International Journal of Advanced and Innovative Research (2278-7844)/

Volume 8 Issue 8

Iterative Inter-Cell Interference Cancellation Receiver for LDPC-Coded MIMO Systems

Jitendra Kushwah[1], Jitendra Kumar Mishra[2]

P.I.E.S College, M.P. 462023, India[1]

ECE Department, RGPV Name, P.I.E.S. College, M.P. 462023, India[2]

Jitendrakushwaha1908@gmail.com[1]

Abstract— This paper presents an implementation of Universal Asynchronous Receiver-Transmitter (UART) controller based on Micro programmed Controller on Field Programmable Gate Array (FPGA. Our design of UART is fully functional and synthesizable. The implementation results demonstrate that the design can operate at a maximum clock frequency of 218.248 MHz. The maximum clock frequency of hardwired implementation of UART controller is 192.773 MHz. We also compare our Micro programmed implementation of UART controller to the standard ROM method; our implementation uses less number of bits and hence small number of storage elements.

Keywords: Receiver Transmitter, Micro programmed Controller, and Field Programmable Gate Array.

I. INTRODUCTION

A UART (Universal Asynchronous Receiver/Transmitter) is the microchip with programming that controls a computer's interface to its attached serial devices. Specifically, it provides the computer with the RS-232C Data Terminal Equipment (DTE) interface so that it can talk to and exchange data with modems and other serial devices. As part of this interface, the UART also:

Converts the bytes it receives from the computer along parallel circuits into a single serial bit stream for outbound transmission

On inbound transmission, converts the serial bit stream into the bytes that the computer handles

Adds a parity bit (if it's been selected) on outbound transmissions and checks the parity of incoming bytes (if selected) and discards the parity bit

Adds start and stop delineators on outbound and strips them from inbound transmissions Handles interrupt s from the keyboard and mouse (which are serial devices with special port s) May handle other kinds of interrupt and device management that require coordinating the computer's speed of operation with device speeds

More advanced UARTs provide some amount of buffering of data so that the computer and serial devices data streams remain coordinated. The most recent UART, the 16550, has a 16-byte buffer thatcan get filled before the computer's processor needs to handle the data. The original UART was the 8250. If you purchase an internal modem today, it probably includes a 16550 UART (although you should ask when you buy it). According to modem manufacturer US Robotics, external modems do not include a UART. If you have an older computer, you may want to add an internal 16550 to get the most out of your external modem.

UARTs are used for serial communication between two devices with minimum wires. The data is sent serially, and no clock signal is sent along with it. The primary function of a UART is parallel-to-serial conversion when transmitting, and serial-to-parallel conversion when receiving. The sender and receiver have separate, unsynchronized, clock signals. In order to synchronize the asynchronous serial data and to insure the data integrity, Start and Stop bits are added to the serial data. An example of the UART frame format is shown in Figure 1. The transmitted character is composed of an 8- bit data byte, sent LSB (least significant bit) first, preceded by a start bit (active low) and followed by a stop bit (active high).

When no character is being transmitted, the line is idle (active high). The line need not go idle between characters, as it is possible for the start bit of a transmission to immediately follow the stop bit of the previous transmission. A field-programmable gate array (FPGA) is a logic device that contains a two dimensional array of generic logic cells and programmable switches. A logic cell can be configured (i.e., programmed) to perform a simple function, and a programmable switch can be customized to provide interconnections among the logic cells. A custom design can be implemented by specifying the function of each logic cell and selectively setting the connection of each programmable switch. Once the design and synthesis are completed, we can use a simple adaptor cable to download the desired logic cell and switch configuration to the FPGA device and obtain the custom circuit [1-8].

