# ANALYSIS OF THE DATA TRANSMISSION IN THE INDUSTRIAL AUTOMATION SYSTEMS

Theodor BORANGIU[1], Marian CROITORU[2]

*În această lucrare este prezentat un protocol de comunicație conceput pentru transferul de date între sistemele de automatizare ce conțin controlere programabile. Prin introducerea unui bit suplimentar pentru identificarea unui caracter de adresă sau date, se reduce volumul de date ce trebuie procesat de stațiile slave. Micşorarea perioadei ciclului de interogare, pentru o mai rapidă actualizare a parametrilor, este realizată prin minimizarea numărului de caractere transmise în funcție de tipul parametrilor. Pentru a elimina erorile datorate distorsiunilor fiecare caracter este transmis în mod asincron.*

*In this paper is presented a communication protocol designed to the data transfer among the automation systems which include programmable controllers. By introducing an additional bit to identify an address or data byte, reduce the volume of data to be processed by the MCUs slave. Decreasing the slave station interrogation cycle period, for faster update of parameters, is realized by minimizing the number of transmitted characters depending on the data type. To eliminate errors due to distortions each character is sent asynchronously.*

**Keywords:** industrial automation, data transmission, communication protocols

## 1. Introduction

The information flowing in the data transmissions is discrete and it is expressed using alphabetical characters, or digital coded numerical ones. As a rule, a continuous potential difference corresponds to each symbol. These signals are called basic band signals. The successive transmission, one after one, of the symbols representing the data is called serial transmission.

In order to identify the symbols at the reception, it is firstly required their defining. From this point of view, there are two types of transmission: synchronous and asynchronous [1].

In the case of the synchronous transmission, the serial symbols, transmitted without pauses, have, all of them, the same duration. The receiver must know this duration and it also must be synchronized with the received

[1] Prof., Faculty of Automation and Computer, University POLITEHNICA of Bucharest, Romania, e-mail: borangiu@cimr.pub.ro

[2] PhD student, Faculty of Automation and Computer, University POLITEHNICA of Bucharest, Romania, e-mail: mariancroitoru_tgv@yahoo.com

symbols. It must sample the received signal at intervals equal to the duration of a symbol and to make decision after each sampling about the received symbol type.

The asynchronous transmission does not need equal duration symbols. It can be used a particular symbol in order to facilitate the message symbols separation, or, through the used code, it is ensured the condition that two successive symbols not to be identical.

A combination of these two transmission methods is the so called “start – stop” transmission. The symbols representing a character are transmitted timely, and these symbols group is preceded by a symbol, called start, which marks the beginning of a character and has the same duration as each character's symbol, and it is followed by a symbol, called stop, which marks the end of the character and has a duration at least equal to that of an information symbol. In such a way, the interval between the characters can be variable without influence the reception.

In the synchronous transmission, as well as in the “start – stop” transmission, the significant signal states are changing at regular periods. The duration of the shortest significant state determines the transmission speed.

When elaborating a code, it is necessary to take into account: the nature of the messages which must be transmitted, protection assured against the disturbances, efficient use of the transmission line [2].

Fig. 1 presents some modes of symbols representation.

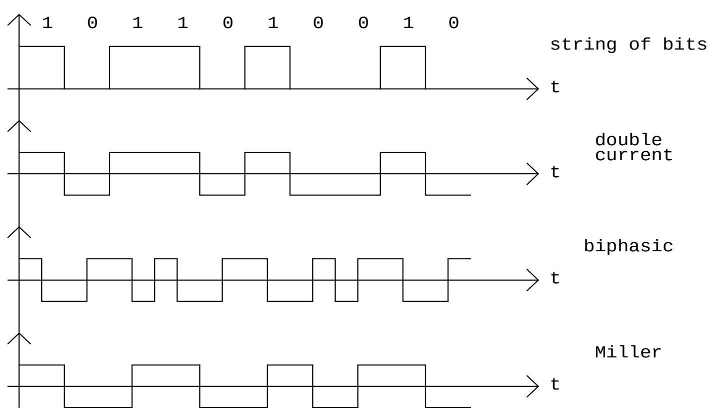


Fig. 1. Basic band signals

The double current transmission presents a better protection against the noise and a more reduced telegraphic distortion. These are the reasons for which it is used the representation through double polarity impulses for the distance data transmission in the basic band [3].

