DAFTAR PUSTAKA

Amir, N. et al.(2021) "Development of Fall Detection Device Using Accelerometer Sensor," *IEEE 7th International Conference on Smart Instrumentation, Measurement, and Application (ICSIMA)*, doi: 10.1109/ICSIMA50015.2021.9526289.

Akbar, A. Z Zaenudin, Z., Mutaqin, Z. and Samsumar, L. (2022) "IoT-Based Smart Room Using Web Server-Based Esp32 Microcontroller," *Formosa Journal of Computer and information Science*, pp. 79-86, doi: 10.55927/fjcis.v1i2.1241.

Alshammari, H. (2023) "The internet of things healthcare monitoring system based on MQTT protocol,"*Alexandria Engineering Journal*, pp. 275-287, doi: 10.1016/j.aej.2023.01.065.

Al-Ali , A. R., et al. (2024) “An IoT-Based Road Bridge Health Monitoring and Warning System”, *Sensors*, 24, 469. 2024, doi.org/10.3390/s24020469.

Chilamkuri, K., Kone, V. (2020) “Monitoring of Varadhi Road Bridge Using Accelerometer Sensor”, *Materials Today: Proceedings*, vol. 33, pp. 367–371, 2020, doi: 10.1016/j.matpr.2020.04.159.

Chiang, C. (2018), "Design of a CMOS MEMS Accelerometer Used in IoT Devices for Seismic Detection," *in IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 8(3), pp. 566-577, doi: 10.1109/JETCAS.2018. 2825604.

Dhuri, A.et al. (2020) “IoT Based Bridge Health Smart Monitoring System”, *International Journal of Engineering Research and Technolog (IJERT*), 8(11), pp. 109–111, ISSN:2278-0181.

Dinesh, P.B. et al.(2018), “Monitoring of Streng of Bridge Using Accelerometer”, *International Journal of Pure and Applied Mathematics*, 119(18), pp. 1503-1511, ISSN: 1314-3395.

Didik R. et al. (2018) “Aplikasi Single-accelerometer MEMS untuk Mengukur Vibrasi 3-sumbu dan Tilt-Angle 2-axis Secara Bersamaan”, *TELKOMNIKA*.

Elhattab, A., et al. (2019) “Extraction of bridge fundamental frequencies utilizing a smartphone mems accelerometer,” *Sensors (Switzerland)*, 19(14) p. 3143, Jul. 2019, doi: 10.3390/s19143143.

F. Alimuddin, Sulaiman, F., Parinduri, I., Abdullah, R., Firmansyah, T. and Syarif, M. (2021) "Accelerometer Sensor Applications Early Warning System Train Accidents due to Landslide at Laboratory Scale," *IOP Conference Series: Materials Science and Engineering*, doi: 10.1088/1757-899X/180/1/012152.

Gunjal, K.A. et al.(2020) “IoT Based Monitoring And Maintenace of Highway Bridges using Wireless Sensor Network”, *International Research Journal of Engineering and Technology* (*IRJET),* 7 (4), pp. 5619–5622.

Hapsari, A.A. et al. (2021) "Accelerometer Sensor Data Analysis Of Bridge Structural Health Monitoring System," I*OP Conference Series: Materials Science and Engineering*, 2021, doi:10.1088/1757-899X/1108/1/012026.

Hoag, A. et al.(2017) “Measuring displacements of a railroad bridge using DIC and accelerometers”, *Journal Smart Structures and Systems*, 19(2), pp. 225–236, Feb. 2017, doi: 10.12989/sss.2017.19.2.225

Khalifa.D.B. and Martel, M. (2020) “Precision tuning of an accelerometer-based pedometer algorithm for IoT devices,” in *IoTaIS - Proceedings:IEEE International Conference on Internet of Things and Intelligence Systems*, IEEE, Jan. 2021, pp. 116–122. doi: 10.1109/IoTaIS50849.2021.9359700.

Koene, I., Klar, V., Vitala, R. (2020) “IoT Connected device for Vibration Analysis and Measurment”, *HardwareX*, vol. 7, pp. 1–15, Apr. 2020, doi: 10.1016/j.ohx.2020.e00109.

Kusumah, H. et al. (2019) “Application of Internet of Things Penerapan Trainer Interfacing Mikrokontroler dan Internet of Things Berbsasis ESP32”. *Jurnal Cerita*, 5(2).

Kashyap, M., Sharma, V., Gupta, N. (2018) “Taking MQTT and NodeMcu to IoT: Communication in Internet of Things”, *Elsevier, Procedia Computer Science* 132, pp. 1611–1618, doi: 10.1016/j.procs.2018.05.126.

Komarizadehasl, S., et al.(2022) "A Novel Wireless Low-Cost Inclinometer Made from Combining the Measurements of Multiple MEMS Gyroscopes and Accelerometers," *Sensor*, doi: 10.3390/s22155605.

Liang,L., et al. (2023) “Self-powered active vibration sensor by peak–valley data processing independent of the environment toward structural health monitoring”, *Next Nanotechnology*, vol. 117, 108935, doi.org/10.1016/ j.nanoen.2023.108935.

Lee, W., Kim, H., Lee, C. and Kim, S. (2022) "Development of Digital Device Using ZigBee for Environmental Monitoring in Underground Mines," *Applied Science*, doi: 10.3390/app122311927

Muddala, S.K.P. et al. (2019) “IoT Based Bridge Monitoring System”, *International Journal of Advanced Research, Ideas an Innovations in Technology (IJARIIT)*, 5(2), pp. 2044–2047

Nababan, P.H.A.,(2008) “Structural Health Monitoring System Alat Bantu

Mempertahankan Usia Teknis Jembatan". Prosiding Seminar Construction and maintenance of main span Suramadu Bridge.

Nugraha Y. P. et al.(2014) "Pemantauan Kemiringan Gedung dan Bangunan Fisik dengan menggunakan Sensor Akselerometer ADXL-335", *Prosiding pertemuan ilmiah xxviii HFI, Jateng dan DIY.*

Ould, S. and Bennett, N. (2021) "Energy Performance Analysis and Modelling of LoRa Prototyping Boards,"*Sensor*, doi: 10.3390/s21237992.

Phuong, D. N. M. et al. (2023) "An IoT DAQ with Piezoelectric Sensor for Bridge Structure Vibration Measurement", *International Conference on System Science and Engineering (ICSSE)*, pp.51-54, DOI: 10.1109/ ICSSE58758.2023.10227152.

Riyadi, M. et al. (2010) "Pendeteksi Posisi menggunakan sensor Accelerometer MMA7260 berbasis Mikrokontroler Atmega 32", *Jurnal undip.ac.id*.

Santoso, D.R., S. Maryanto, S., and Nadhir, A "Application of single MEMS-accelerometer to measure 3-axis vibrations and 2-axis tilt-angle simultaneously," *Telkomnika (Telecommunication Computing Electronics and Control)*, 13(2), pp. 442–450, doi: 10.12928/TELKOMNIKA.v13i2. 1490.

Saidin, S. , et al. (2023) "Vibration-based approach for structural health monitoring of ultra-high-performance concrete bridge", *Case Studies in Construction Materials 18*, e01752, doi.org/10.1016/j.cscm.2022.e01752.

Syaryadhi, M. et al. (2015) "Penggunaan Accelerometer MMA7361 sebagai Alternatif Pengukuran Lendutan pada Jembatan Secara Nirkabel BerbasisATmega 32", *Jurnal Rekayasa Elektrika,* 11(5).

Sun, F. et al. (2021) "Accelerometer-Based Key Generation and Distribution Method for Wearable IoT Devices," *in IEEE Internet of Things Journal*, 8(3), 2021-Feb, pp. 1636-1650, doi: 10.1109/JIOT.2020.3014646.

Sun, H., Zhang, Z., Li, P. (2021) "Design and Analysis of Bridge Inspection System Based on Wireless Communication and Internet of Things Technology", ," *Wireless Communications and Mobile Computing*, pp. 1–5, Oct. 2021, doi: 10.1155/2021/3410844.

Syufrijal Syufrijal, S. (2020) "Prototipe Sistem Pengukuran Jarak dan Kemiringan Otomatis Menggunakan Mikrokontroler Berbasis Internet of Things (IoT)", *Jurnal Otomasi, Kendali, dan Aplikasi Industri*.

Sahifa, A. A. et al. (2020) "Pengiriman Data Berbasis Internet of Things untuk Monitoring Sistem Hemodialisis Secara Jarak Jauh", *Jurnal Teknik ITS*, 9(2), ISSN: 2337-3539 (2301-9271 Print).

Tian, L. et al. (2021) "Full-Field Bridge Deflection Monitoring with Off-Axis Digital Image Correlation," *Sensor,* doi: 10.3390/s21155058.

Umar, N. et al. (2016) “Aplikasi Jaringan Sensor Nirkabel Untuk Sistem Monitoring Deformasi Konstruksi Jembatan”, *SNTEI.*

Yanziah, A. et al (2020) “Analisis Jarak Jangkau LoRa dengan Parameter RSSI dan Packet loss pada area Urban”, *Jurnal Teknologi TECHNOSCIENTIA* ISSN: 1979-8415, 13 (1), E-ISSN: 2714-8025 59.

Yu, E., Xu, G., Han, Y., Hu, P., Townsend, J.F., Li, Y. (2022) “Bridge vibration under complex wind field and corresponding measurements: A review”, *Journal of Traffic and Transportation Engineering (English Edition)***,** 9(3), pp. 339-362, doi.org/10.1016/j.jtte.2021.12.001.

Zhang,L. et al. (2018) "Application of Internet of Things Technology and Convolutional Neural Network Model in Bridge Crack Detection," *IEEE* 6, pp. 39442-39451, doi: 10.1109/ACCESS.2018.2855144.

Zhou, Q. et al. (2019) “Desain dan Implementasi Open LoRa untuk IoT”, *IEEE Access*, vol. 7, pp. 100649–100657, doi: 10.1109/ACCESS.2019.2930243.

Zhou, K. et al. (2023) “Unmanned aerial vehicle-based computer vision for structural vibration measurement and condition assessment: A concise survey”, *Journal of Infrastructure Intelligence and Resilience 2,* 10031, doi.org/10.1016/j.iintel.2023.100031.

L A M P I R A N

Lampiran 1 Publikasi Karya Ilmiah 1,2,3

Internet of things based on bridge slope detection FREE

Nuraeni Umar; Syafruddin Syarif; Dewiani; Merna Baharuddin

AIP Conf. Proc. 2952, 060015 (2024)
https://doi.org/10.1063/5.0211931

Abstract | View article | PDF

Defining key features of smart-dry model: Bibliometric and expert judgement analysis FREE

Puji Rahayu; Inna Sabily Karima

AIP Conf. Proc. 2952, 060016 (2024)
https://doi.org/10.1063/5.0212010

Abstract | View article | PDF

Mobile location based service application for Android based agricultural machinery data

CERTIFICATE

ICCSET 2022

NO.153/ICCSET/FT/UMK/XI/2022

Kampus Merdeka
INDONESIA JAYA

This to certify that

NURAENI UMAR

who has attended the conference as

Presenter

International Conference on Computer Science and Engineering Technology 2022

"The Role of Industry 5.0, Innovation and Infrastructure for SDGs."

Faculty of Engineering - Universitas Muria Kudus, 26 November 2022

De[illegible] of Engineering
[illegible]udus

M[illegible]n, S.T., M.T.

Chief Committee of ICCSET 2022

Tri Listyorini, S.Kom., M.Kom

AIP Conference Proceedings

RESEARCH ARTICLE | JULY 23 2024

Internet of things based on bridge slope detection

Nuraeni Umar; Syafruddin Syarif; Dewiani; Merna Baharuddin

Check for updates

AIP Conf. Proc. 2952, 060015 (2024)
https://doi.org/10.1063/5.0211931

Internet of Things Based on Bridge Slope Detection

Nuraeni Umar [1, 2, a)], Syafruddin Syarif [1, b)], Dewiani [1, c)] and Merna Baharuddin [1, d)]

[1]Department of Electrical Engineering, Hasanuddin University, Gowa, Indonesia
[2]Department of Electrical Engineering Ujung Pandang State Polytechnic
Makassar, Makassar, Indonesia

[a)]Corresponding author : umarn19d@student.unhas.ac.id,
[b)]syafruddin.s@eng.unhas.ac.id, [c)]dewiani@unhas.ac.id, [d)]merna@unhas.ac.id

Abstract. The slope of the bridge is affected by stress, strain, deflection, temperature, and time-dependent properties such as creep and shrinkage in the structure, resulting in changes in the slope of the bridge. This research creates a system that continuously sends data to a wireless system and detects conditions concerning changes that happen on a specific bridge when pitch varies from the typical state. The accelerometer will measure its acceleration directly when moving horizontally and is placed on the Earth's surface to detect an acceleration of the Earth's gravity at its vertical point for acceleration caused by horizontal movement. This research detects the slope of the bridge with the internet of things. The results of this study show that the accelerometer sensor system reads changes in slope, which then sends the data to the network and is received on a cellphone or computer. It uses the Message Queuing Telemetry Transport protocol with a simple and lightweight publish model and is designed for devices with limited capabilities and small bandwidth, high latency, or networks with low bandwidth. The results of this study prove that the system has succeeded in detecting changes in angle from 0° to 44.03°. The differences between accelerometer and protractor measurements ranged from 0.03 percent to 4.8 percent. The Parameters measured in this study were that the output of this system was [93-96] bps, QoS was 0%, and the delay was [1.54–1.46] seconds. And the results obtained prove that this system has excellent performance.

Keywords: Wireless Sensor, Bridge, Mqtt, Iot

INTRODUCTION

For the purpose of early detection of a bridge's slope, this research is a system development that creates a system that integrates an accelerometer sensor with the internet network. As long as the internet network is accessible, this technology transmits data continually and is not geographically constrained. The benefit of this research is to detect the movement of the bridge slope. Knowing this earlier will help the repair process so that it can improve and maintain the strength and safety of bridge users. The bridge can change its strength due to wind speed, temperature, the weight of the load, and others, and for that, the sensor system informs the data of these changes. [1,2,3]. An accelerometer is a device that is widely used for bridge damage detection applications and for monitoring bridges [2,4],[5,6]. Accelerometer, Micro-Electromechanical- For systems connected to the Internet of Things, this versatile wireless sensor unit can be used to measure a variety of different movements [7,8],[9]. The accelerometer can also be used to measure vibrations that occur in vehicles, buildings, and machines, and can also be used to measure vibrations that occur in the earth, engine vibrations, dynamic distances, and speeds with or without the influence of the earth's gravity [8,10]. To determine changes in the slope of the bridge, a detection system for changes in the slope of the bridge is needed with telemetry technology. IoT. Wireless Sensor Networks, wireless communication systems, and computer networks, IoT.development of a sensor system that sends data to the internet network. Studying the bridge detection system will integrate sensor sensing and IoT, the internet, remote communication, digital signal analysis and processing, big data knowledge mining, big data prediction, and other technologies, design, and analysis of the main structure of roads and bridges, and build a professional intelligent digital network based on bridge inspection data

Pancasakti International Conferences Engineering and Computer Science 2022
AIP Conf. Proc. 2952, 060015-1–060015-7; https://doi.org/10.1063/5.0211931
Published under an exclusive license by AIP Publishing. 978-0-7354-5007-3/$30.00

collection, monitoring, analysis, evaluation, and early warning [6]. Message Queuing Telemetry Transport Protocol plays an important role in exchanging data or information between devices in the IoT without knowing each other's identity [11]. Research related to the MPU 6050 accelerometer sensor system, ESP32, internet of things communication system, and telecommunication system performance can be known by measuring throughput, delay, and QoS parameters to obtain telecommunication system performance.

RELATED WORKS

Research related to telecommunications systems, and Wireless Sensor Networks, Wireless Sensor network. At this time wireless sensor networks are growing very rapidly, this is because wireless sensor networks have very wide applications to be implemented in various fields of life. A wireless sensor network consists of a number of sensors referred to as sensor fields. The spread of this sensor can be done randomly or follow a pattern [star, mesh, cluster tree]. Sensors are equipped with memory and communication equipment. The sensor detection system performs data processing to make temporary decisions and sends the decision results to the data processing center or fusion center. The output of the sensor is a binary number, '0' if there is no target, and '1' if there is a target. The Fusion center will make the decision. An accelerometer sensor is a transducer that functions to measure acceleration, detect and measure vibrations, to measure the acceleration due to Earth's gravity. The working principle of this transducer is based on the physical law that when a conductor is moved through a magnetic field, or if a magnetic field is moved through a conductor, an induced voltage will arise in the conductor. The type used is the MPU 6050 accelerometer sensor. Arduino IDE [Integrated Development Environment], and Software IDE [Integrated Development Environment] consist of three parts, including Program Editor, Program Listing, and Uploader. MQTT protocol is a message exchange protocol with a simple and lightweight publish/subscribe model and is designed for devices that have limited capabilities and small bandwidth, high latency. The design principle of MQTT is to minimize network bandwidth and device resource requirements while still ensuring reliability and some level of assurance that a message will be delivered. This principle makes this protocol ideal for application to machine-to-machine [M2M] or Internet of Things communications and for mobile applications where bandwidth and battery capacity are limited [11]. Wearable health monitoring systems require strict medical definitions and criteria as they work under ergonomics and hardware limitations. Primarily these constraints include data rate, delay, QoS, and power consumption, with the goal of maximum data transmission reliability being achieved [12].

