©2010 International Journal of Computer Applications (0975 - 8887)
Volume 1 – No. 17

Test Access Port & Scan Based Technique of In-System Timing Extraction and Control

Richa Maheshwari
Department of Electronics & Communication Engineering
Manav Rachna College of Engineering, Faridabad

Dr. Neeta Awasthi
Department of Electronics & Communication Engineering
Galgotia College of Engineering, Greater Noida

ABSTRACT

Historically, most Printed Circuit Board (PCB) testing was done using bed-of-nail in-circuit test equipment. The progress in the field of miniaturization and integration density has possible to design very complex PCBs, which presents very high testability requirements. Boundary scan is now the most promising technology for testing high complexity PCBs. This paper presents scan-based test access port standard for testing complex ICs and also presents a method which allow the extraction of fine-grained timing information using Test Access Port (TAP).

I. INTRODUCTION

Today's high density devices pose unique manufacturing challenges like the accessibility of test points and the high cost of test equipments. The development of highly complex PCB, raising extreme testability requirements, is enabled by the availability of advanced packaging and mounting technologies, and by integration levels producing components with thousand of gates per pin. Boundary scan design and test (BST) was developed in response to this challenge and is now largely recognized as being able to effectively improve the previous test technologies, essentially in two main areas; lowering the physical accessibility requirements and improving the controllability and Observability (C&O) levels of internal PCB nodes [1].

The JTAG/IEEE 1149 standard is facilitating economical, functional & in-system component testing. As currently employed, the IEEE 1149 TAP can be used to extract information on the functional integrity of ICs and their inter-connections but not the timing-related characteristics. In this paper, we introduce a few simple circuits and techniques which allow the extraction of timing information via an IEEE 1149 compatible TAP [1]. The techniques developed allow a low bandwidth, high latency interface, such as scan-based TAPs, to initiate timing events and capture moderately fine-grained information about signal timing during events. Using only the standard TAP interface, these techniques can extract relative, on-chip timing information.

II. BOUNDARY SCAN

Many system defects occur at the board-level including open or shorted PCB traces and incomplete solder joints. At the board-level "bed-of-nail" testers historically were used to test boards. The increasing complexity of boards and the movement to technologies such as multi-chip modules (MCMs) and surface mount technologies resulted in system designers agreeing on a unified scan-based methodology called BOUNDARY SCAN for testing chips at the board and system level. *Boundary Scan* is a method allowing complete controllability and observability of the boundary pins of a compatible device via software control.

Fig.1 illustrates the basic principle of IEEE 1149.1 boundary scan. All the I/O pins of each IC on the board are connected serially in a standardized scan chain accessed through the test access port so that every pin can be observed and controlled remotely through the scan chain. At the board level, ICs obeying the standard can be connected in series to form a scan chain spanning the entire board [3]. During standard operations, boundary cells are inactive and allow data to be propagated through the device normally. During test nodes, all input signals are captured for analysis and all O/P signals are preset to test down-string devices. The operation of these scan cells is controlled through the TAP controller.

III. TAP AND ITS ARCHITECTURE

The test access port has four or five single bit connections as given in Table-1. When the chip is in normal mode, TRST & TCK are held low & TMS is held high to disable boundary scan. To prevent race conditions inputs are sampled on the rising edge of TCK and output toggle on the falling edge.

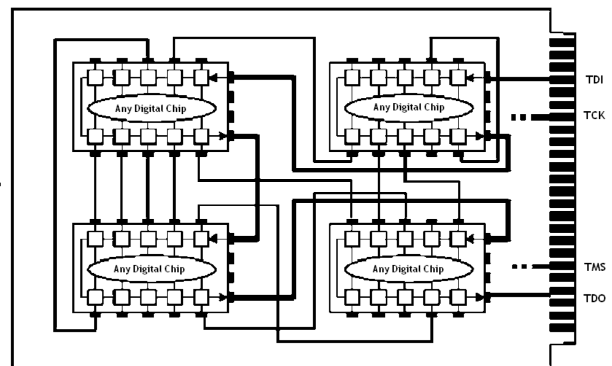

Fig. 1.IEEE 1149 Boundary scan principle

TABLE I
TAP INPUT OUTPUT PINS

TCK	Test Clock	Input	Clock tests into & out of the chip.
TMS	Test Mode Select	Input	Controls test operations.
TDI	Test Data In	Input	Test data into the chip.

