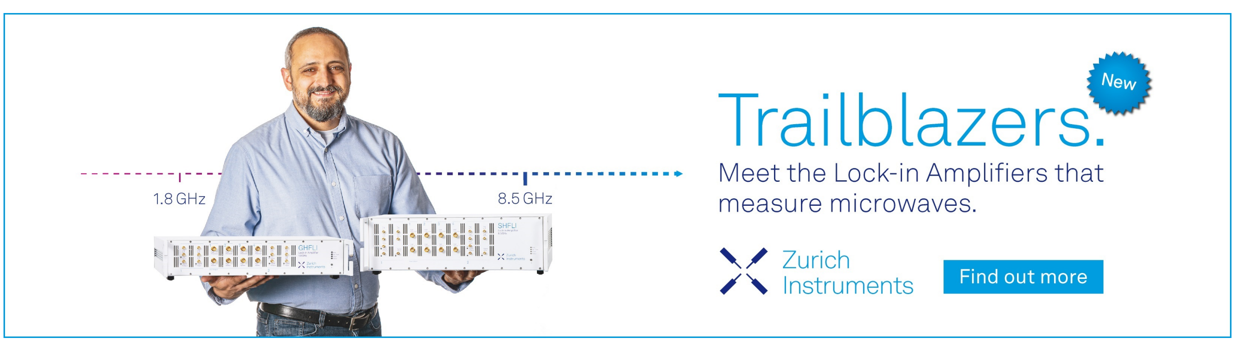
AIP Conference Proceedings

Data transmission system of LPG leakage using telegram

Cite as: AIP Conference Proceedings **2542**, 030004 (2022); https://doi.org/10.1063/5.0103810
Published Online: 10 November 2022

Kasih Rudi Halawa and Damar Widjaja

View Online
Export Citation



© 2022 Author(s).

Data Transmission System of LPG Leakage Using Telegram

Kasih Rudi Halawa[a] and Damar Widjaja[b]

Electrical Engineering Department, Faculty of Science and Technology,
Universitas Sanata Dharma, Yogyakarta, Indonesia

[a]rudyhalawa0199@gmail.com
[b]Corresponding author: damar@usd.ac.id

Abstract. Safety is a main concern of any aspects in human life. Gas leakages in open or closed areas can be dangerous and threatening. There are many difficulties in preventing gas leakage, such as late information to alert the people about the leakage so that the leaked gas will continue to spread to all parts of the building and cause fires that gives substantial material losses. Therefore, it is necessary to have a detection system to provide early warning if gas leakage will occur, this is very important to prevent material losses from fires. IoT based LPG leakage data transmission system was designed to automatically detect the gas leakages from two different locations. There are many devices to build this system, such as NodeMCU as the microcontroller, Telegram as a messager app for notification, Thingspeak as an IoT platform, MQ-5 sensor to sense the presence of gas, and an exhaust fan to pull out the leaked gas from the area. The results of this work show that the systems designed work well. MQ-5 sensor can sense the gas leakage well and the systems are also able to send the Telegram notifications to user, and then the gas leakage data also can be displayed on Thingspeak correctly so that people could be made aware in advance by performing data tranmission on sensor readings.

INTRODUCTION

The development of industrial revolution 4.0 technology is currently making it easier for humans to access information, one of its kind is Internet of Things (IoT) technology. Through technology such as the internet, all human needs can be met [1]. Starting from the need to socialize, access information to the fulfillment of entertainment needs. Now, its presence is more used as social media by the public. One example of social media is Telegram. Telegram is currently a cloud-based instant messaging application which means it can easily transfer conversations from smartphones, tablets, web or desktops focused on speed and security [2]. Telegram is designed to make it easier for users to send text messages, audio, video, images and stickers to each other safely.

In addition, Telegram is not limited to just ordinary users, but can provide open access for application developers, especially in the IoT field who utilize Telegram with the bot service [3].Telegram also has an Application Program Interface (API) that allows software developers to create and integrate Telegram with other applications for free, such as IoT for safety and security [4].

The safety aspect is currently very much needed in various aspects of life such as the safety of houses, buildings, rooms that have important values for the owner. For example, the improper installation of LPG gas cylinders can cause gas leakage which can later trigger an explosion or fire [5]. There are many difficulties in preventing gas leakage, such as late information to the building owner due to being outside or not at the area so that the leaked gas will continue to spread to all parts of the building and cause fires that gives substantial material losses. IoT technology is a solution to overcome this problem. IoT works to find and collect various data from the remote places which will later be processed into more useful data.

