A Testable Design Method for Memories by Boundary Scan Technique

Guo-Hui QIAO[1], Zhong-Liang DENG[1], Ke HAN[1], Dong-Yang YAO[1] and Zhi-Heng YUE[1]

[1] *School of Electronic Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China*

Abstract: This paper presents a design for test the embedded flash in an object System-on-a-chip (SoC). The feature of the Flash TAP (Test Access Port) complies with the IEEE std.1149.1, and it can select different scan chains and other control registers for other test. By the trade-off between the test time and the circuit area, an IST (In System Test) circuit is designed in the SoC. Experiment results on the embedded memory have shown that the proposed method costs small testing timing by the use of IST.

1 Introduction

The increasingly complex design of SoC (System On a Chip) causes not only the chip area increases, and the testability of circuit and system also declines sharply[1]. The test time account for a bigger and bigger proportion of the whole SoC design time, the regular testing methods is facing serious test difficulty. With the design and test cycle shortens, combine the test and design to complete the DFT (Design for Testability) has become an inevitable trend[2]. As a kind of important technique of testable design, boundary scan technique can not only test the test function of entire SoC or PCB, it can also test the connection between each module.

In 1990, IEEE and JTAG (Joint Test Action Group) worked out the boundary scan standard of JTAG[3]. namely IEEE1149.1 standard. This specification provides a complete and standardized method of testable design[4], so it gets the support of the world's most integrated circuit manufacturers and test contractor, such as ARM Company's ARM7TDMI processor[5], Xilinx company's series of FPGA above XC3000.

In this paper, we discussed a testable design method based JTAG boundary scan technique to complete the test of SoC's Flash module. Using this method, it'll be convenient to test the storage control circuit and the quantity of external testing interface will be reduced. The standard test port makes the testing mechanism standardized, thus to improve test efficiency, reduce product development cycle and test cost.

2 Analysis of Common test Structures

In order to test IP (Intellectual Property) core, the common test structure design is shown in Fig. 1, and normally is composed of Source, Sink, TAM (Test Access Mechanism), Wrapper. Source and Sink, which are responsible for generating test stimulus and receive test response respectively, can be designed as required to be internal or external parts of SoC. TAM is a set of methods designed for the delivery of test data between exterior and SoC's internal IP ports. As shown in Fig. 1, test data is delivered from Source to IP ports and from IP ports to Sink through TAM. Wrapper is the interface between IP ports and TAM. There are two main functions of Wrapper: offering appropriate test access to load stimulus and capture response; insulating the pending IP cores and the non-tested IP cores from the testing IP cores.

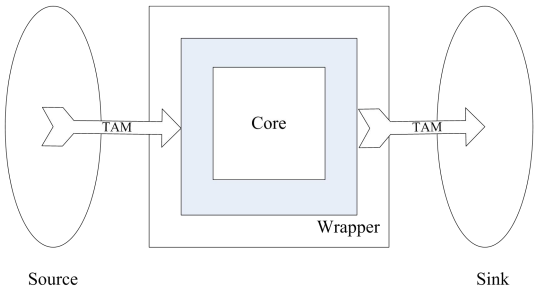

Fig. 1. IP Core Test Structure.

As for the TAM, the wider bandwidth, the faster test vectors' interaction, and the higher efficiency. The bandwidth is restricted by the design of SoC, such as the

bus width, the extra area and time of the testable design, and the number of top ports. TAM normally includes direct access mechanism, IP core transparency access mechanism, boundary scan mechanism and bus based access mechanism.

2.1 The Test Structure Design of the Direct Access Mechanism

The simplest way to test SoC internal IP models is using pin mux to access the internal IP's test ports after choosing external test mode[6].

This method under normal working mode also needs to configure different test modes for the completion of different IP tests. There are three working modes in this direct accessing strategy for each IP model: user mode is the normal working mode; inactive test mode indicates this IP model is being insulated while another IP model is under testing. The output driven is configured as inactive state, and the bidirectional port is set as input state. Active test mode indicates this IP model's ports are linked to exterior directly for external test and observation while other IP models are inactive and cannot be tested.

Although the test structure of the direct access mechanism is simple, its shortcomings cannot be ignored. First, this mechanism is not suitable for the external testing interface with huge number of ports. Besides, the interconnect test between IP models is not supported. In other words, testing a single IP model through this mechanism is available while checking the performance between models is not. Thus, this mechanism cannot meet the requirement for all IP models test. And as the design of SoC is becoming more and more complicated, SoC will contain more and more IP cores. If every IP core is required to be tested, then the cost of area has to be increased.

