DESIGN OF BUILT IN SELF TEST EMBEDDED MASTER SLAVE COMMUNICATION USING SPI PROTOCOL

[1]Thummala Reddisekhar, [2]P. Giri Babu
[1]PG Scholar, Dept. of ECE, Sree Rama Engineering College, Tirupati, Andhra Pradesh 517507.
[2]Associate Professor, Dept. of ECE, Sree Rama Engineering College, Tirupati, Andhra Pradesh 517507.
[1]sekharns494@gmail.com, [2]giribabu.pantra @gmail.com

Abstract

The Serial-Peripheral Interface (SPI) Protocol also called as synchronous serial interface specification is used for communication between single master and single/multiple slaves. With the increase in number of slaves causing high complexity of circuit creates a demand in self-testability feature for SPI module in order to test for fault free circuits. Built-In-Self-Test (BIST) is the answer for self-test in circuits as well as it helps in reduction of maintenance and testing cost. Design of BIST embedded SPI module with Single Master and Single Slave configuration has been introduced in this paper, here 8-bit data is transferred across the module, where the circuit under test (CUT) is being self-tested with BIST feature for its correctness. This SPI module is designed using Verilog Hardware Description Language (HDL) using EDA playground platform for applications like Application Specific Integrated Circuit (ASIC)or System on Chip (SOC).

Keywords- BIST, Memory, Verilog HDL, SPI, SOC.

I. INTRODUCTION

Motorola invented Serial Peripheral Interface (SPI) protocol in the mid 1980's to substitute parallel interfaces and provide high speed transfer of data between modules. SPI become most favored serial communication protocol because of easy interfacing and high-speed transfer. SPI follows full duplex, master-slave communication while transferring and receiving data between them which synchronizes on the rising edge or falling edge of the clock. Data transmission can happen at the same time for both master and slave. There are basically two types of SPI interface, it can be either 3-wired or 4-wired. This research focuses on the popular 4-wired SPI interface.

SPI Protocol is followed even in embedded systems like soc processors and microcontrollers like Programmable Interface Controller (PIC), Advanced Virtual Risc (AVR).

These chips may work as master or slave block which is operated by the inbuilt SPI controller in them. SPI is mostly used in those applications where high-speed data transfer has to take place. (1) SPI's exclusive features include master/slave operation, double buffered data register for transmission and reception, polarity and phase synchronization with serial clock, interrupt capability of CPU with fault deduction. SPI's functional registers are responsible for their recognition.

As SPI is an interfacing protocol. The word "Interface" can be elucidated as a common boundary shared by two different components of computer's software or hardware parts in order to exchange information between them. On the contrary, "protocols" are primarily stated as set of rules that are not related to communications of any sort rather bartering of information across blocks to commune and acknowledge each other is what interfacing protocol stands for (4)

II. DESIGN AND CONTROL

The SPI is communication protocol used for single-master that has one central device activates all the communications with the slaves. The SPI master performs the data sending to a slave and/or requests information from it. The SPI also picks slave by pulling the respective select signal (SS) line. The master generates information onto Master Out-Slave In (MOSI) line while it samples the Master In-Slave Out (MISO) line [66,70,71,86,87].

SPI is a protocol on 4 signal lines [48]. The interfacing between master and slave is shown in the figure.

- SCLK, directs from the bus master towards all slaves; all the SPI signals are synchronized with this signal;
- "SSN" is used to select particular slave to which the master wish to communicate;
- MOSI is a data line from the master to the slaves.
- MISO is a data line from the slaves to the master [48,87].

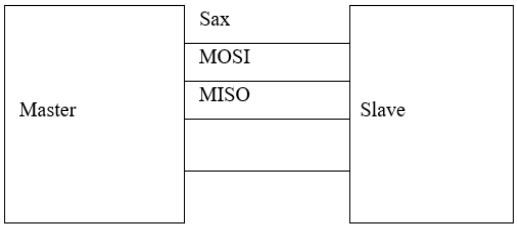

Figure: Interface between SPI Master and Slave

All the device communication happening through SPI protocol are in a master-slave arrangement. Master is the device which helps in generating the main SPI clock with respect to which the data's are synchronized. The frequencies supported by the SPI are much higher than other serial protocols. SPI protocol can only have a single master but in case of slave configuration single to more than one slaves can be connected. Above figure shows connection between a typical SPI Master and SPI Slave blocks. Then each slave block will have it's own chip select signal and the signal will be turned high for disconnecting from the master block. There are two data lines for SPI block i.e. MOSI (Master Out Slave In-signal from master to slave) and MISO(Master In Slave Out-signal from slave to master).

