ISSN: 2456-043X

International Journal of Engineering Science and Generic Research (IJESAR)
Available Online at www.ijesar.in
Volume 2; Issue 5; September-October; 2016; Page No. 95-99

Design of SPI Bus Protocol with Built-In-Self-Test using CA

GANTI PRAVEEN ANAND [1], N.LATHA [2]

[1] PG Scholar, Department of ECE, Brilliant Institute of Engineering and Technology, Hyderabad, India

[2] Assistant Professor, Department of ECE, Brilliant Institute of Engineering and Technology, Hyderabad, India

Abstract

The Serial-Peripheral Interface (SPI) protocol is one of the important bus protocols for connecting with peripheral devices form microprocessor. The complexity of the circuits has aroused with the enormous advancement of IC technology. So, in order to lessen the product failure self-testability in hardware is demanded a lot in recent times. The necessity of self-testability will lead to a solution called Built-in-self-test (BIST). BIST is an effective solution to reduce the huge circuit testing cost. This paper represents design of SPI protocol with BIST using CA. Cellular automata pattern generator used instead of LFSR for high randomness To accomplish compact, stable and reliable data transmission, the SPI is designed with Verilog HDL

Keywords: Serial-Peripheral Interface; Embedded built-in self-test architecture; Verilog HDL;

Introduction

SPI or Serial-Peripheral Interface is a worldwide accepted standard communication protocol. SPI protocol was invented by Motorola. SPI protocol is considered as one of the very best among the systems that are connected to a number of devices and make the communication smooth and fast. SPI as well as others serial protocols such as I2C and 1-wire for instance, are well fitted for data communications from integrated circuits for low or medium data transfer speed to peripherals which are on chip board [1].Several works have been done using VHDL in designing SPI. A comparison between SPI and I2C Implementation over FPGA is shown in [2]. On that paper, a comparative study of those two protocols on FPGA platform is presented and the entire design has been coded in VHDL. For various controlling purposes SPI is implemented. SPI is presented for motion controller in [3]. FPGA Implementation of SPI of Flex Ray Controller is presented in [4]. This paper emphasizes on a new approach of designing SPI with embedded BIST capability using Field Programmable Gate Array (FPGA) technology. Testing of a circuit has become increasingly tough as the scale of integration grows. SPI with the BIST capability provides the specified testability requisites and lowest-price with the highest performance implementation. Much lesser blocks and modules are used to design this SPI so that the testing complexity can be reduced. The SPI protocol architecture, implementation technique of the system, circuit schematic and simulation results will be discussed briefly in the following sections. The system demands of high integration, low bit error rate and low cost can be satisfied by this SPI.

II. SPI PROTOCOL ARCHITECHTURE

In this paper, the SPI protocol implementation uses four logic signals: SCLK, MOSI (Master Output-Slave Input), MISO (Master Input-Slave Output), SS (Slave Select).SCLK is the clock, a unidirectional bus, which fed into the slave devices. MOSI is defined as output from master which is also known as serial data out. MISO is defined as output from slave which is also known as serial data in. SS is an active low signal which is used to select the slave devices. A full duplex data transmission is occurred in SPI clock cycle. Fig. 1 shows the data transfer system of SPI. To transfer a data from master device to slave, there arethree types of data formats required. Fig. 2 shows the data format of SPI protocol.

Corresponding author: GANTI PRAVEEN ANAND

Fig. 1 Data Transfer Type of SPI

A. Control Address

it is 8 bit data format. In Fig 2 the first 0 to 7 bits represent the control bus. SPIE is interrupt enable signal which enable the SPI interrupt flag. SPE is the bit for enabling SPI. DORD is used to determine the data order. If DORD is 0 then LSB will be transmitted first. MSTR is used to select the master or slave mode. CPOL & CPHA are clock polarity & clock phase used for determine the shifted edges of MISO & MOSI data.SPR1 & SPR0 are used to determine the clock rate.

B. Status Address

it is also 8 bit data format. In Fig 2 the 8 to 15 number bits represent the status bus. Here, SPIF is the bit used to determine the serial transfer. WCOL is for determine the collision of transfer. Bit 10 to 14 is reserved bit. SPI2X is used to double the clock speed.

III. PROPOSED ARCHITECTURE

The Proposed structure consists of two modes. One is BIST mode where the SPI can test itself. Another is normal mode. In normal mode the device works like usual SPI protocol.

