Prototype for Forest Fire Early Detection System Using the MQTT Method

Abdullah Sani[1*], Frisca Tryandayani Simanjuntak[1], Hana Mutialif Maulidiah[1], Widya Rika Puspita[1], dan B. Budiana[1]

[1]Jurusan Teknik Elektro, Politeknik Negeri Batam, Batam, Indonesia

*Email: sani@polibatam.ac.id

***Abstract*—The case of forest fires in Batam is one of the natural disasters that often occurs both caused by nature and human error. These forest fires certainly damage forest ecosystems. The condition of the vast forest makes it difficult to monitor the condition of the forest to find the location of the fire. In this study, the authors designed a forest fire early detection system using the concept of Wireless Sensor Network. The system is designed using the MQTT protocol in its data transmission communications. MQTT is a communication protocol that uses the concept of publish-subscribe. Fire information sent by telegram. It is hoped that with this research, it will be able to detect early fires that occur in the forest and be able to become an initial concept for further development. From the test results, it was found that the MQTT protocol sends data well, with subscribers using QoS 0 services having an average delay of 0.050 s, an average throughput of 31355 bps and packet loss of 0%. Subscribers who use QoS 1 services have an average delay of 0.064 s, an average throughput of 30683 bps, and a packet loss of 0%. The system is able to send information notifications to telegram well.**

Keywords: Forest Fires, MQTT, Sensor Node, and Wireless Sensor Network

I. INTRODUCTION

INDONESIA is one of the countries known as the lungs of the world because Indonesia has the 2nd largest amount of forest in the world. This country has a big role in supplying oxygen in the world. The total area of forest in Indonesia that was recorded in 2020 was 125.9 million hectares which has decreased every year due to several factors, one of which is forest fires[1]– [3] . Forest fires that occur in Indonesia are caused by climate change and human activities. One of the areas where forest fires frequently occur is Batam City. This is certainly detrimental to Indonesia, not only the reduction in forest area, but also the destruction of forest ecosystems and surrounding air pollution.

Air pollution caused by forest fires causes the surrounding community to inhale the residual smoke of the fires which can certainly be harmful to health [4], [5]. Therefore, efforts are needed to tackle forest fires. One of the efforts that can be done is to carry out surveillance in the Batam City Forest and be responsive in taking action when a forest fire occurs so that this fire does not spread quickly.

With the extent of forest in Batam City, it is certainly very difficult to monitor forest fire location points. Increasingly developing technologies can be implemented in this case, one of which is utilizing the Wireless Sensor Network concept, such as early detection of forest fires. Wireless Sensor Network is a wireless sensor network technology that functions to monitor and collect information about environmental conditions and then send that information to users [6]– [9]. This system has been made with different performance and different methods. In previous research, researchers designed a prototype Wireless Sensor Network (WSN) for a forest fire system using the LoRa communication method [10], [11].

In 2021, researchers implemented wireless sensors to determine the location of forest fires using the Fuzzy logic method [12], [13]. Another research is the design and implementation of WSN in an IoT-based forest fire monitoring system using the Zigbee communication method [14]– [16]. In this study, a different method was used from previous studies, namely using the Message Queue Telemetry Transport (MQTT) communication method. MQTT is a communication protocol that implements publish and subscribe data [17], [18]

In publish/subscribe there are 3 roles: (1) publisher (Sensor Node) as a client that will publish topics in the form of a gateway to the broker; (2) Subscribers as clients, will get data from the topics that are subscribed to; (3) Broker as server [8]. In this study, 2 sensor nodes and 1 mini server were used to send notifications to Telegram. This prototype is expected to be able to immediately send fire information so that officers can handle forest smoke to prevent large fires.

II. METHOD

A. System Design

In the Wireless Sensor Network, sensor nodes are sensor points consisting of several types of sensors and microcontrollers. Temperature, smoke, and fire sensors are used in this study. These sensors act as inputs, the microcontroller functions to read the sensors and process the read data then the data is sent to the next process using the ESP-Now protocol. The gateway serves as a bridge from the sensor nodes to the MQTT broker. At the gateway there are two microcontrollers. The first microcontroller as an ESP-Now receiver and the second microcontroller as an MQTT publisher.

Brokers are servers that function to classify and process each topic received and send it to each subscriber. In this study using a Mosquito broker, this broker will receive 8 topics. The Client-Subscriber will subscribe to the 8 topics so that they can receive each message, then the message is processed and classified and then sends the information to Telegram.

There are several fire statuses in this system, namely:

- Normal : conditions in area 1 or 2 and or both are in good condition.
- Alert 2 : conditions in area 1 or 2 and/or both detected smoke or increased temperature.

