IMPLEMENTATION OF HDLC PROTOCOL USING FPGA

Ms.Kshitija S.Patil[1], Prof.G.D.Salunke[2], Mrs.Bhavana L. Mahajan[3], Dr.A.S.Hiwale[4]

[1.] *Student (ME),E&TC,GSMCOE,Pune,Maharastra,India,**mlbhavana@gmail.com***
[2.] *Principal,E&TC,GSMCOE,Pune,Maharastra,Indi,**ashiwale@gmail.com***
[3.] *.Student (ME),E&TC,GSMCOE,Pune,Maharastra,India,**kspatil87@gmail.com***
[4.] *Asst. Prof,E&TC,GSMCOE,Pune,Maharastra,India,**geetasalunke@g.mail.com***

Abstract

To successfilly transmit data over any network, a protocol is required to manage the flow or pace at which the data is transmitted. This protocol is defined in Layer 2 of OS1 (Open Systems Interconnection) model. High-level Data Link Control (HDLC) is the most commonly used Layer 2 protocol and is suitable for bit oriented packet transmission mode. This paper discusses thc VHDL modeling of single-channel HDLC Layer 2 protocol Transmitter and its implementation using Xilinx Virtex FPGA as the target technology. The HDLC Transmitter is used to transmit the HDLC frame structure. Implementing the single-channel HDLC protocol Transmitter in FPGA gives you the flexibility upgradability and customization benefits of programmable logic.

Index Terms: *HDLC, OSI, VHDL, Data link layer.*

---***---

1. INTRODUCTION

To successfully transmit data over any network, a protocol is required to manage the flow or pace at which the data is transmitted. This protocol is defined in Layer 2 of **OS1** (Open Systems Interconnection) model. High-level Data Link Control (HDLC) is the most commonly used Layer **2** protocol and is suitable for bit oriented packet transmission mode. This paper discusses the VHDL modeling of single-channel HDLC Layer 2 protocol Transmitter and its implementation using Xilinx Virtex FPGA as the target technology. The HDLC Transmitter is used to transmit the HDLC frame structure. Implementing the single-channel HDLC protocol Transmitter in FPGA gives you the flexibility upgradeability and customization benefits of programmable logic.

HDLC is a multi-layered protocol that is useful across the board. The basic HDLC protocol can be implemented easily and runs well on a microcontroller (except for bit stuffing, which can be worked around). The full-blown HDLC standard is quite a sophisticated tool for controlling a data link.

2. LITERATURE SURVEY

Networks are defined by the International Organization for Standardization (ISO) using the Open Systems Interconnection (OSI) Reference Model. This model consists of seven layers which detail different aspects of a network. The seven layers are, from lowest to highest level of abstraction, Physical layer, Data Link layer, Network layer, Transport layer, Session layer, Presentation layer, and Application layer. HDLC is a specification for the Data Link layer and 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, of which HDLC is a part of, is responsible for passing the data between two nodes on the same network. HDLC 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. The Physical layer specifies how the bits are transmitted. This can include, but is not limited to, the data signal encoding, the connector form factor and pinout, and the cabling interfaces. When the packet is received, using the Physical layer, the packet travels up the seven layers and arrives at the Application. There are two basic types of data link protocols, a master-slave interaction Data Link layer is tasked to establish and maintain point-to-point Wide Area Network (WAN) links. Each data link is actually two, and only two, entities communicating at any one time through called Normal Response Mode (NRM, also known as Unbalanced), and a peer to peer relationship call Asynchronous Balanced Mode. NRM has all connection initialization, configuration and control done by the designated master (or Primary Station), ABM allows either peer (“Balanced”) to initialize and control the link at any time (“Asynchronous”). NRM may have more than one Secondary Station (multidrop, multipoint), but they must communicate by way of the Primary Station.

HDLC is a protocol developed by the International Organization for Standardization (ISO). It falls under the ISO standards ISO 3309 and ISO 4335. It has found itself being used throughout the world. It has been so widely implemented because it supports both half duplex and full duplex communication lines, point to point(peer to peer) and multi-point networks, and switched or non-switched channels. The procedures outlined in HDLC are designed to permit synchronous, code-transparent data transmission. Other benefits of HDLC are that the control information is always in the same position, and specific bit patterns used for control differ dramatically from those in representing data, which reduces the chance of errors.

It has also led to many subsets. Two subsets widely in use are Synchronous Data Link Control (SDLC) and Link Access Procedure-Balanced (LAP-B). This technical overview will be concerned with the following aspects of HDLC:

- Stations and Configurations
- Operational Modes
- Non-Operational Modes
- Frame Structure
- Commands and Responses

3. THE OSI MODEL

The Open Systems Interconnection (OSI) is a seven-layer protocol stack model used as a reference throughout the computer industry. The ISO was developed as a functional reference model only.