In 2018, Andry Stepanus Tumanggor has made a gas leakage system that is able to detect the leakage of LPG, and is only able to send Short Message Service (SMS) automatically to mobile phone if a gas leakage has occurred [6]. The LPG leakage detection system that was also made by Muhammad Gilbie in 2018, is an automatic LPG leakage and fire detection system that will send messages via Telegram to user [5].

The 4th International Conference on Science and Science Education (IConSSE 2021)
AIP Conf. Proc. 2542, 030004-1–030004-11; https://doi.org/10.1063/5.0103810
Published by AIP Publishing. 978-0-7354-4259-7/$30.00

The difference between the research conducted by the authors and Andry Stepanus Tumanggor is that in this study, the LPG leakage notification or warning system will use Telegram. While the differences in the research conducted by the authors compared to Muhammad Gilbie's research are this study develops LPG leakage detection systems from two different locations and also develops the research to use Thingspeak as an IoT platform for remote monitoring.

Therefore, the authors offer innovative device for data transmission system of LPG leakage through this study that can detect the leakage from two different places or locations by using the MQ-5 gas sensor. The results of the gas leakage data reading by the MQ-5 sensor are then processed by the microcontroller to send notifications to the user via Telegram. Furthermore, the microcontroller will also send data to the Thingspeak website for monitoring, then faster actions can be taken so that hazard such as fires caused by LPG leakage would not occur.

THEORETICAL BACKGROUND

Internet of Things (IoT)

The Internet of things is the internetworking of physical devices like vehicles, buildings, electronic or any general appliances and other connected devices embedded with sensors, network connectivity, actuators, et cetra which lets these devices to exchange data among themselves and perform any action as per requirement [7]. It enables sensing and control from remote location. Hence, it creates a platform for integration of physical world with the network infrastructure leading to improved accuracy and efficiency with minimizing the time needed to carry out the process manually.

Telegram

Telegram is a messaging application for smartphones with basic similarities to WhatsApp Messenger [8]. Telegram messenger is a cross-platform messaging application that allows its user to exchange messages without SMS fees, because Telegram uses the same internet data for e-mail, web browsing, and others. Telegram application uses GPRS/3G/4G or WiFi connection for data communication. By using Telegram, users can have online chat, share files, exchange photos and others. Telegram also has a bot feature. One of the main functions of bot is to facilitate human tasks. Telegram is one application that supports this bot. With this bot, it is easier for user to create a special kind of chat application and also replace the task of moderation in a group.

Thingspeak

Thingspeak is an IoT platform which allows user to collect, visualize, analyze live data and react according to it [9]. It helps one to build IoT systems without need of setting up extra servers. The data collection is done using REST API or MQTT. The main component of Thingspeak is its channel which stores data that are sent from various devices. Each channel can save up to eight fields along with device location and url. The channel can be made public which can be seen by other users or private which need the API key to view the data. The private channel can be shared for some specific users.

RESEARCH METHODOLOGY

This study is conducted to detect the leaked gas automatically from two different places and therefore each system is named as Kitchen 1 and Kitchen 2 to differentiate the locations. Both systems have the same process of workings and each system also has NodeMCU as the transmitter for data transmission while both Telegram and Thingspeak act as the receiver's side. Before the sensor works, the NodeMCU will firstly connect to Wi-Fi for internet access.

The block diagram for the data transmission system of LPG leakage is shown in Fig. 1. Overall, these systems consist of hardware and software. The hardware includes the type of sensor required and a microcontroller which is NodeMCU. The MQ-5 sensor is the main component of this device. This sensor functions as an input that can detect the presence of gas leakage in the air. The output of this sensor in the form of voltage will then be processed by the microcontroller and run the instructions according to the program that has been made in Arduino IDE. Different color LEDs are also used to specify the gas leakage for example, red LED indicates the presence of gas in the area and if gas is not yet detected, the green LED will be turned on to indicate that area is safe from gas leakage. Furthermore, if the sensor detects gas leakage in dangerous level, it will automatically work and activate the fan. After uploading the program or instructions into the microcontroller, the data is processed and converted into an active signal.

