CODS: Carbon Monoxide Pollution Detection and Monitoring System Based on The Internet of Things Towards Urban Health Enhancement

Muhammad Yusro [1], Aodah Diamah [1], Setia Budi [2], Ibnuh Sakti [1], Agung Pangestu [3], Rosyid Ridlo Al-Hakim [3], Ari Apriyansa [4]

[1] *Electrical Engineering Department, Faculty of Engineering, Universitas Negeri Jakarta, Jakarta, Indonesia*
[2] *Chemistry Department, Faculty of Mathematics and Natural Sciences, Universitas Negeri Jakarta, Jakarta, Indonesia*
[3] *Electrical Engineering Department, Faculty of Engineering and Computer Science, Jakarta Global University, Indonesia*
[4] *Government Information Engineering Technology, Institut Pemerintahan Dalam Negeri, Sumedang, Indonesia*

***Abstract* – The enhancement of urban air quality is a pressing global issue, crucial for safeguarding public health. Carbon monoxide (CO), a significant air pollutant, poses severe risks to human health. This study focuses on the development of a mobile-based Internet of Things (IoT) system for the detection and monitoring of carbon monoxide (CO) pollution, contributing to the advancement of healthy urban environments. Utilizing the MQ-7 sensor, the system accurately detects and quantifies CO levels, while the u-blox NEO-6M GPS receiver precisely locates measurement sites through satellite signal processing. Integration with the OV5647 camera enables the system to capture images at points of CO detection, enhancing data visualization.**

Processed data are relayed to a virtual private server (VPS) via a Raspberry Pi 3 Model B+, employing the Nginx web server for efficient data management. The MQ-7 sensor demonstrates a CO detection range of 20-100 ppm, with a minimal error rate of 4.87%. GPS accuracy tests reveal an average discrepancy of only 0.89 meters when compared to the Garmin eTrex 10, indicating high reliability. Field tests conducted across three Indonesian locales (Jakarta, Bekasi, and Depok) involved data collection via a quadcopter, culminating in the successful dissemination of CO concentration data on a publicly accessible web page, presented as an interactive map correlating to measurement locations.

***Keywords* – Carbon monoxide, air monitoring, internet of things, smart city, unmanned aerial vehicle.**

DOI: 10.18421/TEM133-85
https://doi.org/10.18421/TEM133-85

Corresponding author: Ari Apriyansa,
Government Information Engineering Technology, Institut Pemerintahan Dalam Negeri, Sumedang, Indonesia
Email: apriyansa.ari@gmail.com

Received: 06 March 2024.
Revised: 18 June 2024.
Accepted: 03 July 2024.
Published: 27 August 2024.

1. Introduction

Air pollution is a significant health hazard, responsible for the premature deaths of approximately 6,100 individuals annually in Jakarta [1]. It is a leading factor in the occurrence of lower respiratory infections, contributing to an estimated 38,000 deaths in Indonesia each year, with 3,000 to 6,000 of these potentially preventable through the improvement of air quality [2]. Moreover, substantial evidence suggests that prolonged exposure to polluted air significantly heightens the risk of contracting severe forms of COVID-19, leading to fatal outcomes [3].

Controlling air pollution is imperative to achieve the 11th Sustainable Development Goal (SDG), which aims to foster inclusive, safe, resilient, and sustainable cities and communities [4].

The surge in energy consumption, coupled with the reliance on petroleum-based energy sources, has exacerbated air pollution. This issue is further compounded by factors such as population growth, lifestyle shifts, and urbanization, which increase the usage of motor vehicles and reduce green spaces, thereby aggravating the air pollution crisis [5].

Air pollution triggers numerous health issues, particularly causing eye irritation. It stems from various sources, including vehicle exhaust emissions due to incomplete combustion, releasing harmful substances such as lead, suspended particulate matter (SPM), nitrogen oxide (NOx), sulfur oxide (SO2), hydrocarbons (HC), carbon monoxide (CO), and other photochemical oxidants [6]. Gas emissions from burning gas (BBG) are comparatively lower, accounting for less than 0.5% [7]. However, vehicle emissions are not the sole contributor to air pollution.

Carbon monoxide (CO), a significant component of exhaust emissions, poses severe health risks due to its ability to react with hemoglobin (Hb) in the blood, impairing the blood's oxygen-carrying capacity. The Occupational Safety and Health Administration (OSHA) stipulates a permissible exposure limit of 50 ppm for CO over an 8-hour period.

In the era of the Internet of Things (IoT), monitoring environmental conditions has become more accessible, with advanced systems employing integrated sensors and IoT-based unmanned aerial vehicles (UAVs) or quadcopters for environmental surveillance [8], [9]. Various sensor-based detectors have been developed for air quality monitoring [10] [11] including the MQ-7 sensor for CO detection, known for its affordability and specific heating requirements [12]. Sakti *et al.* [5], introduced a CO concentration measurement system utilizing modified UAVs for mobile data acquisition, demonstrating the capability to measure CO levels ranging from 20 to 100 ppm. Subsequent enhancements included the integration of the OV5647 camera module, improving the system's functionality within an interactive map [7]. Despite these advancements, there remains a need for further improvements in UAV-based CO detection systems for real-time urban monitoring.

This research proposes the development of an enhanced system capable of mobile monitoring and precise CO concentration measurement using a quadcopter, with field tests conducted in major Indonesian cities. The objective is to devise a solution that not only detects and measures CO levels but also contributes to the broader goal of mitigating air pollution in urban settings.

