

NGHIÊN CỨU THIẾT KẾ GIAO DIỆN TRUYỀN THÔNG MODBUS GIỮA MÁY TÍNH PC VỚI HỆ BIẾN TẦN - ĐỘNG CƠ KHÔNG ĐỒNG BỘ BA PHA

A STUDY OF DESIGN MODBUS COMMUNICATIONS INTERFACE BETWEEN PC COMPUTER WITH THREE PHASE MOTOR-INVERTER

ĐÀO MINH QUÂN, ĐÀO QUANG KHANH*

Khoa Điện - Điện tử, Trường Đại học Hàng hải Việt Nam

*Email liên hệ: khangdq@vamaru.edu.vn

Tóm tắt

Trong bài báo này đề xuất một phương thức truyền thông Modbus giữa máy tính PC với hệ biến tần động cơ không đồng bộ 3 pha qua giao diện được thiết kế bằng ngôn ngữ Visual Basic. Việc xây dựng giao diện truyền thông này, nhằm giúp giảng viên, sinh viên tiếp cận công nghệ truyền thông công nghiệp với hệ truyền động điện hiện đại, dễ dàng cập nhật thông tin và cài đặt cho hệ biến tần qua máy tính PC. Kết quả nghiên cứu đã xây dựng một giao diện trên máy tính trực quan, cập nhật được thông số cài đặt, hiển thị được đầy đủ thông tin hoạt động của một hệ biến tần động cơ không đồng bộ 3 pha. Ngoài ra, kết quả cũng cho thấy khả năng áp dụng ghép nối truyền thông công nghiệp cho điều khiển và giám sát từ xa với các thiết bị điện hiện đại điện trong ngành điện tự động tàu thủy, điện tự động công nghiệp.

Từ khóa: Biến tần, ghép nối truyền thông công nghiệp, ngôn ngữ Visual Basic.

Abstract

This paper, we propose a Modbus communication method between PC and inverter system-asynchronous 3-phase motor through an interface designed in VisualBasic language. The construction of this communication interface is aimed at helping lecturers and students access industrial communication technology with modern electric drive systems, easily updating information and installing the inverter system via a PC. The research results have built an intuitive computer interface, updated the settings, showing the full operation information of a 3 - phase asynchronous motor inverter system. In addition, the results also show the ability to apply industrial communications for remote control and monitoring with modern electrical equipment in shipbuilding and industrial automation.

Keywords: Inverter motor drive, communication interface, visualbasic language.

1. Đặt vấn đề

Hiện nay, các nghiên cứu về ghép nối truyền thông công nghiệp được phát triển rất mạnh mẽ với các thiết bị điện - điện tử công suất, truyền động điện hiện đại [2],... Trong các thiết kế hệ thống người ta thường sử dụng màn hình HMI, PLC kết nối với các thiết bị [8], [9], việc phát triển các ứng dụng như trên chủ yếu được áp dụng cho những điều khiển trực tiếp tại hệ thống, thực hiện các thao tác đơn giản, không giám sát từ khoảng cách xa. Một cách ghép nối truyền thông giám sát từ xa là sử dụng máy tính công nghiệp kết nối với hệ thống thông qua phần mềm SCADA bằng giao thức Modbus [5], [6], [7], tuy nhiên các nghiên cứu này chủ yếu sử dụng phần mềm tích hợp thiết bị sẵn như WinCC hoặc LabVIEW. Việc tự phát triển các phần mềm kết nối và điều khiển bằng máy tính PC với thiết bị điện - điện tử công nghiệp bằng giao thức Modbus vẫn còn là vấn đề cần phải quan tâm nhiều hơn. Ngoài ra, việc tự phát triển phần mềm còn giúp kỹ sư, sinh viên, học viên nghiên cứu sâu hơn về truyền thông công nghiệp, khai thác có hiệu quả hơn các thiết bị điện tử công nghiệp hiện đại. Vì vậy, nhóm tác giả đề xuất tự xây dựng một giao diện phần mềm có thể ghép nối truyền thông, thu thập dữ liệu và điều khiển một thiết bị điện tử công nghiệp phổ biến là biến tần - động cơ không đồng bộ 3 pha. Giao diện phần mềm được xây dựng bằng ngôn ngữ Visual Basic 6.0, là ngôn ngữ thông dụng được dùng để thiết kế giao diện trên máy tính, dễ dàng phát triển được các ứng dụng, đóng gói và cài đặt vào máy tính PC [1].

