RS485 COMMUNICATION SYSTEM

Florin Grofu, ***Constantin Brancusi University of Targu Jiu, ROMANIA***
Constantin Cercel, ***Constantin Brancusi University of Targu Jiu, ROMANIA***

Abstract: *This paper presents possibility of carrying out a remote communication module required to implement a communication network using RS485 standard. An automation application requiring remote communication can have a built-in RS485 communication mode or can be connected to a separate communication module.*

Keywords: *communication network, automation application, RS485 standard*

1. INTRODUCTION

RS485 standard allow implementation of the communication network for a distance over 1200 m using cheap cables by differential transmission.
Automation application which use RS485 standard have built-it in a communication module with the application address and the communication settings parameters.
The main disadvantage of using this system with many identical application is that, if an application may be replaced, the parameters of the new application must be configured.
This paper propose of using a microcontroller based module which ensure the connection of the application to communication network. The module have saved all settings about communication so two identical applications can be changed without any other changes.
RS232 and RS485 are serial communication methods for computers and devices. RS232 is without doubt the best known interface, because this serial interface is implemented on almost all computers available today. But some of the other interfaces are certainly interesting because they can be used in situations where RS232 is not appropriate. RS232 is an interface to connect one DTE, data terminal equipment to one DCE, data communication equipment with a maximum cable length of 50 feet.
One of the main problems with RS232 is the lack of immunity for noise on the signal lines. The transmitter and receiver compare the voltages of the data- and handshake lines with one common zero line. Shifts in the ground level can have disastrous effects. Therefore the trigger level of the RS232 interface is set relatively high at ±3 Volt. Noise is easily picked up and limits both the maximum distance and communication speed. With RS485 on the contrary there is no such thing as a common zero as a signal reference. Several volts difference in the ground level of the RS485 transmitter and receiver does not cause any problems. The RS485 signals are floating and each signal is transmitted over a Sig+ line and a Sig- line. The RS485 receiver compares the voltage difference between both lines, instead of the absolute voltage level on a signal line. This works well and prevents the existence of ground loops, a common source of communication problems. The best results are achieved if the Sig+ and Sig- lines are twisted [1]. The figure 1 below explains why.

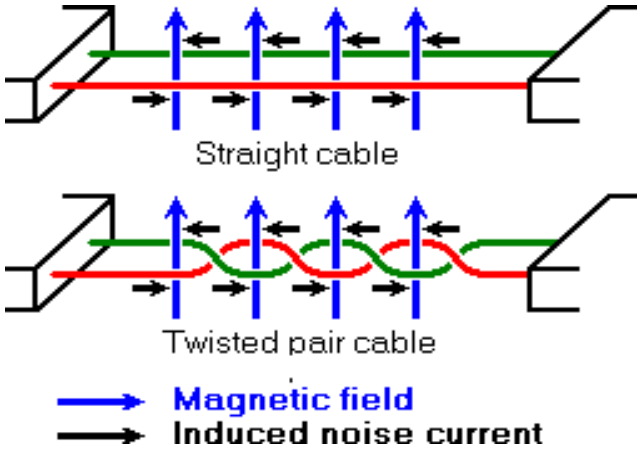

Figure 1.
Noise in straight and twisted pair cables

In the picture above, noise is generated by magnetic fields from the environment. The picture shows the magnetic field lines and the noise current in the RS485 data lines that is the result of that magnetic field. In the straight cable, all noise current is flowing in the same direction, practically generating a looping current just like in an ordinary transformer. When the cable is twisted, we see that in some parts of the signal lines the direction of the noise current is the oposite from the current in other parts of the cable. Because of this, the resulting noise current is many factors lower than with an ordinary straight cable.
Interesting is, that RS232 is the only interface capable of full duplex communication. This is, because on the other interfaces the communication channel is shared by multiple receivers and—in the case of RS485—by multiple senders. RS232 has a separate communication line for transmitting and receiving which—with a well written protocol—allows higher effective data rates at the same bit rate than the other interfaces [1].

2. NETWORK TOPOLOGY WITH RS485

Network topology is probably the reason why RS485 is now the favorite interfaces in data acquisition and control applications. RS485 is the only of the interfaces capable of internetworking multiple transmitters and receivers in the same network.
When using the default RS485 receivers with an input resistance of 12 kΩ it is possible to connect 32 devices to the network.

Currently available high-resistance RS485 inputs allow this number to be expanded to 256. RS485 repeaters are also available which make it possible to increase the number of nodes to several thousands, spanning multiple kilometers. And that with an interface which does not require intelligent network hardware: the implementation on the software side is not much more difficult than with RS232. It is the reason why RS485 is so popular with computers, PLCs, micro controllers and intelligent sensors in scientific and technical applications.

