JOURNAL OF ENGINEERING AND TECHNOLOGY, **Vol. 7 No. 1** June 2009

http://www.iutoic-dhaka.edu/jet

Development of SS7 and IP Based SIGTRAN data exchanger software operated with designed USB E1 card

M. S. Munir*, K. Ahmed*, Md. Shahriar Rahman*,
Md. Saifur Rahman and M. A. Hoque***

Received 19 May 2008; Accepted after revision 18 November 2008

ABSTRACT

This paper presents new software, which can exchange signaling channel data between SS7 and IP based SIGTRAN network using our developed E1 card. Currently packet data is becoming more significant proportion of traffic in communication network compared to voice. So it has become essential to find ways to consolidate voice, data traffic, platforms and services to reduce the operational, maintenance and initial cost of the network. SIGTRAN network is now considered the most promising media on which to build the new integrated services platform which is basically SS7 in the IP packet. For this reason, development of SS7 and SIGTRAN data exchanger has become a very crucial point. In this paper, we have presented a new software and related hardware to perform this data exchange. The software takes SS7 signaling data from our developed USB E1 card for exchanging with SIGTRAN data of Ethernet port. The unique feature of the E1 card is its USB operation which makes the device plug and play. Another crucial point is that the total development cost of the hardware is minute compared to other hardware in the market. For SS7 and SIGTRAN protocol configuration, ITU-T and IETF standards are followed. This makes the data exchanger software format globally acceptable.

Keywords: *SS7, SIGTRAN, E1, USB, PSTN.*

* *Electrical and Electronic Engineering Department, Islamic University of Technology (IUT), Bangladesh. Email: smunir@iut-dhaka.edu*

** *Electrical and Electronic Engineering Department, Bangladesh University of Engineering and Technology (BUET), Bangladesh. Email: saif672@yahoo.com*

1 INTRODUCTION

Previously in [1] and [2], the Operation Support System and Management of SS7 (Signaling System no. 7) which is also known as CCS No. 7 was proposed. The reliability of such network is presented in Ref. [3] whereas network integrity experiences are presented in Ref. [4]. While integrating such networks some practical complexities arise. Such types of complexities are well described by Causevic S [5]. Taking such problems in account, well defined model of SS7 network is presented again by Pogacnik and Sandri [6] and Jingsha He and Yi-shang Shen [7]. Later on, based on those models, supervision of trunk traffic, commutation and transmission system were presented by Lagorce [8]. In such network, protocol interchanger plays a vital role while connecting this network with IP network. In our paper, we have presented a protocol interchanger software and related hardware to perform this protocol interchanging task.

Protocol can be defined as the rules governing the syntax and synchronization of communication. We have worked with two protocols: SS7 and SIGTRAN.

Signaling System no. 7 (SS7) is a set of telephony signaling protocols [9], which are used to set up the vast majority of the world's public switched telephone network telephone calls. SS7 provides a universal structure for telephony network signaling, messaging, interfacing and network maintenance [10]. SIGTRAN is the name given to an IETF working group that produced specifications [11] for a family of protocols that provide reliable datagram service and user layer adaptations for SS7 and ISDN communications protocols. Our developed software along with our designed hardware is capable of taking signaling channel data from PSTN network and analyzes that according to SS7 and SIGTRAN protocol structure defined by ITU-T [9][10][12]. Performance evaluation of SIGTRAN-based signaling links deployed in Mobile Networks

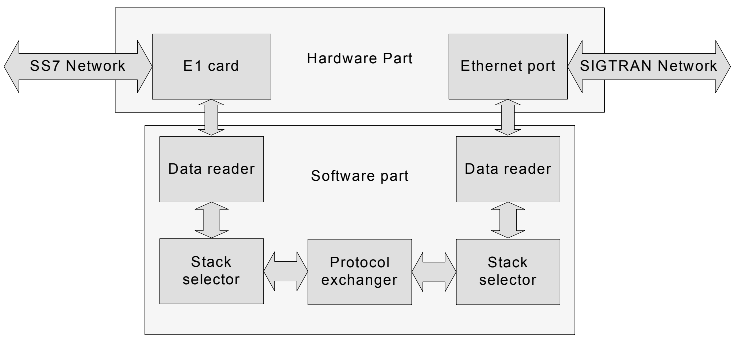

Figure 1: Block diagram of the developed software and hardware.

