Indian Journal of Science and Technology, *Vol 11(32), DOI: 10.17485/ijst/2018/v11i32/131113,* August 2018
ISSN (Print) : 0974-6846
ISSN (Online) : 0974-5645

Control and Monitoring of Electrical Variables of a Level Process using Modbus RTU-TCP/IP Industrial Communication

Wilson Sánchez Ocaña[1]*, Chancúsig Alex[2], Gamboa Ricardo[2], Tipán Diego[2] and Elizabeth Salazar[1]

[1]Departamento de Eléctrica y Electrónica, Universidad de las Fuerzas Armadas ESPE, ID: 60104598, Sangolquí, Ecuador; wesanchez@espe.edu.ec, dvtipng@espe.edu
[2]Departamento de Energía y Mecánica, Universidad de las Fuerzas Armadas ESPE, ID: 60104598, Sangolquí, Ecuador; netzonews@hotmail.com, abchancsigq@espe.edu, rjgamboa@espe.edu

Abstract

Background/Objectives: To develop an open protocol industrial communication network that allows the supervision, control and data acquisition of a frequency converter and an energy meter, which are connected to a pumping and level control system; this process will be carried out from a local, remote control - Human Machine Interface (HMI) and from the web, using Modbus RTU - TCP/IP. **Methods:** The implementation and configuration of equipment and devices of the Communication Network was carried out according to the three main layers that make up an Industrial OSI system, such as: the two-tier environment and connectivity in TCP/IP as well as in the Scada System RTU, controllers, meters, converters and field elements; the Configuration and Linking stage, both in TCP and serial and finally for the Application stage for both the Scada system and for the HMI. **Findings:** The configuration of the devices in TCP/IP used a 24 mask, while in RTU the identifier used was Id 15; the PC_System for the Scada System, the Delta display for the HMI, the PLC S7-1200 for programming the control system, as well as the Sentron Pac 3200 energy meter, monitor and control variables such as line and motor frequency, voltages and currents between lines, power and energy consumption of the industrial system and at the same time it will be possible to generate a record (Web, Server), of data in an Excel format, from any electronic device with internet access, entering through the IP address of the PLC previously configured with a user name and password that allows total access to the readings registered by the PLC. **Applications/Improvements:** In supervision, control and data acquisition systems of industrial processes of any type of variable, with controllers and field devices, both with TCP and RTU connectivity, of multiple brands of recognition or not in the market, improving the control and monitoring in real time, for the analysis, diagnosis and decision making, both local and remote, by maintenance, operation and management personnel.

Keywords: Frequency Inverter, Human Machine Interface (HMI), Modbus TCP/RTU, Sentron PAC 3200, Web Server

1. Introduction

In the area of communications within industrial environments, protocol standardization has been applied and optimized for different levels of automation. Each protocol has a range of applications outside of which its performance decreases and the cost/benefit ratio increases. The Modbus/TCP network is a widely used and accepted de-factor standard, but this is not the only one as there are other protocols for Ethernet at the industrial level such as Ethernet/IP (essentially ControlNet and DeviceNet objects over TCP/IP and UDP), ProfiNet (combining the Profibus protocol, OLE for OPC and TCP/IP process control) and Fieldbus Foundation high-speed Ethernet HSE (places the Foundation Fieldbus H1 protocol over TCP/IP and adds OPC and XML1[1] language.

*Author for correspondence

The Ethernet network is used in conjunction with TCP/IP (protocols used on the Internet), providing a reliable mechanism for transporting data between machines and allowing interoperability between different platforms. Using TCP/IP over Ethernet at the field level in the industry allows for true integration with the corporate network and thus tight control over production[2].

The following article presents the results obtained in the implementation of an industrial control network using Modbus/TCP, to establish the monitoring of the Sentron_PAC, on the other hand, the control and monitoring of the frequency inverter will be carried out through a Modbus/RTU communication, which can be monitored through WinCC[3].

"Modbus is a serial communication protocol developed and published by Modicon in 1979, the most widely used in industrial environments, telecontrol and monitoring systems[4].

Transmission with the Modbus protocol is simple, since different electronic devices are connected to a single bus; this communication bus has a master (Master) and several devices that work as slaves (Slaves). For operation, the master asks questions and the slaves respond; it should be noted that only one of the slaves can do so[5].

It has the MODBUS TC/IP protocol, based on ETHERNET technology as a physical medium; it also has MODBUS RTU and MODBUS ASCII, both based on serial communication, but under different modes of data transmission. This project implements the MODBUS RTU protocol. The MODBUS protocol provides two modes of serial transmission, the ASCII mode and the RTU, used for different applications[6].

The topology used in the development of the MODBUS NETWORK is the one recommended by ROCKWELL AUTOMATION called Daisy Chain shown in Figure 1. This configuration is the simplest since the end of one device is the beginning of the next[7].

It allows the interaction between man and machine in a very graphic and intuitive way. This device is programmable and has an infinite number of libraries, in which we can find buttons, keys, clipart images, display, keyboards, bar graphs, animations, etc., also making it possible to import images from a file where by addressing each value, it is possible to establish both monitoring and control of equipment linked to it[8].