II. RELATED WORK

Dr. GarimaBandhawarkarWakhle et al. suggested that, the UART is the use of the serial communication protocol, which permits the full duplex communication in serial link. They design the hardware implementation of a high speed & competent UART using Field Programmable Gate Array. The UART consists of three components, namely receiver, transmitter & baud rate generator which is also frequency divider. They simulated on Modelsim SE 10.0a and design by using Verilog description language which has been synthesized on FPGA kits like as Spartan3 & Virtex4. After analyzing the comparative analysis conclude that there is a difference in between the number of slices, LUTs and the maximum frequency. The results are quite stable and reliable and have great flexibility with high integration. If we use FIFO in making the UART our design becomes more flexible, stable and reliable which provides highest bps rate. [9]

AmanpreetKaur, AmandeepKaur concludes that, A UART is a full duplex collector and transmitter. It is the chip with programming that controls a PC's an interface to its associated sequential gadgets. It deals with the transmission among sequential and parallel information. The entire procedure of sequential transmission is working on the standard of the move enlist. In information transmission through the UART, when the baud-rate has been begun, both the transmitter and the collector's inward clock are set to the indistinguishable recurrence. [10]

According to Bhavna fertilizer and Rahul residency is concerned, building up a sequential correspondence convention including bit synchronization, programmed baud rate discovery with determination and transport, recurrence division as per the info clock. All modules are reproduced on Xilinx Spartan-3 FPGA improvement board utilizing Verilog programming dialect and. In the recreation part center around check the get information with blunder free and baud rate age at different frequencies. The Baud Rate Generator is fused into UART plan, before the whole structure is combined. The significance of recurrence divider at such places where the client needs bring down regular to work the usefulness. This recurrence divider will naturally modify according to necessity. Watched recreated waveforms at different frequencies between 150 bps to 38400 bps at 50 MHz clock cycles. The reenacted waveforms in this demonstrate the consistency of the HDL execution to depict the engineering and highlights of the baud rate generator with UART structure. [11]

Tooth Yi-Yuan Chen Xue-Jun notice that Universal Asynchronous Receiver Transmitter is for the most part utilizing for sequential correspondence just as for low speed, short-remove, minimal effort information trade among PC and its peripherals. The UART structured with VHDL dialect can be merged into the Field Programmable Gate Array to accomplish steady, conservative and productive information transmission. It's vital for the plan of System on Chip. The consequences of reenactment on Quartus II are totally dependable with the UART. In this paper for structuring it utilizes VHDL as a plan dialect to procure the modules of the UART. The reproduction and test part is finished by utilizing Quartus II programming Altera Cyclone arrangement FPGA chip. The outcomes are dependable and stable, the structure has high coordination, incredible adaptability with some reference esteem. Fundamentally in electronic structuring field, where SOC innovation has as of late turned out to be broadly utilized, this plan demonstrates incredible hugeness. [12]

From the study, it is seen that the usage of UART fundamentally utilizes the on-chip UART IP bad-to-the-bone since it has superior, however it has poor

adaptability and poor transportability, hence forth it is generally unfit to meet the high necessities of the client. With the fast advancement of FPGA delicate center assumes an undeniably critical job in implanted framework, contingent upon the elite, high adaptability, transportability and setup. Huimei Yuan, Junyou Yang and Peipei Pan displayed new technique that gives basic Design of UART IP Soft Core dependent on DMA Mode. [13]

Tooth Yi-Yuan CHEN XUE-Jun has exhibited a paper on Design and reenactment Of UART sequential correspondence Module Based on VHDL. In this paper they introduced that the UART is the microchip with programming that controls a PC's interface to its peripherals. It is the most generally utilized sequential information correspondence circuit ever. The entire procedure of sequential transmission depends on the standard of the move enroll. There are two essential types of sequential transmission that are Synchronous and Asynchronous. In Synchronous sequential correspondence necessitates that the sender and recipient should chip away at a similar clock with each other. Offbeat transmission enables information to be transmitted without sending a clock flag to the collector. This structure utilizes VHDL as the plan dialect to accomplish the modules of the UART. The outcomes are solid and stable. The plan has high incorporation, extraordinary adaptability with some reference esteem [14].

Shouqian Yu, Lili Yi, Weihai Chen, Zhao Jin Wen presented a paper on Implementation of a Multi-channel UART Controller Based on FIFO Technique and FPGA. In this paper they have presented that in several systems such as high data collection system, a high speed control system based on PCI and multi-DSP signal processing system, FIFO is used for complete communication between high speed device and low speed device or to complete communication between the same sub controllers. FIFO is the most important part of these systems and it works as a bridge between different devices. At the same, in our controller, asynchronous FIFO based on FPGA is also the most important part. So the features and capabilities of the asynchronous FIFO determine the features of our controller. The FIFO can be used to complete communication in parallel or serial port [15].

