Design of a Parallel Bus-to-Scan Test Port Converter

ABSTRACT

The IEEE Std. 1149.1 Standard Test Access Port and Boundary-Scan Architecture[1] as well as other scan path methodologies use a serial interface for transmitting data to and from the circuit under test. This serial communication presents an efficiency problem in transferring data between a processor and the scan ring. This paper describes the architecture and features of a device that interfaces a parallel host bus to a serial test bus. The parallel/serial converter, National's Embedded Boundary Scan Controller, SCANPSC100F, integrates several features to simplify board test and offers a way to make scan operations more efficient by managing shift operations directly in hardware.

INTRODUCTION

Scan path methodologies are the dominant technique in implementing Design-For Test in board and system designs. With the advent of high density board/packaging technologies, the need for scan is even more pronounced as physical access to multiple nodes on the board is no longer feasible. Boundary scan has become the generally accepted answer to this access problem. Wide acceptance of boundary scan is most readily evidenced with the creation of the IEEE Std. 1149.1 Standard Test Access Port and Boundary-Scan Architecture[1, 2]. Standardization of boundary scan through 1149.1 allows board designs to contain scannable components supplied by different manufacturers.

The use of 1149.1—or any other scan methodology—presents the problem of efficient data transference to and from the scan ring. Implementing scan operations with only a processor is software intensive, and consequently, very slow. The general solution to this efficiency problem is to manage shift operations in hardware.
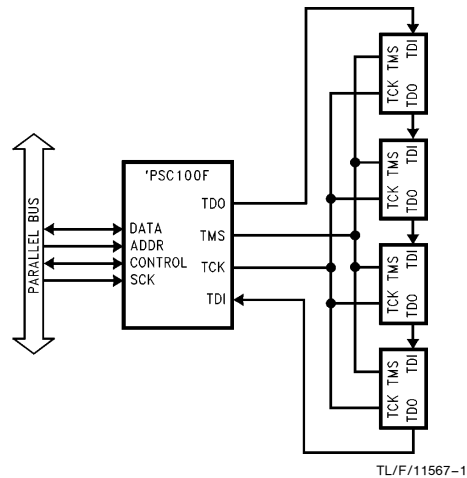
The 'PSC100F was developed to offer a hardware solution to the scan problem. Simply put, the 'PSC100F facilitates the transfer of data between a parallel host bus and a serial test bus. The following sections first define objectives for developing the 'PSC100F. The architecture and basic operation are then described. Next, specific features are outlined along with the trade-off decisions that were made during implementation.

OBJECTIVES

The general objective of the 'PSC100F is to act together with a parallel bus host as a serial bus controller. A general concept of this application is shown in *Figure 1.*

Figure 1 shows the 'PSC100F interfaced to the 1149.1 bus. Because it is very likely that systems will contain other than 1149.1-type devices, one of the main design objectives is a 'PSC100F that is flexible enough to support many types of scan bus styles and standards. One can assume that systems will also contain several devices that employ some proprietary scan protocol. Also, there is another IEEE standard currently being developed that will provide an alternative to 1149.1[3]. To attain serial scan protocol independence, a means must be provided for the protocol to be determined by software.

Flexibility should also be designed into the parallel port to provide a variety of choices for a parallel bus host. A generic read/write protocol should be used. 'PSC100F status

Paper first published at Electro International, 1991.

FIGURE 1. Parallel/Serial Converter, National's Embedded Boundary Scan Controller, SCANPSC100F, General Application

should be supplied to the host with status registers or an interrupt pin. Synchronization between the 'PSC100F and the host should be reviewed as they may not always operate off of the same clock.

Several target applications must evaluated when designing a control mechanism for a serial test port. External instruments could use the 'PSC100F to serialize test vectors to be applied to some unit under test. For example, a 'PSC100F designed into a PC board could allow the PC to act as a simple and inexpensive automatic tester. Embedded in small systems, the 'PSC100F may access a single scan ring from a central control point. The host may be the system processor, a dedicated test processor, or a simple state machine for transferring test vectors between memory and the 'PSC100F. In larger systems, a single ring may cause unacceptable scan throughput. Partitioning the test architecture into many scan rings allows patterns to be scanned and applied to many blocks concurrently, improving throughput. The 'PSC100F should be able to support various combinations of these system test architectures.

