ISSN (Online) : 2319 – 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization Volume 3, Special Issue 4, April 2014

Two days National Conference – VISHWATECH 2014

On 21st & 22nd February, Organized by

Department of CIVIL, CE, ETC, MECHNICAL, MECHNICAL SAND, IT Engg. Of Vishwabharati Academy's College of engineering, Ahmednagar, Maharastra, India

# HDLC Protocol Implementation Using VHDL

Sarika G. Joshi[1], Vaishali S. Dhongde[2], Prof. Mrs.Mansi Wargantwar[3]

Assistant Professor, Dept. of Electronics & Telecommunication, VACOE, University of Pune,A'nagar, MH, India [1,2]

Assistant Professor, Dept. of Electronics & Telecommunication, MITCOE/ Dr. BAMU, Aurangabad, MH, India[3]

**Abstract**: To successfully transmit data over any network, a protocol is required to manage the flow or space at which the data is transmitted. High-level Data Link Control HDLC is a group of protocols for transmitting synchronous data packets between point-to- point nodes. In HDLC, data is organized into a frame. It resides on Layer 2 of the Open System Interconnect model, i.e. data link layer. HDLC uses zero insertion/deletion process called bit stuffing to ensure that the bit pattern of the delimiter flag does not occur in the fields between flags. The HDLC frame is synchronous and therefore realizes on the physical layer to provide method of clocking and synchronizing the transmission and reception of frames. It is the most commonly used Layer 2 protocol and is suitable for bit oriented packet transmission mode.**[1,2]**The project analyses the methods of HDLC procedure implementation commonly used nowadays, & point out their defects. These HDLC procedures are based on Field Programmable Gates Array (FPGA) & specially illustrate how to generate Frame Check Sequence i.e. Cyclic redundancy Check of HDLC in FPGA.These methods are verified by downloading the HDLC modules designed in VHDL into Xilinx9.1 ISE,on FPGA Spartan3 IM Protoboard model - MXS3FK-IM as a target technology, which shows the feasibility of the methods. The programming modules are simple, easy to modify, & superior in practical application. To make the project user friendly a graphical user interface (GUI) is designed in Visual C ++. Which allows the character byte to enter through keyboard and also shows an output data received through GUI

**Keywords**: VHDL, FPGA, FSM; FCS generation, CRC, LFSR

## I. INTRODUCTION

HDLC protocol is the high-level data link control procedures established by International Organization for Standardization (ISO) [1], which are widely used in digital communication and are the bases of many other data link control protocols. There are different set of protocols run on the second layer i.e. Data Link Layer.eg. Synchronous Data Link Control (SDLC) Point to Point (PPP) Link Access protocol for Data (LAPD).The frame format of all these protocol is comparatively same. The basic protocol amongst all these protocol is High data link Control (HDLC) protocol.

Currently the HDLC protocol is available in built in the TCP/IP protocol suit, to implement the successful communication over the network environment..But, IP Cores are not for free, in order to use which we need to buy the licenses. Software programming of HDLC procedures is flexible, which can be used in many different HDLC applications by modification of it. However, the programs take many resources out of the processor and a lot of time when running, additional to which is the difficulty of predicting the signal delays and synchronization. In some applications of HDLC for multi-channel signals, the percentage of the resources occupied is proportional to the number of the channels. So software programming is often used in single channels and low-speed signal processing system.

There are some applications of data transfer, where there is no need to go to the top most application layer. The communication is implemented over the Data Link Layer itself. Furthermore the data link layer is the only layer from the OSI reference model which can be implemented by hard ware only. Therefore it is possible to implement the separate hardware for such applications. By making some changes as per the frame structure of different protocol, it is possible to design the separate hardware for each protocol application. As HDLC is a basic protocol, very simple to implement, and once it get implemented successfully by a separate hardware, any other protocol structure can be

ISSN (Online) : 2319 – 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization Volume 3, Special Issue 4, April 2014