III. PROBLEM DEFINITION

An underrun error occurs when the UART transmitter has completed sending a character and the transmit buffer is empty. In asynchronous modes this is treated as an indication that no data remains to be transmitted, rather than an error, since additional stop bits can be appended. This error indication is commonly found in USARTs, since an underrun is more serious in synchronous systems.

A UART will detect a framing error when it does not see a stop bit at the expected stop bit time. As the start bit is used to identify the beginning of an incoming character, its timing is a reference for the remaining bits. If the data line is not in the expected state (high) when the stop bit is expected (according to the number of data and parity bits for which the UART is set), the UART will signal a framing error. A break condition on the line is also signaled as a framing error.

A parity error occurs when the parity of the number of one-bits disagrees with that specified by the parity bit. Use of a parity bit is optional, so this error will only occur if parity-checking has been enabled.

A break condition occurs when the receiver input is at the space (logic low, i.e., '0') level for longer than some duration of time, typically, for more than a character time. This is not necessarily an error, but appears to the receiver as a character of all zero-bits with a framing error. The term break derives from current loop signaling, which was the traditional signaling used for teletypewriters. The spacing condition of a current loop line is indicated by no current flowing, and a very long period of no current flowing is often caused by a break or other fault in the line.

IV. PROPOSED METHODOLOGY

Erasure coding based forwarding Algorithm

Our erasure-coding based forwarding algorithm can be understood as an enhancement to the simple replication algorithm (srep). A simple replication

strategy in which identical copies of the message are sent over the first r contacts. Here, r is the replication factor. Only the source of the message sends multiple copies. The relay nodes are allowed to send only to the destination; they cannot forward it to another relay. This leads to small overhead as the message flooding is controlled to take place only near the source. This class of forwarding algorithms is also known as the two-hop relay algorithm. There is a natural trade-off between overhead (r) and data delivery latency. A higher r leads to more storage/transmissions but has lower delays.

In srep with a replication factor r, the source sends r identical copies over r contacts and relays are only allowed to send directly to the destination. In the erasure-coding based algorithm, we first encode the message at the source and generate a large number of code blocks. The generated code blocks are then equally split among the first kr relays, for some constant k. In comparison with step, this approach uses a factor of k more relays and each relay carries a factor of 1/k less data. However, the number of bytes generated arerM, the same as the number of bytes generated by srep (r). Now by definition of erasure coding (with rate r, message size M), the message can be decoded at the destination if 1/r of the generated code blocks are received. Since code blocks are divided equally among kr relays, the message can be decoded as soon as any k relays deliver their data if we assume that no code blocks are lost during transmissions to and from a relay. When k = 1, the erasure coding approach has the same effect as the simple replication approach, which is, to use the first r relays and to each carry a copy of the original message.

Benefits of erasure coding in forwarding

In simple replication, r relays are used to improve the delay performance. The erasure-coding based approach, instead, utilizes kr relays for the same amount of overhead. Therefore, one can expect that the chances of at least some relays having low delays are higher, compared to using only r relays. At the same time, erasure coding requires at least k relays to succeed (instead of 1 in srep) before the data can be reconstructed. Therefore, if the number of such low-delay relays is larger than k, the erasure-coding based approach will successfully deliver the message with a lower delay than simple replication. The main observation is that if k is large, the delay distribution converges to a constant. Therefore, with the erasure-coding based approach, one can be almost assured of a constant delay.

V. SIMULATION & RESULT ANALYSIS

We have simulated each and every part of our module separately in modelsim6.4b. The simulation resultsfor the „baudgen", „fifo", „uartrx", and „uart-tx" sub modules are shown in the figures below respectively. We have synthesis top module in Xilinx10.1 [27-30].

