IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 2 Issue 7, July 2015.
www.ijiset.com
ISSN 2348 – 7968

Implementation of UART using VHDL with BIST

Devshree Baghel[1] , Khemraj Deshmukh[2]

[1]Dept. Electronics and Telecommunication Engg.
Shri Shankaracharya Group of Institution,
Bhilai, Chhattisgarh, India

[2] Assistant Professor
Dept.Electronics and Instrumentation Engg.
Shri Shankaracharya Group of Institution,
Bhilai, Chhattisgarh, India

Abstract

Today's testing of VLSI chips are very much complex day by day due to increasing advancement of nano technology. So both front-end and back-end engineers are trying to evolve a system with full testability keeping in mind the possibility of reduced product failures and missed market opportunities. BIST is a testing technique that allows a system to test automatically itself. In this paper, the simulation result performance achieved by BIST enabled UART architecture through VHDL programming is shown. The BIST used here have two modes actually i.e. test mode and UART mode .This technique generate random test pattern automatically, so it can provide less test time compared to an externally applied test pattern and helps to achieve much more productivity at the end.

Keywords: *Built-In-Self-Test (BIST), Universal Asynchronous Receive Transmit (UART)*

1. Introduction

BIST technique has become as a boon, which helps to test a system automatically. Universal Asynchronous Receive/Transmit (UART) has the objectives of firstly to satisfy specified testability requirements and secondly to generate the lowest-cost with the highest performance implementation. UART is an important input/output tool for decades. The additional BIST circuit that increases the hardware overhead increases designs time and size of the chip, which may degrade the performance. [1]
This paper focuses on the design of a UART chip with embedded BIST architecture using VHDL language. The paper describes the problems of (VLSI) testing followed by the behavior of UART that includes both transmitter and Receiver section. [1]
Serial data is transmitted via its serial port. A serial port is most universal parts of a computer. It is a connector where serial line is attached and connected to peripheral devices such as mouse, modem , printer and even to another computer. In contrast to parallel communication, these peripheral devices communicate using a serial bit stream protocol (where data is sent one bit at a time). The serial port is connected to UART, an integrated circuit which handles the conversion between serial and parallel data. [3]

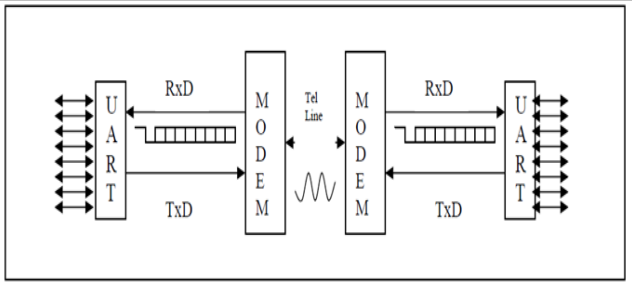

Fig 1: UART data transmission and receive

In this paper, the simulation result is compared with previous work and it has been seen that the result is promising and helps to reduce timing constraints and overall power dissipation.[1]
BIST has capability to test a circuit itself. BIST solution consists of a Test Pattern Generator (TPG), a circuit to be tested, a way to analyze the results as shown in figure below.

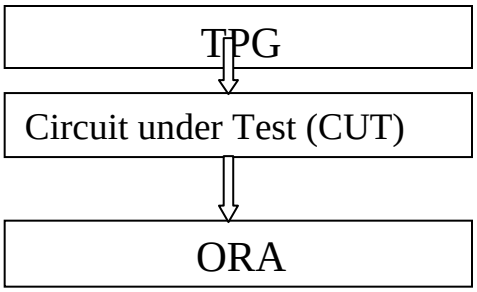

Fig 2: Basic BIST operation

ISSN 2348 – 7968

Where

TPG = Test Pattern Generator
ORA=Output Response Analyzer
BIST has become a major design consideration in Design-For-Testability (DFT) methods and is becoming increasingly important in today's state of the art SoCs.