2. Methodology

This study was conducted in four critical stages, which include: (i) the information gathering phase, involving the review of various related references/literature, analysis of measurement needs, and specifications of the system devices (sensors, controllers, actuators, web, drone); (ii) the planning phase, involving drafting the research plan, setting objectives for each stage, conducting feasibility studies, and creating block diagrams and system schematics; (iii) the development phase, involving the design of initial products, initial testing, and system design evaluation (hardware & software); and (iv) the testing phase, involving testing the system under actual conditions, evaluating the measurement results, and making system improvements (hardware & software).

This study developed a CO Detection and Monitoring System (CODS), beginning with planning the system diagram as shown in Figure 1. The process of measuring CO concentration in the developed system uses an MQ-7 sensor and a GPS module to determine the measurement location. The system also utilizes a Raspberry Pi 3 Model B+ mainboard for processing measurement data, as well as a main cable for all integrated sensors and modules. The results of CODS are stored on a web server and displayed in the form of an interactive map accessible online by the public. Meanwhile, the research developed a mobile detection model using quadcopter for CODS, enabling mobile measurements in places that are dangerous or difficult for humans to reach. This system implements Internet of Things (IoT) technology to access monitoring results anytime, anywhere [5].

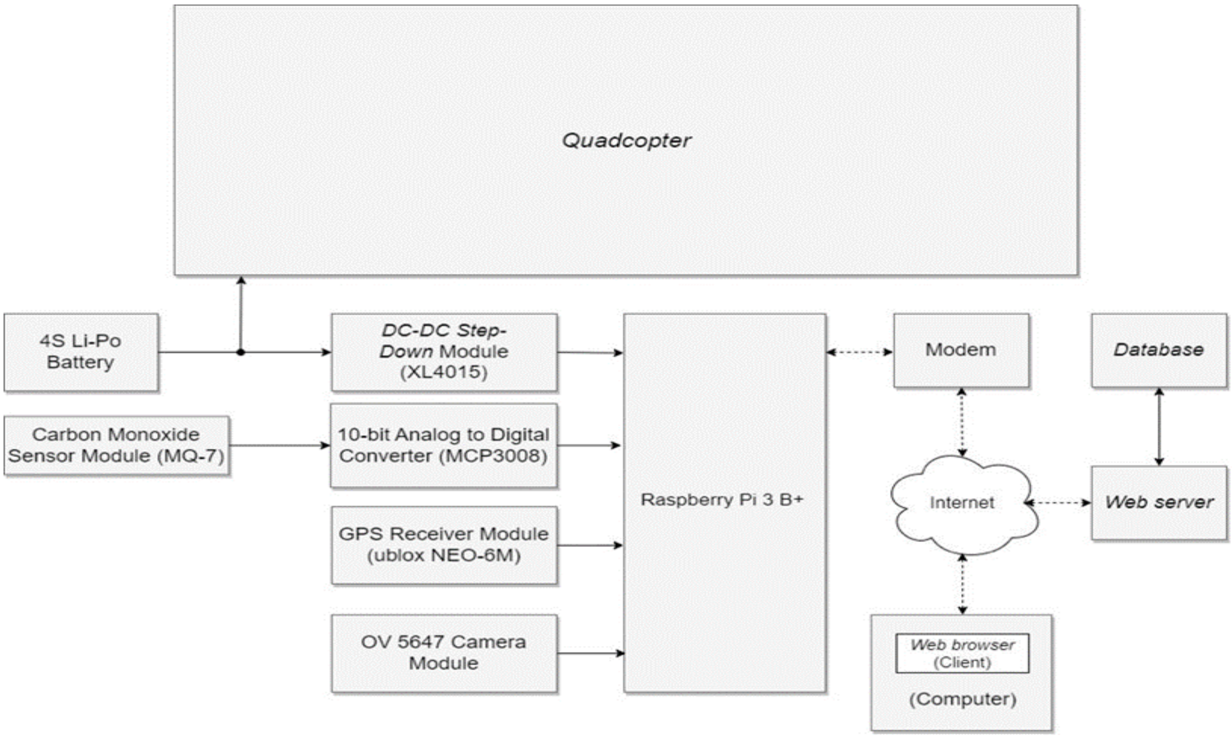


Figure 1. CO detection and monitoring system (CODS) proposed diagram

The design of the MQ-7 sensor module was carried out to comply with the operating principles of the MQ-7 sensor by implementing a switching system on the sensor heater pin using a relay. This circuit controls the voltage changes applied to the sensor heater pin between the heating process (5 V ± 0.1 V) and the detection process (1.4 V ± 0.1 V) on the MQ-7 sensor.

This study utilizes a Raspberry Pi for the integrated mainboard, employing the 3 Model B+ version. The Raspberry Pi 3 Model B+ shield is designed to facilitate the organized and efficient integration of components and modules. Devices integrated into this shield include the u-blox NEO-6M GPS module, a relay, and the 10-bit MCP3008 ADC IC. This shield also features a JST-XH port that connects the shield to the MQ-7 sensor module. The PCB design of the Raspberry Pi 3 shield is shown in Figure 2. Meanwhile, the OV5647 camera module is mounted on the quadcopter. Once all sensors and modules are connected to the Raspberry Pi, the quadcopter will be modified according to the proposed system schematic diagram. Additionally, the final stages of the research include testing and validation.