A. BISTModule

Built-In-Self-Test (BIST) is a design technique where a circuit can test itself. This technique can be easily used in various devices like combinational and sequential logic, memories, multipliers, and other embedded logic blocks. Advanced chip or SOC design is incorporated with large number of core blocks. This is very much difficult to access these chips. So, it is a great challenge to test such embedded chips from outside. Some main challenges among them are the extra testing equipment, cost of testing, level of testing and the testing speed. All these main challenges can be solved by using BIST. The main feature of the BIST system is it gives high speed testing and it can be tested at different test levels. Moreover, no expensive test equipment is needed. Since, BIST is far cheaper than conventional system [5], [7].Fig. 3 shows the structure of the SPI with BIST. The BIST control signal controls the BIST module. In the BIST module, there are four sub blocks. They are three random pattern generators and a comparator.

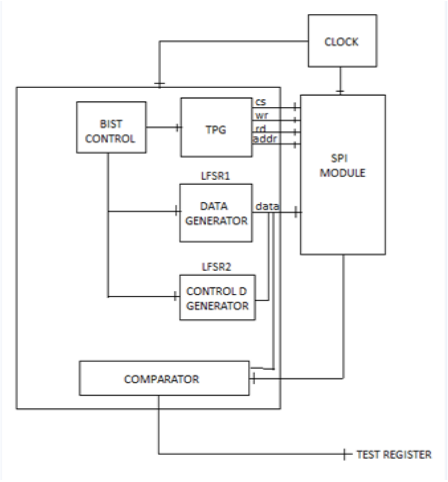


Fig. 2. BIST Structure

1) Random Pattern Generators (RPG):

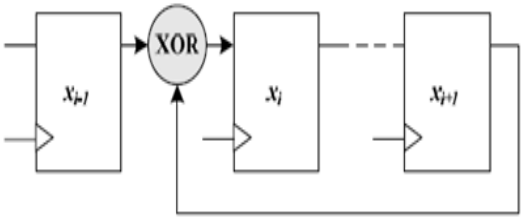
Random Pattern Generator (RPG) generates random patterns which can be used for the verification of device like SPI. The RPG is a part of the BIST in the verification of the circuits. Many methods have been proposed for the BIST equipment design [6], [9].To produce bytes to test the circuit the method of a random pattern generator (RPG) are used. This RPG consists of two LFSRs. LFSR 2 is used to generate the control address. LFSR 1 gives the data. The generated bytes are used directly in the main SPI to obtain better fault coverage. A comparator evaluates the response of the SPI with these bytes.

Proposed design consist the Cellular Automata is used. Instead of LFSR, CA is used LFSRs are most commonly used to build TPGs, but recently there has been interest in CA for test pattern generation. CA generates test vectors which are more random in nature. Highly random vectors help in detection of faults such as the stuck-open faults, delay faults etc. which cannot be easily detected by vectors generated by LFSR.

Cellular Automata (CA):

Cellular Automata (CA) consists of a collection of cells/nodes formed by flip-flops which are logically related to their nearest neighbors using XOR gates . When the value of a node is deter-mined only by two neighboring cells the CA is known as one-dimensional linear CA (for the rest of the text one-dimensional linear CA is referred as a CA). The logical relations which relate a node to its neighbors are known as rules and they de-fine the characteristics of a CA. There are many rules which can be used to construct

a CA register, the most popular being rules 90 illustrated in Figure.



Rule 90

$x_i(t+1) = x_{i-1}(t) \oplus x_{i+1}(t)$

Fig3 : *CA 90 Rule*

The CA pattern generator is designed by using CA rule 90, which generate the random values in TPG for SPI; the CA is shown in figure 3.

2) Comparator:

This is a comparator which is used to compare the received and transmitted bit pattern. And then it gives the value of error. If the comparator gives bit stream of 101 then the device is perfect and running good. If it gives 001 then there are some faults occur in the protocol.