## 2. Factors inducing errors in the data transfer

A criterion for evaluate the data transmission quality may be the percentage of the receiver's incorrect decisions regarding the transmitted symbols or characters. Because, most of the time, a transmission system operates under the error threshold, the quality determination is done depending on the distortion factor.

The individual distortion factor, corresponding to a transition, is given by the ratio between that transition actual moment deviation compared to the corresponding ideal moment (+/-Δt) and a symbol time t

$$\delta = \frac{\pm \Delta t}{t} \cdot 100 \quad (\%) \tag{1}$$

Distortion appears as the signal elements lengthening or shortening. It is caused by the limitation of the frequencies spectrum of the signal formed from rectangular impulses, limitation which determines the symbols interference. In the data transmission systems, the frequency band used is limited by the line on what the transmission is done.

Noting with $x(t)$ the transmission system answer to an impulse with its amplitude equal to one, the signal $y(t)$ from the reception, corresponding to a symbols sequence $\{a_n\}$ at the system entry, will be:

$$y(t) = \sum_n a_n x(t - nT) + z(t) \tag{2}$$

z(t) being the component given by the additive noise.

At the moment $t = kT + t_0$, to being the time of propagation through the system, the actual value is:

$$y(kT + t_0) = \sum_n a_n x(kT - nT + t_0) + z(kT + t_0) \tag{3}$$

Noting $y(kT + t_0) = y_k$

$$y_k = \sum_n a_n x_{k-n} + z_k \tag{4}$$

At the moment $kT + t_0$ the answer have the size $a_k x_0$

$$y_k = a_k x_0 + \sum_{n(n \neq k)} a_n x_{k-n} + z_k \tag{5}$$

The sum term is due to the symbols interference, interference appears due to the dilatation in time of the system answer to each particular impulse $a_k$ as a result of the signal frequencies spectrum limitation.

The random distortion factor is the distortion factor deviation compared to an average value. This results mainly from the noise [3].

### 3. Communication among systems

Data link assures the connection between two automation systems. The communication protocol on the data link designates a set of conventions and rules regarding the format, integrity, time disposal of the messages change between two processes which are in inter-communication and each of them is found in an automation system, the two systems being directly inter-connected [4].

For the most of the cases, the messages which must be changed between the two systems are too long and that is why the string of bits must be segmented so sections of the message to be emitted, which shall be confirmed as they are sent.

From the synchronization point of view, the protocols are character oriented or bit oriented [5]. The character oriented protocols are based on a transmission with byte synchronization. The message consists in a series of bytes, being delimited by a synchronization sequence which determines the message's first byte position. A character is good to become a synchronization character if its code, cyclically rotated, does not become identical to itself, allowing in this way its unique delimitation in the synchronization sequence framework [6].

In Fig. 2 the data transmission frame is presented.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| HEADER | FIRST INDICATOR FRAME | DESTINATION ADDRESS | SOURCE ADDRESS | DATA FIELD LENGTH | DATA | CRC | END INDICATOR FRAME |

Fig. 2. Data transmission frame

The bit oriented protocols are based on bit synchronization. For the bit synchronization transmission, the message is treated as a string of bits. There is a unique bits sequence, having the structure 01111110, which delimits its end.

The message end can be controlled through special characters or through the characters counting.