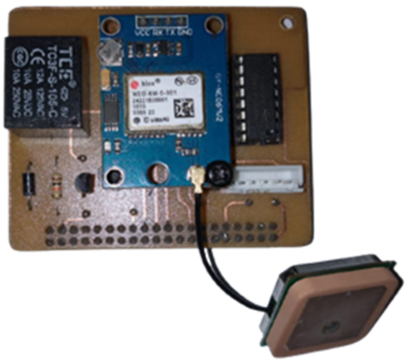


Figure 2. PCB of the Raspberry Pi 3 Model B+

3. Results

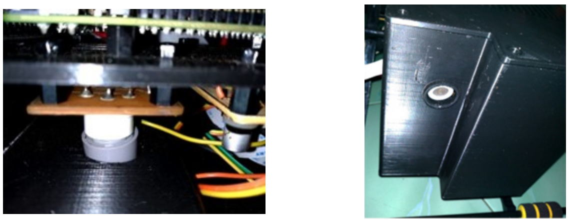
The results consist of the experimental prototype and the testing and validation system. The experimental prototype describes the hardware system built using gas sensors, a camera, GPS, WiFi, a Raspberry Pi, and the construction of a quadcopter. The testing and validation system section explains the measurements conducted on various parts of the prototype and real-time testing at the location.

3.1. Prototype Experiment

The OV5647 camera module is mounted on the front of the quadcopter using a mounting bracket that integrates with the landing gear. The camera is positioned facing downwards, enabling it to capture photographs of the CO gas concentration data collection sites. The actual placement of the OV5647 camera module is shown in Figure 3.

Furthermore, Figure 4 demonstrates that the MQ-7 sensor module has been designed, tested, and then installed on the quadcopter.

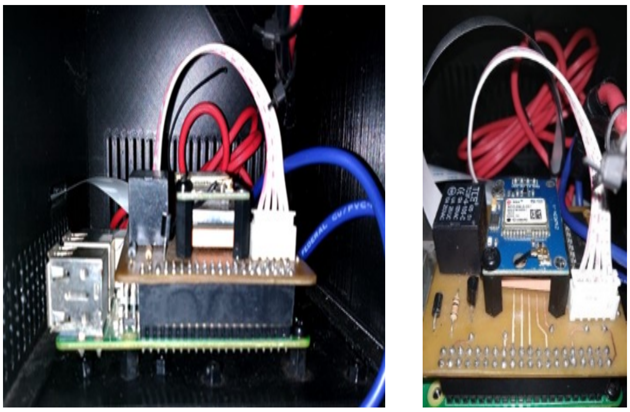
Figure 3. The OV5647 camera module is mounted on the quadcopter



(a) ***(b)***

Figure 4. Placement of the MQ-7 sensor module: (a) quadcopter inner view; (b) quadcopter outer view

Several electronic modules, including the u-blox NEO6-M GPS module, a relay, and the 10-bit MCP3008 ADC IC, are placed on a single PCB. These are combined with the Raspberry Pi as part of the system's processing and control circuit. Their placement is shown in Figure 5.



(a) ***(b)***

Figure 5. Integrated Raspberry Pi experiments: (a) placement of the Raspberry Pi and its shield; (b) installation of the Raspberry Pi shields on the system

To make a POST request to the server for entering data into the database, the Raspberry Pi requires an Internet connection. The experiment utilized a Wi-Fi modem. The Wi-Fi modem is located at the back of the quadcopter, as shown in Figure 6.

Figure 6. Placement of Wi-Fi module on the quadcopter

The electronic components used in this CODS were ultimately integrated into a unified system on the physically modified quadcopter. Modifications were made to enable the quadcopter to carry all hardware components, fly smoothly, and maintain balance. The complete physical appearance of the CODS is further illustrated in Figure 7.

Figure 7. The complete physical appearance of the CODS model

3.2. Testing and Validation System

The radio control for operating the quadcopter utilized the Turnigy TGY-i6S as the transmitter and the Turnigy TGY-iA6C as the receiver mounted on the quadcopter, along with the ground control system (GCS) for operator monitoring station. The GCS employed the Mission Planner software. This software serves as a ground control station for aircraft, helicopters, and rovers, exclusively compatible with the Windows operating system.

It functions both as a configuration utility and dynamic control interface. The GCS represents the Mission Planner software for autonomous vehicles [13]. The primary interface of the Mission Planner software is depicted in Figure 8.

Figure 8. The main view of Mission Planner software

The Quadcopter GCS is wirelessly connected using Telemetry 3DR 433 MHz. A set of telemetry modules comprises separate air and ground modules. The transceiver can utilize UART or USB connections to support both modes. The radio devices enable operators to view real-time data, such as direct GPS positions overlaid on maps, system voltage, directional cues, waypoint navigation, and more. They utilize open-source MAVLink-based ground station software, and calibration on the quadcopter is also conducted prior to flight.

The CO detection results can be accessed on a web page featuring an interactive map. The interface presents 2-dimensional visual objects in two languages (Indonesian and English). Menu options on the website include the homepage, research team, research objectives, research duration, CO detection results and maps, as well as health information (CO impact table within concentration ranges). The web page interface is illustrated in Figure 9.

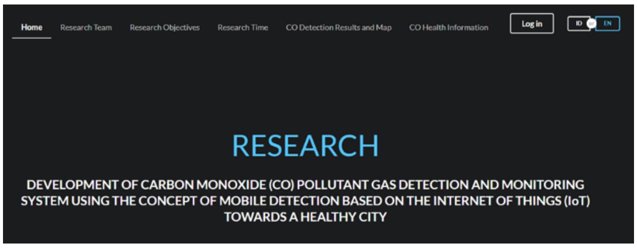


Figure 9. The user-interface CO Detection System (CODS) overview is based on a webpage

The webpage is also accompanied by an interactive map displaying markers with pop-up features containing CO concentration values in parts per million (ppm) and photos of the measurement results. The webpage operates in real-time online as it is connected to the Internet from the CODS quadcopter. The interactive map display for CO concentration mapping is depicted in Figure 10.