SCADA stands for Supervision and Data Acquisition, and is used to create software on computers to facilitate the control and supervision of equipment in the field. In

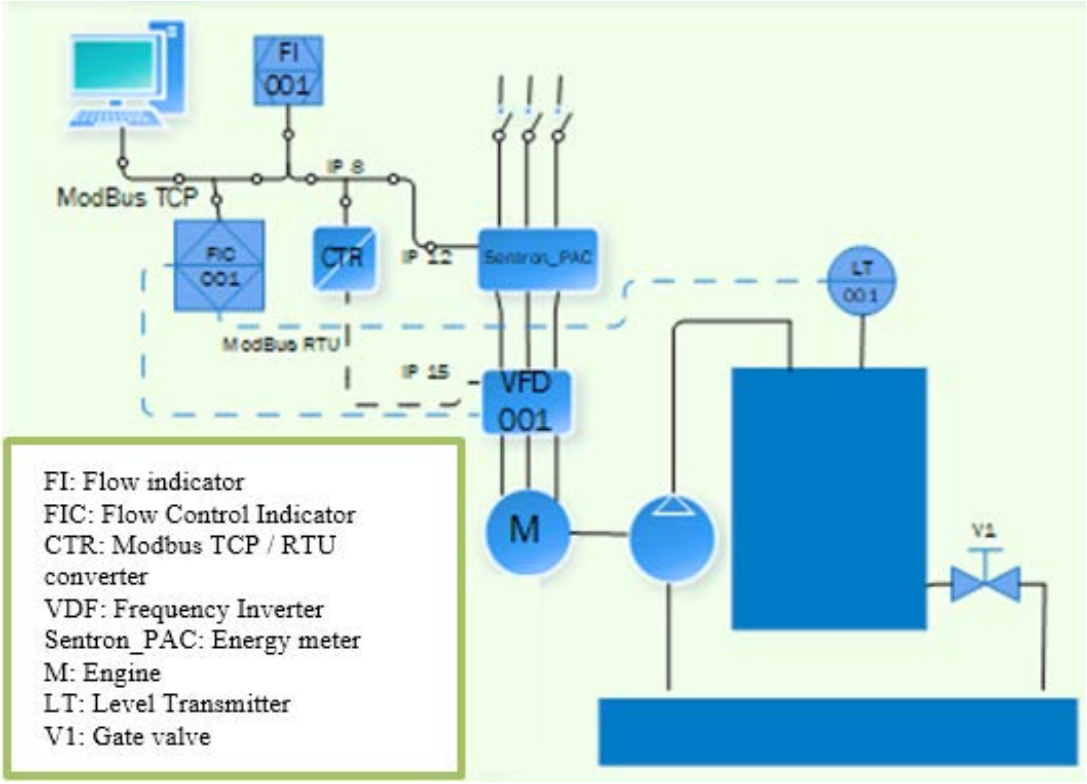

Figure 1. Network topology.

recent years SCADA systems have been a fundamental part of the industry, for the reasons mentioned above. In a SCADA system the master is the computer, while the slaves are all the devices in the field, usually used in programs to be executed in RTU or PLC[9]. This equipment is dedicated to AC motors and is a three-phase powered equipment[10].

2. Design and Methods

The following materials will be used to implement the industrial network:

- Frequency Inverter VFD007E23A.
- Didactic Level Control Station.
- Sentron PAC 3200.
- TIA Portal Software.

2.1 Network Topology

The industrial network is made up of acquisition, control and monitoring equipment linked through a single network that uses Modbus TCP & RTU open communication protocols connected through a converter located in the IP8, a Siemens S7 1200 PLC, as shown in Figure 1, must be properly configured to have reciprocity in communication in order to monitor the output data of the level control station on a PC through an HMI, which must allow to control frequency parameters in the delta frequency converter, generating changes in the Sentron PAC 3200[11] values.

2.2 Instance Data Block

The instance DB stores both a default value and an initial value for each parameter. The value start provides the value that will be used when the FB is executed. The initial value can be changed during the execution of your user program. For the implementation of the example of the industrial network with Modbus TCP/RTU communication protocols, the use of these DB instance blocks is necessary, (Figure 2).

Figure 2. Instance Data Blocks.

2.3 Modbus-TCP (Mb_client)

It is intended to establish a Modbus-TCP communication between an S7-1200 CPU and a Delta frequency inverter, for which purpose the "MB_CLIENT" instruction must be defined and parameterized. A PLC can behave like a Modbus RTU master (Modbus TCP client), i.e. it has the ability to read and write I/O data to a Modbus RTU slave (Modbus TCP server).

The "MB_CLIENT" instruction communicates as a MODBUS TCP client via a PROFINET link on the S7-1200 CPU. No additional hardware modules are required to use the instruction, a link is created between the client and the server, tasks are sent, responses are received, and the link disconnection from the Modbus TCP server is controlled.

To properly configure the MB_CLIENT block as shown in Figure 3, the necessary parameters are listed in (Table 1).