Meanwhile, the software used are Arduino IDE, Telegram and Thingspeak. The way to communicate the NodeMCU via bot can be done through Telegram configuration. The first step to make a Telegram bot is to register a bot account by using the @botfather service from Telegram application on user's smartphone. There are several commands provided by the @botfather. In addition, select the command /newbot to create a new bot and fill in as ordered. The bot registration is done to get a bot token which will be used for programming into the NodeMCU that allows it to communicate with the bot later. The transmission of LPG leakage notification messages to Telegram bot is made into 3 types, they are:

1. "***Watch out! The gas leakage has reached 300 ppm in (kitchen name). Currently, detected gas data is (gas data) ppm***". This notification message will be sent twice when the gas leakage is detected starting from 300 ppm to 399 ppm.
2. "***Warning! The gas leakage has reached 400 ppm in (kitchen name). Currently, detected gas data is (gas data) ppm. Please read this message immediately***". This notification message will be sent three times when the gas leakage is detected starting from 400 ppm to 499 ppm.
3. "***Danger! The gas leakage has reached 500 ppm in (kitchen name). Currently, detected gas data is (gas data) ppm. Please take precautionary action immediately***". This notification message will be sent four times when the gas leakage is detected starting from 500 ppm to 1000 ppm.

All messages are sent by NodeMCU to Telegram as a notification of gas leakage in an area (kitchen) according to gas level detected. For Thingspeak monitoring display, each data is sent in different channels and stored in different fields as well by NodeMCU according to API keys that are provided in the channels created. Through these diffrerent API keys, NodeMCU can differentiate the destination of data transmission for Kitchen 1 and Kicten 2. The API keys that have been obtained from Thingspeak channel must be made same to the API keys included in the programming for NodeMCU. When the API keys are same, NodeMCU will automatically recognize the Thingspeak channel and then connected for data transmission process.

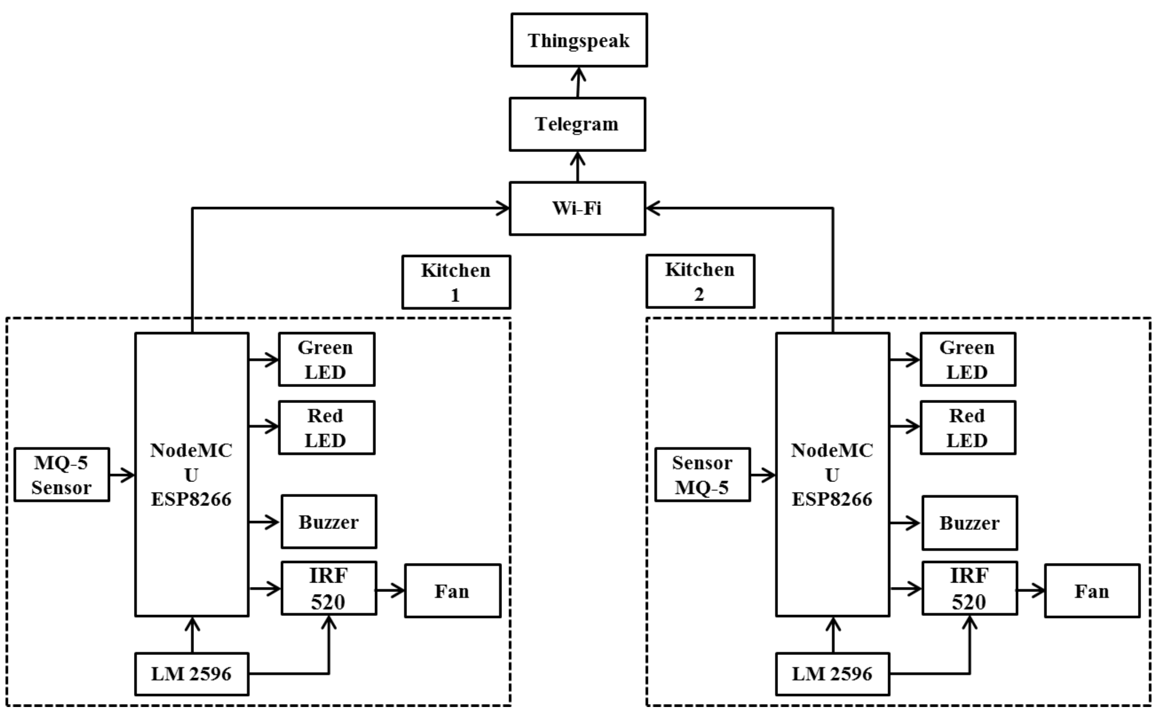
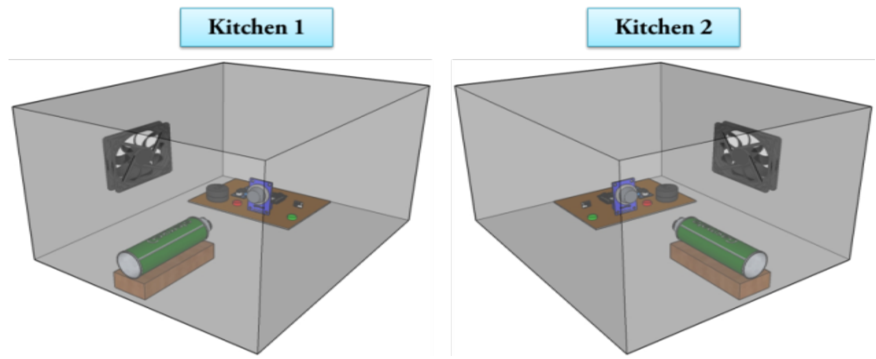
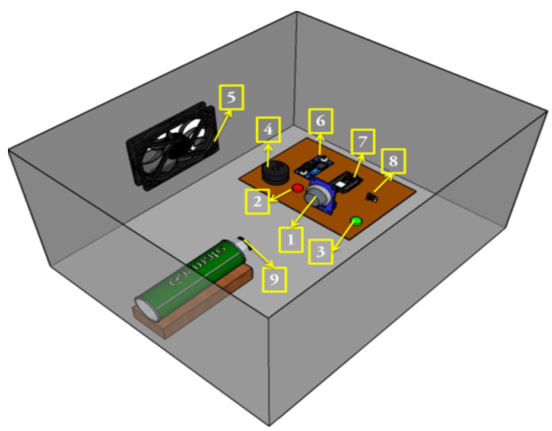