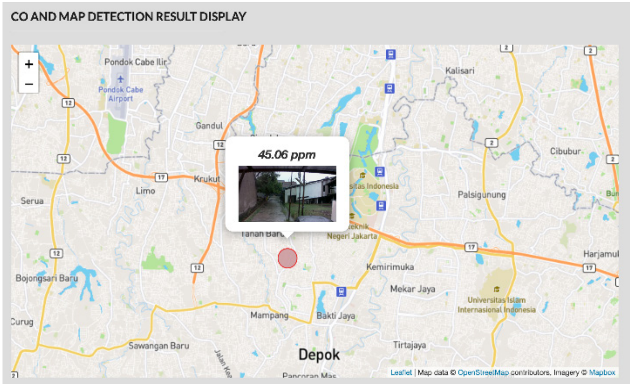


Figure 10. CO measurement location that appears real time on the webpage

Furthermore, Figure 11 shows the CO measurement results at various locations. The table design presents information such as date, time, CO measurement location, latitude and longitude, CO concentration, and detected CO levels. Based on data from the World Health Organization, reference levels for determining CO concentrations have been established in this study, including low levels (below 50 ppm), moderate levels (between 51 and 100 ppm), and high levels (above 100 ppm).

No	Date	Time	Location	Lat & Long	CO Rate	Status
1	07 Nov 2021	16:10:17	Beji, Depok, West Java, Indonesia	-6.38252, 106.80746	36.28	Low Level
2	07 Nov 2021	16:10:05	Beji, Depok, West Java, Indonesia	-6.38252, 106.80746	36.87	Low Level
3	07 Nov 2021	16:09:54	Beji, Depok, West Java, Indonesia	-6.38252, 106.80746	38.4	Low Level
4	07 Nov 2021	16:09:42	Beji, Depok, West Java, Indonesia	-6.38249, 106.80748	39.69	Low Level
5	07 Nov 2021	16:09:31	Beji, Depok, West Java, Indonesia	-6.38249, 106.80748	45.06	Low Level

Figure 11. Results of system measurement on the webpage.

Then, health information in the form of a table detailing the effects of CO within the concentration range of 35-12,800 parts per million displayed on the webpage assists in informing the public about the symptoms experienced by individuals due to CO gas exposure. The table illustrating the effects of CO on health conditions is presented on the webpage as shown in Figure 12.

Carbon Monoxide Concentration	Symptom
35 ppm	Headache and dizziness within six to eight hours of constant exposure.
100 ppm	Slight headache in two to three hours.
200 ppm	Slight headache in two to three hours; lose judgment.
400 ppm	Frontal headache within one to two hours.
800 ppm	Dizziness, nausea, and seizures within 45 minutes; unconscious within 2 hours.
1,600 ppm	Headache, increased heart rate, dizziness, and nausea within 20 minutes; death in less than 2 hours.
3,200 ppm	Headache, dizziness and nausea within 5 to 10 minutes. Death in 30 minutes.
6,400 ppm	Headache and dizziness within one to two minutes. Seizures, stopped breathing, and death in less than 20 minutes.
12,800 ppm	Fainting after 2-3 breaths. Death in less than 3 minutes.

Figure 12. CO effects displayed on the webpage

Before integration into a system, an assessment is conducted on each subsystem, including testing the MQ-7 sensor, testing the u-blox NEO6M GPS receiver module, and testing data transmission to the server. The first step involves evaluating the MQ-7 sensor. According to literature in the form of datasheets, the MQ-7 sensor can be implemented with pre-heating, a process of initial heating before sensor usage to ensure more stable CO gas concentration readings. The initial heating process for the MQ-7 sensor is conducted for a minimum of 48 hours by connecting the heater pin to a 5 V voltage source. The purpose of testing the MQ-7 sensor is to ascertain changes in the output voltage value of the sensor corresponding to the detected CO gas concentration. This testing is performed by introducing mosquito coil smoke containing CO gas into a sealed container fitted with the MQ-7 sensor module and a handheld CO meter. The results of testing the MQ-7 sensor across various CO gas concentration levels are presented in Table 1.

Table 1. The results of the variant of the MQ-7 sensor test on the CO gas concentration

Smart Sensor Handheld CO Meter (ppm)	MQ-7 Sensor Voltage Output (Volts)
20	0.94
30	1.15
40	1.39
50	1.50
60	1.61
70	1.77
80	1.90
90	1.98
100	2.16

Meanwhile, in the operational process, the MQ-7 sensor utilizes a switch system that alternates between heating and sensing processes. During the heating process, a voltage of 5 V ± 0.1 V is applied for 90 seconds, while during the sensing process, a voltage of 1.4 V ± 0.1 V is supplied to the heating pin. The MQ-7 sensor module is operated by measuring the output voltage of the MQ-7 sensor for CO gas concentrations ranging from 20 to 100 ppm. As shown in Table 1, the MQ-7 sensor performs effectively, indicated by the linear relationship between CO gas concentration and the output voltage of the MQ-7 sensor, as depicted in Figure 13.