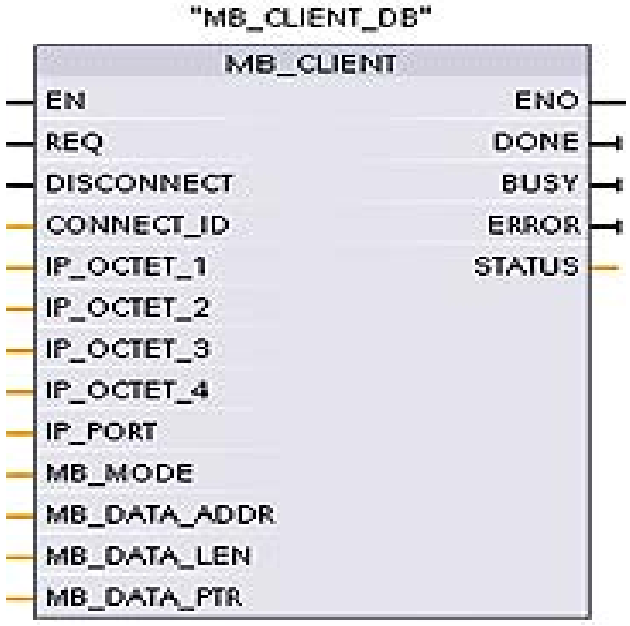

Figure 3. MB_CLIENT instruction.

Table 1. Types of data implemented, within the parameters of the MB_CLIENT

Parameter and type		Data types	Description
REQ	IN	Bool	False: No request for communication True: Communication request
DISCONNECT	IN	Bool	Controls connection and disconnection with a Modbus server device
CONNECT_ID	IN	UInt	Univocally identify each connection inside the PLC.
IP_OCTET_1	IN	USInt	Modbus TCP server IP address: bytes 1 to 4
IP_OCTET_2	IN	USInt	
IP_OCTET_3	IN	USInt	
IP_OCTET_4	IN	USInt	
IP_PORT	IN	UInt	IP port number of the server to which the client will attempt to connect (502)
MB_MODE	IN	USInt	Type of request (read, write or diagnose)
MB_DATA_ADDR	IN	UDInt	Initial address Modbus
MB_DATA_LEN	IN	UInt	Modbus data length
MB_DATA_PTR	IN_OUT	Variant	Log temporarily stores data going to/from a Modbus server
DONE	OUT	Bool	Finished the last request without errors. (True)
BUSY	OUT	Bool	Operation MB_CLIENT in progress
ERROR	OUT	Bool	Error during execution of MB_CLIENT
STATUS	OUT	Word	Execution condition code

The configuration used to establish communication with the SENTRON PAC multimeter is shown in Figure 4, within REQ a PLC's own cycle mark is used to initiate and disconnect the communication in 2.5 Hz intervals, in DISCONNNECT with a value of 0 the PLC tries to establish a connection with the IP address, the bytes corresponding to the IP address of the Modbus server that needs to be established are introduced, the connection address 192.168.0.12 is set, the default communication port corresponds to 502.

The device is configured as bit reading with MB_MODE with 0, MB_DATA_ADDR the Modbus register to be accessed is placed, the number of bits or words to be accessed with MB_DATA_LEN is defined, that is, the 2-bit analysis is performed, finally, the Modbus server data is temporarily recorded using MB_DATA_PTR.

2.4 Mb_Client (Modbus RTU)

To carry out the communication through Modbus RTU protocols and link up within the previously implemented TCP Network, there is the option to configure the network through the MB_COMM_LOAD_DB instruction that handles DB data blocks and allows the configuration of the network and MB_MASTER_DB where the slave addressing required to read or write is carried out, allowing Modbus RTU communication using RS-485 network architectures but an external module is required to perform this communication therefore, in this case, the Modbus RTU network is implemented through a client server that handles the TCP network with the MB_client instruction, where the basic configuration is similar to that configured in the TCP network, therefore the block configuration can be seen in Figure 5.

As an additional configuration different from the TCP network, in the Octet for the IP address of the Modbus RTU server that is required to establish the connection, the address 192.168.0 is set.8 which is the address of the Router where the devices are connected as a frequency inverter and two DELTA screens, in MB_DATA_ADDR is placed the address 48193 that corresponds to the address of the equipment that needs to be connected and it is addressed to the DB70 where the control and reading parameters of the frequency inverter are located with a data length of 2 configured through MB_DATA_LEN.

The MOVE instruction block shown in Figure 6, allows storing a data and moving it to another required address, for this case it is necessary when it is required to modify a value through an instruction commanded from an HMI.

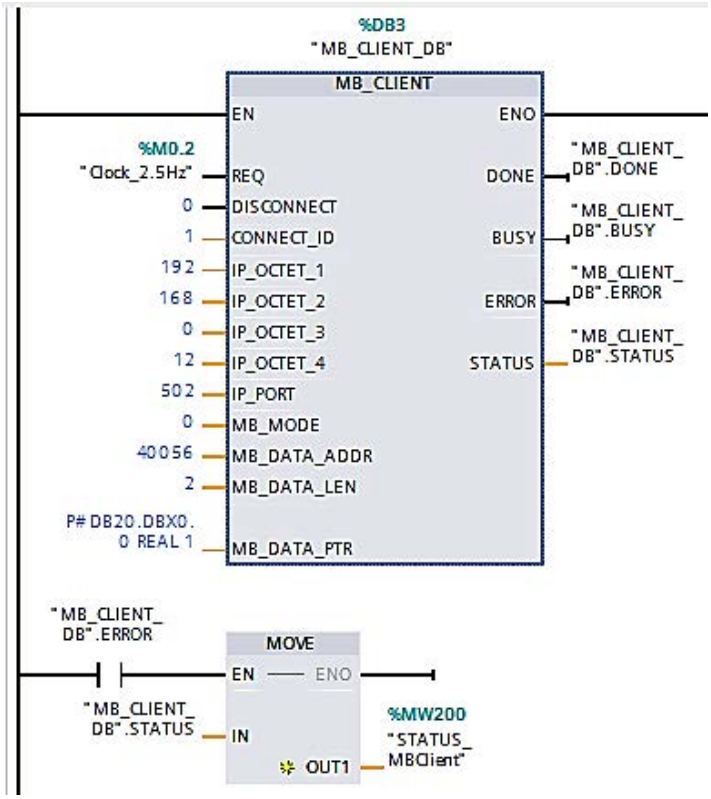

Figure 4. MB_CLIENT instruction for reading SENTRON PAC.