FIGURE 1. General block diagram.

The system model that will be designed in this study can be seen in Fig. 2. The prototype design of the two kitchens is a box measures 35 cm × 30 cm × 20 cm and equipped with a fan.



(a)



(b)

FIGURE 2. System model (a) Kitchen 1 and Kitchen 2, (b) components used for the systems.

The explanation of the components of Fig. 3(b) is as follows:

1. The MQ-5 sensor is used to sense the leaked gas.
2. The red LED is used as an indicator that gas leakage has occurred.
3. The green LED is used as an indicator that the gas in a room is still safe or has not leaked.
4. Buzzer functions to make a sound as a notification of a gas leakage.
5. Fan is used to take the leaked gas out from the area.
6. NodeMCU as a microcontroller in the system to process each input according to the program that has been flashed/uploaded.
7. LM2596 is used to lower the voltage from 12V to 5V.
8. IRF520 is used as a driver for fan.
9. Canned LPG (portable gas).

General Flowchart

The flowchart of the programming that will be made can be seen in Fig. 3. The flowchart is started by the input/output initialization of the system includes the analog reading from the the gas sensor which is then coverted into digital signal by the microcontroller. In this study, the NodeMCU must be connected to a Wi-Fi network or hotspot for internet access so that LPG leakage notification messages can be sent and received on the user's smartphone through Telegram and transmitted gas leakage data can also be stored and displayed on Thingspeak.