[13] is very important. This can be done with our software. Also the incorrect data can be detected by the proposed software.

Our developed software takes SS7 protocol data from SS7 network through designed USB E1 card and SIGTRAN data from SIGTRAN network through Ethernet port using “Data reader” window of the software. Then the data is processed according to the protocol stack selected by the “Stack manager” window of our software. Then the data of a selected protocol is exchanged with the other protocol using the “Protocol exchanger” window of our software. The block diagram of the working principle of our developed hardware and software is shown in Fig.1.

2 SOFTWARE ARCHITECTURE

2.1 Protocol Stack

The first part of the development of our software was to develop the protocol stack of SS7 and SIGTRAN protocol.

2.1.1 SS7 Stack

The SS7 protocol stack, shown in Fig.2, borrows partially from the OSI Model of a packetized digital protocol stack. OSI layers 1 to 3 are provided by the Message Transfer Part (MTP) of the SS7 protocol. For circuit related signaling such as the Telephone User Part (TUP) or the ISDN User Part (ISUP), the User Part provides layers 4 to 7. Where as for non-circuit related signaling, the Signaling Connection and Control Part (SCCP) provide layer 4 capabilities to the SCCP user [14].

Network Service Part (NSP) and it provides end-to-end addressing and routing. TUP is a link-by-link signaling system used to connect calls. ISUP provides a circuit-based protocol to establish, maintain and end the connections for calls.

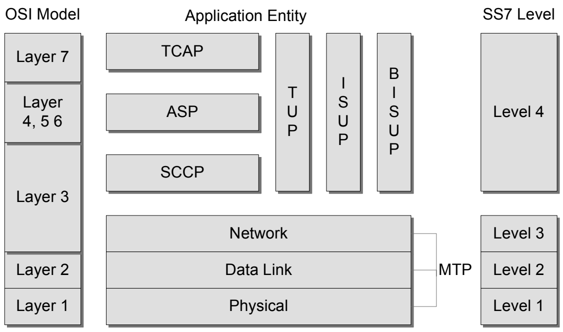

Figure 2: SS7 Protocol Stack.

TCAP is used to create database queries and invoke advanced network functionality mobile services (MAP) [15] etc.

Each SS7 data is transmitted in MTP (Message Transfer Part). Each MTP contains different SS7 layers, which are well defined by ITU-T. The flag bits (F) indicated start or end of MTP in SS7 protocol [16]. In Fig.3, the Forward Indicating Bit (FIB) and Forward Sequence Number (FSN) are used for synchronizing each MTP message with the next MTP message. While Backward Indicating Bit (BIB) and Backward Sequence Number (BSN) are used for synchronizing each MTP message with the previous MTP message. Length Indicator (LI) is used for calculating the length of each MTP message. SIF contains each type of SS7 layer messages e.g. ISUP, TUP, TCAP and SCCP. There is a 'Message Type', the MTP structure for SS7 data [17] used in our software is shown in Fig.3. Using these structures we have developed the SS7 protocol stack of our software.

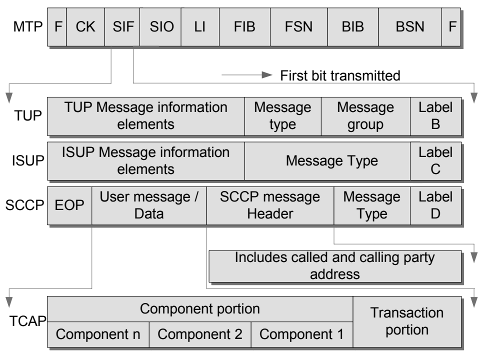

Figure 3: MTP structure of SS7 data.