Two days National Conference – VISHWATECH 2014

On 21st & 22nd February, Organized by

Department of CIVIL, CE, ETC, MECHNICAL, MECHNICAL SAND, IT Engg. Of Vishwabharati Academy's College of engineering, Ahmednagar, Maharastra, India

designed separately by the hardware as per the application needed. As the development and widely usage of FPGA, HDLC procedures can be implemented in FPGA by hardware programming. By adopting hardware processing technology, FPGA devices can be programmed repeatedly. Considering speed and flexibility, FPGA can process multi-channel signals in parallel, and the real-time capacity is predictable and able to be simulated. In small or medium-sized bulk of communication products using FPGAs to replace ASIC devices for HDLC procedures implementation is a proper choice. In this paper, we propose a method of HDLC procedures.

## II. HDLC PROTOCOL STRUCTURE

HDLC protocol is bit-oriented protocol which is much more flexible and more efficient than character-oriented protocol (such as PPP used in internet) [4]. The basic transfer unit of HDLC protocol is frame, and its structure is shown as fig. 1.

| 1 byte | 1-2 bytes | 1 byte | variable | 2 byte | 1 byte |
|---|---|---|---|---|---|
| Flag | Address field | Control field | Information | FCS | Flag |

Fig 1: HDLC frame structure

In the structure, "Flag field" is for the frame synchronization flag, and is the same binary codes "01111110" for both frame header and frame end; as "Address field" is either 1 or 2 byte to hold for the address of receiving station. "Control Field" is 1 byte long, used to send the additional control information about the type of data field. "Data field" is on behalf of user data which should be a multiple of 8 bits; [3]

HDLC is the transparent protocol. i.e. In order to avoid the conflict between the duplication of flag sequence in the information frame as user data, it uses the "bit stuffing."When sending data, if five "1"s emerge continuously in bit stream excluding frame synchronization flags, we should insert a "0" after the fifth "1" manually; when receiving data, if five continuous "1"s emerge and the next is "0", we should remove the following "0" to revert the data. HDLC protocol supports three types of frame structure Information frame; supervisory frame; and unnumbered frame.[1]. The papers discuss the implementation of data sending and receiving along with the supervisory control command in a frame.

### *A. HDLC Frame structure*

The details of all the **fields** of HDLC frame structure is as follows.[4,5,6]

*Flag field:* HDLC uses the 1 byte start and end flag sequence in order to differentiate between start and end of frame. This flag sequence is constant as "01111110". HDLC is the transparent protocol. i.e. In order to avoid the conflict between the duplication of flag sequence in the information frame as user data, it uses the "bit stuffing."When sending data, if five "1"s emerge continuously in bit stream excluding frame synchronization flags, we should insert a "0" after the fifth "1" manually; when receiving data, if five continuous "1"s emerge and the next is "0", we should remove the following "0" to revert the data.

*Address field:* The address field (A) identifies the primary or secondary stations involvement in the frame transmission or reception. Each station on the link has a unique address. In an unbalanced configuration, the A field in both commands and responses refer to the secondary station. In a balanced configuration, the command frame contains the destination station address and the response frame has the sending station's address. The address of the destination can be either 8 bit or 16 bit. Structure of 8 bit address field is as shown in table I.

TABLE I
8 Bit Address Field

| 1 Bit (Unicast / Multicast Address) | 6 bits(Node Address) | 1 Bit End of Address Field( EOAF) |
|---|---|---|

- 0:- Uncast Address
- 1:- Broad cast Address

ISSN (Online) : 2319 – 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization Volume 3, Special Issue 4, April 2014

Two days National Conference – VISHWATECH 2014

On 21st & 22nd February, Organized by

Department of CIVIL, CE, ETC, MECHNICAL, MECHNICAL SAND, IT Engg. Of Vishwabharati Academy's College of engineering, Ahmednagar, Maharastra, India

- Node address: - the Address copied from the last 6 bits of IP address from Network Layer. [17]
- End of address field (EOAF): If the value of this one bit field is 1,it indicates the end of address field. and, if 0 indicates the address is 16 bit wide.