3. RS485 FUNCTIONALITY

Default, all the senders on the RS485 bus are in tri-state with high impedance. In most higher level protocols, one of the nodes is defined as a master which sends queries or commands over the RS485 bus. All other nodes receive these data. Depending of the information in the sent data, zero or more nodes on the line respond to the master.
There are other implementations of RS485 networks where every node can start a data session on its own. This is comparable with the way Ethernet networks function.

4. THE COMMUNICATION MODULE

The structure of the system using the communication module is presented in fig.2.

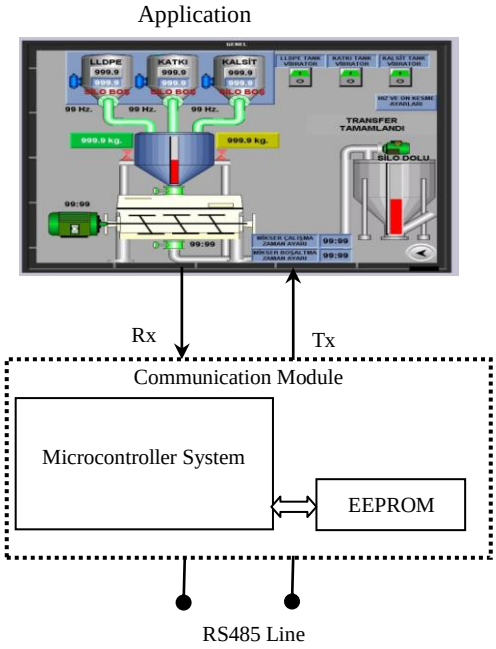

Figure 2. System structure

We observe that it is required to have at least two serial interfaces. As the microcontrollers with two serial interfaces is expensive, was selected for implementation the solution with two microcontrollers with fast communication between them (parallel) and serial communication with the application and with the communication network. The structure of this type of multiprocessor module is presented in fig. 3.

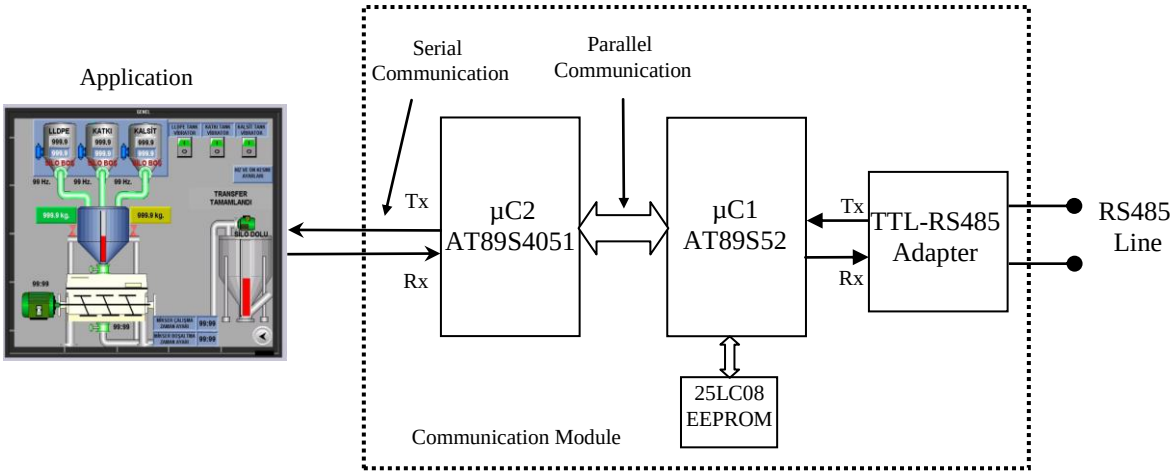

Figure 3. Communication module structure

The µC1 microcontroller is AT89S52 type and ensure the communication with RS485 network, multiprocessor communication type. For this data packets contain one command byte, (with the 9th bit 1 logic) and more data bytes (with the 9th bit 0 logic).

All communication systems by default are set on receive state to accept only commands.

As command is used the address of the module.

Module who recognize it address switch to receive mode and save the data packet. The validity of data packet is checked by calculate the the CRC and compare with corresponding receive value. For data packets under 128 received bytes and 128 sending bytes is used the standard intern memory of the microcontrollers. For large packets can be used other microcontrollers from 8051 family with intern memory up to 1K, more than required for most automation application.

If the data packet is intended to the application, data will be transferred to the second microcontroller µC2 - AT89S4051 type, which will send the data to the application, receiving an answer and sending back to the first microcontroller who wait to release the communication line and send the data to the network. Also, the data can be addressed to the communication module. So the modules can be interrogated or configured remotely.

