MULTI-USER SECURE CONFIGURATION INTERFACE BASED ON A SERIAL PORT SWITCH

[1]İsmail Can, [1]Gurhan Kolcu1 , [2]Radosveta Sokullu
[1]Dept. of Electrical-Electronics Engineering, Ege University,
35100, Bornova, Izmir
BSc Students
ismailcan_net@yahoo.com
gurhan21@yahoo.com
Assistant Professor,
[2]Dept. of Electrical-Electronics Engineering, Ege University,
35100, Bornova, Izmir
rsokullu@bornova.ege.edu.tr

ABSTRACT

The system presented in this article constitutes a serial port switch design and the proprietary broadcast based protocol developed to realize its functions. The system can be used in configuring network devices simultaneously by multiple admins, which both increases the security and provides a possibility for the admins to exchange information among themselves. Everybody can view all the processes present in the system at any time. The information exchange among the users can be peer-to-peer, cross switching and broadcast in structure.

I. INTRODUCTION

At present, most of the programmable electronic devices are designed and produced to be programmed via the serial port (1, 2). Manageable network devices, such as routers and switches are configured via the serial port. In order to configure network devices, the devices have to be connected to a computer and special software is needed to communicate with the operating systems of the network devices. In the traditional approach, the devices had to be connected one at a time and it was impossible for the configuration to be viewed and made by multiple users simultaneously (3, 4). The most important part of the system presented here, the "brain", is the serial port switch. The supporting software provides user interface and control functions. Four of the ports of the switch are dedicated to the user and four are dedicated to the network devices. In order to be able to realize the functions of the system we have developed a proprietary broadcast based protocol for the physical and data link layer. The hardware and software developed allow multiple users and multiple networks devices to:

- connect at the same time,
- simultaneously carry out configuration tasks,
- to simultaneously view and exercise control over the configuration process Except for configuration purposes, users can transfer files full-duplex in peer- topeer, cross switching or broadcast fashion. The rest of the paper is organized as follows: in section 2 the hardware and software structure of the system is described in detail; in section 3 the advantages of the proposed design are summarized and compared to other similar solutions; in the conclusion possible application situations are discussed.

II. The Structure of the System

The system consists of software and a hardware part. The serial port switch constitutes the hardware part. The software, realized in Visual Basic, provides the interface to the microcontrollers and the computers. The main structure of the system is shown in the Figure-1. Here A, B, C and D are the computers that provide physical interfaces for the users. In the middle, there is a serial port switch, to which manageable network devices, marked 1, 2, 3 and 4, are connected. The whole system is made operational by the proprietary protocol we have developed.

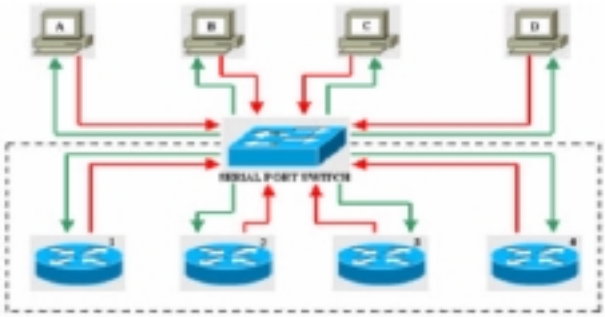

Fig. 1. Structure of the system

Any user can connect to any user or network device via the serial port switch at any desired time. An important point here is that in order to be able to connect to the part of the system separated by the dashed line the user has to enter a valid password. Unless two admins enter the system with a valid password, configuration cannot be carried out. The minimum number of required admins can be determined at setup time. In Figure-1, the part of the system separated by the dashed line can be placed in a special room providing outside access only to the switch ports connected to the four computers. The admins can connect to those outlets to enter the system while Access to the rest of the system is limited for security reasons.

