Design of Enhanced HDLC Protocol using VHDL

Ku. Rupal P. Bende[1] Mr. A. P. Bagade[2] Mrs. S. R. Salwe[3]
[1]M.Tech. Student (V.L.S.I) [2,3]Assistant Professor
[1,2,3]Department of Electronics & Telecommunication Engineering
[1,2,3]B.D.C.E, Wardha, India

Abstract— Data communication over the network is efficiently carried out with the help of protocol. Protocols are specified for each layer of OSI reference model. HDLC is one of the protocols defined for data link layer. It is bit oriented protocol and used to send the data in proper frame. This paper reveals design and simulation of HDLC transceiver by using VHDL which also shows that bit rate can be increased significantly.

Key words: HDLC, VHDL, FCS, FPGA, Verilog HDL, Xilinx

I. INTRODUCTION

High level data link control (HDLC) protocol is developed by the ISO for data link layer. Its current standard is ISO 13239. It is widely used because it supports half duplex, full duplex communication lines, point to point, multi-point networks, and switched or non-switched channels. In HDLC protocol, the control information is always in the same position, which reduces the chance of errors. HDLC is bit orientated protocol in which each bit has significance. The position and value of each bit in the data stream determines its function, thus increasing the efficiency of the protocol. It is suitable for Frame Relay switches, Video conferencing on ISDN, SONET Termination, X.25layer-2 protocol, Cable Modem, Private packet data networks & switches. It is the basis for many data link protocol such as LAPB, LAPD, PPP etc. Some vendors, such as Cisco, implemented this protocol as Cisco HDLC

HDLC is a specification for the Data Link layer which lies between the Physical layer and the Network layer. The Network layer is responsible for passing a packet of data through an internetwork, which can consist of many individual local area networks and even wide area links. The Data Link layer is responsible for passing the data between two nodes on the same network. HDLC, which is specified for data link layer, takes packets from the Network layer and attaches and address, control, and data integrity information to them. Once formatted, the packets are sent "down the wire using the Physical layer. Thus in HDLC protocol, the data is arranged in frame. Frames are responsible for transporting the data to the next point.

A. HDLC Frame Structure

In this protocol, data is encapsulated in frame which includes address field, control field and FCS. The frame format is shown below:

Flag (8-bit)	Address (8 or 16 bit)	Control (8 bit)	Data (variable in size usually multiple of 8 bit)	FCS (8 or 16 bit)	Flag (8 bit)

Fig. 1: Structure of HDLC frame

1) Flag

It is an 8-bit field. Every frame starts and ends with flag sequence 01111110. At any time, the transmission of frame can be aborted by sending the abort flag which is bit sequence of 01111111. It is used to distinguish two successive frames.

2) Address

It is an 8 Bit or 16 bit field. It consists of the address of the receiver.

3) Control field

It is an 8 bit field. It controls 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 traffic between sending and receiving stations.

4) Data or Information

It may contain any number of bits usually multiple of 8. It consists of data i.e. number of bits; the sender actually wants to transmit to the receiver.

5) FCS

This field contains a 16-bit, or 32-bit cyclic redundancy check bits. It is used for error detection. The fields are transmitted from left to right, least significant bit first.

HDLC uses a technique called bit-stuffing to differentiate bit sequence from a flag field. Every time the user wants to send a bit sequence having more than 5 consecutive 1s, it inserts (stuffs) one redundant 0after the fifth 1.This extra zero is inserted regardless of whether the sixth bit is another one or not. Its presence informs the receiver that the current sequence is not a flag. Once the receiver has seen the stuffed 0, it is dropped from the data and the original stream is restored.

This paper aims design and simulation of HDLC transceiver using Xilinx suit of software at maximum bit rate. Designing this protocol using VHDL gives higher speed, flexibility and more efficiency.

II. LITERATURE REVIEW

HDLC is one of protocol specified for data link layer. This protocol is commonly used in networking. It is the basis for many other important data link control protocols, such as LAPB, LAPD and PPP. Many chips are designed for HDLC controllers and they are programmed using high level of programming language. This controller chips can also be designed using hardware description language. The complete review in [1] shows that the number of HDLC controller is designed, using VHDL or Verilog language. The design of HDLC controller involves the design of transmitter and receiver which are used to transmit and receive the HDLC frames. These HDLC controllers are designed at different bit rate. The maximum bit rate achieved is 155.5 Mbps [7]. The controllers are simulated and implemented by using Xilinx suite of software of different versions on the target FPGA device. Implementing the protocol in FPGA gives the flexibility upgradability and customization benefits of programmable logic. These

controllers are also designed according to the need of particular application [8].