Can be identified in this mode what communication module is active, can be changed the address and the communication parameters.

In the situation that is required that the information to be processed by all modules, these have two addresses: one common and one can be changed by software.

The module addresses and the communication parameters is saved in the EEPROM memory.

A new module has an initial individual address which is changed when is connected in the network with the desired value. Important is that at one moment in the network two modules must not have the same individual address to avoid the communication collision.

For testing was implemented a system with two communication modules and two identical applications. The structure is presented in figure 4.

The RS232-RS485 converter is an automat converter which in normal mode set the RS485 line as receive line. When a data packet is sent the line switch to transmission line, and get back to receive line after data transmission. In figure 5 is presented the practical implementation of the system with a command application system to control (ON/OFF) of a 6 LED's.

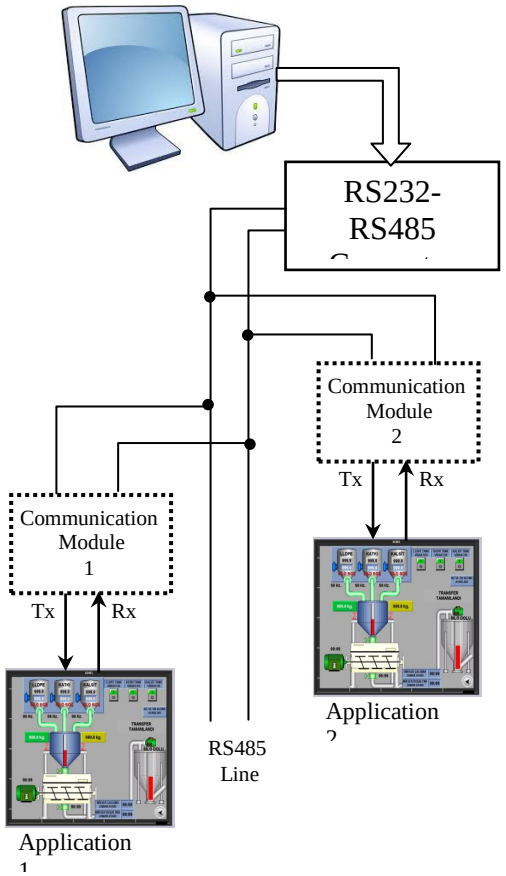

Figure 4. Implemented System structure

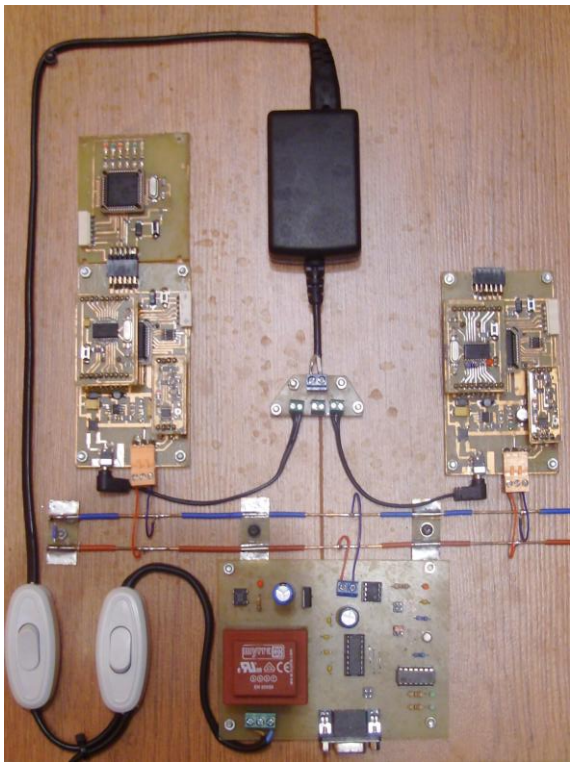

Figure 5. Implemmentation System

To test the hardware system was implemented a software application with an intuitive graphical user interface to control and command the system. The application was developed in LabWindows/CVI and has to sections: on for the communication network parameters management and one for the command of the application. In the management section of the network communication parameters (fig.6a) can be set the follow parameters: communication port, baud rate, data bits, stop bits), also can identify the modules connected in the network, modify the address of a module from network, select the desired application.
In the command section (fig.6b) can test the communication with a selected application having possibility to control 6 LED's independently and view the data packets send and received.