2.1.2 SIGTRAN Stack

The SIGTRAN protocols, shown in Fig.4, specify the means by which SS7 messages can be reliably transported over IP networks [18]. The architecture identifies two components:

(i) A common transport protocol for the SS7 protocol layer being carried and
(ii) An adaptation module to emulate lower layers of the protocol.

For example, if the native protocol is MTP (Message Transport Layer) Level 3, the SIGTRAN protocols provide the equivalent functionality of MTP Level 2. If the native protocol is ISUP or SCCP, the SIGTRAN protocols provide the same functionality as MTP Levels 2 and 3. If the native protocol is TCAP, the SIGTRAN protocols provide the functionality of SCCP (connectionless classes) and MTP Levels 2 and 3.

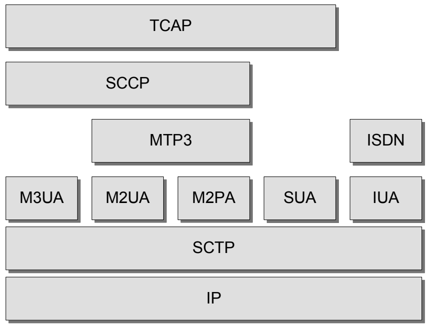

Figure 4: SIGTRAN Protocol Stack.

2.2 SS7 and SIGTRAN Data Interchanger

While performing the protocol interchanging operation, a specific part of a protocol is converted to the equivalent portion of the other protocol. For example, the MTP3 of SS7 protocol is converted to M3UA of the SIGTRAN protocol [19]. The MTP1 layer of SS7 protocol is converted to IP of SIGTRAN protocol and MTP2 layer is converted to SCTP of SIGTRAN protocol. TCAP, SCCP and ISUP layer of SS7 remains same in SIGTRAN but a SIP layer is added in case of SIGTRAN protocol. So the task of the data exchange between SS7 and SIGTRAN protocol is performed according to the structure as shown in Fig.5.

Our software exchanges SS7 protocol data via our designed USB E1 card and SIGTRAN protocol data from Ethernet port. This exchange of data between two different protocols takes place by analyzing the data [20] and maintaining the structure mentioned in Fig.5. The flow chart of the developed software is shown in Fig.6.

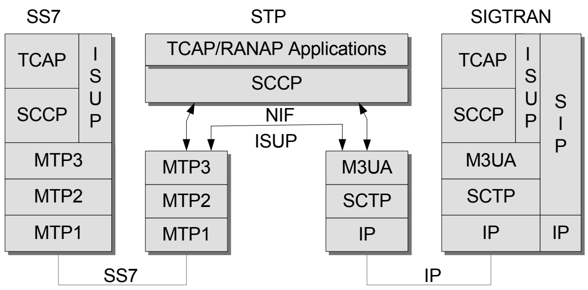

Figure 5: SS7 and SIGTRAN protocol interconnection structure.

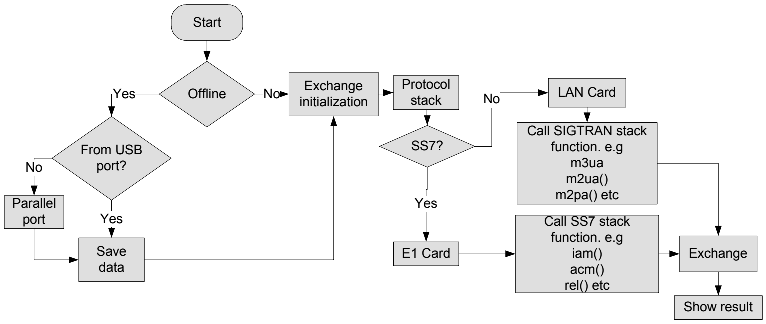

Figure 6: Flow chart of the developed software.