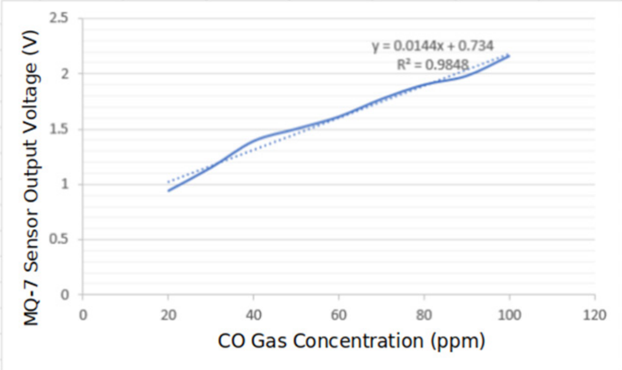


Figure 13. Correlation graph of CO concentration values with the output voltage of the MQ-7 sensor

Subsequently, a calibration process is conducted to determine the percentage of error in MQ-7 sensor readings. This process involves exposing the MQ-7 sensor to pure CO gas. The results of the MQ-7 sensor calibration are presented in Table 2.

Table 2. The results of the MQ-7 sensor test on CO gas detection

Parameter	Pure CO Gas Concentration (ppm)*	MQ-7 Sensor Measurement Result (ppm)	Error Rate (%)
Carbon monoxide (CO)	100	104.87	4.87

**Note: pure CO gas concentration adopted from the intelligent sensor handheld CO meter.*

The calculation of the percentage error value in Table 2 is explained by Equation 1.

$$Error\ (\%) = \left|\frac{100-104{,}87}{100}\right| \times 100\% \ldots\ldots\ldots\ldots(1)$$

Furthermore, a testing process was conducted on the u-blox NEO-6M GPS receiver module. The u-blox NEO-6M GPS module was tested by calculating the difference between the latitude and longitude values received by the u-blox NEO-6M GPS module and those received by the Garmin eTrex 10. The results of the testing of the u-blox NEO-6M GPS receiver module can be seen in Table 3.

Table 3. The comparative test results of the u-blox NEO-6M GPS receiver module

No	u-blox NEO-6M GPS Receiver Position Data		Garmin eTrex 10 Position References		
	Latitude (Lat_1)	Longitude ($Long_1$)	Latitude (Lat_2)	Longitude ($Long_2$)	Difference Distance (m)
1	-6.26235	106.87879	-6.26236	106.87879	1.11
2	-6.26235	106.87879	-6.26236	106.87879	1.11
3	-6.26235	106.87879	-6.26235	106.87879	0.00
4	-6.26235	106.87879	-6.26235	106.87879	0.00
5	-6.26235	106.87879	-6.26236	106.87879	1.11
6	-6.26235	106.87879	-6.26236	106.87879	1.11
7	-6.26235	106.87879	-6.26236	106.87879	1.11
8	-6.26235	106.87879	-6.26236	106.87879	1.11
9	-6.26235	106.87879	-6.26236	106.87879	1.11
10	-6.26235	106.87879	-6.26236	106.87879	1.11

The difference value calculation is obtained using the Euclidean distance method, shown in Equation 2.

$$Distance = \sqrt{(Lat_1 - Lat_2)^2 + (Long_1 - Long_2)^2} \times 111.319\ Km \ldots\ldots\ldots\ldots(2)$$

Equation 2 indicates distance = difference between 2 data (m); Lat1 = Latitude reading from Garmin eTrex 10; Lat2 = Latitude reading from u-blox NEO-6M; Long1 = Longitude reading from Garmin eTrex 10; Long2 = Longitude reading from u-blox NEO-6M; 111.319 km = Constant value for 1 degree of the Earth.

After successful sensor testing and data acquisition, the next step is to transmit the data to the server.

Testing of data transmission to the server by the Raspberry Pi is conducted using the request module with Python. This module enables HTTP request transmission using the POST request method in this study. If the request is successful, the server will respond to the Raspberry Pi. Subsequently, a webpage accessible via the Internet will display the data successfully inserted into the database. Figure 14 illustrates the map and data measurement results displayed on the webpage.