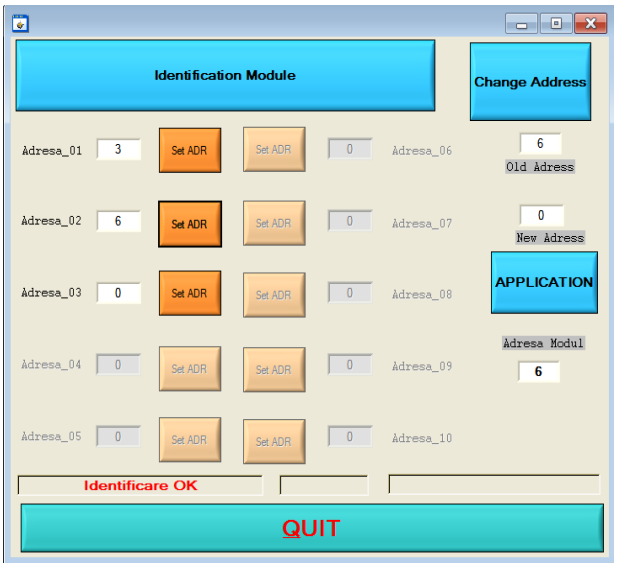

Figure 6a. Software application

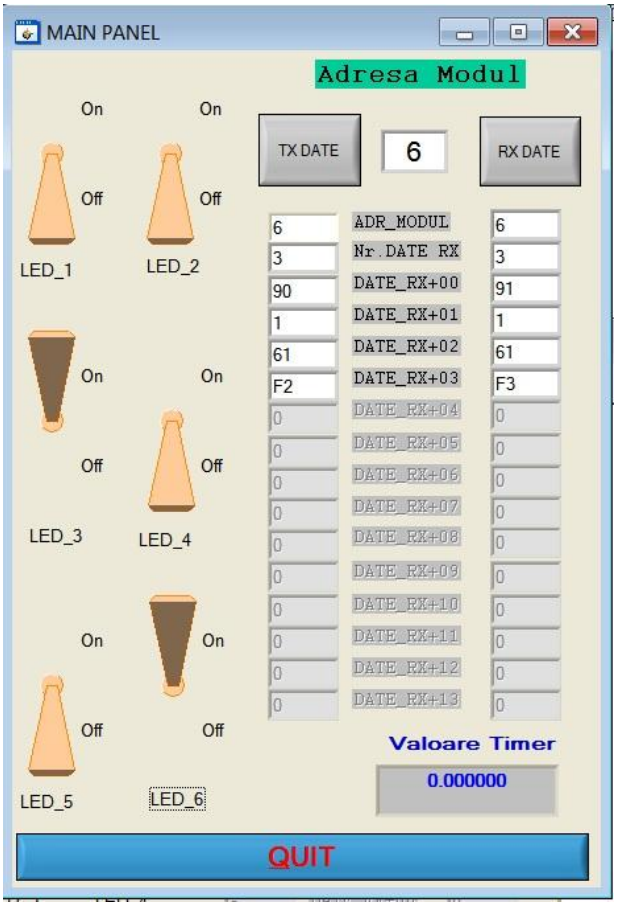

Figure 6b. Software application

CONCLUSIONS

The communication module is quiet simple, but allow both communication with the application and interrogation and remote setting of the existing modules from network.
This structure allow applications to not be individualized as address, and allow interchanging without special software configurations of the address and the communication parameters. Having two microcontrollers can easily make changes about communication with the application, if there are some particularities, and changes of the communication protocol in RS485 network. The only restriction is to not change de parallel communication protocol between microcontrollers.

So, can be developed programs by different teams for the application management and for the network communication management.

REFERENCES

[1]. RS485, specifications and in depth tutorial - Lammert Bies http://www.lammertbies.nl/comm/info/RS-485.html

[2]. Trim-the-fat-off-RS-485-designs -- EE Times. 2000. http://www.eetimes.com/document.asp?doc_id=1224773.

[3]. How Far and How Fast Can You Go with RS-485?--. Maxim IC. 2006. http://www.maximintegrated.com/app-notes/index.mvp/id/3884.

[4]. The art and science of RS-485: Termination--. Circuit Cellar. 1999. http://www.canarysystems.com/nsupport/ArtScienceRS485.pdf.

[5]. RS-422 And RS-485 Applications Ebook-- B&B Electronics. 2010. http://www.bb-elec.com/Learning-Center/All-White-Papers/Serial/RS-422-and-RS-485-Applications-eBook/RS-422-RS-485-eBook-graphics-embedded.pdf.

[6]. Guidelines for Proper Wiring of an RS-485 (TIA/EIA-485-A) Network-- Maxim Integrated. 2001. http://www.maximintegrated.com/app-notes/index.mvp/id/763