Jurnal Ilmiah Teknik Elektro Komputer dan Informatika (JITEKI)
Vol. 8, No. 2, June 2022, pp. 307-317
ISSN: 2338-3070, DOI: 10.26555/jiteki.v8i2.23835

UDP Pervasive Protocol Design and Implementation on Multi Devices using MyRIO

Mochammad Hannats Hanafi Ichsan, Rizal Maulana, Octavian Metta Wisnu Wardhana
Computer Engineering, Computer Science Faculty, Brawijaya University

ARTICLE INFO

Article history:

Received April 11, 2022
Revised June 23, 2022
Accepted July 26, 2022

Keywords:

Sensor Network;
Pervasive Computing;
User Datagram Protocol (UDP);
MyRio;
LabView;
State Machine

ABSTRACT

Pervasive Computing is one of the practical computing applications to facilitate computer operations by minimizing human interaction with computers. Pervasive Computing has been developed using UDP protocol to recognize the other devices without manual configuration. NI MyRIO device is one of the most reliable devices for the prototyping process. However, there are still not many implementations of data transmission using specific protocols. And the direction of use for smart homes or smart environments is still not widely done. This research contribution implemented Pervasive UDP protocols on PC devices and two NI MyRIO using LabVIEW programming language. UDP protocols are used because they do not require a handshake to recognize another device to reduce delays and have smaller data sizes due to the absence of recognition fields and sequence fields. Each device uses a dual-state machine system design that has a function to detect other devices automatically and act as an application to use the address of another device. PC represents the host, and MyRIO represents the client. Using the same state machine to detect all devices can recognize more than one device on the same network. The obtained test results show that all functional testing scenarios succeeded 100%. The discovery time is averaged at 0.202754 seconds for First MyRIO as First Client and 0.303201 seconds for Second MyRIO as Second Client. The delay in sending data from the host to the client is no more than 2 seconds. Based on this research, MyRIO has the ability to pervasive Computing with other devices. And can be used for prototyping models with good capabilities.

Corresponding Author:

Mochammad Hannats Hanafi Ichsan, Computer Engineering, Computer Science Faculty, Brawijaya University
Jl. Veteran No. 8, Malang 65145, East Java, Indonesia
Email: hanas.hanafi@ub.ac.id

1. INTRODUCTION

Internet of Things is one of the results of the development of technology that contributes to bringing up some intelligent devices such as smartphones, smart homes, smart TV, smartwatches, etc. [1, 2, 3, 4, 5]. In 2020, it was estimated that there would be more than 26 billion connected devices [6, 7, 8]. The high potential of the Internet of Things is an impetus to implement pervasive Computing on IoT devices [9, 10, 11]. Pervasive Computing is a field of research from ubiquitous Computing that provides a revolutionary paradigm for computing models [12, 13, 14]. There is a concept of invisible and integration [15, 16, 17]. The system requires minimal interaction from human to Computer, Computer to machine, and machine to machine that can be integrated with other devices in the environment [18, 19, 20]. In other words, pervasive Computing is suitable for automating smart devices and allowing them to recognize all devices in the background for data exchange [21, 22].

The User Datagram Protocol (UDP) is one of the TCP / IP transport layer protocols that supports communication in small data segments [23, 24]. The UDP is usually called the lightweight protocol [25, 26, 27] because the data is small, so sending information is relatively fast [28, 29]. Implementing pervasive

Computing requires UDP because this protocol is not using sequence and acknowledgment fields [30, 31]. Thus, the data size becomes smaller. UDP protocol also does not need a handshaking process, and it saves memory and processor resources [32]. MyRIO device is one of the most reliable devices for the prototyping process [33, 34]. MyRIO is used because it has a strong device resistance, embedded sensors, and a suitable communication device [35, 36].

Previous research [37] has built a lightweight UDP pervasive protocol prototype using client-host architecture. UDP protocol is suitable to be applied because this protocol is lighter and can be used to send a broadcast to all devices [38, 39, 40]. But it only creates a design implemented on two PC as a host and client. Another research implemented a design from the previous study to a MyRIO device [41]. But that research is only focused on a state machine that has been already designed before. Its design is only used for one client. It cannot be connected if the host is broadcasting to several clients.

This research contribution focuses on designing and analyzing performance in UDP pervasive multi-device protocol implementation as host and MyRIO as a client. MyRIO represents the smart device that attaches to the environment. The client can automatically recognize each other on the same network and support more than two devices. It will redesign the state machine so it can communicate with multi-devices. This research will also be focused on functional and non-functional design. The function of each device represents the available design, and the non-functional design is defined by how the system can be assessed for its reliability. In addition, some security features will also be added to the system in the form of an authentication process. With this feature, even if the devices can be connected pervasively, some access control can still be differentiated between many users or things.

2. METHOD

This section will describe some steps to be taken in doing this research. The first step is to define the system requirement. The next step will be to design the system topology and the sequence diagram that illustrates data communication between devices. After that, a state diagram for each device will be determined to realize it. State machine structure in LabVIEW programming language will be used to implement that state diagram. After downloading the program to each device, an experiment will fulfill the system requirement. Measurement of time delay between states will also be done to know the performance achieved by the system.

2.1. System Requirement

The pervasive characteristic can be seen in the system's ability to detect device names, services, and data types automatically. To achieve that, some functional requirements must be made. The applicable requirement is shown in Table 1. In this experiment, three devices will be used. The host device is represented by a PC (Personal Computer). The other two are the client represented by MyRIO.

