and JSN-SR04 sensor	1
16	LoRa E32 433T20D	1

TABLE II. RECEIVER MODULE COMPONENTS.

No	Component	Lots
1	Resistor SMD 0805 470R, 10K and 1K	12
2	Capasitor SMD 0805 22pF, 100nF 470nF, 4.7uF, 10uF	3
3	Elco 470uF/16V and Capacitor 100nF	3
4	Transistor SOT23 BC817	2
5	IC CH340G, Crystal 12 Mhz, USB female type B connector, diode 1N4007 and diode schottky 5A	1
6	Diode M&, Block terminal 3 pin and 2 pin	3
7	Led 5mm	7
8	IC 7809, 7805, AMS 117 3.3V, optocoupler PC817, relay, Transistor BC547	1
9	Pin header female and male	To taste
10	ESP32, LoRa E32 433T20D, sirine 12VDC and OLED	1

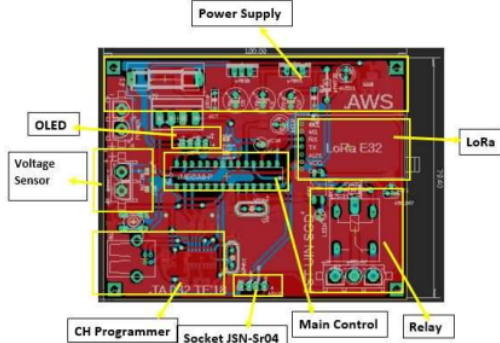Figure 3 is the overall PCB layout of the sender module and Figure 4 is the printed PCB.

Fig. 3. Pcb layout of sender module.

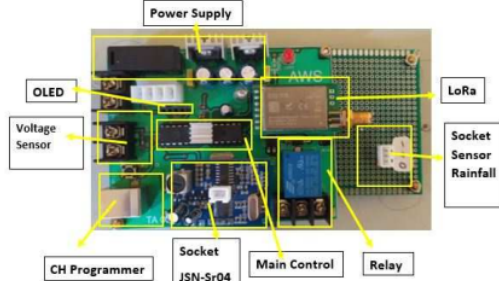Figure 3 is the PCB layout of the sender module designed using the EAGLE application.

Fig. 4. PCB sender module.

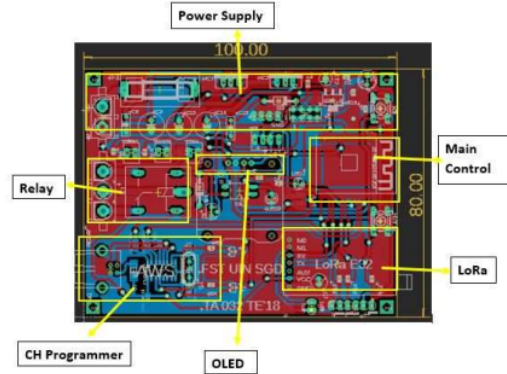Figure 4 is the sender module pcb that has been soldered and components installed.

Fig.5. Pcb layout of receiver module.

Figure 5 is the PCB layout of the receiver module designed using the EAGLE application.
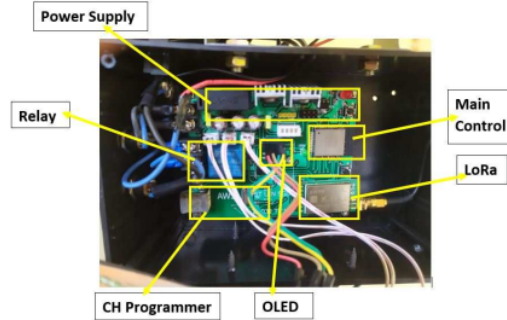

Fig. 6. PCB receiver module.

Figure 6 is a PCB that has been soldered and installed components.

Fig. 7. Sender module system and receiver modul system.

C. Software Design

In the design of the program using Arduino IDE software. This software uses C / C ++ language programming with several supporting libraries on the Arduino IDE. The creation of this program is used to run the sensors or components in the sending module according to the function of each sensor.
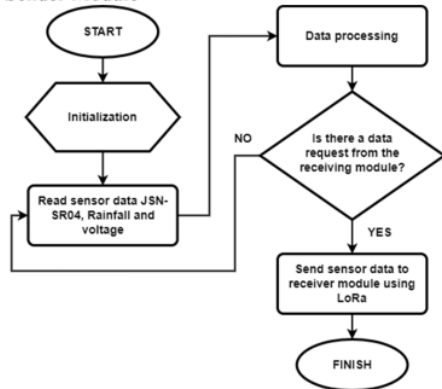
a. Sender Module

Fig. 8. Sender module flowchart.