The data link control protocols include the transmission errors detection [4] [5]. The errors detection is done using one of the following methods: VRC/LRC-8, VRC/CRC-16, CRC-12, CRC-16. After an error presence is signaled, its correction shall be done, by generally using the re-transmission. The re-transmission technique is strongly bound to the data flow control on the connection [7].

O source having the alphabet *X*, with the probability set $P_x$ and which generates the typical series of length N

$$[X] = [x_1 x_2 ... x_n] \tag{6}$$

$$[P_x] = [p_1 p_2 ... p_n] \tag{7}$$

$$[X_n]=[x_{i_1}x_{i_2}...x_{i_N}] \tag{8}$$

where: $i=\overline{1,n}$

All the typical series have the same probability:

$$p(X_n)=p_1^{N_1}p_2^{N_2}...p_n^{N_n} \tag{9}$$

where $N_1N_2......N_n$ represent the symbols number $x_1x_2.....x_n$ from the series *Xn*

For an N very big: $N_i=Np_i$ (10)

The information quantity got when a series *Xn* is realized is:

$$I(X_n)=-\log p(X_n)=-N\sum_{i=1}^{n}p_i\log p_i \tag{11}$$

The own average information on symbol (entropy), at the communication channel entry, is equal to:

$$H(X)=-\sum_{i=1}^{n}p(x_i)\log p(x_i) \tag{12}$$

A source information flow is given by the product between the source entropy and the symbols average number in a second.

The events field entropy at the destination is:

$$H(Y)=-\sum_{j=1}^{m}p(y_j)\log p(y_j) \tag{13}$$

The average error is given by the entropy *H(Y/X)* and represents a measure of the output field uncertainty when the input field is known:

$$H(Y/X)=-\sum_{i=1}^{n}\sum_{j=1}^{m}p(x_i,y_j)\log p(y_j/x_i) \tag{14}$$

where: $p(x_i,y_j)$ is the probability to emit $x_i$ and to receive $y_j$

$$p(y_j/x_i)=\frac{p(x_i,y_j)}{p(x_i)} \tag{15}$$

If the channel has no disturbances, by the reception of the symbol $y_j$ there is the certainty for the transmitted symbol $x_i$. In this case, *H(Y/X)* = 0.

A channel which is uniform both to the input and to the output is a double uniform channel. In this case, it is compulsory that the number of symbols from the source alphabet to be equal to the number of symbols of the receptor alphabet, that is n = m.

In order to determine a measure for the information transmission efficiency and to find its upper limit, Shannon defined the channel capacity as being the trans-information maximum value

$$C=\max I(X;Y)=\max[H(Y)-H(Y/X)] \tag{16}$$

The symmetrical binary channel capacity is:

$$C = \max H(Y) + p \cdot \log p + (1-p) \cdot \log(1-p) \tag{17}$$

For p(y1) = p(y2) it results max H(Y) = 1

So, the channel capacity, when the input symbols x1 and x2 are used with the same probability, is:

$$C = 1 + p \cdot \log p + (1-p) \cdot \log(1-p) \tag{18}$$

If we introduce redundant symbols in the code word, these can be used for errors detection or correction.

$$v = [a_1 ... a_m a_{m+1} ... a_n] \tag{19}$$

Where: $a_1 a_2 ... a_m$ are redundant symbols

$a_{m+1} a_{m+2} ... a_n$ are information symbols

The determination operation of the redundant symbols depending on the information symbols is called coding. The redundant symbols are obtained from the information symbols linear combinations.

The code words so compound have the property to be recognized as such because they have the corresponding corrector null. If during the transmission process some errors are introduced, the corrector is different of zero, and the system can perform the errors correction.

In order to detect the errors, there are used codes with the parity checking or polynomial codes.

The parity checking is performed by each character supplementary introduced bit testing so that the total number of symbols „1" to be even or uneven.

The crossed parity checking is performed to the data blocks formed from a series of characters to what it is added a checking character whose symbols are determined by the same rule of the parity determination, taking into account symbols of the same position of all the characters from which the block is formed for each symbol from this character.