2. Proposed Methodology

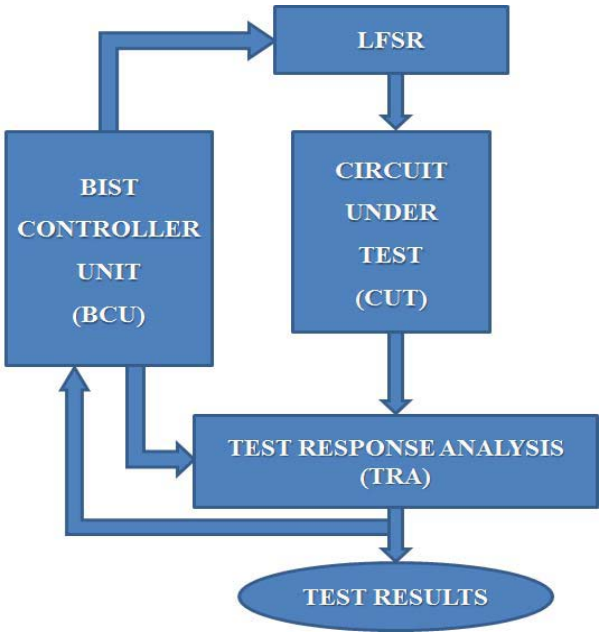

Fig 3: Proposed Methodology

The brief introductions of BIST architecture component are given below.

Circuit Under Test (CUT): It is the portion of the circuit tested in BIST mode. It can be sequential, combinational or a memory. It is delimited by their Primary Input (PI) and Primary Output (PO).

Test Pattern Generator (TPG): It generates patterns for the CUT. It is a dedicated circuit or a microprocessor. The patterns may be generated in pseudorandom or Deterministically.

Test Response Analysis (TRA): It analyses the value sequence on PO and compares it with the expected output.

BIST Controller Unit (BCU): It controls the test execution; it Manages the TPG, TRA and reconfigures the CUT and the multiplexer.[1]

2.1 Circuit Under Test (UART)

Universal Asynchronous Receiver Transmitter is an integrated circuit, which is used for transmitting and receiving data asynchronously via the serial port on the Computer . It contains a parallel-to-serial converter for data transmitted from the computer and a serial-to parallel converter for data coming in via the serial line.[3]

The UART also has a buffer for temporarily storing data from high-speed transmissions as shown in Figure 4.

Functionally UART consist of –
(i)CPU internal logic
(ii) Modem control section
(iii) Transmitter section
(iv) Receiver section.

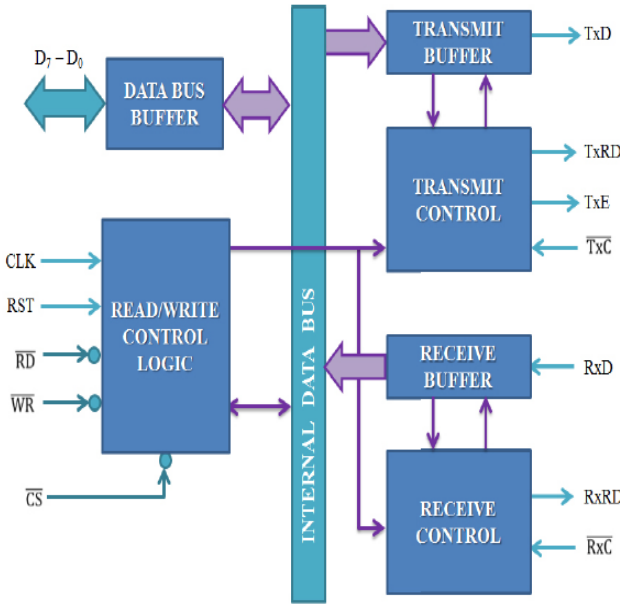

Fig 4: UART architecture block diagram

In addition to the basic job of converting data from parallel to serial for transmission and from serial to parallel on reception, a UART will usually provide additional circuits for signals that can be used to indicate the state of the transmission media and to regulate the flow of data in the event that the remote device is not prepared to accept more data. For example, the device connected to the UART is a modem. UART must have a larger internal buffer to store data coming from the modem until the CPU has time to process it. If the memory buffer used to store the data isn't large enough an overflow might occur. The size of the buffer depends on the design of the UART.[3]