- Alert 1 : conditions in area 1 or 2 and or both are detected with smoke and an increase in temperature. In this condition it is advisable to check the detected area.
- Emergency : conditions in area 1 or 2 and or both are detected by fire.

The ranger or person concerned must have the telegram application and join a bot called FireForestDetection_ in order to receive fire information. Figure 2 is a flow diagram of the working tool system. Each sensor node sends data to the broker through the gateway. Subscribers subscribe to the topic of the broker and process each incoming payload to determine whether or not a fire occurs. Figure 3 is a flow chart on the sub-processes of sensor node 1, sensor node 2, and gateway. Figure 4 is a flow chart on the subscriber sub-process.

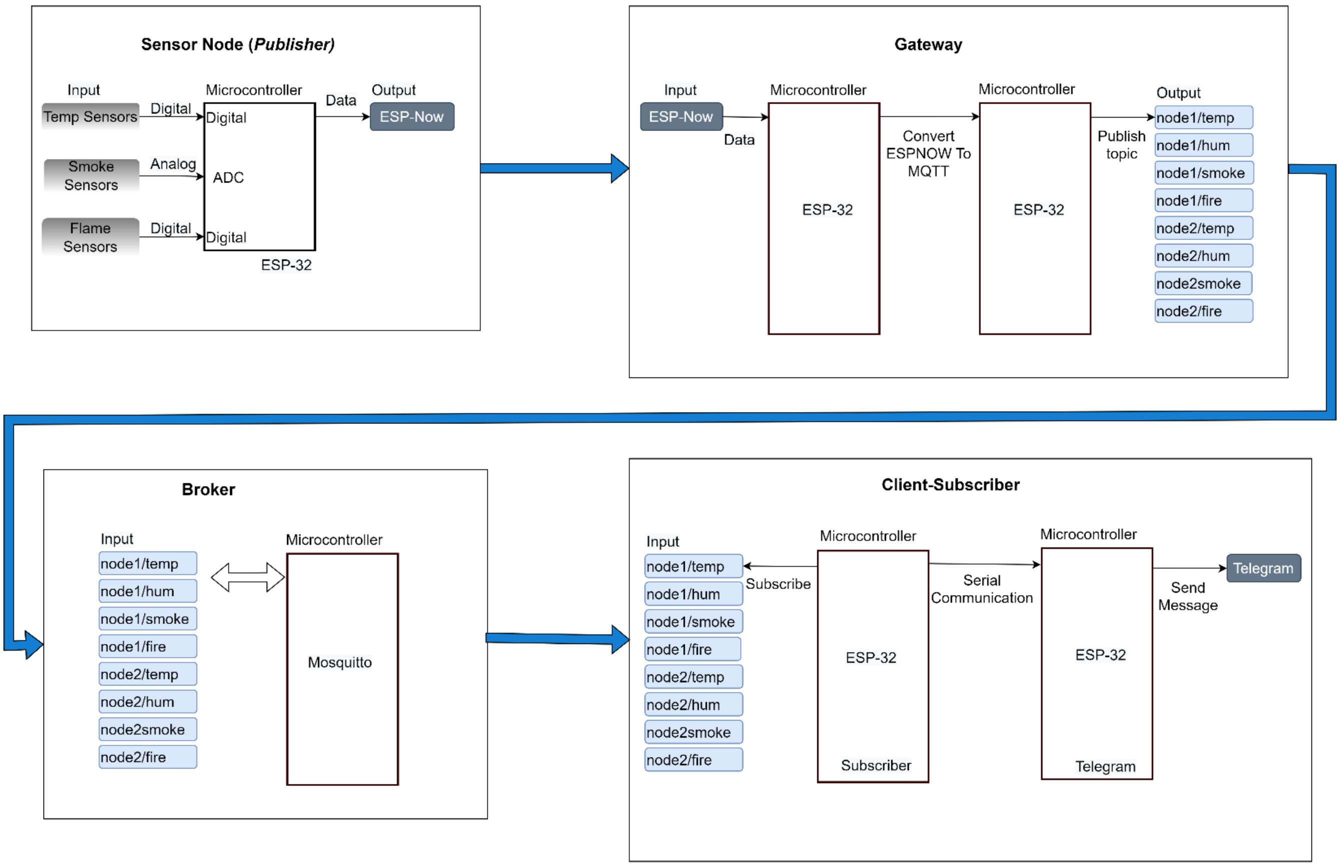

Fig. 1. Block Diagram System

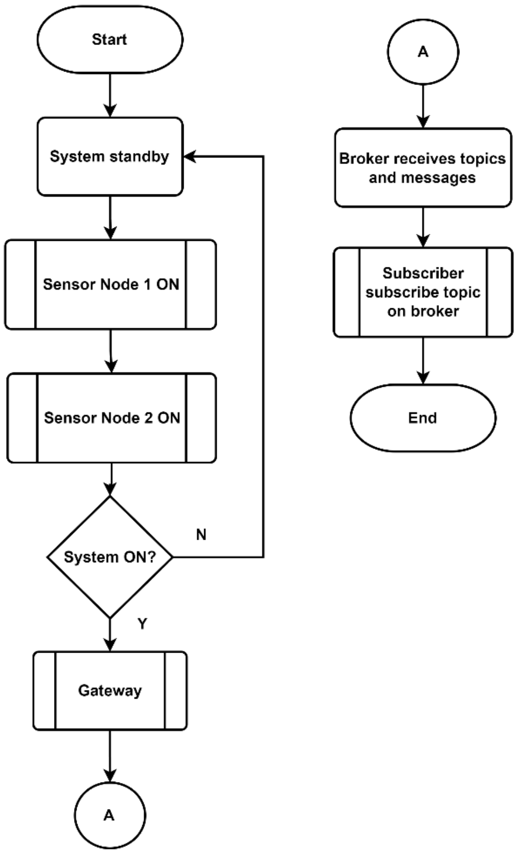

Fig. 2. Flowchart the working tool system

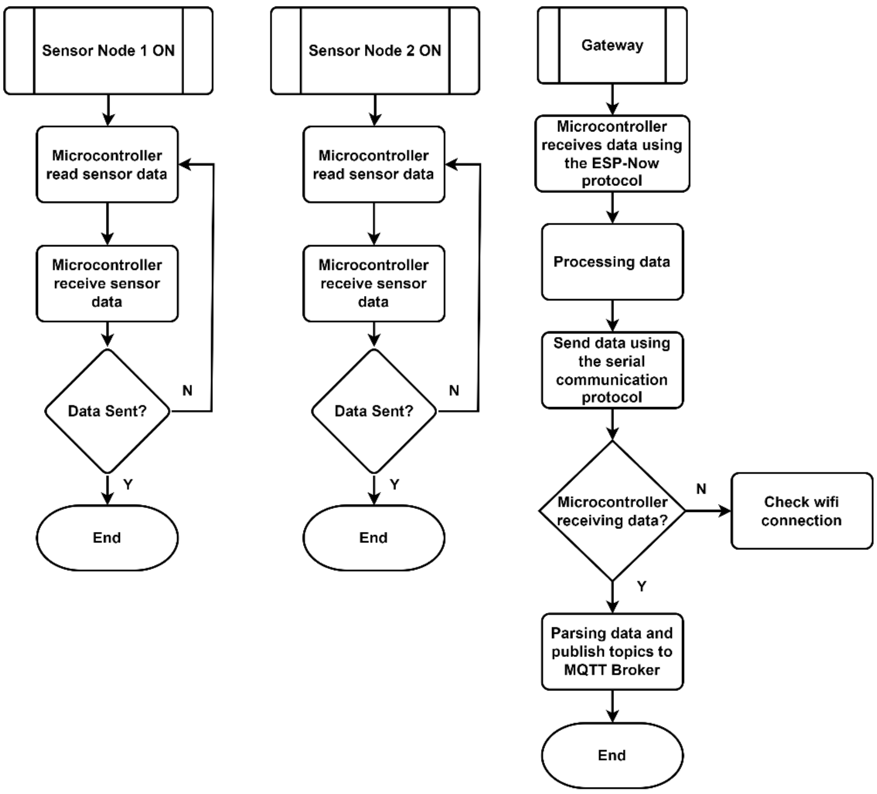

Fig. 3. Flowchart on the sub-processes of sensor node 1, sensor node 2, and gateway