2.2 The Test Structure Design of the IP Core Transparency Access Mechanism

In this test access mechanism, the non-tested IP cores are transparent, which means the test vectors will not be changed when they pass through these non-tested IP cores. It is inevitable for some test vectors to pass through non-tested IP cores when transmited from exterior to testing IP cores or from testing IP cores to chip's output pins.

This test access mechanism is proposed by F. Bouwman in 1992[7]. F. Bouwman proposed a testable design method to test the macro models on a chip. This method bases on boundary scan technology and uses models' function information to simplify the test of circuits.

The simplest design of transparency strategy is shown in Fig. 2. The solution is placing a mutiplexer between IP core's input port and output port, which is a direct method to block the influence of inner logic on test vector, implementing the transparency.

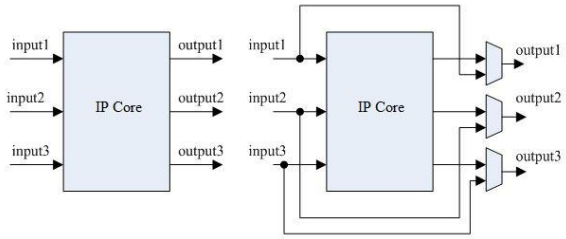

Fig. 2. Transparent IP Core with Multiplexer.

The design of this method is simple but still not available when facing with SoC's complex internal structure. The number of IP core's interface is too large to afford because too many mutiplexers are needed. Besides, the cabling between input and output is also very complicated.

In 1996, I. Ghosh proposed a structurized IP core transparency TAM strategy which is improved afterwards. Altough similar with hierarchical bounary scan technology, H-SCAN (High Level Alternative to Full-Scan) technology is outstanding in low-cost of area and short-time of test vectors' carryout. This method is triggerd by SoC's existing mutiplexers and triggers. Scanning path is constructed by combining these two components. Through the choice of the mutiplexer, non-texted IP cores enter into tranparency state, and the pending IP core starts testing.

In Fig. 3, there are two modes, user mode and H-SCAN mode, in a transparent core. The mode is selected by TEST. REG A and REG B both are 16-bit registers, and there is a 16-bit mutiplexer between them. The REG A, REG B and mutiplexer compose a parallel scan path. One port of mutiplexer is used as the channel for test data. In the user mode, two registers are isolated. In the H-SCAN mode, the scan path is constructed.

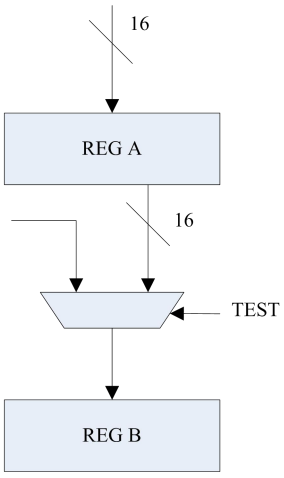

Fig. 3. The Structure of H-SCAN.

Since reusing the mutiplexers on the chip, H-SCAN can reduce the cost of area. In the meantime, the test efficiency is increased due to muti-scanning path. However, this

structure relies on the SoC's test logic structure. If there is no mutiplexer in the test logic structure, it is necessary to add.

The test access mechanisms above have kinds of limitations. However, the most widely used method is the JTAG testable design. Because of its early relevant standards, this method has great value in reusability.

3 System Structure of JTAG Testable Design

The boundary scan standard of JTAG defines a kind of standard boundary scan structure and testing interface, the basic idea is to insert a shift register unit between external I/O port and internal logic circuit[8]. These shift register units distribute in the boundary of the chip, so called boundary scan register (Boundary-Scan Register Cell). The boundary scan registers can link up to each other, forming a boundary scan chain around the chip[9] (Boundary-Scan Chain). General chip provides several independent scan chains, setting the external JTAG interface to access boundary scan registers then to observe and control chip's I/O terminal, to complete the test in chip level, board level and system level[10].

There are five testing interfaces in the standard JTAG, which are used to control the whole test process. It has four inputs and one output, which are defined as follows:

TDI——test data input terminal, to serial input control data or instruction;

TDO——test data output terminal, can read serial data or instruction of the last status;

TMS——test status selection input terminal, used to control the status transition of TAP, control the status of TAP;

TCK——clock input of the test, sampling the input along the rise;

nTRST——test reset input terminal, used to reset TAP controller.

