M.Sridevi, Dr P.Sudhakar Reddy / International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com
Vol. 2, Issue 5, September- October 2012, pp.2217-2219

Design And Implementation Of Hdlc Protocol On Fpga

M.Sridevi*, Dr P.Sudhakar Reddy**
*(Department of ECE, Srikalasthiswara institute of technology, Srikalasthi, Chittor, Andhra Pradesh)
**(Department of ECE, Srikalasthiswara institute of technology, Srikalasthi, Chittor, Andhra Pradesh)

ABSTRACT

High-level Data Link Control (HDLC) procedure is one of the most important protocols in digital and data communication system development. HDLC procedures are the high-level data link control procedures established by ISO, which are widely used in digital communication and are the bases of many other data link control protocols. HDLC procedures are commonly performed by FPGA/ASIC devices and embedded software programming. Here we are proposing new implementation of HDLC protocol for sensor network data processing system design and implementation based on Low power FPGA Architecture. The over all System Architecture will be designed using Veri-log and simulation, synthesis and implementation (Translation, Mapping, Placing and Routing) will be done using various FPGA based EDA Tools like Xilings. Finally the proposed system architecture performance (speed, area, power and throughput) will be compared with already exiting system implementations.

Keywords - ASIC, EDA TOOLS, FPGA, HDLC, Veri-log.

I. Introduction

High-level Data Link Control (HDLC) procedure is
one of the most important protocols in digital communications. HDLC procedures are commonly performed by ASIC (Application Specific Integrated Circuit) devices, software programming. These ASIC devices are manufactured by china and these have some problems those are lack of flexibility indifferent applications, On chip data storage capacity in ASIC is limited, We need only few ASIC chips to meet the required HDLC design, Implementation of HDLC procedure in FPGA is costly because of using software like Xilings. 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, software programming is often used in single channels and low-speed signal processing system. 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.

II. HDLC Module

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.

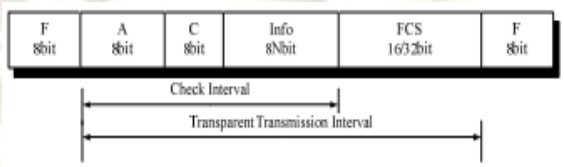

Fig 1: HDLC bit frame.

- FLAG: flag is for synchronous transmission “01111110” (7E, hexadecimal)

for both frame start and frame end.

- Address: the field for address of receiving station.
- control: controls the communication process
- . information : the information field contains the transported data.
- CRC: cyclical redundancy check is used for error detection.(16bit)

II.1 Bit stuffing

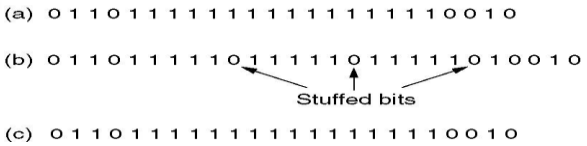

Fig 2: Bit stuffing.

(a) The original data.

(b) The data as they appear on the line.

(c) The data as they are stored in receiver's memory after unstuffing.

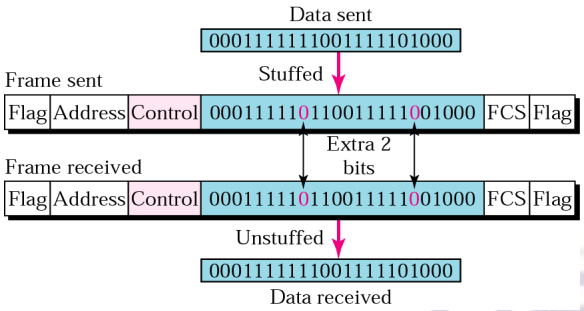

Fig 3: Bit stuffing and removing.

And the proposed technique has the new architecture by including some of the techniques like pipeline, resource sharing and will further investigating the optimization of various system parameters (area, power, throughput, etc). Even I am including some application (Network Sensor Data Processing) for this proposed system.

II.2 CSC

The cyclic redundancy check, or CRC, is a technique for detecting errors in digital data, but not for making corrections when errors are detected. The user data which should be a multiple of 8 bits;"CRCL" and "CRCM" are the 8 bits of LSB and MSB of 16 bit CRC(Cyclic Redundancy Check). The frame check sequence(FCS) to check the data transmission reliability. The FCS is zero insertion and removal for transparent transmission.