Figure 8 is a flowchart or algorithm for the sender module. Figure 8 is an algorithm or illustration of the performance of the sending module system when it is turned on. The initial process starts by initializing the header – header, variable declaration, constants, functions, and connecting the Long Range (LoRa) sender module with the LoRa receiver module. Then the sensor will detect or read the distance, precipitation intensity, and battery voltage. After that, the sensor readings will be processed by the Atmega328P microcontroller so that it becomes data on the water level in centimeters (cm), rainfall intensity in millimeters (mm), and voltage in units (volts). Then the system will check whether there is a data request from the receiver module, if there is none then the system will loop from the sensor data reading process and will check the data request again from the receiver module. If there is a request from the receiving module, then the data will be sent to the receiving module using LoRa.

b. Receiver Module

Figure 9 is a flowchart or algorithm for the receiving module.

Figure 9 is an algorithm or illustration of the performance of the receiving module system when it is turned on. The initial process initializes the headers – headers, variable declarations, constants, functions – functions, connects the ESP32 to the WiFi network, and connects the LoRa receiver module with the LoRa sender module. Then this receiver module will make a data request to the sender module, and the data sent by the sender module will be checked and classified into flood levels. Water level data (K) in this rule is taken based on Ciyasana containment conditions, namely in Rancaekek Kulon, Rancaekek District, Bandung Regency, West Java, Indonesia, and rainfall (C) based on data from the Meteorology, Climatology, and Geophysics Agency (BMKG). When the water level is more than 180 cm, the flood level is at level 3 and the siren will be turned on 6 times. Variable C or precipitation is used only to monitor rainfall

intensity. Then the processed data will be sent to the firebase and Thingspeak databases with ESP32 connected to the WiFi network and will be displayed on the AWS android application.
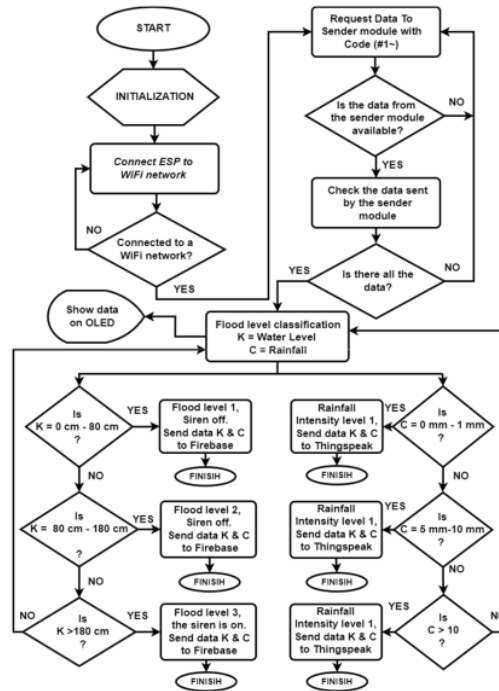

Fig. 9. Receiver module flowchart.

c. AWS Android Application

In designing this android application using the Android Studio. This AWS application is used to monitor water level, rainfall intensity and battery voltage. In this application, there is also a flood level in the form of messages. The app uses a firebase database to store temporary data for display on AWS applications. A view of these AWS applications can be seen in Figure 10.
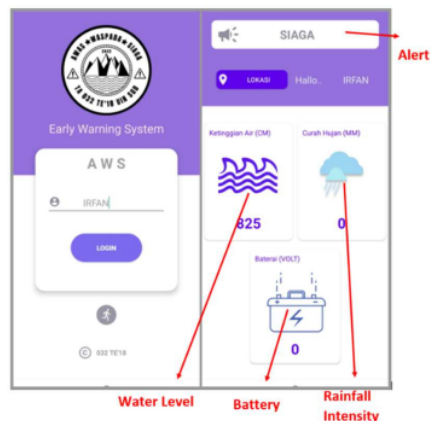

Fig. 10. AWS android app view.

III. Testing and Analysis

Sensor calibration is divided into two stages, namely the calibration of the JSN-SR04 sensor and the rainfall sensor. The calibration process is just done on the sender module where there is a JSN-SR04 sensor and rainfall.

a. Step 1. Calibration of the JSN-SR04 sensor

The first step of calibrating the JSN-SR04 sensor is to measure the distance with a meter measuring instrument and then compare it with the measurement results using the JSN-SR04 sensor. After taking 39 data, it was found that the average error ratio was 3.36%. A comparison graph of the measurement results between the meter measuring instrument and JSN-SR04 can be seen in Figure 11.
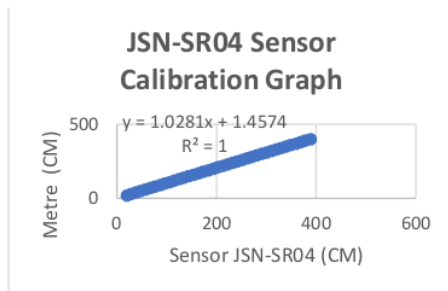

Fig. 11. JSN-SR04 sensor calibration graph.