Internet of Things [IoT] technology is a concept where an object has the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction. Today's information technology is highly dependent on data. IoT is a global network infrastructure, which connects physical and virtual objects through the exploitation of data capture and communication capabilities. The infrastructure consists of the existing network and the internet and its network development. All of these will offer object identification, sensor, and connection capabilities as the basis for the development of independent cooperative applications and services. It is also characterized by a high degree of autonomous data capture, event transfer, network connectivity, and interoperability. A microcontroller system is a chip that has a central processing unit and various other components such as clock and reset, memory [both program and data], and input/output [serial and parallel] which are built into one IC [Integrated Circuit]. The combination of several components in one chip makes the microprocessor system more compact, practical, and efficient. There are many choices of microcontrollers that have integrated network modules so that they can be directly connected to the internet that they can be implemented for IoT/WSN-based equipment, for example, NodeMCU ESP32. ESP32 is the name of the microcontroller, ESP32 offers a standalone WiFi network solution as a microcontroller link to a WiFi network. The ESP32 uses a dual-core processor running on Xtensa LX16 instructions. Table 1. shows the ESP32 specifications.

TABLE 1. ESP32. Specifications

No	Attributes	Details
1	Voltage	3.3 Volts
2	Processor	Tensilica L108 32 bit
3	Processor Speed	Dual 160MHz
4	RAM	520 K
5	GPIO	34
6	ADC	7
7	Support 802.11	11b/g/n/e/i
8	Bluetooth	BLE [Bluetooth Low Energy]
9	SPI	3
10	I2C	2
11	UART	3

Research instruments: The specifications for the requirements are as follows: Table 2. shows the hardware and specifications used. Software in table 3. shows the software and specifications used.

TABLE 2. Hardware Table

No	Hardware	Description
1	Laptops	Processor: Intel® Core™ i7-7700HQ, CPU @ 2.80GHz [8 CPUs]. Memory: 8192 MB RAM. HDD: 1TB.
2	AccelerometerSensor	Digital interface [1 I/O pin with TTL level]
3	Microcontroller ESP32	CPU: @ 80 MHz [default] or 160 MHz Memory: 32 KB instructions, 80 KB user data Wi-Fi : 80.11 b/g/n standard
4	*Power Supply*	5V, 1000mA

TABLE 3. Software

No	Software
1	Windows 10 64-bit operating system
2	Arduino IDE
3	MQTT Lens

This research will produce a bridge slope detection system based on the internet of things, the data obtained is the sensor detects changes in the slope of the bridge in degrees, this data is sent to the internet and will be received by computers and cellphones. This telecommunications system will have performance, to determine the performance of the telecommunications system from bridge slope detection, measurements are carried out, namely parameter measurementsThroughput [bps], Delay [second], and Packet Loss [%].

SYSTEM DESIGN

A waterfall method is an approach to systematic and sequential software development that starts from the level of system progress throughout the analysis, design, program code, testing, and maintenance. Develop an IoT-based bridge slope detection system. The benefit is knowing the movement of the bridge slope from the initial conditions. The need for a bridge slope change detection system, so that traffic safety on the bridge can be known. This system is very dependent on the internet network. This system will measure the change in angle on the bridge. System design. Figure 1. Is a block diagram of an IoT-based bridge slope detection system. And in Figure 2. Displays the flowchart of the IoT-based bridge slope detection system. And in Figure 3. Showing schematic IoT Based on bridge slope detection systems.

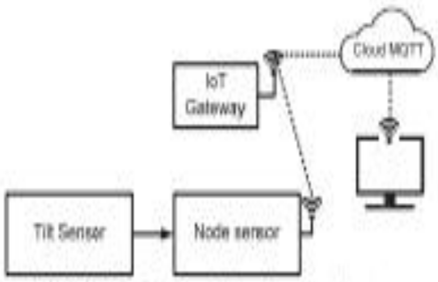


FIGURE 1. IoT Based on bridge slope detection system

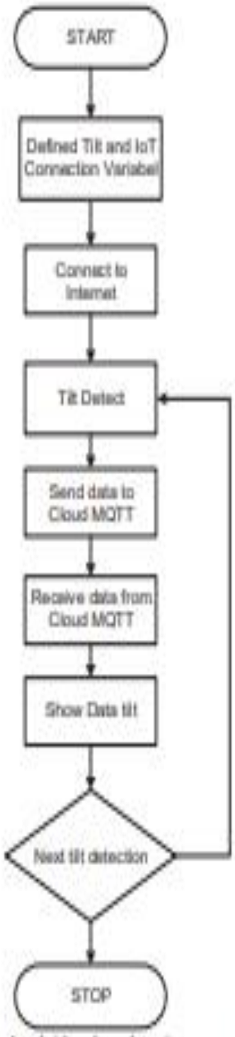


FIGURE 2. IoT Based on bridge slope detection systems flow chart.

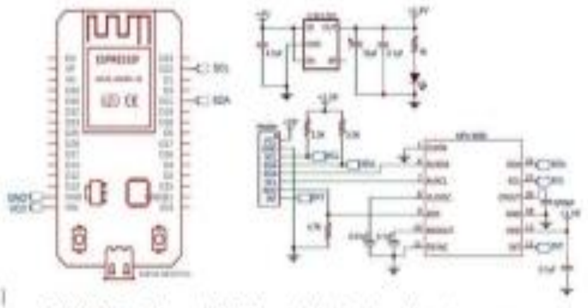


FIGURE 3. Schematic IoT Based on bridge slope detection systems

To see the performance of the telecommunications system, parameter measurements can be carried out, Throughput is the measured network performance [bps], and Delay [seconds] is the time required for the transmission of data packets from the sender to the receiver Packet Loss [%]. packet loss is the number of packets lost during the transmission process from source to destination. This telecommunication system demands maximum reliability within a certain delay limit. Primarily these constraints include data rate, delay, QoS, and power consumption, with the goal of maximum data transmission reliability being achieved.

RESULTS AND DISCUSSION

From the design of the bridge slope detection system, it produces a bridge slope detection device based on the internet of things. This system performs bridge slope detection measurements, as well as measures throughput, QoS, and Delay. Figure 4. Showing is the result of the design is an IoT-based tilt detection tool.

Figure 4. IoT Systems.

From the results of measuring the IoT-based bridge slope detection system, the data obtained throughput, QoS, and system delay, data are in table 4.

TABLE 4. Table of Measurement of Throughput, QoS, Delay

Data from measurement: IoT Based on bridge slope detection systems	**Throughput**	**QoS**	**Delay**	**Category latency**	**Category Throughput**
	93 bps	0%	1.54 second	Good	Excellent
	94 bps	0%	1.52 second	Good	Excellent
	96 bps	0%	1.46 second	Excellent	Excellent

From the results of measuring the IoT-based bridge slope detection system, the data obt4ined throughput, QoS, and system delay, data are in table 5.

TABLE 5. Table of Slope Measurements

No.	Tilt Angle		
	Accelerometer Sensor	Angle Meter	Error [%]
1	0°	0°	0
2	4.77°	5°	4.8
3	9.90°	10°	1.0
4	14.84°	15°	1.0
5	20.01°	20°	0.05
6	24.99°	25°	0.04
7	30.09°	30°	0.3
8	35.01°	35°	0.028
9	39.83°	40°	0.43
10	44.03°	45°	2.2

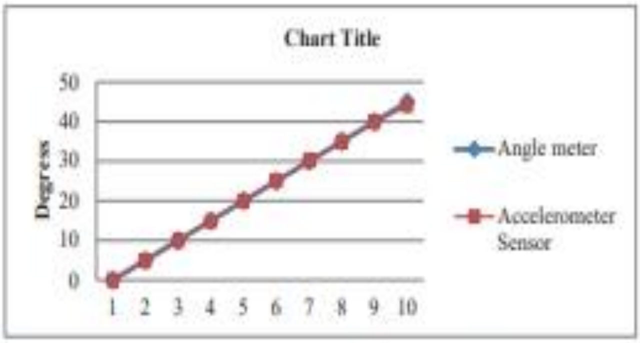


FIGURE 5. Degrees comparison of different angle meters and accelerometer sensors

CONCLUSION

This study produced a technique for detecting bridge slopes. The technology can recognize changes in the bridge's slope because data is supplied continuously. This study develops a bridge slope detection system that is interconnected with the internet of things; the system transmits a change in angle from 0° to 44.03° to the IoT and receives it through a mobile device. Measurements from the protractor and the accelerometer varied by 0.03 to 4.8 percent. PrIn producing system, information from sensors is transmitted online and then received by a computer or smartphone. The parameters measured in this study were that the output of this system was [93-96] bps, QoS was 0%, and the delay was [1.46-1.54] seconds. The results obtained prove that this system has excellent performance.

REFERENCES

1. Dhuri A, Jogale P, Kadam S, Kawade L, Kore S. IOT based Bridge Health Monitoring System. Int J Res Appl Sci Eng Technol. 2020;8[11]:811–6.
2. Dinesh Pb, Raja TJNLDPCK. Monitoring of Streng of Bridge Using Accelerometer. Int J Pure Appl Math. 119.
3. Muddala D, Kamble D, Nimbalkar P, Patil R, Penchala SK. IOT based Bridge Health Monitoring System. Int J Res Appl Sci Eng Technol. 2020;5[2]:811–6.
4. Chilamkuri K, Kone V. Monitoring of varadhi road bridge using accelerometer sensor. In: Materials Today: Proceedings. 2020. p. 367–71.
5. Elhattab A, Uddin N, OBrien E. Extraction of bridge fundamental frequencies utilizing a smartphone mems accelerometer. Sensors [Switzerland]. 2019;19[14].
6. Sun H, Zhang Z, Li P. Design and Analysis of Bridge Inspection System Based on Wireless Communication and Internet of Things Technology. Wirel Commun Mob Comput. 2021;2021.

7. Koene I, Klar V, Vitala R. IoT Connected device for Vibration Analysis and Measurment. J Elsevier, GNU GPL v3.
8. Gunjal KA, Gunjal PD, Gunjal TC. IoT Based Monitoring And Maintenace of Highway Bridges using Wireless Sensor Network.
9. Zhang L, Zhou G, Han Y, Lin H, Wu Y. Application of Internet of Things Technology and Convolutional Neural Network Model in Bridge Crack Detection. IEEE Access. 6:39442–39451,.
10. Hoag A, Hoult NA, Take WA, Moreu F, Le H, Tolikonda V. Measuring displacements of a railroad bridge using DIC and accelerometers. J Smart Struct Syst. 19[2].
11. Kashyap M, Sharma V, Gupta N, MQTT T, IoT N. Communication in Internet of Things, Elsevier, Procedia Computer Science 132 [2018] 1611–1618, 1877-0509 © 2018 The Authors. Published by Elsevier Ltd;
12. Akbar MS, Yu H, Cang S, Delay. Reability, and Throughput Based QoS Profile: A MAC Layer Performance Optimization Mechanism for Biomedical Applications in Wireless Body Area Sensor Networks, Hindawi. J Sens [Internet]. 2016, Arti:17. Available from: http://dx.doi.org/10.1155/2016/7170943,

Lampiran 2 Publikasi karya ilmiah 1,2,3

[IJRES] Editor Decision: ACCEPTED - a Scopus indexed journal (SJR Q3)

Eksternal
Kotak Masuk

A

Selvakumar Manickam lewat kirimdata.id

19 Apr 2024, 09.00 (10 hari yang lalu)

kepada saya

--Paper ID # 21248

Dear Prof/Dr/Mr/Mrs: Eni Nuraeni Umar,

Your article titled "Internet of Things and Long-Range Based on Bridge Slope Early Detection Systems" is accepted for publication in the International Journal of Reconfigurable and Embedded Systems (IJRES), p-ISSN 2089-4864, e-ISSN 2722-2608, a peer-reviewed and open access journal distributed under the terms and conditions of the Creative Commons Attribution-ShareAlike 4.0 International License, and indexed by Scopus (https://www.scopus.com/sourceid/21101061447). This journal has been included in the SCImago Journal Rank (SJR, https://www.scimagojr.com/journalsearch.php?q=21100901206&tip=sid&clean=0)
and has an SJR of 0.218 (Q3). Congratulations!

Your cooperation is very appreciated.

Thank you

Best Regards,
Prof. Dr. Ir. Tole Sutikno, ASEAN Eng.
Editor-in-Chief, International Journal of Reconfigurable and Embedded Systems
https://www.scopus.com/sourceid/21101061447
ijres@iaesjournal.com

International Journal of Reconfigurable and Embedded Systems (IJRES)
http://ijres.iaescore.com

[IJRES] Keputusan Redaksi : DITERIMA - jurnal terindeks Scopus (SJR Q3)
Eksternal
Kotak Masuk
Selvakumar Manickam lewat kirimdata.id
19 Apr 2024, 09.00 (10 hari yang lalu)
kepada saya
--Kertas ID #21248
Yang Terhormat Prof/Dr/Bapak/Ibu : Eni Nuraeni Umar,
Artikel Anda berjudul "Internet of Things dan Jangka Panjang Berbasis Bridge Sistem Deteksi Dini Lereng" diterima untuk dipublikasikan di Jurnal Internasional Sistem yang Dapat Dikonfigurasi Ulang dan Tertanam (IJRES), p-ISSN 2089-4864, e-ISSN 2722-2608, jurnal peer-review dan akses terbuka didistribusikan berdasarkan syarat dan ketentuan Creative Commons Lisensi Internasional Attribution-ShareAlike 4.0, dan diindeks oleh Scopus (https://www.scopus.com/sourceid/21101061447).
Jurnal ini telah termasuk dalam SCImago Journal Rank (SJR, https://www.scimagojr.com/journalsearch.php?q=21100901206&tip=sid&clean=0
) dan memiliki SJR sebesar 0,218 (Q3). Selamat!
Kerja sama Anda sangat kami hargai.
Terima kasih
Salam,
Prof.Dr.Ir. Tole Sutikno, ASEAN Eng.
Pemimpin Redaksi, Jurnal Internasional yang Dapat Dikonfigurasi Ulang dan Tertanam
Sistem https://www.scopus.com/sourceid/21101061447
ijres@iaesjournal.com

Jurnal Internasional Sistem yang Dapat Dikonfigurasi Ulang dan Tertanam (IJRES)
http://ijres.iaescore.com

Intelektual Pustaka
Media Utama

iaes
Institute of Advanced Engineering and Science

Scopus® indexed
CiteScoreTracker 2024: 1.3 | h-index: 7
(Last updated on 05 May, 2024)

SJR SCImago Journal & Country Rank
Q4 Hardware and Architecture

SJR 2023	SNIP 2023	CiteScore 2023
0.159	0.374	1.5

CERTIFICATE

No. 21248/IJRES/A/7/2024

International Journal of Reconfigurable and Embedded Systems

is hereby awarding this certificate to

Nuraeni Umar, Syafruddin Syarif, Dewiani Dewiani, Merna Baharuddin

as **Authors** for paper entitled:

Internet of things and long range-based bridge slope early detection systems

for Vol 13, No 3: November 2024

July 19, 2024

Prof. Dr. Paolo Crippa
Editor-in-Chief

ISSN 2089-4864
http://ijres.iaescore.com

Internet of things and long range-based bridge slope early detection systems

Nuraeni Umar[1,2], Syafruddin Syarif[1], Dewiani[1], Merna Baharuddin[1]
[1]Department of Electrical Engineering, Faculty of Engineering, Hasanuddin University, Gowa, Indonesia
[2]Department of Electrical Engineering, Faculty of Telecommunication, Ujung Pandang State Polytechnic, Makassar, Indonesia

Article Info

Article history:

Received Jan 3, 2024
Revised Mar 21, 2024
Accepted Apr 19, 2024

Keywords:

Accelerometer
Bridge
ESP32 LoRa
Internet of things
Slope

ABSTRACT

This research proposes an internet of things (IoT) and long range (LoRa)-based bridge slope status monitoring and warning system that is wireless, low-cost, and user-friendly, with continuous data sent. Bridge inspection officers can easily obtain bridge slope data via a web browser on a cell phone. The design uses Arduino integrated development environment (IDE) software and an ITGMPU accelerometer sensors, TTGO ESP32, cellphones, successfully identified tilt angle variations from 0.11° to 15.2° were the research's outputs, and and they were continuously transmitted to the bridge inspection officer's mobile phone. Measurements of throughput, quality of service (Packet loss), and latency characteristics have been made to assess the internet network's performance. The network system performance statistics show an average measured network delay of 1.2 mseconds, a throughput of 85 bps, and a Packet loss of 0%. Consequently, the system performs well and the internet network performance falls into the very good range.