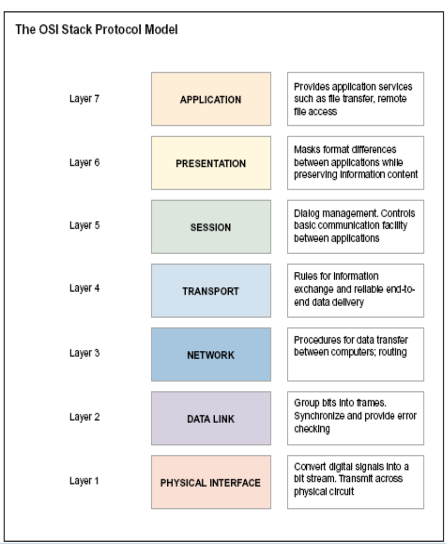

Fig.1 The OSI stack protocol model

One must note that some well known protocols such as TCP/IP and Signaling System 5 (SS5) were developed before the OSI model, and therefore do not exactly align with the defined OSI boundaries, where protocol stacks like IBM's SNA have a very close relationship. However, when used as a reference, the OSI model gives us a general idea of what to expect from the various layers of a given protocol stack.

3.1 Data Link Layer

Networks are defined by the International Organization for Standardization (ISO) using the Open Systems Interconnection (OSI) Reference Model. This model consists of seven layers which detail different aspects of a network. The seven layers are, from lowest to highest level of abstraction, Physical layer, Data Link layer, Network layer, Transport layer, Session layer, Presentation layer, and Application layer.

The Data Link layer is tasked to establish and maintain point-to-point Wide Area Network (WAN) links. Each data link is actually two, and only two, entities communicating at any one time through a physical port. There are two basic types of data link protocols, a master-slave interaction called Normal Response Mode (NRM, also known as Unbalanced), and a peer to peer relationship called Asynchronous Balanced Mode. NRM has all connection initialization, configuration and control done by the designated master (or Primary Station), ABM allows either peer ("Balanced") to initialize and control the link at any time ("Asynchronous"). NRM may have more than one Secondary Station (multidrop, multipoint), but they must communicate by way of the Primary Station.

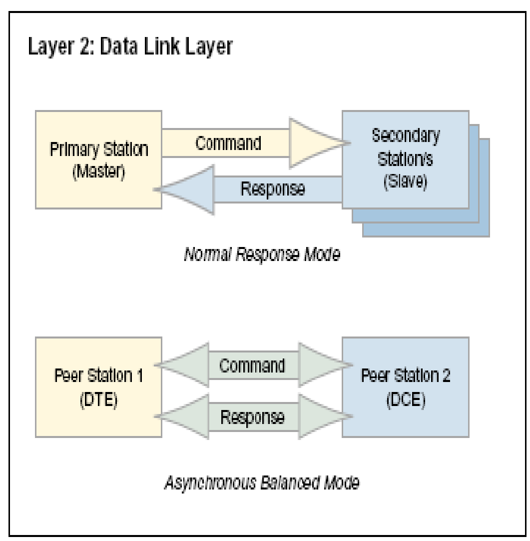

Fig.2 Layer 2:Data Link Layer

3.2 Bit Orientated Protocol (Bop)

Character orientated protocols are still inefficient. This is because a character is used to convey meaning. As the number of meanings increase, the overhead involved also increases, as a character is used to signal the meaning.

In bit orientated protocols, each bit has significance. The position and value of each bit in the data stream determines its function. Thus, a single character can hold 254 different meanings in a bit orientated protocol. This reduces the information needed to convey additional information, thus increasing the efficiency of the protocol.

Examples of these types of protocols are,

- X.25 CCITT standard for packet data transmission
- HDLC high level data link control (adopted by ISO in 1950's)
- SDLC synchronous data link control (developed by IBM)

Links between sender and receiver can be either half duplex, full duplex or both. Information can be sent across the network in two different ways, travelling different routes to the receiver (*datagram*), or travelling the same route (*virtual circuit*).
Information is packaged into an envelope, called a FRAME. Each frame has a similar format

- header containing routing and control information
- body
- tail containing checksum data

Frames are responsible for transporting the data to the next point. Consider data that is to be sent from a source to a destination. This involves several intermediate points (called stations). The data is placed into a frame and sent to the next station, where the frame is checked for validity and if valid, the data extracted. The data is now repackaged into a new frame and sent by that station to the next station, and the process repeats till the data arrives at the destination. When a station transmits a frame, it keeps a copy of the frame contents till the frame is acknowledged as correctly received by the next station. When a station receives a frame, it is temporarily stored in a buffer and checked for errors. If the frame has errors, the station will ask the previous station to resend the frame. Frames that are received without errors are also acknowledged, at which point the sending station can erase its copy of the frame. A receiving station has a limited amount of buffer space to store incoming frames. When it runs out of buffer space, it signals other stations that it cannot receive any more frames. Data is placed into frames for sending across a transmission link. The frame allows intelligent control of the transmission link, as well as supporting multiple stations, error recovery, intelligent (adaptive) routing and other important functions.