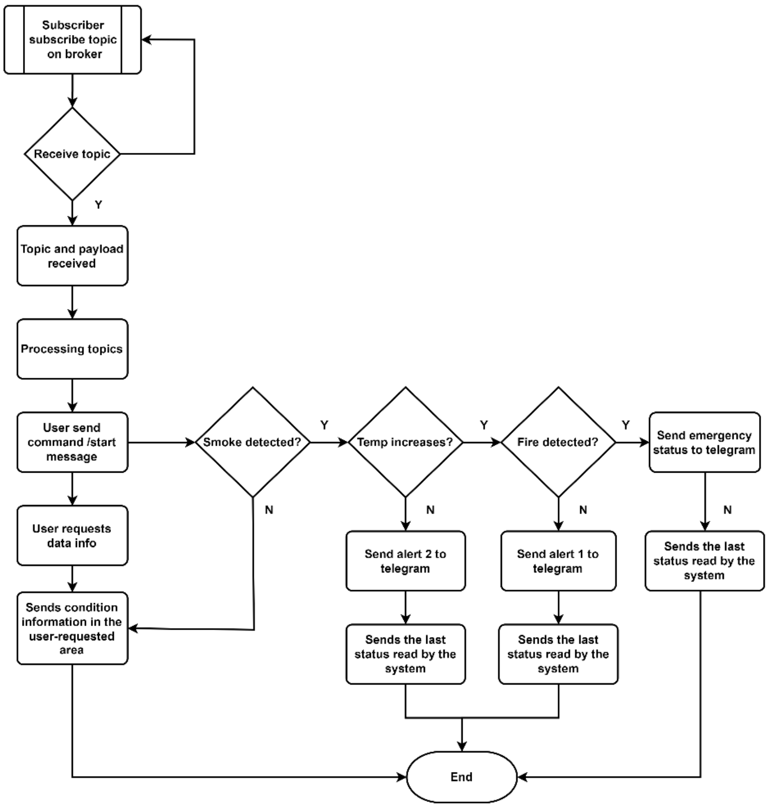

Fig. 4. Flowchart of the subscriber sub-process.

B. Prototype Design

The design of the prototype design is the stage of designing the desired sensor position and designing the prototype design for the gateway and client-subscriber as shown in Figure 5. Where the sensors and other components are placed in the box.

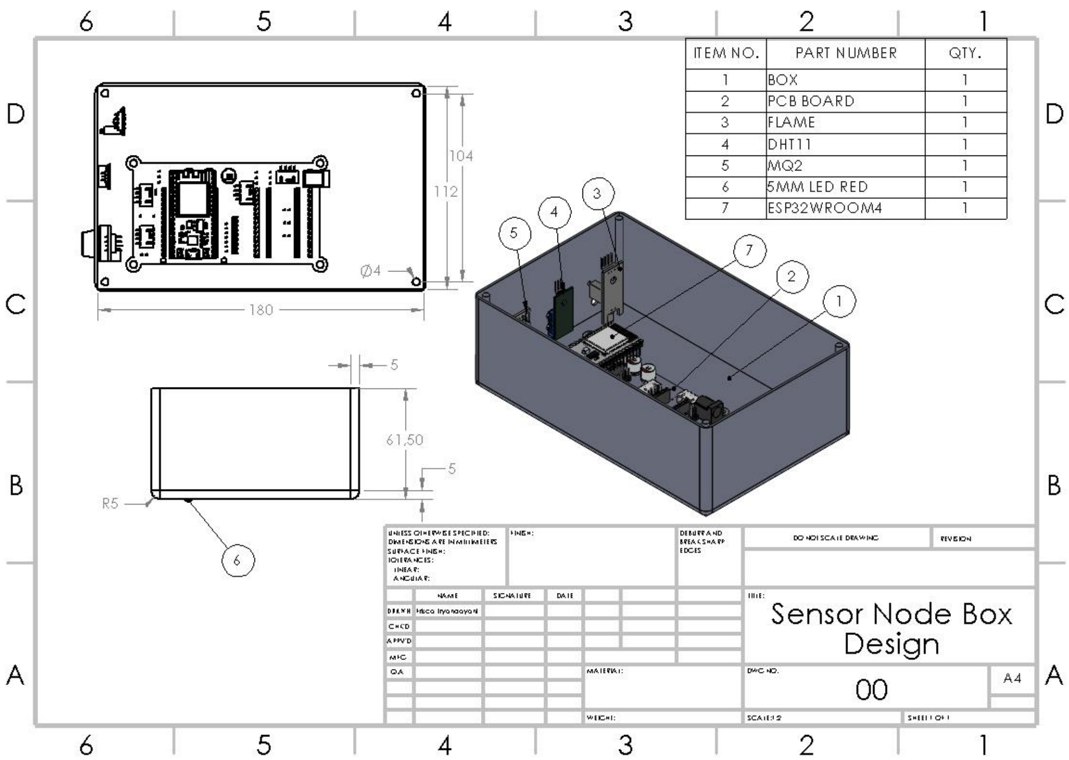

Fig. 5. Prototype Design

III. RESULTS AND DISCUSSION

A. Sensor Node to Gateway Testing

This test was carried out to determine the maximum distance of the ESP-Now protocol in transmitting data. In this test, the node is not placed in the box and is tested by sending 10 data from the sensor node to the gateway. The parameters used are delay and packet loss. Data retrieval in this test uses a maximum distance of 60 m with an interval of 4 m. Average delay is the value of the overall delay divided by the number of packets sent. Packet loss is data lost when sending data. To calculate the average delay and packet loss using equations 1 and 2.