Specific design of boundary scan mainly consists of the TAP control circuit, instruction register and decoding circuit, data register:

(1) Instruction register (IR) and decoding circuit. Instruction register is used for meeting the different control needs of boundary scan unit. The decoding circuits can decode different instruction; give the corresponding control signal to meet the different needs of the test.

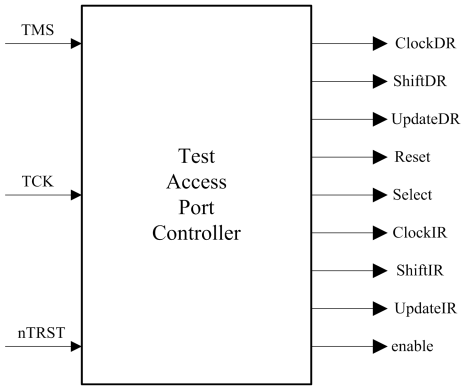

Fig. 4. Structure of TAP Controller.

(2) The design of data register (DR), which is used to store the data for testing or the result. Boundary scan chain is one of them. The boundary scan chain inserts around the logic for test, cooperates with access to instruction register to test internal logic and the external connection of the chip conveniently. Besides that, data register also includes bypass register, product identity register, which are used for different test requirements.

(3) TAP controller, as shown in Fig. 4, its main function is to generate control signal that is used to access instruction register and data register.

Where ClockIR, UpdateIR, ShiftIR are used for access to instruction register and ClockDR, UpdateDR, ShiftDR are used for access to data register.

IEEE1149.4 defines TAP controller status machine, as shown in Fig. 5. All of the internal control signals are realized by decoding statuses.TAP controller will be automatically initialized to test logic reset status when power on (Test-Logic Reset), it'll also enter test logic reset status through the nTRST or TMS keep high for five clock periods. If it turns into instruction register operate status (Select-IR-Scan), TDI and TDO will be connected by instruction register. Then we can input the needed instruction sequence through the TDI, and the decoding circuit will automatically decode and generate the corresponding execution of the control signals. When it turns into data register operate status (Select-DR-Scan), TDI and TDO will be automatically connected by data register. Then we can serial input the data for testing through the TDI, put the test vector into the memory port, at the same time, the test results will be serial output from the TDO.

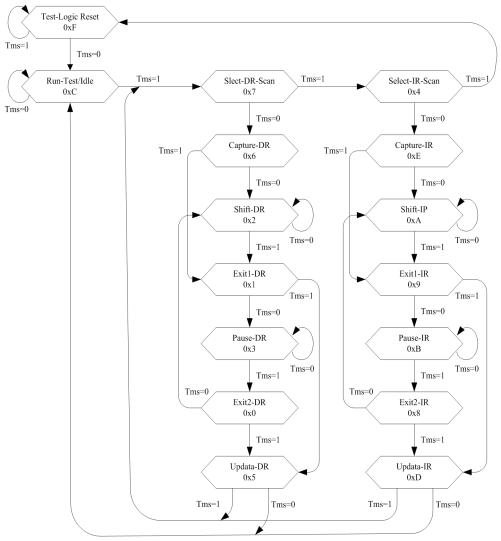

Fig. 5. Status Transition of TAP Controller.

4 The JTAG Test Implementation of Flash

As shown in Fig. 6, what used in our SoC design is Flash with 128kx8 of TSMC, whose control ports include X address line, Y address line, erase control signal, write gating signal, data input and output lines, etc. In order to facilitate testing Flash, in addition to the definition of the bypass register instruction register, ID register which comply with the std.1149.1, according to the characteristic of Flash, we design two boundary scan chains (Scan Chain). The two scan chains all belong to data register. Each one is made by several JTAG scanning units. Each of the scanning units can be configured to input unit that can obtain external signal or output unit. Relying on the shift register, the data can be output to every JTAG scanning unit through the TDI signal of JTAG, the data of each scanning unit can also be read from TDO signal line.

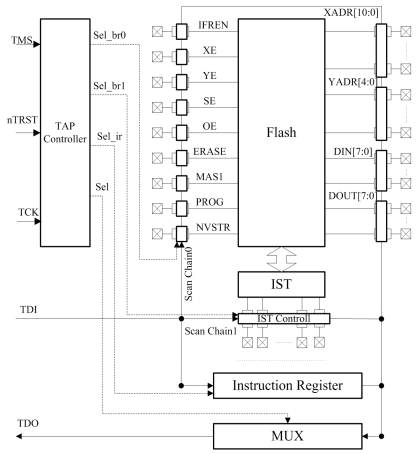

Fig. 6. Design of Flash Testing Scan Chain.