In the polynomial codes, a code polynomial can be associated to each code word, whose coefficients are binary symbols which form the code combination.

The correction through re-transmission is performed as follows:

- the block reception way is confirmed positive or negative
- the correctly received blocks are confirmed only. In case of non confirmation, after a while, the information block is re-transmitted

## 4. Industrial communication systems

The performance of control systems is no longer simply determined by the programmable logic controllers, but also to a great extent by the environment in which they are located. Apart from plant visualization, operating and monitoring, this also means a high-performance communication system.

Distributed automation systems are being used increasingly in manufacturing and process automation. This means that a complex control task is divided into smaller subtasks with distributed control systems.

Communications networks are the backbone of modern automation solutions. They allow the interconnection and exchange of information between individual automation components and devices. A particularly important aspect is the integrity of the data. Problems affecting the data transmitter or the transmission medium must be detected and signaled so that the user can rely on the information being correct[8].

One automation solution is seldom the same as another. An industrial communications system must be flexible and capable of adaptation to the particular situation on the shop–floor.

PROFIBUS and PROFINET are standardized solutions characterized by their unusual ability to combine total integration with a high degree of application orientation [9]. With its standard protocol, PROFIBUS encompasses all sub-processes found in factory and process automation, including safety-related communication and motion control applications [10]. It thereby provides the ideal basis for ensuring horizontal automation system integration. PROFINET also features a standard protocol which, in addition to horizontal communication, also supports vertical communication, thereby linking the field level with the corporate management level. Therefore, both communication systems are able to facilitate network-wide, integrated solutions that are optimized for the relevant automation tasks [11] [12].

PROFIBUS devices communicate using the standardized PROFIBUS DP (Decentralized Periphery) communication profile which defines the rules governing communication. At the heart of the communication profile is what is known as the master/slave concept, whereby a master (active communication peer) polls the associated slaves (passive communication peers) cyclically. When polled, a slave will react by sending a response frame to the polling master. A request frame contains the output data, e.g., setpoint speed of a drive, and the associated response frame contains the input data, e.g., the latest measured value from a sensor. In one bus cycle, the master polls, exchanges I/O data with all associated slaves. This polling cycle is repeated as fast as possible. This action is initiated by the master (typically under user program control) between I/O cycles to read and/or write slave parameter data. This type of communication is referred to as acyclic communication. There can be more than one master on a PROFIBUS system. In such systems, access rights are passed from one master to the next (token passing).

At the heart of the system is the PROFIBUS DP protocol, which is identical for all applications.

PROFIBUS PA (Process Automation) designates a specific selection of PROFIBUS technology components (modular system components) meeting the particular requirements of process automation. PROFIBUS PA encompasses all technology components used to connect intelligent field devices to controllers, control systems and engineering stations and offering ideal solutions for process automation.

The PA profile classifies the devices used in process automation as transmitters, actuators, devices for digital inputs and outputs, or analyzers. For each device class, the profile specifies the associated functions and parameters which can be used to adapt the device functions to the individual application and process conditions. The specification is based on function blocks and parameters types are classified as input, output and internal. The profile also specifies how the services of the PROFIBUS communication protocol are used. For example, process data exchanged cyclically is based on a standard format for all process automation devices. In addition to the measured value and/or manipulated value, this format also features a status byte providing information about the quality of the value and possible limit violations.

MBP (Manchester-encoded, Bus Powered) technology, a 2-wire technology which combines the functions of data transmission and power supply, is usually used on PROFIBUS PA.