*Control field:* HDLC uses the control field(C) to determine how to control the communications process. This field contains the commands, responses and sequences numbers used to maintain the data flow accountability of the link, defines the functions of the frame and initiates the logic to control the movement of data traffic between sending and receiving stations.
The project used the fixed sequence number for the information frame.
*Frame check sequence:*FCS" is Frame Check Sequence field of 2 bytes, the calculation of the entire frame, excluding the frame synchronization flags used to check the correctness of the received data frame sequence. FCS in HDLC procedures adopts Cyclical Redundancy Check (CRC) which is widely used in digital communication applications.

## III. HDLC MODULES DESIGN

To implement the HDLC protocol the system is basically divided in to two sections, i.e. encoding-and-sending module (transmitter section) and receiving-and-decoding module (receiving section). The detail block diagram of the transceiver system is shown in Fig. 2.

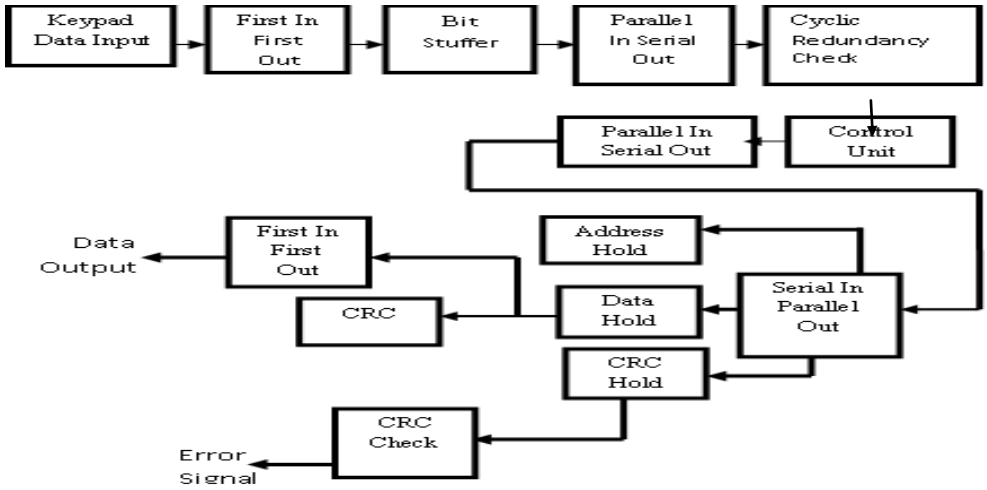


Fig.2: Transceiver block diagram

### A. *Transmitter*

The transmitter block consists of FIFO, two PISOs, CRC block, Temporary register, MUX. Graphical user interface is designed to enter the data and display e successfully received data, and error if any. The data to be transmitted is entered through a keyboard. It is then applied to the FIFO of transmitter through a serial interfacing RS-232. FIFO act as a memory to store the character entered through keyboard. The data which comes first in the FIFO is transmitted as it is to the PISO block paralleled. PISO converts parallel data into serial data. The serial data is then given to the CRC block .CRC is implemented using linear feedback shift register (LFSR) as shown in Fig. 3.The 16 bit CRC is Calculated by using a standard polynomial X16+ $X^{15}$ +$X^{12}$ + 1as per the CCITT standards.[9,10] This 16-bit CRC is stored in temporary register .The sixfields of the frame i.e. Flag, Address, Control, Data, CRC; Flag are given to the MUX. The total Frame size is of 56bits.MUX transmits all the fields 1 by 1 to the PISO.