$$Average\ delay = \frac{Total\ delay}{Number\ of\ packet} \tag{1}$$

$$Packet\ loss = \frac{Data\ packets\ sent - Data\ packets\ received}{Data\ packets\ sent} \times 100\% \tag{2}$$

$$Delay = \frac{Duration\ of\ observation}{Number\ of\ packet - 1} \tag{3}$$

$$Throughput = \frac{Number\ of\ bytes}{Duration\ of\ observation} \times 8\ bit \tag{4}$$

$$Average\ delay = \frac{Total\ delay}{Number\ of\ trials} \tag{5}$$

$$Average\ throughput = \frac{Total\ throughput}{Number\ of\ trials} \tag{6}$$

TABLE I
SENSOR NODE TEST RESULT TO GATEWAY

No	Distance(m)	Delay average (ms)	Packet sent	Packet received	Packet Loss
1	4	1,1	10	10	0%
2	8	1	10	10	0%
3	12	1	10	10	0%
4	16	1	10	10	0%
5	20	0,9	10	10	0%
6	24	0,8	10	10	0%
7	28	0,9	10	10	0%
8	32	0,9	10	10	0%
9	36	1	10	10	0%
10	40	1,2	10	10	0%
11	44	0,9	10	10	0%
12	48	333	10	7	30%
13	52	1000	10	2	80%
14	56	251	10	5	50%
15	60	-	10	0	100%

Based on Table 1, the test results of the sensor node to the ESP-Now gateway can be concluded that the maximum distance the ESP-Now protocol transmits data is 44-meters with a total delay of 0.9 ms and packet loss of 0%.

B. MQTT Gateway Testing to Subscriber (On)

This test was conducted to determine the performance of MQTT in publishing topics with the help of the Wireshark application. In this test, subscribers are connected and use QoS 1 and 0. The test was carried out 15 times with the length of observation of each trial which was 2 minutes. The parameters used are delay, throughput, and packet loss. Delay is the delay time in sending packets calculated using equation 3. Throughput is the average speed at which packets are delivered calculated using equation 4. Packet loss is calculated using equation 2. To get the average value of the entire test on delay and throughput is calculated using equations 5 and 6.

Figure 6 is the result of data capture by Wireshark using QoS 0 service. Figure 7 is the result of data capture by Wireshark using QoS 1 service. Based on table 2 it can be concluded that QoS 0 has an average delay of 0.050 s, an average throughput of 31355 bps and packet loss of 0%.

In table 3, the calculation of delay, throughput and packet loss uses equations 3, 4 and 2. The average value of the overall delay and throughput test is calculated using equations 5 and 6. Based on table 3, it can be concluded that QoS 1 has an average delay of 0.064 s, an average throughput of 30683 bps, and packet loss of 0%.

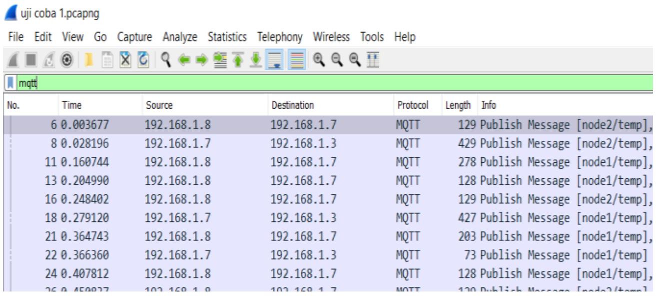

Fig. 6. Wireshark *Data Capture* Results at QoS 0

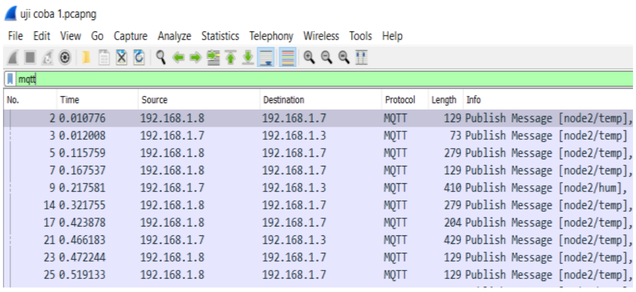

Fig. 7. Wireshark Data Capture Results at QoS 0

TABLE 2
SUBSCRIBER QOS 0 RESULT

No	Trial	Total bytes	Total packet	Long Observation(s)	Throughput(bps)	Delay(s)	Packet loss
1	Trial 1	464538	1974	120	30952	0,061	0%
2	Trial 2	461898	1945	120	30855	0,062	0%
3	Trial 3	459808	1870	120	30593	0,064	0%
4	Trial 4	465219	1985	120	30985	0,061	0%
5	Trial 5	509428	2810	120	33988	0,043	0%
6	Trial 6	513740	2885	120	34260	0,042	0%
7	Trial 7	517898	2957	120	34484	0,041	0%
8	Trial 8	516706	2920	120	34307	0,041	0%
9	Trial 9	521628	3015	120	34710	0,040	0%
10	Trial 10	511929	2860	120	34112	0,042	0%
11	Trial 11	518382	2963	120	34517	0,041	0%
12	Trial 12	502806	2713	120	33453	0,044	0%
13	Trial 13	511971	2831	120	34012	0,043	0%
14	Trial 14	310886	1941	120	20745	0,062	0%
15	Trial 15	275438	1783	120	18352	0,067	0%
		Total			470324	0,752	0%
		rata-rata			31355	0,050	0%

