Prototype of Data Acquisition Module

Aldi Roby Syafriansyah[1] and Rifqi Amalya Fatekha[2]

[1,2] Politeknik Negeri Batam, Batam Kepulauan Riau 29461, Indonesia
[1] aldirobyy@gmail.com
[2] rifqi@polibatam.ac.id

Abstract. Microcontrollers are common components used to control electronic systems. To implement a microcontroller according to the desired scheme, a program or sketch upload process is required. If there is a change in the desired program scheme, the developer must modify the program and reupload the sketch to the microcontroller board. This is not a problem if the uploaded program sketch is not complicated, but if the uploaded program is complicated it can cause the microcontroller not responding / freeze due to limited specifications that can only perform low computations. In order to reduce the burden borne by the microcontroller, the us-er programming scheme is carried out by the PC, by instructing the microcontroller to receive commands from the PC and forwarding the readings from the input pins to the PC by means of UDP communication, so the microcontroller is con-verted to function as a DAQ (Data Acquisition). This research uses Arduino Mega 2560 as a controller, Wiznet W5500 Lite, as a communication module to the PC. With the concept of data acquisition, there is a communication latency be-tween the DAQ device and the PC, in this study the average latency obtained is 4.99 ms. The data acquisition system studied also has a switching component, the switching used is optocoupler because optocoupler switching has a stable voltage output. In this way, it can allow controlling electronic systems with high computation, and has a stable output according to the supply given to the output pin.

Keywords: Data Acquisition, UDP Communication, Controller, Arduino.

1 Introduction

Microcontrollers are common components used to control electronic systems. [1], [2], [3]. To implement the microcontroller according to the desired scheme, a program or sketch upload process is required. If there is a change in the desired program scheme, the developer must modify the program and reupload the sketch to the microcontroller board. This is not a problem if the uploaded program sketch is not complicated, but if the uploaded program is complicated it can cause the microcontroller not responding / freezing due to the limited specifications that can only do low computations.

Data Acquisition is one type of device that allows computers to function as electronic system controllers [4], [5]. The role of the data acquisition system is to connect the computer with external electronic devices so that the computer can read data from these electronic devices and provide signals to other electronic devices [6], [7], [8]. There are

L. Lumombo et al. (eds.), *Proceedings of the 7th International Conference on Applied Engineering (ICAE 2024)*,
Advances in Engineering Research 251,
https://doi.org/10.2991/978-94-6463-620-8_22

many data acquisition devices available on the market, such as LabJack, National Instruments, and Measurement Computing. However, some of them have unaffordable prices and use Universal Serial Bus (USB) connectivity to the computer. For example, the Labjack U3 has a maximum voltage of 3.6VDC at the I/O pins. [9] and the National Instrument USB-6009 has a maximum voltage of 5.8V. With the switching concept, this can increase the flexibility of data acquisition I/O. To connect a data acquisition device (DAQ) with a master controller (PC), efficient communication is required. Nowadays, communication technology has ad-vanced, and computers can communicate with other computers through computer networks. [10], [11]. There are many communication protocols available for data communication in computer networks. In this research, we will use a UDP-based communication protocol. The center of this research is to develop the microcontroller into a data acquisition device (DAQ) using Arduino Mega as the DAQ controller. The data acquisition design will use a Wiznet W5500 module to connect the microcontroller to a wired computer network, with the PC acting as the main controller in this system.

2 Method

In order to reduce the burden borne by the microcontroller, the user programming scheme is done by a PC, so the microcontroller is only a bridge that connects the sensor and actuator devices to the PC. System testing was carried out using Arduino MEGA 2560, Wiznet W5500 Lite and PC.

Fig. 1. Arduino Mega 2560

Arduino Mega 2560 functions as a data acquisition processing unit, Arduino mega has a clock speed of 16MHz, Arduino Mega 2560 has many I / O pins as many as 54 Pins (15 PWM pins), so it has good system flexibility.