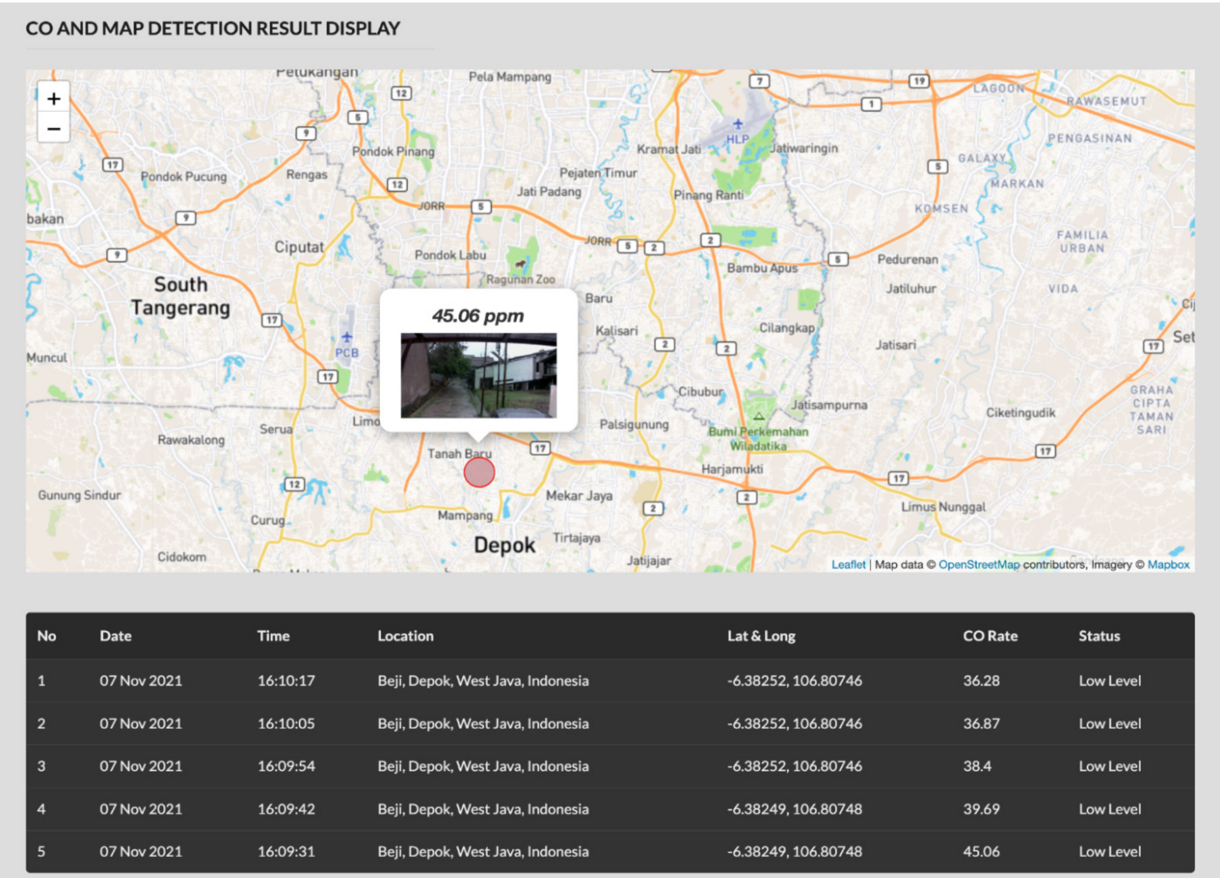


Figure 14. Map and measurement data display on the webpage (Indonesian language)

The final stage involves evaluating the overall system by flying the quadcopter at three locations in Indonesia (Jakarta, Depok, and Bekasi), then observing the CO concentration values, location coordinates, and quadcopter altitude.

This test is conducted at three separate times: morning, afternoon, and night. The overall test results are presented in Table 4.

Table 4. The results of the u-blox NEO-6M GPS receiver module

Location (Indonesian City)	Time	Altitude (m)	CO Gas Concentration (ppm)	Latitude	Longitude
Jakarta	Morning	≤ 5.52	33.62	-6.19367	106.87831
	Noon	≤ 4.54	47.56	-6.19367	106.87828
	Night	≤ 5.01	21.32	-6.19366	106.87826
Bekasi	Morning	≤ 6.32	25.47	-6.23622	106.94686
	Noon	≤ 5.43	27.32	-6.23623	106.94683
	Night	≤ 6.21	30.61	-6.23619	106.94676
Depok	Morning	≤ 4.73	22.13	-6.38252	106.80746
	Noon	≤ 6.54	20.57	6.38249	106.80748
	Night	≤ 4.82	20.03	-6.38249	106.80748

4. Discussion

The MQ-7 sensor was selected based on its cost-effectiveness and long-term usability, up to approximately 5 years. In its deployment, the MQ-series sensor requires pre-heating to ensure consistent readings, a process performed once before sensor usage. The pre-heating of the MQ-7 sensor lasts for a minimum of 48 hours, with the heater pin connected to a 5V voltage source. This pre-heating step is essential for achieving stable readings [12].

The MQ-7 sensor operates under 2 conditions: a 60-second heating process with the heater sensor connected to a 5V voltage source, followed by a 90-second sensing process with the heater sensor pin connected to a voltage of 1.4V. This alternating process necessitated the design of a relay driver module circuit controlled by a program on the Raspberry Pi 3 Model B+. The MQ-7 sensor was tested across CO gas concentration variants by recording its output voltage in the range of 20-100 ppm CO gas concentrations.

To determine sensor percentage error readings, calibration of the MQ-7 sensor was performed using pure CO gas with a concentration of 100 ppm, followed by comparison with sensor readings. In the calibration process, the sensor reading yielded 104.87 ppm. This value indicates a percentage error of 4.87%. Calibration was conducted in a gas analyzer laboratory accredited by the Indonesian Accreditation Committee (Komite Akreditasi Nasional/KAN), an institution responsible for accreditation.

The obtained error value of 4.87% is relatively low compared to similar research by Priyanta, which reported an error value of 9.4% [14].

Based on Table 3, the u-blox NEO-6M GPS module underwent testing to determine its capability in collecting CO gas data locations. The GPS data were transmitted through National Marine Electronics Association (NMEA) strings, including $GPGSV, $GPRMC, $GPGSA, $GPGGL, and $GPVTG [15]. In this research, the $GPRMC NMEA message was processed to extract the required latitude and longitude values [16]. These values were then compared with the latitude and longitude values received by the Garmin eTrex 10 GPS, which has an estimated position error of 3 meters, using the Euclidean distance method. The test involved 10 data collection instances, yielding an average distance difference of 0.89 meters. Similarly, Pamungkas [17] employed the Euclidean distance method in their research, designing an Android application for calculating coordinate distances based on latitude and longitude values.

Additionally, the method of comparing the position accuracy level of the GPS receiver module was conducted by Firdaus and Ismail [17], aiming to assess the accuracy levels of the u-blox NEO-6M and u-blox NEO-M8N GPS receivers. Each GPS receiver was tested against the geographic positions indicated by Google Maps. The test results for the u-blox NEO-6M GPS receiver revealed an average difference (shift) of 1.75 meters.

One component of the CO measurement system integrated with this drone involves the utilization of a web server.