TABLE 3
SUBSCRIBER QOS 1 RESULT

No	Trial	Total bytes	Total packet	Long Observation(s)	Throughput(bps)	Delay(s)	Packet loss
1	Trial 1	483904	2350	120	32311	0,051	0%
2	Trial 2	457649	1857	120	30573	0,065	0%
3	Trial 3	466048	1995	120	31026	0,060	0%
4	Trial 4	461361	1922	120	30780	0,062	0%
5	Trial 5	460998	1913	120	30730	0,063	0%
6	Trial 6	461891	1927	120	30802	0,062	0%
7	Trial 7	459397	1879	120	30612	0,064	0%
8	Trial 8	465572	1997	120	31051	0,060	0%
9	Trial 9	460340	1899	120	30712	0,063	0%
10	Trial 10	457008	1868	120	30485	0,064	0%
11	Trial 11	460365	1897	120	30692	0,063	0%
12	Trial 12	431108	1468	120	28768	0,082	0%
13	Trial 13	456371	1841	120	30465	0,065	0%
14	Trial 14	457205	1832	120	30477	0,066	0%
15	Trial 15	461167	1922	120	30754	0,062	0%
		Total			460239	0,953	0%
		rata-rata			30683	0,064	0%

C. *MQTT Gateway Testing to Subscribers (Off)*

In MQTT there is a retained messages feature. This test was conducted with the subscriber off condition, but the publisher still published the topic by activating the retained messages feature. This test was conducted to determine the performance of the retained messages feature offered by MQTT. Figure 8 and 9 are the results of publishing topics done by publishers when subscribers are not connected. Publisher publishes 4 topics 10 times. Figures 10 and 11 are the topics that enter when subscribers connect using QoS 0 and 1. Based on the tests that have been done, it can be concluded that the retained messages feature on MQTT is a feature that saves the last data published by the publisher to the broker and sends it to subscribers when connected. In QoS services 0 and 1 there is no difference in these retained messages feature.

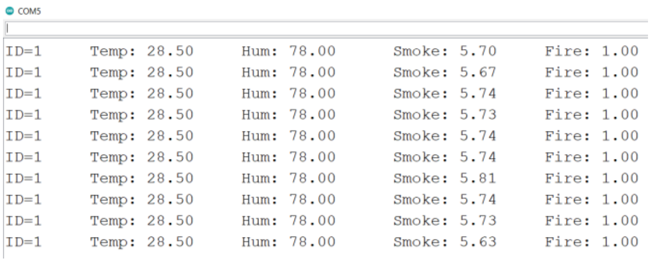

Fig. 8. Results of Publisher when Publishing Topic (QoS 0)

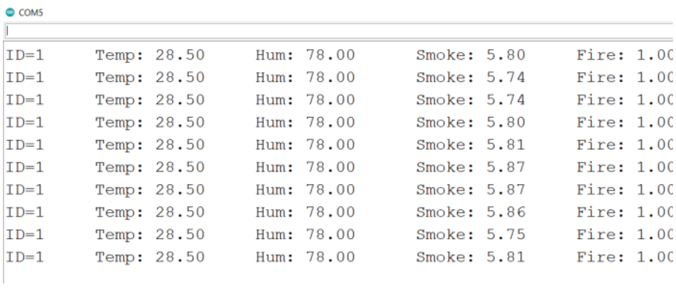

Fig. 9. Results of Publisher when Publishing Topic (QoS 1)

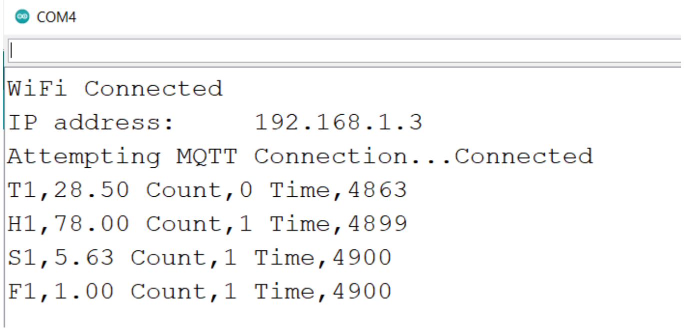

Fig. 10. Results of Using the Retained Messages Feature (QoS 0)