Fig. 2. Wiznet W5500 Lite

The component in Figure 2, has a function as a microcontroller connector to the internet network through the ethernet port that has been provided, with SPI connectivity, the communication speed is 80MHz. The Arduino MEGA 2560 is connected to the PC using an ethernet cable connected through the W5500 Lite connected to the Arduino MEGA 2560. PC and Arduino MEGA 2560 communicate using UDP proto-col. Here is a picture of the system diagram,

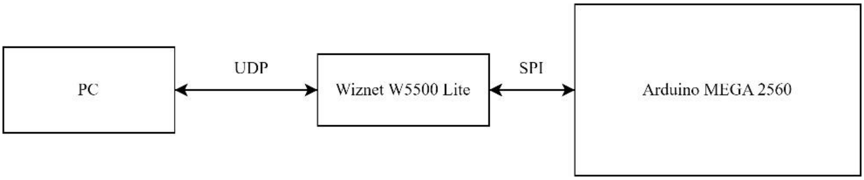

Fig. 3. Diagram of data acquisition system

In **Fig. 3**, the Arduino MEGA 2560 is connected to a PC using an ethernet cable con-nected through a W5500 Lite connected to the Arduino MEGA 2560. The PC and Ar-duino MEGA 2560 communicate using the UDP protocol, With the above components, here is an overview of the data acquisition built.

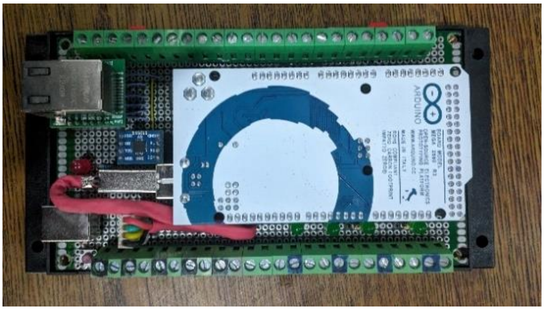

Fig. 4. Inside view of data acquisition

Fig. 5. Outside view of data acquisition

The data acquisition I/O circuit built is as follows

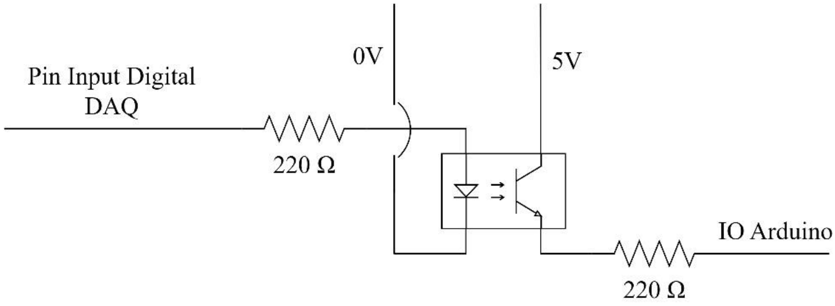

Fig. 6. Data acquisition digital input circuit

The data acquisition built has 8 digital input pins, in **Fig. 6**, the digital input circuit uses an optocoupler which functions as a switch that gives 5V to the Arduino pin triggered by the sensor from the data acquisition digital input pin, the optocoupler used is PC817, so that Arduino can detect the condition of the Data acquisition digital input pin.

8 Analog input pins, in analog input, the analog input pin terminal is directly connected to the analog pin on the Arduino, so the limitation that can only be read by the analog pin is 0V to 5V.

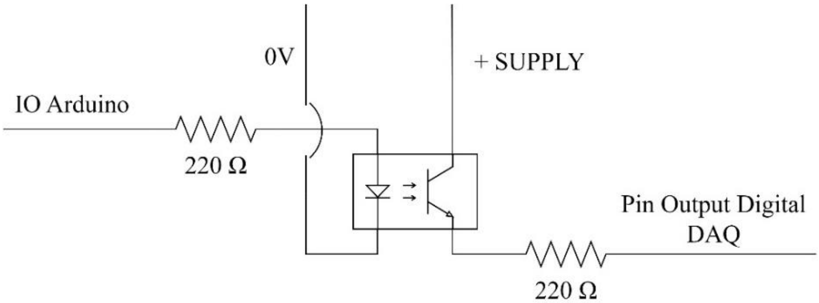

Fig. 7. Digital output circuit and PWM

The data acquisition output pins are 8 digital output pins, 8 pwm output pins. Seen in **Fig. 7**, the data acquisition digital output pin is triggered by the Arduino IO, so that if the Arduino IO pin is active (HIGH) then the data acquisition output pin will get a supply.