Table 1. System Functional Requirement

Variable	Requirement	Description
x	Device X can automatically detect device Y	Device X can recognize the IP address of device Y, which is in the same Wireless Local Area Network
y	Device X can send its status to device Y	Device X can send its data and or online status to device Y when two devices are already connected in the same Wireless Local Area Network
z	Device Y can receive status from device X	Device Y can know all the data and or status from device X. It also can change the setting or configuration of device X

2.2. System Design

The PC is tasked to receive data sent by MyRIO. It is also functioned to control and configure the settings of the MyRIO. These three devices are connected by Wifi from a local wireless access point, as shown in Fig. 1. When firstly activated, MyRIO will be sent a broadcast that contains its device name and IP address, so the other device which received it will know the identity of MyRIO. On the other hand, the PC will automatically enter the "listening" state when activated. When the PC receives the broadcasted message, it will do a duplication check and then store the identity of the devices. After that, the PC will send a reply message which contains its own identity, device name, and IP address. MyRIO will also do a duplication check and then store the information. So now, those devices are already known about each other.

After that sequence, MyRIO can send its device status to the known PC. The device status can contain various variables or data. It can be in sensor reading, pressed buttons, etc. PC can also be sent a request or control the status of MyRIO. For example, it can lock or unlock the device or ask for the username and password to configure the setting. In this case, MyRIO will do an authentication process and send the result to the PC. The sequence diagram of these processes can be seen in Fig. 1. The design in Fig. 1 is based on previous

research. In the design of prior research, the host cannot recognize multi-device at one moment. This research was redesigned so the host could be connected with a multi-device client.

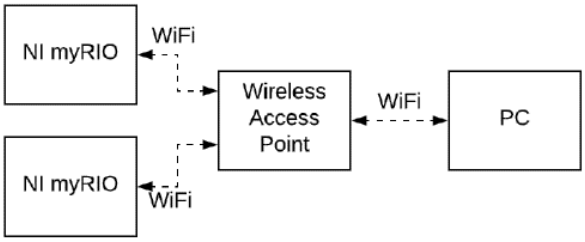

Fig. 1. System Block Diagram

2.3. State Machine Design

The algorithm for the pervasive will be presented in the form of a state machine diagram that represents in Fig. 2. This form is chosen because this system uses an event-triggered mechanism. In the PC / host, there is the two-state diagram. One is for auto-detection, and the other one is for data communication. These two states work together at the same time. When the PC is activated for auto-detection, it will do a "broadcast" state and switch to a "listening" state. When it receives a broadcast (BC) or broadcast reply (Reply BC) from other devices when in the "listening" state, it will go to a "check duplication" state. It will change to the "Reply BC" state if it receives a broadcast. But if it gets a broadcast reply, it will go to the "listening" state again.