A bit orientated protocol sends information as a sequence of bits. An example of a bit orientated protocol is HDLC. Frames are used as a transport mechanism to transport data from one point to another. A frame contains error checking information which allows data to be sent reliably from a sender to a receiver.

Three frames types are defined, and data is normally send using Information frames. At any one time, a number of Information frames can be unacknowledged by a secondary station, and this is called the *sliding window value*, which defaults to 2, but can be negotiated when a call is first established.

4. HDLC PROTOCOL

HDLC is a specification for the Data Link layer and 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, of which HDLC is a part of, is responsible for passing the data between two nodes on the same network. HDLC 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.

The Physical layer specifies how the bits are transmitted. This can include, but is not limited to, the data signal encoding, the connector form factor and pinout, and the cabling interfaces. When the packet is received, using the Physical layer, the packet travels up the seven layers and arrives at the Application.

High-Level Data Link Control, also known as HDLC, is a bit oriented, switched and non-switched protocol. It is a data link control protocol, and falls within layer 2, the Data Link Layer, of the Open Systems Interface(OSI) model as shown in figure.

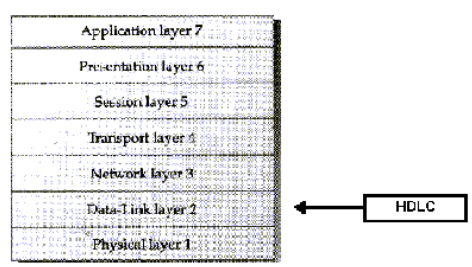

Fig.3 Open system interface model

HDLC is a protocol developed by the International Organization for Standardization(ISO). The original ISO standards for HDLC are:

- ISO 3309 — Frame Structure
- ISO 4335 — Elements of Procedure
- ISO 4159 — Unbalanced Classes of Procedure
- ISO 4254 — Balanced Classes of Procedure

The current standard for HDLC is ISO 13239, which replaces all of those standards. It has been so widely implemented because it supports both half duplex and full duplex communication lines, point to point(peer to peer) and multi-point networks, and switched or non-switched channels. The procedures outlined in HDLC are designed to permit synchronous, code-transparent data transmission. Other benefits of HDLC are that the control information is always in the same position, and specific bit patterns used for control differ dramatically from those in representing data, which reduces the chance of errors.

It has also led to many subsets. Two subsets widely in use are Synchronous Data Link Control (SDLC) and Link Access Procedure-Balanced (LAP-B).

4.1 Purpose

Communication protocols are sets of rules that govern the interaction of concurrent processes in distributed systems. A protocol specification consists of five parts. To be complete, each specification should include:

1. The *service(s)* to be provided by the protocol
2. The *assumptions* about the environment in which the protocol is executed
3. The *vocabulary* of messages used to implement the protocol
4. The *encoding* (format) of each message in the vocabulary
5. The *procedure rules* guarding the consistency of message exchanges

This CPD attempts to address all of these parts, describing the protocol without ambiguity.

4.2 Technical Overview

Data link protocols may be grouped into three broad categories depending on the packet framing method associated with the given protocol. The three types are character-oriented, byte count-oriented, and bit-oriented. Character oriented protocols use special characters to delimit the packet frame structure. An example of this type of protocol is IBM's Binary Synchronous Communications Protocol, or BISYNC.

Byte count-oriented protocols identify the packet structure with a special character in the beginning followed by a count of the bytes for the data part of the packet. Kermit is an example of this type of protocol. PPP uses the third type, the bit-orineted protocol. This protocol identifies the beginning and end of a packet with a specialized sequence of bits called a flag character. Suppose you specify that no bit pattern shall have more than six "1" bits in a row unless it is a flag character. So the bit pattern of 01111110 will always specify a flag, and this flag will always delimit the beginning and the end of a packet. The High (level) Data Link Control (HDLC) protocol as specified by the International Standards Organization (ISO) uses beginning and ending flags.PPP is one of many derivative protocols from HDLC. HDLC is defined by a number of ISO specifications: notably the documents ISO 3309, ISO 4335, and ISO 5809 among others. The following is a listing of some of the most important implementations of HDLC (with the technology they support in parentheses): LAPB(x.25), LAPD(ISDN), LAPX(TELETEX), PPP(many technologies), SDLC(SNA). Frame relay is also a derivative of HDLC, but frame relay does not support an overlying network layer.

