Article

Soil moisture sensor based on Internet of Things LoRa

Puput Dani Prasetyo Adi [1*], Victor Marudut Mulia Siregar [2] , Akio Kitagawa [3]

[1] Department of Electrical Engineering, University of Merdeka Malang; puput.danny@unmer.ac.id;
[2] Department of Computer Engineering, Politeknik Bisnis Indonesia; victor.siregar2@gmail.com;
[3] Micro Electronics Research Laboratory (MeRL), Kanazawa University; kitagawa@merl.jp

* Corresponding author: puput.danny@unmer.ac.id

Abstract: This study discusses the Performance of Moisture soil sensor, which is helpful as a sensor to detect soil moisture levels used as plant nutrition. The monitoring process is carried out in real-time using internet of things technology based on LoRa or Long-Range Radio Frequency (IoT-LoRa). The application server used is Thingspeak. With this research, agricultural processes and monitoring can be carried out dynamically and efficiently. Therefore, when the soil conditions are dry, or the soil moisture level is <300, it will immediately affect automatic watering by opening the valve or rotating the Servo motor. Furthermore, the watering process can be done automatically by looking at the soil conditions on potted plants or agricultural land. The position of the sensor on the ground level is not immediately removed and moved to another place, but it is always in the same condition and location, so the Adaptive Data Rate mechanism is used for the management of Power Consumption on IoT-LoRa

Keywords: moisture, soil, monitoring, IoT, plants, LoRa

Citation: P.D.P.Adi, Siregar.V.M.M, Soil moisture sensor based on Internet of Things LoRa. *Iota*, **2021**, ISSN 2774-4353, Vol.01, 02. https://doi.org/10.31763/iota.v1i2.495

Academic Editor : Pranolo.A

Received : 15 Maret 2021

Accepted : 10 April 2021

Published : 12 May 2021

1. Introduction

Soil humidity is an essential factor that determines plant fertility, except for fertilizers and other parameters, current trends, plants are not planted on large farms, now switching to hydroponic types of plants that depend on the quality of water used, plus other factors such as humidity, lighting, and the right room temperature. Therefore, not all types of plants use hydroponics. Types of plants that can be grown hydroponically include, e.g., lettuce, tomatoes, radishes, kale, cucumbers, spinach, nuts, leeks, and basil strawberries, mint leaves, blueberries, peppers, carrots, and Corn.

Therefore, what is the treatment for plants that cannot use hydroponics? Plants that can't be cultivated using hydroponics need to use soil on agricultural land or be placed in pots. For plants grown using soil, it is necessary to use a system that can detect soil conditions and how the following action is to a soil that is too wet, less humid, or dry. These factors are essential to determine the quality of the resulting plant, even the plant's life; if the soil is too dry, the plant will likely wither and die.

LoRa 915 MHz [15] and 920 MHz [13] testing has been tested in previous studies [1] using Dragino LoRa and BME280 Temperature and Humidity sensors at Kanazawa University Japan. Different RSSI values were obtained in this study, using the Adaptive Data Rate (ADR) [14] mechanism as energy efficiency or battery savings for the LoRa sensor node. From research [1] it was obtained the RSSI value from 1 meter to 1 km. At 1 meter, RSSI shows a value of -30 dBm, and at a distance of 1 km around -130 dBm, there are three classifications in the RSSI Analysis, i.e., Free Space, Building, and Hill and Tree Obstacles. The development of data analysis is at the level of comparison of Spreading Factor [16], [18], Bandwidth, Time on Air, and other analyzers on LoRa and LoRaWAN.

2. Literature Review

Figure 1 is Moisture Soil Sensor v.1.4 which is used as the main and essential sensor in this research, and figure 2 is a Moisture soil sensor and LoRa IoT working system. Furthermore, Table 1 is the Moisture Soil Sensor Specifications; the minimum state or Voltage is 3.3. Volt DC and the maximum is 5 Volt DC, and you should use a voltmeter to find out the specific Voltage used [22]. And the current (mA) is 0 ~ 35 mA. Three soil states are classified on this moisture sensor, i.e., in dry soil sensor (0-300), in humid soil sensor (300-700), and in water sensor (700-950) [1], [19].

Accordingly, the specifications for Moisture Soil Sensor v.1.4 are shown in table 1. However, these specifications are not entirely the same, they can be configured, and even the values can be changed.

The stages of the IoT-LoRa Moisture configuration are as follows:

1. Complete Programming Moisture Soil sensor and Arduino.
2. Complete the Moisture Soil Sensor [2],[3], Arduino, and LoRa 915 MHz Programming.
3. Communication test LoRa 915 MHz transmitter and LoRa 915 MHz receiver, point to point. [Figure 3], [Figure 6]
4. Test Communication LoRa 915 MHz transmitter and LoRa 915 MHz receiver, point to multi-point. [Figure 6]
5. Test the uplink data of LoRa 915 MHz transmitter to LoRa Gateway 915 MHz
6. Analysis by the SDR LoRa Signal Analyzer
7. Test and configure the Thingspeak Server application