Corresponding Author:

Nuraeni Umar
Department of Electrical Engineering, Faculty of Telecommunication
Ujung Pandang State Polytechnic
St. P. Kemerdekaan KM. 10 Tamalanrea, Makassar, Indonesia
Email: nuraeni_umar@poliupg.ac.id

1. INTRODUCTION

Bridges play a crucial function in reducing travel distances in land transportation. Indonesia is made up of islands connected by bridges, as well as rivers that need flyovers and bridges. Even though the infrastructure is designed to last for a very long time, damage to bridge structures typically develops at an uncontrollable rate. Excessive variations in a bridge's slope might cause damage to the structure. In order to allow for logical action to be made regarding the bridge, bridge inspection officials constantly visit and evaluate the bridge's condition. The issue is that because bridges are dispersed so far, it takes thousands of kilometers to reach a bridge for monitoring. By overcoming this and creating a bridge slope early detection system, data may be quickly accessed through a mobile device. Through the use of a web browser, the bridge slope early detection system continuously transmits data remotely to the officer's mobile phone. This is important to know since you can calculate the angle change using this tool.

The system makes use of a message queuing (MQTT) computer network or internet network, long range (LoRa) and the internet of things (IoT). Telemetry transport as data transmission to mobile phones with this method, inspection officials can do so quickly and affordably without having to travel to the bridge's location. The performance of the internet network, including throughput, quality of service (Packet loss), and latency characteristics, is monitored to assess the network's quality. Investigate the creation of a system that combines accelerometer sensors with LoRa and IoT technologies to detect a bridge's slope initially. To monitor vibrations, an accelerometer sensor is mounted on the bridge body [1], and computing the bridge's vertical deflection [2]. The three measurement axes of the MMA7361L accelerometer sensor type are the x, y, and z axes. The aim of this accelerometer is to measure an object's acceleration in gravitational units (g).

Radio is a wireless or wireless communication system. This technology is not limited by cable length. Radio is a simple communication medium with long distance reach, the function of radio media is to spread messages, information, music in all directions over long distances and then be received by radio receivers. Radio transmits with electromagnetic waves, through a modulation process where the information signal is combined with the carrier signal. Radio can be used to store and transmit digital information or data, broadcast radio transmitters only transmit in one direction, the LoRa system itself has a transmitter and receiver. LoRa is a radio system that is frequency modulated, with a frequency band of 433-868 MHz, 915-923 MHz, for the Asian area the frequency used is around 923-925 MHz. LoRa technology uses low power, its range reaches 15 km in remote areas, this technology is good for transmitting small data, namely around 10-20 bytes. The bridge slope change detection system was built using the LoRa radio network.

The bridge tilt detection system was created using an IoT-connected accelerometer sensor. IoT is the internet of everything, an internet network connected to a sensor system, which functions to transmit sensor detection data. IoT is the most advanced technology and its implementation continues to develop in various fields. IoT has a significant impact on every facet of contemporary culture [3], the application of wearable IoT devices is becoming more advanced, increasingly widespread, starting from social networking, payments, and navigation, to health monitoring [4], applying accelerometers with IoT devices to earthquake detection [5], reading sensors and sending data for warnings before a disaster occurs [6]. The slope of the building and physical structure was observed using a data acquisition system utilizing the ADXL335 accelerometer sensor [7]. Accelerometer sensor instrumentation system, signal conditioning, with ATmega 32 microcontroller [8]. To analyze the strength of the bridge using an accelerometer (ADXL345) as a vibration sensor [9]. Accelerometers are devices that are widely used to determine bridge vibrations [10], [11]. Control all nodes via the IEEE 802.11 wireless network, easily and sample deviations, vibrations due to load and water level [12]. The use of a single microelectromechanical systems (MEMS) accelerometer, namely the MMA7361L, for the tilt sensor [13]. Wireless communication and IoT transmission are powerful extensive cognitive processing and highly effective capabilities in the field of bridge inspection [14].

This research will measure the throughput, delay, packet loss parameters, which previous research used the MMA7361L accelerometer sensor is connected to Zigbee 802.15 [15]. Automatic distance and slope measurement system using an IoT-based microcontroller [16], [17]. IoT MEMS accelerometer for translational measurements of motion line actuators and vibrations of Great rotating machines [18]. Accelerometer to detect jerks on the bridge, vibration sensor to detect vibrations that occur on the bridge [19]. IoT technology and sensors in identifying cracks in bridge structures and building a bridge structure health monitoring system [20]. The radio frequency identification (RFID), real time control (RTC) and liquid crystal display (LCD) interfacing modules are connected to the ESP32 microcontroller, and the IoT [21]. Wi-Fi module, ESP8266-12 for connecting to the system internet. The concept of publisher and subscriber is used for communication [22]. IoT using WiFi and MQTT can display blood pressure, blood flow, dialysate temperature, and conductivity of dialysate fluid [23]. LoRa technology is suitable for IoT application scenarios to transmit small amounts of data over long distances and to transmit data with low power consumption [24], [25]. Based on the reference, it is very interesting to realize the accelerometer, LoRa and IoT sensor system by building a LoRa and IoT based bridge tilt detection system.

2. RESEARCH METHOD

The development of an early slope detection system can aid in bridge monitoring and maintenance, preventing further damage and extending the bridge's life. The gadgets are divided into two groups: hardware (which includes an ESP32 microcontroller, an accelerometer sensor, an internet network, a laptop, and a smartphone) and software (which includes the Arduino integrated development environment (IDE) program, windows 10 64-bit). Employees that check bridges utilize a laptop or smartphone with a web browser to access real-time data on bridge slope early detection. The accelerometer sensor sends a digital signal to the ESP32 microprocessor, which interprets changes in the bridge's slope as an angle. The ESP32 has a LoRa transmitter and receiver system that can send signals to an internet system or deliver data to the cloud.

Utilizing an accelerometer, ESP32, LoRa, internet connection, and cellphone, the system was constructed. The internet is a computer network that is connected to each other via the internet protocol package (TCP/IP) so that it is connected throughout the world. IoT systems are devices that connect to the internet, collect data and perform actions or controls. System testing/experiments were carried out in the microprocessor laboratory of the telecommunications engineering study program, electrical engineering department, Ujung Pandang State Polytechnic. An accelerometer sensor that is integrated with the LoRa and IoT systems is utilized to detect changes in the inclination angle of the bridge. Figure 1 shows the design of this system. A bridge slope early detection model utilizing LoRa and IoT.

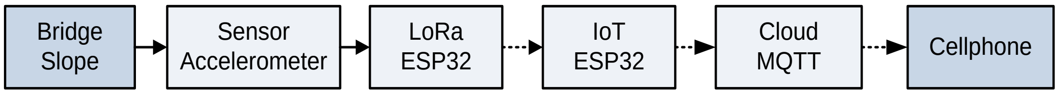


Figure 1. System model for early detection of bridge slope using LoRa and IoT

The bridge tilt detection system uses an accelerometer sensor to read changes in deviation that occur, and data on electrical quantities, namely digital signals entering the ESP32 microprocessor with Lora and IoT cloud received by the smartphone. The accelerometer sensor will detect changes in tilt angle, the data is sent to the ESP32 microprocessor transmitting data via radio communication, namely LoRa which consists of a transmitter and receiver, The transmitter then transmits to the internet network system, received by the cellphone. It is hoped that the system will function to detect changes in the angle of the bridge, be able to transmit data continuously, data will be received via the computer network by the inspection officer's cellphone properly.

An accelerometer is a transducer that is capable of detecting acceleration and measuring vibrations to measure the Earth's gravitational acceleration. This is in accordance with the type and type of accelerometer sensor, namely the MPU 6050 accelerometer sensor with three measurement axes, namely the x-axis, y-axis and z-axis. Message exchange with the publish/subscribe model in MQTT is an alternative to the client-server model, where clients (publishers/subscribers) communicate directly with other endpoints on a topic through brokers whose job is to filter messages and distribute them [23]. IoT technology is a concept where objects can transmit data over a network without human-to-human or human-to-computer interaction. Internet of MEMS. The IoT is most closely related to machine-to-machine (M2M) communications. The ESP32 microcontroller provides a standalone WiFi network solution in the form of a microcontroller bridge to the WiFi network. ESP32 uses a dual-core processor running on Xtensa LX16 [6] instructions.

Building an early detection system for bridge tilt using LoRa and IoT. Research Instruments; specifications for device requirements are as follows: laptop, cellphone, accelerometer sensor, ESP32 microcontroller, power supply 5 Volt. Software: windows 10 64-bit operating system using Arduino IDE, MQTT lens. This research uses the ITGMPU accelerometer and the TTGO ESP 32 IoT strap which is equipped with the LoRa system.

Measuring internet network performance, the parameters measured are throughput (bps), delay (seconds), packet loss. The quality of the network sent will be measured for the data sent, to determine the network's ability to provide better services, guaranteeing the level of network performance; throughput parameters are the effective data transfer speed, the total number of packet arrivals during the time interval in bits per second, Packet loss regulates the quality of data traffic services, overcomes packet loss and network delays. Throughput, Packet loss, and network delay are transmission process times that depend on the distance from the data origin to the data destination. This system measures on the slope of the bridge and measures that quality from network used. Obtaining network performance is important for determining the quality of internet network services, for identifying performance problems and network efficiency.

Figure 2 shows the steps to build this system. To make it easier to understand the process of this system, it can be seen in Figure. 2 which shows an overview of the process flow/steps of the bridge angle tilt detection system. From the first picture, start with the starting steps, the second is the declaration of tilt variables port, solid-state drive (SSD), password, cloud MQTT, the third step is network connection LoRa, the fourth step is the tilt detection process, the fifth step is sending data to the LoRa recipient, the sixth step is giving a 10 second delay , step seventh step receive data from LoRa delivery, eighth step internet network connection, ninth step send data to the cloud, tenth step receive data to the cloud, eleventh step display slope data, twelfth step detect again, end of the thirteenth step program stops. Figure 3 show bridge slope the initial detection system using LoRa and IoT. From Figure 3, this system is simply built using an ITGMPU type accelerometer and ESP32 type TTGO, IoT internet network.

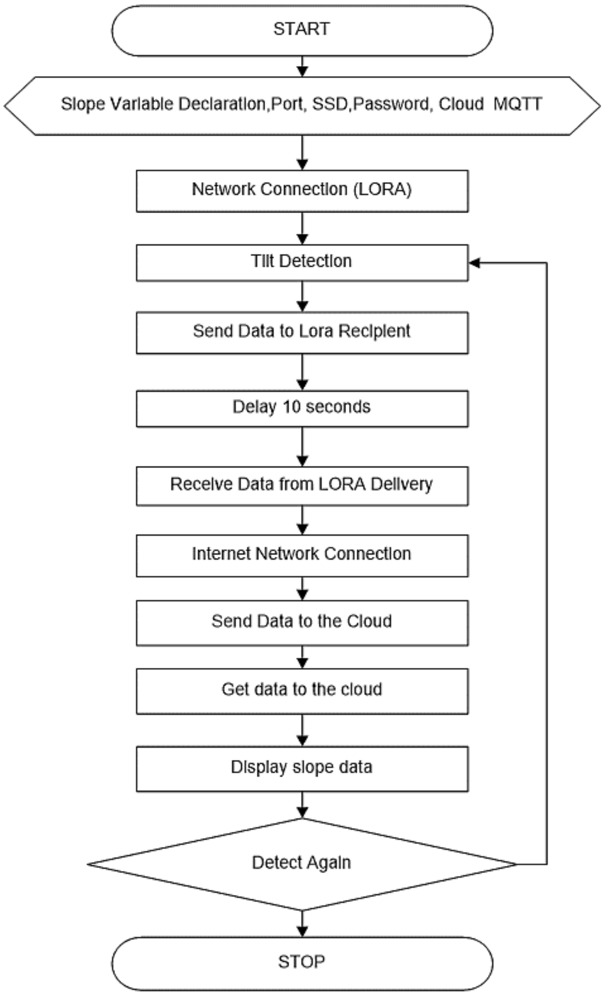


Figure 2. Bridge slope system flow chart for early detection using LoRA and IoT

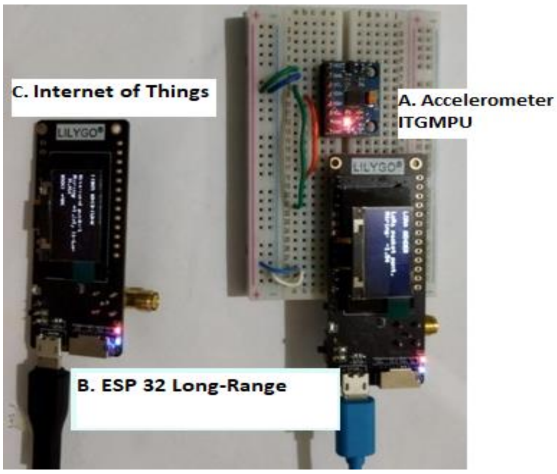


Figure 3. Bridge slope the initial detection system using LoRa and IoT

Figure 4 show data from measuring the bridge slope early detection system using LoRa and IoT. Figure 4 shows the bridge tilt angle detection data display on the inspection officer's cellphone. This system measures on the slope of the bridge and measures that quality from network used.

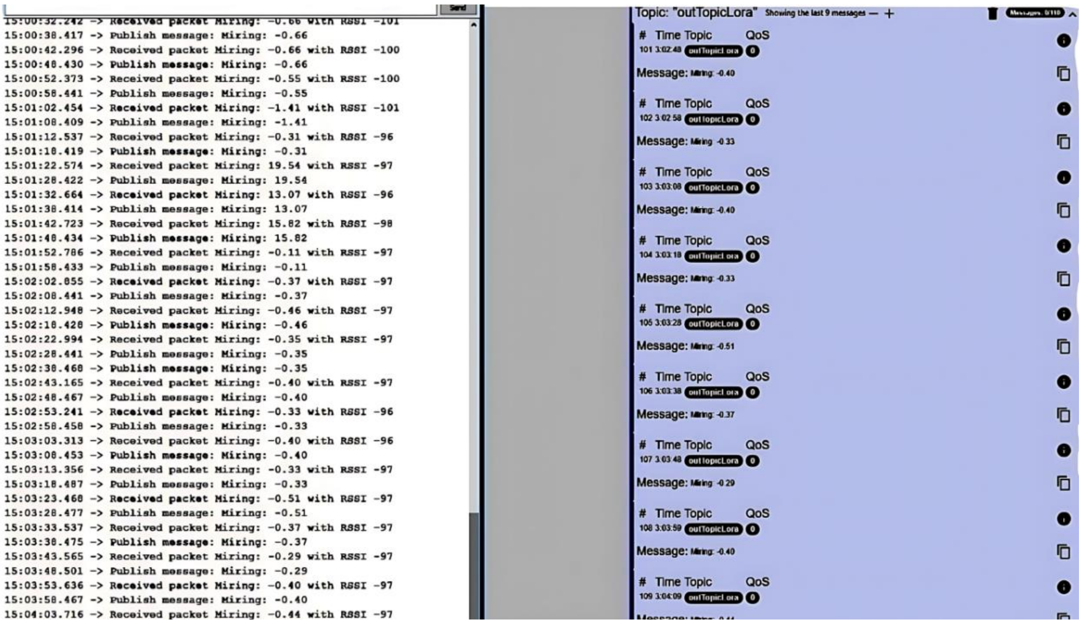


Figure 4. Display data from the slope of the bridge detection system using LoRa and IoT

3. RESULTS AND DISCUSSION

The accelerometer system receives changes in the tilt angle of the bridge, the data will be forwarded to the ESP32 which has a LoRa system, LoRa will send the data to the LoRa receiver and then the LoRa receiver sends it to the IoT internet, which can be viewed using a web browser page on a smartphone. Data from early detection of bridge slope using accelerometer, IoT, LoRa sensors produces early detection of bridge slope of 0.11^{o} to 15.2^{o}. The latest findings involve creating a low-cost tool by utilizing an ESP32 type TTGO microcontroller in conjunction with an accelerometer sensor.

This system measures on the slope of the bridge and measures that quality from network used. To determine the quality of internet network services, delay parameters are measured.

Throughput, Packet loss, for identify network performance and efficiency. This research measures system performance, namely throughput of 85 bps, Packet loss 0% and delay of around 1.2 mseconds. From the results obtained, the performance of the bridge slope early detection system has very good performance based on the standards issued by TIPHON.

Data from measurements from the IoT and LoRa-based early detection system for bridge slopes obtained from the results of early detection of bridge slopes with throughput, Packet loss, system delay, in Table 1. This system provides benefits, namely the ease of obtaining data on changes in bridge slope quickly, continuously on mobile phones. Measuring throughput parameters, Packet loss, system delay, to determine the quality of the network used. The bridge angle tilt detection system can be implemented on the bridge. The data are shown in Table 1.

