DEVELOPMENT OF A SIMPLE GRAPHIC INTERFACE FOR TriOS SENSORS

George Alexandru SMEU[1]

The aim of this experiment is to create a simple software product for real-time monitoring of a TriOS UV-VIS sensor. The monitored equipment consists of the central unit and sensors. Communication between the software interface and the physical equipment will be via Ethernet cable. The main advantages and disadvantages of such an application are discussed, as well as its possible uses.

Keywords: TriOS sensor, graphic interface, HMI, software development, Modbus, standalone software application

1. Introduction

The original approach presented in this paper involves developing a software application, Windows PC based, to be used with a ModBUS TCP protocol device, such as the TriOS UV-VIS sensor. Normally the data from such sensors is collected via various protocols by a PLC, which in turn requires a complete database and SCADA system for the user to interpret. As in the following application, the beneficiary did not have a requirement for a PLC automation and SCADA system, but only the need for a simple HMI capable of remote data transfer and interpretation. For this project, such an HMI has been developed.

TriOS is a German company specialized in optical absorption technology, also known as spectrophotometry. Their sensors are equipped with light emitting sources and light detecting phototransistors. These two components are mounted on either side of a stainless steel or titanium shaft, with a measurement gap in the middle. This positioning allows the measured media, in this case water, to flow through the measurement gap in a continuous fashion. The light emitted by the source is directed using a system of lenses to the measurement gap where it is presented to the water media. On the other side of the measurement gap, also through a system of lenses, the phototransistor unit is installed. This unit receives the remaining light that has passed through the water media in the measurement gap. [1]

[1] PhD student, Electrical Engineering Faculty, University POLITEHNICA of Bucharest, Romania, e-mail: george.smeu@icloud.com

This measurement principle is referred to as photo spectral absorbency or transmittance. Certain chemicals that are dissolved within the water sample can cause a coloration within the water. This coloration can also be visible (with a color present in the visible part of the light spectra) or it can be invisible, detectable within the ultra violet (UV) or infrared (IR) spectra. [2]

If a light source is presented to the water sample, at a certain color (spectra), the chemical with the similar color present in the water will absorb a part of the emitted light. The resulting light intensity is measured by means of phototransistor, and thus the absorption quantity is determined. This absorption is then compared with a built in calibration formula, in order to translate the light absorption value into engineering chemical concentration, such as parts per million (ppm) or milligrams per liter (mg/l). [3][4]

The TriOS sensor is connected via cable to a controller, the TriBox 2. This controller is responsible for setting up the sensor, and for converting the basic raw sensor data into transmission units such as Modbus protocol registries. The values saved in these registries are polled by means of software query answer techniques, such as a Modbus poll. [5]

2. Developing a standalone software application at LukOil

As part of a monitoring equipment supply project, the client requests a simple software application for remote monitoring of parameters outputted by the TriOS sensor. The location of the sensor and software application will be the LukOil Petro-Chemical Plant "Teleajen" Ploiești. The entire monitoring area is considered an Ex (potentially explosive) area, so appropriate equipment should be chosen. The parameters monitored at the entrance to the biological basins of the treatment plant are CCO-Cr [6] (represents the mass concentration of oxygen equivalent to the amount of potassium dichromate consumed for oxidation in acid medium of dissolved and suspended organic matter present in wastewater), phenol concentration and TSS (amount of solid suspension).

The basic principle of the measuring set refers to the acquisition of data by the physical equipment, the display of the latest measurement data on the device screen, the display of the latest measurement data on the software interface, the recording of all data in a database accessible online.

The software must be able to successfully record the information acquired from the measuring equipment in the database, display the data of the last measurement, have an authentication interface for data security and benefit from the possibility to recall the database values.

In principle, the program will consist of 3 windows: The first window will be the authentication window, where the users will enter their access data, and will be directed to the second window if the data is correct. The second window represents the interface between the user and the measuring equipment. It will have several fields that will be updated with the measured information at the interval time set by the measuring equipment. When you press the database button, the program launches a third window in which you can analyze the database registered from the beginning of the process until now (figure 1.). The meaning of the HMI in window 2 refers to the Human Machine Interface.