2. Nội dung

2.1. Ghép nối truyền thông máy tính PC với biến tần Invt GD20

2.1.1. Giao thức truyền thông Modbus với biến tần Invt GD20

Truyền thông Modbus là một giao thức truyền thông phổ biến trong công nghiệp, các thiết bị có thể giao tiếp với nhau thông qua giao thức truyền thông

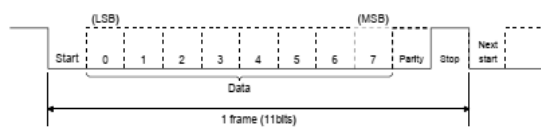
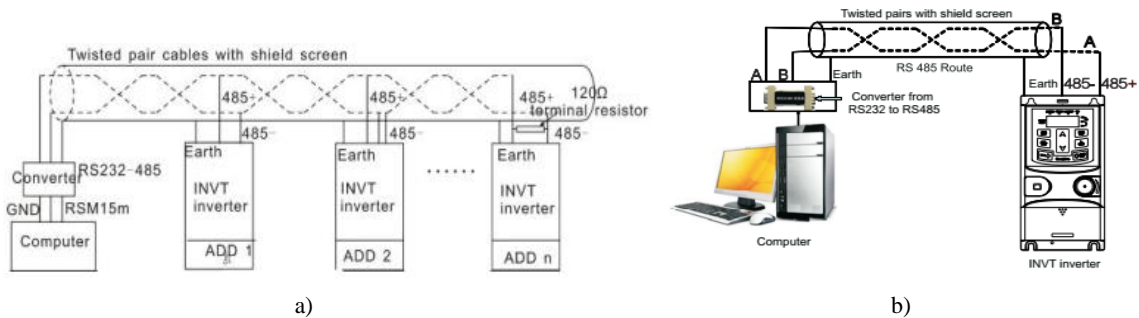
này qua cổng truyền RS485 hoặc RS422 và mô hình dạng Master-Slave. Trong truyền thông Modbus có ba chế độ truyền thông là: ASCII, RTU và TCP [3], tuy nhiên với biến tần Invt GD20 chỉ được cấu hình hai chế độ truyền thông là ASCII và RTU. Một thiết bị Master điều khiển các Slave, thiết bị Master sẽ gửi tín hiệu lệnh, yêu cầu tới các thiết bị Slave khác và thực thi lệnh đó Hình 1a. Trong bài báo này, tác giả sử dụng thiết bị Master là máy tính PC, thiết bị Slave là biến tần Invt GD20, có sơ đồ ghép nối như Hình 1b [4].

2.1.2. Cấu hình truyền thông Modbus với biến tần Invt GD20

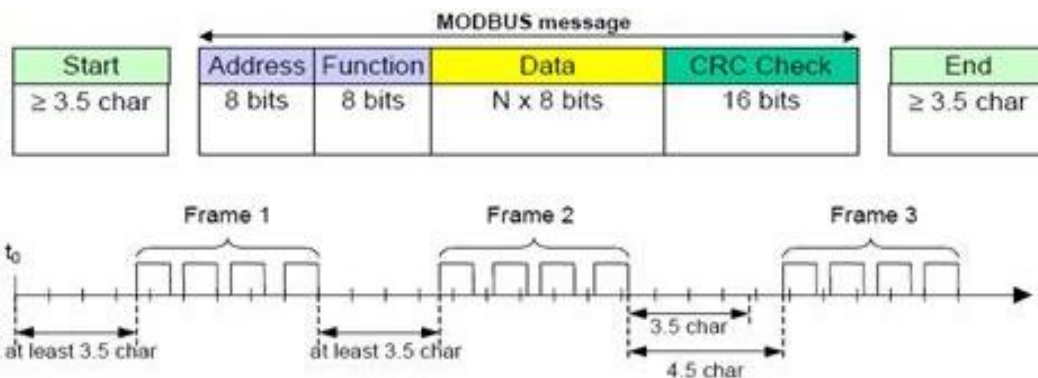
Truyền thông Modbus trong biến tần Invt GD20 là chế độ truyền thông RTU. Mỗi byte (8 bit) của khung truyền được mã hóa thành 2 ký tự ASCII (mỗi ký tự được mã hóa trên một nibble), do đó thông tin truyền thông trong chế độ RTU cho một lần truyền mang được nhiều thông tin hơn so với chế độ ASCII. Khung truyền thông dữ liệu trong chế độ RTU được thể hiện trong Bảng 1 [4].