In order for the system to run, the following process flow diagram is used.

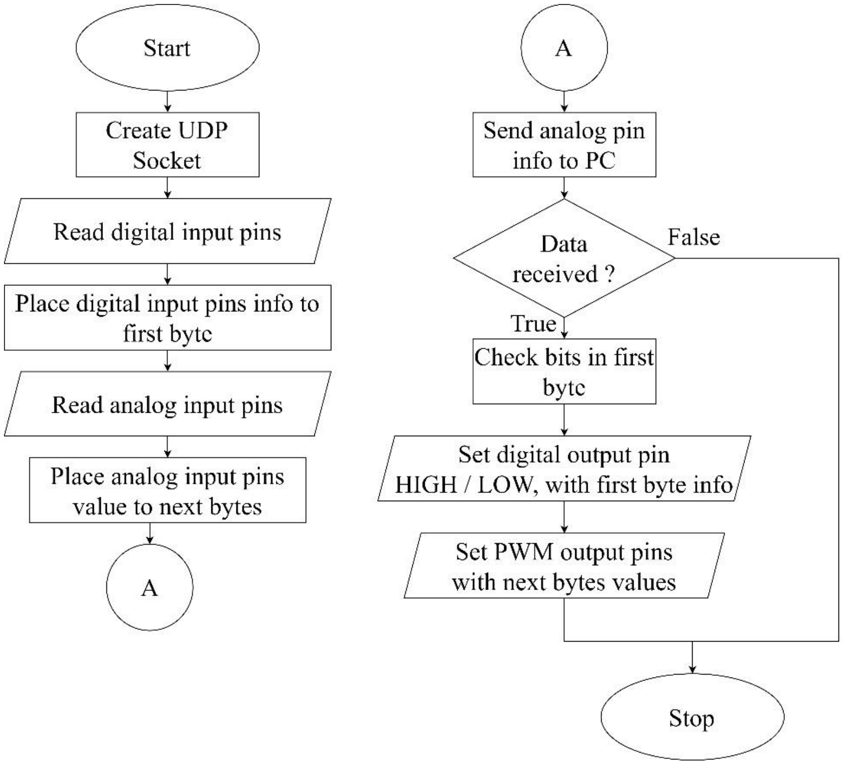

Fig. 8. Data Acquisition flowchart

Based on **Fig. 8** at startup, the data acquisition makes a UDP socket, then reads the digital input pins totaling 8 pins, the information of 8 digital input pins is arranged in one byte, after that read the analog input pins totaling 8 pins, 1 analog pin has a size of 1 byte, the arrangement of the analog input pin information is arranged after the first byte containing information on 8 digital input pins, after arranging all digital and analog input information, the information is sent to the PC. After that, data acquisition checks whether there is data received from the PC, if there is, the data sent is used as a reference for digital and analog outputs.