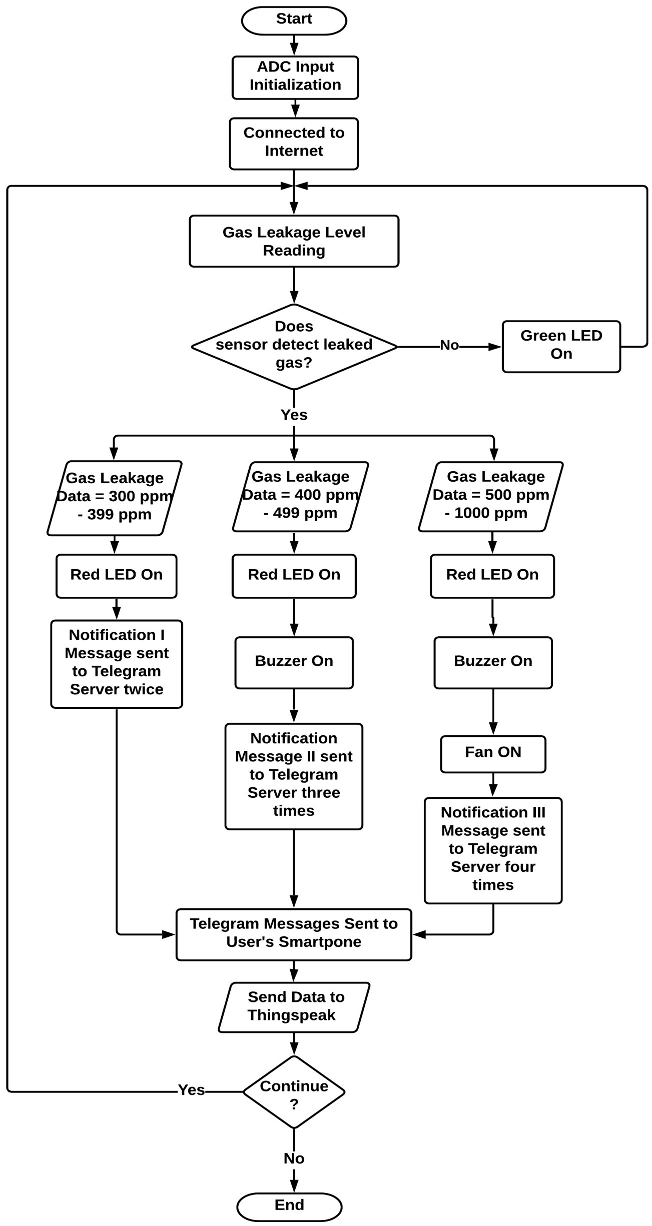


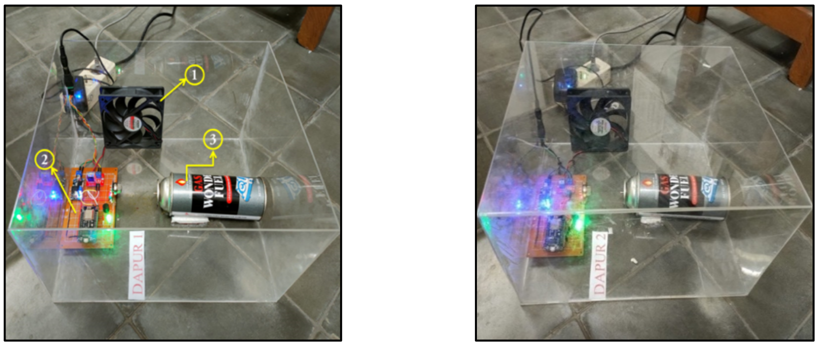
FIGURE 3. General flowchart.

Every value recorded by the sensor is checked against the threshold values. These threshold values are decided based upon the nature of the gas and also on the behaviour of the sensor. When the sensor has not yet detected any gas leakage in the air, the systems will turn the green LED on. As the opposite, when the sensor has detected the leaked gas, the systems wil work into three conditions. Firstly, if the gas leakage is detected, starting from 300 ppm to 399 ppm, the red LED is on and the notification I will be sent twice to Telegram. Secondly, if the gas leakage is detected, starting from 400 ppm to 499 ppm, the red LED and buzzer are on and the leakage notification II will be sent three times to Telegram. Lastly, if the gas leakage is detected starting from 500 ppm to 1000 ppm, the red LED, buzzer and fan are on and the leakage notification III will be sent four times to Telegram. After sending these notifications to user's smartphone through Telegram, these systems then record the sensor readings and maintain a database at different timestamps to Thingspeak in real time.

RESULT DAN DISCUSSION

Hardware Implementation

For the hardware implementation, the physical prototype is made from acrylic material and all components are placed into the prototype. The physical implementation of the prototype can be seen in Fig. 4 and Table 1 contains the description of prototype implementation. Figure 5 shows the wiring system used and Table 2 shows the description of electronic wiring system.