Bảng 1. Khung truyền thông Modbus trong chế độ RTU

Tốc độ Baud	9600/19200/38400/57600
Start bit	1 bit
Data bit	8 bit
Parity bit	0 bit
Stop bit	1 bit

Hình 1. (a) Sơ đồ ghép nối Modbus giữa máy tính PC với nhiều biến tần qua chuẩn RS485, (b) Sơ đồ ghép nối Modbus giữa máy tính PC với biến tần Invt GD20 qua chuẩn RS485



Hình 2. Gói dữ liệu truyền thông Modbus trong chế độ RTU

Với biến tần Invt GD20, chế độ truyền thông Modbus được cài đặt bằng cài đặt thông số truyền thông cho biến tần ở địa chỉ P14 là địa chỉ dùng để giao tiếp Modbus.

P14.00: Dùng để định địa chỉ giao tiếp, có thể chọn từ 1~247;

P14.01: Dùng để chọn tốc độ Baud;

P14.02: Cài đặt khung truyền.

Cấu trúc gói dữ liệu truyền thông Modbus của biến tần Invt GD20 trong chế độ RTU được thể hiện trong Hình 2 [4].

- Byte địa chỉ ADDR: xác định thiết bị mang địa chỉ được nhận dữ liệu (đối với Slave) hoặc dữ liệu nhận được từ địa chỉ nào (đối với Master). Địa chỉ này được quy định từ 0-254;

- Byte mã lệnh CMD: được quy định từ Master, xác định yêu cầu dữ liệu từ thiết bị Slave, trong đó 03h: đọc dữ liệu tức thời dạng Byte từ Slave, 06h: ghi dữ liệu tức thời dạng Byte vào Slave.

- Byte dữ liệu DATA: xác định dữ liệu trao đổi giữa Master và Slave;

- Byte CRC: 2 byte kiểm tra lỗi của hàm truyền, giá trị của Byte CRC là 16 Bit.

Quá trình Master nhận dữ liệu được thể hiện trong Bảng 2, trong bảng thể hiện việc nhận 2 byte dữ liệu liên tiếp là 0002H từ địa chỉ 0004H từ biến tần Invt GD20 có địa chỉ giao tiếp 01H [4].

Bảng 2. Cấu trúc gói dữ liệu truyền thông cho việc nhận dữ liệu

START	T1-T2-T3-T4
ADDR	01H
CMD	03H
High bit of the start address	00H
Low bit of the start address	04H
High bit of data number	00H
Low bit of data number	02H
CRC low bit	85H
CRC high bit	CAH
END	T1-T2-T3-T4

Quá trình Master truyền dữ liệu được thể hiện trong Bảng 3, trong bảng thể hiện việc truyền 2 byte dữ liệu liên tiếp là 1388H tới địa chỉ 0004H của biến tần có địa chỉ Slave giao tiếp là 02H.

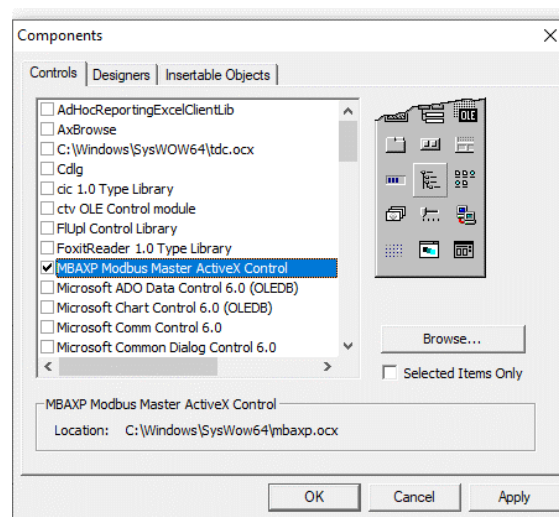
Bảng 3. Cấu trúc gói dữ liệu truyền thông cho việc truyền dữ liệu

START	T1-T2-T3-T4
ADDR	02H
CMD	06H
High bit of writing data address	00H
Low bit of writing data address	04H
High bit of data content	13H
Low bit of data content	88H
CRC CHK low bit	C5H
CRC CHK high bit	6EH
END	T1-T2-T3-T4

Quá trình Master truyền dữ liệu được thể hiện trong Bảng 3, trong bảng thể hiện việc truyền 2 byte dữ liệu liên tiếp là 1388H tới địa chỉ 0004H của biến tần có địa chỉ Slave giao tiếp là 02H.