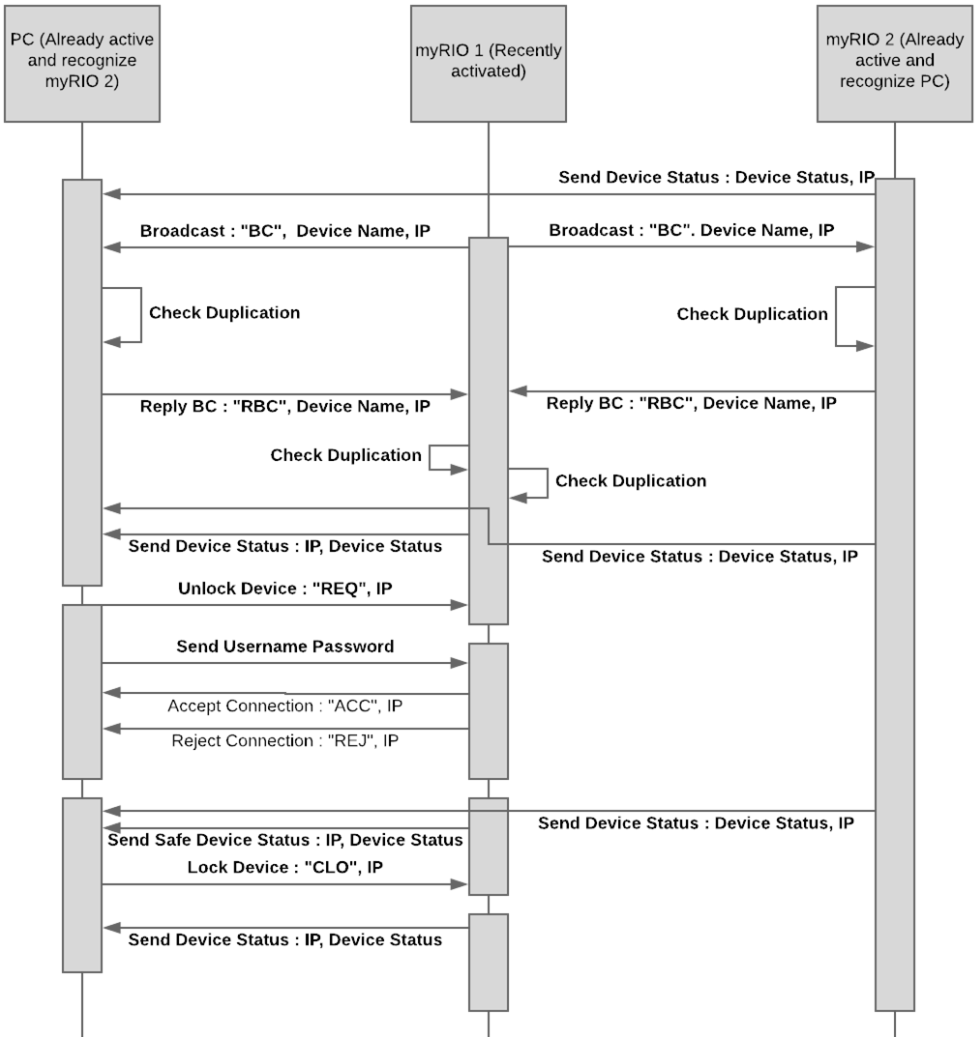

Fig. 2. Sequence Diagram of Data Communication

Data communication is done between two devices that already know each other. First, it will do an "initialization" to open the communication port. Then it goes to the "listening" state. While in the "listening" state, the PC will receive data about device status from the other device. Then the user can use connect button to control that device after the successful authentication in the "authentication" state. If the username and password are wrong, it will switch back to the "listening" state. This process is shown in Fig. 3.

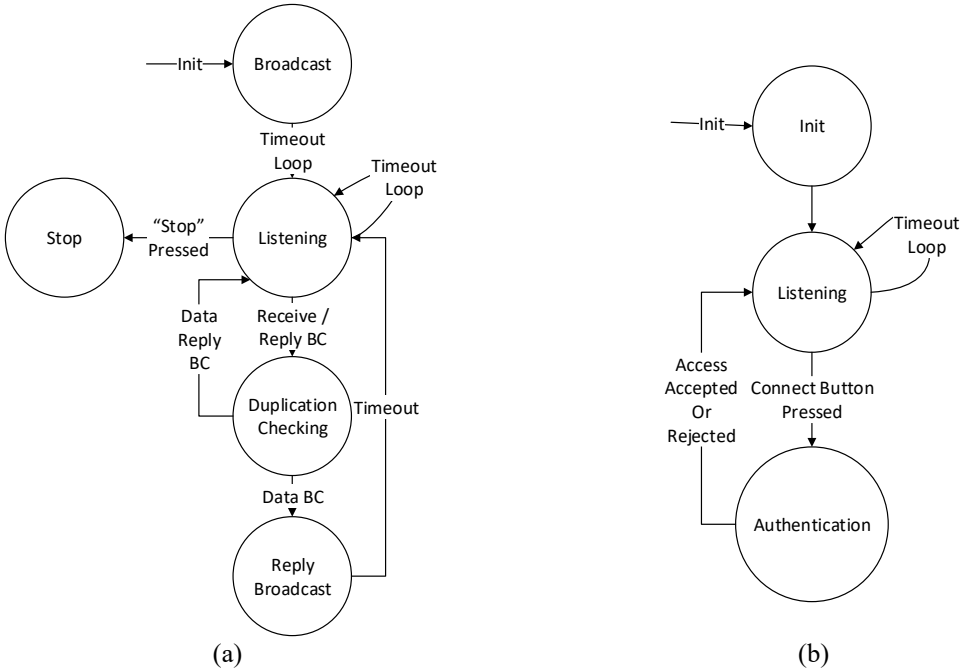

Fig. 3. (a) Auto Detection State Machine Diagram in P.C./host. (b) Data Communication State Machine Diagram in P.C./host

In the MyRIO, two different state machines run together simultaneously, as shown in Fig. 4. The first one functions as auto-detection, and the other one is for data communication. The state diagram for auto-detection is the same as in PC. The difference is in the data communication. First, it does an "initialization" and then switches to an "idle" state. In this state, it sends data about its status periodically.

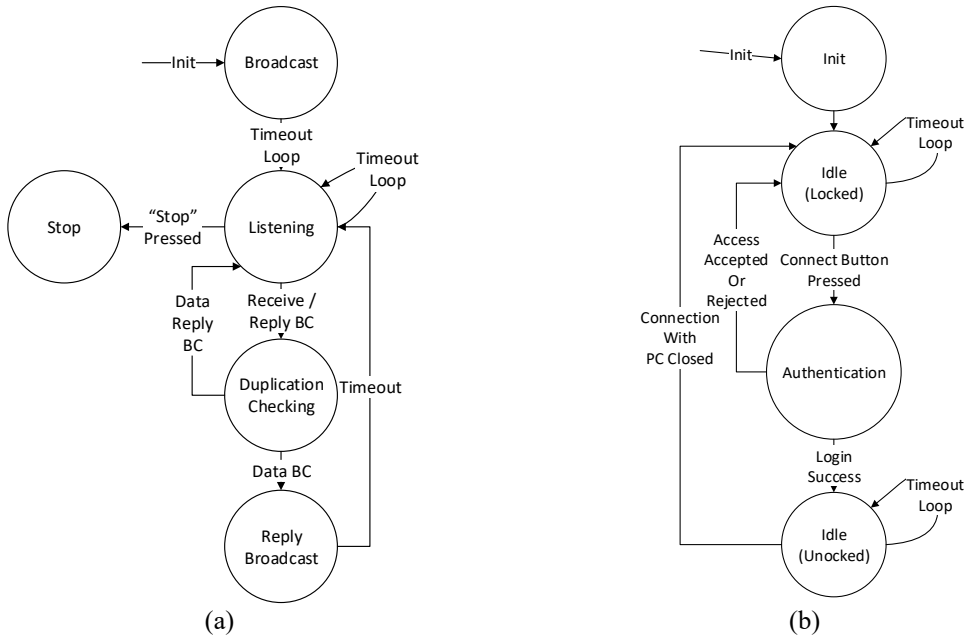

Fig. 4. (a) Auto Detection State Machine Diagram in MyRIO/client. (b). Data Communication State Machine Diagram in MyRIO

This status can change depending on the condition and treatment it experienced, such as its sensor reading or pressing a button. To reset the device status or change the configuration setting, MyRIO needs to be Unlocked first. MyRIO will enter an "authentication" state when the PC requests this process. In this state, it waits for the data about the username and password from the PC. If the username and password are correct, it will enter the "idle (Unlocked)" state. Otherwise, it will switch back to the "idle (Locked)" state. This process is shown in Fig. 4.