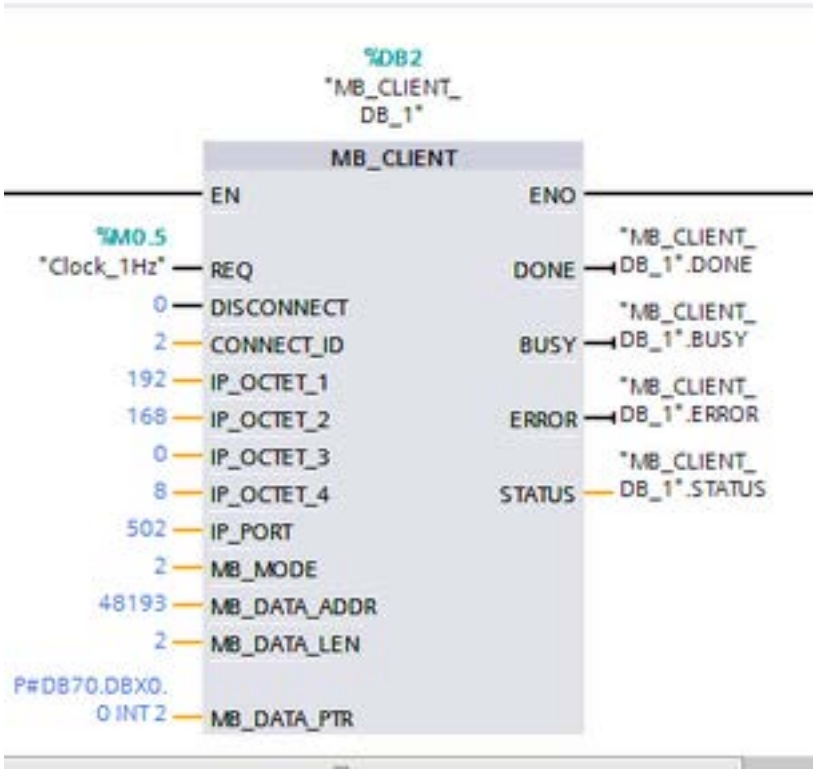

Figure 5. MB_CLIENT for frequency converter reading.

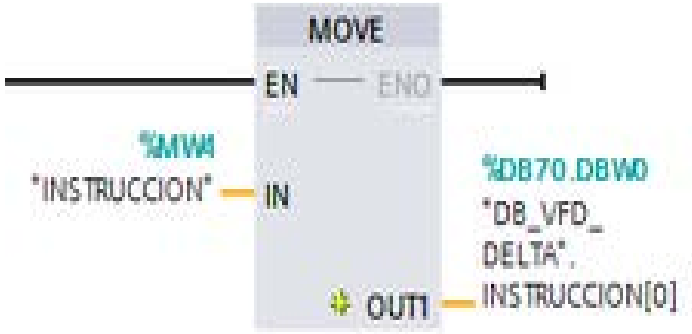

Figure 6. Move for instruction register changes.

This instruction modifies register 48193 of the inverter to be able to control the RUN, STOP, and INVERSION at any moment of time and through the block shown in Figure 7 it is intended to modify the frequency values, all these data are handled through the DB70 instance data block.

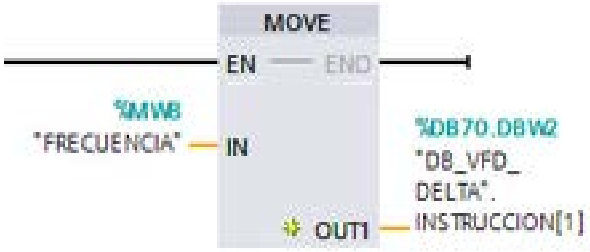

Figure 7. Move for frequency register changes.

It is intended to establish a Modbus-TCP communication between an S7-1200 CPU. Data logging allows the control program to use data logging instructions to store runtime information in permanent log files within a flash memory (CPU or memory card), the format used to store records is Comma Separated Value (CSV).

Data log instructions allow you to create, open, write records and then close the log files, the advantage of this type of instruction is the ease of choosing the values to be stored using a data buffer, i.e. temporary storage is used for a new record.

When all current data values are updated, the DataLogWrite instruction can be executed to transfer data from the buffer to the log, to manage the logs, the web server integrated into the PLC can be used to manage the logs and then store the logs in a computer, and the CSV files can be analyzed using Excel.