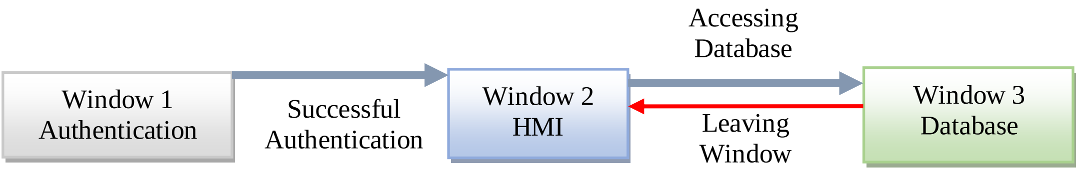

Fig. 1. Program architecture

Fig. 2. TriOS ProPS and TriBOX 2

The selected equipment (figure 2.) is configured for data transmission to a Modbus / TCP master. The configured communication port is 502, the set IP address is 192.168.0.50 and the Modbus address of the Modbus / TCP slave will be 20. This address in the Modbus network is assigned to the translation algorithm, called LSA (Linear Substance Analysis).

The protocol is largely a "query and answer" system. When querying certain memory registers of the measuring equipment, it will respond with the binary value of the data contained in the registers in question. In the first instance, the memory addresses to be queried are set to obtain the desired data set. Then you need to specify and consider what type of information a data set contains (character, integer, float, etc.). Finally, the information is saved in the database in translated binary form. [7]

In order to know what information is required of the measuring equipment and at which addresses it is located, the manufacturer provides Table 1.

Table 1

Data registries table [8]

Poll address	Data type	Read/Write	About
20-39	String	Read	Equipment name, for example LSA
40	Boolean	Read/Write	On (1) Off (0)
41-46	Date and time	Read	Sample interval
47	Short integer	Read	Number of concentrations to be read
1000-1063	Float 32	Read	Up to 32 measurement results in floating point. Usually, a module has fewer results and only the first values are used.
1064-1383	Float 32	Read	Name of results. Each name can be up to 20 characters long, so 10 registers are reserved for this.
1384-1639	Float 32	Read	Name of measurement units. Each name can be up to 16 characters long, so 8 registers are reserved for this.

The sensor's response to user queries will be in the form of 16-bit data packets. At the same time, it is very important to know the type of data received from the equipment, in order to translate them from binary into general use information (values, names, units, etc.).

The information required by the user is:

- To find out the name of the equipment, call the registers from no. 20 to no. 39. The data is of string type;
- To find out the date and time of the last measurement performed by the equipment, call the registers from no. 41 to no. 46. The data are of date and time (integer) type;
- To find out the values of the last measurement performed by the equipment, call the registers from no. 1000 to no. 1063. The data are of 32 Float type;
- In order to find out the name of the measured value (physical significance), the registers from no. 1064 to no. 1383 are called. The data are of string type;
- To find out the name of the unit of measurement of the measured value, call the registers from no. 1384 to no. 1639. The data are of string type.

During the program run, certain instructions must be invoked to automatically send the read information to the database. Thus, in the following, the database is set up.

In the MySQL Workbench [9] programming environment, the link to the local MySQL server that will be used during the operation of the program is created. A database named "trios" (figure 3.) is created. Two tables are created inside this database: "login", which will contain the data needed to authenticate inside the program, and the second table, "date", which will contain all the information stored during the running of the program.

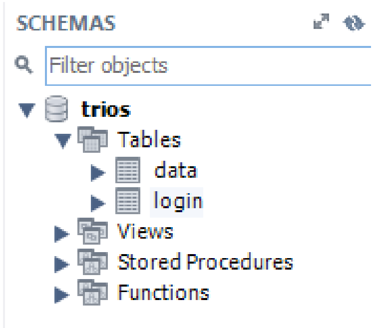

Fig. 3. The database

The "login" table (figure 4.) contains three columns: "Nr" for the number, "username" for the username, "password" for the user password.