Table 1. Measurement table for the bridge tilt early detection system with IoT and LoRa, as well as throughput, Packet loss, and delay

Number of test	Accelerometer sensor tilt angle (^{o})	Throughput (bps)	Packet loss (%)	Delay (msecond)	Category: throughput, Packet loss, latency
1	0.11	89	0	1.12	Very good
2	1.41	85	0	1.23	Very good
3	13.7	85	0	1.24	Very good
4	15.2	81	0	1.34	Very good

4. CONCLUSION

Based on the results of the research and discussion, it shows that the system with the ITGMPU accelerometer sensor, TTGO ESP 32, LoRa, IoT, cellphone, the system sends data continuously, accessed using a web browser page. The system functions well. The benefit is that it makes it easier to monitor the slope angle of the bridge on a smartphone. The bridge tilt angle detection system succeeded in identifying tilt angle variations of 0.11^o to 15.2^o, which is the research output and continues to be sent to the bridge inspection officer's cell phone. The system for measuring internet network performance, the results obtained in this research were throughput of 85 bits per second, Packet loss of 0% and to measure network delay, an average delay of 1.2 mseconds was obtained. The system has internet network performance in the very good category. This research can be developed by providing information on normal or dangerous conditions to bridge users.

ACKNOWLEDGEMENT

This work was supported by Public Service Agency of Ujung Pandang State Polytechnic, DIPA BLU fiscal year 2024 number: SP DIPA-023.18.2.677613/2024.

REFERENCES

[1] A. A. Hapsari, E. Supriyanto, A. Hasan, A. Suharjono, "Accelerometer Sensor Data Analysis Of Bridge Structural Health Monitoring System," I*OP Conference Series: Materials Science and Engineering*, 2021, doi:10.1088/1757-899X/1108/1/012026.

[2] L Tian, J. Zhao, B. Pan, and Z. Wang, "Full-Field Bridge Deflection Monitoring with Off-Axis Digital Image Correlation," *Sensor*, 2021. doi: 10.3390/s21155058.

[3] D. B. Khalifa and M. Martel, "Precision tuning of an accelerometer-based pedometer algorithm for IoT devices," in *IoTaIS 2020 - Proceedings: 2020 IEEE International Conference on Internet of Things and Intelligence Systems*, IEEE, Jan. 2021, pp. 116–122. doi: 10.1109/IoTaIS50849.2021.9359700.

[4] F. Sun, W. Zang, H. Huang, I. Farkhatdinov, and Y. Li, "Accelerometer-based key generation and distribution method for wearable IoT devices," *IEEE Internet of Things Journal*, vol. 8, no. 3, pp. 1636–1650, Feb. 2021, doi: 10.1109/JIOT.2020.3014646.

[5] C. T. Chiang, "Design of a CMOS MEMS accelerometer used in IoT devices for seismic detection," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 8, no. 3, pp. 566–577, 2018, doi: 10.1109/JETCAS.2018.2825604.

[6] A. Dhuri, S. Kadam, P. Jogale, L. Kawade, and P. S. Kore, "IoT based bridge health smart monitoring system," *International Journal of Engineering Research and Technolog*, vol. 8, no. 11, pp. 109–111, 2020.

[7] N Amir, R Dziyauddin, N. Mohamed, L. Latiff, and N. Ismail, "Development of Fall Detection Device Using Accelerometer Sensor," *IEEE 7th International Conference on Smart Instrumentation, Measurement, and Application (ICSIMA)*, 2021, doi: 10.1109/ICSIMA50015.2021.9526289.

[8] Alimuddin, F. Sulaiman, I Parinduri, R. Abdullah, T. Firmansyah, and M. Syarif, "Accelerometer Sensor Applications Early Warning System Train Accidents due to Landslide at Laboratory Scale," *IOP Conference Series: Materials Science and Engineering*, 2021. doi: 10.1088/1757-899X/180/1/012152.

[9] P. B. Dinesh, T. Lakshmi, and D. C. KishoreRaja, "Monitoring of streng of bridge using accelerometer,"

International Journal of Pure and Applied Mathematics, vol. 119, no. 18, pp. 1503–1511, 2018.

[10] K. Chilamkuri and V. Kone, “Monitoring of varadhi road bridge using accelerometer sensor,” *Materials Today: Proceedings*, vol. 33, pp. 367–371, 2020, doi: 10.1016/j.matpr.2020.04.159.

[11] A. Elhattab, N. Uddin, and E. OBrien, “Extraction of bridge fundamental frequencies utilizing a smartphone mems accelerometer,” *Sensors (Switzerland)*, vol. 19, no. 14, p. 3143, Jul. 2019, doi: 10.3390/s19143143.

[12] S. K. P. Muddala, D. K. Divya, P. Nimbalkar, and R. Patil, “IoT based bridge monitoring system,” *International Journal of Advanced Research, Ideas an Innovations in Technology*, vol. 5, no. 2, pp. 2044–2047, 2019.

[13] D. R. Santoso, S. Maryanto, and A. Nadhir, “Application of single MEMS-accelerometer to measure 3-axis vibrations and 2-axis tilt-angle simultaneously,” *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 13, no. 2, pp. 442–450, 2015, doi: 10.12928/TELKOMNIKA.v13i2.1490.

[14] H. Sun, Z. Zhang, and P. Li, “Design and analysis of bridge inspection system based on wireless communication and internet of things technology,” *Wireless Communications and Mobile Computing*, pp. 1–5, Oct. 2021, doi: 10.1155/2021/3410844.

[15] W. Lee, H Kim, C. Lee, and S. Kim, "Development of Digital Device Using ZigBee for Environmental Monitoring in Underground Mines," *Applied Science*, 2022, doi: 10.3390/app122311927

[16] S. Komarizadehasl, M. Komary, A. Alahmad, J. Lozano-Galant, G. Ramos, and J. Turmo, "A Novel Wireless Low-Cost Inclinometer Made from Combining the Measurements of Multiple MEMS Gyroscopes and Accelerometers," *Sensor*, 2022, doi: 10.3390/s22155605.

[17] I. Koene, V. Klar, and R. Viitala, “IoT connected device for vibration analysis and measurement,” *HardwareX*, vol. 7, pp. 1–15, Apr. 2020, doi: 10.1016/j.ohx.2020.e00109.

[18] A. Hoag, N. A. Hoult, W. A. Take, F. Moreu, H. Le, and V. Tolikonda, “Measuring displacements of a railroad bridge using DIC and accelerometers,” *Smart Structures and Systems*, vol. 19, no. 2, pp. 225–236, Feb. 2017, doi: 10.12989/sss.2017.19.2.225.

[19] K. A. Gunjal, P. D. Gunjal, and T. C. Gunjal, “IoT based monitoring and maintenace of highway bridges using wireless sensor network,” *International Research Journal of Engineering and Technology*, vol. 7, no. 4, pp. 5619–5622, 2020.

[20] L. Zhang, G. Zhou, Y. Han, H. Lin, and Y. Wu, “Application of internet of things technology and convolutional neural network model in bridge crack detection,” *IEEE Access*, vol. 6, pp. 39442–39451, 2018, doi: 10.1109/ACCESS.2018.2855144.

[21] A. Akbar, Z Zaenudin, Z Mutaqin, and L Samsumar "IoT-Based Smart Room Using Web Server-Based Esp32 Microcontroller," *Formosa Journal of Computer and information Science*, pp. 79-86, 2022, doi: 10.55927/fjcis.v1i2.1241.

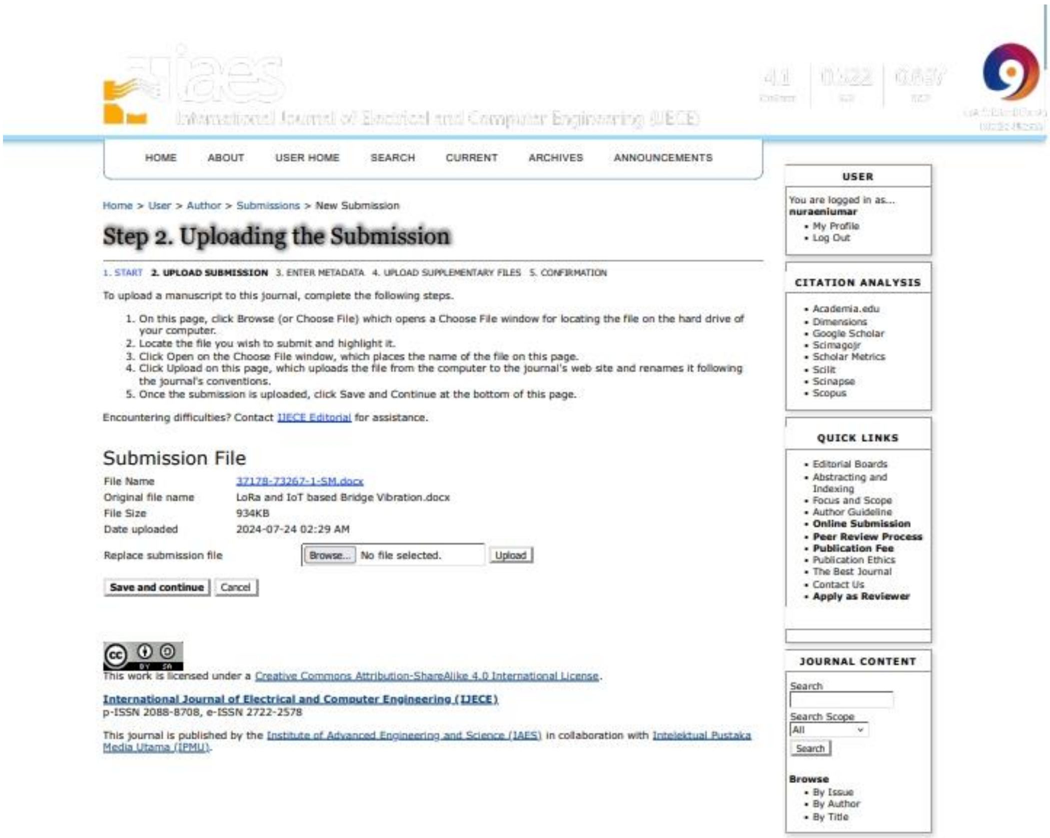
[22] M. Kashyap, V. Sharma, and N. Gupta, “Taking MQTT and NodeMcu to IoT: communication in internet of things,” *Procedia Computer Science*, vol. 132, pp. 1611–1618, 2018, doi: 10.1016/j.procs.2018.05.126.

[23] H. Alshammari, "The internet of things healthcare monitoring system based on MQTT protocol,"*Alexandria Engineering Journal*, pp. 275-287, 2023, doi: 10.1016/j.aej.2023.01.065

[24] S. Ould, and N Bennett, "Energy Performance Analysis and Modelling of LoRa Prototyping Boards,"*Sensor*, 2021, doi: 10.3390/s21237992.

[25] Q. Zhou, K. Zheng, L. Hou, J. Xing, and R. Xu, “Design and implementation of open LORa for IoT,” *IEEE Access*, vol. 7, pp. 100649–100657, 2019, doi: 10.1109/ACCESS.2019.2930243.

Lampiran 3 Publikasi karya ilmiah 1,2,3



Internet of Things and Long Range Based Bridge Vibration

Nuraeni Umar[1,2], Syafruddin Syarif[1] , Dewiani[1], Merna Baharuddin[1]
[1]Department of Electrical Engineering, Hasanuddin, Gowa, Indonesia
[2]Department of Electrical Engineering, Ujung Pandang State Polytechnic, Makassar, Indonesia

Article Info

Article history:

Received month dd, yyyy
Revised month dd, yyyy
Accepted month dd, yyyy

Keywords:

Vibration
Bridge
Accelerormeter
ESP32
IoT
LoRa

ABSTRACT

The aim of the research is to monitor vibrations on bridges connected to Long Range networks and the Internet of Things received on mobile phones. This system is efficient, low cost, easy to use and simple. It is important to monitor bridge vibrations for maintenance purposes. Building a system using Accelermeter sensors, ESP32 microprocessors, LoRa, IoT, cellphone. This system obtaining vibration warning data on cellphones. The Lora system is used to address areas without an internet connection, Lora will send data to areas where an internet connection is available. The research results show that bridge vibration monitoring is functioning well, vibration alerts are obtained on cellphones, detected with a vibration value of 0.01 to 0.29 g and internet network performance obtained for throughput parameters is around 90 bps, Packet loss 0%, Delay 1.2 mseconds. The internet network performance is in the very good category.

Corresponding Author:

Nuraeni Umar
Department of Electrical Engineering, Ujung Pandang State Polytecnic
St. Perintis Kemerdekaan Km. 10 Tamalanrea, Makassar 90245, Indonesia
Email: nuraeni_umar@poliupg.ac.id

1. INTRODUCTION

Bridges play a crucial function in reducing travel distances in land transportation. Indonesia is made up of islands connected by bridges, as well as rivers that need flyovers and bridges. Even though the infrastructure is designed to be strong enough to last for a long time, damage to bridge structures typically happens at an uncontrolled rate. A bridge will sustain damage if its slope fluctuates excessively. To enable logical action to be done on the bridge, bridge inspection personnel visit and evaluate the bridge's condition on a constant basis.

In response to these challenges, the integration of advanced technologies, such as the Internet of Things (IoT) and long-range, low-power communication (LoRa), into infrastructure development and monitoring systems emerges as a promising solution. Through the utilization of sensors and smart devices, these advanced technologies play a crucial role in obtaining, transmitting, retaining, and analyzing real-time data. This not only offers significant insights into the safety and state of bridges but also provides valuable information for, optimizing maintenance strategies, and enhancing the overall safety of transportation infrastructure .

Numerous scholarly works have explored technologies related to this work. [1] has mentions the importance and explains the importance of Structural Health Monitoring System. Assessment of the health condition of an infrastructure is required to be carried out continuously without stopping so that rational action can be taken. Generating systems that enable the Real time monitoring of bridges have been done by [2,3,4,5,6], by using sensors or sensor networks, with notifications using wireless cellular network and local notification by buzzer and LCD displays.

One of the technologies that is seen as evolving the fastest is the Internet of Things (IoT). It has an effect on nearly every facet of contemporary life [7]. Various monitoring purposes in everyday life which implemented IoT has been discussed in [8,9,10,11]. A work on a system implementing IoT for seismic detection is presented in [12] and for smart parking is reported in [13]. In order to identify cracks in bridge structures, [14] examined the usefulness of IoT technology in practice and developed an IoT-based bridge structure health monitoring system. Numerous IoT system on bridge monitoring have also reported in [15,16,17].

The research conducted by the authors is the development and application of previous research in various monitoring utilizing IoT, communications and sensors technologies. In the development and application of IoT technology, the authors utilize IoT with ESP32 connect to Cloud MQTT [18,19,20], accelerometer [21,22] sensors in MPU6050 sensor [23,24] for the of vibration the bridge readings. LoRA point to point communication also applied to accommodate area which are not covered by the internet connection. Previously in [25] two LoRa-based systems for Structural Health Monitoring (SHM), which is to monitor the variation of inclination in the structure located in Italy, are presented. This research also equipped with LoRA to transmit sensor data of the bridge.

The wind field will influence vibrations on bridges [26], while span bridges experience vibrations caused by vehicles crossing them [27]. Vibration sensors are of great interest in the field of monitoring the structural health of bridges [28], for vibration measurements carried out with accelerometer sensors [29], measuring structural vibrations is very important, to be utilized by bridge health monitoring and management services [30], [31]. creating a sensor system that measures vibrations that occur on bridges is very important, creating a low cost system, practical in operation.

It is clear from the reviewed literature that a gap exists in terms of affordable systems that can be applied. To address this gap, this paper proposes a LoRa and IoT based bridge vibration detection systems. The system's objective is to enhance existing solutions by providing a low-cost system to address issues associated with bridge safety and management. By using a sensor that is embedded at the bridge, the system aims to monitor the conditions of the bridge, which is the Vibration information of the bridge. The sensor readings are to be processed by the microcontroller ESP32 device. Moreover, the bridge Vibration status will be transmitted via LoRa and IoT to cloud MQTT. In addition, a mobile application will be developed to allow relevant technicians and engineers access to reports and alerts of the bridge's vibration status. Moreover, performance of the data transmission also have been observed.

2. METHOD

This study develops a vibration reading system for an IoT and LoRa-based bridge. There are two components to the Bridge vibration detecting system: software and hardware. The accelerometer sensor, ESP32 microcontroller, integrated LoRa, IoT Cloud internet network, smartphone, and laptop make up the hardware instrument. The Arduino IDE software is utilized. Using a laptop or smartphone web browser, one can access this system. The accelerometer sensor continuously transmits real-time monitoring data for bridge vibration detection to the cellphone's web browser. The data received by the LoRa transmitter and subsequently received by the LoRa receiver will be transmitted by the sensor as the first stage in sending data to the ESP32 Module, will transmit data received by the LoRa transmitter and then received by the LoRa receiver. The LoRa receiver system sends data to the Web application system, IoT network, bridge vibration data will be received by the cellphone. Research procedures on the model in Figure 1. System Long Range and Internet of Things Based Bridge Vibration.