The consequence of PPP's relationship to HDLC is that the basic packet structure of PPP resembles very closely the structure of the HDLC packet. In HDLC terms the data packet transmitted across the link is called a frame, and a frame will consist of the following elements:

Field	**Length**
Flag fields	8 bits
Address field	8, or multiples of 8 bits
Control fields	8, or multiples of 8 bits
Information field	variable length, not used in some frames
Frame check sequence field (FCS) or CRC.	14 bits1

4.3 Hdlc Stations And Configurations

HDLC specifies the following three types of stations for data link control:

1.Primary Station

2.Secondary Station

3.Combined Station

Primary Station

Within a network using HDLC as it's data link protocol, if a configuration is used in which there is a primary station, it is used as the controlling station on the link. It has the responsibility of controlling all other stations on the link(usually secondary stations). Despite this important aspect of being on the link, the primary station is also responsible for the organization of data flow on the link. It also takes care of error recovery at the data link level(layer 2 of the OSI model).

Secondary Station

If the data link protocol being used is HDLC, and a primary station is present, a secondary station must also be present on the data link. The secondary station is under the control of the primary station. It has no ability, or direct responsibility for controlling the link. It is only activated when requested by the primary station. It only responds to the primary station. The secondary station's frames are called responses. It can only send response frames when requested by the primary station.

A combined station is a combination of a primary and secondary station. On the link, all combined stations are able to send and receive commands and responses without any permission from any other stations on the link. Each combined station is in full control of itself, and does not rely on any other stations on the link. No other stations can control any combined station.

HDLC also defines three types of configurations for the three types of stations:

- Unbalanced Configuration
- Balanced Configuration
- Symmetrical Configuration

Unbalanced Configuration

The unbalanced configuration in an HDLC link consists of a primary station and one or more secondary stations. The unbalanced occurs because one stations controls the other stations. In a unbalanced configurations, any of the following can be used:

- Full - Duplex or Half - Duplex operation
- Point to Point or Multi-point networks

An example of an unbalanced configuration can be found below in figure 4.

Balanced CONFIGURATION

The balanced configuration in an HDLC link consists of two or more combined stations. Each of the stations have equal and complimentary responsibility compared to each other. Balanced configurations can used only the following:

- Full - Duplex or Half - Duplex operation
- Point to Point networks

An example of a balanced configuration can be found below.

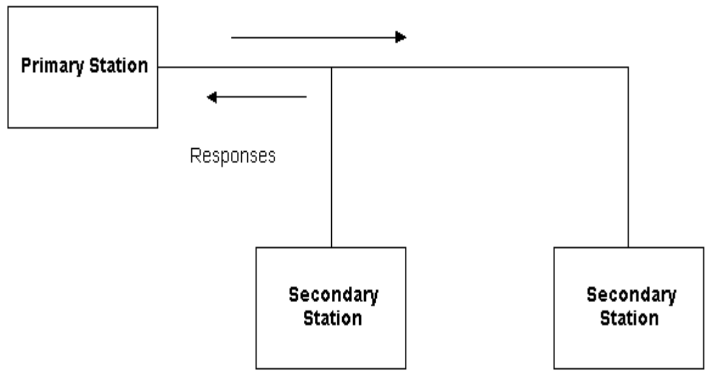

Fig. 4 An unbalanced configuration

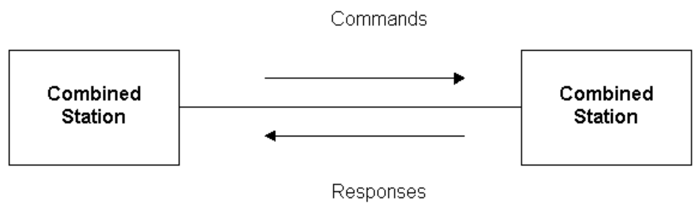

Fig. 5 A balanced configuration

Symmetrical Configuration

This third type of configuration is not widely in use today. It consists of two independent point to point, unbalanced station configurations. In this configurations, each station has a primary and secondary status. Each station is logically considered as two stations.

4.4 Hdlc Operational Modes

HDLC offers three different modes of operation. These three modes of operations are:

- Normal Response Mode(NRM)
- Asynchronous Response Mode(ARM)
- Asynchronous Balanced Mode(ABM)

Normal Response Mode

This is the mode in which the primary station initiates transfers to the secondary station. The secondary station can only transmit a response when, and only when, it is instructed to do so by the primary station. In other words, the secondary station must receive explicit permission from the primary station to transfer a response. After receiving permission from the primary station, the secondary station initiates it's transmission. This transmission from the secondary station to the primary station may be much more than just an acknowledgment of a frame. It may in fact be more than one information frame. Once the last frame is transmitted by the secondary station, it must wait once again from explicit permission to transfer anything, from the primary station. Normal Response Mode is only used within an unbalanced configuration.

Asynchronous Response Mode

In this mode, the primary station doesn't initiate transfers to the secondary station. In fact, the secondary station does not have to wait to receive explicit permission from the primary station to transfer any frames. The frames may be more than just acknowledgment frames. They may contain data, or control information regarding the status of the secondary station. This mode can reduce overhead on the link, as no frames need to be transferred in order to give the secondary station permission to initiate a transfer. However some limitations do exist. Due to the fact that this mode is Asynchronous, the secondary station must wait until it detects and idle channel before it can transfer any frames. This is when the ARM link is operating at half-duplex. If the ARM link is operating at full-duplex, the secondary station can transmit at any time. In this mode, the primary station still retains responsibility for error recovery, link setup, and link disconnection.