ISSN (Online) : 2319 – 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization Volume 3, Special Issue 4, April 2014

Two days National Conference – VISHWATECH 2014

On 21st & 22nd February, Organized by

Department of CIVIL, CE, ETC, MECHNICAL, MECHNICAL SAND, IT Engg. Of Vishwabharati Academy's College of engineering, Ahmednagar, Maharastra, India

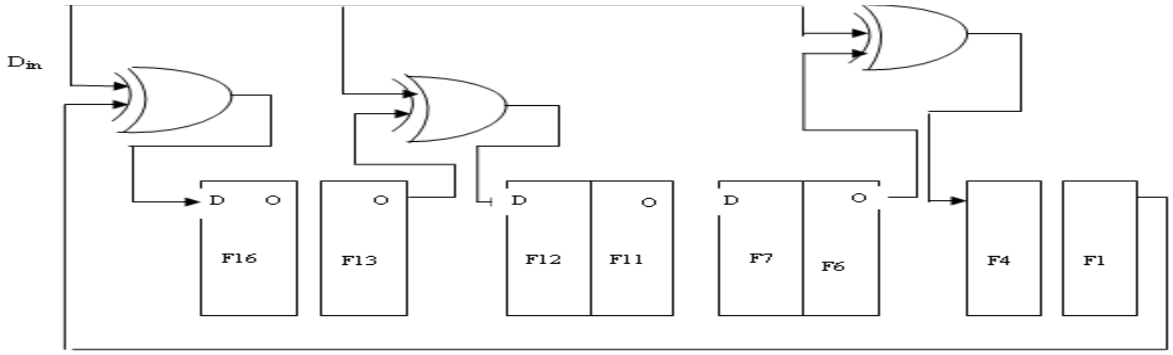


Fig.3: Shift Sequence Using Linear Feedback Shift Register

The final output of transmitter is comes from PISO.The receiver performs a similar calculation for all address, control, and information bits, as well as for all the FCS fields received. It then compares the result. When a match occurs, the frame valid status bit is set to '1'. When the result does not match, the receiver sets the CRC error bit to '1'. The transmitter and the receiver automatically perform this FCS generation, transmission and checking functions.

*B. Receiver.*

The 56 bit frame generated for the 8 bit data from the transmitter section is applied to the serial in parallel out of the receiver.The output of SIPO is a complete frame in parallel form which consists of start flag, address, control, data, and CRC and end flag. The data that is given to the input of transmitter is of 8-bits.At receiver output also we obtain 8-bit data. This data is hold in the data hold circuit If continuous 5 or more than 5 one's are there in the data field same as that of FLAG field then the additional ZERO is inserted at D5 position to separate data from flag, and set the STUF signal as 1.Project is designed for testing the data as letters A-Z, a-z & alphabets 0-9, and the special character such as',",:.;~ etc. The character"~" contains the ASCII value same as the flag bits as '01111110'.Fot this character the data get stuffed out and shows the 'stuff data' output signal at logic one, that show the Bit Stuffing ,which is the unique feature of HDLC Protocol. This block is used to hold the address of the receiving station which is fixed for the given configuration The new CRC is calculated on the received 8 bit data .The calculation and working of receiver CRC is same as the CRC at transmitter side. The calculated CRC at the receiver side is compared with the CRC received by transmitter side. If both are found to be equal the CRC error is 'zero'. If no match is found then the CRC error is set to 'one'.The Synchronous FIFO is a First-In-First-Out memory queue. Its control logic performs all the necessary read and write pointer management, generates status flags, and generates optional handshake signals for interfacing with the user logic. It acts as memory which stores the received 8 character byte. The output of the FIFO is the final output of the receiver.

To make the system user friendly the input is provided through a keyboard. It is then applied to the transmitter section using the universal asynchronous receiver transmitter (UART) at the baud rate of 9600 baud. The entire system flow of the HDLC protocol is shown in Fig. 4

ISSN (Online) : 2319 – 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization Volume 3, Special Issue 4, April 2014

Two days National Conference – VISHWATECH 2014

On 21st & 22nd February, Organized by

Department of CIVIL, CE, ETC, MECHNICAL, MECHNICAL SAND, IT Engg. Of Vishwabharati Academy's College of engineering, Ahmednagar, Maharastra, India