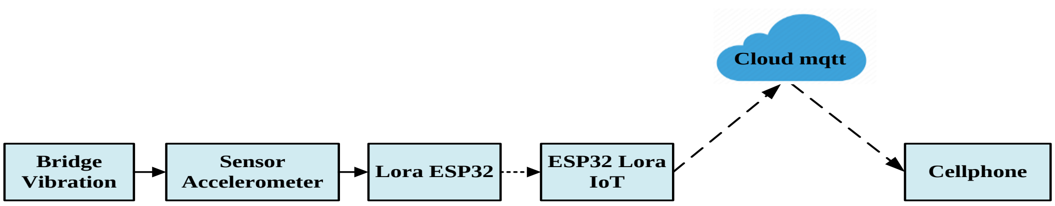


Figure 1. System LoRa and IoT Based Bridge Vibration.

The process carried out by this system can be seen in Figure 2. Flow chart System LoRa and IoT Based Bridge Vibration. It is a Flow chat with the initial step of declaring vibration, after that the process continues with LoRa, and IoT Network variables, then Initiated LoRa Network, then Get Vibration value, transmitting vibration data by LoRa sender, given a delay value of 2 seconds, continues with the LoRa Gateway get vibration data process, then publishes vibration data to the MQTT cloud, continues to subscribe data to the MQTT cloud, will show vibration data, then read the data again and finish, stop.

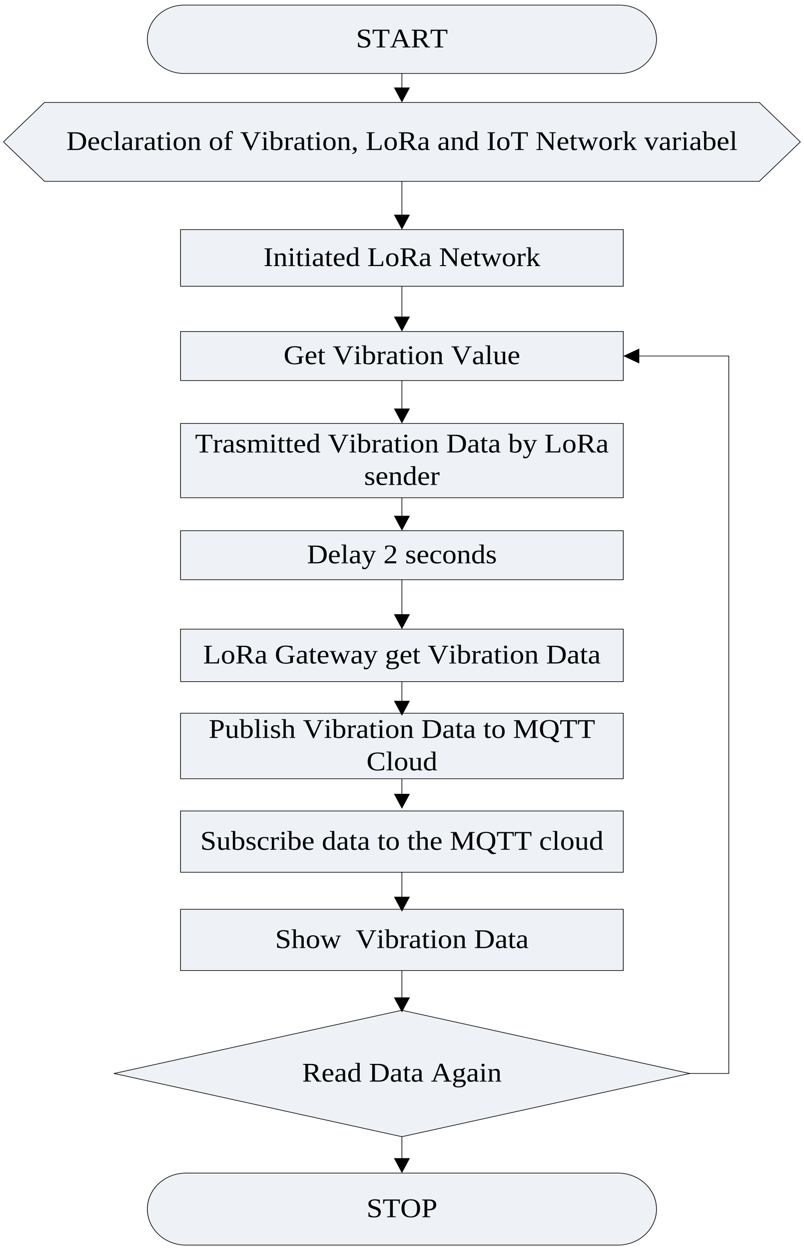


Figure 2. Flow chart System LoRa and IoT Based Bridge Vibration.

The practical implemented system, the accelerometer sensor will register the vibrations, send the data to the LoRa network, which will subsequently send it to the internet network so that cellphones may receive it. Figure 3. Bridge vibration using LoRa and IoT systems includes a 5 volt power supply, In the experimental implemented design system vibrasi bridge used a cellphone, an ESP32 LoRa, IoT and an accelerometer sensor. To run programs on this device, the Arduino IDE software is utilized.

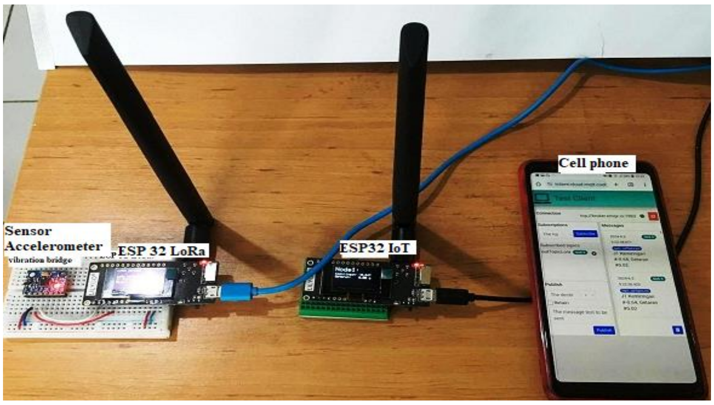


Figure 3. Bridge vibration using LoRa and IoT systems

Figure 4. Show Bridge Vibration Based on LoRa and IoT. This is the display design of the data received by the cell phone. The system sends data continuously by displaying the bridge vibration measurement data on the cellphone.

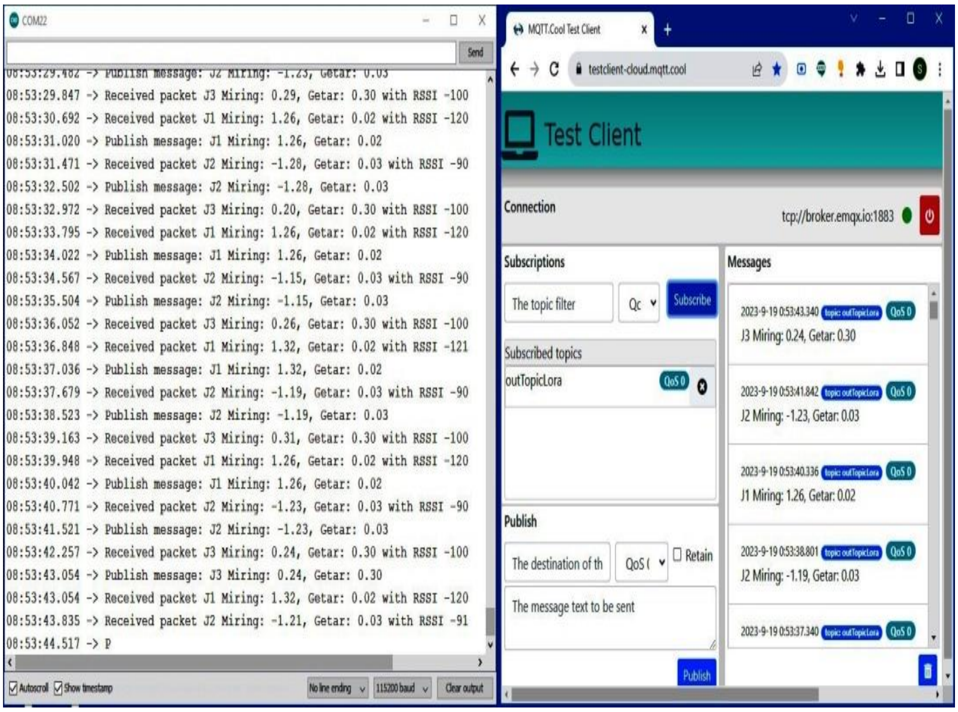


Figure 4. Show Bridge Vibration Based on LoRa and IoT.

3. RESULTS AND DISCUSSION

The accelerometer sensor is used in designing a vibration detection system with LoRa and IoT. After detecting the vibration, it is then sent to the ESP32 LoRa then the ESP32 IoT, transmitted to the MQTT cloud, and received by the cellphone. The Arduino IDE software successfully processes this bridge vibration system, functions to complete data transmission, running sensors to detect bridge vibrations. This system is simple, cheap and easy to operate in measuring vibrations on bridges. The vibration measurement results obtained were 0.01 to 0.29 Hz, this data is useful for bridge supervisors for routine bridge maintenance data.

Measuring the performance of the internet network used is useful for developing the quality of the location's network. From measuring the throughput parameters, the result was 90 bps, while the Packet loss was obtained at 0%, for a data delay of 1.2 seconds. Based on the data, the bridge vibration system has very good internet network performance according to the TIPHON criteria.

The proposed system succeeded in displaying data vibrasi bridge on cellphone, and succeeded in measuring internet network performance for throughput, Packet loss, delay parameters. Data on measurement results are available in table 1. The performance of bridge vibration systems.

Table 1. The performance of bridge vibration systems.

Bridge Vibration (g)	**Throughput (bps)**	**Packet loss (%)**	**Delay (second)**	**Category: Throughput, Packet loss, Delay**
0,01	91	0	1,1	Very Good
0,02	90	0	1,2	Very Good
0,03	90	0	1,2	Very Good
0,29	90	0	1,3	Very Good

4. CONCLUSION

The findings of this research include Lora and IoT-based bridge vibration detection, the benefits of continuous transmission of vibration data, low-cost system design, system efficiency that is easy to access, and the ability to send cellphone alerts regarding the severity of vibrations that occur on bridges. to overcome areas without internet, it uses (Tx LoRa and Rx LoRa) to communicate vibration data via a Long Distance radio network and then transfer it to the IoT internet network, LoRa operates in places without internet network availability. Building this system through design and production with ESP32 microprocessor, LoRa, IoT, and accelerometer sensors. The research findings show that bridge vibration monitoring is effective, vibration alerts can be received on cellphones, and vibrations can be detected using internet networks, and vibration values 0.01 to 0.29 g. Performance measured for throughput metrics is approximately 90 bps, packet loss 0%, delay 1.2 mseconds. Internet network performance is very good. For further system development, it can be equipped with a positioning system so that the measurement coordinate points are obtained precisely.

ACKNOWLEDGEMENT

This work was supported by Public Service Agency of Ujung Pandang State Polytechnic, DIPA BLU fiscal year 2024 number: SP DIPA-023.18.2.677613/2024.

REFERENCES

[1] A. A. Hapsari, E. Supriyanto, A. Hasan, A. Suharjono, "Accelerometer Sensor Data Analysis Of Bridge Structural Health Monitoring System," I*OP Conference Series: Materials Science and Engineering*, 2021, doi:10.1088/1757-899X/1108/1/012026.

[2] L Tian, J. Zhao, B. Pan, and Z. Wang, "Full-Field Bridge Deflection Monitoring with Off-Axis Digital Image Correlation," *Sensor*, 2021. doi: 10.3390/s21155058.

[3] D. B. Khalifa and M. Martel, "Precision tuning of an accelerometer-based pedometer algorithm for IoT devices," in *IoTaIS 2020 - Proceedings: 2020 IEEE International Conference on Internet of Things and Intelligence Systems*, IEEE, Jan. 2021, pp. 116–122. doi: 10.1109/IoTaIS50849.2021.9359700.

[4] F. Sun, W. Zang, H. Huang, I. Farkhatdinov, and Y. Li, "Accelerometer-based key generation and distribution method for wearable IoT devices," *IEEE Internet of Things Journal*, vol. 8, no. 3, pp. 1636–1650, Feb. 2021, doi: 10.1109/JIOT.2020.3014646.

[5] C. T. Chiang, "Design of a CMOS MEMS accelerometer used in IoT devices for seismic detection," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 8, no. 3, pp. 566–577, 2018, doi: 10.1109/JETCAS.2018.2825604.

[6] A. Dhuri, S. Kadam, P. Jogale, L. Kawade, and P. S. Kore, "IoT based bridge health smart monitoring system," *International Journal of Engineering Research and Technolog*, vol. 8, no. 11, pp. 109–111, 2020.

[7] N Amir, R Dziyauddin, N. Mohamed, L. Latiff, and N. Ismail, "Development of Fall Detection Device Using Accelerometer Sensor," *IEEE 7th International Conference on Smart Instrumentation, Measurement, and Application (ICSIMA)*, 2021, doi: 10.1109/ICSIMA50015.2021.9526289.

[8] Alimuddin, F. Sulaiman, I Parinduri, R. Abdullah, T. Firmansyah, and M. Syarif, "Accelerometer Sensor Applications Early Warning System Train Accidents due to Landslide at Laboratory Scale," *IOP Conference Series: Materials Science and Engineering*, 2021. doi: 10.1088/1757-899X/180/1/012152.

[9] P. B. Dinesh, T. Lakshmi, and D. C. KishoreRaja, "Monitoring of streng of bridge using accelerometer," *International Journal of Pure and Applied Mathematics*, vol. 119, no. 18, pp. 1503–1511, 2018.

[10] K. Chilamkuri and V. Kone, "Monitoring of varadhi road bridge using accelerometer sensor," *Materials Today: Proceedings*, vol. 33, pp. 367–371, 2020, doi: 10.1016/j.matpr.2020.04.159.

[11] A. Elhattab, N. Uddin, and E. OBrien, "Extraction of bridge fundamental frequencies utilizing a smartphone mems accelerometer," *Sensors (Switzerland)*, vol. 19, no. 14, p. 3143, Jul. 2019, doi: 10.3390/s19143143.

[12] S. K. P. Muddala, D. K. Divya, P. Nimbalkar, and R. Patil, "IoT based bridge monitoring system," *International Journal of Advanced Research, Ideas an Innovations in Technology*, vol. 5, no. 2, pp. 2044–2047, 2019.

[13] D. R. Santoso, S. Maryanto, and A. Nadhir, "Application of single MEMS-accelerometer to measure 3-axis vibrations and 2-axis tilt-angle simultaneously," *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 13, no. 2, pp. 442–450, 2015, doi: 10.12928/TELKOMNIKA.v13i2.1490.

[14] H. Sun, Z. Zhang, and P. Li, "Design and analysis of bridge inspection system based on wireless communication and internet of things technology," *Wireless Communications and Mobile Computing*, pp. 1–5, Oct. 2021, doi: 10.1155/2021/3410844.

[15] W. Lee, H Kim, C. Lee, and S. Kim, "Development of Digital Device Using ZigBee for Environmental Monitoring in Underground Mines," *Applied Science*, 2022, doi: 10.3390/app122311927

[16] S. Komarizadehasl, M. Komary, A. Alahmad, J. Lozano-Galant, G. Ramos, and J. Turmo, "A Novel Wireless Low-Cost Inclinometer Made from Combining the Measurements of Multiple MEMS Gyroscopes and Accelerometers," *Sensor*, 2022, doi: 10.3390/s22155605.

[17] I. Koene, V. Klar, and R. Viitala, "IoT connected device for vibration analysis and measurement," *HardwareX*, vol. 7, pp. 1–15, Apr. 2020, doi: 10.1016/j.ohx.2020.e00109.

[18] A. Hoag, N. A. Hoult, W. A. Take, F. Moreu, H. Le, and V. Tolikonda, "Measuring displacements of a railroad bridge using DIC and accelerometers," *Smart Structures and Systems*, vol. 19, no. 2, pp. 225–236, Feb. 2017, doi: 10.12989/sss.2017.19.2.225.

[19] K. A. Gunjal, P. D. Gunjal, and T. C. Gunjal, "IoT based monitoring and maintenace of highway bridges using wireless sensor network," *International Research Journal of Engineering and Technology*, vol. 7, no. 4, pp. 5619–5622, 2020.

[20] L. Zhang, G. Zhou, Y. Han, H. Lin, and Y. Wu, "Application of internet of things technology and convolutional neural network model in bridge crack detection," *IEEE Access*, vol. 6, pp. 39442–39451, 2018, doi: 10.1109/ACCESS.2018.2855144.