The connection between a PROFIBUS DP and a PROFIBUS PA segment is accomplished using segment couplers or DP/PA links. Essentially, both components perform the following tasks:

- converts the asynchronous RS485 bus physics into the synchronous MBP bus physics
- supplies voltage for the PA segment and limits the segment current supply
- decouples the transmission speeds of RS485 and MBP bus physics

The DP/PA link appears on the DP bus as a separate modular slave device with the connected PA devices appearing as plug-in “modules”. An essential feature of the DP/PA link is the provision of a totally isolated address space for its connected PA devices (nontransparent solution). It has to be configured separately and restricts the total amount of data which can be transferred to and from the connected PA devices to 244 bytes. The cyclic data from all the connected PA devices is compressed into a single DP telegram.

## 5. Implementation of a communication protocol for industrial automation systems

The programmable controllers, used in the automation systems, have at least one serial interface which can operate synchronously or asynchronously with a variable number of data bits [13]. Setting the Multi-processor Communication mode enables a filtering function of incoming frames received by the USART Receiver. Frames that do not contain address information will be ignored and not put into the receive buffer. This effectively reduces the number of incoming frames that has to be handled by the CPU, in a system with multiple MCUs that communicate via the same serial bus.

If the Receiver is set up for frames with 9 data bits, then the ninth bit is used for identifying address and data frames. When the frame type bit (the ninth bit) is one, the frame contains an address. When the frame type bit is zero the frame is a data frame.

For an MCU to act as a Master MCU, it uses a 9-bit character frame format. The ninth bit must be set when an address frame or cleared when a data frame is being transmitted.

The communication protocol we have devised utilizes frames containing characters of 8 data/address bits, a selection bit, a parity bit for errors checking and synchronization bits (start and stop bits). In Fig. 3 the framework structure is presented.

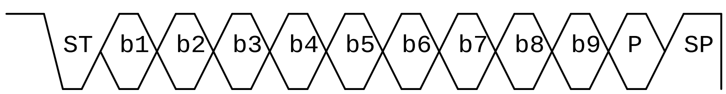


Fig. 3. Frame format

Where:
ST – start bit
b1-b8 – data/address bits
b9 – selection bit
P – parity bit
SP – stop bit

The start bit is always low and the stop bit is always high. Parity can be odd or even.

The bit b9 is used for an address frame indication (b9 = '1') or data (b9 = '0') transmitted by the MCU master towards MCUs slave.

The Multi-processor Communication is done by first decoding an address frame to find out which MCU has been addressed. If a particular Slave MCU has been addressed, it will receive the following data frames as normal, while the other Slave MCUs will ignore the received frames until another address frame is received. This effectively reduces the number of incoming frames that has to be handled by the CPU.

In order to reduce the slave stations interrogation cycle period used different frames numbers for the data exchange depending on the parameters type / transmitted data. The minimum characters number is three: a character for the address, a character for the control and a character for the data. The maximum number of the transmitted characters to an interrogation is 65: a character for the address, 2 character for the control and 62 characters for the data.

Table 1 presents the characters types used for the data transfer.

*Table 1*

**Characters types**

| BYTE | b1 | b2 | b3 | b4 | b5 | b6 | b7 | b8 | b9 |
|---|---|---|---|---|---|---|---|---|---|
| 1 | ADDRESS BYTE | | | | | | | | 1 |
| 2 | TYPE | | PARAMETER NUMBER / BYTES NUMBER | | | | | | 0 |
| 3 | PARAMETER VALUE / CHECKING BYTE | | | | | | | | 0 |
| 4 | PARAMETER VALUE / DATA BYTE | | | | | | | | 0 |
| 5 | DATA BYTE | | | | | | | | 0 |
| | | | | | | | | | |
| 65 | DATA BYTE | | | | | | | | 0 |

As shown in the structure of the second character (the control character), used two bits for the transmitted data type and 6 bits for identifying the parameter or for establishing the transmitted data volume.