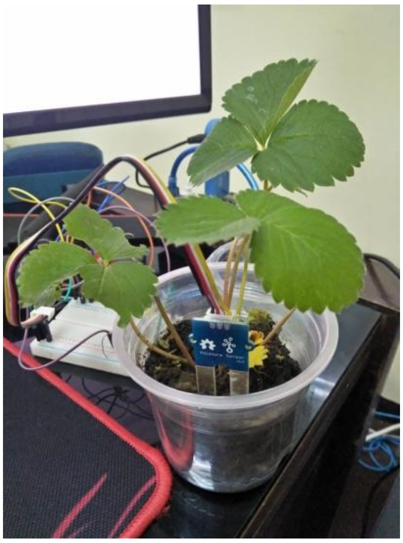

Figure 1. Moisture Soil Sensor v.1.4 on Strawberry Plant [Puput_data]

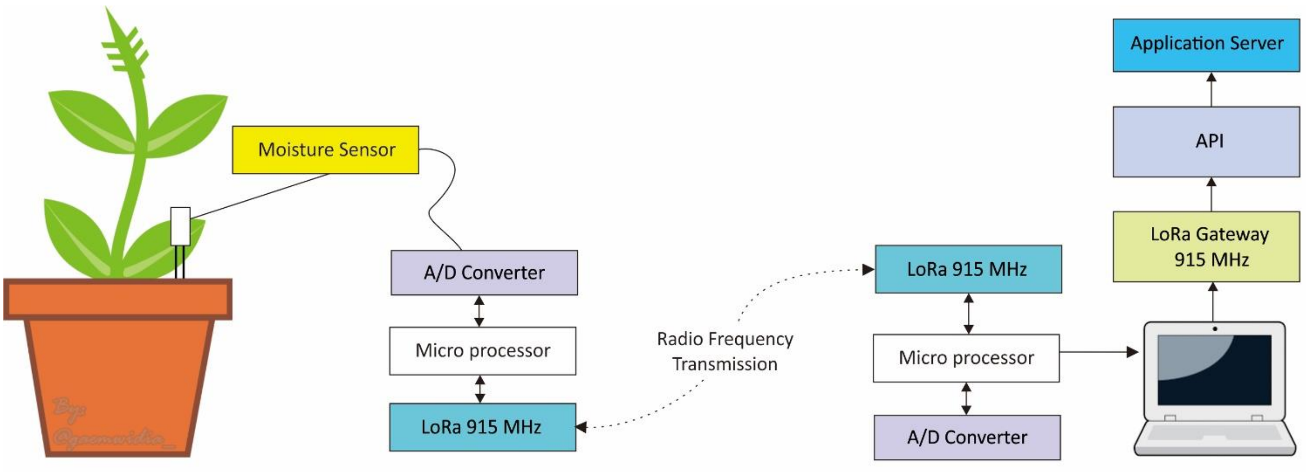

Figure 2. Moisture soil sensor and LoRa IoT working system

TABLE I. MOISTURE SOIL SENSOR SPECIFICATION

Item	Condition	Min	Typical	Max	Unit
Voltage	-	3.3	/	5	V
Current	-	0	/	35	mA
	A sensor in dry soil	0	~	300	/
Output Value	A sensor in humid soil	300	~	700	/
	Sensor in water	700	~	950	/

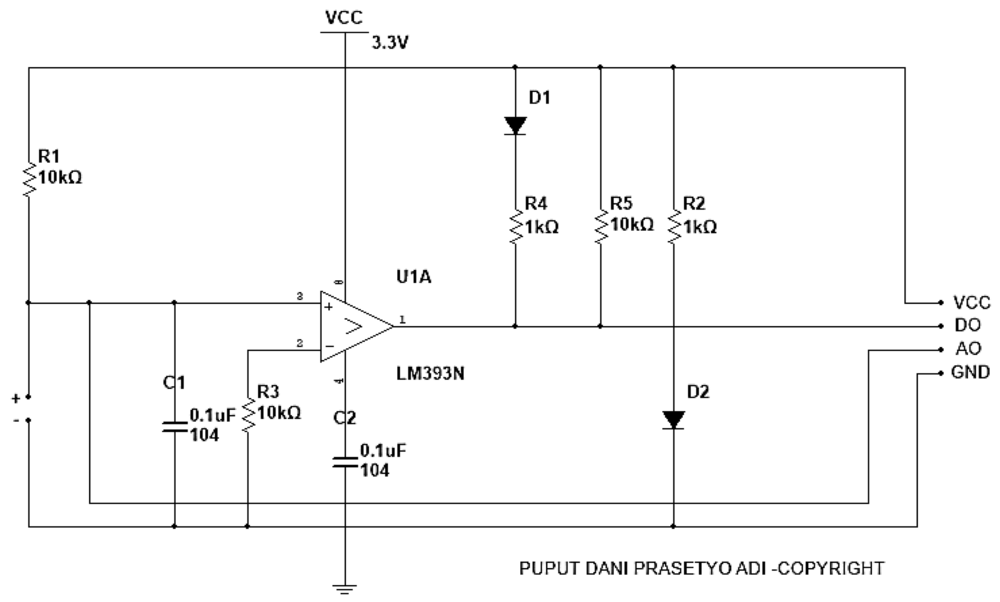

Figure 3. Moisture Sensor pins

Schematic Moisture sensor [7],[8] can be seen in Figure 2 [4], for further testing is carried out using three conditions, outside the soil, in the soil (strawberry plant pots), and water. Arduino schematic and Moisture sensor [5],[6] are shown in Figure 3. Figure 2 shows a system or architecture built using LoRa. The LoRa used is a 915 MHz Dragino LoRa type.