[21] A. Akbar, Z Zaenudin, Z Mutaqin, and L Samsumar "IoT-Based Smart Room Using Web Server-Based Esp32 Microcontroller",*Formosa Journal of Computer and information Science*, pp. 79-86, 2022, doi: 10.55927/fjcis.v1i2.1241.

[22] M. Kashyap, V. Sharma, and N. Gupta, "Taking MQTT and NodeMcu to IoT: communication in internet of things," *Procedia Computer Science*, vol. 132, pp. 1611–1618, 2018, doi: 10.1016/j.procs.2018.05.126.

[23] H. Alshammari, "The internet of things healthcare monitoring system based on MQTT protocol", *Alexandria Engineering Journal*, pp. 275-287, 2023, doi: 10.1016/j.aej.2023.01.065

[24] S. Ould, and N Bennett, "Energy Performance Analysis and Modelling of LoRa Prototyping Boards,"*Sensor*, 2021, doi: 10.3390/s21237992.

[25] Q. Zhou, K. Zheng, L. Hou, J. Xing, and R. Xu, "Design and implementation of open LoRa for IoT," *IEEE Access*, vol. 7, pp. 100649–100657, 2019, doi: 10.1109/ACCESS.2019.2930243.

[26] E. Yu, G. Xu, Y. Han, P. Hu, J.F. Townsend, Y. Li, "Bridge vibration under complex wind field and corresponding measurements: A review", *Journal of Traffic and Transportation Engineering (English Edition)*, vol. 9, Issue 3, pp. 339-362, 2022, doi.org/10.1016/j.jtte.2021.12.001

[27] S. Saidin , S. A. Kudus , A. Jamadin , M. A. Anuar , N. M. Amin , A. B. Z. Ya , K. Sugiura, "Vibration-based approach for structural health monitoring of ultra-high-performance concrete bridge", *Case Studies in Construction Materials 18*, e01752, 2023, doi.org/10.1016/j.cscm.2022.e01752.

[28] L. Liang, X. Wang, M. Li, Z. Wang, M. Jiang, Y. Wu, H. Zheng, "Self-powered active vibration sensor by peak–valley data processing independent of the environment toward structural health monitoring", *Next Nanotechnology,* vol. 117, 108935, Dec. 2023, doi.org/10.1016/j.nanoen.2023.108935,

[29] K. Zhou, Z. Wang, Y-Q. Ni, Y. Zhang, J. Tang, "Unmanned aerial vehicle-based computer vision for structural vibration measurement and condition assessment: A concise survey", *Journal of Infrastructure Intelligence and Resilience 2*, 10031, 2023, doi.org/10.1016/j.iintel.2023.100031.

[30] D. N. M. Phuong, D. T. Toan, "An IoT DAQ with Piezoelectric Sensor for Bridge Structure Vibration Measurement", *International Conference on System Science and Engineering (ICSSE)*, pp.51-54, 2023, DOI: 10.1109/ICSSE58758.2023.10227152.

[31] A. R. Al-Ali , S. Beheiry, A. Alnabulsi, S. Obaid, N. Mansoor, N. Odeh, A. Mostafa, "An IoT-Based Road Bridge Health Monitoring and Warning System", *Sensors,* 24, 469. 2024, doi.org/10.3390/s24020469.

Lampiran 4 Listing Program

LoRa Receiver (Gateway IoT):

```
/*
 LoRa32Receiver_oled_mqtt
*/
#include <WiFi.h>
#include <PubSubClient.h>

//Libraries for LoRa
#include <SPI.h>
#include <LoRa.h>

//Libraries for OLED Display
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

//define the pins used by the LoRa transceiver module
#define SCK 5
#define MISO 19
#define MOSI 27
#define SS 18
#define RST 23//14
#define DIO0 26

//433E6 for Asia
//866E6 for Europe
//915E6 for North America
#define BAND 915E6

//OLED pins
#define OLED_SDA 21//4
#define OLED_SCL 22//15
#define OLED_RST 16
#define SCREEN_WIDTH 128 // OLED display width, in pixels
#define SCREEN_HEIGHT 64 // OLED display height, in pixels

Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire,
OLED_RST);

String LoRaData;

const char* ssid = "PNUP HOTSPOT";
const char* password = "";
const char* mqttServer = "broker.emqx.io";
//const char* mqtt_server = "broker.mqtt-dashboard.com";
int port = 1883;
```

```
String stMac;
char mac[50];
char clientId[50];

long lastMsg = 0;
char msg[50];
int value = 0;

WiFiClient espClient;
PubSubClient client(espClient);

const int ledPin = 2;

void setup() {
  //initialize Serial Monitor
  Serial.begin(115200);
  pinMode(ledPin, OUTPUT);
  //reset OLED display via software
  pinMode(OLED_RST, OUTPUT);
  digitalWrite(OLED_RST, LOW);
  delay(20);
  digitalWrite(OLED_RST, HIGH);

  //initialize OLED
  Wire.begin(OLED_SDA, OLED_SCL);
  if(!display.begin(SSD1306_SWITCHCAPVCC, 0x3c, false, false)) { // Address
0x3C for 128x32
    Serial.println(F("SSD1306 allocation failed"));
    for(;;); // Don't proceed, loop forever

  }

  display.clearDisplay();
  display.setTextColor(WHITE);
  display.setTextSize(1);
  display.setCursor(0,0);
  display.print("LORA RECEIVER ");
  display.display();

  Serial.println("LoRa Receiver Test");

  //SPI LoRa pins
  SPI.begin(SCK, MISO, MOSI, SS);
  //setup LoRa transceiver module
  LoRa.setPins(SS, RST, DIO0);

  if (!LoRa.begin(BAND)) {
    Serial.println("Starting LoRa failed!");
```

```
    while (1);
  }
  Serial.println("LoRa Initializing OK!");
  display.setCursor(0,10);
  display.println("LoRa Initializing OK!");
  display.display();

 randomSeed(analogRead(0));

  delay(10);
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(ssid);

  wifiConnect();

  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
  Serial.println(WiFi.macAddress());
  stMac = WiFi.macAddress();
  stMac.replace(":", "_");
  Serial.println(stMac);

  client.setServer(mqttServer, port);
  client.setCallback(callback);
}

void wifiConnect() {
  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
}

void mqttReconnect() {
  while (!client.connected()) {
    Serial.print("Attempting MQTT connection...");
    long r = random(1000);
    sprintf(clientId, "clientId-%ld", r);
    if (client.connect(clientId)) {
      Serial.print(clientId);
      Serial.println(" connected");
      client.subscribe("topicName/led");
    } else {
```

```
      Serial.print("failed, rc=");
      Serial.print(client.state());
      Serial.println(" try again in 5 seconds");
      delay(5000);
    }
  }
}

void callback(char* topic, byte* message, unsigned int length) {
  Serial.print("Message arrived on topic: ");
  Serial.print(topic);
  Serial.print(". Message: ");
  String stMessage;

  for (int i = 0; i < length; i++) {
    Serial.print((char)message[i]);
    stMessage += (char)message[i];
  }
  Serial.println();

  if (String(topic) == "topicName/led") {
    Serial.print("Changing output to ");
    if(stMessage == "on"){
      Serial.println("on");
      digitalWrite(ledPin, HIGH);
    }
    else if(stMessage == "off"){
      Serial.println("off");
      digitalWrite(ledPin, LOW);
    }
  }
}

void loop() {

  //try to parse packet
  int packetSize = LoRa.parsePacket();
  if (packetSize) {
    //received a packet
    Serial.print("Received packet ");

    //read packet
    while (LoRa.available()) {
      LoRaData = LoRa.readString();
      Serial.print(LoRaData);
    }

    //print RSSI of packet
```

```
    int rssi = LoRa.packetRssi();
    Serial.print(" with RSSI ");
    Serial.println(rssi);

   // Dsiplay information
   display.clearDisplay();
   display.setCursor(0,0);
   display.print("LORA RECEIVER");
   display.setCursor(0,20);
   display.print("Received packet:");
   display.setCursor(0,30);
   display.print(LoRaData);
   display.setCursor(0,50);
   display.print("RSSI:");
   display.setCursor(30,50);
   display.print(rssi);
   display.display();
  }

  delay(10);
    if (!client.connected()) {
    mqttReconnect();
  }
  client.loop();

  long now = millis();
  if (now - lastMsg > 10000) {
  //if (now - lastMsg > 2000) {
    lastMsg = now;
    //++value;
    //snprintf (msg, 75, "hello world #%ld", value);
    LoRaData.toCharArray(msg, LoRaData.length()+1);
    Serial.print("Publish message: ");
    Serial.println(msg);
    client.publish("outTopicLora", msg);
  }
}
```

Lora sender Jembatan 1:

```
/*
lora32Sender_oled_gy521_v3_1s_v2_J1
*/

//Libraries for LoRa
#include <SPI.h>
#include <LoRa.h>

//Libraries for OLED Display
```

```
#include <Wire.h>
#include <Adafruit_MPU6050.h>
#include <Adafruit_Sensor.h>
//#include <MPU6050_light.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

//define the pins used by the LoRa transceiver module
#define SCK 5
#define MISO 19
#define MOSI 27
#define SS 18
#define RST 23//14
#define DIO0 26

//433E6 for Asia
//866E6 for Europe
//915E6 for North America
#define BAND 915E6

//OLED pins
#define OLED_SDA 21//4
#define OLED_SCL 22//15
#define OLED_RST 16
#define SCREEN_WIDTH 128 // OLED display width, in pixels
#define SCREEN_HEIGHT 64 // OLED display height, in pixels

Adafruit_MPU6050 mpu;
//packet counter
int counter = 0;
float x=0;
float v=0;

Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire,
OLED_RST);

void setup() {
  //initialize Serial Monitor
  Serial.begin(115200);

  //reset OLED display via software
  pinMode(OLED_RST, OUTPUT);
  digitalWrite(OLED_RST, LOW);
  delay(20);
  digitalWrite(OLED_RST, HIGH);

  //initialize OLED
  Wire.begin(OLED_SDA, OLED_SCL);
```

```
  if(!display.begin(SSD1306_SWITCHCAPVCC, 0x3c, false, false)) { // Address
0x3C for 128x32
    Serial.println(F("SSD1306 allocation failed"));
    for(;;); // Don't proceed, loop forever
  }

  display.clearDisplay();
  display.setTextColor(WHITE);
  display.setTextSize(1);
  display.setCursor(0,0);
  display.print("LORA SENDER ");
  display.display();

  Serial.println("LoRa Sender Test");

  //SPI LoRa pins
  SPI.begin(SCK, MISO, MOSI, SS);
  //setup LoRa transceiver module
  LoRa.setPins(SS, RST, DIO0);

  if (!LoRa.begin(BAND)) {
    Serial.println("Starting LoRa failed!");
    while (1);
  }
  Serial.println("LoRa Initializing OK!");
  display.setCursor(0,10);
  display.print("LoRa Initializing OK!");
  display.display();
while (!Serial)
    delay(10); // will pause Zero, Leonardo, etc until serial console opens

  Serial.println("Adafruit MPU6050 test!");

  // Try to initialize!
  if (!mpu.begin()) {
    Serial.println("Failed to find MPU6050 chip");
    while (1) {
      delay(10);
    }
  }
  Serial.println("MPU6050 Found!");

  mpu.setAccelerometerRange(MPU6050_RANGE_8_G);
  Serial.print("Accelerometer range set to: ");
  switch (mpu.getAccelerometerRange()) {
  case MPU6050_RANGE_2_G:
    Serial.println("+-2G");
    break;
```

```
case MPU6050_RANGE_4_G:
  Serial.println("+-4G");
  break;
case MPU6050_RANGE_8_G:
  Serial.println("+-8G");
  break;
case MPU6050_RANGE_16_G:
  Serial.println("+-16G");
  break;
}
mpu.setGyroRange(MPU6050_RANGE_500_DEG);
Serial.print("Gyro range set to: ");
switch (mpu.getGyroRange()) {
case MPU6050_RANGE_250_DEG:
  Serial.println("+- 250 deg/s");
  break;
case MPU6050_RANGE_500_DEG:
  Serial.println("+- 500 deg/s");
  break;
case MPU6050_RANGE_1000_DEG:
  Serial.println("+- 1000 deg/s");
  break;
case MPU6050_RANGE_2000_DEG:
  Serial.println("+- 2000 deg/s");
  break;
}

mpu.setFilterBandwidth(MPU6050_BAND_5_HZ);
Serial.print("Filter bandwidth set to: ");
switch (mpu.getFilterBandwidth()) {
case MPU6050_BAND_260_HZ:
  Serial.println("260 Hz");
  break;
case MPU6050_BAND_184_HZ:
  Serial.println("184 Hz");
  break;
case MPU6050_BAND_94_HZ:
  Serial.println("94 Hz");
  break;
case MPU6050_BAND_44_HZ:
  Serial.println("44 Hz");
  break;
case MPU6050_BAND_21_HZ:
  Serial.println("21 Hz");
  break;
case MPU6050_BAND_10_HZ:
  Serial.println("10 Hz");
  break;
```

```
  case MPU6050_BAND_5_HZ:
    Serial.println("5 Hz");
    break;
  }

  Serial.println("");
  delay(100);
}

void loop() {

  sensors_event_t a, g, temp;
  mpu.getEvent(&a, &g, &temp);
  x=constrain ((a.acceleration.y*90/9.78), -90, 90);
  Serial.print("x: ");
  Serial.println(x);
  v=sqrt(sq(g.gyro.x) + sq(g.gyro.y) + sq(g.gyro.z));
  Serial.print("v: ");
  Serial.println(v);

  //Serial.println(constrain ((a.acceleration.y*90/9.78), -90, 90));


  //Serial.print("Sending packet: ");
  //Serial.println(counter);

  //Send LoRa packet to receiver
  LoRa.beginPacket();
  //LoRa.print("hello ");
  //LoRa.print(counter);
  LoRa.print("J1 Miring: ");
  LoRa.print(x);
  LoRa.print(", Getar: ");
  LoRa.print(v);
  LoRa.endPacket();

  display.clearDisplay();
  display.setCursor(0,0);
  display.println("LORA SENDER");
  display.setCursor(0,20);
  display.setTextSize(1);
  display.print("LoRa packet sent.");
  display.setCursor(0,30);
  display.print("J1 Miring:");
  display.setCursor(50,30);
  display.print(x);
  display.setCursor(0,40);
  display.print("Getar:");
```

```
  display.setCursor(50,40);
  display.print(v);
  display.display();

  //counter++;

  //delay(10000);
  delay(1100);
}
```