2.5 Web Server12

2.5.1 DataLogCreate

Allows to initialize a new log file and automatically open the write operations, the data log is mainly used to store

Table 2. Data types for parameters –DataLogWrite

Parameter and type		Data types	Description
REQ	IN	Bool	Rising edge starts the operation.
ID	In/Out	DWord	Numerical record identifier. Only used as input for DataLogWrite instruction (0)

Table 3. Types of data for parameters –DataLogCreate

Parameter and type		Data types	Description
REQ	IN	Bool	Rising edge starts the operation.
RECORDS	IN	UDint	Maximum number of records before overwriting the entry.
FORMAT	IN	UInt	Registration Format: CSV (1)
TIMESTAMP	IN	UInt	Enable date and time within the log.
NAME	IN	Variant	Name of the record.
ID	In/Out	DWord	Numerical record identifier
HEADER	In/Out	Variant	Record column headers for the top row of the encrypted data array in the CSV file
DATA	In/Out	Variant	Specifies the individual data elements (columns) of a record

the process data according to the execution time inside the PLC memory, the data types used in DataLogCreate are shown in Tables 2-3.

According to (Figure 8), to configure the DataLogCreate, a start bit must be added within the REQ parameter, in RECORDS the number of records to be stored within a data block is set, the output format of the records in CSV is defined within the FORMAT parameter, the user can activate the date and time of the records within TIMESTAMP, to add the file name the parameter NAME is used, the identifier of the block corresponding to DataLog is set to ID, finally the column titles and the data to be stored are added using HEADER and DATA respectively.

2.5.2 DataLogWrite

Write a record in the specified record. For a DataLogWrite operation to be allowed, the existing target record must be open. STEP 7 automatically creates the associated instance DB when the instruction is entered.

Table 2 shows the data types for the parameters for the DataLogWrite block.

The write block configuration shown in Figure 9 requires the REQ parameter defined with the mark of the PLC operating at 1 Hz and the numerical identifier of the ID register.

3. Result and Discussion

3.1 Modbus TCP-(Mb-Client)

By using the Mb-Client block, it is possible to access the information corresponding to variables measured from the SENTRON PAC multimeter such as currents, voltages, powers, etc.

In addition, the operating intervals are defined from the Mb-Client block, the measurement is made with a frequency of 2.5 Hz generated from the PLC cycle marks, within the block the IP address of the multimeter is used within the network is 192.168.0.12. Previously a data block was created in TIAPortal where an array was created with Real data that can be linked to the variable to be monitored, for the case of reading voltages, currents, voltages and power each array consists of three elements per block, however, for variables such as frequency a different block of a single element was used, another alternative to optimize resources is to use a data block that includes all the process variables that will later be shown to the user through an HMI.

MB_DATA_ADDR assigns the initial Modbus address of the data to be accessed. The MB_CLIENT statement uses an MB_MODE entry instead of a function code entry. The combination of the values MB_MODE and MB_DATA_ADDR determines the function code used in the actual Modbus message.

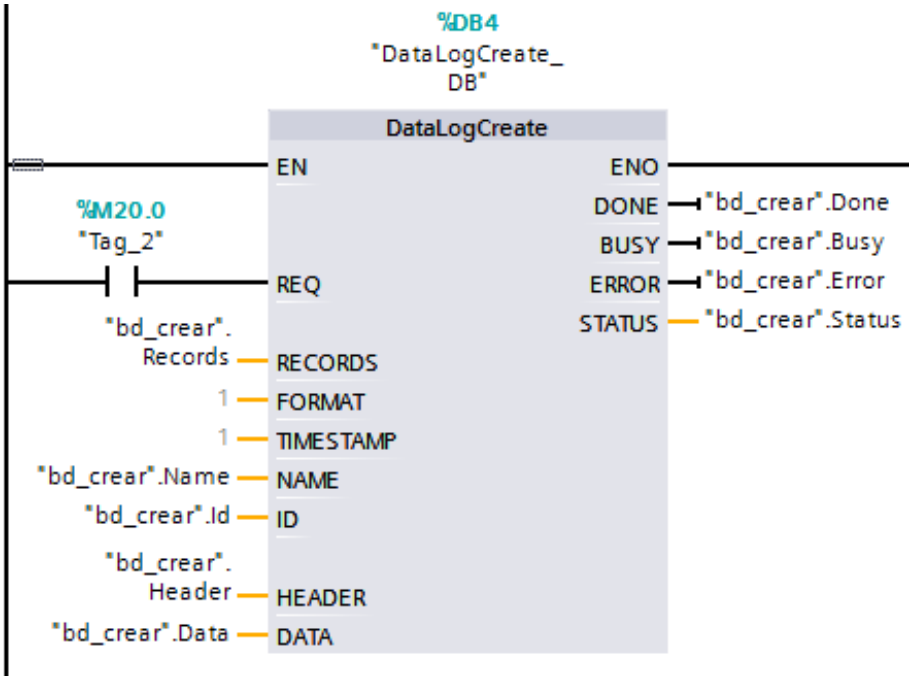

Figure 8. DataLogCreate Instruction.

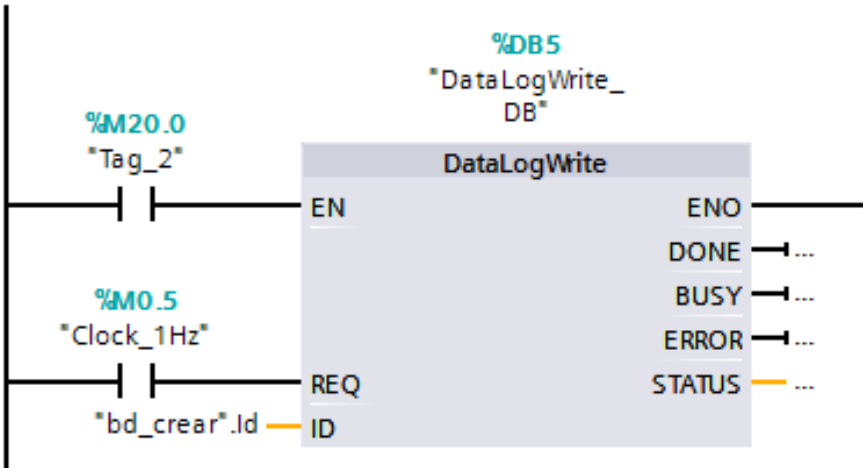

Figure 9. DataLogWrite Instruction.