III. Hardware

The hardware includes four microcontrollers communicating together in parallel. Each microcontroller has an external RAM. Four Max 232 are used to convert the computers' signal levels to the TTL levels and twelve buffers IC are used for the switching element. In Figure-2, computers are connected to B1, B2, B3, B4 ports and network devices are connected to C1, C2, C3, and C4 ports. The frames, which come from the B ports, are processed by the corresponding microcontroller. Each B port is continuously listened to by one microcontroller. After the microcontroller receives the destination address, it makes a decision according to this address. It has to communicate with the other microcontrollers to determine the state of the destination port. If the destination port is busy, the microcontroller takes the frame to its external RAM. In this state the controller communicates with the others for a predefined period of time. When the destination is idle, it sends the frame to the destination from its external RAM. Twelve buffer ICs are used per switching element. These ICs are 4-bit 3-state buffers. There are four separate buffers in each IC. The inputs are short-circuited

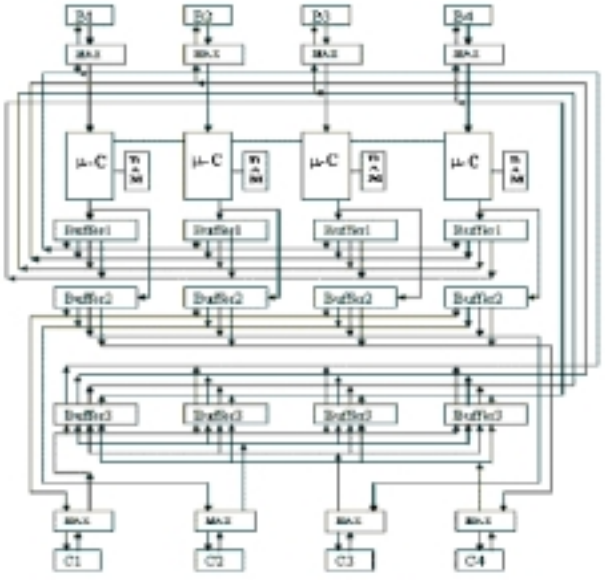

Fig. 2. Structure of the Serial Port Switch

in every IC. Buffers are disabled in the idle state. When a destination address is received, only that destination's buffer is enabled and the others are disabled. A virtual circuit is established from source to destination. When any computer has to communicate with any network device it uses a Buffer2 to send data to that device and a Buffer3 to receive data from that device. Buffer1s are used for communications between computers. The "enable" - "disable" commands for the Buffers are provided by the microcontrollers. The status of each computer and each device can be viewed by means of the computer interface designed. Control frames are used to provide convergence. When a computer has a frame to send or it is ready to receive a frame, it sends a control frame to the serial port switch. The switch broadcasts this frame to every port except the one it has come from. Control frames are also sent when the switch starts to empty the RAM and when that process is finished. The main purpose of the control frames is to provide timely information exchange between the computers themselves that is why they are not sent to the device ports.

IV. *Protocol and Software Description*

After the stage of the hardware design was completed a corresponding protocol for realizing its functions and the supporting software were developed. For the software part VB 6 was used. In this system, every user can view the status of all stations on his own comp uter screen. Everyone knows which computer is communicating with which device or computer. There are three types of frames used in the system (Fig.3). The first type is the data frame. It has a fixed size of 1024 bytes and is used for carrying user data information between two computers when the "Send File" option for sending a text file is activated. The second type is the control frame. Its size is bytes and it is used to broadcast information when there is any change in the status of the system. The third type is the configuration frame with variable size, which is used for communication between a computer and a device. It is used for sending configuration commands to a network device. Another application of this type of frame is sending a character string from one computer to another directly from the keyboard.