LoRa Sender Jembatan 2:

```
/*
lora32Sender_oled_gy521_v3_1s_v2_J2
*/

//Libraries for LoRa
#include <SPI.h>
#include <LoRa.h>

//Libraries for OLED Display
#include <Wire.h>
#include <Adafruit_MPU6050.h>
#include <Adafruit_Sensor.h>
//#include <MPU6050_light.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>

//define the pins used by the LoRa transceiver module
#define SCK 5
#define MISO 19
#define MOSI 27
#define SS 18
#define RST 23//14
#define DIO0 26

//433E6 for Asia
//866E6 for Europe
//915E6 for North America
#define BAND 915E6

//OLED pins
#define OLED_SDA 21//4
#define OLED_SCL 22//15
#define OLED_RST 16
#define SCREEN_WIDTH 128 // OLED display width, in pixels
#define SCREEN_HEIGHT 64 // OLED display height, in pixels

Adafruit_MPU6050 mpu;
```

```
//packet counter
int counter = 0;
float x=0;
float v=0;

Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire,
OLED_RST);

void setup() {
  //initialize Serial Monitor
  Serial.begin(115200);

  //reset OLED display via software
  pinMode(OLED_RST, OUTPUT);
  digitalWrite(OLED_RST, LOW);
  delay(20);
  digitalWrite(OLED_RST, HIGH);

  //initialize OLED
  Wire.begin(OLED_SDA, OLED_SCL);
  if(!display.begin(SSD1306_SWITCHCAPVCC, 0x3c, false, false)) { // Address
0x3C for 128x32
    Serial.println(F("SSD1306 allocation failed"));
    for(;;); // Don't proceed, loop forever
  }

  display.clearDisplay();
  display.setTextColor(WHITE);
  display.setTextSize(1);
  display.setCursor(0,0);
  display.print("LORA SENDER ");
  display.display();

  Serial.println("LoRa Sender Test");

  //SPI LoRa pins
  SPI.begin(SCK, MISO, MOSI, SS);
  //setup LoRa transceiver module
  LoRa.setPins(SS, RST, DIO0);

  if (!LoRa.begin(BAND)) {
    Serial.println("Starting LoRa failed!");
    while (1);
  }
  Serial.println("LoRa Initializing OK!");
  display.setCursor(0,10);
  display.print("LoRa Initializing OK!");
  display.display();
```

```
while (!Serial)
    delay(10); // will pause Zero, Leonardo, etc until serial console opens

  Serial.println("Adafruit MPU6050 test!");

  // Try to initialize!
  if (!mpu.begin()) {
    Serial.println("Failed to find MPU6050 chip");
    while (1) {
      delay(10);
    }
  }
  Serial.println("MPU6050 Found!");

  mpu.setAccelerometerRange(MPU6050_RANGE_8_G);
  Serial.print("Accelerometer range set to: ");
  switch (mpu.getAccelerometerRange()) {
  case MPU6050_RANGE_2_G:
    Serial.println("+-2G");
    break;
  case MPU6050_RANGE_4_G:
    Serial.println("+-4G");
    break;
  case MPU6050_RANGE_8_G:
    Serial.println("+-8G");
    break;
  case MPU6050_RANGE_16_G:
    Serial.println("+-16G");
    break;
  }
  mpu.setGyroRange(MPU6050_RANGE_500_DEG);
  Serial.print("Gyro range set to: ");
  switch (mpu.getGyroRange()) {
  case MPU6050_RANGE_250_DEG:
    Serial.println("+- 250 deg/s");
    break;
  case MPU6050_RANGE_500_DEG:
    Serial.println("+- 500 deg/s");
    break;
  case MPU6050_RANGE_1000_DEG:
    Serial.println("+- 1000 deg/s");
    break;
  case MPU6050_RANGE_2000_DEG:
    Serial.println("+- 2000 deg/s");
    break;
  }

  mpu.setFilterBandwidth(MPU6050_BAND_5_HZ);
```

```
  Serial.print("Filter bandwidth set to: ");
  switch (mpu.getFilterBandwidth()) {
  case MPU6050_BAND_260_HZ:
    Serial.println("260 Hz");
    break;
  case MPU6050_BAND_184_HZ:
    Serial.println("184 Hz");
    break;
  case MPU6050_BAND_94_HZ:
    Serial.println("94 Hz");
    break;
  case MPU6050_BAND_44_HZ:
    Serial.println("44 Hz");
    break;
  case MPU6050_BAND_21_HZ:
    Serial.println("21 Hz");
    break;
  case MPU6050_BAND_10_HZ:
    Serial.println("10 Hz");
    break;
  case MPU6050_BAND_5_HZ:
    Serial.println("5 Hz");
    break;
  }

  Serial.println("");
  delay(100);
}

void loop() {

  sensors_event_t a, g, temp;
  mpu.getEvent(&a, &g, &temp);
  x=constrain ((a.acceleration.y*90/9.78), -90, 90);
  Serial.print("x: ");
  Serial.println(x);
  v=sqrt(sq(g.gyro.x) + sq(g.gyro.y) + sq(g.gyro.z));
  Serial.print("v: ");
  Serial.println(v);

  //Serial.println(constrain ((a.acceleration.y*90/9.78), -90, 90));


  //Serial.print("Sending packet: ");
  //Serial.println(counter);

  //Send LoRa packet to receiver
  LoRa.beginPacket();
```

```
  //LoRa.print("hello ");
  //LoRa.print(counter);
  LoRa.print("J2 Miring: ");
  LoRa.print(x);
  LoRa.print(", Getar: ");
  LoRa.print(v);
  LoRa.endPacket();

  display.clearDisplay();
  display.setCursor(0,0);
  display.println("LORA SENDER");
  display.setCursor(0,20);
  display.setTextSize(1);
  display.print("LoRa packet sent.");
  display.setCursor(0,30);
  display.print("J2 Miring:");
  display.setCursor(50,30);
  display.print(x);
  display.setCursor(0,40);
  display.print("Getar:");
  display.setCursor(50,40);
  display.print(v);
  display.display();

  //counter++;

  //delay(10000);
  delay(1200);
}
```

Lora sender jembatan 3:

```
/*
lora32Sender_oled_gy521_v3_1s_v2_J3
*/

//Libraries for LoRa
#include <SPI.h>
#include <LoRa.h>

//Libraries for OLED Display
#include <Wire.h>
#include <Adafruit_MPU6050.h>
#include <Adafruit_Sensor.h>
//#include <MPU6050_light.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
//define the pins used by the LoRa transceiver module
#define SCK 5
```

```
#define MISO 19
#define MOSI 27
#define SS 18
#define RST 23//14
#define DIO0 26

//433E6 for Asia
//866E6 for Europe
//915E6 for North America
#define BAND 915E6

//OLED pins
#define OLED_SDA 21//4
#define OLED_SCL 22//15
#define OLED_RST 16
#define SCREEN_WIDTH 128 // OLED display width, in pixels
#define SCREEN_HEIGHT 64 // OLED display height, in pixels

Adafruit_MPU6050 mpu;
//packet counter
int counter = 0;
float x=0;
float v=0;

Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire,
OLED_RST);

void setup() {
  //initialize Serial Monitor
  Serial.begin(115200);

  //reset OLED display via software
  pinMode(OLED_RST, OUTPUT);
  digitalWrite(OLED_RST, LOW);
  delay(20);
  digitalWrite(OLED_RST, HIGH);

  //initialize OLED
  Wire.begin(OLED_SDA, OLED_SCL);
  if(!display.begin(SSD1306_SWITCHCAPVCC, 0x3c, false, false)) { // Address
0x3C for 128x32
    Serial.println(F("SSD1306 allocation failed"));
    for(;;); // Don't proceed, loop forever
  }

  display.clearDisplay();
  display.setTextColor(WHITE);
  display.setTextSize(1);
```

```
  display.setCursor(0,0);
  display.print("LORA SENDER ");
  display.display();

  Serial.println("LoRa Sender Test");

  //SPI LoRa pins
  SPI.begin(SCK, MISO, MOSI, SS);
  //setup LoRa transceiver module
  LoRa.setPins(SS, RST, DIO0);

  if (!LoRa.begin(BAND)) {
    Serial.println("Starting LoRa failed!");
    while (1);
  }
  Serial.println("LoRa Initializing OK!");
  display.setCursor(0,10);
  display.print("LoRa Initializing OK!");
  display.display();
while (!Serial)
    delay(10); // will pause Zero, Leonardo, etc until serial console opens

  Serial.println("Adafruit MPU6050 test!");

  // Try to initialize!
  if (!mpu.begin()) {
    Serial.println("Failed to find MPU6050 chip");
    while (1) {
      delay(10);
    }
  }
  Serial.println("MPU6050 Found!");

  mpu.setAccelerometerRange(MPU6050_RANGE_8_G);
  Serial.print("Accelerometer range set to: ");
  switch (mpu.getAccelerometerRange()) {
  case MPU6050_RANGE_2_G:
    Serial.println("+-2G");
    break;
  case MPU6050_RANGE_4_G:
    Serial.println("+-4G");
    break;
  case MPU6050_RANGE_8_G:
    Serial.println("+-8G");
    break;
  case MPU6050_RANGE_16_G:
    Serial.println("+-16G");
    break;
```

```
  }
  mpu.setGyroRange(MPU6050_RANGE_500_DEG);
  Serial.print("Gyro range set to: ");
  switch (mpu.getGyroRange()) {
  case MPU6050_RANGE_250_DEG:
    Serial.println("+- 250 deg/s");
    break;
  case MPU6050_RANGE_500_DEG:
    Serial.println("+- 500 deg/s");
    break;
  case MPU6050_RANGE_1000_DEG:
    Serial.println("+- 1000 deg/s");
    break;
  case MPU6050_RANGE_2000_DEG:
    Serial.println("+- 2000 deg/s");
    break;
  }

  mpu.setFilterBandwidth(MPU6050_BAND_5_HZ);
  Serial.print("Filter bandwidth set to: ");
  switch (mpu.getFilterBandwidth()) {
  case MPU6050_BAND_260_HZ:
    Serial.println("260 Hz");
    break;
  case MPU6050_BAND_184_HZ:
    Serial.println("184 Hz");
    break;
  case MPU6050_BAND_94_HZ:
    Serial.println("94 Hz");
    break;
  case MPU6050_BAND_44_HZ:
    Serial.println("44 Hz");
    break;
  case MPU6050_BAND_21_HZ:
    Serial.println("21 Hz");
    break;
  case MPU6050_BAND_10_HZ:
    Serial.println("10 Hz");
    break;
  case MPU6050_BAND_5_HZ:
    Serial.println("5 Hz");
    break;
  }

  Serial.println("");
  delay(100);
}
```

```
void loop() {

  sensors_event_t a, g, temp;
  mpu.getEvent(&a, &g, &temp);
  x=constrain ((a.acceleration.y*90/9.78), -90, 90);
  Serial.print("x: ");
  Serial.println(x);
  v=sqrt(sq(g.gyro.x) + sq(g.gyro.y) + sq(g.gyro.z));
  Serial.print("v: ");
  Serial.println(v);

  //Serial.println(constrain ((a.acceleration.y*90/9.78), -90, 90));


  //Serial.print("Sending packet: ");
  //Serial.println(counter);

  //Send LoRa packet to receiver
  LoRa.beginPacket();
  //LoRa.print("hello ");
  //LoRa.print(counter);
  LoRa.print("J3 Miring: ");
  LoRa.print(x);
  LoRa.print(", Getar: ");
  LoRa.print(v);
  LoRa.endPacket();

  display.clearDisplay();
  display.setCursor(0,0);
  display.println("LORA SENDER");
  display.setCursor(0,20);
  display.setTextSize(1);
  display.print("LoRa packet sent.");
  display.setCursor(0,30);
  display.print("J3 Miring:");
  display.setCursor(50,30);
  display.print(x);
  display.setCursor(0,40);
  display.print("Getar:");
  display.setCursor(50,40);
  display.print(v);
  display.display();

  //counter++;

  //delay(10000);
  delay(1300);
}
```

Lampiran 5 Data sheet MPU6050

10 Serial Interface Considerations (MPU-6050)

10.1 MPU-6050 Supported Interfaces

The MPU-6050 supports I^2C communications on both its primary (microprocessor) serial interface and its auxiliary interface.

10.2 Logic Levels

The MPU-6050's I/O logic levels are set to be either VDD or VLOGIC, as shown in the table below.

I/O Logic Levels vs. *AUX_VDDIO*

AUX_VDDIO	MICROPROCESSOR LOGIC LEVELS (Pins: SDA, SCL, AD0, CLKIN, INT)	AUXILLARY LOGIC LEVELS (Pins: AUX_DA, AUX_CL)
0	VLOGIC	VLOGIC
1	VLOGIC	VDD

Note: The power-on-reset value for *AUX_VDDIO* is 0.

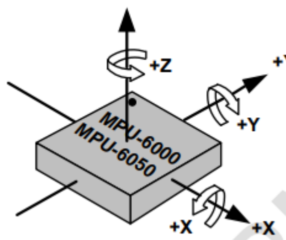
VLOGIC may be set to be equal to VDD or to another voltage. However, VLOGIC must be ≤ VDD at all times. When *AUX_VDDIO* is set to 0 (its power-on-reset value), VLOGIC is the power supply voltage for both the microprocessor system bus and the auxiliary I^2C bus, as shown in the figure of Section 10.3. When *AUX_VDDIO* is set to 1, VLOGIC is the power supply voltage for the microprocessor system bus and VDD is the supply for the auxiliary I^2C bus, as shown in the figure of Section 10.4.

11 Assembly

This section provides general guidelines for assembling InvenSense Micro Electro-Mechanical Systems (MEMS) gyros packaged in Quad Flat No leads package (QFN) surface mount integrated circuits.

11.1 Orientation of Axes

The diagram below shows the orientation of the axes of sensitivity and the polarity of rotation. Note the pin 1 identifier (•) in the figure.



Orientation of Axes of Sensitivity and Polarity of Rotation

Lampiran 6 Data sheet MQTT

7.1.1 MQTT Server

An MQTT Server conforms to this specification only if it satisfies all the statements below:

1. The format of all Control Packets that the Server sends matches the format described in Chapter 2 and Chapter 3.
2. It follows the Topic matching rules described in Section 4.7.
3. It satisfies all of the MUST level requirements in the following chapters that are identified except for those that only apply to the Client:

- Chapter 1 - Introduction
- Chapter 2 - MQTT Control Packet format
- Chapter 3 - MQTT Control Packets
- Chapter 4 - Operational behavior
- Chapter 6 - (if MQTT is transported over a WebSocket connection)
- Chapter 7 - Conformance Targets

A conformant Server MUST support the use of one or more underlying transport protocols that provide an ordered, lossless, stream of bytes from the Client to Server and Server to Client [MQTT-7.1.1-1]. However conformance does not depend on it supporting any specific transport protocols. A Server MAY support any of the transport protocols listed in Section 4.2, or any other transport protocol that meets the requirements of [MQTT-7.1.1-1].

7.1.2 MQTT Client

An MQTT Client conforms to this specification only if it satisfies all the statements below:

1. The format of all Control Packets that the Client sends matches the format described in Chapter 2 and Chapter 3.
2. It satisfies all of the MUST level requirements in the following chapters that are identified except for those that only apply to the Server:

- Chapter 1 - Introduction
- Chapter 2 - MQTT Control Packet format
- Chapter 3 - MQTT Control Packets
- Chapter 4 - Operational behavior
- Chapter 6 - (if MQTT is transported over a WebSocket connection)

A conformant Client MUST support the use of one or more underlying transport protocols that provide an ordered, lossless, stream of bytes from the Client to Server and Server to Client [MQTT-7.1.2-1]. However conformance does not depend on it supporting any specific transport protocols. A Client MAY support any of the transport protocols listed in Section 4.2, or any other transport protocol that meets the requirements of [MQTT-7.1.2-1].

Lampiran 7 Data sheet ESP 32 LoRa dan IoT

Wi-Fi

• 802.11b/g/n • 802.11n (2.4 GHz), up to 150 Mbps • WMM • TX/RX A-MPDU, RX A-MSDU • Immediate Block ACK • Defragmentation • Automatic Beacon monitoring (hardware TSF) • Four virtual Wi-Fi interfaces • Simultaneous support for Infrastructure Station, SoftAP, and Promiscuous modes Note that when ESP32 is in Station mode, performing a scan, the SoftAP channel will be changed. • Antenna diversity

Bluetooth®

• Compliant with Bluetooth v4.2 BR/EDR and Bluetooth LE specifications • Class-1, class-2 and class-3 transmitter without external power amplifier • Enhanced Power Control • +9 dBm transmitting power • NZIF receiver with –94 dBm Bluetooth LE sensitivity • Adaptive Frequency Hopping (AFH) • Standard HCI based on SDIO/SPI/UART • High-speed UART HCI, up to 4 Mbps • Bluetooth 4.2 BR/EDR and Bluetooth LE dual mode contro• Synchronous Connection-Oriented/Extended (SCO/eSCO) • CVSD and SBC for audio codec • Bluetooth Piconet and Scatternet • Multi-connections in Classic Bluetooth and Bluetooth LE • Simultaneous advertising and scanning

CPU and Memory

• Xtensa® single-/dual-core 32-bit LX6 microprocessor(s) • CoreMark® score: – 1 core at 240 MHz: 504.85 CoreMark; 2.10 CoreMark/MHz• 448 KB ROM • 520 KB SRAM • 16 KB SRAM in RTC • QSPI supports multiple flash/SRAM chips

Clocks and Timers

• Internal 8 MHz oscillator with calibration • Internal RC oscillator with calibration • External 2 MHz ~ 60 MHz crystal oscillator (40 MHz only for Wi-Fi/Bluetooth functionality) • External 32 kHz crystal oscillator for RTC with calibration • Two timer groups, including 2 × 64-bit timers and 1 × main watchdog in each group • One RTC timer • RTC watchdog

Advanced Peripheral Interfaces

• 34 programmable GPIOs – Five strapping GPIOs – Six input-only GPIOs – Six GPIOs needed for in-package flash/PSRAM (ESP32-D0WDR2-V3, ESP32-U4WDH) • 12-bit SAR ADC up to 18 channels • Two 8-bit DAC • 10 touch sensors • Four SPI interfaces• Two I2S interfaces • Two I2C interfaces • Three UART interfaces • One host (SD/eMMC/SDIO) • One slave (SDIO/SPI) • Ethernet MAC interface with dedicated DMA and IEEE 1588 support • TWAI®, compatible with ISO 11898-1 (CAN Specification 2.0) • RMT (TX/RX) • Motor PWM • LED PWM up to 16 channels

Power Management

• Fine-resolution power control through a selection of clock frequency, duty cycle, Wi-Fi operating modes, and individual power control of internal components • Five power modes designed for typical scenarios: Active, Modem-sleep, Light-sleep, Deep-sleep, Hibernation • Power consumption in Deep-sleep mode is 10 µA • Ultra-Low-Power (ULP) coprocessors • RTC memory remains powered on in Deep-sleep mode Security • Secure boot • Flash encryption • 1024-bit OTP, up to 768-bit for customers • Cryptographic hardware acceleration: – AES – Hash (SHA-2) – RSA – ECC – Random Number Generator (RNG)