Asynchronous Balanced Mode

This mode uses combined stations. There is no need for permission on the part of any station in this mode. This is because combined stations do not require any sort of instructions to perform any task on the link.

Normal Response Mode is used most frequently in multi-point lines, where the primary station controls the link. Asynchronous Response Mode is better for point to point links, as it reduces overhead. Asynchronous Balanced Mode is not used widely today.

The "asynchronous" in both ARM and ABM does not refer to the format of the data on the link. It refers to the fact that any given station can transfer frames without explicit permission or instruction from any other station.

4.5 Hdlc Non-Operational Modes

HDLC also defines three non-operational modes. These three non-operational modes are:

- Normal Disconnected Mode(NDM)
- Asynchronous Disconnected Mode(ADM)
- Initialization Mode(IM)

The two disconnected modes(NDM and ADM) differ from the operational modes in that the secondary station is logically disconnected from the link(note the secondary station is not physically disconnected from the link). The IM mode is different from the operations modes in that the secondary station's data link control program is in need of regeneration or it is in need of an exchange of parameters to be used in an operational mode.

5. COMPARE AND CONTRAST

HDLC is a VTEL proprietary implementation of the standard HDLC packet based protocol. That is to say, it uses the basic packet structure, bit-stuffing and CRCs as defined by the HDLC protocol, but the data format used inside the packet is defined by VTEL. Data is sent in 254 byte packets, which after bit-stuffing and CRCs, may be a slightly larger number of bytes worth of data. The data contained in each packet is defined by a VTEL header value, and may include audio, video, and/or data in any packet. All data types are passed as bytes of data. Data is handled in a FIFO (First In, First Out) manner, with a minimum of buffering. Because outgoing data is processed as it is received from the encoders, and incoming data is sent down to the decoders in the same way, with both audio and video following the same physical and logical path, there are no time stamps involved.

H.320 is an ITU defined standard implementation, covered by many different individual standards. It is a synchronous serial bit stream, in which every bit in the stream is expected to be of a particular data type as specified in a series of capabilities and commands exchanged between the endpoints. Because it is a synchronous bit stream, there is essentially no elasticity allowed in the data. That is, if the connection is made at 384 Kbps, it is expected that there will be exactly 384,000 bits of data each second. Further, because it is a serial bit stream, the data must be muxed and demuxed at the bit field level. H.320, in fact, has no real notion of bytes at all, except that each DS0 of multi-channel connection is 8-bits wide; however, even here, these fields are referred to as “octets” rather than bytes. This is because the data fields in each DS0 may be any number of bits in width.

Put another way, all data in an HDLC connection are byte size pieces of information. Data fields in an H.320 connection may be any number of bits in size, from 1 bit of alignment/command/capability data, 2 or 4 or 5 bits of audio data, 1 to many octets of video data, and similarly diverse size “data” fields. Further, these bits must be laid out in a very tightly specified manner. Finding HDLC packets is based upon recognizing flag data bytes, which separate individual packets, and then unstuffing bits, parsing the VTEL defined header, and separating the data bytes. Finding H.320/H.221 Frame Alignment is based upon searching at every bit position for an 8-bit pattern (the Frame Alignment Signal) which is spread across a number of octets within the serial bit stream, based upon the connection line rate. Once this pattern is found, data must be demuxed, based upon the number of DS0's comprising the total bandwidth. Again, the data that is of interest does not fall into byte format, but rather must be extracted by accumulating a number of bits, which may lie some distance apart. Unfortunately, there is no easy way to explain how this mux/demux operation works. Suffice it to say that there is a lot of bit twiddling going on in both the output and input directions.

HDLC essentially does not care at what rate a connection is made. It doesn't really care about the accuracy of the Network clock. It is merely looking for packets, at whatever rate the data gets clocked onto the Comm subsystem. If there are gaps in the data, either within an individual packet or between packets, it should not cause a problem. The HDLC search algorithm is merely looking for packets at whatever rate they happen to appear. Whatever data is received is passed on to the appropriate decoder as it is received. Short breaks in the data stream often go unnoticed, as there is a certain amount of buffering of data between the Comm subsystem and the decoders.

H.320, however, is extremely sensitive to the Network clock. And the Network Interface Cards used in VTEL equipment are likewise extremely sensitive to the Network clock. As stated previously, H.320 expects to receive precisely the number of bits prescribed in the capabilities and commands that it has received. If gaps are presented within the H.320 defined serial bit stream, they will cause a loss of H.221 Frame Alignment, which will cause a break in the synchronous data. This will require the system to re-align to the serial bit stream, go through another capabilities exchange, and restart processing audio and video. This will cause a break in the video and audio, often resulting in frozen video and dropped audio, as data received during this realignment will usually be lost and not buffered.