Nr	username	password
1	1	1
2	admin	admin
NULL	NULL	NULL

Fig. 4. „Login" table

The "data" table (figure 5.) contains: "Nr" for the criterion number, "Day" for the day of the last measurement, "Month" for the month of the last measurement, "Year" for the year of the last measurement, "Hour" for the last measurement hour," Minutes "for the last measurement minute, "Second" for the last measurement second, "Name1", "Value1" and "Unit1" for the name, value and unit of measurement for the first measured parameter, "Name2", "Value2" and "Unit2" for the name, value and unit of measurement for the 2nd measured parameter, "Name3", "Value3" and "Unit3" for the name, value and unit of measurement for the 3rd measured parameter.

Nr	Day	Month	Year	Hour	Minute	Second	Name1	Value1	Unit1	Name2	Value2	Unit2	Name3	Value3	Unit3
1	22	05	2018	12	15	43	COD	234,12	mg/l	BOD	45,36	mg/l	TSS	425,12	mg/l
2	22	05	2018	12	17	43	COD	235,56	mg/l	BOD	43,20	mg/l	TSS	445,35	mg/l
3	22	05	2018	12	19	43	COD	234,22	mg/l	BOD	44,15	mg/l	TSS	453,25	mg/l
4	22	05	2018	12	21	43	COD	248,69	mg/l	BOD	45,89	mg/l	TSS	425,36	mg/l
5	22	05	2018	12	23	43	COD	236,57	mg/l	BOD	41,33	mg/l	TSS	485,36	mg/l
	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

Fig. 5. „Data" table

The "Nr" column of the "data" table (figure 6.) is set with the auto increment function, which means that with each "insert" command in the database, the server automatically moves to a new row, incrementing the value in the "Nr" with 1.

Column Name	Datatype	PK	NN	UQ	BIN	UN	ZF	AI	Default
Nr	INT(11)	✓	✓					✓	
Day	VARCHAR(45)								NULL
Month	VARCHAR(45)								NULL
Year	VARCHAR(45)								NULL
Hour	VARCHAR(45)								NULL
Minute	VARCHAR(45)								NULL
Second	VARCHAR(45)								NULL
Name1	VARCHAR(45)								NULL
Value1	VARCHAR(45)								NULL
Unit1	VARCHAR(45)								NULL
Name2	VARCHAR(45)								NULL
Value2	VARCHAR(45)								NULL
Unit2	VARCHAR(45)								NULL
Name3	VARCHAR(45)								NULL
Value3	VARCHAR(45)								NULL
Unit3	VARCHAR(45)								NULL

Fig. 6. Building the „data" table

The Dev-C ++ development environment [10] is used to develop the acquisition programs. Independent scripts are used in order to query the different registries of data required from the sensor. The result received is read and interpreted. In case of error, a methodology is designed to describe the error, to interpret the incomplete information, to recall in case of incomplete information, and to give up in case of inadmissible values. All valid data is saved in the SQL database. For the acquisition of the date and time of the last measurement performed by the equipment, it is not necessary to convert the information received from it, as it is coded as standard in the form "Date & Time", integer. After the launch of this script, entitled "modbus_date_time.cpp", the following information is obtained in the database (figure 7.):

Nr	Day	Month	Year	Hour	Minute	Second
1	22	05	2018	12	15	43

Fig. 7. The database after running „modbus_date_time.exe" script

Following the “modbus_values.exe” script, the database will be completed as follows (figure 8.):

Name1	Value1	Unit1	Name2	Value2	Unit2	Name3	Value3	Unit3
COD	234,12	NULL	BOD	45,36	NULL	TSS	425,12	NULL

Fig. 8. Database after running „modbus_values.exe” script

For the measurement unit, the interpretation of the data is similar to the part of the program that calls the name of the measured parameters. The results are also Character type and must be displayed sequentially next to the Unit 1, Unit 2, Unit 3 boxes. This program is the simplest of the three because it interprets only three strings. After running the program, the database is completed as follows (figure 9.):