III. IMPLEMENTED WORK

Improvement in data communication over a network is need of an hour and till now, maximum bit rate achieved is 155.5Mbps. By keeping this in mind and to make the data communication speedy, here we are implementing the design to achieve more bit rate. Our implementation work is divided into 4 sections- A. Design of module B. Operation C. Software tools used D. Simulation result.

A. Design of System

Hence the design includes design of transmitter and receiver and they are mapped together to form single system for proper communication. The basic operation of HDLC transceiver is shown in fig 2.

Initially, the design of each module is described and later the working is explained. Thus The main blocks of transmitter are:

1) 8-bit register: It is responsible for capturing the data on the rising edge of clock signal.
2) Address insertion: This block contains the address of the destination, which is of 8 bits. This can be any arbitrary address, or the broadcast or "All-Stations" Address, which are all one.
3) FCS generation: It is 16 bit sequence added in message bits for error detection purpose. It is basically generated by using a shift register and X-OR gates or linear feedback shift register (LFSR)
4) Zero insertion: When the bit sequence having more than 5 consecutive 1s is detected, this block inserts (stuffs) one redundant 0.
5) Flag generation block: A specific bit pattern 01111110 is added in the beginning of the frame. Thus total 48 bits are transmitted by the transmitter.

Now following are the main blocks of receiver:

- Flag detection block- Flag bits are checked.
- Zero bit detection- All the extra zeros are detected and removed.
- FCS block- FCS bits are detected and checked for error free transmission.
- Address detector- Address of receiver is detected and matched with the destination.
- Data – finally information /data the transmitter transmit is received at the receiver.

The function of FCS bits is to detect error. All designs are coded in VHDL. Synthesis and simulated is done by using Xilinx tool of ISE 13.2.

B. Operation

At the transmitter side, when the system is initialized, the data and address is loaded on rising edge of the clock signal and reset signal is high.

After the loading of whole data, the FCS bits are included the frame data. FCS i.e. Frame Check Sequence is sequence of bits that are appended in data for error detection. It is calculated by using 16 bit Cyclic Redundancy Check (CRC) method. This CRC code is generated through a process of long division. This can be implemented using linear feedback shift register (LFSR). Then extra zeros are added as per bit insertion process. This operation involves inserting a zero after any sequence of 5 consecutive ones in the transmitted data stream. Flag bits are also added which is 8 bit in length. These whole data of frame is sent bit by bit from out data signal of transmitter.

Now the out data signal of transmitter is mapped with the input data of receiver. Thus total data of frame is received at input data of receiver. Busy signal of transmitter is mapped with rx data signal of receiver. Now the receiver receives the data when rx data signal is high. After the reception of data, the flag sequence and control bits are removed. After all the extra zeros are detected and removed from the stream of bits. FCS bits are detected. The FCS bits are also calculated from received bits. Now the calculated FCS at the receiver side is compared with the FCS received from transmitter side. If both are found to be equal, then there is no error. FCS bits are matched if they are not equal. Hence this designed protocol provides error free transmission of data.

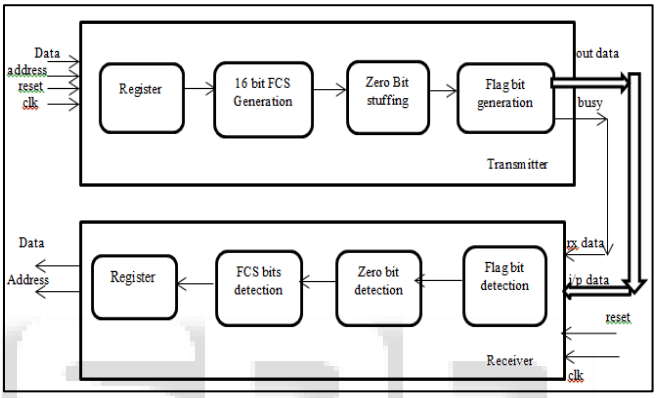

Fig. 2: Block Diagram of HDLC transceiver

C. Software Tools used

The system is coded in VHDL language. VHDL is a hardware description language used in electronic design automation to describe digital and mixed-signal systems such as field-programmable gate arrays and integrated circuits. VHDL can also be used as a general purpose parallel programming language. It is simple language to describe any system and has large capabilities of designing system.

Now Xilinx ISE is a best software tool for synthesis and analysis of VHDL or Verilog designs, enabling the developer to synthesize ("compile") their designs, perform timing analysis, examine RTL diagrams, and configure the target device with the programmer.