HDLC is a very feature rich protocol, in that any HDLC capable endpoint can use any audio or video algorithm, and it can pass PC data, FarEndCameraControl, send/receive slides and annotations. H.320 can be a very feature rich protocol, but it is dependent upon the capabilities presented by all endpoints to determine which features may be used during any given conference. That is, a fully enabled endpoint, which connects to a basic model type endpoint, may not be able to use all its features, as it will be limited to what the basic model can do. For instance, not all H.320 endpoints support FarEndCameraControl, or data sharing.

As this should demonstrate, HDLC, compared to H.320, is about as 'apples-to-oranges' as a comparison can get. The biggest problem for ATM networks is the lack of elasticity. If HDLC packets get delayed slightly by the buffering required to traverse the network, it is no big problem, provided the delays are not sufficient to allow the inbound buffers to dry up. If H320 data get delayed, then the H.221 framing gets lost, and data is lost. Further, if buffers within the network either dry up or overflow, this condition will always result in lost data and discontinuities in the audio and video.

5.1 HDLC vs PPP

Both HDLC and PPP are data link layer protocols. HDLC (High-Level Data Link Control) is a communication protocol used at the data link layer of computer networks, developed by ISO (International Organization for Standardization), and was created out of IBM's SDLC (Synchronous Data Link Control). PPP is a data link layer protocol based on HDLC and is very similar to HDLC. Both are WAN (Wide Area Network) protocols and work well to connect point-to-point leased lines.

5.1.1 What is HDLC?

HDLC came in to existence only when IBM submitted SDLC to various standards committees and one of them (ISO) modified SDLC and created HDLC protocol. HDLC is regarded as a compatible superset of SDLC. It is a bit-oriented synchronous protocol. HDLC supports synchronous, full-duplex operation. HDLC has an option for 32-bit checksum and HDLC support the Point-to-point and Multipoint configurations. HDLC identifies "primary" node type, which controls others stations that are called "secondary" nodes. Only a primary node will control the secondary nodes. HDLC supports three transfer modes and they are as follows. First one is the Normal Response Mode (NRM) in which secondary nodes cannot communicate with a primary until the primary has given permission. Secondly, the Asynchronous Response Mode (ARM) allows secondary nodes to talk without primary's permission. Finally, it has Asynchronous Balanced Mode (ABM), which introduces a combined node, and all ABM communication happens between these kinds of nodes only.

5.1.2. What is PPP?

As mentioned above, PPP is a data link layer protocol based on HDLC, and is very similar to HDLC. It is used for the direct communication between two nodes. Transmission encryption privacy, authentication and compression are provided by PPP. Authentication is provided by PAP (Password Authentication Protocol) and more commonly by CHAP (Challenge Handshake Protocol) protocols. It is used for various types of networks that are made up of different physical mediums such as trunk line, fiber optics, serial cable, cellular telephone and phone line. It is very popular among ISPs (Internet Service Providers) as a medium for providing the customers with dial-up access to the Internet. To provide DSL (Digital Subscriber Line) services to their customers, service providers use Point-to-Point Protocol over Ethernet (POPoE) and Point-to-Point Protocol over ATM (POPoA), which are two encapsulated forms of PPP. PPP is used for both synchronous and asynchronous circuits. It works with different network protocols such as IP (Internet Protocol), IPX (Internetwork Packet Exchange), NBF and AppleTalk. Broadband connections also use PPP. Although PPP was designed somewhat after the original HDLC specifications, PPP includes many additional features that had been available only in proprietary data link protocols at that moment of time.

Although, HDLC and PPP are very similar WAN data link layer protocols used for point-to-point communications, they do have their differences. Unlike HDLC, PPP is not proprietary when used in a Cisco router. Several sub-protocols make up the functionality of PPP. PPP is feature-rich with dial-up networking features and is used heavily by ISPs to provide Internet to their customers. Unlike HDLC, PPP can be used with both synchronous and asynchronous connections.

5.2 HDLC vs SDLC

HDLC and SDLC are communication protocols. SDLC (Synchronous Data Link Control) is a communication protocol used at the data link layer of computer networks, developed by IBM. HDLC (High-Level Data Link Control) is again a data link protocol, developed by ISO (International Organization for Standardization), and was created out of SDLC.