B. SPIStructure

Fig. 4 shows the basic SPI structure. In Fig 4 it is depicted that there are three registers. They are Control, Status & Data registers. Data register is a shift register. Here, as the data goes from master to slave it is Serial Data Out (SDO) for master and Serial Data In (SDI) for slave. And when the slave register is full, it starts to transmit data to master. And then SDO and SDI are reversed. Master Clock Generator generates the clock and gives it to the slave. The Slave Select Decoder is a decoder controlled by the control register. This slave select decoder selects the slave devices when multiple peripheral devices are needed to be connected. Table I demonstrates the operating modes with reset and reset_n signal. In the table it is seen that, BIST mode is on when reset pin is set low i.e. 0 & reset_n is 1 and vice versa.

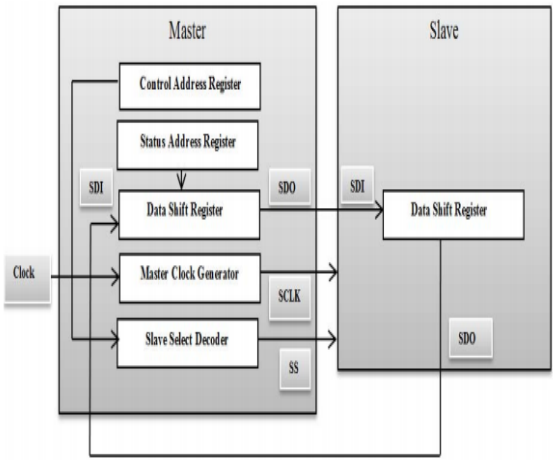


Fig. 4. SPI Module Architecture

TABLE I. OPERATING MODES OF SPI

reset_n	reset	BIST Mode	NORMAL Mode
0	1	OFF	ON
1	0	ON	OFF

In Fig. 4 the architecture of SPI module is depicted. Here, it is seen that the master is consists of five main modules. Here, three registers are described broadly in previous section. The slave select decoder is used to select the peripheral devices. In Table I, the two main modes are shown. It is shown that when reset = 1 and reset_n = 0, then normal mode begins and master starts to communicate with its slave devices. Afterwards, when reset = 0 and reset_n = 1, then the BIST mode is turned on and the circuit tests itself.