This web server functions as a repository for receiving HTTP requests from the Raspberry Pi, which are then processed by a PHP program hosted on the web server. The processed results are subsequently returned to the Raspberry Pi in response to the request. The term "web server" encompasses both hardware and software components that collaborate to facilitate the overall functionality of the system. From a hardware perspective, a web server typically comprises a computer equipped with web server software and the necessary website file components. Connected to the Internet, the web server enables data exchange with other devices. On the software side, the web server encompasses various modules governing user access to files stored on the computer. The HTTP server, a vital software component, interprets URLs and HTTP requests, enabling access to the web content stored on the server. Users can access HTTP servers through domain names, and the server delivers the requested content accordingly. In the CO measurement system integrated with this quadcopter, the web server hardware employs a virtual private server (VPN), while Nginx is utilized as the software component. In this research, a webpage is loaded onto the browser, presenting CO gas concentration mapping as an interactive map. The measured CO gas concentration values and location data are inserted into respective columns of the database tables configured in phpMyAdmin.

The testing phase was conducted in three distinct metropolitan cities in Indonesia: Jakarta, Bekasi, and Depok. These cities were selected due to their reported high levels of CO pollution, as documented in previous studies [18], [19]. The testing procedure involved flying a quadcopter in loiter mode and measuring carbon monoxide concentrations at specific points. Prior to flight, the quadcopter underwent calibration to ensure accurate sensor readings and device connectivity. Calibration procedures included rectifying any errors that could lead to inaccurate sensor measurements. Measurements were then taken at each location during morning, afternoon, and evening periods. The data obtained from the tests revealed elevated CO gas concentrations at various points in Jakarta, particularly during midday. This heightened concentration is attributed to the extensive use of motorized vehicles, resulting in significant pollution from exhaust emissions. The dense vehicular traffic observed at measurement locations in Jakarta correlates with the elevated levels of CO gas [20], reflecting the city's status as the capital of Indonesia and its dense population of motorcycle users and other forms of public transportation [21], [22].

In affirmation, several advancements from prior research Sakti *et al*. [5] highlighted:

1. System testing was extended to encompass three metropolitan cities (Jakarta, Bekasi, and Depok).
2. The website underwent enhancements, including the incorporation of a location photo menu that enables monitoring on maps. Upon clicking the marker indicating a location, a photo of the corresponding site is displayed alongside the CO concentration value.
3. Server-side development of the system/tool was undertaken. In previous research, the Raspberry Pi served as a server powered by the drone battery. Consequently, when the drone was inactive, the Raspberry Pi server also shut down, rendering the website inaccessible. In this study, the Raspberry Pi functions solely as a CO data processor. Subsequently, the CO data is transmitted to the website server, ensuring continued accessibility even when the drone is inactive.
4. A camera module was integrated into the system. A camera module was affixed to the drone to capture images of locations where CO gas was detected. Consequently, when the Raspberry Pi records CO data, images of the respective locations are presented on the website.

5. Conclusion

The development of a mobile CO detection model utilizing an IoT-based quadcopter addresses the challenge of measuring in hazardous or inaccessible locations, facilitating remote monitoring. The primary component employed for measuring CO gas concentration is the MQ-7 sensor, complemented by the u-blox NEO-6M GPS receiver module for location determination. Processing of measurement data is carried out using the Raspberry Pi 3 Model B+. Subsequently, the processed data is transmitted to a publicly accessible web server, where it is showcased on a webpage featuring an interactive map for public access.

Testing the CO gas detection system at three distinct locations (Jakarta, Bekasi, and Depok) and three different times (morning, afternoon, and evening) revealed varying results. Specifically, CO detection in Jakarta during daylight hours surpassed levels observed in the other two cities. Conversely, CO levels in Bekasi were higher during nighttime testing. However, overall CO concentrations across the three cities remain within acceptable limits (ranging from low to medium levels), as per health standards, indicating a safe environment.

Furthermore, the CO gas detection results are presented comprehensively on the webpage, incorporating data such as date, time, measurement location, latitude, longitude, CO levels, and detected CO levels.

In accordance with information from reputable health sources, this study establishes thresholds for categorizing CO levels: low level (below 50 PPM), medium level (ranging from 51 ppm to 100 ppm), and high level (exceeding 100 ppm).

Acknowledgements

This research was funded by LPPM Universitas Negeri Jakarta (2023). The authors thank all those associated with the Faculty of Engineering of the Universitas Negeri Jakarta laboratories for their support and assistance.

References:

[1]. Greenpeace Indonesia. (2020). *Polusi udara memakan biaya Rp 21 triliun di Jakarta pada tahun 2020.* Greenpeace Indonesia. Retrieved from: https://www.greenpeace.org/indonesia/siaran-pers/5389/polusi-udara-memakan-biaya-rp-21-triliun-di-jakarta-pada-tahun-2020/ [accessed: 16 February 2024]

[2]. BBC News Indonesia. (2020). *Kualitas udara Jakarta selama PSBB membaik, namun "tingkat polutan berbahaya PM 2.5 tetap konsisten."* BBC News Indonesia. Retrieved from: https://www.bbc.com/indonesia/indonesia-52755813[accessed: 16 February 2024].

[3]. Wu, X., Nethery, R. C., Sabath, B. M., Braun, D., & Dominici, F. (2020). Exposure to air pollution and COVID-19 mortality in the United States: A nationwide cross-sectional study. *MedRxiv : The Preprint Server for Health Sciences*. Doi: 10.1101/2020.04.05.20054502

[4]. United Nations. (2020). *Sustainable Cities and Human Settlements.* United Nations. Retrieved from: https://sdgs.un.org/topics/sustainable-cities-and-human-settlements [accessed: 17 February 2024].