An algorithm needs to help power or battery efficiency; an ADR (Adaptive Data Rate) mechanism is used. This mechanism only works when the end node is static or not moving. Schematic Moisture sensor can be seen in Figure 3, for further testing is carried out using three conditions, outside the soil, in the soil (strawberry plant pots), and in water. Arduino schematic and Moisture sensor are shown in Figure 3. Figure 8 shows a system or architecture built using LoRa. The LoRa used is a 915 MHz Dragino LoRa type. An algorithm needs to help power or battery efficiency; an ADR (Adaptive Data Rate) mechanism is used. This mechanism only works when the end node is static or not moving.

LoRa is one of the LPWAN (Low Power Wide Area Network) devices, a specialist on small data rates (0.3 kbps - 5.5 kbps). Based on Range vs. Power LoRa, it is shown in table 2. Furthermore, table 3 is the ISM Band LoRa; this sets the Frequency for each region or country when using the LoRa module. But in this case, LoRa is a Free License; therefore, it can be used freely by users to develop.

TABLE II. RANGE VS. POWER WSNS

Technology	Wireless Communication	Range	Tx Power
Bluetooth	Short Range	10 m	2.5 mW
WiFi	Short Range	50 m	80 mW
3G/ 4G	Cellular	5 km	5000 mW
LoRa	LPWAN	2-5 km (urban) 5-15 km (rural) >15 km (LoS)	20 mW

TABLE III. ISM BAND LORA

Region	Frequency (MHz)
Asia	433
Europe, Russia, India, Africa (Part)	863-870
US	902-928
Australia	915-928
Canada	779-787
China	779-787, 470-510

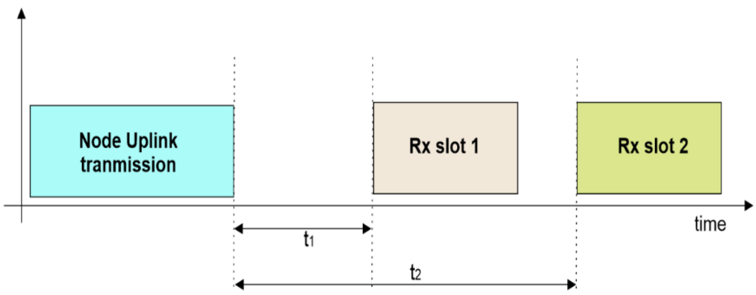

Figure 4. LoRa Class A

LoRaWAN has 3 specifications for Class devices [9]. There is a tree class ie, All (A), Beacon (B), and Continues (C), e.g., Class A in Figure 4. Class A, All battery-powered devices [11], each device uplink to the gateway and is followed by two short downlinks receive windows. B, Beacon same as class A but these devices also open extra receive windows at the scheduled time; C, Continues same as A; however, these devices are continuously listening. Hence these devices use more power and are often mainly powered.

RSSI stands for Received Signal Strength Indicator; RSSI uses dBm units. The RSSI value is getting closer to 0 means that the greater the LoRa signal produced or detected, which means that the distance between Tx and Rx is very close. Moreover, Figure 5 is a LoRa transmitter, and Figure 6 is a LoRa receiver.

$$P(d) = P_0 - 10\, n_p \log d/d_0 \quad (1)$$

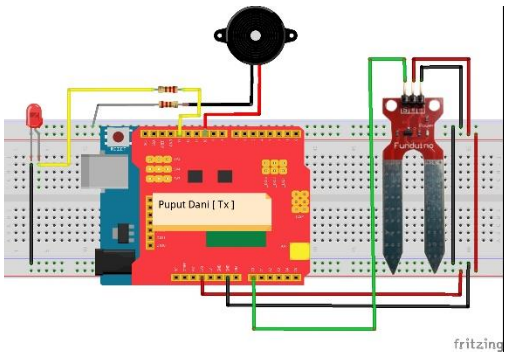

Figure 5. Moisture, Arduino, and Dragino LoRa 915 MHz [Tx]

A negative value represents RSSI. e.g., -30 dBm is a strong signal, while -120 dBm is a weak signal. In the previous study [13, 17, 22], LoRa RSSI was defined in equation 1.