The offline operation of the software involves reading data from USB or parallel port. The data is saved for exchanging to the other protocol data. The online exchanging operation involves selection of SS7 or SIGTRAN protocol data. According to the detected protocol, E1 or LAN card is selected for acquisition of data and related functions are called for converting the data into the other protocol data.

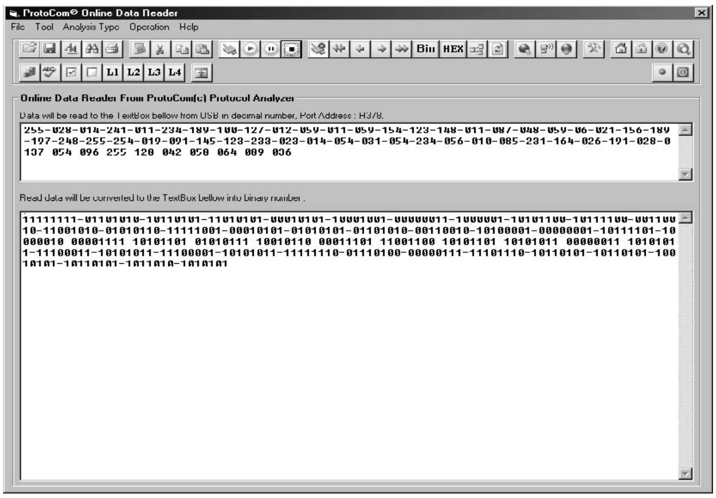

Figure 7: Data reader window of our developed software.

There are three part of the software which performs the data exchange operation. They are as follows:

(i) Data reader

(ii) Stack manager and
(iii) Data exchanger

Part 1: "Data reader" window of our software is used to take SS7 data via E1 card or SIGTRAN data via Ethernet port. At first the user has to specify the source to extract data for exchange. The selected source will be used as the base protocol which will be converted to the other protocol data. The software view for performing this task is shown in Fig.7.

Part 2: "Stack manager" window of our software lets the user to choose which message in the SS7 data is to be exchanged. It is very useful operation when the SS7 data length is very high. By choosing specific message, then user can reduce the number of exchanged messages so that only the desired message is transmitted. The software window of "Stack manager" is shown in Fig.8.

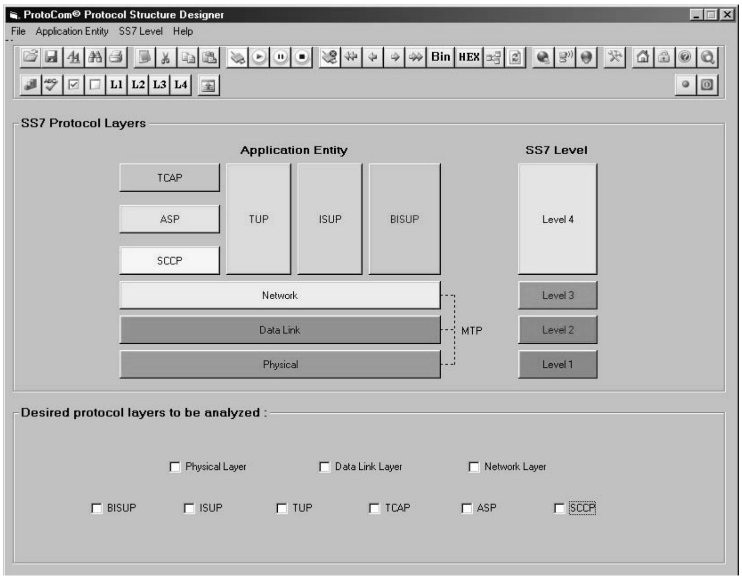

Figure 8: Stack Manager window of our developed software.