Hence for complete analysis of the system, Xilinx ISE of version 13.2 is used. Here the design is synthesized with Spartan 3 as target FPGA family. This Spartan 3 family has its various built-in features that solve designer's challenge throughout the system. This family provides a broad capability for chip-to-chip communications through programmable support for the latest I/O standards, digital Delay-Locked Loops (DLLs) for clock signal synchronization on the FPGA. The Spartan-3 family builds on the success of the earlier Spartan-IIE family by increasing the amount of logic resources, the capacity of internal RAM, the total number of I/Os, and the overall level of performance as well as by improving clock management functions.

D. Simulation Result

Initially the input data 11101111 and address 10010001 is loaded on rising edge of clock signal and reset signal is high. When reset becomes zero, the whole data including input data, address, FCS bits and flag sequence starts transmitting. The out data line indicates the flow of bits that are transmitting. The busy line indicates whether the transmission is completed or not. When busy line is 1, the transmission is going on and when it is 0, the transmission is completed.

The out data signal of transmitter is mapped with the input data signal of receiver.

Thus at the receiver side, the whole data is received at i/p data line of receiver. When rx data signal is high, the receiver can receive the data. Now Simulation result can be analysed by considering two cases.

1) Consider the case when data is transmitted without any noise or error. Due to this, transmitted and received bits are equal.

At the transmitter side, the bits of frame are transmitted in such a way:

01111110 10010001 00101001 11011111 10110100001011001 01111110

At the receiver, received bits are:

01111110 10010001 00101001 11011111 10110100001011001 01111110

2) Consider a situation when some redundant bits i.e. noise is added in the transmitted bits. Due to this, transmitted and received bits are not equal. In such case, Data received at receiver is:

01111110 10 1 010001 001 1 01001 11011 1 111 1011010 1 0001011001 1 01111110

Bits which are bold shows noise bits. After reception of bits at receiver, all bits are detected. The FCS bits are calculated by using XOR and shift register. Then the calculated FCS is compared with the FCS received from transmitter side. Due to noise, they do not get matched. Hence there is error correction logic for matching the bits. After that, correct data and address is received. Thus in this way data 11011111 and address 10010001 is received at the receiver.

The simulation result, after considering both cases, of whole system is shown in figure 3. The design is coded in VHDL. Synthesis and simulation is done by using Xilinx ISE 13.2 tool.

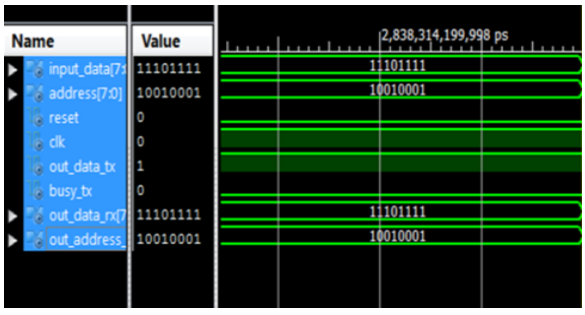

Fig. 3 Simulation result of transceiver

After synthesis of design, the detailed report of delay and device utilization is obtained. The design is synthesized with Spartan 3 FPGA as target device. Table 1 summarizes the device utilization for Spartan 3 FPGA family with device XC3S200 FPGA with a speed grade of 5. As per our aim, higher bit rate can be obtained by reducing the delay of system. Thus less delay can be obtained by designing the system properly and by selecting the suitable family of FPGA From delay; higher bit rate can be computed. From synthesis report, total delay value is obtained. By using this value, bit rate is computed which is 180.66 Mbps.

Device utilization summary			
Logic utilization	used	available	utilization
Number of slices	158	1920	8%
Number of sliced flipflop	139	3840	3%
Number of 4 i/ps LUT	257	3840	6%
Number of bonded IOBs	36	141	25%
Number of GCLKs	1	8	12%

Table 1: Device Utilization Summary

Device utilization shows how many components of FPGA family is used for designing the protocol. From table, it seems that the designed system consumes less chip area.

IV. RESULT

The HDLC transceiver is designed using VHDL code. The codes are simulated in using Xilinx ISE 13.2 tool by selecting Spartan 3 FPGA family of device XC3S200 with a speed grade of -5. The transceiver is found to be running at 180.66Mbps.

Operating frequency or Bit rate	
Previous Design	Presented design
155.5 Mbps	180.6Mbps

Table 2: Comparison of Bit Rate

Above Table shows that the presented design achieves the bit rate of 180.6 Mbps which is more than the bit rate achieved in previous work. Thus the aim of increasing more bit rate is achieved.