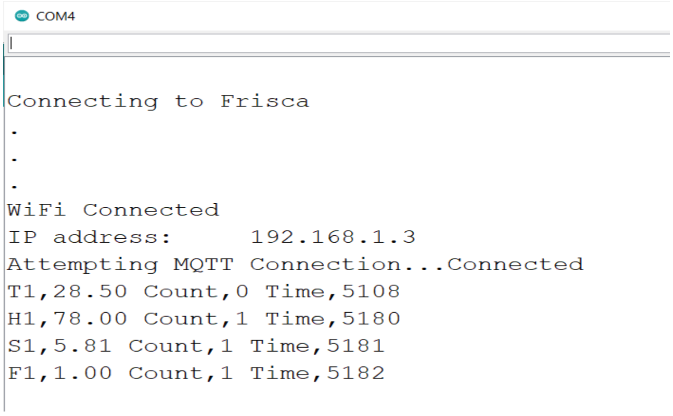

Fig. 11. Results of Using the Retained Messages Feature (QoS 1)

D. *Subscriber Testing to Telegram*

In this test, a trial was carried out in sending notifications to telegram 10 times to find out whether the notification was successfully received or not successfully received. Figures 12 display telegram notifications on each fire status at nodes 1 and 2. Based on Tables 4 and 5 it can be concluded that telegram successfully transmits data both on node 1 and node 2.

Fig. 12. Telegram Notification Display on Node 1

E. Tool Testing

Fig. 13. Tool Testing Locations

In this test (Figure 13), the tool has been positioned in the simulation place to find out whether the tool works well or not in sending fire information to telegram. Figure 16 is the test location of the tool. In this test, the distance between node 1 and node 2 is 5.9 meters.

TABLE 4
RESULT OF DISTANCE BETWEEN SENSOR NODES TO THE GATEWAY

Distance (cm)	Height Gateway from the ground (cm)	Height Node from the ground (cm)	Status
100	60	60	Success
150	60	60	Success
200	60	60	Success
250	60	60	Success
300	60	60	Success
350	60	60	Success
400	60	60	Success
450	60	60	Success
500	60	60	Not Success

Based on Table 4, it can be concluded that the maximum distance between nodes and gateways in sending data if nodes and gateways are placed in boxes is 4.5 m. Figure 13 is a picture of the position of nodes 1 and node 2 where nodes are placed 60 cm above the ground.From the tests that have been carried out can be seen in tables 4, it can be concluded that the system is good at sending fire information to telegrams both on node 1 and node
2.

IV. CONCLUSION

Based on the results of the design and testing of the Prototype Forest Fire Early Detection System Using the MQTT Method, it can be concluded that the system is capable of knowing the condition of the forest so that when a fire is detected, the system is able to send that information to a telegram. The maximum distance between sensor nodes and gateways using the ESP-Now protocol is different if the nodes and gateways are placed in a box and not in a box. Where is the maximum distance from the node to the gateway where the node and gateway are not in the box, namely 44-meters with a total delay of 0.9 ms and 0% packet loss and the maximum distance from the node to the gateway if the node and gateway are placed in a box with a closed position, namely 4.5- meters. By obtaining these results, it can be designed to position the position of the sensor node towards the gateway. In the MQTT protocol, subscribers who use QoS 0 services get an average delay of 0.050 s, an average throughput of 31355 bps and a packet loss of 0%. For subscribers who use QoS 1 services, the average delay is 0.064 s, the average throughput is 30683 bps, and 0% packet loss. Based on the results of testing these two QoS services, it can be concluded that QoS 0 service is better than QoS 1. Both of these services are used according to system requirements. On this system using QoS 0 service is good. The MQTT protocol provides a retained messages feature that can be used so that subscribers who are not connected and when connected again can receive the last message published by the publisher. On this system this feature cannot be used. The MQTT protocol uses the publish-subscribe concept and with its QoS service it makes subscribers sure to receive messages and the MQTT protocol is very well implemented in this system because this protocol utilizes the IoT system so that it can be implemented in a wide area.

REFERENCES

[1] "InfoPublik - Luas Kawasan Hutan Indonesia 125,9 Juta Hektare." https://www.infopublik.id/read/259865/luas-kawasan-hutan-indonesia-1259-juta-hektare-.html (accessed Aug. 11, 2023).

[2] R. B. Edwards, R. L. Naylor, M. M. Higgins, and W. P. Falcon, "Causes of Indonesia's forest fires," *World Dev*, vol. 127, p. 104717, Mar. 2020, doi: 10.1016/J.WORLDDEV.2019.104717.