To read/write the records of the SENTRON multimeter, a range of MB_DATA_ADDR defined from 40001 to 49999 or 400001 to 465535 is used; the initial Modbus address used is shown in Table 4.

Table 4. Data types for parameters - DataLogCreate

Variable	Modbus address	MB_DATA_ADDR
Frequency	55	40056
Voltage A-B	7	40008
Voltage B-C	9	40010
Voltage C-A	11	40012
Current A	13	40014
Current A	15	40016
Current A	17	40018
Apparent power	63	40064
Active power	65	40066
Reactive power	67	40068

Once the reading blocks have been generated, the PLC's on-line connection can be established to verify that there is communication and the respective reading of each variable.

3.2 Modbus RTU-(Mb-Client)

Through the Modbus TRU Network implemented through a client server using the MB_Client instruction that manages a block of DB instance data for the control and monitoring of a frequency converter in charge of controlling the level flow of a tank, the operating frequency and the instruction parameters for RUN, STOP and Inversion, are shown in the following (Table 5).

Table 5. VFD Parameters

Variable	Modbus address	MB_DATA_ADDR
Frequency	38 (Reading)	48193
Run	2	48193
Stop	1	48193
Investment	48	48193

Through an HMI created in Dopsoft independently to monitor and control the frequency values of the inverter it can be observed that the values are identical in both the PLC instance data block and the HMI, this is done as a test object to verify the correct operation of the TCP/RTU network, and Figure 10 shows the parameters mentioned.

Figure 10. Frequency converter readings.

The frequency values shown in the previous figure do not correspond to scaled values, so the measured value or the value to which the frequency inverter is to be set must be multiplied by 100 at the input or the reading value must be divided by the same number to obtain the actual measured value.

With the MOVE instruction shown in Figure 11, an output equal to 2 is generated and entered as a parameter to start the pump that performs the level control of the station. This register is added in the INSTRUCTION parameter created inside the DB70 for the control and monitoring of the inverter, therefore, there is a didactic control, the same configuration for the STOP and INVERSION instruction.

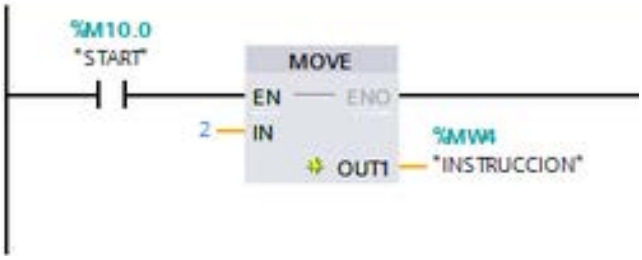

Figure 11. Manipulation of the instruction.

3.3 HIM in WINCC

In the Network topology shown in Figure 12 all the reading and control parameters are monitored from a computer which must have an HMI implemented with all the reading and control addresses of the network, therefore, the implementation is shown below, where you can see the reading parameters of the SENTRON PAC that correspond to both voltages, currents and powers.

These values are taken through the address provided by the DBS instance data blocks configured for each value, in addition, bonuses were implemented that allow easy navigation between windows to observe all network data and control those required.

Figure 13 shows an HMI specifically developed to control the controls of the DELTA frequency inverter, where you can also observe and manipulate the numerical output of the frequency value at which you want the pump of the level control station to operate.

3.4 Graph of Variable Curves

In order to graphically generate the curves of the acquired data, the operating frequency values, one of the line voltages and one of the current are addressed within the