Table 2 presents the data types depending on the two bits values

*Table 2*

**Data types**

| b1 | b2 | Operation |
|---|---|---|
| 0 | 0 | parameter transfer (8 bits) with the identification number given by the other 6 bits (b3 – b8) |
| 0 | 1 | parameter transfer (16 bits) with the identification number given by the other 6 bits (b3 – b8) |
| 1 | x | block data transfer containing a number of bytes given by the other 6 bits (b3 – b8) |

After the bytes reception, the destination station transmits a confirmation character depending on the received data correctness. The received data correctness checking is performed by the parity bit checking. The parity checking is performed by each character supplementary introduced bit testing so that the total number of symbols „1” to be even or odd.

Fig. 4 show how the master station transmits a new value for a parameter of the slave station.

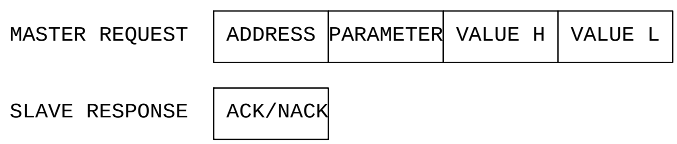


Fig. 4. Change parameter value

Fig. 5 show how the master station require a parameter value of slave station.

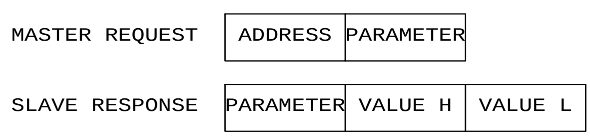


Fig. 5. Request parameter value

Fig. 6 show how the master station request the value of several parameters of the slave station.

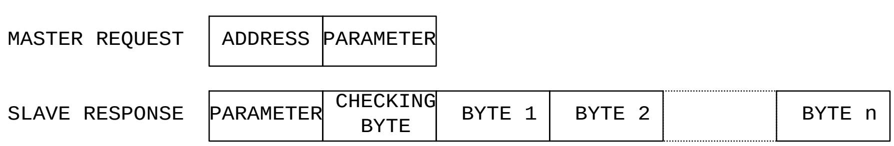


Fig. 6. Request parameters value

The crossed parity checking is performed to the data blocks formed from a series of characters to what it is added a checking character whose symbols are determined by the same rule of the parity determination, taking into account symbols of the same position of all the characters from which the block is formed for each symbol from this character.

Generally, the transmitting means is the twisted pair line in accordance with RS 485 standard [14]. The transmission speed is limited by the cable length, being reduced at the same time with this length increasing. To minimize reflections, the line should be terminated at both ends in its characteristic impedance.

In Fig. 7, it is presented the influence of the cable length on the data transmitted through a cable of 1200 m length (DI – transmitted inlet data, Vy – Vz – differential voltage in the line, RO – received outlet data).

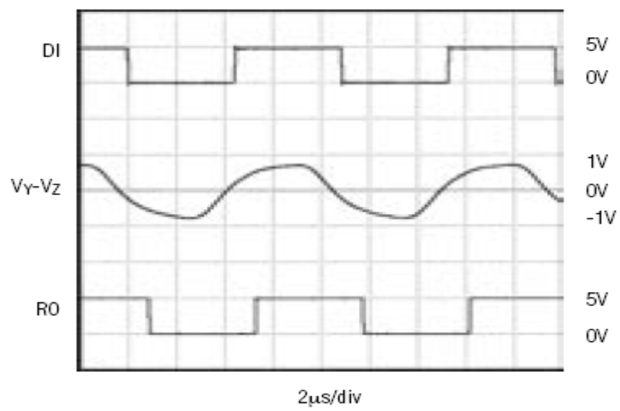


Fig. 7. The influence of the cable length on the data transmitted

## 4. Conclusions

The research presented in this paper was directed toward reducing the amount of data to be processed by the MCUs slave (programmable controllers), thus decreasing the slave station interrogation cycle period. Has resulted a quick update of parameters. This is beneficial to transfer data at greater distances where transfer speed is lower.