III.SIMULATION RESULTS

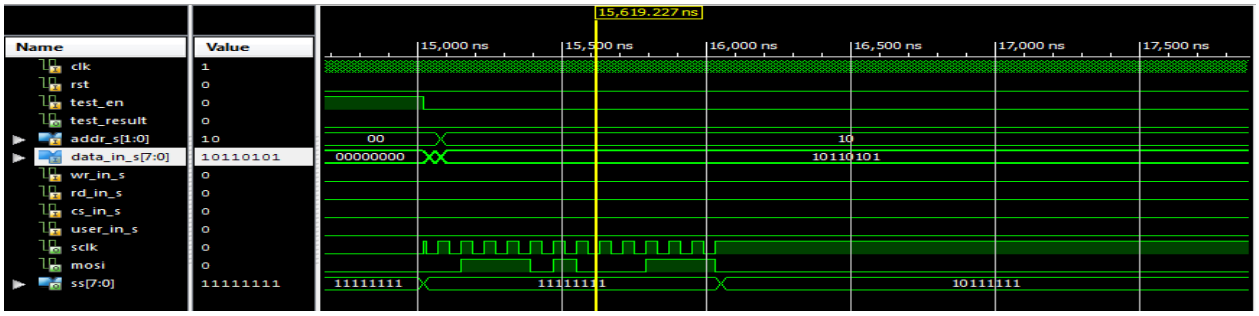


Fig 5: SPI mode results

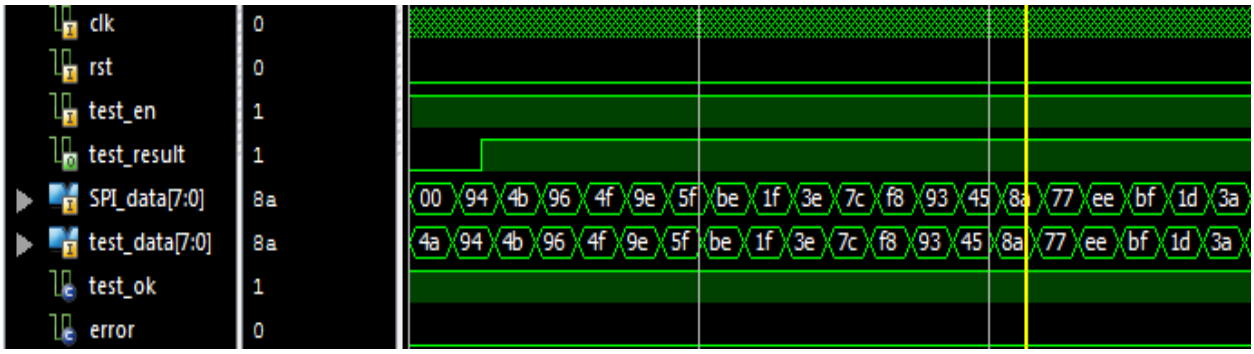


Fig 6: SPI BIST mode

IV.CONCLUSION

In this paper, an CA based SPI with BIST capability is presented. Here all the modules are designed and simulated with Verilog HDL. Then the system is downloaded in the Xilinx Spartan-3E FPGA (XC3S500E). This SPI is much more flexible, speedy, low cost, and stable with respect to conventional one. This SPI control bus architecture can enable the industrial fabrication of chip in a way where only a pressing of one switch can test itself. So that, it would save valuable time and cost of testing circuits significantly.

V.REFERENCES

1. F. Leens, "An introduction to I2C and SPI protocols," IEEE Instrumentation & Measurement Magazine, vol.12, no.1, pp.8-13, February 2009.
2. K. Oudjida, M. L. Berrandjia, R. Tiar, A. Liacha, K. Tahraoui,"FPGA implementation of I2C & SPI protocols: A comparative study, "in Proc. *16th IEEE International Conference on Electronics, Circuits, and Systems,* pp.507- 510, Dec. 2009.
3. N.Q. B. M. Noor and A. Saparon, "FPGA implementation of high speed serial peripheral interface for motion controller," in Proc. *2012 IEEE Symposium on Industrial Electronics and Applications (ISIEA),* pp.78-83, Sept. 2012.
4. A.N. Gaidhane and M.P. Khorgade, "FPGA Implementation of Serial Peripheral Interface of Flex Ray Controller," in Proc. *13th International Conference on Computer Modeling and Simulation (UKSim),* pp.128-132, Apr. 2011.
5. M. Bushnell and V.D. Agarwal, *Essentials of electronic testing for digital, memory and mixed-signal VLSI circuits*, Kluwer Academic Publishers, 2000.
6. S. Jamuna and V.K. Agrawal, "Implementation of BIST structure using VHDL for VLSI circuits," *International Journal of Engineering Science and Technology,* vol. 3, no. 6, pp. 5041-5048, June 2011.
7. V.K. Agrawal, C.R. Kime, K.K., Saluja, "A tutorial on BIST, part 1:Principles," *IEEE Design & Test of Computers*, vol. 10, No.1, pp.73-83,March 1993.
8. J. Bhasker, *Verilog® HDL synthesis: a practical primer*, Star Galaxy Publishing, 1998.
9. M.Y.I. Idris and M. Yaacob, "A VHDL implementation of BIST technique in UART design," in Proc. *Conference on Convergent*

Technologies for the Asia-Pacific Region (TENCON), pp.1450-1454, Oct. 2003.

10. S. Saha, M. A. Rahman, A. Thakur, "Design and Implementation of a BIST Embedded High Speed RS-422 Utilized UART over FPGA," in Proc. *2013 Fourth International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, pp. 1-5, Jul.2013.

11. S. Saha, M. A. Rahman, A. Thakur, "Design and Implementation of a BIST Embedded Inter-Integrated Circuit Bus Protocol over FPGA," in Proc. *2013 International Conference on Electrical Information and Communication Technology (EICT)*, pp. 1-5, Feb. 2014.

AUTHOR DETAILS

	**GANTI PRAVEEN ANAND** has completed his B.Tech in Electronics and Communication Engineering from Arjun College of Technology & sciences, J.N.T.U.H Affiliated College in 2014. He is pursuing his M.Tech in VLSI System Design from Brilliant Institute of Engineering & Technology, J.N.T.U.H Affiliated College.
	**N.Latha** is an Assistant Professor at Brilliant Institute of Engineering and Technology, Hyderabad in ECE Department. She received her B.Tech degree in Electronics and Communication Engineering from Srivenkateshwara Engineering collage, Hyderabad and M.Tech degree in Embedded Systems from Brilliant Institute of Engineering and Technology, Hyderabad. Her research interest is Embedded Systems.