3. RESULT AND DISCUSSION

The state transition experiment is done by observing system behaviors when given a specific event that triggered the transition between two states. The observed state diagrams are the auto-detection state diagram and the data communication state diagram. The results are displayed in Table 2, Table 3, and Table 4.

Table 2. State Transition Experiment Results for Auto-detection State Machine Diagram

No.	Initial State	Test Case	Expected Destination State	Result	Status
1.	Broadcast	Data broadcast to 255.255.255.255	Listening	After the Broadcast has been sent, the program enters the listening state	Valid
2.	Listening	Receiving a broadcast reply from other devices	Check Duplication	After receiving the broadcast reply, the program enters the check duplication state	Valid
3.	Listening	Receiving broadcasts from other devices	Check Duplication	After receiving the Broadcast, the program enters the check duplication state	Valid
4.	Listening	The stop button is pressed by the user	Stop	Program enter stop state	Valid
5.	Check Duplication	Data in the previous Broadcast contain device identity that is not in the existing list	Save device identity to the array and then reply to the Broadcast	The essence of the new device is saved to the array, and then the program will enter the reply BC state	Valid
6.	Check Duplication	Data in the previous Broadcast contain device identity that is already in the existing list	Reply the Broadcast	Program enter reply BC state	Valid
7.	Check Duplication	Data in the previous broadcast reply contain device identity that is not in the existing list	Save device identity to the array and then go to the listening state	The essence of the new device is saved to the array, and then the program will enter a listening state	Valid
8.	Check Duplication	Data in the previous broadcast reply contain device identity that is already in the existing list	Go to the listening state	Program enters the listening state	Valid
9.	Reply BC	Send its device identity to the broadcast sender	Go to the listening state	After sending the reply, the program will enter the listening state	Valid
10.	Stop	Program stop while loop	Stop	Program terminated because the condition for stopping the while loop will become true	Valid

The first test tests the transition between state machines that have been designed. PC testing and myRIO can detect other devices automatically for verification and discovery time during the client detection process. This test is carried out to check whether the state switching acts as previously designed. Based on Table 2, it can be concluded that the tests using state transition checking were stated to have been successfully tested following the design based on the status obtained by each test.

Each row in Table 3 of the Recognized Devices and Discovery Time column represents the results on the first myRIO device and the second MyRIO device from the same Broadcast Sent time. Based on Table 3, it was found that the two devices were recognized within 0.3 seconds and 0.1 seconds longer than the discovery time when there was only one device. And the second myRIO was recognized about 0.2 seconds after the first MyRIO was recognized. Based on the results obtained, it can be concluded that when both send ACKs simultaneously, the PC can only process one ACK data at a time. The PC cannot store both simultaneously and can only process them sequentially.

Table 3. State Transition Experiment Results for Data Communication State Machine Diagram in MyRIO

No.	Initial State	Test Case	Expected Destination State	Result	Status
1.	Init	Opening port to send and receive data	Idle (Locked), Status Safe	After opening the port, the program enters the idle (Locked) state, and the device status is Safe	Valid
2.	Idle (Locked), Status Safe	Detecting some motion from the sensor	Idle (Locked), status motion detected	After detecting motion, the device status became "motion detected," and the state is still "idle (Locked)." The sensor data is sent to the PC every 5 seconds periodically.	Valid
3.	Idle (Locked), Status Safe	The button in MyRIO is pressed	Idle (Locked), status button pressed	After pressing the button, the device status became "button pressed," and the state is still "idle (Locked)." The data is sent to the PC every 5 seconds periodically.	Valid
4.	Idle (Locked), Status Safe	PC sent a request to authenticate	Authentication	Program enters authentication state	Valid
5.	Idle (Locked), Status Motion Detected	Detecting some motion from the sensor	Idle (Locked), status motion detected	After detecting motion, the device status is still "motion detected," and the state is still "idle (Locked)." The sensor data is sent to the PC every 5 seconds periodically.	Valid
6.	Idle (Locked), Status Motion Detected	The button in MyRIO is pressed	Idle (Locked), status button pressed	After pressing the button, the device status became "button pressed," and the state is still "idle (Locked)." The data is sent to the PC every 5 seconds periodically.	Valid
7.	Idle (Locked), Status Motion Detected	PC sent a request to authenticate	Authentication	Program enters authentication state	Valid
8.	Idle (Locked), Status Button Pressed	Detecting some motion from the sensor	Idle (Locked), status motion detected	After detecting motion, the device status became "motion detected," and the state is still "idle (Locked)." The sensor data is sent to the PC every 5 seconds periodically.	Valid
9.	Idle (Locked), Status Button Pressed	The button in MyRIO is pressed	Idle (Locked), status button pressed	After pressing the button, the device status is still "button pressed," and the state is still "idle (Locked)." The data is sent to the PC every 5 seconds periodically.	Valid
10.	Idle (Locked), Status Button Pressed	PC sent a request to authenticate	Authentication	Program enters authentication state	Valid
11.	Authentication	PC sent data about username and password correctly before 10 seconds passed away	Idle (Unlocked), status safe	MyRIO sends acceptance acknowledgment to the PC Program, enters an "idle (Unlocked)" state, and the status becomes "safe."	Valid
12.	Authentication	PC sent data about username and password incorrectly before 10 seconds passed away	Idle (Locked)	MyRIO sends rejection acknowledgment to PC Program, enters an "idle (Locked)" state, and the status is still like before	Valid
13.	Authentication	Data about username and password is not received until 10 seconds have been passed away	Idle (Locked)	MyRIO sends rejection acknowledgment to PC Program, enters an "idle (Locked)" state, and the status is still like before	Valid
14.	Idle (Unlocked), status safe	Detecting some motion from the sensor	Idle (Unlocked), status safe	After detecting motion, the device status is still "safe," and the state is still "idle (Unlocked)." The sensor data is sent to the PC every 5 seconds periodically.	Valid
15.	Idle (Unlocked), status safe	The button in MyRIO is pressed	Idle (Unlocked), status safe	After pressing the button, the device status is still "safe," and the state is still "idle (Unlocked)." The sensor data is sent to the PC every 5 seconds periodically.	Valid
16.	Idle (Unlocked), status safe	PC sent a request to lock MyRIO	Idle (closed), status safe	Program enters an "idle (Locked)" state, and status is still "safe."	Valid