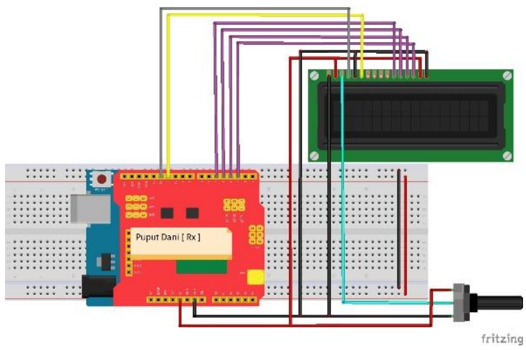

Figure 6. Arduino, LCD, and Dragino LoRa 915 MHz [Rx]

Pseudocode 1 is a program sequence on a moisture sensor, the core of the program is on the Analog sensor (A0) from the Moisture sensor, from this stage programming Arduino and LoRa 915 MHz follow it, then configuring Arduino, LoRa 915 MHz, and LoRa Gateway [Figure 7, 8]. As shown at the stages of the IoT-LoRa Moisture configuration, the next step is how to make Point-to-point communication using LoRa 915 MHz to get RSSI data, *Pseudocode 1* is basic C ++ programming on Arduino to obtain the value of

Moisture soil sensor, furthermore, how to send the sensor data by Dragino LoRa 915 MHz and obtain the RSSI data. *Pseudocode 2* is how to transfer data from Tx to Rx using Dragino LoRa 915 MHz. Furthermore, *Pseudocode 2* is a program from the Arduino and LoRa Library to transmit Moisture sensor data.

Figure 7. LG01-P Single Channel LoRa IoT Gateway [Puput_data]

```
1. Integer Parameters Initialize and Arduino Pin Address
   sensorPin = A0;
   sensorValue;
   buzzer=10;
2. Point limitation Moisture sensor
   limit = 300;
3. Initialize of Boudrate
   Serial.begin(9600);
4. Initialize of Pin OUTPUT (LED)& Buzzer
    pinMode(13, OUTPUT);
    pinMode(buzzer,OUTPUT);
5. Initialize of analogRead (A0)
    sensorValue = analogRead(sensorPin);
6. Output text on the Serial monitor from the Moisture Sensor
   AnalogRead
    Serial.println(" ");
    Serial.println(sensorValue);
7. Limitation decision from Moisture sensor and Condition the
   output LED & Buzzer
    if (sensorValue<limit) {
    digitalWrite(13, HIGH);
    tone(buzzer, 1000, 200);}
    else { digitalWrite(13, LOW);
    noTone(buzzer);}
8. Give the Delay from Sensor AnalogRead
     delay(1000);
```