RESULT ANALYSIS

[1] ***Testing***

our core in Xilinx10.1.starter kit. The test circuit does have an incremented that takes one parellel data stored in The complete core was also tested by connecting it with the PC by a RS232 cable after implementing a test circuit incorporating the rx-fifo after getting received by the UART module, increments it by one and then gives it back to the tx-fifo which is then transmitted serially by the module the reading from the fifo, incrementation and writing to the fifo is done one character at a time when a switch(connected with the test circuit through adebouncer otherwise multiple eading and writing of NULL-character could produce erroneous result)is pressed .Here we first typed-in 4 characters(displayed locally in the heper terminal window) which are stored in the fifo, then pressed the assigned switch 4 times which eturns the stored characters incremented by one and are displayed again in the hyperterminal (as shown in Figure 5.1)

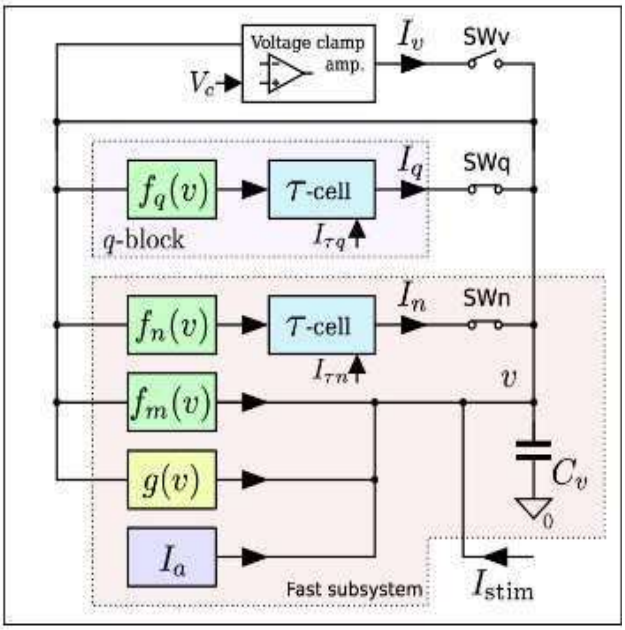

Figure 1: Block Diagram of VLSI.

In the above figure 1 the UART transmitter has been shown.

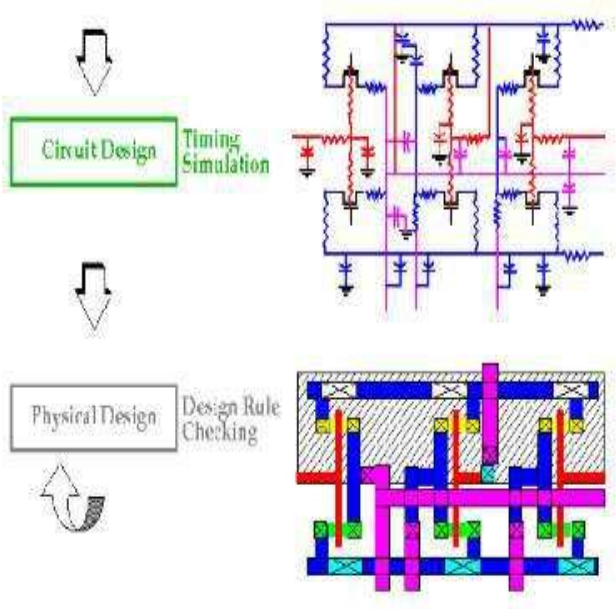

Figure 2: Circuit Design of VLSI.

In the above figure 2 the circuit design of VLSI has been shown.

Area and Power

Total percentage of the device (spartan 3-an) resources utilized to implement our module has been calculated by Xilinx10.1 synthesizer which is found to be nominal.

Table 1: Estimated Values.

Device Utilization Summary (Estimated Values)

Logic Utilization	**Used**	**Available**	**Utilization**
Number of Slices	35	465	7.52%
Number of Slice Flip-Flops	53	931	5.69%
Number of 4 Input LUTs	43	932	4.61%
Number of Bonded IOBs	22	232	9.48%
Number of GCLKs	1	244	2.43%

The table above 1 explains about the Number of Slice, Number of Slice Flip-Flops, Number of 4 Input LUTs, Number of Bonded IOBs, Number of GCLKs comparisons has been shown.