SDLC was developed by IBM in 1955 to be used in Systems Network Architecture (SNA) environments. It was synchronous and bit-oriented and was one of the first of its kind. It surpassed the synchronous, character-oriented (i.e. Bisync from IBM) and synchronous byte-count-oriented protocols (i.e. DDCMP from DEC) in efficiency, flexibility and speed. Various link types and technologies such as point-to-point and multipoint links, bounded and unbounded media, half-duplex and full-duplex transmission facilities and circuit-switched and packet-switched networks are supported. SDLC identifies “primary” node type, which controls others stations, which are called “secondly” nodes. So the secondary nodes will be controlled only by a primary. Primary will communicate with secondary nodes using polling. Secondary nodes cannot transmit without the permission of the primary. Four basic configurations, namely, Point-to-point, Multipoint, Loop and Hub go-ahead can be used to connect primary with secondary nodes. Point-to-point involves only one primary and secondary while Multipoint means one primary and many secondary nodes. Loop topology is involved with Loop, which is essentially connecting primary to the first secondary and last secondary again connected to primary so that intermediate secondaries pass messages through one another as they respond to the requests of the primary. Finally, Hub go-ahead involves an inbound and outbound channel for the communication to secondary nodes.

HDLC came in to existence only when IBM submitted SDLC to various standards committees and one of them (ISO) modified SDLC and created HDLC protocol. It is again a bit-oriented synchronous protocol. Despite the fact that several features used in SDLC are omitted, HDLC is regarded as a compatible superset of SDLC. SDLC Frame format is shared by HDLC. Fields of HDLC has the same functionality of those in SDLC. HDLC too, supports synchronous, full-duplex operation as SDLC. HDLC has an option for 32-bit checksum and HDLC does not support the Loop or Hub go-ahead configurations, which are clear minor differences from

SDLC. But, the main difference comes from the fact that HDLC supports three transfer modes as opposed to one in SDLC. First one is the Normal response mode (NRM) in which secondary nodes cannot communicate with a primary until the primary has given permission. This is actually the transfer mode used in SDLC. Secondly, the Asynchronous response mode (ARM) allows secondary nodes to talk without primary’s permission. Finally it has Asynchronous balanced mode (ABM) which introduces a combined node, and all ABM communication happens between these kinds of nodes only.

In summary, SDLC and HDLC are both data link layer network protocols. SDLC was developed by IBM while HDLC was defined by ISO using SDLC as the basis. HDLC has more functionality, although, some features of SDLC are not present in HDLC. SDLC can be used with four configurations while HDLC can be used with only two. HDLC has an option for 32-bit checksum. Major difference between these two is the transfer modes that they have. SDLC has only one transfer mode, which is NRM but, HDLC has three modes including NRM.

6. HDLC FEATURES

- Flexible data buffers with multiple buffers per frame

- Separate interrupts for frames and buffers
- Received frames threshold to reduce interrupt overhead
- Four address comparison registers with mask
- Maintenance of four 14-bit error counters
- Flag/abort/idle generation/detection
- Zero insertion/deletion
- 14/32-bit CRC-CCITT generation/detection
- Detection of non-octet aligned frames and frame too long
- Programmable number of flags between successive frames.

These are the basic features of the controller in HDLC mode. Flexible data buffers with the ability to have multiple buffers per frame. Separate Interrupts available for frames as well as buffers. Each time a buffer is closed an event is generated which can create an interrupt to the core. There is a similar event generated when a frame is completed. Received frames threshold to reduce interrupt overhead. If a large number of frames are expected that could result in an unacceptable interrupt overhead the interrupt can be generated based on multiple frames instead of each one. Maintenance of 14-bit error counters. These simply count the defined errors giving the user statistics for errors. Flag, abort and idle, generation and detection. Transmitted frames have the start and end flags automatically added to the frame before transmission, and the abort and idle line states generated as necessary. The receiver automatically recognizes and strips off the flags, and recognizes abort and idle states on the line.Zero insertion and deletion. Because it is necessary to have a predefined flag at the start and end of a frame to enable the controller to recognize it, it’s not possible to have data within the frame that mimics a flag. However, it is not desirable to have a situation where any desired character cannot be used for data. This is handled by a process commonly called bit stuffing. An HDLC flag is six ones with a leading and trailing zero, value hex seven E. To ensure that value within the data cannot be confused with a flag, if the transmitter sees a zero followed by five ones in the data stream it automatically adds a zero into the stream. At the receiver, if a zero that is followed by five ones is received, the controller deletes the following bit stripping the zero insertion.

- 14- or 32-bit CRC generation and detection, giving the user the ability to select which size is appropriate for the application. As data is transmitted, a CRC calculation is performed on the data and when the end of the frame is reached, the CRC value is attached to the end of the frame. Similarly, as data is received the same calculation is performed and when the end of the frame is recognized, the value at the end of the frame is compared to that calculated by the receiver. If they do not match, a CRC error indication is generated.
- Detection of non-octet aligned frames and frames that are too long. As the serial stream of bits is received they are accumulated into bytes. If for some reason there is a number of bits that are not divisible by eight then an event is generated to alert the user. There is also a parameter to define the maximum length allowed for a frame and if a frame is received longer than that value an event is generated.
- Programmable number of flags between successive frames. If a continuous stream of frames is transmitted, is it desirable to have them directly adjacent to each other? In other words, the end flag immediately next to the start flag of the next frame. If it is desirable to introduce a delay between frames, perhaps due to timing constraints, then the user can select to automatically introduce up to 15 flags between successive frames Handled automatically by both receiver and transmitter Handled automatically.