===== *Pseudocode 1* =====

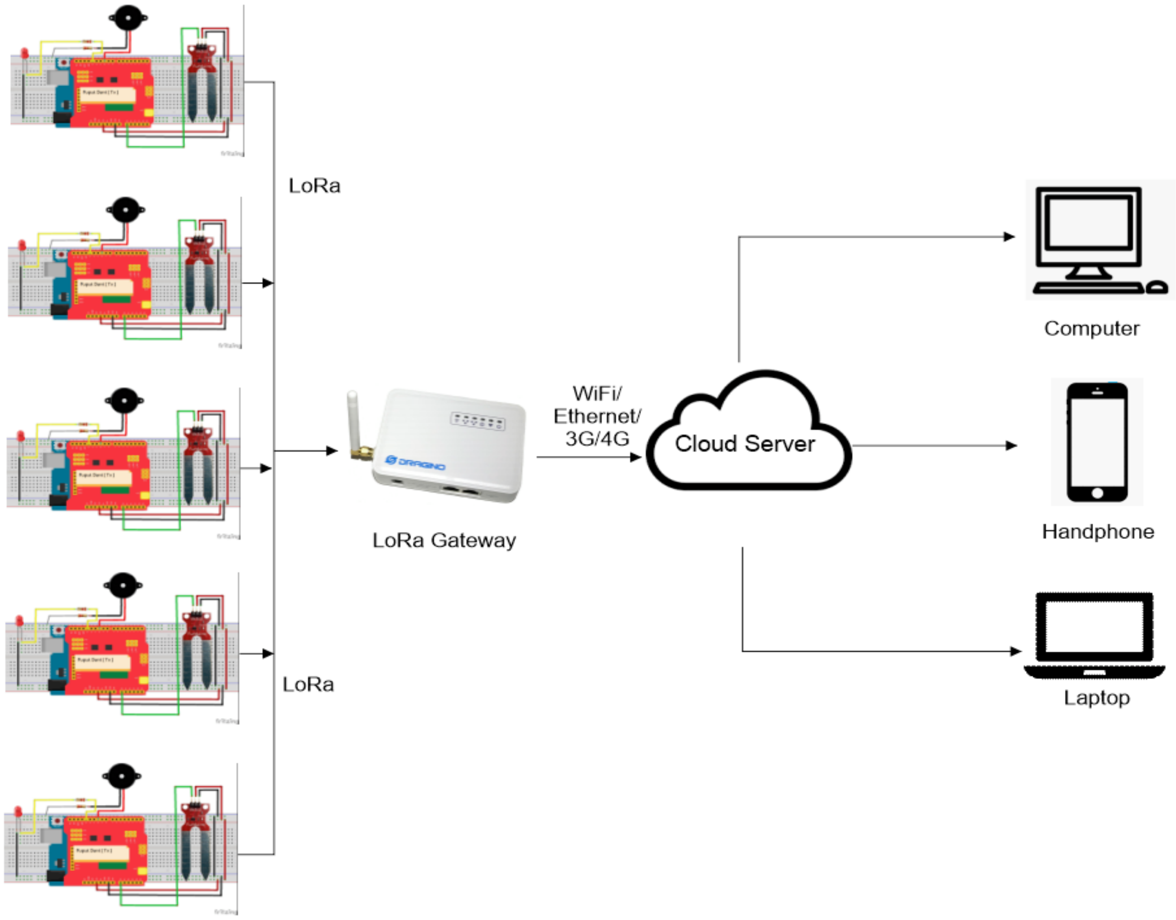

Figure 8. a Moisture sensor IoT-LoRa Architecture

```
1. Initialize of Library
   #include <RH_RF95.h>
   #include <ThingSpeak.h>
   #include <SPI.h>
   #include <LoRa.h>
2. Initialize of Moisture Sensor
   int sensorPin = A0;
   int sensorValue;
   int limit = 300;
   int buzzer=10;
3. Initialize of Serial Boudrate
   Serial.begin(9600);
4. Pin Output Mode
   pinMode(13, OUTPUT);
    pinMode(buzzer,OUTPUT);
5. LoRa Frequency initialize
   while(!Serial);
```

```
  Serial.println("LoRa Sender :");
  (!LoRa.begin(915E6)) // 915MHz Freq
  Serial.println("Starting LoRa Failed !");
6. Analog Read and Output Process
  sensorValue = analogRead(sensorPin);
  if (sensorValue<limit)
  digitalWrite(13, HIGH);
  tone(buzzer, 1000, 200);
  else
  digitalWrite(13, LOW);
  noTone(buzzer);
7. Delay of Data Moisture
  delay(1000);
8. Serial Print Output
  Serial.print("Moisture.Sensor_Value: ");
  Serial.print(sensorValue);
  Serial.println();
9. Command of LoRa Transmitter
  LoRa.beginPacket();
  LoRa.print("Moisture.Sensor_Value: ");
  LoRa.print(sensorValue);
  LoRa.endPacket();
  delay(0);}
```

===== ***Pseudocode 2*****=====**

```
1. Initialize of LoRa Library
   #include <SPI.h>
   #include <LoRa.h>
2. Initialize of Serial Boudrate
    Serial.begin(9600);
    while (!Serial);
    Serial.println("LoRa Receiver");
3. Initialize of LoRa Frequency
    if (!LoRa.begin(915E6)) {
    Serial.println("Starting LoRa failed!");
    while (1);
4. LoRa Packet Initialize
    int packetSize = LoRa.parsePacket();
    if (packetSize)
5. Received a packet
    Serial.print("Received packet '");
```

```
6. Read packet
    while (LoRa.available()) {
      Serial.print((char)LoRa.read());
7. Print packet RSSI
    Serial.print("' with RSSI ");
    Serial.println(LoRa.packetRssi());

                    ======= Pseudocode 3 ======
```

3. Methods

Adaptive Data Rate (ADR) [10],[12] is one of the methods used to maintain LoRa throughput (kbps) and minimize the possibility of Packet Loss (kbps). The algorithm. There are 3 ADR processes [20], i.e., the Uplink, Downlink, and ADR Response. When transmitting LoRa data, if ADR is activated, it will immediately adjust the ADR Schedule and LoRa data bits until the data transmitting process is complete on the thingspeak Figure 10 and Figure 11.

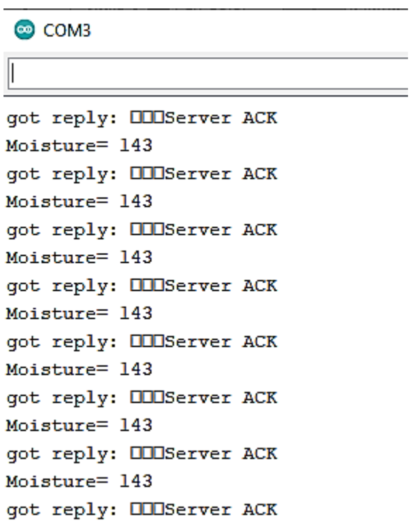

Figure 9. ACK Reply

Figure 9 shows the ACK Reply showing that the communication from LoRa to LoRa Gateway and API Application Server was a success. Henceforth, this Moisture sensor data is displayed on the Thingspeak Application server [21].

4. Result and Discussion

In Experiment 1, the Moisture Sensor is placed in water, and the average value is shown as in table 2, i.e., 396 to 417, and if the Moisture sensor is placed outside the pot, it shows a value of 0 to 5, more details are shown in figure 10. in Figure 10, the sensor output shows insignificant values based on the area or area selected, e.g., outside the pot and in the water, and 1 meter RSSI shows a value of -109 to -107 dBm.