From Figure 11, a linear regression equation is obtained from the comparison of measurements between the metre measuring instrument and the JSN-SR04 sensor, that is:

$$y = 1,0281x + 1,4574 \tag{1}$$

with a linearity value of R2 = 1. Equation 1 is entered into the code on the sending module microcontroller (Atmega328P) so that the JSN-SR04 sensor gets good measurement results. After equation 1 is inserted into the sending module microcontroller (Atmega328P) then data collection is carried out again, namely by measuring the distance using a meter measuring instrument and a JSN-SR04 sensor and then the results are compared again. After another comparison, the average result of the error ratio is getting smaller, which is 0.68%.

b. Step 2. Calibration of the rainfall sensor

In the process of calibrating the rainfall sensor using a comparison measuring instrument, namely a measuring cup. The first step is to measure the volume of water with a measuring cup and then record it. After that, it is inserted into the rainfall sensor and then recorded again the results. The calibration process of this rainfall sensor was carried out as many as 4 experiments, in 1 data experiment taken as many as 8 data. After the data retrieval process is complete, then the data is compared and calculated error ratio. After knowing the error ratio in each experiment, it is then averaged and the average error ratio results in 4 experiments are 11.14%. The error ratio in the rainfall sensor calibration process is large due to the limitations of the calibration tool. Figure 12 is the Rainfall sensor calibration process.

Fig. 12. Process rainfall sensor calibration.

TABLE III. Flood level Clasification Test In Lab

No	Water level measured (cm)	Flood Level	Sirine	Provision	
				Water Level (cm)	Level Flood
1	59	1	Off	0 – 120	1
2	122	2	Off	121 - 180	2
3	255	3	On	>180	3

TABLE IV. Clasification Rainfall Level In The Lab

No	Rainfall intensity measured (mm)	Level rainfall intensity	Provision	
			Rainfall intensity (mm)	Level ranfall intensity
1	0.18	1	≤ 1	1
2	1.32	2	$> 1 \, \& \leq 10$	2
3	10.42	3	>10	3

Tests for the classification of flood levels and rainfall intensity levels were carried out in the lab. The provisions for the Ciyasana water level can be seen in Table III and the provisions for rainfall levels are based on data from the BMKG as shown in Table IV. The provisions for the water level of Ciyasana can be seen in Table III and the provision of rainfall levels based on data from the BMKG as in Table IV.

After the sensor calibration process is complete then performs an overall test of the system. This test was carried out in the containment of Ciyasana Rancaekek Kulon, Rancaekek District, Bandung Regency, West Java, Indonesia. This testing process is to turn on the entire system from hardware to software. It can be seen in Figure 13 when installing sensors in the Ciyasana dam.

Fig. 13. Process installation sensor JSN-SR04.

After the JSN-SR04 sensor is installed, it then measures the distance from the sensor placement to the riverbed with a meter measuring instrument as shown in Figure 14. This measurement is carried out to ensure that the distance on the Ciyasana dam is correct.

Fig. 14. Distance measurement.

After taking measurements, it is known that the distance from the placement of the sensor to the riverbed or the base distance is 290 cm. Then this value is entered into the equation in the Arduino IDE code:

$$TMA = base\ distance - water\ level \quad (2)$$

The data obtained in testing the entire system is 130 data, which can be seen in Figure 15. Figure 15 is the data on flood levels and rainfall levels contained in accordance with the provisions in Table III and Table IV. The flood level line is inline gray with the orange rainfall level line which is at level 1 because the condition is contained in the presence of no rain and the water level condition is in the position of 6-8 cm.
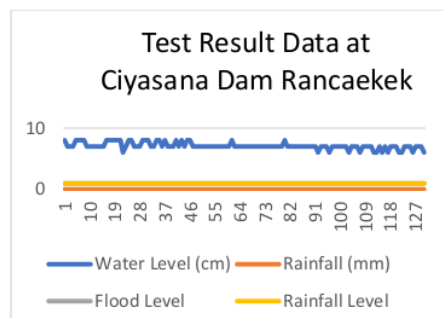

Fig. 15. Graph of Test results data at Ciyasana Rancaekek dam.

IV. CONCLUSION

In this research, the performance of this flood disaster warning system can detect water level and rainfall intensity. This system will provide flood and rainfall level information to the AWS application on the mobile phone sent by the receiver module connected to the WiFi network so that water level and rainfall conditions can be monitored anywhere using the mobile phone. In addition, the system can detect changes in water level and precipitation and will be classified into flood levels so that in addition to informing AWS applications, it can also provide flood level information in the form of siren sounds.