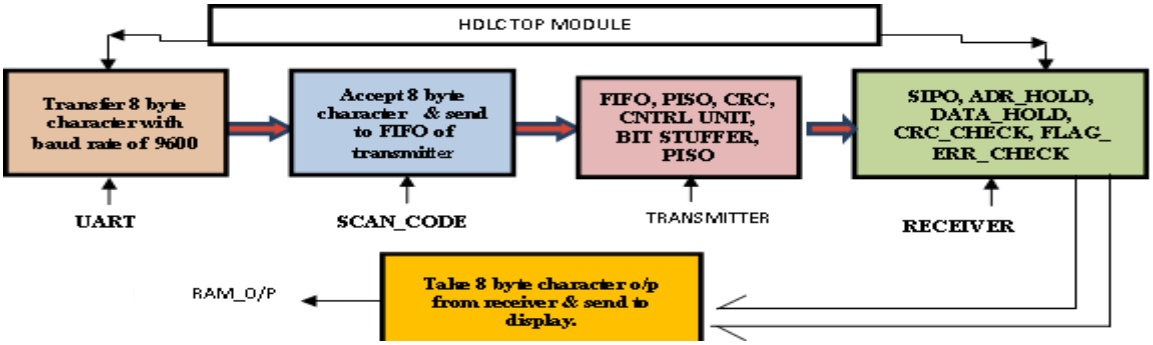


Fig.4: System Development Flow

## GRAPHICAL USER INTERFACE

To make the system user friendly, more efficient and easy to operate a graphical user interface(GUI) is created in microsoft visual studio2008. The 8 byte character input is given through keyboard to the GUI. This Data input is applied to the Scan code module to scan the data one by one, using RS 232 serial interface.Then output of scan code is given to the HDLC Transmitter's FIFO module, for the formation of 56 bit frame structure.

The successfully received data, displayed on the output screen of the GUI, using RS-232 serialinterface.GUI is designed to displaed frame aborted signal and error signal generated if any.The system is tested for the text data, numeric data and also for the flag bit as a data input to check the bit stuffing, which is the unique feature of HDLC protocol.The screen shot input and output display of the GUI for the above said data inputs are shown in Fig.5 and Fig. 6