[3] K. Prasad Vadrevu *et al.*, "Forest fires and management efforts in Indonesia (a review)," *IOP Conf Ser Earth Environ Sci*, vol. 504, no. 1, p. 012013, May 2020, doi: 10.1088/1755-1315/504/1/012013.

[4] I. Cobelo *et al.*, "The impact of wildfires on air pollution and health across land use categories in Brazil over a 16-year period," *Environ Res*, vol. 224, p. 115522, May 2023, doi: 10.1016/J.ENVRES.2023.115522.

[5] R. Libonati *et al.*, "The Effect of Forest Fire Events on Air Quality: A Case Study of Northern Colombia," *Fire 2022, Vol. 5, Page 191*, vol. 5, no. 6, p. 191, Nov. 2022, doi: 10.3390/FIRE5060191.

[6] N. Varela, J. L. Díaz-Martinez, A. Ospino, and N. A. L. Zelaya, "Wireless sensor network for forest fire detection," *Procedia Comput Sci*, vol. 175, pp. 435–440, Jan. 2020, doi: 10.1016/J.PROCS.2020.07.061.

[7] L. Yu, N. Wang, and X. Meng, "Real-time Forest fire detection with wireless sensor networks," *Proceedings - 2005 International Conference on Wireless Communications, Networking and Mobile Computing, WCNM 2005*, vol. 2, pp. 1214–1217, 2005, doi: 10.1109/WCNM.2005.1544272.

[8] G. Zheng, L. Zhang, and J. Li, "An Energy Supply System for Wireless Sensor Network Nodes," *http://dx.doi.org/10.1155/2012/603709*, vol. 2012, May 2012, doi: 10.1155/2012/603709.

[9] N. Hendrarini, M. Asvial, and R. F. Sari, "Wireless Sensor Networks Optimization with Localization-Based Clustering using Game Theory Algorithm," *International Journal of Technology*, vol. 13, no. 1, pp. 213–224, 2022, doi: 10.14716/IJTECH.V13I1.4850.

[10] M. Ragnoli *et al.*, "LoRa-Based Wireless Sensor Network System for Aquatic Elements and Flood Early Warning Monitoring," *Lecture Notes in Electrical Engineering*, vol. 999, pp. 124–128, 2023, doi: 10.1007/978-3-031-25706-3_20/COVER.

[11] K. P. Kamble, "LoRa Communication based Wireless Sensor Monitoring system," *International Conference on Sustainable Computing and Data Communication Systems, ICSCDS 2022 - Proceedings*, pp. 1127–1131, 2022, doi: 10.1109/ICSCDS53736.2022.9760796.

[12] F. Z. Rachman *et al.*, "Design of the early fire detection based fuzzy logic using multisensor," *IOP Conf Ser Mater Sci Eng*, vol. 732, no. 1, p. 012039, Jan. 2020, doi: 10.1088/1757-899X/732/1/012039.

[13] B. Sarwar, I. S. Bajwa, S. Ramzan, B. Ramzan, and M. Kausar, "Design and Application of Fuzzy Logic Based Fire Monitoring and Warning Systems for Smart Buildings," *Symmetry 2018, Vol. 10, Page 615*, vol. 10, no. 11, p. 615, Nov. 2018, doi: 10.3390/SYM10110615.

[14] A. Zohourian *et al.*, "IoT Zigbee device security: A comprehensive review," *Internet of Things*, vol. 22, p. 100791, Jul. 2023, doi: 10.1016/J.IOT.2023.100791.

[15] V. D. Gavra and O. A. Pop, "Usage of ZigBee and LoRa wireless technologies in IoT systems," *2020 IEEE 26th International Symposium for Design and Technology in Electronic Packaging,*

SIITME 2020 - Conference Proceedings, pp. 221–224, Oct. 2020, doi: 10.1109/SIITME50350.2020.9292150.

[16] A. Allakany, A. Saber, S. M. Mostafa, M. Alsabaan, M. I. Ibrahem, and H. Elwahsh, “Enhancing Security in ZigBee Wireless Sensor Networks: A New Approach and Mutual Authentication Scheme for D2D Communication,” *Sensors 2023, Vol. 23, Page 5703*, vol. 23, no. 12, p. 5703, Jun. 2023, doi: 10.3390/S23125703.

[17] R. Herrero, “RTP transport in IoT MQTT topologies,” *Internet of Things and Cyber-Physical Systems*, vol. 3, pp. 37–44, Jan. 2023, doi: 10.1016/J.IOTCPS.2023.02.001.

[18] Z. Kegenbekov and A. Saparova, “Using the MQTT Protocol to Transmit Vehicle Telemetry Data,” *Transportation Research Procedia*, vol. 61, pp. 410–417, Jan. 2022, doi: 10.1016/J.TRPRO.2022.01.067.