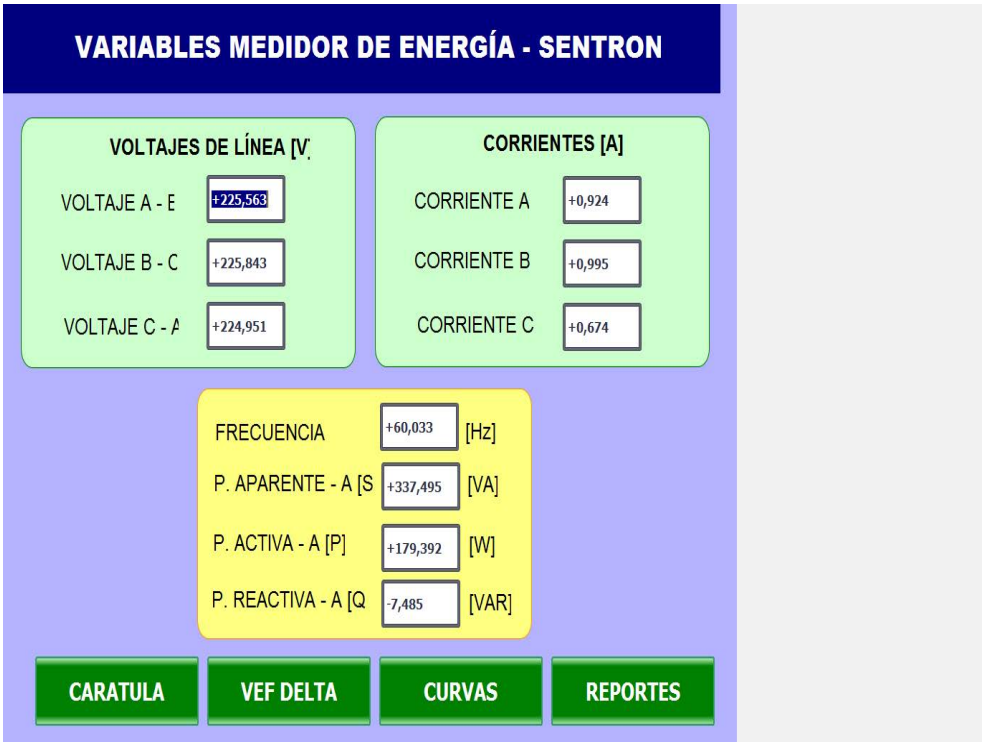

Figure 12. SENTRON PAC Data.

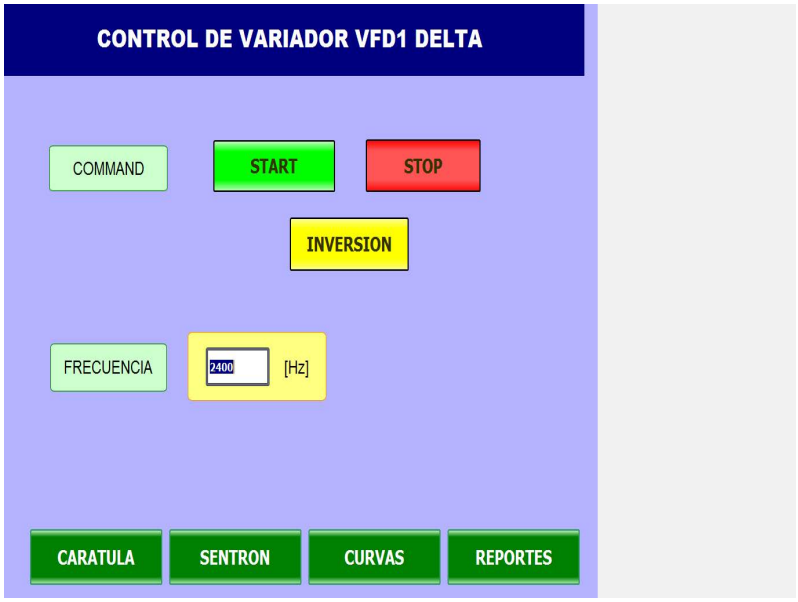

Figure 13. Frequency Inverter Data.

WinCC, which, at the beginning of the sequence, will result in the following (Figure 14).

In the previous figure it can be seen that the Line voltage represented by the red color cure oscillates in the value of 220V as average voltage, with certain intervals where there are voltage rise flanks that are minimum, the black color curve represents the frequency value whose value is 60Hz, corresponding to a suitable value of the network operation, in the lower part a curve is observed in blue color that represents the change of the current value during the operation whose value is greater than zero but there are no significant peaks of variation of the same, all these values are in function of time.

3.5 Web Server

To generate a backup of the information measured from the SENTRON PAC multimeter, the PLC's own Web

Figure 14. Graph of results obtained.

server was used to generate a file who's CSV can be used to insert the variables to be recorded in magnitude, date and time.

Before configuring the DataLogCreate block, a data block must be created with the necessary configuration parameters, that is to say, the maximum number of records to be monitored, the name of the file and the columns to be generated must be defined as shown in Figure 15.

Each variable of interest is copied to the data block corresponding to the DataLogCreate (Data) instruction. The creation of the registers is controlled by the DataLogWrite block, after which the PLC server is accessed using a web browser and the address corresponding to the IP address of the PLC is inserted in the search bar, in this case 192.168.0.6, is accessed with a user name and password defined in the configuration of the controller and in the "File Browser/DataLogs" path the file containing the information of each variable is displayed, in addition, this file can be downloaded to be analyzed from a computer, as shown in Figure 16.

	Records	UDInt	25
	Name	String	'REGISTRO_SEN...
	Id	DWord	16#0
	Header	String	'Voltaje L12, Vo...
▶	Data	Struct	
	Done	Bool	false
	Busy	Bool	false
	Error	Bool	false
	Status	Word	16#0

Figure 15. DataLogCreate Configuration Parameters.