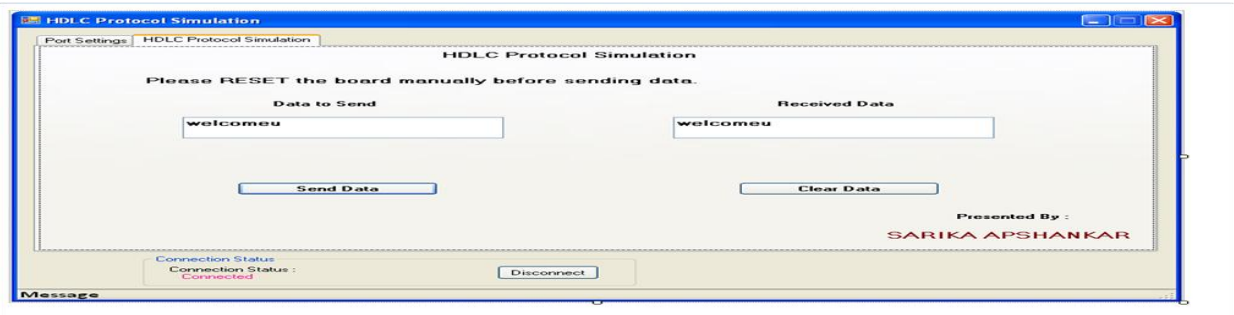


Fig.5: Connection Establishment through GUI

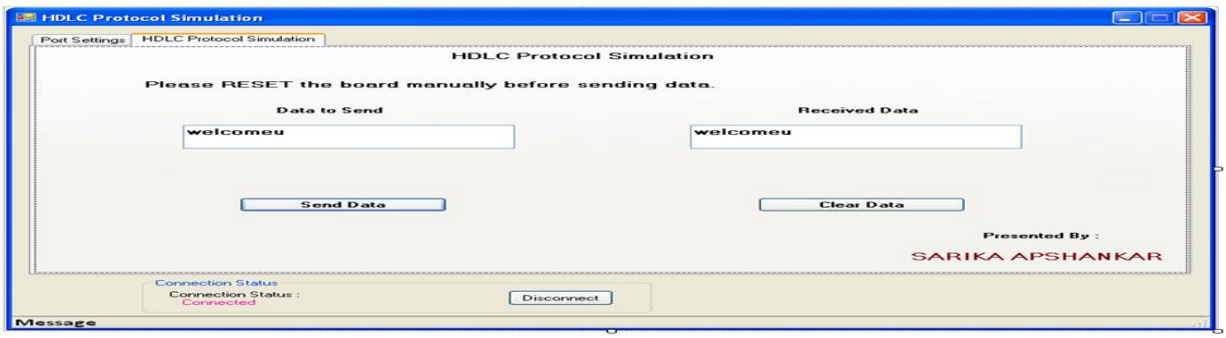


Fig.6: Data transmission and reception through GUI

ISSN (Online) : 2319 – 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

An ISO 3297: 2007 Certified Organization Volume 3, Special Issue 4, April 2014

Two days National Conference – VISHWATECH 2014

On 21st & 22nd February, Organized by

Department of CIVIL, CE, ETC, MECHNICAL, MECHNICAL SAND, IT Engg. Of Vishwabharati Academy's College of engineering, Ahmednagar, Maharastra, India

## IV. RESULT

Single channel HDLC protocol Transceiver has been successfully implemented in the Xilinx Spartan-III series of FPGA. Result of the system developed is observed on Spartan-II and Spartan-III shown in table II.

TABLE II
Result verified on various technologies

| Parameter Value | Sparten-2 XC2s200-5pq208 | Sparten-3 XC3s400-4pq208 | Vertex XCV400-4bg432 |
|---|---|---|---|
| Number of Slice Flip-Flop | 882 | 879 | 882 |
| Number of 4 Input Look up Tables (LUTs) | 2021 | 1989 | 2022 |
| Number of Slice Flip-Flop | 882 | 879 | 882 |
| Number of 4 Input Look up Tables (LUTs) | 2021 | 1989 | 2022 |
| Number of Dual port RAM | 48 | 48 | 48 |
| Number of IOBs | 16 | 18 | 16 |
| Number of IOB Flip-Flops | 11 | 11 | 11 |
| Number of Generic Clocks(GCLK)s | 4 | 4 | 4 |
| Total gate Count | 23757 | 23868 | 23763 |

Performance parameter are also compared with the previous work done are summarized in the table III [11,12,13]

TABLE III
Comparison with Previous work done

| Parameter Value | Single Channel HDLC Transmitter | MT8952B Based HDLC Transceiver | Sparten-3 XC3s400-4pq208 |
|---|---|---|---|
| Number of Slice Flip-Flop | 316 | ---- | 879 |
| Number of 4 Input LUTs | ---- | 4289 | 1989 |
| Number of Dual port RAM | ---- | ---- | 48 |
| Number of IOBs | 68 | 21 | 18 |
| Number of IOB Flip-Flops | ---- | 9 | 11 |
| Number of GCLKs | ---- | 4 | 4 |
| Total Gate Count | ---- | 35037 | 23868 |

## V. CONCLUSION

During the development of the VHDL code, it has been a goal to make it synthesizable. By synthesizing the design, some constrains on areas and timing has been found. This increase linearly with increase in parameters and that are taken as a proof that the VHDL model could be synthesized well. Each module of the transceiver is thoroughly simulated both before and after synthesis and is well optimized. The advantage of implementing this HDLC protocol transceiver in FPGA is that it gives flexibility, upgradability and customization benefits of programmable logic device. From all the above results it is concluded that better system utilization is done on Spartan-III as compared to the Spartan-II, single channel HDLC transmitter and also MT8952B Based HDLC Transceiver.

Using internal storage in FPGA, It solves the problem that Application Specific Integrated Circuits (ASICs) require external storage devices.HDLC is a basic protocol from the data link layer control. As it is designed successfully, now, it can be further upgraded to any other protocol by making the changes in the structure of control field of that particular protocol.