BIST Architecture

Built-In Self-Test (BIST), as the name suggests is a technique in which the circuit is capable of testing itself. It reduces testing and maintenance cost and also reduces cost of automatic test pattern generation (ATPG). the BIST system hierarchy and all three levels of packaging mentioned earlier. The system has several PCBs, each of which, in turn, has multiple chips. The system Test Controller can activate self-test simultaneously on all PCBs. Each Test Controller on each PCB can activate selftest on all chips on the PCB. The Test Controller on a chip executes self-test for that chip, and then transmits the result to the PCB Test Controller, which accumulates test results from all chips on the board and sends the results to the system Test Controller.

The system Test Controller uses all of these results to isolate faulty chips and boards. System diagnosis is effective only if the self-test procedures are thorough. For BIST, fault coverage is a major issue. Other issues are chip area overhead, its impact on chip yield, the cost of the additional chip pins required for test, the performance Penalty in terms of added circuit delay, and extra power requirements. For BIST, the test engineer frequently, but not always, modifies the chip logic to make all latches and flip-flops controllable, perhaps by using the scan technique.

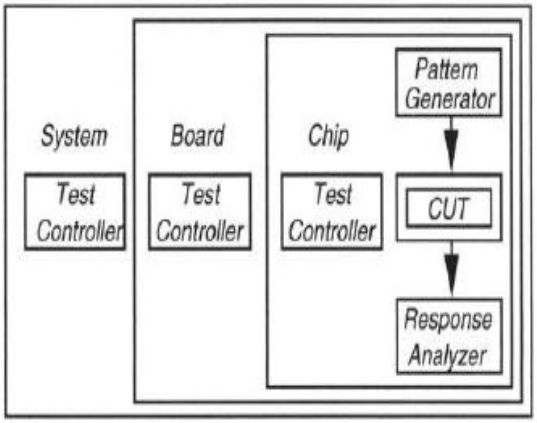

Figure: BIST hierarchy

- **BIST Pattern Generation:** The following hardware pattern generation approaches have been used. 2.1.1 ROM. One method is to store a good test-pattern set (from an ATPG program) in a ROM on the chip, but this is prohibitively expensive in chip area, and will not be discussed further.

- **LFSR:** Another method is to use a linear feedback shift register (LFSR) to generate pseudo-random tests. This frequently requires a sequence of 1 million or more tests to obtain high fault coverage's, but the method uses very little hardware and is currently the preferred BIST pattern generation method.

- **Binary Counters:** A binary counter can generate an exhaustive test sequence, but this can use too much test time if the

number of inputs is huge. For example, with 64 inputs and the test-pattern generator clocked at 100 MHz, this takes 51,240,955.8 hours of test time to generate all 264 patterns, which is impractical. Therefore, this type of pattern generator must be partitioned. Also, the binary counter requires more hardware than the typical LFSR pattern generator.

- **Modified Counters:** Modified counters have also been successful as test-pattern generators, but they also require long test sequences. **LFSR and ROM:** One of the most effective approaches is to use an LFSR as the primary test mode, and then generate test-patterns with an ATPG program for the faults that are missed by the LFSR sequence. These few additional test-patterns can either be stored in a small ROM on the chip for a second test epoch, they can be embedded in the output of the LFSR, or they can be embedded in a scan chain in order to augment the stuck-fault coverage to100%.

III. EXISITING SYSTEM

i2c Protocol Architecture

The i2c protocol architecture shows that the i2c protocol uses two main buses: data bus and clock. Data bus is a bidirectional pathway which can move information from master to slave and the other way around Clock is a unidirectional pathway which goes from the master to the slave gadgets. The clock is first generated by master device and initiates a data transfer. Slave at that point gives back an acknowledgement signal, shows that the exchange of data mainly involves three components:

- **Slave Address:** By this address, the slave device address to which the data must be transmitted is determined by the master. It is 8 bit long.
- **Word Address:** By this address, the address of the data is conveyed by the master to the slave. It is also 8 bit long
- **Data Value:** The master transmits the information to the slave in the form of data value. Data values consist of 8 bits.