In addition to providing a means of serializing data, the 'PSC100F should also provide features that simplify typical test operations. Pattern generation and compression features should be considered as well as a counter to count scanned bits and to count clocks during Built-In Self-Test (BIST) operations. Other features could be built in to simplify typical operations performed when using the 1149.1 standard, e.g., the SAMPLE operation.

When using the 'PSC100F, the assumption is made that testability is being designed into the system; therefore, to maintain a hierarchical test structure, the 'PSC100F must also be easily testable. Features must be implemented so that the 'PSC100F is fully testable from the parallel bus, or from a dedicated test port if size and cost allow. Also, some bypass mechanism should be implemented so that an external source (e.g., an ATE) may have access to the serial bus.

ARCHITECTURE AND BASIC OPERATION

The basic structure of the 'PSC100F is summarized by the block diagram in *Figure 2*. The conversion between parallel and serial data is basically managed by a bank of double-buffered, parallel-to-serial shift registers. Double buffering allows a parallel write or read to occur while data is shifting, thus improving throughput.

Three shifter/buffers are used for out-going data and one for in-coming. The pin names for the serial port follow the same convention as described in 1149.1. The Test Data Out (TDO) and Test Mode Select (TMS0:1) ports are used to output serial data, while data is shifted in at Test Data In (TDI). The Test Clock (TCK) is also generated and is derived directly from the 'PSC100F's clock, SCK. TDO and TDI are used to scan test data. If used in an 1149.1 application, TMS0:1 can be used to supply control to the Test Access Ports (TAP) of the 1149.1 board components. With two TMS lines, up to 2 scan chains can be controlled in parallel. However, as data appearing at the TMS0:1 outputs is completely user specific, other protocols can be implemented.

Read and write operations are performed with an 8-bit asynchronous parallel interface. For write operations, R/$\overline{W}$ and $\overline{CEB}$ lines are forced low. Data and address are then loaded with the STB signal. The write operation is synchronized to SCK before data is transferred to the addressed register. Read operations occur with R/$\overline{W}$ held high. When reading status bits for polling operations, a command must be issued to update status so data will not change while a read is taking place. Handshaking between the 'PSC100F and the host processor can be accomplished 3 ways: 1) By generating wait-states with the Ready ($\overline{RDY}$) signal, 2) By using any of 3 event-driven interrupts via the INT signal, or 3) by polling status bits in the Status Register.

Three mode bits are used to mask interrupt events which cause the INT pin to go high. Interrupts may be generated by expiration of the 32-bit counter, by the transmission of a byte of data from the TDO port, or by the reception of a byte by the TDI port.

The general problem of serial data flow is managed by controlling TCK, which is derived directly from SCK. If the host cannot keep up with the serial flow of data, TCK will be stopped in the low state. A 32-bit counter is included to control the number of bits being shifted. The number of bits to be scanned is first loaded into the counter by 4 consecu-

tive writes to its address. Shifting may progress until the counter expires, causing TCK to stop. If the number of bits scanned is not a multiple of 8, software must align the last byte shifted and mask the unused bits.

The functionality of the 'PSC100F in any given application is dictated by the contents of the Mode Registers. Each shifter/buffer has a specific mode bit with which it can be enabled or disabled. If a shifter/buffer is not enabled, it has no impact on serial data flow and retains its last state. This is particularly useful when control data remains static during long shifts of test data. For example, during the Shift-DR state of the 1149.1 TAP Controller, TMS is held low for the duration of time that data is being scanned. Instead of shifting a long pattern of zeros from the TMS port, the TMS shifter/buffer can be turned off and the processor can spend all its time transferring data to and from the TDI and TDO ports.