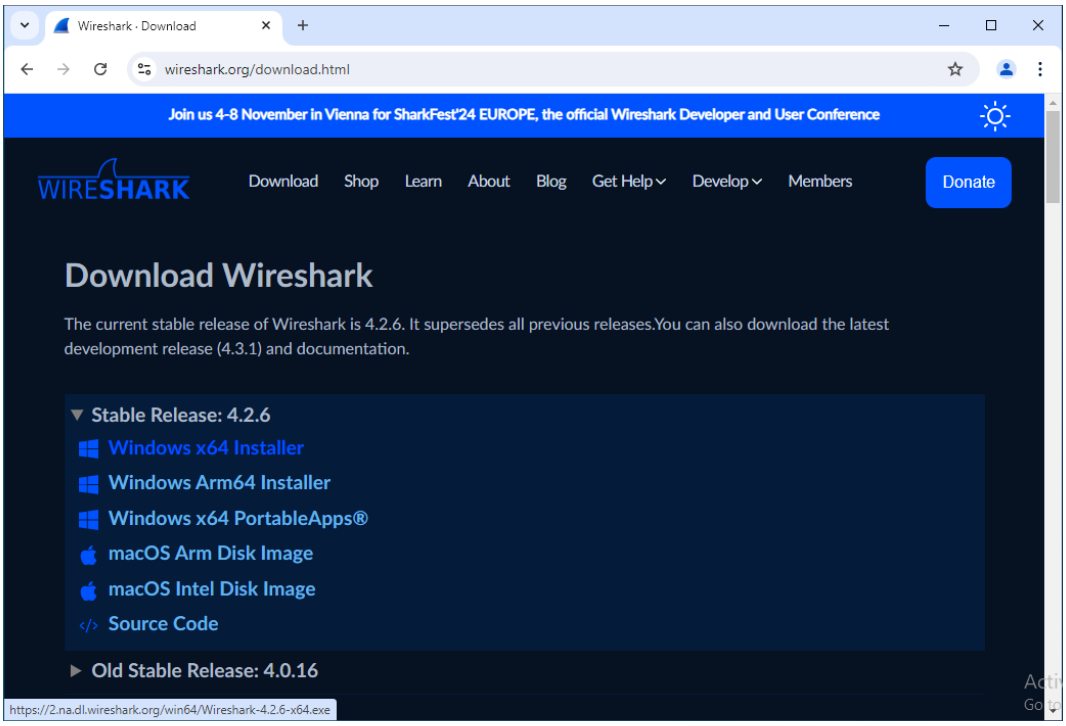
LoRa

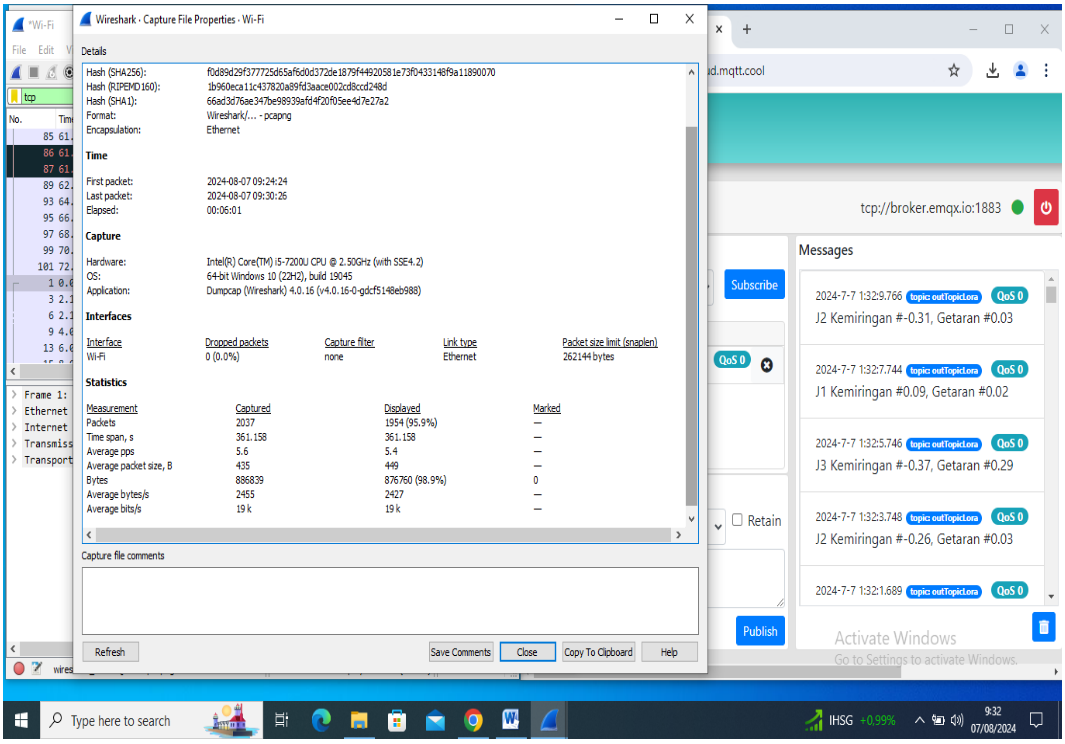
Working voltage: 1.8~3.7v
Acceptable current: 10~14mA
Transmit current: 120mA@+20dBm
90mA@+17dBm
29mA@+13dBm
Operating frequency: 433/868/915MHz
Transmit power: +20dBm
Receive sensitivity: -139dBm@LoRa &62.5Khz&SF=12&146bps
-136dBm@LoRa &125Khz&SF=12&293bps
-118dBm@LoRa &125Khz&SF=6&9380bps
-123dBm@FSK&5Khz&1.2Kbps
Frequency error: +/-15KHz
FIFO space: 64Byte
Data rate: 1.2K~300Kbps@FSK
0.018K~37.5Kbps@LoRa
Modulation Mode: FSK,GFSK,MSK,GMSK,LoRa TM， OOK
Interface form: SPI
Operating temperature: -40℃- +85℃

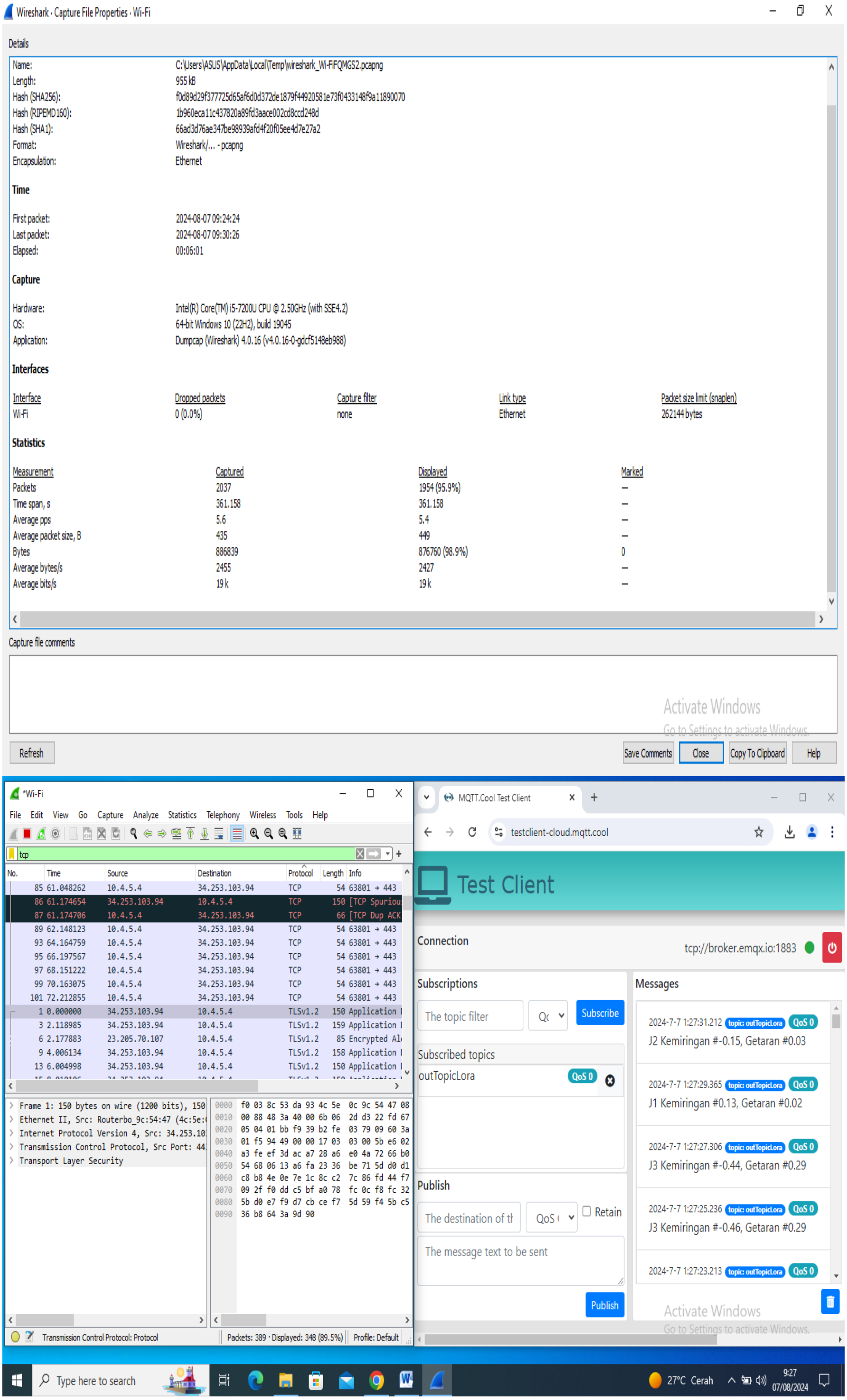
Digital RSSI function

Automatic frequency correction
Automatic gain control
Fast wake-up and frequency hopping
Highly configurable data packet handler
SMA Antenna
IP5306

Lampiran 8 Data Wireshark






Wireshark · Capture File Properties · Wi-Fi
Details
Name: C:\Users\ASUS\AppData\Local\Temp\wireshark_Wi-FiFQMGS2.pcapng
Length: 955 kB
Format: Wireshark/... - pcapng
Encapsulation: Ethernet
Time
First packet: 2024-08-07 09:24:24
Last packet: 2024-08-07 09:30:26
Elapsed: 00:06:01
Capture
Hardware: Intel(R) Core(TM) i5-7200U CPU @ 2.50GHz (with SSE4.2)
OS: 64-bit Windows 10 (22H2), build 19045
Application: Dumpcap (Wireshark) 4.0.16 (v4.0.16-0-gdcf5148eb988)
Interfaces
Interface Dropped packets Capture filter Link type Packet size limit (snaplen)
Wi-Fi 0 (0.0%) none Ethernet 262144 bytes
Statistics
Measurement Captured Displayed Marked
Packets 2037 1954 (95.9%)
Time span, s 361.158 361.158
Average pps 5.6 5.4
Average packet size, B 435 449
Bytes 886839 876760 (98.9%) 0
Average bytes/s 2455 2427
Average bits/s 19 k 19 k
Capture file comments
Refresh
Save Comments
Close
Copy To Clipboard
Help
*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
tcp
No. Time Source Destination Protocol Length Info
Frame 1: 150 bytes on wire (1200 bits), 150
Ethernet II, Src: Routerbo_9c:54:47 (4c:5e:
Internet Protocol Version 4, Src: 34.253.10
Transmission Control Protocol, Src Port: 44
Transport Layer Security
Transmission Control Protocol: Protocol
Packets: 389 · Displayed: 348 (89.5%)
Profile: Default
MQTT.Cool Test Client
testclient-cloud.mqtt.cool
Test Client
Connection
tcp://broker.emqx.io:1883
Subscriptions
The topic filter
Subscribe
Subscribed topics
outTopicLora
QoS 0
Messages
2024-7-7 1:27:31.212
J2 Kemiringan #-0.15, Getaran #0.03
2024-7-7 1:27:29.365
J1 Kemiringan #0.13, Getaran #0.02
2024-7-7 1:27:27.306
J3 Kemiringan #-0.44, Getaran #0.29
2024-7-7 1:27:25.236
J3 Kemiringan #-0.46, Getaran #0.29
2024-7-7 1:27:23.213
Publish
The destination of th
Retain
The message text to be sent
Publish
Activate Windows
Go to Settings to activate Windows.
Type here to search
27°C Cerah
9:27
07/08/2024

Lampiran 9 Daftar Riwayat Hidup

I. IDENTITAS DIRI

1.1	Nama	Nuraeni Umar, S.T, M.T
1.2	Jenis Kelamin	Perempuan
1.3	Jabatan Fungsional	Pembina Utama Muda/ IV c
1.4	NIP	19620912 198803 2 004
1.5	NIDN	0012096206
1.6	TempatdanTgl lahir	Ujung Pandang / 12 September 1962
1.7	E-mail	aeni12345@yahoo.com
1.8	Nomor Telepon/HP	+62 85299791290
1.9	Perguruan Tinggi	Politeknik Negeri Ujung Pandang Dosen sejak 1987 sampai sekarang
1.10	Alamat Kantor	Jl. Perintis Kemerdekaan KM 10 Tamalanrea Makassar, (e-mail) pnup@poliupg.ac.id
1.11	Nomor Telepon/Faks	(0411) 585 365 / (0411)586 043

II. RIWAYAT PENDIDIKAN

	D3	**S1**	**S1**	**S-2**
Nama Perguruan Tinggi	Universitas Hasanuddin	Universitas Muslim Indonesia	Universitas Hasanuddin	Institute Teknologi Bandung (ITB)
Bidang Ilmu	Telekomunikasi	Telekomunikasi	Elektro	Telekomunikasi
Tahun Masuk-Lulus	1982-1986	1989-1994	1992-1996	2002-2005
Gelar	Ahli Madya	Sarjana Teknik (S.T)	Sarjana Teknik (S.T)	Master Teknik (M.T)
Judul Skripsi/Tesis/ Disertasi	Studi pada TVRI Ujung Pandang	Perencanaan komunikasi *very high frequency* (VHF) di Kab. Selayar	Studi penangkal petir gedung PLN Wilayah VII.	Wavelet sebagai *pulse shaping filter* pada BPSK dan QPSK

III. PUBLIKASI ARTIKEL ILMIAH

No	**Tahun**	**Judul**	**Jurnal**
1.	2024	Internet of Things and Long Range based bridge Vibration	International Journal of Electrical and Computer Engineering (IJECE)
2.	2024	Internet of Things and Long Range Based Bridge Slope Early Detection Systems	International Journal of Reconfigurable and Embedded Systems (IJRES), Vol. 13, No. 3, November 2024,pp.652~658, ISSN: 2089-4864.

No	Tahun	Judul	Jurnal
3.	2024	Internet of Things Based on Bridge Slope Detection.	International Conference on Computer Science and Engineering Technology 2022 (ICCSET/ AIP 23 Juli 2024)
4.	2023	Sensor MQ-2 Deteksi Asap Rokok Berbasis Internet of Things	Jurnal ELEKTERIKA Vol 2 2023
5.	2023	Penggunaan Sensor Fr-04 Untuk Atap Jemuran	Jurnal ELEKTERIKA Vol 1 2023
6.	2020	Desain Dan Implementasi Interface Wireless Berbasis Internet of Things	SNP2M
7.	2019	Sistem Pencacah Adaptif dg Pintu Otomatis berbasis Mikrokontroler.	SNTEI

IV. PENGALAMAN PENGABDIAN KEPADA MASYARAKAT

No	Tahun	Judul Pengabdian Kepada Masyarakat	Pendanaan Sumber
1.	2024	PKM Pelatihan Pembuatan Pakan Kambing di desa pucak Kecamatan tanralili kabupaten Maros.	DIPA PNUP
2.	2023	Edukasi Pengelolaan Sampah pada Masyarakat Disepanjang Sungai Pute Rammang Rammang Di kampungMassaloeng Desa Salenrang Kecamatan Bontoa Kabupaten Maros.	DIPA PNUP
3.	2022	Pelatihan Pemanfaatan Kotoran Sapi Menjadi Bahan Bakar Bio Gas yg Ramah Ligkungan Program PPDM di Kampung Massaleang Desa Salenrang, Kec. Bontoa Kabupaten Maros	DIPA PNUP
4.	2021	Pelatihan Peningkatan Budaya K3 Transportasi Wisata Sungai di Dusun Rammang-Rammang, Desa Salenrang, Kec. Maros	DIPA PNUP
5.	2020	Pemasangan Instalasi Air Bersih pada Mushallah Melalui PPDM Pembangunan Mushallah Di Massaleang Desa Salenrang Maros	DIPA PNUP
	2019	PKM Instalasi Listrik Paud Dusun Mambue Nisombalia di Desa Binaan.	DIPA PNUP

V. PEMAKALAH SEMINAR ILMIAH

No	Tahun	Judul Artikel Ilmiah	Tema Seminar	Penyelenggara	Tempat & Waktu
1.	2022	Internet of Things Based on Bridge Slope Detection.	The Role of Industry 5.0, Innovation and Infrastructure for SDGs.	International Conference on Computer Science and Engineering Technology 2022 (ICCSET)	Universitas Muria Kudus 26 Nov 2022 (On Line)