Part 3: "Data exchanger" is the last part of our software with which the selected portion of the base protocol data is exchanged to the other protocol data. During this exchange process, the exchanged data can be saved for analysis. Our analyzer mode has the following features which meets almost all the features of current available protocol analyzers:

- Summary view displays MTP2, MTP3 information and SS7 Message types, Called and Calling number, SCCP message type, SSN, INAP information, and more

- Hex/Binary view displays the frame information in HEX and ASCII format
- Exports detailed and summary information to a comma delimited file for subsequent import into a database or spreadsheet
- Supports filtering and search features based on Frame length, FSN, BSN, SSN and so on
- Hex Dump View displays the frame information in HEX and BINARY format.
- View of duration of completed call, OPC, DPC, CIC, Called and Calling Party Numbers, and more [21].

This part of the software has some handy features. For example, the operation of 'ZOOM' is to group the SS7 or SIGTRAN messages that are originated and destined to same point. Both the binary and HEX view of the data is available. There is option for manually editing the binary raw data for remove any part of the SS7 data. The analyzer analyzes the SS7 data into two parts. At one part, it just detects which bit denotes which filed of SS7 message and after that it calculates the value of a specific SS7 message bit to tell the significance of the saved data. The software window performing data exchanging is shown in Fig.9.

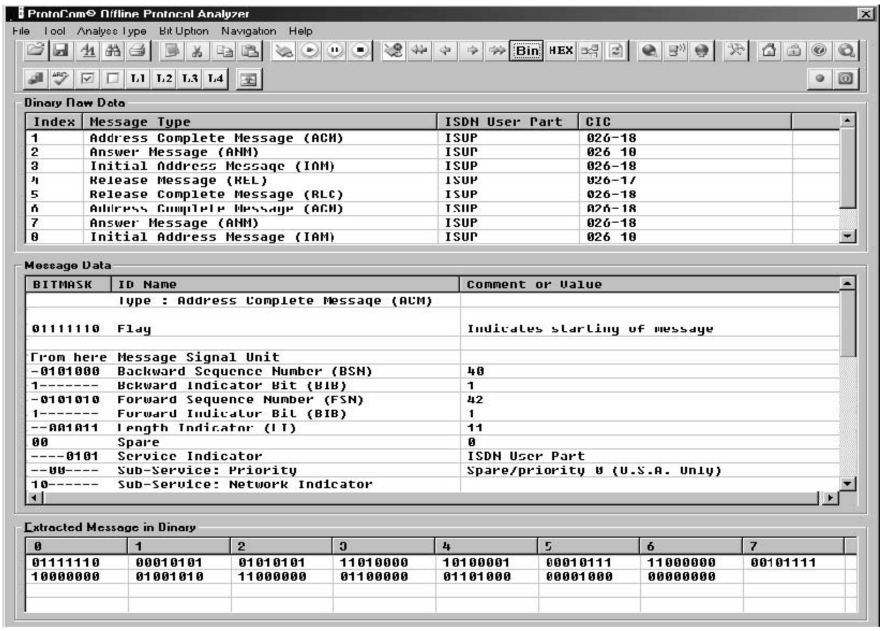

Figure 9: Data Exchanger window of our developed software.

3 HARDWARE ARCHITECTURE

For reading data from SS7 link, we have designed an USB E1 card with which the data of signaling channel of SS7 link can be directly saved by our developed software. Our USB E1 card is an enhanced third-generation hardware that consolidates the essential pieces of industry-standard test equipment into a powerful, PC-Based USB E1 solution. The IC packages used for our design are as follows:

PEF 2256: The PEF 2256 is the latest addition to Infineon's FALC® family of sophisticated E1/T1/J1 framer and Line Interface Unit (LIU) transceivers which supports all standard E1/T1/J1 functions [22].

ATmega16: The ATmega16 is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture, which achieves throughputs approaching 1 MIPS per MHz allowing optimization of power consumption [23].