The following sequence describes the actions that typically occur when using the 'PSC100F to apply test data through an 1149.1 boundary scan chain. The sequence begins with the TAP Controller *(Figure 3)* of each device on the board in the Run-Test Idle state. The objective in this example is to capture data in the Capture-DR state and shift 500 bits of new data into the scan path while shifting out 500 bits which resulted from the capture. The new data shifted in will be applied to the system logic in the Update-DR state. The TAP will then be returned to the Run-Test Idle state.

1. The host writes to the 'PSC100F, accessing one of the mode registers and enabling the 32-bit counter and one TMS port.

2. Three clock cycles are needed to get from Run-Test Idle to Shift-DR (see *Figure 3*). The pattern needed on TMS is 001 (LSB shifted first). First the counter is loaded with (00000003)h which will cause 3 clocks and 3 TMS bits to be issued. This is accomplished with 4 consecutive writes to the counter's address.

3. The pattern XXXXX001 is loaded into the TMS shifter/buffer by writing to its address. (The 5 most significant bits will not be used.)

4. TCK now runs for 3 clock cycles and stops. While transitioning from the Capture-DR state to the Shift-DR state, data was captured into the boundary scan register on the board.
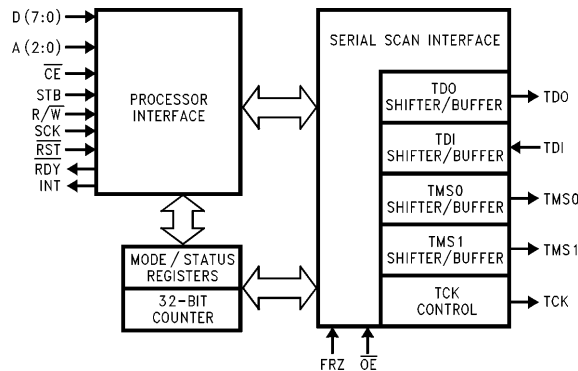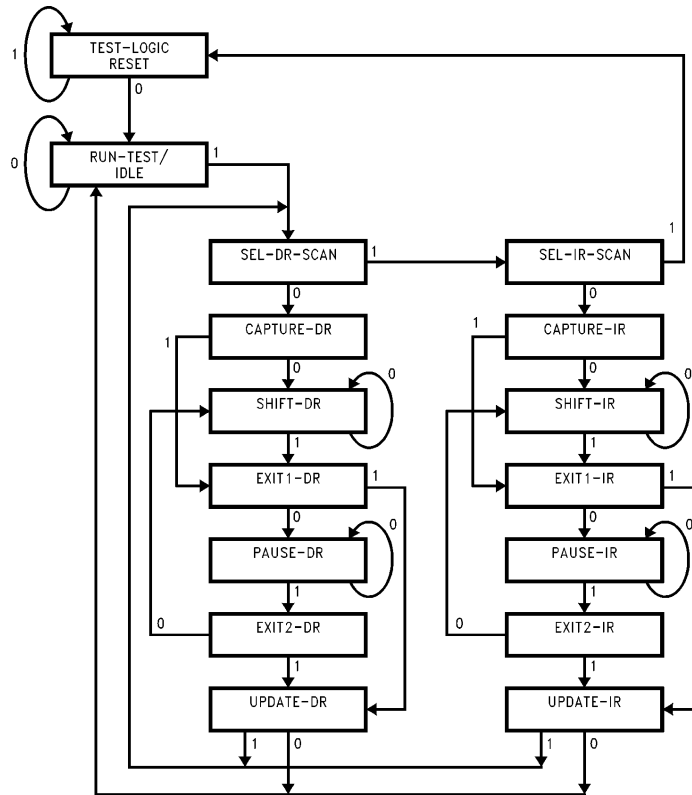

FIGURE 2. Embedded Boundary Scan Controller, SCANPSC100F Architecture

TL/F/11567–2

FIGURE 3. 1149.1 TAP Controller

TL/F/11567–3

5. With the TAP controller residing in the Shift-DR state, the next step is to configure the 'PSC100F to shift 500 bits of new data out of TDO while shifting the 500 captured bits into TDI. This action requires a write from the processor to the 'PSC100F Mode register 0 to disable TMS, to enable TDO and TDI, and set the Auto TMS High Enable Bit. With the TMS buffer disabled, the TMS pin retains the last logic level shifted out of the TMS buffer (i.e., the logic low which transitioned the TAP controller into the shift-DR state). This will hold all TAP controllers connected to the 'PSC100F in the shift-DR state.