The following procedure should be used to exchange data in Multi-processor Communication mode using this protocol:

1. All Slave MCUs are in Multi-processor Communication mode (MPCM).
2. The Master MCU sends an address frame, and all slaves receive and read this frame.

3. Each Slave MCU reads the receive register and determines if it has been selected. If so, it clears the MPCM bit, otherwise it waits for the next address byte and keeps the MPCM setting.
4. The addressed MCU will receive all data frames until a new address frame is received. The other Slave MCUs, which still have the MPCM bit set, will ignore the data frames.
5. When the last data frame is received by the addressed MCU, the addressed MCU sets the MPCM bit and waits for a new address frame from Master. The process then repeats from 2.

As a result of this protocol implementation, are obtained the following advantages:

- reducing of the frames number which must be processed by the slave stations by means of using a bit what establishes if a data byte or an address was transmitted
- minimizing the number of transmitted characters depending on the data type
- decreasing the slave station interrogation cycle period by means of minimizing the transmitted characters number
- a more rapid updating of the parameters values thanks to the interrogation cycle period reducing
- the value of a parameter of 8 / 16 bits can represent a reference value or some input or output signals values
- the received data correctness checking is performed for each byte and for all data bytes
- because each character has synchronization bits (start bit and stop bit), the errors due to the lack of synchronization between the two stations are eliminated

Table 3 presents the slave station interrogation cycle period, percentage of the maximum period, obtained by minimizing the number of transmitted characters depending on the data type.

*Table 3*

**Interrogation cycle period**

| DATA | % |
|---|---|
| parameter transfer (8 bits) | 4,6 |
| parameter transfer (16 bits) | 6,2 |
| data block (10 bytes) | 20 |
| data block (20 bytes) | 35,4 |
| data block (30 bytes) | 50,7 |
| data block (40 bytes) | 66 |
| data block (50 bytes) | 81,5 |
| data block (60 bytes) | 97 |

Using this protocol, we can increase the speed of data exchange four times compared to a standard protocol.

This protocol accepts a maximum synchronization error between the two stations (master and slave) by 12%.

## R E F E R E N C E S


[1] *R. Dobrescu*, Transmiterea datelor (Data transmission), Editura Academiei Române, Bucureşti, 2005

[2] *A. Mateescu, N. Dumitriu, L. Stanciu*, Semnale şi sisteme (Signals and systems)– Aplicații în filtrarea semnalelor, Editura Teora, Bucureşti, 2001

[3] *I. Bănică*, Comunicații de date (Data communication), Litografia UPB, Bucuresti, 2000

[4] *J.G. Proakis*, Digital communication, ed. 6, Mc. Graw Hill, New York, 2003

[5] *R. Freeman*, Practical data communication, John Wiley, 2002

[6] *S. Popescu*, Transmisia datelor (Data transmission), Editura Matrix Rom, Bucureşti, 2003

[7] *I. Constantin*, Principiile transmisiunilor de date (Principles of data transmission), Editura Politehnica Press, Bucureşti, 2002

[8] *S. Cojocaru, C. Radoi, S. Stancescu*, The analysis of CAN and Ethernet in distributed real-time systems, Buletin Stiintific UPB Seria C, ISSN 1454-234x, **Vol. 71**, Iss 4, 2009

[9] Profibus Network Simatic Net – 6gk1970-5ca20-0aa1, 2000

[10] *Compendium Simovert masterdrives*, Automation and drives, motion control system, Siemens AG 2006

[11] *K. Tindell, A. Burns, A.J. Wellings*, Calculating Controller Area Network message response times, Control Engineering Practice, **Vol. 3**, 1995

[12] *F. Halsall*, Data communications, computer network and open system, Addison-Wesley, 1996

[13] *C.N. Thurwachter*, Data and Telecommunications – Systems and applications, Prentice Hall, New Jersey, 2000

[14] *EIA RS-485*, Standard for electrical characteristics of generators and receivers for use in balanced digital multipoints systems, 1993.