FT245BM: The FT245BM is the 2nd generation of FTDI's popular Single Chip USB Parallel FIFO bi-directional Data Transfer I.C. The FT245BM provides an easy cost-effective method of transferring data to / from a peripheral and a host P.C. at up to 8 Million bits (1 Megabyte) per second [24].

74HC244: These octal buffers and line drivers are designed specifically to improve both the performance and density of 3-state memory address drivers, clock drivers and bus-oriented receivers and transmitters.

Using these ICs, we have designed the USB E1 card. The block diagram of the designed hardware is shown in Fig.10.

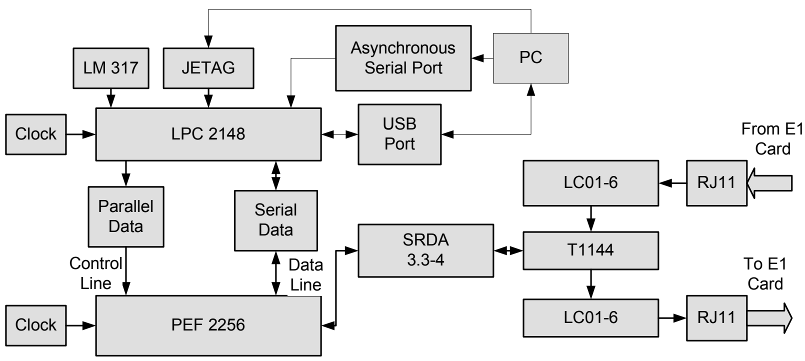

Figure 10: Block diagram of the designed USB E1 card.

In the block diagram as shown in Fig.10, SS7 data enters in the E1 card through RJ11 port and the data passes through a surge protector (LC01-6). For impedance matching, T1144 transformer is used. This data then enters into PEF 2256 IC where the multiplexed data are converted into serial data which is read from PC through USB port as the LPC 2148 microcontroller sends the serial data to the USB port. The operation of the PEF 2256 is controlled by an 8-bit parallel line from LPC 2148. Both PEF 2256 and LPC 2148 has separate clock input. The firmware of the LPC 2148 microcontroller is downloaded using the JETAG port.

For avoiding complex PCB, we have designed the E1 card in two parts. They are,

(i) Transceiver module: It includes
- Transceiver IC
- Synchronizing Clock
- Impedance matching transformer
- Transient voltage Suppressor
- Connector for E1 link

(ii) Controlling module: It includes
- Microcontroller
- USB interface
- Clock
- Serial and Parallel interface
- Power supply unit

The schematics of the controlling and transceiver module are shown in Fig.11 and Fig.12.

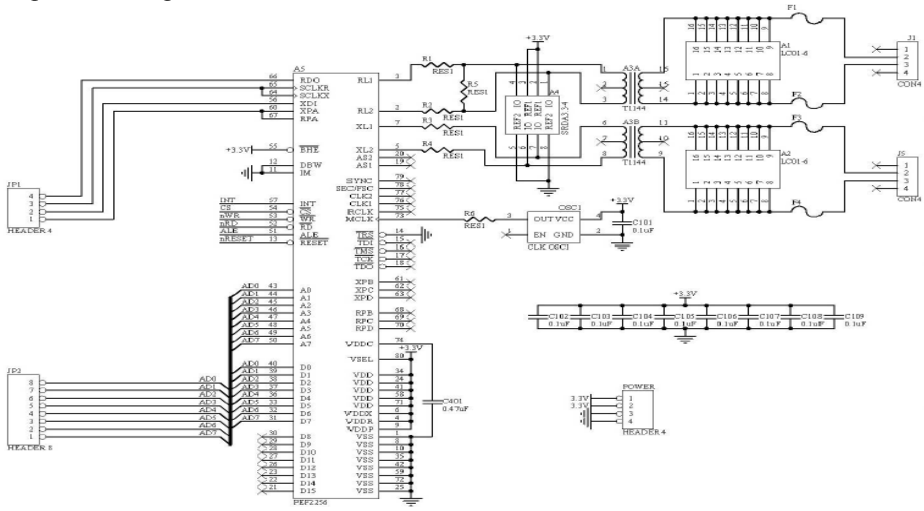

Figure 11: Transceiver module of the E1 card.