TDO	Test Data Out	Output	Test data out of the chip.
TRST	Test Reset Signal	Input	Optional active low signal to reset the tap controller.

The basic test architecture is shown in Fig.2, It consists of:

a) A TAP interface pins
b) Test Data Registers (DR) to collect data from the chip
c) Instruction Register (IR) specifying the type of test to perform
d) TAP controller which controls the scan of bits through the instruction and test DR.

A. TAP Controller

The TAP controller is a 16-state Finite State Machine (FSM) as illustrated by Fig.3, controlling operations associated with boundary scan size. The basic operation is controlled through 4 pins: TCK, TMS, TDI and TDO. The TCK and TMS pins direct signals between TAP controller states. The TDI and TDO pins receive the data input and output signals for the scan chain. Optionally, the 5th pin, TRST, can be implemented as an asynchronous reset signal to TAP controller. Working in conjunction with the TAP controller is an IR providing which type of test to perform [1]. The IEEE 1149.1 standard requires that all compliant devices must perform the following 3 instructions.

1. EXTEST: This instruction performs a PCB interconnect test. The EXTEST inspection places an IEEE 1149.1 compliant device into an external boundary test mode and selects the boundary scan register to be connected between TDI and TDO. During this instruction, the boundary scan sizes associated with outputs are preloaded with test patterns to test downstream devices.

2. SAMPLE/PRELOAD: This instruction allows an IEEE 1149.1 compliant device to remain in its functional mode and selects the boundary scan register to be connected between the TDI and TDO pins. During this instruction, the boundary scan register can be accessed via a data scan operation, to take a sample of the functional data entering and leaving the device. This instruction is also used to preload test data into the boundary scan register prior to loading an EXTEST instruction [2].

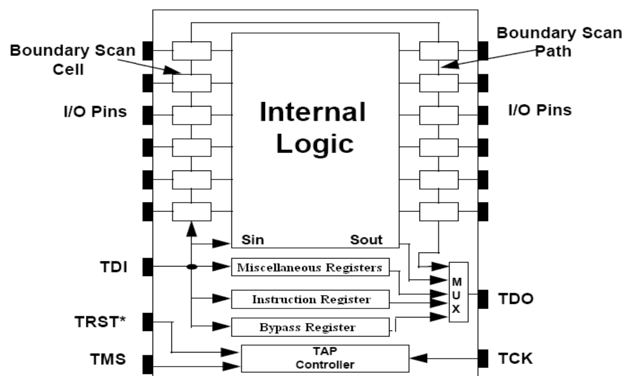

Fig.2. Basic Test Architecture

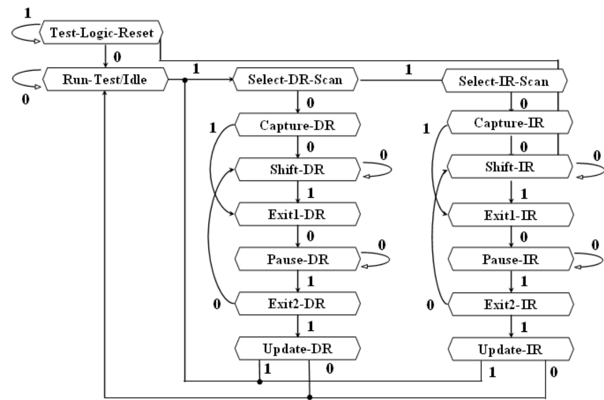

Fig.3. State Diagram of TAP Controller

3. BYPASS: A device's boundary scan chain can be skipped using the bypass instruction, allowing the data to pass through the bypass register. This allows efficient testing of a selected device without incurring the overhead of traversing through other devices. The bypass instruction allows an IEEE 1149.1 compliant device to remain in functional mode and select the bypass register to be connected between the TDI and TDO pins. The bypass instruction allows serial data to be transferred through a device from the TDI pin to the TDO pin without affecting the operation of the device.