Table 4. State Transition Experiment Results for Data Communication State Machine Diagram in PC.

No	Initial State	Test Case	Expected Destination State	Result	Status
1.	Init	Opening port to send and receive data	Listening	After opening the port, the program enters an idle (Locked) state, and the device status is Safe	Valid
2.	Listening, connect button not pressed	Receiving MyRIO status (Safe, Motion Detected, Button Pressed) every 10 seconds periodically	Listening, receive data status periodically	After receiving data, status, data value, and time when receiving data are displayed in an array	Valid
3.	Listening, connect button not pressed	Not receiving MyRIO status (Safe, Motion Detected, Button Pressed) until 10 seconds have been passed away	Listening, displaying "Device Lost."	The state is still "listening," and "device lost" is displayed	Valid
4.	Listening, connect button not pressed	Connect button is pressed. A request to authenticate is being sent.	Authentication	Program enters "authentication" state and sends data containing the address of the device that wants to be connecter	Valid
5.	Authentication	Data about username and password is sent correctly before 10 seconds have been passed away	Listening, displaying the device identity that is connected	PC receives acceptance acknowledgment and goes back to the listening state while also indicating device identity that is connected	Valid
6.	Authentication	Data about username and password is sent incorrectly before 10 seconds have been passed away	Listening, displaying wrong authentication message	PC receives rejection acknowledgment and goes back to the listening state while also showing the wrong authentication message	Valid
7.	Authentication	Data about username and password is not being sent until 10 seconds have been passed away	Listening, displaying failed authentication message	PC receives rejection acknowledgment and goes back to the listening state while also communicating failed authentication message	Valid
8.	Listening, connect button is pressed	Connect button deactivated	Listening, displaying Locked device	Program enters "idle (Locked)" state	Valid

Testing MyRIO device can act as an intelligent device object aims to determine the validity of the implementation of state machine data exchange on a PC with MyRIO and determine the delay that occurs when sending data.

By looking at the state transition experiment results, it can be concluded that the program fulfills the design requirement to respond correctly based on an event-triggered mechanism. The following experiment measures time delay in the data communication state. Time measurement is done at the time to 1. Discover one MyRIO device by PC. 2. Discover two MyRIO Devices by PC. 3. Delay switching from Locked to the Unlocked state. 4. Delay switching from Unlocked to the Locked state. The results are displayed in Table 5, Table 6, Table 7, and Table 8.

Table 5. Discovery Time for One MyRIO

No	Time in Hour Minutes Seconds		Delay (seconds)
	Broadcast Sent (host)	Broadcast Received (client)	
1.	03:33:32.244	03:33:32.448	0.20434
2.	03:33:50.539	03:33:50.741	0.201762
3.	03:33:14.432	03:33:14.634	0.201895
4.	03:33:43.418	03:33:43.620	0.201886
5.	03:33:59.555	03:33:59.758	0.203888
Averages Delay			**0.202754**

The test has been done five times. The request delivery delay test was carried out, and the results are shown in Table 6. Based on Table 6, the average request delivery delay was 0.303 seconds, and it can be concluded that the request delivery delay was relatively fast.

Table 6. Discovery Time for Two MyRIOs

No.	Time in Hour Minutes Seconds			Delay Host-client1 (seconds)	Delay Host-client2 (seconds)
	Broadcast Sent (host)	Broadcast Received (client1)	Broadcast Received (client2)		
1.	03:49:28.302	03:49:28.505	03:49:28.706	0.203223	0.404165
2.	03:50:04.304	03:50:04.507	03:50:04.708	0.202888	0.403768
3.	03:50:25.899	03:50:26.101	03:50:26.301	0.201886	0.401786
4.	03:50:45.630	03:50:45.834	03:50:46.035	0.203907	0.405414
5.	03:52:21.931	03:52:22.133	03:52:22.333	0.202542	0.402426
Averages Delay				**0.303200**	

The test has been done five times for the delay in changing MyRIO status to Unlocked, and the results are shown in Table 7. Based on Table 7, the average delay from changing status to Unlocked is 0.0014 seconds. With this, it can be concluded that changing MyRIO status to Unlocked is relatively fast.