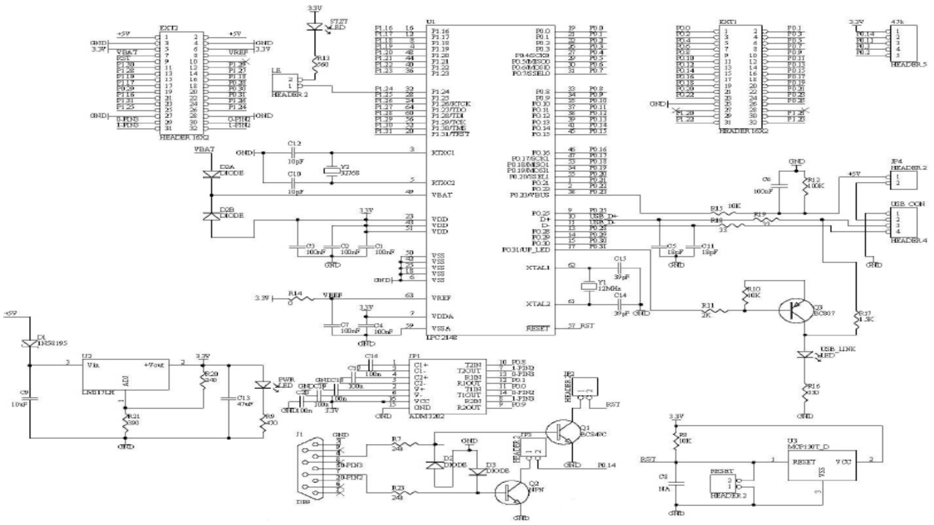

Figure 12: Controlling module of the E1 card.

4 RESULTS

The hardware designing part is completed. The Fig.13 and Fig.14 show the designed USB E1 card.

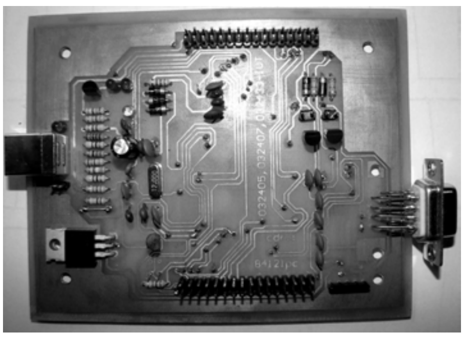

Figure 13: Top layer of the designed PCB of the E1 card

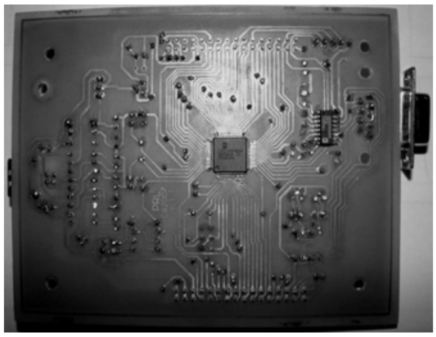

Figure 14: Bottom layer of the designed PCB of the E1 card

The analyzed messages of Fig.7, Fig.8 and Fig.9 show the output of our hardware and software. SS7 and SIGTRAN data can be read [Fig.7] according to the selected structure [Fig.8] and exchanged using our hardware and software

[Fig.9] successfully. One of the major achievements of our work is the tremendous reduction of the cost. The manufacturing cost of the hardware is very minute compared to the market price of similar range hardware. A PCI slot E1 card costs about $500 in the current market whereas our hardware costs only $35.