Pseudocode 4 is used to set the Dragino LoRa 915 MHz Gateway or LG01-P Single Channel LoRa IoT Gateway; furthermore, Moisture sensor data will be sent via the Application Server Thingspeak Mathworks.

```
unsigned long myChannelNumber = 1348794;
const char * myWriteAPIKey = "193TS6NE36S1QINU";
uint16_t crcdata = 0;
uint16_t recCRCData = 0;
float frequency = 915;
  Bridge.begin(BAUDRATE);
  ThingSpeak.begin(client);
  if (!rf95.init());
  rf95.setFrequency(frequency);
  // Setup Power,dBm
  rf95.setTxPower(13);
  rf95.setCADTimeout((unsigned long)5);

              ===== Pseudocode 4 ====
```

Uplink, Downlink, and ADR Response. When transmitting LoRa data, if ADR is activated, it will immediately adjust the ADR Schedule and LoRa data bits until the data transmitting process is complete on the thingspeak Figure 11 and Figure 12.

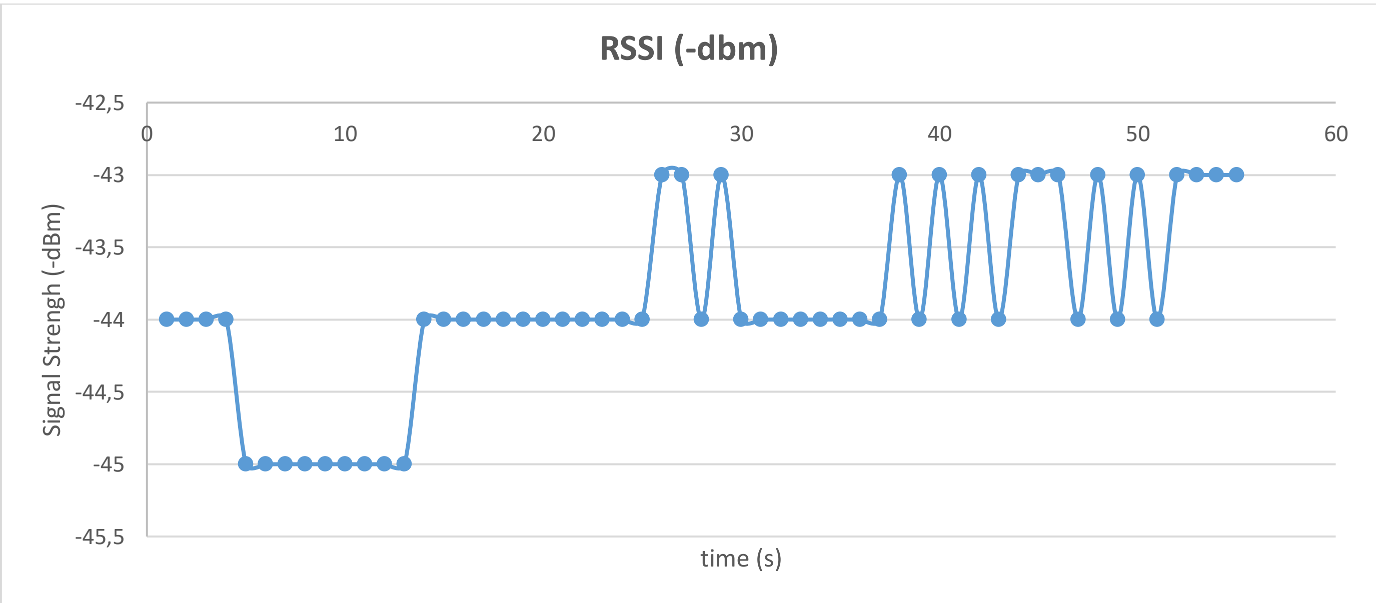

Figure 10. RSSI (-dBm) on 1 meter Tx and Rx LoRa

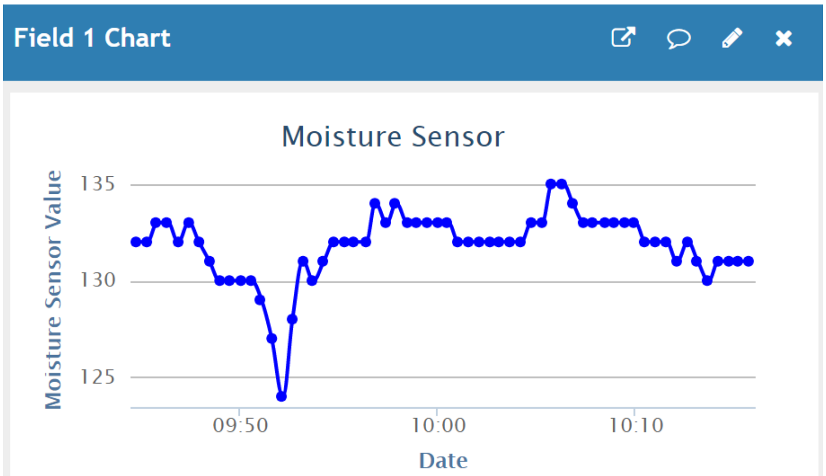

Figure 10. real-time data moisture sensor in IoT

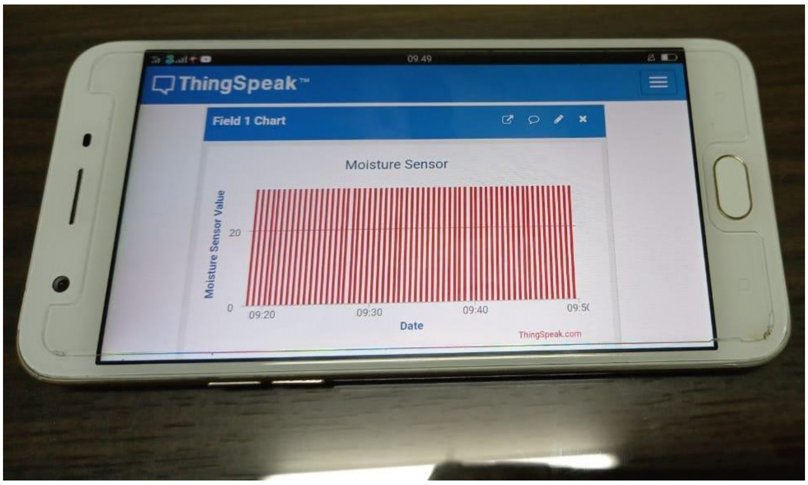

Figure 11. Moisture sensor LoRa 915 and Thingspeak Based on Smartphone

5. Conclusions and Suggestion

The IoT based moisture sensor has been successfully displayed on IoT using the Mathworks MATLAB Application Server through the Application Programming Interface (API), analysis is carried out on LoRa Receive Signal Strength at the gateway position. RSSI (-dBm) shows a value of -103 dBm to -104 dBm at a distance of 1 meter, and this will experience continuous attenuation in accordance with the obstacles obtained from the Transmitter to Receiver or LoRa Gateway LG01-P. The more end device or moisture soil sensors, the bigger the data packet received by the Gateway LG01-P which causes greater packet loss of data.

The development of Smart Agriculture from an IoT based moisture soil sensor system is the development of the number of End nodes if 100 End nodes moisture sensors are connected to the Thingspeak Application Server, and the analysis carried out is on the Power Consumption factor and data collisions on the Moisture sensor data transmission from the End Node sensor. To the Application Server, e.g., Uplink and Downlink data.

Author Contributions: Conceptualization; Puput Dani Prasetyo Adi (P.D.P Adi),Victor M.M.Siregar (V.M.M.S), methodology; P.D.P Adi, software; P.D.P Adi, V.M.M.S, validation; P.D.P Adi, formal analysis; P.D.P Adi, investigation; P.D.P Adi, data curation; P.D.P Adi; writing—original draft preparation; P.D.P Adi, writing—review and editing; P.D.P Adi, visualization; P.D.P Adi, supervision; P.D.P Adi, project administration; P.D.P Adi, S.V.M.M, funding acquisition; P.D.P Adi, V.M.M.S, have read and agreed to the published version of the manuscript.

Acknowledgments: Thanks to all those who have helped complete this research, especially for Professor Akio Kitagawa. Thanks to Mathworks for providing the Free Version on the Thingspeak Application Server so that sensor data can be displayed on the Thingspeak Channel Application Programming Interface.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Puput Dani Prasetyo Adi, Akio Kitagawa, "A performance of radio frequency and signal strength of LoRa with BME280 sensor", Telkomnika Vol 18, No 2, 2020, DOI:10.12928/telkomnika.v18i2.14843