ISSN 2348 – 7968

2.2 UART Transmitter Section

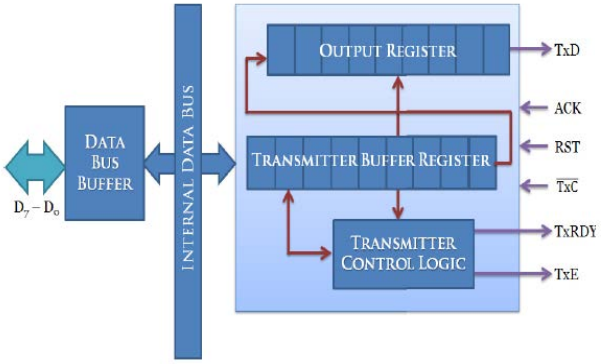

Fig 5: UART transmitter

UART transmitter module consists of an output register, a transmitter buffer register and transmitter control logic. The i/o pins are:

- TxD is the port for serial data output
- TxC bar is the transmitter clock input
- TxE is the transmitter control logic
- ACK & RST are two control inputs

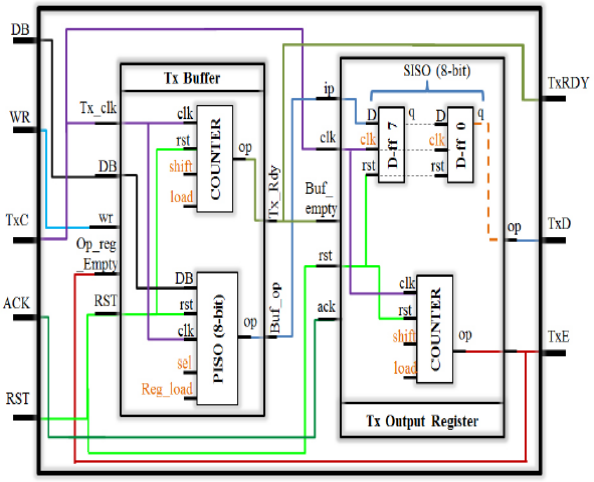

Fig 6: Internal architecture of UART transmitter

The PISO block shown in Fig 6 takes 2 input signals PISO_sel and Reg_load. The first one is used to derive third function of the block i.e. hold the data bits when output register is not ready-this is done just by deactivating the clock inside the block. The 2nd one i.e Reg_load is high when 8 data bits are loaded parallel in PISO register i.e. when output register is ready, the Counter_3_bits counts and all 8bits are shifted from PISO to Tx_Out_Reg.[1]

The 3-bit-counter "Counter_3_bit SISO" in Tx_Out_Reg i.e. the output register of the transmitter module counts 0 to 7 while transmitting the 8 bit serially outside the UART module. When all 8 bits are transmitted the output register is empty and ready to take next 8 bits from PISO register.[1]

2.3 UART Receiver Section

Figure 7 shows the architecture of UART receiver section. The followings are the brief description of each block-

Input Register: I/P register is a serial input and serial output register which consists of 8 D-flip flops to store and shift 8 bit input data. The input data of the register is the serially transmitted data from the transmitter.

Receiver buffer register: It is a 8 bit serial in parallel out register. The data of i/p register is shifted and enters bit by bit in every rising clock edge in Receiver Buffer Register. After entering 8 bit data, the buffer gets full.