(a) (b)

FIGURE 4. Prototype implementation (a) Kitchen 2 and (b) Kitchen 2.

TABLE 1. Description of the prototype.

No	Description
1	Exhaust Fan
2	Canned LPG
3	PCB for Electronic Circuit

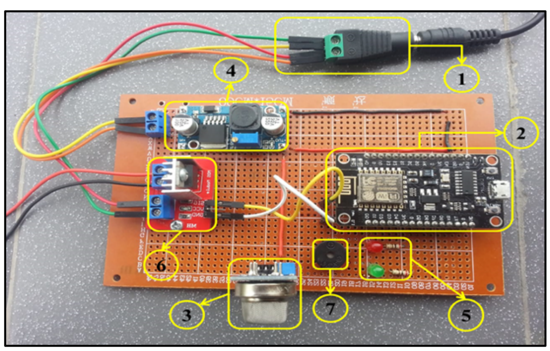


FIGURE 5. Electronic wiring for gas leakage system.

TABLE 2. Description of electronic wiring for gas leakage system.

No	Description
1	Power Supply 12V
2	NodeMCU
3	MQ-5 Sensor
4	LM2596
5	Red LED and Green LED
6	IRF520
7	Buzzer

Table 3 shows the test result data of the whole system. This test is used to measure the success rate of the data transmission system. Testing the whole system was done by spraying the gas into the prototype sourced from gas can in various level. The testing includes the ability of each system to send notifications to Telegram and then being able to send data to Thingspeak. All the experiments show that the systems were able to work in every condition as designed.

TABLE 3. Overall system test results.

Test	Gas Detected (ppm)		Gas Indicator		Gas Status		Send message to Telegram		Send Data to Thingspeak		Note
	Kitchen 1	Kitchen 2	Kitchen 1	Kitchen 2	Kitchen 1	Kitchen 2	Kitchen 1	Kitchen 2	Kitchen 1	Kitchen 2	
1	-	-	Not Leaked	Not Leaked	Not Detected	Not Detected	No	No	Yes	Yes	Successful
2	310	-	Leaked	Not Leaked	Detected	Not Detected	Yes	No	Yes	Yes	Successful
3	-	550	Not Leaked	Leaked	Not Detected	Detected	No	Yes	Yes	Yes	Successful
4	375	-	Leaked	Not Leaked	Detected	Not Detected	Yes	No	Yes	Yes	Successful
5	-	519	Not Leaked	Leaked	Not Detected	Detected	No	Yes	Yes	Yes	Successful
6	469	-	Leaked	Not Leaked	Detected	Not Detected	Yes	No	Yes	Yes	Successful
7	717	-	Leaked	Not Leaked	Detected	Not Detected	Yes	No	Yes	Yes	Successful
8	608	-	Leaked	Not Leaked	Detected	Not Detected	Yes	No	Yes	Yes	Successful
9	-	990	Not Leaked	Leaked	Not Detected	Detected	No	Yes	Yes	Yes	Successful
10	-	996	Not Leaked	Leaked	Not Detected	Detected	No	Yes	Yes	Yes	Successful
11	-	310	Not Leaked	Leaked	Not Detected	Detected	No	Yes	Yes	Yes	Successful
12	581	826	Leaked	Leaked	Detected	Detected	Yes	Yes	Yes	Yes	Successful
13	597	484	Leaked	Leaked	Detected	Detected	Yes	Yes	Yes	Yes	Successful
14	366	-	Leaked	Not Leaked	Detected	Not Detected	Yes	No	Yes	Yes	Successful
15	640	-	Leaked	Not Leaked	Detected	Not Detected	Yes	No	Yes	Yes	Successful

Table 3 shows that both systems work well. The system experiment is carried out when the gas leaked and has not leaked and also when gas is detected or not detected. It can be seen in the first experiment when the gas in the two kitchens has not leaked and the gas status is not detected, then the transmission process of leakage notification does not occur. In addition, the third experiment, the gas in Kitchen 1 has not leaked while the gas in Kitchen 2 has leaked. For that, Kitchen 1 does not send leakage notifications, while Kitchen 2 does send notifications automatically with gas lekagae level detected is 550 ppm. Besides, the two systems also send the gas leakage data reading to Thingspeak continuously. In experiment if the gas leakage is detected ≥ 500 ppm, the exhaust fan will be turned on automatically as shown in Fig. 6.