2. Mochamad Firman Ghazalia, Ketut Wikantikaa, Agung Budi Hartoa,Akihiko Kondoh" Generating soil salinity, soil moisture, soil pH fromsatellite imagery and its analysis", Information Processing in Agriculture, elsevier, DOI : 10.1016/j.inpa.2019.08.003
3. J.R.R.N. Jayarathnea, T.K.K. Chamindu Deepagoda, Timothy J. Cloughb, Steve Thomas, Bo Elberling, Kathleen M. Smits" Effect of aggregate size distribution on soil moisture, soil-gas diffusivity, and N2O emissions from a pasture soil", Geoderma, elsevier, October 2020, DOI: 10.1016/j.geoderma.2020.114737
4. Jori Uusitalo, Jari Ala-Ilomäki, Harri Lindeman, Jenny Toivio and Matti Siren, "Modelling soil moisture – soil strength relationship of fine-grained upland forest soils", Silva Fennica vol. 53 no. 1 article id 10050, 8 March 2019, https://doi.org/10.14214/sf.10050
5. Marcelo R.Romero, Alexis Wolfel, Cecilia I. Alvarez Igarzabal, "Smart valve: Polymer actuator to moisture soil control", Sensors and Actuators B: Chemical, Volume 234, 29 October 2016, Pages 53-62, DOI: 10.1016/j.snb.2016.04.104
6. Stuart Batterman, Adarsh Kulshrestha, Hsiang-Yin. Cheng, "Hydrocarbon Vapor Transport in Low Moisture Soils", Environmental Science and Technology 29(1):171-80, PubMed, January 1995, DOI: 10.1021/es00001a022
7. Hülya Bakır, Salih Evren, Hikmet BIRHAN, Taşkın Öztaş, "Effects of Mulching and Snow Cover on Soil Moisture, Soil Temperature and Soil Losses", january 2016, Journal of Advanced Agricultural Technologies 3(4):305-309, DOI: 10.18178/joaat.3.4.305-309
8. Manuel Ricardo Pérez Cerquera, Diego MendezDiego Mendez, "Microstrip Time Domain Transmission (TDT) Application for Moisture Soil Measurements", Conference: 2018 IEEE International Symposium on Antennas and Propagation and USNC-URSI Radio Science Meeting At: Boston, Massachusetts, October 2018, DOI: 10.1109/APUSNCURSINRSM.2018.8609400
9. P. D. P. Adi and A. Kitagawa, "A Study of LoRa Performance in Monitoring of Patient's SPO2 and Heart Rate based IoT," Int. J. Adv. Comput. Sci. Appl., vol. 11, no. 2, 2020, doi: 10.14569/IJACSA.2020.0110232.
10. Norhane Benkahla, Hajer Tounsi, Ye-Qiong Song, Mounir Frikha, "Review and experimental evaluation of ADR enhancements for LoRaWAN networks", January 2021, Springer, Telecommunication Systems, DOI: 10.1007/s11235-020-00738-x
11. P. D. P. Adi and A. Kitagawa, "Quality of Service and power consumption optimization on the IEEE 802.15.4 pulse sensor node based on Internet of Things," Int. J. Adv. Comput. Sci. Appl., 2019, [Online]. Available: http://www.scopus.com/inward/record.url?eid=2-s2.0-85066760495&partnerID=MN8TOARS.
12. Kunho Park, Junhyun Park, Junhyun Park, A-Hyun Lee, Chong-kwon Kim, "An Energy Efficient ADR Mechanism Considering Collision Rate in LoRa Network", December 2020, KIISE Transactions on Computing Practices 26(12):535-540, DOI: 10.5626/KTCP.2020.26.12.535