7. RESEARCH CHALLENGES

This paper can be used as single-channel HDLC protocol we can able to implement multi-channel HDLC controller with few modifications in the VHDL coding. The main difference single channel and multi channel HDLC controller is that a multiplexer and a demultiplexer are used in the transmitter and receiver respectively. The main advantage of this project is error correction using CRC-14 polynomial and through this controller we can achieve error free communication. Thus this project can be used in MODEMS where error detection part is performed by this HDLC controller.

HDLC forms the basic standard for other protocols like LAPB, LAP, LLC, LAN etc., this controller can be used to generate frame for any of these protocols with slight modification in the coding part. This HDLC controller can also support ISDN frame format and thus the flexibility in this controller is very high.

8. CONCLUSION

The high level data link control (HDLC) protocol defined by the ISO provides a transparent transmission Service at the data link layer of the ISO reference model. Many protocol suites use an HDLC link layer, including X.25, The IP point-to-point protocol (PPT) and SNA. It has been so widely implemented because it supports half duplex and full duplex communication lines, point-to-point (peer to peer) and multipoint networks, and switched or non switched channels. The procedures outlined in HDLC are designed to permit synchronous, code-transparent data transmission. The HDLC Protocol is a general purpose protocol, which Operates AT the data link layer of the OSI reference model. The protocol uses the services of a physical layer, to provide communications path between the transmitter and receiver. The users of the HDLC service provide PDUs, which are encapsulated to form data link layer frames.

HDLC is the basis for a wide variety of data link layer protocols. Regardless of the data link protocol, most HDLC framing, bit stuffing, and CRC generation is performed by the serial controller chips with the specific protocol stack software supplying the rest of the necessary fields (see Layer 2 diagram on page two). This means that most WAN cards can be adapted to any HDLC derived protocol

REFERENCES

[I] K.C&hang, “Digiful Design & Modeling with VHDL and Synthesis”, IEEE Computer Society Press, 1995.

[2] J.F.Wakerly, “Digital Design principles & practices”, Prentice Hall, New Jersey, 2000.

[3] Z.Navabi, “VHDL Analysis & Modeling of Digital systems”, McGraw Hill Inc., 1993.

[4] Xilinx Application Note,; “Single-Ch’anid HDLC Core v2.0 , Xilinx Inc., 2tOl.

[5I “The Programmable Logic Data Book”, Xilinx Inc., 1994.

[4] Z.Navabi, “VHDL Analysis & Modeling of Digital systems”, McGraw Hill Inc., 1993.

[5] A.Tannenbaum, “[Computer Networks”, Prentice Hall of India, 1993.

[6] Guozheng Li Nanlin Tan State Key Lab. of Rail Traffic Control & Safety, Beijing Jiaotong Univ., Beijing, China “Design and Implementation of HDLC Protocol and Manchester Encoding Based on FPGA in Train Communication Network”, International Journal of Scientific & Engineering Research Volume 2, Issue 3, March-2011 5 ISSN 2229-5518 IJSER © 2011 http://www.ijser.org Information and Computing (ICIC), 2010 Third International Conference

[7] Lu, Y., Z. Wang, L. Qiao and B. Huanq, 2002. "Design and implementation of multi-channel high speed HDLC data processor," IEEE International Conference on Communications, Circuits and Systems, and West Sino Expositions, 2: 1451-1455

[8] Amendola, A.M. Benso, A. Corno, F. Impagliazzo, L. Marmo, P. Prinetto, P. Rebaudengo, M. Sonza Reorda, M. CRIS, Napoli “Fault behavior observation of a microprocessor system through a VHDL simulation-based fault injection experiment Design Automation Conference, 1994, with EURO-VHDL '94 and Exhibition, Proceedings EURO-DAC '94, European ,14-20 Sep 1994

[9] Arshak, K. Jafer, E. McDonagh, D. Ibala, C.S. Univ. of Limerick, Limerick, “Modelling and simulation of wireless sensor system for health monitoring using HDL and Simulink mixed environment” Computers & Digital Techniques, IET Sept. 2005

[10] Gheorghiu, V., S. Kameda, T. Takagi, K. Tsubouchi and F. Adachi, 2008. "Implementation of frequency domain equalizer for single carrier transmission," In Proceedings of the 4th International Conference on Wireless Communications, Networking and Mobile Computing, WiCOM '08.

[11] 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 procedures based on FPGA” , Anti- ounterfeiting, Security, and Identification in Communication, 2009. ASID 2009. 3rd International Conference, 20-22 Aug. 2009

[12] Meng, X. and V. Chaudary, 2009. "Boosting data throughput for sequence database similarity searches on FPGAs using an adaptive buffering scheme," Parallel Computing, 35(1): 1-11