IV. BOUNDARY SCAN TESTING

In a board design there usually can be many compliant devices. All these devices can be connected together to form a single scan chain as illustrated in Fig.1,. Alternatively, multiple scan chains can be established so parallel checking of devices can be performed simultaneously [4]. The off-board TAP control device can perform different test during board manufacturing without the need of bed-of-nail equipment. One of the first tests that should be performed for a PCB test is called the infra-structure test. The test is used to determine whether all the components are installed correctly. This test relies on the fact that the last two bits of the IR are always "01". By shifting out of the IR of each device in the chain, it can be determined whether the device is properly installed. This is accomplished through sequencing the TAP controller for IR read.

After the infra-structure test is successful, the board level interconnect test can begin. This is accomplished through the EXTEST command. This test can be used to check out "opens" and "shorts" on the PCB. The test patterns are preloaded into the output pins of the driving devices. Then they are propagated to the receiving devices and captured in the input boundary scan cells. The result can then be shifted out through the TDO pin for analysis. These patterns can be generated and analyzed automatically, via software programs. This feature is normally offered through tools like Automatic Test Pattern Generation (ATPG) or Boundary Scan Test Pattern Generation (BTPG).

V. TAP BASED TIMING EXTRACTION

TAP based timing extraction and adjustment offers new opportunities for both inexpensive, in-system testing and timing adaptation for high performance operation. Timing extraction can be directly employed for speed grading and delay fault testing [3]. With adjustment circuitry, critical on-chip signals can be tuned after fabrication to achieve desired timing relations. This adjustment can be performed in-system, allowing adjustment to account for system-level and environmental variation as well as IC fabrication variation.

A. Sample Register

A sample register is a string of latches enabled at closely spaced time intervals. Each latch "samples" the binary value of the signal under test during the time it is enabled. By rippling the latch enables in rapid succession, the sample register captures a discrete representation of the time behavior of a signal. A simple sample registers as shown in Fig.4. can be implemented using a pair of inverters to provide the fixed, inter-sample-bit delays.

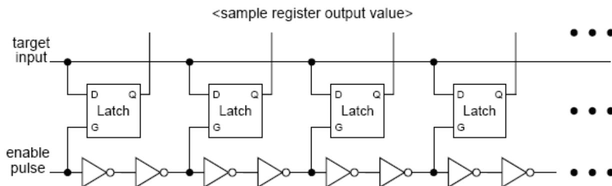

Fig.4. Sample Register

Target Data	000000000111111111...
Sample 1	[000000]000111111111...
Sample 2	000000[000111]111111...
Sample 3	000000000111[111111]...

Fig.5. Sliding Window

B. Sliding Window

If we can cause the event we are timing to occur under scan control, we can trade-off time for space. We do not have to capture the entire waveform in a single event. We can reuse a small, finite-length sample register in time to capture a long waveform. The temporal placement of the sample register can be controlled via the TAP, and the composite waveform can be reconstructed off chip [4]. Fig.5, shows the basic sliding window concept where a small, fixed-size, sample register is used to capture a portion of the discrete-time waveform for a signal with each timing event. If we vary the placement of the capture window and repeatedly fire the timing event, we can capture the entire waveform over a series of such samples.

Fig.6 shows one possible implementation for the sliding window. The sample pulse is recycled after rippling through several sample delays. A scan-loaded configuration is compared against a trip count to allow the sample pulse to be recycled for a pre-determined number of times. After the enable pulse settles following a trigger event, the sample register will contain the values corresponding to the last time the pulse was allowed to ripple through the delay chain. By recycling the ripple from a point prior to the end of the sample register we can do two things: (1) provide overlap between sample windows and (2) make sure that there is always an inverter-pair delay between adjacent samples used to reconstruct the longer waveform.

The choice of how many bits to include in the sample register and recycle path will depend on the relative speed of operation of various logic functions in the target technology [3]. For example, a technology with 100 ps minimum inverter delays and a maximum counter operational frequency of 500 MHz would require a minimum of 10 inverter pairs in the recycling portion of the sample delay chain.