2.2. Thiết kế giao diện và lập trình ghép nối truyền thông trên máy tính PC

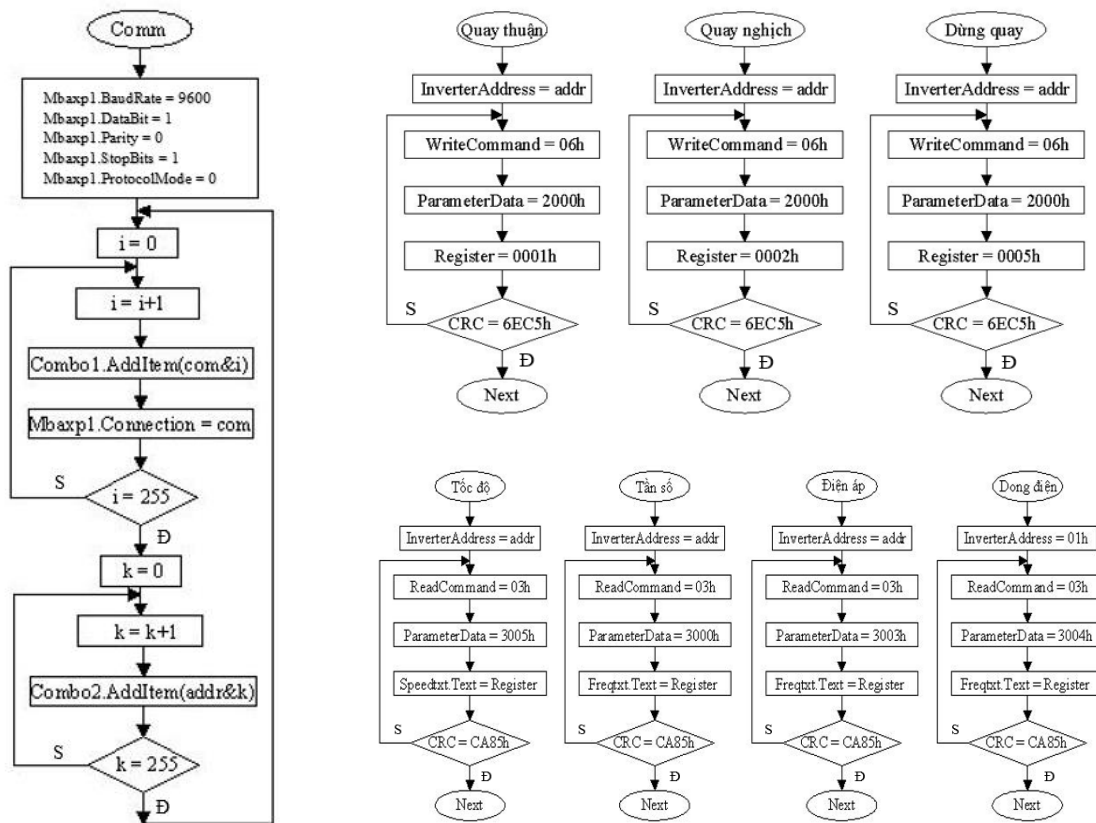
Việc thiết kế một giao diện trên máy tính PC sử dụng hệ điều hành Windows có thể dùng nhiều ngôn ngữ lập trình khác nhau như Visual C++, Visual C#, Visual Basic, Delphi,... [3]. Trong bài báo này, nhóm tác giả sử dụng ngôn ngữ Visual Basic 6.0 (VB6.0), là ngôn ngữ lập trình hướng đối tượng đơn giản, trực quan trên môi trường Windows, VB6.0 cung cấp một bộ công cụ hoàn chỉnh để đơn giản hóa việc triển khai lập trình ứng dụng, ghép nối máy tính và cho phép sử dụng các thư viện liên kết động có phần mở rộng *.dll, *.ocx. Có thể nói đây là cách nhanh và tốt nhất để phát triển các ứng dụng trên Microsoft Windows.



Hình 3. Điều khiển Modbus ActiveX Control trong Visual Basic 6.0

Để lập trình giao thức truyền thông Modbus trong môi trường Visual Basic ta phải sử dụng một Modbus ActiveX Control (Hình 3). Khi đó, trên thanh công cụ của VB 6.0 ta có một điều khiển Modbus với các thuộc tính của nó, cho phép ta dễ dàng sử dụng cho việc lập trình ghép nối truyền thông Modbus bằng VB 6.0 [1].