13. P. D. P. Adi and A. Kitagawa, "Performance Evaluation of Low Power Wide Area (LPWA) LoRa 920 MHz Sensor Node to Medical Monitoring IoT Based," 2020 10th Electr. Power, Electron. Commun. Control. Informatics Semin., pp. 278–283, Aug. 2020, doi: 10.1109/EECCIS49483.2020.9263418.
14. Derek Heeger, et.al, "Secure LoRa Firmware Update with Adaptive Data Rate Techniques", March 2021, Sensors 21(7):2384, DOI: 10.3390/s21072384
15. Puput Dani Prasetyo Adi and Akio Kitagawa, "Performance Evaluation of E32 Long Range Radio Frequency 915 MHz based on Internet of Things and Micro Sensors Data" International Journal of Advanced Computer Science and Applications(IJACSA), 10(11), 2019. http://dx.doi.org/10.14569/IJACSA.2019.0101106
16. Jaber Babaki, et.al, "Dynamic Spreading Factor and Power Allocation of LoRa Networks for Dense IoT Deployments", Conference: 2020 IEEE 31st Annual International Symposium on Personal, Indoor and Mobile Radio Communications, August 2020, DOI: 10.1109/PIMRC48278.2020.9217283
17. Puput Dani Prasetyo Adi and Akio Kitagawa, "ZigBee Radio Frequency (RF) Performance on Raspberry Pi 3 for Internet of Things (IoT) based Blood Pressure Sensors Monitoring" International Journal of Advanced Computer Science and Applications(IJACSA), 10(5), 2019. http://dx.doi.org/10.14569/IJACSA.2019.0100504
18. Smangaliso Mnguni, "Performance Evaluation of Spreading Factors in LoRa Networks", In book: Towards new e-Infrastructure and e-Services for Developing Countries, 12th EAI International Conference, AFRICOMM 2020, Ebène City, Mauritius, December 2-4, 2020, Proceedings, DOI:10.1007/978-3-030-70572-5_13
19. Laura García, Lorena Parra, et.al., "Deployment Strategies of Soil Monitoring WSN for Precision Agriculture Irrigation Scheduling in Rural Areas", Sensors 2021, 21, 1693. https://doi.org/10.3390/s21051693
20. Rachel Kufakunesu, et.al., "A Survey on Adaptive Data Rate Optimization in LoRaWAN: Recent Solutions and Major Challenges", Sensors 2020, 20, 5044; doi:10.3390/s20185044
21. Puput Dani Prasetyo Adi, Akio Kitagawa, Junichi Akita, "Finger Robotic control use M5Stack board and MQTT Protocol based", 2020 7th International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE), DOI: 10.1109/ICITACEE50144.2020.9239170, 29 October 2020
22. Puput Dani Prasetyo Adi, et.al, "A Performance Evaluation of ZigBee Mesh Communication on the Internet of Things (IoT)", 2021 3rd East Indonesia Conference on Computer and Information Technology (EIConCIT), DOI: 10.1109/EIConCIT50028.2021.9431875, 17 May 2021
23. P D P Adi, et.al, "A Study of Programmable System on Chip (PSoC) Technology for Engineering Education", Journal of Physics: Conference Series, Volume 1899, 2nd Workshop on Engineering, Education, Applied Sciences and Technology (WEAST) 2020 5 October 2020, Makassar, Indonesia