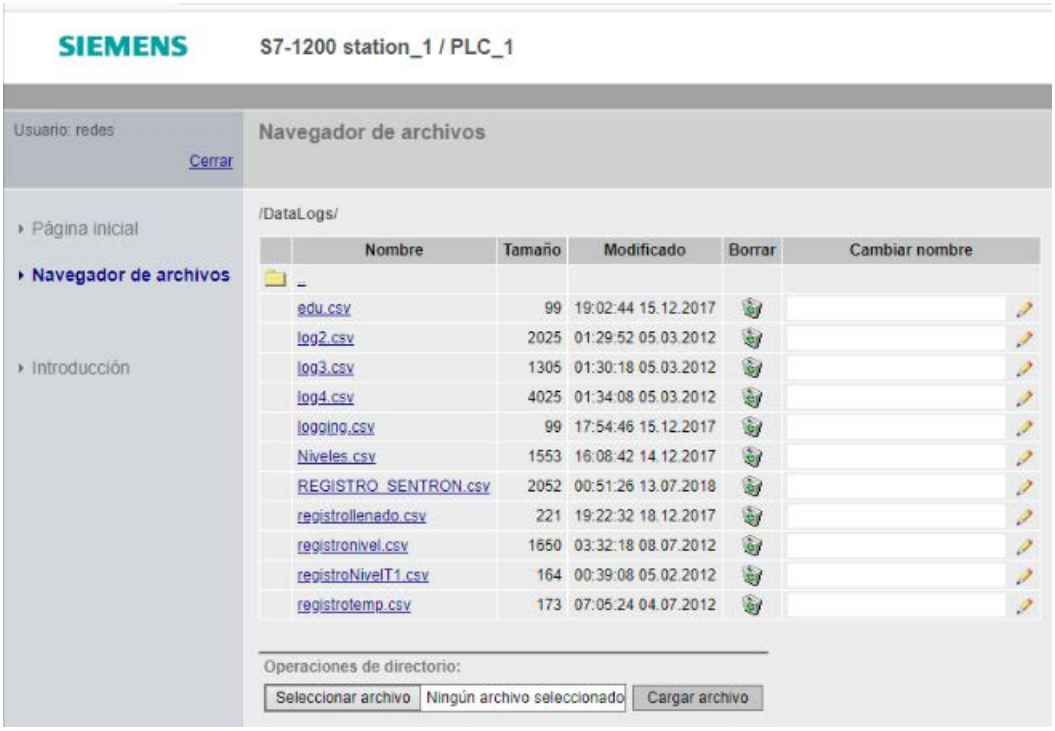

Figure 16. Web server file browser.

4. Conclusion

In order to manipulate the data logs stored in the Siemens web server through the data log, it is necessary to provide administrator permissions through a user name and password on the web page, which are configured in the program that is downloaded to the PLC. Therefore, the main page will contain basic elements for PLC control; on the other hand, when accessing the complete configuration of the PLC with user permissions, it will be possible to access it through its IP address.

Through the Modbus TCP/RTU communion, implemented in the SentronPAC and in the frequency variator respectively, the data emitted by them was captured by programming in the TIA Portal software, which was used for the implementation of an HMI through WinCC, in order to monitor and manipulate the values of the two devices.

The network configured with Modbus TCP/RTU open communication protocols implemented through the MB_Client function block allows configuring devices as a server/client to access the configured records through the instance data blocks, through which the data values of the Sentron PAC Energy meter and the DELTA frequency inverter can be read in real time.

5. References

1. Hurtado J. Introducción a las Redes de comunicación industrial. Departamento de Electricidad-Electrónica. I.E.S. Himilce – Linares; 2018. p. 1–19.
2. Protocolos de comunicaciones industriales. Date accessed: 30/04/2018. https://vestertraining.com/protocolos-comunicacion-redes-industriales/.
3. Xiu-Fen Z, Hong-Yu W. Application of Win CC Software to Building Automation System. International Conference on Future Power and Energy Engineering (ICFPEE 2010); 2010. p. 19–21.
4. Anaguano L, Lucía J. Dise-o e implementación de un sistema de monitoreo remoto de parámetros Eléctricos. Universidad de las fuerzas armadas-espe. Ingeniería en electrónica y automatización. 2013.
5. Perez J, Javier V. Dise-o e Implementación de una Red Industrial Para la Máquina Devanadora RI.TE HTC 700 de Industrial Textilana S.A. Latacunga; 2011.
6. Guerrero V, Martinez L. Comunicaciones Industriales. México DF: Alfaomega; 2010. p. 1–410.
7. SIEMENS. Simatic S7-1200 Programmable controller.
8. Gallo L, Herrera D. Dise-o e implementación de una Red Industrial Utilizando protocolo Modbus y Comunicación Inalámbrica con tecnología Allen Bradley para Monitoreo y control local y remoto de las estaciones de nivel, Flujo Y Presión En El Laboratorio De Redes Industriales Y Control De Procesos De La Espe Extensión Latacunga. Departamento de Eléctrica y Electrónica de la Universidad de Las Fuerzas Armadas ESPE Extensión Latacunga; 2016. p. 1–7.
9. Qué Son Los Sistemas Scada Y Su Importancia En La Industria 4.0. Date accessed: 30.03.2017. https://oasys-sw.com/que-son-sistemas-scada-industria-40/.
10. Sánchez OW. Scada of distributed systems through multiple industrial communication protocols, International Journal of Pure and Applied Mathematics. 2018; 119(15):1–8.
11. Módulo didáctico de una red de comunicación industrial Modbus RTU-TCP. Date accessed: 27/05/2018. https://www.eae-publishing.com/catalog/details/store/pt/book/978-620-2-11332-8/módulo-didáctico-de-una-red-de-comunicación-industrial-modbus-rtu-tcp?locale=gb.
12. Sánchez OW. Design of an HMI in Web Server of PLC's S7-1200/1500 for the Control of a Multivariable Process of a Didactic Module. World Conference on Information Systems and Technologies; 2018. p. 248–56. PMid: 29339738, PMCid: PMC5770411.