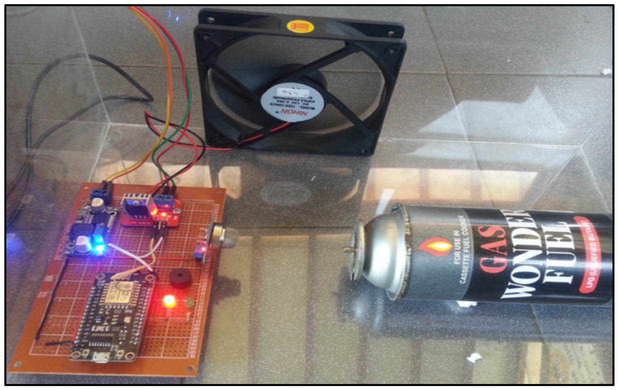


Figure 6. Exhaust fan test.

The Test of Notifications Transmission Process to Telegram

Table 4 shows the test results of notifications transmission to Telegram. This test is used to measure the delay required for sending messages from NodeMCU to Telegram. The working principle of Telegram notifications is to send gas leakage data information to the user's smartphone obtained from the MQ-5 sensor in the form of messages. Notification messages sent consist of text and numerical gas leakage data. In addition, Telegram notification messages are divided into three types based on the level limits of the leaking gas that have been detected by the MQ-5 sensor. The results of the implementation of sending gas leak notification messages using Telegram can be seen in Fig. 7.

TABLE 4. The test results of notification transmission to Telegram.

Test	Detected Gas (ppm)		Send Message						Delay(second)	
	Kitchen 1	Kitchen 2	Notification 1		Notification 2		Notification 3		Kitchen 1	Kitchen 2
			Kitchen 1	Kitchen 2	Kitchen 1	Kitchen 2	Kitchen 1	Kitchen 2		
1	350	469	Yes	No	No	Yes	No	No	5s	6s
2	690	403	No	No	No	Yes	Yes	No	4s	4s
3	330	449	Yes	No	No	Yes	No	No	4s	4s
4	528	923	No	No	No	No	Yes	Yes	7s	6s
5	510	523	No	No	No	No	Yes	Yes	5s	5s
6	424	568	No	No	Yes	No	No	Yes	4s	4s
7	505	390	No	Yes	No	No	Yes	No	4s	4s
8	868	457	No	No	No	Yes	Yes	No	5s	4s
9	424	471	No	No	Yes	Yes	No	No	6s	5s
10	647	513	No	No	No	No	Yes	Yes	4s	4s

Based on the Table 4, the required delay for message transmission of Kitchen 1 and Kitchen 2 varies between 4s until 7s. The delays of message transmission for both kitchens are strongly influenced by the network quality. If the network is in bad condition, then more delay continues.

When the sensor has read the gas leakage ranging from 300 ppm to 399, the NodeMCU send the first Telegram notification message twice as shown in Fig. 7(a). Next, NodeMCU is also able to send the second type of Telegram notification message three times when the gas leakage value is detected starting from 400 ppm to 499 ppm as shown in Fig. 7(b). Lastly, when the gas leakage value is detected ranging from 500 ppm to 1000 ppm, the NodeMCU send the third Telegram notification message four times to the user's smartphone as shown in Fig. 7(c).

FIGURE 7. The results of gas leakage notification messages (a) first notification message for Kitchen 1 and Kitchen 2, (b) second notification message for Kitchen 1 and Kitchen 2, (c) third notification message for Kitchen 1 and Kitchen 2.

The Test of Data Transmission Process to Thingspeak

Table 5 shows the test results of data transmission process. This test is aimed to observe the data transmission process from NodeMCU to Thingspeak. These systems use different channel between Kitchen 1 and Kitchen 2. Channel 1 is used to display the gas leakage data from kitchen 1, while channel 2 is used to display the gas leakage data from kitchen 2. Thingspeak is ineffective to be used for displaying the data obtained from different sources. If both systems use the same channel, then one of them will lose a lot of data during data transmission. For this reason, diffrerent channels are created to prevent the data collision. The authors has tried to use two channels for monitoring system of each place and the result is data can be sent in real time, data loss can be avoided as compared to the use of one channel for monitoring as shown in Fig. 8.