Through the TAP controller we can control which instruction register to access, and then choose different scan chain to complete the corresponding test. Scan chain0 directly connects each port of Flash, it controls through the shift register. According to the timing sequence of Flash, the test can be completed by sending each test vector ordinarily. Scan chain0 have 41 scanning units, the definition of scan chain registers are shown in Table 1.

Table 1. Setting Word's Margins.

Scan chain register	*Control signal*
BSR[0]	NVSTR
BSR[1]	PROG
BSR[2]	MAS1
BSR[3]	ERASE
BSR[4]	OE
BSR[5]	SE
BSR[6]	YE
BSR[7]	XE
BSR[8]	IFREN
BSR[13:9]	YADR[4:0]
BSR[24:14]	XADR[10:0]
BSR[32:25]	DIN[7:0]
BSR[40:33]	DOUT[7:0]

The Table 1 shows, this scan chain includes all I/O of Flash, through this scan chain we can directly control Flash to test.

If we solely depend on external direct access to test Flash, more test vectors would be used and the test time would be longer, considering that, we add the design of IST (In System Program) module to our design, as shown in Fig. 7. Scan chain1 is inserted into the port of IST. Through the IST control circuit, we only need to input simple control word to finish the test of Flash module, we can read and write full chip, write tessellated or diagonally to complete the BIST (Build in Self Test) of Flash.

There are seven scanning units in scan chain1, mainly used to access IST control circuit. The control order of this scan chain is: IST control signal R/W, used to control the read-write of register. Control instructions 3 to 0 (Command [3-0]) are used to control IST to execute different operations. Data line 7 to 0 (Data [7:0]) are used to input data for writing when write full chip, or input data for comparing when read full chip. Finish signal of the operation, used to learn whether the IST finished. Marking signal of the F/S, use for marking the success of Flash operations. Therefore, through this scan chain we can control IST, thus we can finish operating Flash by inputting simple control word and it can simplify the process, save the cost more than accelerate the test time.

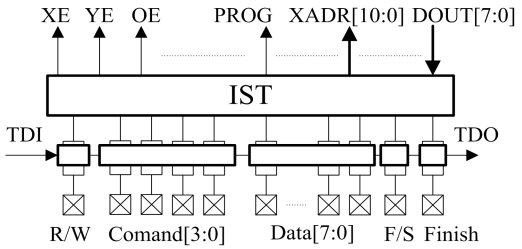

Fig. 7. Detail Drawing of The Scan Chain of Ist Circuit.

5 Simulations and Result

We use the EDA development tool of Synopsys Company to simulate and the entire verification environment is programmed by System Verilog language. We write test vector according to the read-write timing sequence of Flash then output the simulation model through the JTAG control signal written by System Verilog and then input through the boundary scan interface of Flash to complete the test of Flash. Fig. 8 shows the process of writing test data through the JTAG testing interface. The process is to write the data A5 to the place whose address XADR is 0x009, YADR is 0x00 in C2.

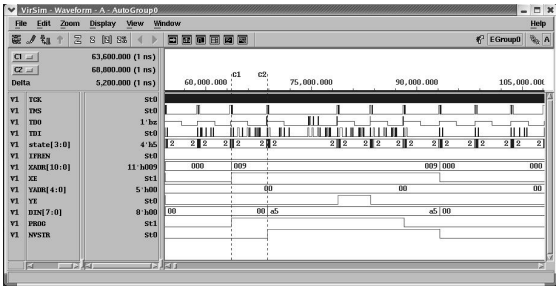

Fig. 8. Write Data Simulation Waveform with Flash.

The process of reading the data A5 from the place whose address XADR is 0x009, YADR is 0x00 is shown in Fig. 9. The data will be removed from TDO and the JTAG simulation test host will cache the data after receiving it and then distinguish whether it is A5. BSR [40:33] insert in DOUT and we can see the TDO output from the Fig. 9.

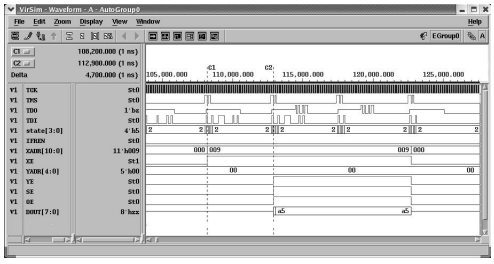

Fig. 9. Write data simulation waveform with Flash.

From what have been discussed we know that controlling the read-write of Flash directly through the scan chain0 will make the writing of test vector complicated,because the test vector for each signal have to be modified when the signal changes with time.