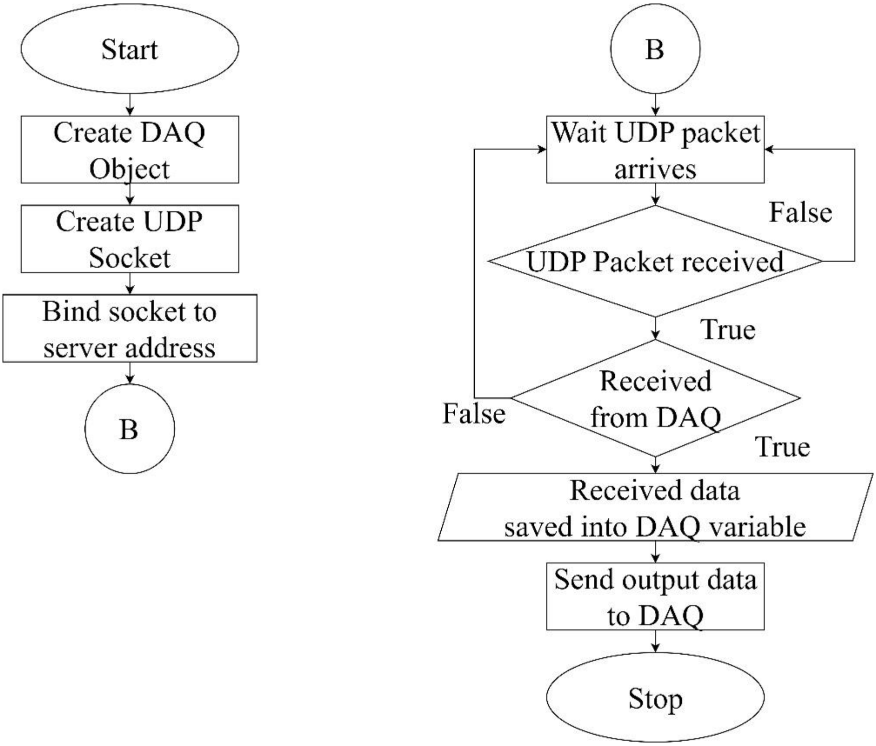

Fig. 9. PC Flowchart

On the PC part, based on **Fig. 8**, the PC must create a data acquisition object, then create a UDP socket, the PC receives a UDP packet, then the PC checks whether the packet comes from the data acquisition IP, if it is correct, then the packet is saved, then forwards the digital output and analog output data to the data acquisition. Here is the link to the data acquisition library used https://github.com/Karuizawaa/BiDAQ.

2.1 User Datagram Protocol (UDP)

UDP, which stands for User Datagram Protocol, is a TCP/IP transport layer protocol that supports unreliable, connectionless communication between hosts on net-works using TCP/IP [12], [13], [14]. UDP is a protocol that allows software on a computer to send messages to other computers over a network without the need for initial communication. [15], [16]. UDP does not guarantee that the data sent will reach the destination safely and completely, but UDP is faster than TCP. [17], [18]. UDP works by sending data in the form of datagrams, which are small data packets that do not have complex headers. [19], [20]. A UDP datagram consists of two parts, the UDP header and the data. [21], [22]. UDP communication can use more than two devices, the way UDP works in general is a server-client relationship, one side of the device must function as a server, and the other device can be a client. [23], [24]. In this re-search, the PC acts as a server and the DAQ device acts as a client. Communication testing is done by calculating the communication time.

2.2 Switching

Arduino mega 2560 has an I / O voltage specification of 5VDC, and I / O Cur-rent per pin of 20mA, so that the I / O can be flexible, a switching is needed based on the Arduino I / O. [25], [26]. There are several electronic components that can function as switching such as relays, transistors, and optocouplers. The electronic switch components above have different ways of working, and have their respective ad-vantages and disadvantages. Relay is an electronic switch component that works mechanically, the relay concept is a coil and contact, which works by shifting the contact com-contact due to the influence of the coil which is electrified. Optocou-plers, also known as optoisolators, are semiconductor devices that allow electrical signals to be transmitted between two electrically isolated circuits. An optocoupler consists of two main parts, an LED and a photodetector. The LED emits light when given an electric current. This light is then captured by the photodetector, which con-verts it into an electrical signal. This electrical signal can then be used to control other devices in the second circuit. Transistor is an electronic switch component that works semiconductor, Transistor based on the way of control is divided into two, namely Bipolar Junction Transistor (BJT) and Field Effect Transistor (FET). BJT transistor legs consist of Emitter, Collector, and Base, while the FET transistor legs consist of Drain, Source, and Gate. The working concept of the transistor is as a current tap controlled from the base leg. BJT works by controlling switching based on the current passing through the Base leg, while FET works by controlling the voltage passing through the Gate leg. The advantage of the transistor is that the time required for switching is short, but the disadvantage of the FET is that the current that can flow is not as large as the relay.