5 CONCLUSIONS

In this paper, we have presented a new SS7 and SIGTRAN protocol exchanger software and USB E1 card hardware. Protocol exchanger is going to be very necessary to consolidate voice, data traffic, platforms and services to reduce the operational, maintenance and initial cost of the network. This software will enable to expand the PSTN and mobile network using IP network which is very cheap compared to SS7 network. Different value added services will also be easier to provide through the SIGTRAN network to the SS7 network compared to the current available option. So we expect that this software will be a feasible solution of current demand of merging voice and data traffic network.

REFERENCES

[1] Causevic S. (1993) Development of TMN Operation Support System for CCS NO.7 Management. *Informacija Telekomunikacije Automati*, vol. 12, no. 1-3: pp. 245-54.

[2] Schmid P, Ris C and Saeteroy B. (1994) Management of the CCS Hash 7 network: An evolutionary approach. *Tec - Ascom Technical Magazine*, no.2: pp. 8-17.

[3] FCC. (1993) Network Reliability: A Report to the Nation. *Compendium of Technical Papers, Presented by the Federal Communications Commission's Network Reliability Council,* June'93.

[4] Hoang B, Bastien D and Lewin B. (1993) Common Channel Signaling Network Integrity Experiences from the US. *IEEE International Conference on Communications'93. Technical Program*, Conference Record (Cat. No.93CH3261-5) Geneva, Switzerland, 23-26 May'93. vol.2: pp. 644-9.

[5] Causevic S. (1994) Development complexity of Network Management solutions. *Ascom Technical Magazine*, no.2: pp. 18-22.

[6] Pogacnik F and Sandri R. (1994) A Model of Common Channel Signalling Network Management", *Conference on Telecommunications*, ConTEL 94 Zagreb, Croatia, 15-17 June 1994, published in *Informacija Telekomunikacije Automati*, vol. 13, no.1-3: pp. 345-52.

[7] Jingsha He and Yi-shang Shen. (1995) A SS7 Network Management Model. *Telephony*, Vol. 228, Issue 23: pp. 28-31.

[8] Lagorce P. (1996) Supervision of Trunk Traffic. *Commutation & Transmission*, vol. 17, no. 2: pp. 51-60.

[9] ITU-T Recommendations for SS7 standards, Q.700 to Q.716.

[10] Fred Halsal, "Multimedia communication". *McGrow Hill*.

[11] IETF reference document for SIGTRAN standards, "Framework Architecture for Signaling Transport", RFC-2719, http://www.ietf.org/rfc/rfc2719.txt.

[12] Performance Technologies protocol standard document. www.pt.com.

[13] Chukarin, Pershakov A and Samouylov N. (2007) Performance of Sigtran-based Signaling Links Deployed in Mobile Networks. *ConTel.*

[14] Dawis E.P. (2001) Architecture of an SS7 protocol stack on a broadband switch platform using dualistic Petri nets. *Communications, Computers and signal Processing. PACRIM.*

[15] ITU-T reference document Q.701, Page 5.

[16] "Tutorial on Signaling System 7 (SS7)" from Performance Technologies, Page 6. www.pt.com.

[17] ITU-T reference document Q.700 page 14.

[18] SIGTRAN Protocol Suite by Jim Darroch, Page 8.

[19] Reference document on "SS7 over IP signaling transport and SCCP" by International Engineering Consortium, Page 16. www.iec.org.

[20] Signaling System # 7 protocol analyzer software. www.gl.com/ss7.html.

[21] GL communication protocol analyzer software standard www.gl.com/protocol_analysis.html.

[22] Falc2256 product brief of Infenion Technologies, Page 1. www.infenion.com.

[23] ATmega16L Datasheet from ATMEL Corporation. Page 1. www.atmel.com.

[24] FT245BM datasheet from Future technologies Ltd. Page 1. www.ftdichip.com