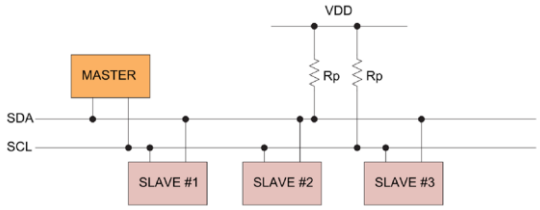

Figure: i2C Architecture

III. PROPORSED SYSTEM

In this research an SPI module having self-testing capability has been introduced. The basic components of BIST architecture has been introduced within SPI architecture block diagram where the designed CUT is able to self-test itself. The fig3 is the block diagram for BIST embedded SPI protocol , where the test patterns are generated by the TPG block and they are send through Master block's MOSI pin to the Slave.

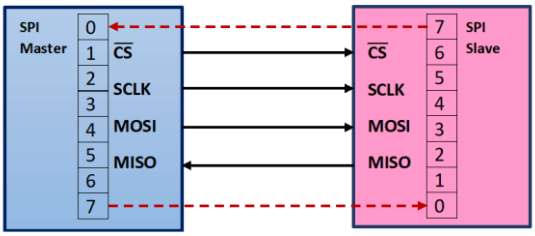

Figure: SPI with Master-Slave configuration

Where slave reads the data sent from the master and segregates the received data and then finally passes to the CUT designed within the SPI slave block for performing the operation.

After completion of the arithmetic operation. The results from the CUT are sent from the Slave block to the Master block through MISO pin .The received ALU results is then fed to the ORA block designed within the SPI Master for checking the correctness of the CUT.

a) Test Pattern Generator (TPG):

The TPG block proposed here uses LFSR (linear feedback shift register) technique to generate test patterns. Linear feedback shift register is also called as pseudo random pattern generator. An n bit LFSR has maximum sequence length of 2^n-1.

For example if we take n=4 bit LFSR. The test patterns generation can be explained from the

figure.

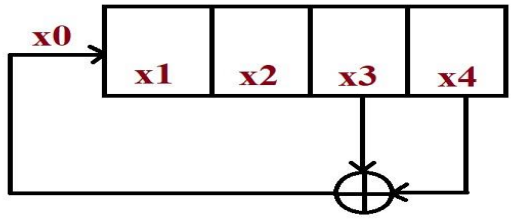

Figure: Test Pattern Generator (TPG)

b) ORA

The ORA block presented in this research uses MISR (Multiple Input-Serial-Register) technique for compaction of the responses coming from the CUT. The compacted response is then compared to the golden signature which determines if there is any fault present or not. MISR operation can be explained from figure.

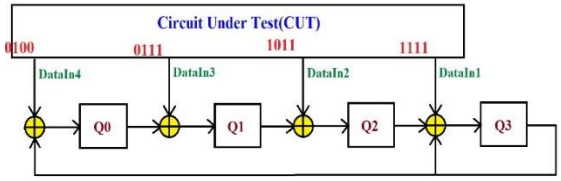

Figure: Circuit Under Test (CUT)

IV. SIMULATION RESULTS

The entire simulation and result obtaining using test vectors are done with the help of Xilinx software.