## ACKNOWLEDGMENT

First and foremost, I would like to express my deepest gratitude to **Mrs. M. R.Vargantwar** Guide, Electronics and Communication Engineering Department, for her invaluable support, guidance, motivation and encouragement

ISSN (Online) : 2319 – 8753
ISSN (Print) : 2347 - 6710

International Journal of Innovative Research in Science, Engineering and Technology

*An ISO 3297: 2007 Certified Organization* *Volume 3, Special Issue 4, April 2014*

**Two days National Conference – VISHWATECH 2014**

**On 21st & 22nd February, Organized by**

**Department of CIVIL, CE, ETC, MECHNICAL, MECHNICAL SAND, IT Engg. Of Vishwabharati Academy's College of engineering, Ahmednagar, Maharastra, India**

throughout the period this work was carried out. Her readiness for consultation at all times, her concern and assistance even with practical things have been extremely helpful. It was a great pleasure to work under her guidance.My sincere thanks to all the teaching and non-teaching staff members of my college and those who knowingly and unknowingly have contributed in their own way in completion of this work.

## REFERENCES

[1] Andrew S. Tenenbaum ,"Computer Networks", ,PHI,ISBN 81-203 2175-8, 4th Edition, pp. 16-23, 175-183, 187-190, 225- 228.
[2] Fourauzan B, "Data communication and Computer Networks", Tata McGraw Hill, 4th Edition, pp. 284-294,340-344.
[3] James F. Kurose, Ross, "Computer Networking-a top down approach featuring the internet ", Pearson Education, 3rd Edition, pp. 28-36, 72-78, 445-456.
[4] Comer D. "Computer networks and internet", Pearson Education, 4th Edition, pp.78-80.
[5] Olifer&Olifer,"Computer Networks-principles, technologies & protocols for network design", Wiley, pp. 108-126.
[6] S. HamedJavadi and Ali Peiravi , "Design & implementation of High bit Rate HDLC transceiver Based on a modified MT8952B Controller", INSInet Publication, 2009, pp. 4125-4131.
[7] Jun Wang, Wenhao Zhang, Yuxi Zhang, Wei Wu"Design and Implementation of HDLC Procedures Based on FPGA" School of Electronic and Information Engineering Beihang University (BUAA) Beijing, China, pp. 20-22.
[8] ISO/IEC 13239, "Information technology –Telecommunications and Information exchange between systems – High-level data link control (HDLC) procedures," International Organization for Standardization, July 2002, pp. 10-17.
[9] William Stallings, "Data & Computer Communication", Prentice Hall of India 2000, 7th Edition, pp. 171-180,207-213.
[10] William Stallings, "Wireless Communication and Networks" Prentice hall of India 2000, 2nd Edition, pp.192-201.
[11] Syed ManzoorQasim and Shuja A. Abbasi," FPGA Implementation of a Single-Channel HDLC Layer-2 Protocol Transmitter using VHDL", ICM 2003, Dec. 9-11, Cairo, Egypt, pp. 265-268.
[12] GAO Zhen-Bin LIU Jian-Fei, "FPGA implementation of a multi- channel HDLC protocol transceiver" School of Information Engineering Hebei University of Technology Tianjin, China, pp. 1300-1302.
[13] K. Sakthidasan, mohammedMohammed,"DESIGN& Implementation of HDLC Controller using VHDL", International Journal of Science & Engineering Research, Volume 2, Issue 3,March-2011, ISSN 2229-5518.
[14] Request For Comments: 3572 & RFC 2176.
[15] Michael Gschwind&Valentinaalapura, " VHDL Design Methodology for FPGAs ", Technische University WiCn.
[16] K.C. chang, "Digital Design &Modelling with VHDL and Synthesis", IEEE Computer Society Press 1997, pp. 177-185,242-246
[17] J.F. wakerly, "Digital Design principle & Practice", Prentice Hall New Jersey, 2000, pp. 542-553,850-858
[18] Xilinx Application Note, "Single Channel HDLC core v2.0", Xilinx Inc, 2001.