Table 7. Time Delay to Switch from Locked to Unlocked State

No.	Time in Hour Minutes Seconds		Delay (seconds)
	Broadcast Sent (host)	Broadcast Received (client)	
1.	03:16:51.324	03:16:51.325	0.001
2.	03:17:46.548	03:17:46.550	0.002
3.	03:18:10.422	03:18:10.423	0.001
4.	03:18:39.491	03:18:39.492	0.001
5.	03:19:15.575	03:19:15.577	0.002
Averages Delay			**0.0014**

The test has been done five times for the delay of the MyRIO status change from Locked to Unlocked, and the results are shown in Table 8. Based on Table 8, the average delay from changing status to Locked is 0.9384 seconds. With this, it can be concluded that the process of changing MyRIO status to Locked is relatively slower than the previous process.

From the time measurement experiments, it can be concluded that PC cannot simultaneously process acknowledgment from two MyRIO. It can be seen from Table 5 and Table 6 that the discovery for the second MyRIO is made after the acknowledgment process from the first MyRIO has been done. So, adding more devices to be discovered can make discovery time longer to be finished for all of the devices. On the other hand, the time delay to switch from a Locked to an Unlocked state seems shorter than the time delay between Unlocked and closed states. But overall, the wait is approximately only one second; thus, the system already fulfills all the requirements in the design processes.

Table 8. Time Delay to Switch from Unlocked to Locked State

No.	Time in Hour Minutes Seconds		Delay (seconds)
	Broadcast Sent	Broadcast Received	
1.	03:21:52.054	03:21:52.986	0.932
2.	03:22:54.639	03:22:55.647	1.008
3.	03:23:35.522	03:23:36.524	1.002
4.	03:24:19.381	03:24:20.399	1.018
5.	03:25:33.255	03:25:33.987	0.732
Averages Delay			**0.9384**

4. CONCLUSIONS AND FUTURE WORKS

UDP pervasive multi-device protocol is working well with its dual-state machine system design. The design is based on host-client configuration, PC represents the host, and MyRIO represents both clients. The first state machine is used to detect another device's addresses automatically. The second state machine uses another device address obtained from the first state machine. With the proposed system design in this research, all devices can be detected, and the system function works properly. The average discovery time of this system

design is 0,202754 seconds when there is one MyRIO on the network and 0,303201 seconds when there are two MyRIO on the network. The delay of sending data from PC to MyRIO is no more than 2 seconds.

This research is enriched by adding several features from previous works. The first important feature is the system can connect many devices pervasively. The other features are detecting motion, authentication, and safe / not safe status. We were adding that feature to aim at the security of things at home, such as a security box or household things that need security. This feature is essential for intelligent home applications. The system could be integrated with data encryption so the data that the host or client sends could be safer.

For future works, the system design of two state machines in one loop system can be the basis for developing LabVIEW-based UDP Pervasive Multi-Device protocol applications in the future. The first state machine remains to recognize the device using UDP Pervasive. In contrast, the design can be used on the second state machine for other applications from the Pervasive UDP implementation.

Acknowledgments

Thanks to the Robotics and Embedded System Laboratory, Faculty of Computer Science, Brawijaya University provides the facilities for this research. This research is one of several outcomes of "Hibah Peneliti Pemula" HPP 2021 From LPPM University of Brawijaya. The previous research that has been published is on EECSI 2016 with the title "Lightweight UDP Pervasive Protocol in Smart Home Environment Based on LabVIEW," IJECE 2018 with the title "UDP Pervasive Protocol Integration with IoT for Smart Home Environment using LabVIEW."

REFERENCES

[1] S. Sepasgozar, R. Karimi, L. Farahzadi, F. Moezzi, S. Shirowzhan, S. M. Ebrahimzadeh, F. Hui, and L. Aye, "A Systematic Content Review of Artificial Intelligence and the Internet of Things Applications in Smart Home," *Applied Sciences,* vol. 10, no. 9, p. 3074, 2020, https://doi.org/10.3390/app10093074.

[2] S. ur Rehman and V. Gruhn, "An approach to secure smart homes in cyber-physical systems/Internet-of-Things," in *2018 Fifth International Conference on Software Defined Systems (SDS)*, 2018, https://doi.org/10.1109/SDS.2018.8370433.

[3] H. Touqeer, S. Zaman, R. Amin, M. Hussain, F. Al-Turjman, and M. Bilal, "Smart home security: challenges, issues and solutions at different IoT layers," *The Journal of Supercomputing,* vol. 77, p. 14053–14089, 2021, https://doi.org/10.1007/s11227-021-03825-1.

[4] P. K. D. Pramanik; S. Pal; P. Choudhury, "Beyond Automation: The Cognitive IoT. Artificial Intelligence Brings Sense to the Internet of Things," in *Lecture Notes on Data Engineering and Communications Technologies*, Springer, Cham, 2017, p. 1–37, https://doi.org/10.1007/978-3-319-70688-7_1.

[5] S. Munirathinam, "Chapter Six - Industry 4.0: Industrial Internet of Things (IIOT)," *Advances in Computers,* vol. 117, no. 1, pp. 129-164, 2020, https://doi.org/10.1016/bs.adcom.2019.10.010.

[6] T. Alsboui, Y. Qin, R. Hill, and H. Al-Aqrabi, "Enabling distributed intelligence for the Internet of Things with IOTA and mobile agents," *Computing,* vol. 102, p. 1345–1363, 2020, https://doi.org/10.1007/s00607-020-00806-9.