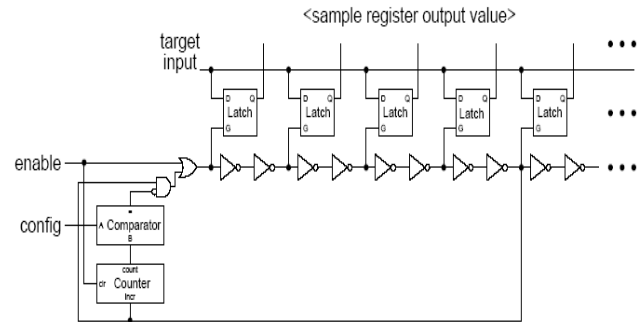

Fig.6. Recycling Sample Register

C. Calibration

With the circuits shown so far, we only know when the signal is occurring in units of inverter-pair delays. Since process and environmental variation can easily account for a factor of two variances in an inverter delay, inverter-pair delays alone are not sufficient to extract fine-grained timing information [2]. If a known timing source, such as the component clock, is available we can mux the sample input between the sample register and the known timing source to calibrate the inter-sample-bit delay time.

D. Summary

Combining these techniques we can repeatedly fire an event we wish to time and sample its behavior in narrow, fixed-size windows. By integrating the information acquired across multiple samples at varying window offsets and calibrating to known frequency and phase sources, we can build up an accurate, discrete-time representation of a signal on the IC.

VI. APPLICATIONS

TAP-based timing extraction immediately presents us with two large classes of opportunities to exploit: (1) Operational speed/timing extraction and (2) In-system timing tuning.

A. Timing

The TAP-based timing extraction techniques introduced above support speed and delay fault testing. Sample registers designed into the IC can monitor strategic control signals. Signal timing can be observed in response to applied test patterns.

B. Delay Adjustment

On-chip delay adjustment allows us to tune the timing of events to the target system. Such tuning can be used to account for environmental variations (*e.g.* temperature) and variation in system parameters (*e.g.* off-chip propagation delays). In-system delay adjustment allows the IC to achieve the highest performance in the final operational environment without sacrificing performance to worst-case parameter possibilities [4]. In-system delay adjustment provides most of the advantages of self-timed logic without incurring the complexity and testability problems associated with asynchronous logic.

VII. LIMITATIONS AND COST

A. Area

The primary limitation is, of course, the area required for each sample register. In an experimental test chip fabricated in a 0.8µ process, a 16-bit sample register occupied just under 350µ × 150µ. This prototype sample register included the inverter chain, sample latch, shift register, and a 16-bit configuration register, but did not include any recirculation or calibration circuitry [3].

B. Configuration and Test Latency

Test latency using a sample register is moderately large compared to standard, functional, scan tests or timing tests on a high-speed IC tester. High speed VLSI chip tester time is expensive. One would not want to make excessive use of TAP-based timing extraction during premium operating time on a high-speed tester.

VIII. CONCLUSION

Board level testing has become more complex with the inclusive use of fine pitch, high pin count devices. However, with the use of boundary scan the implementation of board level testing is done more efficiently and at lower cost.

The techniques presented can be exploited using standard, scan-based test-access ports and operate both post-fabrication and in-system. The timing information extracted is suitable for evaluating the operational timing characteristics of an IC including speed grading and delay fault testing. This timing information can also be used in conjunction with delay adjustment circuitry to tune IC timing to match system requirements.

ACKNOWLEDGEMENT

We would like to express our deepest gratitude and appreciation to Prof. M.S.Giri who has been an invaluable source of support, guidance, and encouragement. The deepest appreciation must also go to all our colleagues in the Electronics Department who were of great help and support. Authors would like to express sincere gratitude to Rakesh Bharadia for his constructive comments.

REFERENCES

[1] IEEE Standards Committee. IEEE Standard Test Access Port and Boundary-Scan Architecture. IEEE, 345 East 47th Street, New York, NY 10017-2394, July 1990. IEEE Std 1149.1-1990.

[2] Andr´e DeHon, Thomas F. Knight Jr., and Thomas Simon. Automatic Impedance Control. In ISSCC Digest of Technical Papers, pages 164–165. IEEE, February 1993.

[3] Mark Horowitz, Andy Chan, Joe Cobrunson, Jim Gasbarro, Thomas Lee, Wing Leung, Wayne Richardson, Tim Thrush, and Yasuhiro Fujii. PLL Design for a 500MB/s Interface. In ISSCC Digest of Technical Papers, pages 160–161. IEEE, February 1993.

[4] Mark G. Johnson. A Variable Delay Line PLL for CPU-Coprocessor Synchronization. IEEE Journal of Solid-State Circuits, 23(5):1218–1223, October 1988.