[5]. Sakti, I., Yusro, M., & Diamah, A. (2021). A Quadcopter for Monitoring System of Carbon Monoxide Levels Based on Internet of Things. *2021 International Conference on Intelligent Technologies (CONIT),* 1–5. Doi: 10.1109/CONIT51480.2021.9498469

[6]. Ismiyati, I., Marlita, D., & Saidah, D. (2014). Pencemaran Udara Akibat Emisi Gas Buang Kendaraan Bermotor. *Jurnal Manajemen Transportasi & Logistik (JMTransLog), 1*(3). Doi: 10.54324/j.mtl.v1i3.23

[7]. Yusro, M., Diamah, A., Sakti, I., Firmansyah, I., & Budi, S. (2022). Design of carbon monoxide (co) pollutant gas detection and monitoring system using mobile detection model based on internet of things (IoT). In *Conference on Broad Exposure to Science and Technology 2021 (BEST 2021)*, 189-196. Atlantis Press. Doi: 10.2991/aer.k.220131.032

[8]. Pangestu, A., Mohammed, M. N., Al-Zubaidi, S., Bahrain, S. H. K., & Jaenul, A. (2021). An internet of things toward a novel smart helmet for motorcycle. In *AIP conference proceedings*, *2.* AIP Publishing. Doi: 10.1063/5.0037483

[9]. Singh, P. K., & Sharma, A. (2022). An intelligent WSN-UAV-based IoT framework for precision agriculture application. *Computers and Electrical Engineering, 100*, 107912. Doi: 10.1016/j.compeleceng.2022.107912

[10]. Saxena, P., & Shukla, P. (2023). A review on recent developments and advances in environmental gas sensors to monitor toxic gas pollutants. *Environmental Progress & Sustainable Energy, 42*(5). Doi: 10.1002/ep.14126

[11]. Zhang, Y., Zhu, Y., Zeng, Z., Zeng, G., Xiao, R., Wang, Y., Hu, Y., Tang, L., & Feng, C. (2021). Sensors for the environmental pollutant detection: Are we already there? *Coordination Chemistry Reviews, 431,* 213681. Doi: 10.1016/j.ccr.2020.213681

[12]. Kobbekaduwa, N., De Mel, W. R., & Oruthota, P. (2021). Calibration and Implementation of Heat Cycle Requirement of MQ-7 Semiconductor Sensor for Detection of Carbon Monoxide Concentrations. *Advances in Technology, 1*(2). Doi: 10.31357/ait.v1i2.5068

[13]. Ardupilot. (2021). *Mission Planner Overview*. Ardupilot. Retrieved from: https://ardupilot.org/planner/docs/mission-planner-overview.html [accessed: 15 February 2024].

[14]. Priyanta, I. F., Rivai, M., & Dikairono, R. (2016). Pemetaan Distribusi Gas Polutan Menggunakan Quadcopter Berbasis Autonomous Waypoint Navigation. *Jurnal Teknik ITS, 5*(2), 154–159. Doi: 10.12962/j23373539.v5i2.16238

[15]. Mattos, M., Rodrigues, A., Meireles, R., & Aguiar, A. (2023). Vehicle-to-infrastructure IEEE 802.11ad Wi-Fi dataset. *Data in Brief, 47*, 108964. Doi: 10.1016/j.dib.2023.108964

[16]. Elleuch, W., Wali, A., & Alimi, A. M. (2015). An Investigation of Parallel Road Map Inference from Big GPS Traces Data. *Procedia Computer Science, 53,* 131–140. Doi: 10.1016/j.procs.2015.07.287

[17]. Pamungkas, C. A. (2019). Aplikasi Penghitung Jarak Koordinat Berdasarkan Latitude dan Longitude dengan Metode Euclidean Distance dan Metode Haversine. *Jurnal INFORMA Politeknik Indonusa Surakarta, 5*(2). Doi: 10.46808/informa.v5i2.74

[18]. Nugroho, R. T. (2023). Analysis of The Effect of Large Population on Nitrogen Dioxide and Carbon Monoxide Levels in Java Island Using Sentinel-5P. *IOP Conference Series: Earth and Environmental Science, 1127*(1), 012028. Doi: 10.1088/1755-1315/1127/1/012028

[19]. Malik, J. et al. (2022). Study on Wood in Houses as Carbon Storage to Support Climate Stabilisation: Study in Four Residences around Jakarta Municipal City. *Forests, 13*(7), 1016. Doi: 10.3390/f13071016

[20]. Shen, Y., Wu, T., Lian, A., Gao, J., Peng, F., Song, G., Wu, X., Cui, Y., Liu, X., Wan, Y., Yan, J., & Xue, Y. (2024). Dynamic emission characteristics and control strategies of air pollutants from motor vehicles in downtown Beijing, China. *Journal of Environmental Sciences, 136,* 637–646. Doi: 10.1016/j.jes.2023.01.019

[21]. Irawan, M. Z., Belgiawan, P. F., Tarigan, A. K. M., & Wijanarko, F. (2020). To compete or not compete: exploring the relationships between motorcycle-based ride-sourcing, motorcycle taxis, and public transport in the Jakarta metropolitan area. *Transportation, 47*(5), 2367–2389. Doi: 10.1007/s11116-019-10019-5

[22]. Asvial, M., Gita Pandoyo, M. F., & Setyo Arifin, A. (2018). Internet of Things Solution for Motorcycle Riders to Overcome Traffic Jam in Jakarta Using EBkSP. *2018 International Conference on Information and Communication Technology Convergence (ICTC),* 636–638. Doi: 10.1109/ICTC.2018.8539395