3 Result and Discussion

The data acquisition product to be created will have 8 digital output pins, 8 analog output pins, 8 digital input pins, 8 analog input pins. In order for communication to be efficient, the data sent is made concise in such a way. The data sent from the acquisition data to the PC is 9 bytes, the first byte contains information of 8 digital input pins, and the remaining 8 bytes are value information of 8 analog input pins, the data sent from the PC to the acquisition data is 9 bytes, the first byte contains information of 8 digital output pins, and the remaining 8 bytes contain information of 8 output pwm pins. Testing of the acquisition data is done by connecting the PC to the acquisition data directly using an ethernet cable. Tests were carried out to obtain PC communication latency with acquisition data, output testing and input testing.

3.1 Latency Testing

Latency testing is done to determine the responsiveness of data acquisition, by calculating the time it takes to send and receive at once to one data acquisition, latency calculation is assisted using the chrono library. Here are the latency results from 10 times looping:

Table 1. Communication Latency

Iteration	Latency (millisecond)
1	1.320536
2	5.351233
3	5.699955
4	5.403788
5	5.090760
6	5.377483
7	5.802971
8	5.392597
9	5.035795
10	5.521856

Based on the data results from **Table 1.** Communication Latency, the average latency obtained is 4.9996974 ms, with this latency the communication frequency obtained is around 200 Hz, or the responsiveness of data acquisition can be said to be 200 commands in one second.

3.2 Switching

The switch is an electronic component that functions as a switch but is electronically controlled, with the switch there are advantages obtained, namely that it can distribute electricity with higher power by recognizing switching tests carried out by providing PWM output to the controller. When you want to build data acquisition there is one type of switch that is used, namely the PC817 optocoupler. Switching testing uses a Tektronix TDS-1012C-EDU oscilloscope. This test aims to determine the characteristics of the switch component. The switching test uses an optocoupler device with part number PC817, the following is a schematic of the PC817 test circuit.

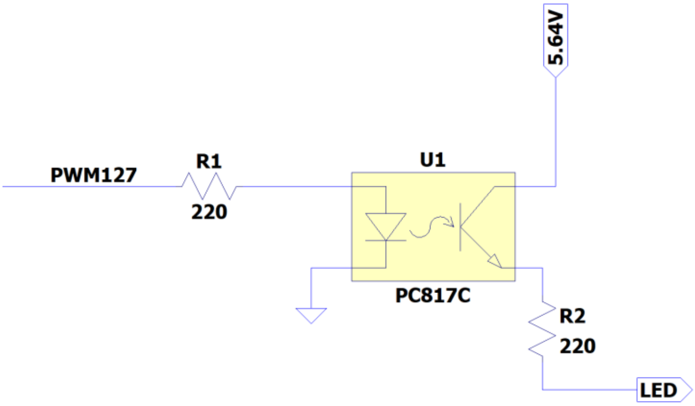

Fig. 10. PC817 test schematic

Testing the output results of the **Fig. 10** circuit, on the anode leg is given input from the Arduino PWM, and given a resistor of 220 Ohm, on the collector leg there is a 220 Ohm resistor and output line, here are the results of the test on the circuit above.