Setting the Auto TMS High Enable bit provides a key efficiency gain when shifting data to and from the 'PSC100F. With this bit set, TMS pin will automatically drive a high logic level, for one TCK cycle, when the counter reaches a count of 00000001. This high logic level transitions all TAP controllers to the EXIT1-DR state which shifts out/in the last bit and completes the shift operation.

6. TCK begins clocking when the first byte of data is written to TDO. The host exchanges 500 bits of data with the boundary scan chain by several consecutive write/read cycles. During this step, the processor may be unable to keep up with the serial flow of data, causing the TDO shifter/buffer to become empty or the TDI shifter/buffer to become full. If either of these conditions occurs before the counter expires, TCK will be stopped and restarted to prevent overflow and underflow conditions. When 499 clocks are issued, the counter expires causing TCK to stop.

7. To complete the example, the TAP Controller must be sequenced from Exit1-DR (where it is now), through Update-DR, and back to Run-Test Idle.

8. As TDO and TDI are no longer needed, their respective mode bits are cleared. The counter is loaded with (00000002)h which is how many clocks are needed to return to Run-Test Idle. The necessary TMS data is loaded: XXXXXX01.

9. Two TCK pulses are now issued and bring the TAP back to the Run-Test Idle state. Data that was shifted into the boundary scan chain in steps 7 and 8 has been updated during the Update-DR state.

FEATURES AND TRADEOFFS

This section describes many of the specific features included in the Embedded Boundary Scan Controller design. The reasoning for each feature is detailed.

Technology: The 'PSC100F is fabricated in a 1.5 μm CMOS process with the option of having either CMOS- or TTL-compatible input levels. CMOS is desirable for its "zero" static power consumption. When not in test mode, a system's power dissipation should not be impacted by dormant test circuitry.

Package: Pin count is kept to 28 for minimal impact on component cost and board area consumption. The 'PSC100F has been designed to fit both surface mount and through-hole type packages. Although surface mount packages will receive the most use, smaller companies that do not have

the luxury of surface mount manufacturing will use through-hole technology while still enjoying the low test costs of boundary scan.

Parallel Interface: A generic read/write protocol was chosen for the processor interface to ensure that compatibility will be preserved with all off-the-shelf processors. Three options are given for handshaking: interrupts, polling, and wait-state generation using the Ready ($\overline{RDY}$) signal. These options will give the user flexibility in transferring data. The data bus width was kept to 8 bits to minimize pin count and die size. Using a 16-bit bus would have meant going to the next highest standard package size: 44-pin. A significant increase in die area would also be seen. The improvement in scan throughput by using 16-bits did not outweigh cost.

Serial Port: Multiple ports were considered to maximize scan ring partitioning capabilities, but the package size constraint allows only one port. However, supplying two TMS pins allows up to two rings in parallel when using 1149.1.

Scan Protocol Support: An option was considered to include a "master" TAP Controller and control TMS directly in hardware. ICs have been presented that implement this kind of serial port controller[4, 5]. With this method, scan operations can be controlled by issuing a predefined set of commands. This option would allow a free-running TCK, with overflow and underflow prevented by using the Pause state of the 1149.1 TAP controller. Although this method of controlling scan would be less software intensive, there are several drawbacks. The controller would be restricted to using the 1149.1 protocol which may not be compatible with other proprietary scan methods. The command set would presuppose certain sequences through the TAP Controller, thus limiting flexibility. For these reasons, the "master TAP" method was not chosen. Instead, protocol and sequencing are completely controlled by software. Data overflow and undefflow is prevented by simply stopping TCK.