Từ những thông tin cần hiển thị của biến tần Invt GD20 như: điện áp, dòng điện, tốc độ, tần số,... và thông tin cần điều khiển biến tần như: quay thuận, quay nghịch, điều chỉnh tần số (tốc độ quay, nhóm tác giả xây dựng một giao diện điều khiển như Hình 5, lưu đồ thuật toán của chương trình được thể hiện trong Hình 4.



Hình 4. Thuật toán lập trình giao diện truyền thông Modbus cho máy tính PC ghép nối với biến tần

2.3. Kết quả và thảo luận

Trên cơ sở nghiên cứu và xây dựng thuật toán trong các Mục 2.1, Mục 2.2, sử dụng môi trường thiết kế giao diện Visual Basic, nhóm tác giả đã thiết kế một giao diện truyền thông Modbus RTU trên máy tính PC cho hệ biến tần - động cơ không đồng bộ 3 pha như trong Hình 5. Sau khi kết nối máy tính PC với biến tần bằng chuẩn truyền thông RS485 nhóm tác giả đã hoàn thành hệ thống thí nghiệm truyền thông Modbus máy tính PC - biến tần như trong Hình 6.

Hệ thống hoạt động trong chế độ không tải được thực hiện các như sau:

- Thiết lập cổng Comm cho Master là máy tính PC và địa chỉ của Slave là biến tần trên các Combo của giao diện, xác nhận bằng nút Connect trên giao diện như trong Hình 5;
- Thiết lập chế độ vẽ đồ thị, đặt tần số tại khung SV;
- Cho hệ thống hoạt động bằng cách nhấn vào nút quay thuận FWD hoặc nút quay nghịch REV. Khi đó, hệ biến tần - động cơ sẽ hoạt động, động cơ quay thuận hoặc nghịch với các tốc độ khác nhau

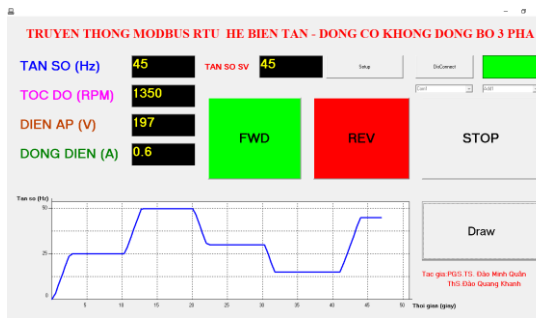
theo các tần số đặt trên khung SV được thể hiện trong Bảng 4;

- Muốn dừng động cơ ta nhấn vào nút Stop hoặc dừng kết nối nhấn vào nút Disconnect trên giao diện của máy tính PC.

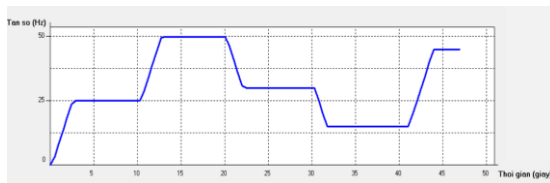
Bảng 4. Tần số đặt cho hệ biến tần - động cơ và các thông số đo và hiển thị được trên phần mềm

Stt	Tần số đặt (Hz)	Dòng điện (A)	Điện áp (V)	Tốc độ (RPM)
1	25	0,6	110	750
2	50	0,6	219	1500
3	30	0,6	131	900
4	15	0,6	66	450
5	45	0,6	175	1350

Nhận xét: Từ hệ thống thí nghiệm như trong Hình 6 và những giá trị đặt tần số như trong Bảng 3, các thông tin như điện áp, dòng điện, tốc độ quay được hiển thị, cập nhật thời gian thực trên giao diện. Giao diện đã hiển thị chính xác, nhiều thông tin cập nhật thời gian thực hơn so với màn hình LED của biến tần chỉ hiển thị được một thông tin trong cùng thời điểm.