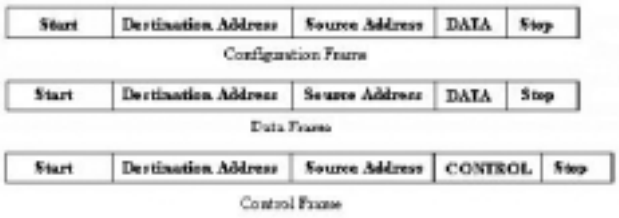

Fig. 3. Types of frames

In the communication among computers, the transfer file is divided into fixed size frames. After an encapsulation

process the data is sent through the serial port. Control frames are sent by computers for request and acknowledgment purposes. The switch can also send control frames to broadcast its status. The control frame has a fixed size
but its control part can carry a different code word depending on the operation. The transmitter and the receiver station can interpret a code word using their VB interface software. Stations learn the status of the whole network by means of control frames. When the switch receives a control frame, it interrupts all transmission until it processes the control frame. It sends the control frame to every port except the one it came from. After this operation the switch returns to its previous transmission. The stations, which receive the control frame, change their status accordingly as shown in Fig. 4. In this figure the example given is based on cross-switching. There is a transfer from Station A to Station C and from Station B to Station D and it can be seen at the same time. A file can be sent using the "Send File" option. The originating path of the text file can be seen in the text box below the status part. This text box is also used to show character strings sent from the keyboard. The transfers among the computers are connection oriented. First a request is sent as a control frame and the station waits for an acknowledgement. After the station receives the acknowledgement it starts to transmit. Other users can see the transmission on their interfaces but they have no access to the data being transmitted. The communication between the computers and the manageable devices is connectionless. This is due to the fact that the operating

Fig. 4. Status Interface

system of the manageable network devices do not support transport layer communication through the serial port. The interfaces presented in Fig. 5 and in Fig.6 are used for configuration. As it was mentioned earlier the process of configuration requires at least two admins to be connected to the system at the same time. That is why before it can start the initiating admin broadcasts a control frame. The screenshot in Fig.6 shows that the validation process is complete and configuration can start. In case no second admin joins in with a valid password the configuration cannot be carried out.

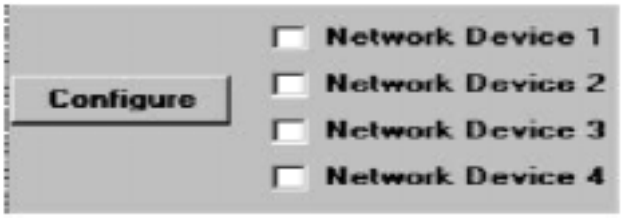

Fig. 5. Configuration Interface

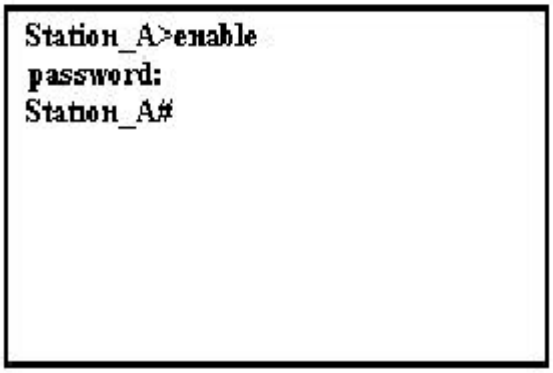

Fig. 6. Screenshot from the interface at the beginning of configuration

V.Comparison

In most switch designs today the basic switch element (SE), the Banyan switch, has 2 input/ output (IO) ports. In order to design a 4 IO system 4 basic elements are needed (2). Such an approach requires taking routing decision twice, one for each stage, which increases the latency of the switch. On the other hand, the failure of an interstage connection reflects on more than one IO connection (Fig.7). As can be seen from Fig. 8 the proposed 4-port basic SE provides a greater number of IO connections and reduces the effect of a failure of an Rx-Tx connection only to a single IO connection. Another factor that has to be taken into account is the control complexity. With only 2 input ports the microcontroller has to check the state of only 2 output ports, while in the case of a 4-port SE each time 4 output ports have to be checked before transmission. The proposed 4-port SE requires only 4 RAM to the Tx side (Fig. 8), while the traditional approach (Fig. 7) would need 8 RAM. The design presented above has good scalability. It can be used as basic element in multi-element design solutions. In Fig. 9 a 16 IO example is given, containing 8 basic SE. Complexity is increased due to the additional stage introduced, but it has good flexibility and is still a much better solution compared to a 16 IO design realized with traditional Banyan elements. As a whole it can be said

that this new design reduces the number of RAMs needed in half, reduces the latency and increases the reliability.