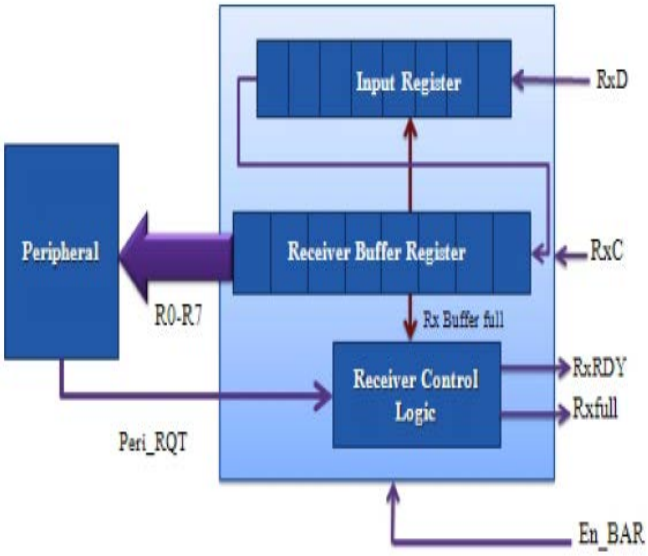

Fig 7: UART receiver section architecture

2.4 Memory BIST

Some of the advantages of a Memory BIST controller:

a. Flexibility to create different data backgrounds, addressing schemes, and test algorithms. The fault coverage depends heavily in using different combinations of data backgrounds and addressing schemes.[5]

ISSN 2348 – 7968

b. Flexibility to accommodate new test algorithms developed for newly identified defects. This helps in improving the test quality, which is especially useful for military, medical, and automotive applications.[5]

c. Helping failure analysis thereby expediting the yield learning period of a fabrication process.[5]

d. Allows better management of test time as different sets of algorithms can be applied at different phases of test, such as wafer sort, burn-in test, parametric, package, etc. [5]

3. Expected Output

The overall operation of UART is tested by using BIST. Our aim is to get high speed testing of circuit. The memory BIST is proposed for that purpose. Memory BIST have less testing time as compared to logic BIST and it also affects the overall area of the circuit.

4. Conclusion

In this paper, we have described UART architecture tested by memory BIST. This supports a wider variety of algorithms, is capable of running BIST at full-speed, and is modular in nature for optimizing area overhead versus flexibility during the design phase .UART transmitter and receiver is designed with less area and testing time required for that is also less. The proposed architecture is not only more flexible but also can be extended easily to incorporate complex data and addressing schemes that may be required to support complex algorithms in the future.

5. References

[1] Tapas Tewary, Atanu Sen, A novel approach to realize Built-in-self-test (BIST) Enabled UART using VHDL ECE department of Academy of Technology IEEE, Howrah, West Bengal.

[2] Mohd Yamani Idna Idris, Mashkuri Yaacob, Zaidi Razak “A VHDL Implementation of UART Design with BIST Capability” Faculty of Computer Science and Information Technology University of Malaya Kuala Lumpur.

[3] J. Norhuzaimin and H.H Maimun , The Design of High Speed UART Department Of Electronic, Faculty Of Engineering.

[4] Himanshu Patel, Sanjay Trivedi, R. Neelkanthan, V. R. Gujraty, A Robust UART Architecture Based on Recursive Running Sum Filter for Better Noise Performance, Space Applications Centre (ISRO), Ahmedabad,Gujarat

[5] Xiaogang Du, Nilanjan Mukherjee, Wu-Tung Cheng, Full-Speed Field-Programmable Memory BIST Architecture ECE Dept., the University of Iowa, Iowa City, IA 52242

[6] Neil H.E. Weste, Kim Haase, David Harris , A. Banerjee , “CMOS VLSI Design: A circuits and Systems Perspective”, Pearson Education.

[7] K. Zarrineh, and S. J. Upadhyaya, “On programmable memory built-in self test architectures”, Design, Automation and Test in Europe Conference and Exhibition 1999. Proceedings , 1999, pp. 708 -713.

[8] M.S. Harvey,Generic UART Manual,Silicon Valley,December 1999

[9] Mentor Graphics Corporation, “MBIST Architect Process Guide, V8.2004_4”, August 2004

[10] Kamran Zarrineh and Shambhu J. Upadhyaya, “On Programmable Memory Built-In Self-Test Architecture”, In Proceedings of Design, Automation and Test in Europe 1999.