But using the scan chain1 can decrease the test control input, operating Flash through the IST will save a lot of testing time. Take full chip writing for example, if the operating frequency of TCK is 5MHz, the input of each a set of test vector probably need 8200 ns (42 cycles), the time of realize full chip writing through scan chain0 is about 7.5 s and if the operating frequency of TCK is 10MHz, the time is 4.5s. But using the scan chain1 can save input time of each test vector, the time to realize full chip writing is shorten to 2.5 s. But the weakness of IST is complexity of the error locating. Considering comprehensively, we can combine the two test plans, which is that the most of the test can be completed by scanning chain1, including the test of checkerboard and diagonal, and scanning chain0 can be used for test positioning and the analysis of specific problem when we found problems.

6 Conclusions

This paper proposed a testable design method of Flash based on the detailed analysis of the boundary scan technique. This testing method of Flash has been applied to the high-capacity storage card that is designed by our lab, the simulation result and actual chip test are indicate that this method can effectively save the test time and cost. This paper just applies the scheme to the test of Flash, how to use the combination of boundary scan technique and built-in test to realize DFT design of the entire chip should be studied in further research.

Acknowledgment

This work is supported by the National High Technology Research Development Program (“863” Program) of China (NO. 2015AA016501).

References

1. Van Der Wolf, Pieter (Synopsys, Eindhoven, Netherlands), Geuzebroek, Jeroen, “SoC infrastructures for predictable system integration” Proceedings - Design, Automation and Test in Europe, DATE, p 857-861, 2011, Proceedings - Design, Automation and Test in Europe Conference and Exhibition, DATE 2011.
2. Guo, Jian-Min (Department of Automation, Xiamen University, Xiamen, 361005, China); Luo, De-Lin, “A functional enhancement methodology to JTAG controller in complex SOC” Proceedings of 2009 4th International Conference on Computer Science and Education, ICCSE 2009, p 1128-1131, 2009,

Proceedings of 2009 4th International Conference on Computer Science and Education, ICCSE 2009.

3. Zhang, Pin(School of Electronic Engineering, Tianjin University of Technology and Education, Tianjin, 300222, China); Xing, Zuocheng, “Design of structure of debugging software in CPU based on JTAG” Applied Mechanics and Materials, v 58-60, p 1866-1870, 2011, Information Technology for Manufacturing Systems II.
4. Balasubramanian, Lakshmanan (Texas Instruments India Private Limited, Bangalore, India), Sabbarwal, Puneet, Mittal, Rajesh Kumar, Narayanan, Prakash, Dash, Ranjit Kumar, Kudari, Anand Devendra, Manian, Srikanth, Polarouthu, Sudhir; Parthasarathy, Harikrishna, Vijayaraghavan, Ravi C., Turkewadikar, Sachin, “Circuit and DFT techniques for robust and low cost qualification of a mixed-signal SoC with integrated power management system” Proceedings - Design, Automation and Test in Europe, DATE, p 551-554, 2011, Proceedings - Design, Automation and Test in Europe Conference and Exhibition.
5. K. S. Kushal, S. Chetan and Shivaputra, "Design and Implementation of a Smart Automated OUT-ward (SAT OUT) System Using ARM Processor for Aircrafts," Computational Intelligence and Communication Networks (CICN), 2012 Fourth International Conference on, Mathura, pp. 463-467, 2012.
6. S. Chen and L. Xu, "A boundary-scan test bus controller design for mixed-signal test," Wireless Communications, Networking and Information Security (WCNIS), 2010 IEEE International Conference on, Beijing, China, pp. 22-25, 2010.
7. F. Bouwman, S. Oostdijk, R. Stans, B. Pouya. Macro Testability: The Results of Production Device Application. Test Conference, 1992. Proceedings., International. P.232-241.
8. Melo, Rodrigo A., David M. Caruso, and Salvador E. Tropea. "Memory-mapped I/O over dual port BRAM on FPGA." Programmable Logic (SPL), 2012 VIII Southern Conference on. IEEE, 2012.
9. T. Borroz, "Flexible high performance architecture for boundary scan execution hardware," IEEE AUTOTESTCON, 2015, National Harbor, MD, pp. 232-235, 2015.
10. Mandal, Debashis (Department of Electronics and Electrical Communication Engineering, Indian Institute of Technology Kharagpur, Kharagpur-721302, India), Bhattacharyya, T.K., “Implementation of CMOS low-power integer-N frequency synthesizer for SOC design” Journal of Computers, v 3, n 4, p 31-38, April 2008.