[7] H. Espinoza, G. Kling, F. McGroarty, M. O'Mahony, and X. Ziouvelou, "Estimating the impact of the Internet of Things on productivity in Europe," *Heliyon,* vol. 6, no. 5, 2020, https://doi.org/10.1016/j.heliyon.2020.e03935.

[8] P. Singh, "Internet of Things Based Health Monitoring System: Opportunities and Challenges," *International Journal of Advanced Research in Computer Science,* vol. 9, no. 1, pp. 224-228, 2018, https://doi.org/10.26483/ijacrs.v9i1.5308.

[9] M. H. H. Ichsan, W. Kurniawan, G. E. Setyawan and I. A. K. Sandy, "WSN performance based on node placement by genetic algorithm at smart home environment," *TELKOMNIKA,* vol. 17, no. 1, pp. 299-306, 2019, https://doi.org/10.12928/telkomnika.v17i1.11621.

[10] O. Ali, M. K. Ishak, M. K. L. Bhatti, I. Khan, and Ki-Il Kim, "A Comprehensive Review of Internet of Things: Technology Stack, Middlewares, and Fog/Edge Computing Interface," *Sensors,* vol. 22, no. 3, p. 995, 2022, https://doi.org/10.3390/s22030995.

[11] J. H. Nord, A. Koohang, and J.Paliszkiewicz, "The Internet of Things: Review and theoretical framework," *Expert Systems with Applications,* vol. 133, pp. 97-108, 2019, https://doi.org/10.1016/j.eswa.2019.05.014.

[12] S. A. Alshqaqi, A. T. Zahary, and M. Zayed, "Ubiquitous Computing Environment: literature review," in *2019 First International Conference of Intelligent Computing and Engineering (ICOICE)*, 2019, https://doi.org/10.1109/ICOICE48418.2019.9035157.

[13] Z. Yu and A. Dey, "Inaugural editorial of CCF transactions on pervasive computing and interaction," *CCF Transactions on Pervasive Computing and Interaction,* vol. 1, no. 2, 2019, https://doi.org/10.1007/s42486-019-00009-y.

[14] M. Rath, "A Methodical Analysis of Application of Emerging Ubiquitous Computing Technology With Fog Computing and IoT in Diversified Fields and Challenges of Cloud Computing," *International Journal of Information Communication Technologies and Human Development (IJICTHD),* vol. 10, no. 2, p. 13, 2018, https://doi.org/10.4018/IJICTHD.2018040102.

[15] O. F. AbdelWahab, A. I. Hussein, F. A. H. Hamed, M. H. Kelash, A. A. M. Khalaf, and H. M. Ali, "Hiding data in images using steganography techniques with compression algorithms," *TELKOMNIKA,* vol. 17, no. 3, pp. 1168-1175, 2019, https://doi.org/10.12928/telkomnika.v17i3.12230.

[16] A. Burrows, D. Coyle, and R. Gooberman-Hill, "Privacy, boundaries and smart homes for health: An ethnographic study," *Health & Place,* vol. 50, pp. 112-118, 2018, https://doi.org/10.1016/j.healthplace.2018.01.006.

[17] S. Freigang, L. Schlenker, and T. Köhler, "A conceptual framework for designing smart learning environments," *Smart Learning Environments,* vol. 2018, p. 5, 2018, https://doi.org/10.1186/s40561-018-0076-8.

[18] M. H. H. Ichsan, W. Kurniawan and S. R. Akbar, "UDP Pervasive Protocol Integration with IoT for Smart Home Environment using LabVIEW," *International Journal of Electrical and Computer Engineering,* vol. 8, no. 6, p. 5342, 2018, https://doi.org/10.11591/ijece.v8i6.pp5342-5350.

[19] A. Mayub, S. Fahmizal, M. Shidiq, U. Y. Oktiawati and N. R. Rosyid, "Implementation smart home using internet of things," *TELKOMNIKA,* vol. 17, no. 6, pp. 3126-3135, 2019, https://doi.org/10.12928/telkomnika.v17i6.11722.

[20] M. M. Akhtar, D. R. Rizvi, M. A. Ahad, S. S. Kanhere, M. Amjad, and G. Coviello, "Efficient Data Communication Using Distributed Ledger Technology and IOTA-Enabled Internet of Things for a Future Machine-to-Machine Economy," *Sensors,* vol. 21, no. 13, p. 4354, 2021, https://doi.org/10.3390/s21134354.

[21] A. Y. Shdefat, N. Mostafa, L. Saker, and A. Topcu, "A Survey Study of the Current Challenges and Opportunities of Deploying the ECG Biometric Authentication Method in IoT and 5G Environments," *Indonesian Journal of Electrical Engineering and Informatics (IJEEI),* vol. 9, no. 2, pp. 119-141, 2021, https://doi.org/10.52549/ijeei.v9i2.2890.

[22] Y. Shanmugarasa, H. Y. Paik, S. S. Kanhere, and L. Zhu, "Towards Automated Data Sharing in Personal Data Stores," in *2021 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops)*, 2021, https://doi.org/10.1109/PerComWorkshops51409.2021.9431001.

[23] Y. Wang, T. Gamage and C. H. Hauser, "Security Implications of Transport Layer Protocols in Power Grid Synchrophasor Data Communication," *IEEE Transactions on Smart Grid,* vol. 7, no. 2, pp. 807-816, 2016, https://doi.org/10.1109/TSG.2015.2499766.

[24] V. S. Chakravarthi, "M2M Communication Protocols," in *Internet of Things and M2M Communication Technologies*, Springer, Cham, 2021, p. 167–190, https://doi.org/10.1007/978-3-030-79272-5_11.