Value1	Unit1	Name2	Value2	Unit2	Name3	Value3	Unit3
234,12	mg/l	BOD	45,36	mg/l	TSS	425,12	mg/l

Fig. 9. Database after running „modbus_units.exe” script

With the help of the Visual C ++ 2010 Express development environment [11][12], all the graphical interface windows were created in a timely manner. Initially, the possibility of user authentication was implemented to prevent the possibility of data theft. This page will contain a username field, a user password field, an exit button, a confirmation button, the developer logo, and a link to the developer webpage. The authentication window looks like this (figure 10.):

Fig. 10. Authentication window

Once the authentication is verified with the database, window 1 disappears and window 2 is launched. Window 2 is shown in (figure 11.).

Window 3 will contain a field for displaying the database (if Database is pressed), a load button for the database, and an exit button. Window 3 looks like this (figure 12.).

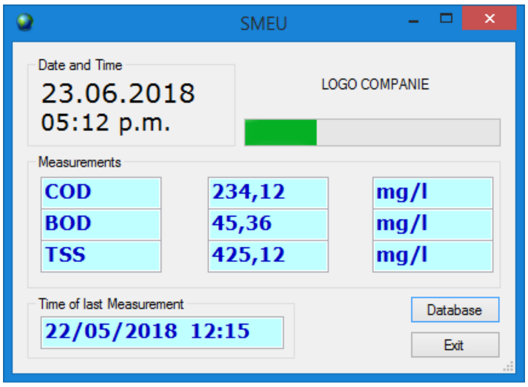

Fig. 11. HMI Window

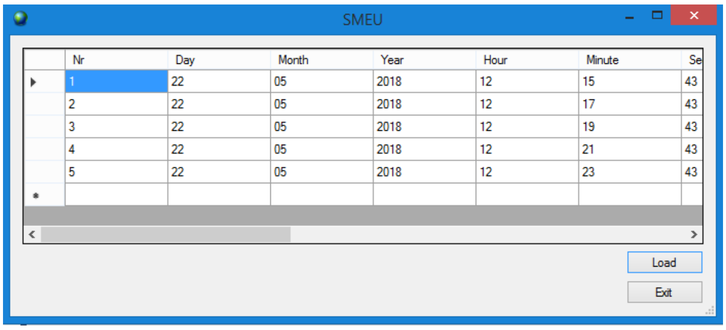

Fig. 12. Database window .1

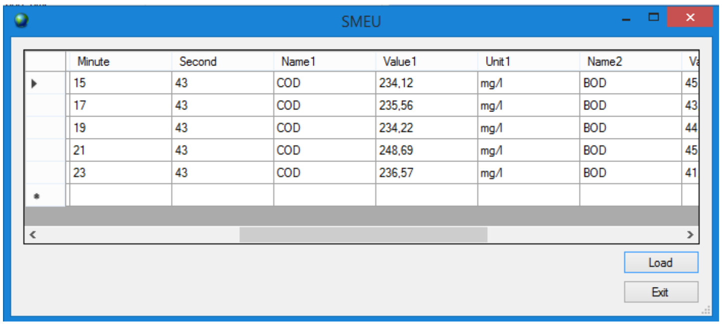

Fig. 13. Database window .2

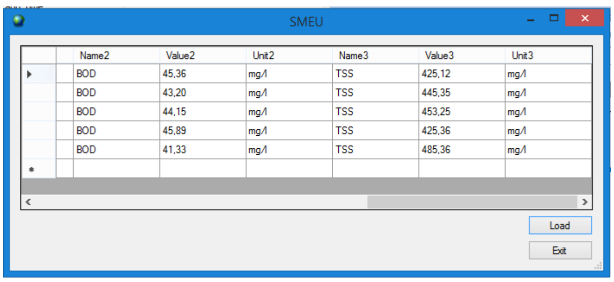

Fig. 14. Database window .3

To sum up the complete functioning flow of the application software, the following diagram is used:

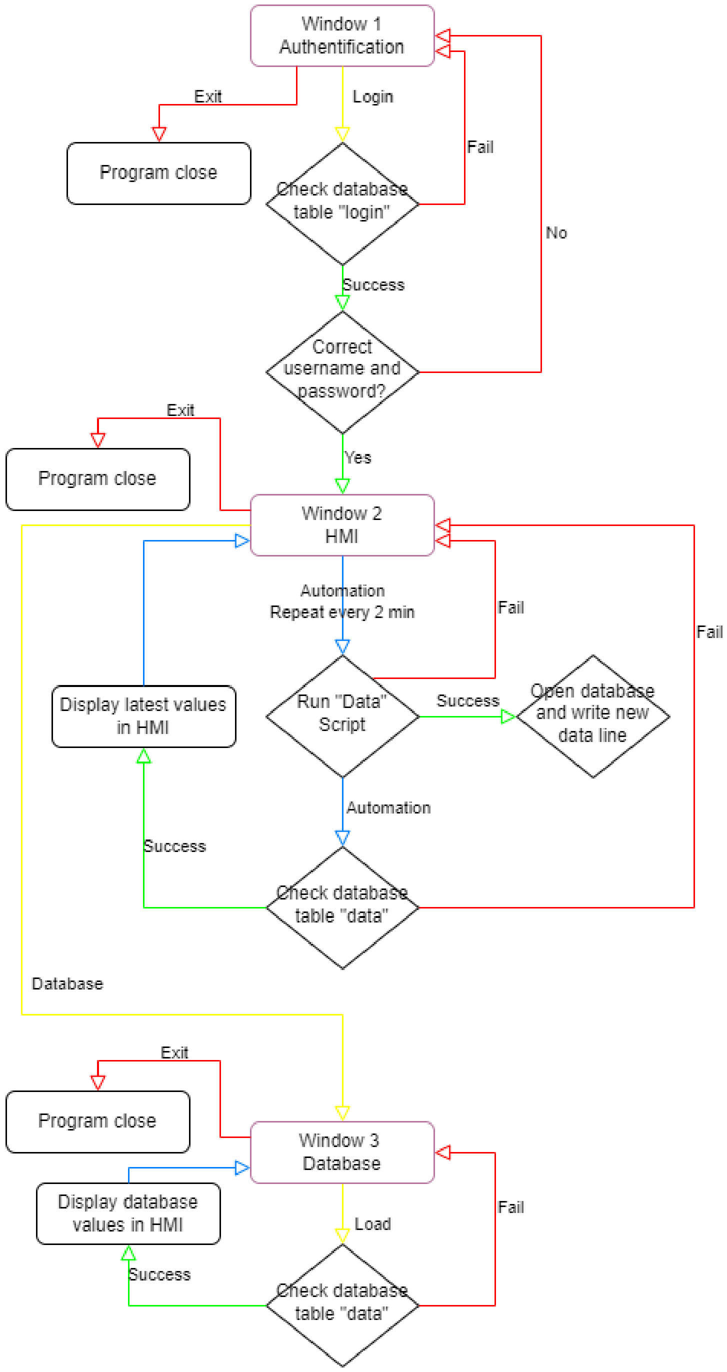

Fig. 15. Software flow diagram

3. The application's advantages and disadvantages

The above presented application is a simple custom build for the TriOS ProPS LSA sensor. All data acquired by means of custom developed scripts is interpreted and saved in the MySQL database in a specific format. When calling up the database from the data HMI windows, the program is built specifically to read the database in a custom way so as not to display wrong values to the operator.

This entire software pack is custom built for the TriOS sensor. The main advantage is that there will be no errors in reading data, writing data to the database, displaying the acquired information in the HMI. Also, the program is already set for future improvement and development, such as: a custom date and time period in which the data to be displayed, a custom graph plotting option, a database export function, a comparison function between different sets of data and so on.

The main disadvantage is that the program is custom built to the TriOS sensor. If at one point a different type of sensor is to be used, or the current sensor becomes obsolete, or if the new sensor requires a different type of data acquisition language, the entire script needs to be updated to the new communication protocol. The HMI window and the Database window can remain the same because the data is recalled from the current database, but the data query and data save to the database need to be updated.