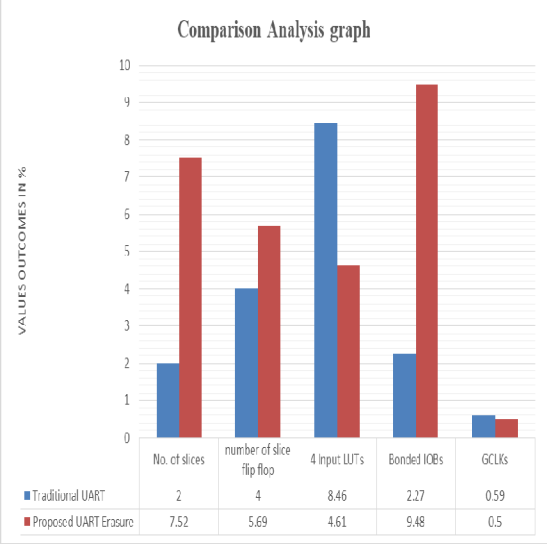

Figure 3: Estimated values.

The above figure 3 represents the estimated values according to the resources available and used.

VII. CONCLUSION & FUTURE WORK

We have concluded that the authors had used FIFO and Shift register separately for storing the data and in some paper they have used baud rate generator with single frequency. In this we had used FIFO as well as Shift register and also an automatic baud rate generator which will change its baud rate according to the change in frequency. This design uses VHDL language to achieve the modules of the UART. We have designed our UART module in generic form which is operating fine with no under run error and can be customized to make it free from overrun error with the capability provided and so can be made available as IPcore (Intellectual Property-Core) by simply coating it with a proper wrapper.

The future work in this thesis may include the manufacture of the board and performing direct tests on the communication and peripheral modules on the board. Data rates and limit to which the data rate can be pushed for normal working of the board can be done. It will serve as a base for students to work on the different data communication modules and peripherals test the data rates using serial UART and I/O pins. Measurement of error rates at different speeds can be carried out.

REFERENCES

[1] FANG Yi-Yuan CHEN Xue-Jun, Design and Simulation of UART Serial Communication Module Based on VHDL Shanghai University of Engineering Science, 2011

[2] AmanpreetKaur, AmandeepKaur, An approach for designing a universal asynchronous receiver transmitter (UART), International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622 www.ijera.com Vol. 2, Issue 3, May-Jun 2012, pp.2305-2311.

[3]en.wikipedia.org/wiki/synchronous_serial_communication.

[4] Cowley, John (2007) communication and networking, An introduction, Springer, ISBN 9781846286452

[5]. www.pccompi.com

[6]. Shiva, S.G University of Alabama in Huntsville, AL, proceedings of the IEEE (volume 67, Issue: 12).

[7]. Comparison of VHDL, Verilog and System Verilog, by StephenBailey, www.model.com

[8]. C. H. Roth, Digital System Design Using VHDL, PWSPublishing Company, 2008.

[9] Dr. GarimaBandhawarkarWakhle, ItiAggarwal and ShwetaGaba, Synthesis and Implementation of UART using VHDL Codes, International Symposium on Computer, Consumer and Control, 978-0-7695- 4655-1/12© 2012 IEEE DOI 10.1109/IS3C.2012.10.

[10] AmanpreetKaur, AmandeepKaur, An Approach For Designing A Universal Asynchronous Receiver Transmitter (UART), IJERA Vol. 2, Issue 3, May-Jun 2012, pp. 2305-2311.

[11] Bhavna manure and Rahul tanwar, UART with automatic baud rate generator and frequency divider Journal of Information Systems and Communication on 15feb 2012.

Mr. Jitendra Kumar Mishra he is working as AssociateProfessor and Head of the Department of Electronics and communication in PIES, Bhopal (RGPV). He has received Master of Technology and Bachelors of engineering respectively in Digital communication from BUIT, Bhopal and from RGPV, Bhopal. He has more than 10 years of teaching experience and publish 20+ papers in International journals, conferences etc. His areas of Interests are Antenna Wave Propagation, Digital Signal Processing, Wireless Communication, Image Processing etc.