REFERENCES

[1] B. M. Shankar, T. J. John, S. Karthick, B. Pattanaik, M. Pattnaik, and S. Karthikeyan, "Internet of Things based Smart Flood forecasting and Early Warning System," *Proc. - 5th Int. Conf. Comput. Methodol. Commun. ICCMC 2021*, no. Iccmc, pp. 443–447, 2021, doi: 10.1109/ICCMC51019.2021.9418331.

[2] J. G. Natividad and J. M. Mendez, "Flood Monitoring and Early Warning System Using Ultrasonic Sensor," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 325, no. 1, 2018, doi: 10.1088/1757-899X/325/1/012020.

[3] T. Yanuarto, "Bertemunya Bencana Hidrometeorologi Basah dan Kering," 2021. https://www.bnpb.go.id/index.php/berita/review-bencana-juli-2021-bertemunya-bencana-hidrometeorologi-basah-dan-kering (accessed Oct. 15, 2021).

[4] W. Astuti and A. Fauzi, "Perancangan Deteksi Banjir Menggunakan Sensor Kapastif Mikrokontroler ATMega328p dan SMS Gateway," *J. Inform.*, vol. 5, no. 2, pp. 255–261, 2018, doi: 10.31311/ji.v5i2.3868.

[5] S. Ammari, W. Wildian, and H. Harmadi, "Rancang Bangun Sistem Peringatan Dini Banjir Berdasarkan Tingkat Kekeruhan Air Hulu Sungai dengan Turbidity Sensor SEN0189 dan Transceiver nRF24L01+," *J. Fis. Unand*, vol. 8, no. 3, pp. 240–244, 2019, doi: 10.25077/jfu.8.3.240-244.2019.

[6] C. K. Khuen and A. Zourmand, "Fuzzy Logic-Based Flood Detection System Using Lora Technology," *Proc. - 2020 16th IEEE Int. Colloq. Signal Process. its Appl. CSPA 2020*, no. Cspa, pp. 40–45, 2020, doi: 10.1109/CSPA48992.2020.9068698.

[7] M. J. Subashini, "Early Warning System using Weather Forecasting Data and Wireless Sensor Networks-A Review," no. Icicv, pp. 132–135, 2021.

[8] D. P. Jose and A. A. Thomas, "IoT Based Water Management using HC-12 and jango," *2019 Int. Conf. Data Sci. Commun.*, pp. 1–6, 2019.

[9] D. S. Rani, G. N. Jayalakshmi, and V. P. Baligar, "Low Cost IoT based Flood Monitoring System Using Machine Learning and Neural Networks: Flood Alerting and Rainfall Prediction," *2nd Int. Conf. Innov. Mech. Ind. Appl. ICIMIA 2020 - Conf. Proc.*, no. Icimia, pp. 261–267, 2020, doi: 10.1109/ICIMIA48430.2020.9074928.

[10] Valentina Sophia Manullang, "Modifikasi Penakar Hujan Otomatis Tipe Tipping Bucket Dengan Hall Effect Sensor Ats276," *J. SPEKTRUM*, p. 7, 2019, [Online]. Available: https://media.neliti.com/media/publications/221419-modifikasi-penakar-hujan-otomatis-tipe-t.pdf

[11] M. H. Kurniawan, S. Siswanto, and S. Sutarti, "Rancang Bangun Sistem Keamanan Sepeda Motor Dengan Sidik Jari Dan Notifikasi Panggilan Telepon Berbasis Atmega 328," *PROSISKO J. Pengemb. Ris. dan Obs. Sist. Komput.*, vol. 6, no. 2, pp. 152–165, 2019.

[12] EBYTE, "Chengdu Ebyte Electronic Technology Co., Ltd." https://www.ebyte.com/en/product-view-news.html?id=130 (accessed Nov. 20, 2021).

[13] N. A. Azmi Ali and N. A. Abdul Latiff, "Environmental Monitoring System Based on LoRa Technology in Island," *Proc. - 2019 IEEE Int. Conf. Signals Syst. ICSigSys 2019*, pp. 160–166, 2019, doi: 10.1109/ICSIGSYS.2019.8811066.

[14] M. Babiuch, P. Foltynek, and P. Smutny, "Using the ESP32 microcontroller for data processing," *Proc. 2019 20th Int. Carpathian Control Conf. ICCC 2019*, pp. 1–6, 2019, doi: 10.1109/CarpathianCC.2019.8765944.

[15] W. Li, C. Yen, Y. Lin, S. Tung, and S. Huang, "JustIoT Internet of Things based on the Firebase Real-time Database," *Proc. - 2019 IEEE Int. Conf. Smart Manuf. Ind. Logist. Eng. SMILE 2019*, pp. 43–47, 2019.