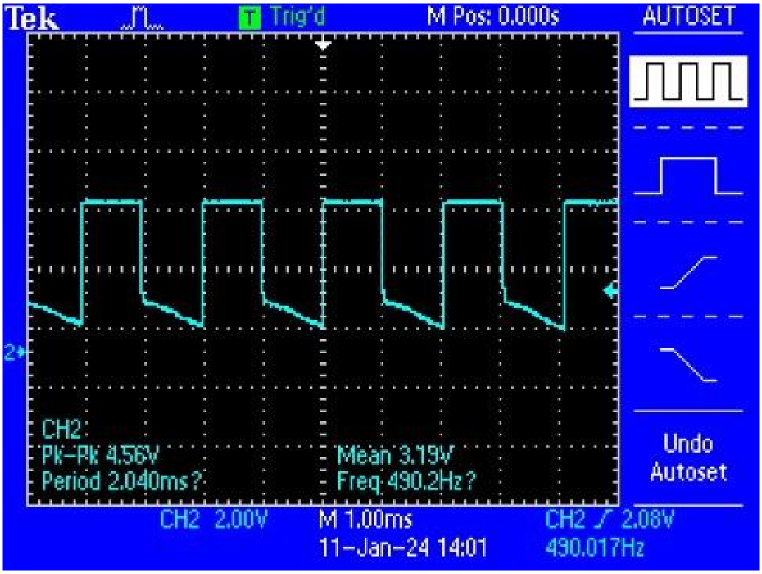

Fig. 11. PC817 testing resistor LED voltage

Testing in **Fig. 11**, the positive probe is positioned between the collector leg and R2, the negative probe is positioned after the LED. Based on the results of the **Fig. 11** test, the peak of the LED and resistor voltage is 4.56V when the pulse is on, the volt-age rises quickly as without transition, while when the pulse is off, there is a transition to inactivity. This is because the optocoupler works using the concept of light and light capture as switching.

References

1. Ronaldo, P. J., Wijaya, I., Menrisal, M.: Perancangan Media Pembelajaran Pengenalan Mikrokontroler Menggunakan Augmented Reality pada Mata Pelajaran Sistem Komputer. *Journal of Research and Investigation in Education*, pp. 70–74, Sep. 2023. doi: 10.37034/residu.v1i3.150.
2. Horowitz, P., Hill, W.: *The Art of Electronics*. Cambridge University Press, 2015.
3. Asadi, F.: Digital Input/Output (I/O). In: *Essentials of Arduino Boards Programming: Step-by-Step Guide to Master Arduino Boards Hardware and Software*, pp. 33–126. Apress, Berkeley, CA (2023). doi: 10.1007/978-1-4842-9600-4_2.
4. Wang, X.: Data Acquisition in Vehicular Ad Hoc Networks. *Commun. ACM*, vol. 61, no. 5, pp. 83–88, Apr. 2018. doi: 10.1145/3197544.
5. Wojdyla, J. A., et al.: DA+ Data Acquisition and Analysis Software at the Swiss Light Source Macromolecular Crystallography Beamlines. *J Synchrotron Radiat*, vol. 25, no. 1, pp. 293–303, Jan. 2018. doi: 10.1107/S1600577517014503.
6. Montgomery, B., Nakroshis, P.: labjack-controller: Robust and Easy Data Collection with Labjack T-Series DAQs in Python. *J Open Source Softw*, vol. 4, no. 38, p. 1448, Jun. 2019. doi: 10.21105/joss.01448
7. Ajjour, Y., Wachsmuth, H., Benzmüller, H., Stuckenschmidt, C., et al.: Data Acquisition for Argument Search: The args.me Corpus. In: *KI 2019: Advances in Artificial Intelligence*, pp. 48–59. Springer International Publishing, Cham (2019).
8. Upadhyay, D., Sampalli, S.: SCADA (Supervisory Control and Data Acquisition) systems: Vulnerability Assessment and Security Recommendations. *Comput Secur*, vol. 89, p. 101666, 2020. doi: https://doi.org/10.1016/j.cose.2019.101666
9. Bombicz, P., et al.: LabJack-like Crystal Structures of Halogenated 2-Phenylbenzimidazoles – Are They Isostructural? *Acta Crystallographica Section A*, vol. 76, no. a1, p. a90, Aug. 2020. doi: 10.1107/S0108767320099092
10. Guo, H., Liu, J., Zhao, L.: Big Data Acquisition Under Failures in FiWi Enhanced Smart Grid. *IEEE Trans Emerg Top Comput*, vol. 7, no. 3, pp. 420–432, 2019. doi: 10.1109/TETC.2017.2675911.
11. Xia, X., et al.: An Automated Driving Systems Data Acquisition and Analytics Platform. *Transp Res Part C Emerg Technol*, vol. 151, p. 104120, 2023. doi: https://doi.org/10.1016/j.trc.2023.104120.
12. Ilyas, B., Raouf, S. M., Abdelkader, S., Camel, T., Said, S., Lei, H.: An Efficient and Reliable Chaos-Based IoT Security Core for UDP/IP Wireless Communication. *IEEE Access*, vol. 10, pp. 49625–49656, 2022. doi: 10.1109/ACCESS.2022.3173338.
13. Szymańska, E., Petrović, L., Marković, I., Petrović, I.: Mobile Robot Teleoperation via Android Mobile Device with UDP Communication. In: *2021 44th International Convention on Information, Communication and Electronic Technology (MIPRO)*, pp. 1143–1148. (2021). doi: 10.23919/MIPRO52101.2021.9596800.
14. Li, Y., Du, F., Wang, P., Yin, Y.: A Multi-robot Topic Communication Method Based on TCP and UDP. In: *Proceedings - 2020 Chinese Automation Congress, CAC 2020*, pp. 668–672. Institute of Electrical and Electronics Engineers Inc. (Nov. 2020). doi: 10.1109/CAC51589.2020.9327802.