[25] G. Fairhurst and T. Jones, Transport Features of the User Datagram Protocol (UDP) and Lightweight UDP (UDP-Lite), University of Aberdeen, 2018, https://doi.org/10.17487/RFC8304.

[26] S. Majumder, S. Ray, D. Sadhukhan, M. K. Khan, and M. Dasgupta, "ECC-CoAP: Elliptic Curve Cryptography Based Constraint Application Protocol for Internet of Things," *Wireless Personal Communications,* vol. 116, p. 1867–1896, 2021, https://doi.org/10.1007/s11277-020-07769-2.

[27] J. O. Agyemang, J. J. Kponyo, J.D. Gadze, H. Nunoo-Mensah, and D. Yu, "A Lightweight Messaging Protocol for Internet of Things Devices," *Technologies,* vol. 10, no. 21, pp. 1-20, 2022, https://doi.org/10.3390/technologies10010021.

[28] P. Kumar and S. Verma, "Implementation of modified OLSR protocol in AANETs for UDP and TCP environment," *Journal of King Saud University - Computer and Information Sciences,* vol. 1, no. 1, 2019, https://doi.org/10.1016/j.jksuci.2019.07.009.

[29] A. Churcher, R. Ullah, J. Ahmad, S. u. Rehman, F. Masood, M. Gogate, F. Alqahtani, B. Nour and W. J. Buchanan, "An Experimental Analysis of Attack Classification Using Machine Learning in IoT Networks," *Sensors, MDPI,* vol. 21, no. 2, p. 446, 2021, https://doi.org/10.3390/s21020446.

[30] L. Tong, K. Zheng, K. Xu, R. A. Jadhav, T. Xiong, K. Winstein, and K. Tan, "Revisiting Acknowledgment Mechanism for Transport Control: Modeling, Analysis, and Implementation," *IEEE/ACM Transactions on Networking,* pp. 1-15, 2021, https://doi.org/10.1109/TNET.2021.3101011.

[31] A. Mondal, S. Bhattacharjee, and S. Chakraborty, "Viscous: An End to End Protocol for Ubiquitous Communication Over Internet of Everything," in *2017 IEEE 42nd Conference on Local Computer Networks (LCN)*, 2017, https://doi.org/10.1109/LCN.2017.79.

[32] Y. Yilmaz, L. Aniello, and B. Halak, "ASSURE: A Hardware-Based Security Protocol for Internet of Things Devices," in *Authentication of Embedded Devices*, Springer, Cham, 2021, p. 55–87, https://doi.org/10.1007/978-3-030-60769-2_3.

[33] F. Adamo, G. Andria, A. D. Nisio, C. G. C. Carducci, A. Lay-Ekuakille, G. Mattencini, and M. Spadavecchia, "Designing and Prototyping A Sensor Head For Test and Certification of UAV Components," *International Journal*

on Smart Sensing and Intelligent Systems, vol. 10, no. 3, pp. 646-672, 2017, https://doi.org/10.21307/ijssis-2017-228.

[34] N. Kurniawati, A. Abimanyu, and Muhtadan, "Design of Sorting Machine Prototype in Electronic Circuit Based on NI-MyRIO 1900," *Journal of Physics: Conference Series,* pp. 1-10, 2020, https://doi.org/10.1088/1742-6596/1436/1/012064.

[35] P. A. M. Devan, G. Manisha, K. G. T. Rajarajeswari, M. Priyanga, and K. Sangeetha, "Fire Safety and Alerting System in Railways," in *IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)*, 2018, https://doi.org/10.1109/RTEICT42901.2018.9012364.

[36] D. Renaux, R. Linhares, F. Pottker, A. Lazzaretti, C. Lima, A. C. Neto, and M. Campaner, "Designing a Novel Dataset for Non-intrusive Load Monitoring," in *2018 VIII Brazilian Symposium on Computing Systems Engineering (SBESC)*, 2018, https://doi.org/10.1109/SBESC.2018.00045.

[37] W. Kurniawan, M. H. H. Ichsan, S. R. Akbar, and I. Arwani, "Lightweight UDP Pervasive Protocol in Smart Home Environment Based on Labview," in *IAES International Conference on Electrical Engineering, Computer Science and Informatics*, 2017, https://doi.org/10.1088/1757-899X/190/1/012009.

[38] C. C. Sobin, "A Survey on Architecture, Protocols and Challenges in IoT," *Wireless Personal Communications,* vol. 112, p. 1383–1429, 2020, https://doi.org/10.1007/s11277-020-07108-5.

[39] M. A. A. da Cruz, J. J. P. C. Rodrigues, P. Lorenz, P. Solic, J. Al-Muhtadi and V. H. C. Albuquerque, "A proposal for bridging application layer protocols to HTTP on IoT solutions," *Future Generation Computer Systems,* vol. 97, pp. 145-152, 2019, https://doi.org/10.1016/j.future.2019.02.009.

[40] D. S. Kolluru and P. B. Reddy, "Review on communication technologies in telecommunications from conventional telephones to smart phones," in *AIP Conference Proceedings 2407*, 2021, https://doi.org/10.1063/5.0074088.

[41] W. Kurniawan, M. H. H. Ichsan and S. R. Akbar, "UDP Pervasive Protocol Implementation for Smart Home Environment on MyRIO using LabVIEW," *International Journal of Electrical and Computer Engineering (IJECE),* vol. 8, no. 1, pp. 113-123, February 2018, https://doi.org/10.11591/ijece.v8i1.pp113-123.