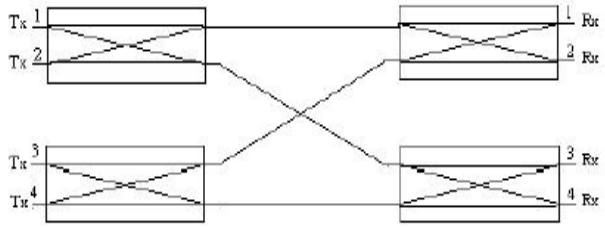

Fig. 7. An example of a typical Banyan switch design

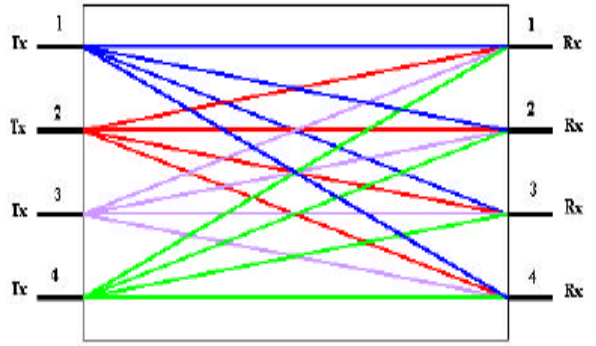

Fig. 8. Proposed 4-port switch element

VI.Conclusion

In this paper we present the hardware and software of a multi-user system that we have designed based on a serial port switch. Its main advantages are the simplicity of the design, collision free communication, easy to use interface combined with the possibility of providing high level of security. We believe it can be used in configuring network devices simultaneously by multiple admins which increases the security and provides information exchange possibilities.

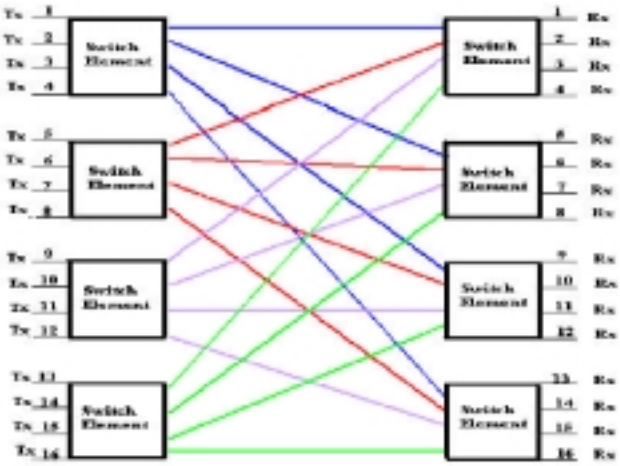

Fig. 9. An example of a 16 IO design

On the other hand it can also be used for educational purposes. An instructor and several students can work together on configuration tasks or the instructor can supervise the work of his students. It can also be used in data communication classes as a tool for visualizing information exchange. The number of ports can be increased.

VII. References

1. Tanenbaum, A.S.: Computer Networks, Fourth Edition, Prentice Hall, ISBN 0-13-0384488-7
2. Stallings, W.: Data and Computer Communications, Sixth Edition, Prentice Hall, ISBN 0-13-086388-2
3. Liu H.I., Wu C.: Virtual private dial-up services over multiprotocol label switching networks, Computer Communications 25 (2001), 884-889
4. Marzo I.L., Vila Pere, Fabrega L.: A distributed simulator for network resource management investigation, Computer Communications (2003), 1-10