Hình 5. Giao diện truyền thông Modbus trên máy tính PC để ghép nối với biến tần

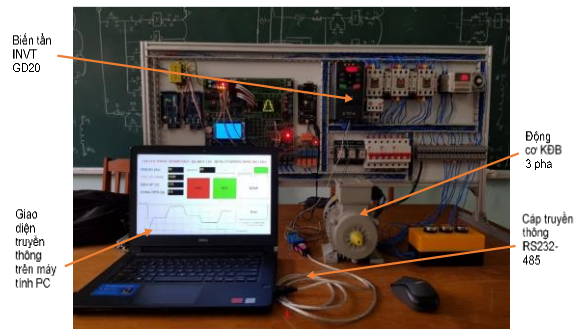


Hình 7. Hiện thị thông tin của hệ biến tần - động cơ trên máy tính

Ngoài ra, giao diện cũng tự động cập nhật các dữ liệu tần số đặt và tự động đọc chính xác tần số từ hệ thống và vẽ đồ thị đặc tính thay đổi tần số như thể hiện Hình 7. Kết quả cho thấy, từ các tần số đặt khác nhau từ máy tính, hệ thống hoạt động ổn định, các đại lượng đọc được chính xác. Với việc thiết kế hệ thống đơn giản, giao diện được bố trí rõ ràng, trực quan, hiển thị đầy đủ các thông tin hoạt động của biến tần, thuận tiện cho việc giảng dạy trong phòng thực hành.

5. Kết luận

Truyền thông Modbus từ một thiết bị Master với một Slave là hệ Biến tần - động cơ được sử dụng phổ biến trong công nghiệp mang tính ứng dụng cao trong thực tế, thiết bị Master có thể thực hiện bằng nhiều phương án khác nhau như: màn hình HMI, giao diện WinCC-PC, Vi điều khiển, PLC,... Trong bài báo này, hệ Biến tần-động cơ được ghép nối thành công với máy tính PC thông qua truyền thông Modbus bằng ngôn ngữ Visual Basic đã bổ sung một phương án giao tiếp mới bằng ngôn ngữ lập trình hướng đối tượng. Kết quả hoạt động của hệ thống cho thấy hệ thống làm việc ổn định, chính xác, việc thao tác thí nghiệm dễ dàng, phù hợp với giảng viên, sinh viên trong giảng dạy và học tập về truyền thông công nghiệp và điều khiển máy điện. Tuy nhiên, hạn chế của bài báo là chỉ dừng ở việc thí nghiệm khi hệ thống hoạt động ở chế độ không tải, giao diện truyền thông được phát triển trong môi trường Visual Basic 6.0 còn đơn giản, là môi trường



Hình 6. Hệ thống thí nghiệm kết nối truyền thông Modbus giữa máy tính PC và biến tần - động cơ

thiết kế cổ điển. Trong tương lai, nhóm tác giả sẽ tiếp tục nghiên cứu thiết kế giao diện truyền thông phong phú hơn, bằng phiên bản Visual Basic mới hơn, thí nghiệm với hệ thống hoạt động ở nhiều chế độ tải khác nhau.

Bài báo là kết quả đề tài nghiên cứu Khoa học cấp Trường năm học 2019-2020: “Nghiên cứu ghép nối truyền thông vi điều khiển với biến tần - động cơ”.

TÀI LIỆU THAM KHẢO

- [1] Nguyễn Thị Kiều Duyên, Lập trình Visual Basic đơn giản và hiệu quả, NXB Trẻ, 2000.
- [2] Bùi Quốc Khánh, Nguyễn Văn Liễu, Cơ sở truyền động điện, NXB KHK, 2009.
- [3] Đinh Anh Tuấn, Đào Minh Quân, Mạng truyền thông công nghiệp tàu thủy, NXB Hàng hải, 2015.
- [4] Shenzhen INVT Electric Co.,Ltd, Operation Manual Goodrive20 Series Inverter, 2015.
- [5] Savas Sahin Modbus - Based SCADA/HMI Applications, Journal of Information Technology and Application in Education, 2013
- [6] Sachintha Kariyawasam, Real-Time Simulation of a Microgrid Control System using modbus Communication, RTDS Technologies Inc., Winnipeg, Canada, 2018.
- [7] Traian Turc, Gas Plant SCADA Software Application, University of Targu Mures, 2015.
- [8] <https://songnguyen.vn/huong-dan-truyen-thong-modbus-rtu-bien-tan-yaskawa-voi-plc-delta-dvp.html>
- [9] <https://dienminhquang.com/truyen-thong-modbus-fx3u-mitsubishi-voi-bien-tan-m200-nidec-huong-dan/>.

Ngày nhận bài:	19/02/2020
Ngày nhận bản sửa:	27/02/2020
Ngày duyệt đăng:	10/03/2020