One drawback of this method is more software involvement and, consequently, a possible detriment to performance. In this application, however, performance is out-weighed by functionality and flexibility. Another possible drawback is that TCK must be gated. Gating TCK introduces a relatively large and unpredictable skew from part to part due to process variation. This skew will impact a high performance system if testing occurs while running at normal operating speed; however, it is anticipated that a majority of scan applications will occur while the system is in a slow "single step" mode. At low speeds, TCK skew will have no impact.

Clock Speed: The 'PSC100F is designed to run at up to 25 MHz over military and commercial temperature ranges.

Output Buffering: All outputs have 24 mA DC current drive and are capable of switching 50Ω transmission lines. The high drive capability of the parallel port reduces the need for buffering when communication occurs over the backplane. The serial port also gains from the high drive. For example, if using the 'PSC100F to serialize data for a PC tester, the strong drive of the outputs allows a lower number of buffer stages between the PC and the test head. On a typical board using 1149.1 components, TMS and TCK may see extremely high loads as these are connected to every scannable component The large outputs can handle these loads without external buffering. In addition to high drive, the output buffers contain noise suppression circuitry that will minimize ground bounce and undershoot.

External Tester Access: By forcing the $\overline{OE}$ pin high, all serial port pins will be TRI-STATEd. This allows an external test resource to access the serial port without contention with the 'PSC100F.

Pattern Generation and Compaction: The TDO shifter/buffer and 16 additional flip-flops may be reconfigured as a 32-bit linear feedback shift register (LFSR) for pseudo-random pattern generation (PRPG). The TDI shifter/buffer can be configured as a 16-bit LFSR for serial signature compaction.

Loopback Mode: By setting a mode bit, data appearing at TDI can be looped back through the TDO port. This simplifies read-only operations by leaving the scan path in its state prior to shifting.

Testability: The 'PSC100F is highly testable from the parallel port. All registers are directly or indirectly readable and writeable. A test loopback mode is provided so data appearing at TDO or TMS0:1 can be scanned back into the TDI shifter/buffer. The 32-bit counter can be placed in single step mode for easy testability. Adding a dedicated test port was considered, but again, package constraints prevented this.

SUMMARY

The presented device enhances transfer of data between a parallel processor bus and a serial test bus. The protocol for applying serial patterns is completely controlled by software for maximum flexibility. Data overflow and underflow are prevented by stopping the test clock, TCK. A 32-bit counter and other features have been integrated to help make the 'PSC100F a viable means for improving the efficiency of serial test data communication.

REFERENCES

1. IEEE 1149.1, *"Standand Test Access Port and Boundary-Scan Architecture",* IEEE Std. 1149.1-1990.

2. Maunder, C. and Tulloss, R., *"The Test Access Port and Boundary-Scan Architecture",* IEEE Computer Society Press, 1990.

3. IEEE P1149.2, (Proposed) Serial Scan Standard, currently being developed by the IEEE P1149.2 Working Group.

4. Vining, Sue, "Tradeoff Decisions Made for a P1149.1 Controller Design", International Test Conference Proceedings, August 29–31, 1989, pp. 47–54.

5. Yau, Chi W. and Jarwala, Najmi, *"The Boundary-Scan Master: Target Applications and Functional Requirements",* International Test Conference Proceedings, September 10–12, 1990, pp. 311–315.

LIFE SUPPORT POLICY

NATIONAL'S PRODUCTS ARE NOT AUTHORIZED FOR USE AS CRITICAL COMPONENTS IN LIFE SUPPORT DEVICES OR SYSTEMS WITHOUT THE EXPRESS WRITTEN APPROVAL OF THE PRESIDENT OF NATIONAL SEMICONDUCTOR CORPORATION. As used herein:

1. Life support devices or systems are devices or systems which, (a) are intended for surgical implant into the body, or (b) support or sustain life, and whose failure to perform, when properly used in accordance with instructions for use provided in the labeling, can be reasonably expected to result in a significant injury to the user.

2. A critical component is any component of a life support device or system whose failure to perform can be reasonably expected to cause the failure of the life support device or system, or to affect its safety or effectiveness.