In comparison with commonly used OPC servers from PLC manufacturers such as SIEMENS, EATON or SCHNEIDER ELECTRIC, this lone software solution does not require such high investment in hardware and software packs. When comparing the complexity of the TriOS custom-built software solution with the PLC + Server + SCADA solution, the former is much simpler by design and easier to implement and maintain in comparison with the latter.

Other similar software products, such as QModMaster, IFM's AL1341, Micro820 Modbus TCP Master and so on, will offer the possibility of Modbus TCP data acquisition, post processing and display, to a certain extent. These solutions either lack the simplicity of a custom-built HMI, or the database linked to the data acquisition and recall window or require hardware modules to be installed between the data source and the operator's PC interface. All the required hardware installed in case of the TriOS software solution is an ethernet cable between the TriOS controller and the operator's PC, or closest switch/router.

4. Conclusions

During operation, the application displays the desired parameters, together with their measured values and measurement units, the tabled database, easy to interpret, prevents accidental exit from the application through the confirmation dialog window, prevents fraudulent launch through the authentication system in the window 1.

The entire software pack is fast, light and easy to run. Most of the processes operate in the background so that the operator is not distracted by needless data. In case the script cannot pull the data from the sensor, specific error messages are show for the operator to interpret the problem. The utility of this software product can be summed up to its lack of complexity, and for this application this best fits the beneficiary's needs. In this case, a quality Laboratory PC unit was used to directly access the data from the TriOS sensor. There is no need for complex SCADA, PLC and other hardware installations. A simple and direct PC-Sensor solution was required, designed and delivered. Such and application can be easily adapted to any type of sensor that can communicate via Modbus protocol, by simply changing the data registry address in the query letters. Even if an analog sensor is used, a programmable logic controller (PLC) can acquire the 4-20mA signal. The measured value can be interpreted and scaled directly to a concentration value, which is then saved to a specific data registry address. Thus, the presented application is universally adaptable. As a result, this particular application was a success given the fact that the beneficiary could benefit from the real time remote data acquisition this software provided. The sensor installed in an ATEX area is now easily under observation from a safe location, and the local laboratory tasked with data comparison can easily access the data via the software HMI and compare results.

R E F E R E N C E S

[1] Germer T., Zwinkels J., Tsai B., Spectrophotometry: Accurate measurement of optical properties of materials, cited 16.01.2022

[2] Lindon J., Tranter G.E., Koppenaal D., Encyclopedia of Spectroscopy and Spectrometry, cited 16.01.2022

[3] Thomas O., Burgess C., UV-visible spectrophotometry of water and wastewater, second edition, cited 16.01.2022

[4] Gryczynski Z.K., Gryczynski I., Practical fluorescence spectroscopy, CRC Press, cited 16.01.2022

[5] Vacca J.R., Handbook of sensor networking, Taylor & Francis Ltd, cited 16.01.2022

[6] Narayanaswamy R., Wolfbeis O.S., Optical Sensors: Industrial environmental and diagnostic applications, cited 16.01.2022

[7] RTA automation [online]. [2017] [cited 18.06.2017], URL: http://www.rtaautomation.com/modbustcp/

[8] Krause S. E., TriOS Tribox 2 Modbus Slave, Technical support, TriOS Mess- und Datentechnik GmbH

[9] MySql Work Bench [online]. [2017] [cited 18.06.2017], URL: http://mysqlworkbench.org/learn/

[10] Bloodshed [online]. [2017] [cited 18.06.2017], URL: http://www.bloodshed.net/devcpp.html

[11] Locke B., Visual C++ for Visual Basic Developers, 2002.

[12] Iulia Ioana Anghel, Roberta Ștefana Călin, Maria Lavinia Nedelea, Iulia Cristina Stănică, Cătălin Tudose, Costin Anton Boiangiu, Software development methodologies: a comparative analysis, U.P.B. Sci. Bull., Series C, Vol. 84, Iss. 3, 2022