TABLE 5. Test results of data transmssion process to Thingspeak.

Test	Gas Leakage Lamp		Detected Gas (ppm)		Delay(second)	
	Kitchen 1	Kitchen 2	Kitchen 1	Kitchen 2	Kitchen 1	Kitchen 2
1	ON	ON	302	627	4s	16s
2	ON	ON	433	330	9s	4s
3	ON	ON	706	994	18s	19s
4	ON	ON	351	407	4s	8s
5	ON	ON	853	419	16s	9s
6	ON	ON	633	882	17s	16s
7	ON	ON	680	593	15s	15s
8	ON	ON	610	376	16s	5s
9	ON	ON	987	442	19s	9s
10	ON	ON	363	657	5s	16s
11	ON	ON	742	643	17s	15s
12	ON	ON	490	908	11s	18s
13	ON	ON	652	563	17s	16s
14	ON	ON	647	477	16s	9s
15	ON	ON	309	368	4s	4s

FIGURE 8. Thingspeak monitoring implementation.

As shown in the Table 5, the delay needed for data transmission of Kitchen 1 and Kitchen 2 varies between 4s until 19s because the working process of program uploaded in each system is made into a sequence. In other words, Thingspeak will receive and display the data if Telegram notification messages has been sent firstly by NodeMCU to user's smartphone then to Thingspeak. When NodeMCU sends the notification I to Telegram twice, the delay to

Thingspeak is around 4s until 5s. When NodeMCU sends the notification II to Telegram three times, the delay to Thingspeak is around 8s until 9s and 11s. Lastly, When NodeMCU sends the notification III to Telegram four times, the delay to Thingspeak is around 15s until 19s. The more messages sent to user's smartphone, the longer delay needed by the NodeMCU to send data to Thingspeak. Moreover, Thingspeak also has characteristic feature to update new data every 15s. The delay of data transmission for both kitchens is also strongly influenced by the network quality. If the network is bad, then more data will be lost caused by the long data transmision delay.

The working principle of this IoT based data monitoring is to display gas leakage data obtained from the MQ-5 gas sensor using the internet to Thingspeak. The presentation of the data is displayed in graph with an indicator lamp for each place (kitchen). As shown Fig. 8, the lamp indicator is on to indicate that the gas lekage has occurred in Kitchen 1.

CONCLUSION

The data transmission system of LPG leakage is successfully completed. Each system works well as it was designed. The MQ-5 sensor can sense the gas leakage well and both systems are also able to send the Telegram notifications to user, and then the gas leakage data also can be displayed on Thingspeak correctly. In addition, the duration of data transmission to Telegram is faster than the duration of data transmission to Thingspeak.

ACKNOWLEDGMENTS

Thanks to the Institute of Research and Community Service (Lembaga Penelitian dan Pengabdian kepada Masyarakat) Sanata Dharma University for funding and facilitating this research.

REFERENCES

1. S. F. Soliha, J. Interaksi **4**, 1–10 (2015).
2. S. Putri Nova, Online Journal of FISIP Students **5**, 1–11 (2018).
3. R. R. Prabowo and Kusnadi, J. Digit **10**, 189–195 (2020).
4. K. P. Tedjo, "Sistem Pemantauan Suhu, pH, dan Kejernihan Air dengan Layanan Telegeram API dan Website pada Raspberry Pi 3," Bachelor thesis, Diponegoro University, 2017.
5. M. Gilbie, "Prototype of Esp8266 Based Fire and Gas Leakage Detector Via Telegram," thesis, Mercu Buana Universitys, 2018.
6. A. S. Tumanggor, "Alarm Pendeteksi Kebocoran Gas Menggunakan SMS sebagai Peringatan Berbasis AT-Mega 328p," Diploma thesis, Sumatera Utara University, 2018.
7. R. K. Kodali, Greeshma R.N.V., and K. P. Nimmanapalli, and Y. K. Yogi Bora. "IOT Based Industrial Plant Safety Gas Leakage Detection System" in *4th International Conference on Computing Communication and Automation-2018*, pp. 1–5.
8. Yuliza, J. Electr. Eng. **9**, 34–56 (2018).
9. D. Nettikadan and S. Raj, Int. J. Appl. Eng. Res. **13**, 13402–13408 (2018).