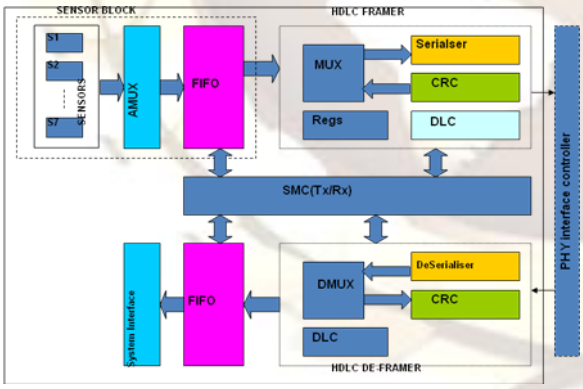

Fig 4: HDLC module design.

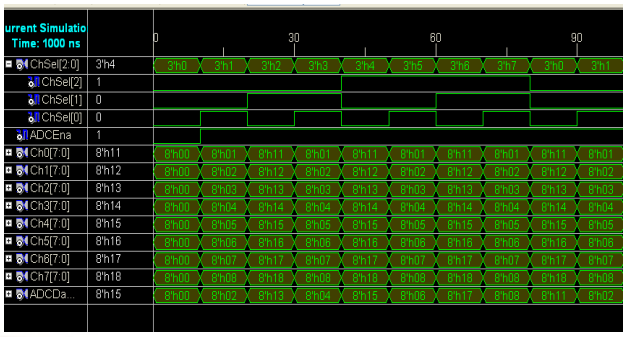

Fig 5: HDLC Transmitter output.

Device Utilization Summary				
Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slice Flip Flops	104	17,344	1%	
Number of 4 input LUTs	239	17,344	1%	
Logic Distribution				
Number of occupied Slices	151	8,672	1%	
Number of Slices containing only related logic	151	151	100%	
Number of Slices containing unrelated logic	0	151	0%	
Total Number of 4 input LUTs	257	17,344	1%	
Number used as logic	212			
Number used as a route-thru	39			
Number used for Dual Port RAMs	16			
Number of bonded IOBs	79	250	31%	
Number of BUFGMUXs	2	24	8%	

Table 1: HDLC transmitter device utilization.

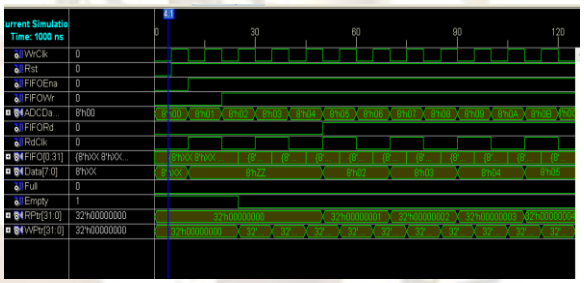

Fig 6: HDLC Receiver output.

Device Utilization Summary				
Logic Utilization	Used	Available	Utilization	Note(s)
Total Number Slice Registers	102	17,344	1%	
Number used as Flip Flops	74			
Number used as Latches	28			
Number of 4 input LUTs	119	17,344	1%	
Logic Distribution				
Number of occupied Slices	96	8,672	1%	
Number of Slices containing only related logic	96	96	100%	
Number of Slices containing unrelated logic	0	96	0%	
Total Number of 4 input LUTs	150	17,344	1%	
Number used as logic	103			
Number used as a route-thru	31			
Number used for Dual Port RAMs	16			
Number of bonded IOBs	41	250	16%	
IOB Flip Flops	8			
IOB Latches	16			
Number of BUFGMUXs	2	24	8%	

Table 2: HDLC receiver device utilization.

III. Conclusion

We designed HDLC procedures' sending and receiving RTL level modules in Veri-log, and downloaded them into FPGA and had them tested successfully, and also we have overcome the problems existed in the previous technique that uses ASIC which is not flexible so this proposed method

can be more useful for many applications like a Communication protocol link for RADAR data processing . Real time embedded sensor data and transmission procedures. Basic communication protocol controller in embedded SOC. Various Material sensor wireless data processing system.

VI. Acknowledgements

The author M. Sridevi thankful to Associative Professor Sri Dr.P.Sudhakar Reddy for his valuable suggestions.

References

[1] Henriksson T, Dake Liu, "Implementation of fast CRC calculation," Proceedings of *the Asia and South Pacific, Design Automatation Conference, pp 563-564*, January 2003.

[2] Ming-Der Shieh, Hsin-Fu Lo, Ming-Hwa Sheu, "High-speed generation of LFSR signatures," Processdings of *Ninth Asia, Test Symposium, pp 222-227*, December 2000.

[3] Mitel Semiconductor, "MT8952B HDLC Protocol Controller," *Mitel Semiconductor Inc., pp 2-14*, May 1997

[4] ISO/IEC 13239, "Information technology - -Telecommunications and Information exchange between systems – High-level data link control (HDLC) procedures," *International Organization for Standardization, pp 10-17*, July 2002.

[6] Li Wang, "Computer Network Practical Guide", Beijing: *Tsinghua University Press, pp 72-78*, 2007. (in Chinese)