V. CONCLUSION

Thus HDLC transceiver is designed and successfully simulated using Xilinx ISE 13.2 tool. The aim of paper to design the transceiver at the maximum bit rate is achieved. It is capable of detecting and correcting the error at the receiving end and hence provides error free transmission of information.

By designing the protocol using VHDL, it can be modified easily and speed can also be increased. Thus the controller can be made simple, upgradeable, flexible and efficient.

This protocol can be used in X.25, Frame relay and ISDN network. It is suitable for the application where maximum throughput matters, also suitable for point to point application like leased line.

VI. FUTURE SCOPE

This concept can be used for designing Multi-channel HDLC controller with few modifications in the VHDL coding. The application of HDLC can be extended by certain modification. It can be used for wireless data transmission. Hence it can be used in wireless LAN and other application.

ACKNOWLEDGEMENT

We would like to thanks Mr. P.R. Indurkar, Associate Professor, Electronics and Telecommunication department,

B.D.C..E for his valuable suggestions. We would thanks to our college for providing valuable facilities which helps us in our research work. We also express thanks to our parents, friends and colleagues.

REFERENCES

[1] Rupal P. Bende,Prof.A.P.Bagade,Prof.S.R.Salwe," Review on Design of HDLC Protocol using HDL", International Journal of Innovative Research in Electrical, Electronics, Instrumentation and control engineering, Vol. 4, Issue 2, February 2016.
[2] Syed Manzoor Qasim and Shuja A. Abbasi ," FPGA Implementationof a Single-Channel HDLC Layer-2 Protocol Transmitter using VHDL", International Conference on Electrical, Electronics and System Engineering, 2003.
[3] S. Hamed Javadi and Ali Peiravi,"Design and Implementation of a High Bit Rate HDLC Transceiver Based on a Modified MT8952 Controller", Australian Journal of Basic and Applied Sciences,2009.
[4] K.Sakthidasan, Mohammed Mahommed,"Design of HDLC Controller Using VHDL", International Journal of Scientific & Engineering Research Volume 2, Issue 3, March-2011.
[5] Harpreet Singh, Navneet Kaur, Vinay Chopra and Dr. Amardeep Singh," Optimization of multi - channel HDLC protocol transceiver using Verilog", International Journal of Computer Science Issues, Vol. 9, Issue 2, No 2, March 2012.
[6] Armaan Hasan Nagpurwala,Sundaresan C, Chaitanya CVS, "Implementation of HDLC Controller Design using Verilog HDL", International Conference on Electrical, Electronics and System Engineering,2013.
[7] Gaurav Chandil, Priyanka Mishra,"Design and Implementation of HDLC Controller by Using Crc-16", International Journal of Modern Engineering Research (IJMER) Jan.-Feb. 2013.
[8] Jai Karan Singh1, Mukesh Tiwari, Mohd Firoz Warsi,"Implementation of HDLC Protocol Based DDR-RAM Radar Processing System", International Journal of VLSI and Embedded Systems-IJVES -May - June 2013.
[9] Shubham Fadnavis,"An HVD Based Error Detection and Correction Code in HDLC Protocol Used for Communication", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 2, Issue 6, June 2013.
[10] Shashank Rampelly, Santhosh Rao Seri, Gnaneshwara Chary, Krishanm Raju, "Data Communication Using HDLC Protocol", International Journal of Innovative Research in Electrical, Electronics, Instrumentation and control engineering ,Vol. 2, Issue 5, May 2014.
[11] Gao, Zhen-bin and Jian-Fei Liu, "FPGA implementation of a multi-channel HDLC protocol transceiver", In Proceedings of the 2005 International Conference on Communications, Circuits and Systems, 2: 1300-1302,2005.
[12] Lu, Y., Z. Wang, L. Qiao and B. Huanq, "Design and implementation of multi-channel high speed HDLC data processor," IEEE International Conference on Communications, Circuits and Systems, and West Sino Expositions, 2: 1471-1475,2002.
[13] Jun Wang; Wenhao Zhang; Yuxi Zhang; Wei Wu; Weiguang Chang; Sch. of Electron. & Inf. Eng., Beihang Univ. (BUAA), Beijing, China "Design and implementation of HDLC procedure based on FPGA" , Anti-ounter feiting, Security, and Identification in Communication, ASID, 3rd International Conference, 20-22 Aug.2009.
[14] Wang Lie, Yi Mingvol, "Design of HDLC Controller based on XILINX FPGA", International Conference on Electrical, Electronics and System Engineering IEEE, 2011.