15. Iyengar, J., Thomson, M.: RFC 9000: QUIC: A UDP-Based Multiplexed and Secure Transport, 2021. [Online]. Available: https://www.rfc-editor.org/info/rfc9000.
16. Ioana, A., Burlacu, C., Korodi, A.: Approaching OPC UA Publish–Subscribe in the Context of UDP-Based Multi-Channel Communication and Image Transmission. *Sensors (Switzerland)*, vol. 21, no. 4, pp. 1–19, Feb. 2021. doi: 10.3390/s21041296.
17. Pekez, N., Popović, A., Kovačević, J.: Ethernet TCP/IP-Based Audio Interface for DSP System Verification. *IEEE Consumer Electronics Magazine*, vol. 10, no. 1, pp. 45–50, 2021. doi: 10.1109/MCE.2020.2988616.
18. Giuliano, R., Vizzarri, A., Calderone, A., Mazzenga, F.: Communication Transport Protocol Strategies for Rail Applications. *Applied Sciences (Switzerland)*, vol. 12, no. 6, Mar. 2022. doi: 10.3390/app12063013.
19. Ma, T.: Robot Queue Following Improvement Based on UDP Communication. In: *2021 IEEE 3rd International Conference on Civil Aviation Safety and Information Technology (ICCASIT)*, pp. 792–795. (2021). doi: 10.1109/ICCASIT53235.2021.9633521.
20. Milanovic, A., Srbljic, S., Sruk, V.: Performance of UDP and TCP Communication on Personal Computers. In: *2000 10th Mediterranean Electrotechnical Conference. Information Technology and Electrotechnology for the Mediterranean Countries. Proceedings. MeleCon 2000 (Cat. No.00CH37099)*, pp. 286–289 vol.1. (2000). doi: 10.1109/MELCON.2000.880422.
21. Nathi, R. A., Sutar, D.: Secured and Lightweight Communication Scheme on UDP for Low End IoT Devices. In: *2019 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, pp. 1–6. (2019). doi: 10.1109/ANTS47819.2019.9118096.
22. Ardita, M., Suryani Faradisa, I., Romadhon Parada Dian Palevi, B.: Evaluasi Unjuk Kerja Latency Komunikasi IoT Berbasis UDP untuk Aplikasi pada Intelligent Transportation System (ITS). *Vol. 13*, p. 2022.
23. Li, Y., Yu, J., Chen, L., Hu, Y., Chen, X., Wang, J.: Packet Transport for Maritime Communications: A Streaming Coded UDP Approach. *China Communications*, vol. 19, no. 9, pp. 10–25, 2022. doi: 10.23919/JCC.2022.09.002.
24. Eggert, L., Fairhurst, G., Shepherd, G.: UDP Usage Guidelines, Mar. 2017. RFC Editor. doi: 10.17487/RFC8085.
25. Kumar, D. M., Pooja, B., Sowmith, D., Rakesh, P., Deepthi, D.: Arduino Based Accident Prevention, Detection and Reporting System. *International Journal of Research Publication and Reviews*, vol. 4, no. 12, pp. 2702–2707, Dec. 2023. doi: 10.55248/gengpi.4.1223.123509.
26. Asadi, F.: Reading Different Sensors with Arduino. In: *Essentials of Arduino Boards Programming: Step-by-Step Guide to Master Arduino Boards Hardware and Software*, pp. 309–315. Apress, Berkeley, CA (2023). doi: 10.1007/978-1-4842-9600-4_11.