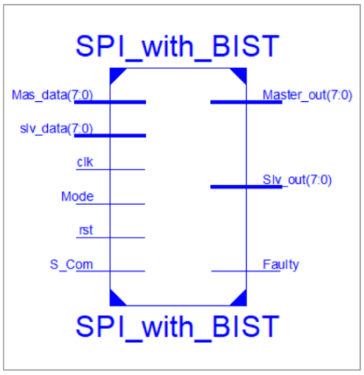

Figure: SPI with BIST block

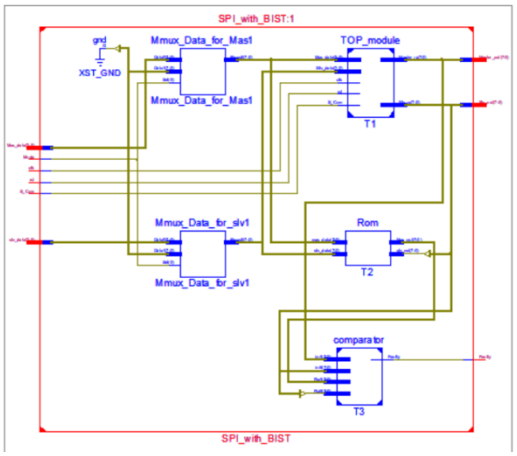

Figure: SPI with internal block

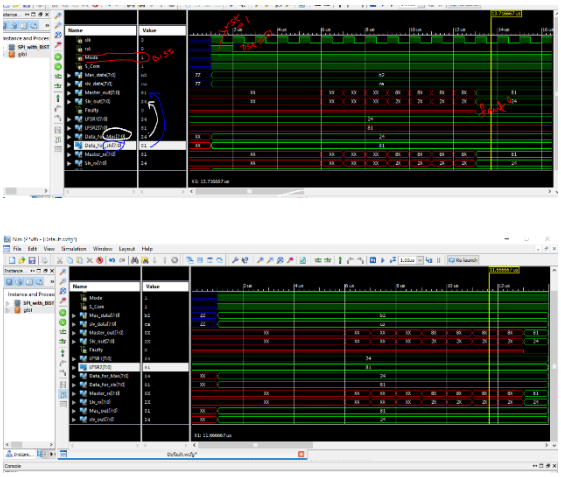

Figure: Proposed System Outputs

V. CONCLUSION

In this research, a BIST embedded SPI protocol with master-slave configuration has been designed successfully using Verilog HDL and simulated using Cadence Xcelium 20.09 in EDA Playground Platform. The self-testability feature of BIST has been well exercised in the proposed Model. Where, the designed CUT uses SPI protocol for data transfer and has the capability to test itself for checking the correctness of the Circuit under test.

REFERENCES

1. Choudhury, G.K.Singh, R.M.Mehra, “Design and Verification Serial Peripheral Interface (SPI) Protocol for Low Power Applications”, International Journal of Innovative Research in Science, Engineering and Technology, Vol. 3, Issue 10, October 2014

2. Vineeth B, Dr. B. Bala Tripura Sundari, "UVM Based Testbench Architecture for Coverage Driven Functional Verification of SPI Protocol", 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)

3. PallaviPolsani, V. Priyanka B., Y. Padma Sai, "Design & Verification of Serial Peripheral Interface (SPI) Protocol", International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8 Issue-6, March 2020

4. Muhammad Hafeez, Azilah Saparon, "IP Core of Serial Peripheral Interface (SPI) with AMBA APB Interface", IEEE 9th Symposium on Computer Applications & Industrial Electronics (ISCAIE),2019

5. Shumit Saha, Md. Ashikur Rahman, Amit Thakur, "Design and Implementation of SPI Bus Protocol with Built-In-Self-Test Capability over FPGA" International Conference on Electrical Engineering and Information & Communication Technology (ICEEICT) 2014

6. Deepika, Jayanthi K Murthy "Interrupt Enabled Priority Based Master Slave Communication using SPI Protocol", International Journal of Innovative Technology and Exploring Engineering (IJITEE) , ISSN:2278-3075, Volume-9 Issue-9, July 2020

7. Xingchun Liu,Yandan Liu "Multi-functional Serial Communication Interface Design Based on FPGA", 3rd IEEE International Conference on Computer and Communications,2017

8. Bhagyashri, G. Allawagol, Vidyashree AC, Narendra Kumar, "Design of SPI IP to communicate with I2C bus of a microcontroller" International Journal of Industrial Electronics and Electrical Engineering, ISSN: 2347-6982, Special Issue, Sep.-2016

9. Xiaole Cui, Miaomiao Zhang, Qiujun Lin, Xiaoxin Cui, Anqi Pang, "Design and Test of the In-Array Build-In Self-Test Scheme for the Embedded RRAM Array",IEEE ELECTRON DEVICE LETTERS, VOL. 38, NO. 5, MAY 2017

10. Abhas Singh, Gurram Mahanth Kumar, Abhijit Aasti, "Controller Architecture for Memory BIST Algorithms" IEEE International Students' Conference on Electrical, Electronics and Computer Science,MAY 20202021 3rd International