OpenCORES's

Boundary Scan Implementation

Abstract

This document describes Boundary Scan Implementation (software and hardware solution. It is fully IEEE 1149.1 compliant.

Date: August 6, 2000
Version: 1.0

Author: Igor Mohor (igorm@opencores.org)

Boundary-Scan Architecture and compliance to the IEEE Std 1149.1

The dedicated test access port (TAP) is fully compliant to the IEEE Std. 1149.1. It consists of 5 dedicated signal pins, a 16-state TAP Controller and three test data registers: idcode register, bypass register and boundary scan register. The implementation provides the capability to:

- Read the ID code of the IC (version, part number, manufacturer's id)
- Perform boundary scan operations to check circuit board electrical continuity
- Bypass the IC for a given circuit board test by reducing the boundary scan register to a single cell
- Sample the IC system pins during operation and transparently shift out the result in the boundary scan register
- Disable the output drive to pins during circuit board testing

The implementation consists of 16-state TAP controller, 4-bit instruction register, 3 test registers (a 1-bit bypass, a 32-bit idcode register and a 14-bit boundary scan register) and 5 dedicated signal pins (signals). Many additional test registers are to be added in the future. The five signals connected to the TAP are:

- TCK - a Test Clock input that synchronizes the test logic.
- TMS - a pulled-up input Test Mode Select that is sampled on the rising edge of TCK to sequence the TAP controller's state machine
- TDI - a pulled-up Test Data Input that is sampled on the rising edge of TCK
- TDO - a three-stateable Test Data Output that changes on the falling edge of TCK during the ShiftIR and ShiftDR TAP controller states.
- TRST - a pulled-up, active low, asynchronous reset that provides the TAP initialization

Pull-up and pull-down resistors are to be added by the user (not implemented).

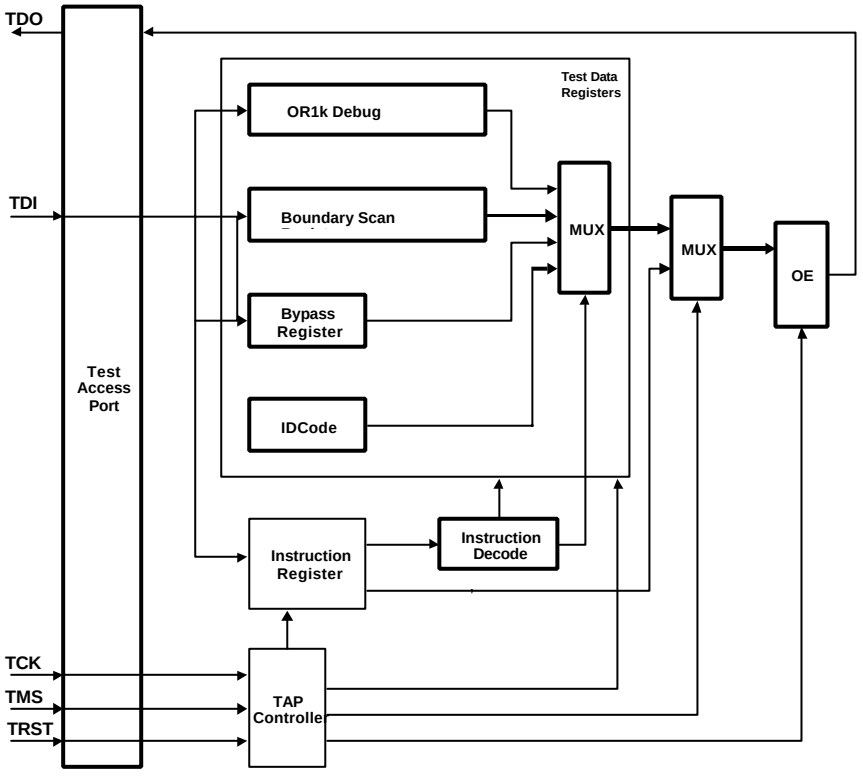

Figure 1: Test Logic Block Diagram

The TAP Controller and Operation

The TAP is controlled by the test clock (TCK) and test mode select (TMS) inputs. These two inputs determine whether an instruction register scan or data register scan is performed. The TAP consists of a small controller design, driven by the TCK input, which responds to the TMS input as shown in the **state diagram in Figure 2**. The IEEE Std 1149.1 test bus uses both clock edges of TCK. TMS and TDI are sampled on the rising edge of TCK, while TDO changes on the falling edge of TCK.

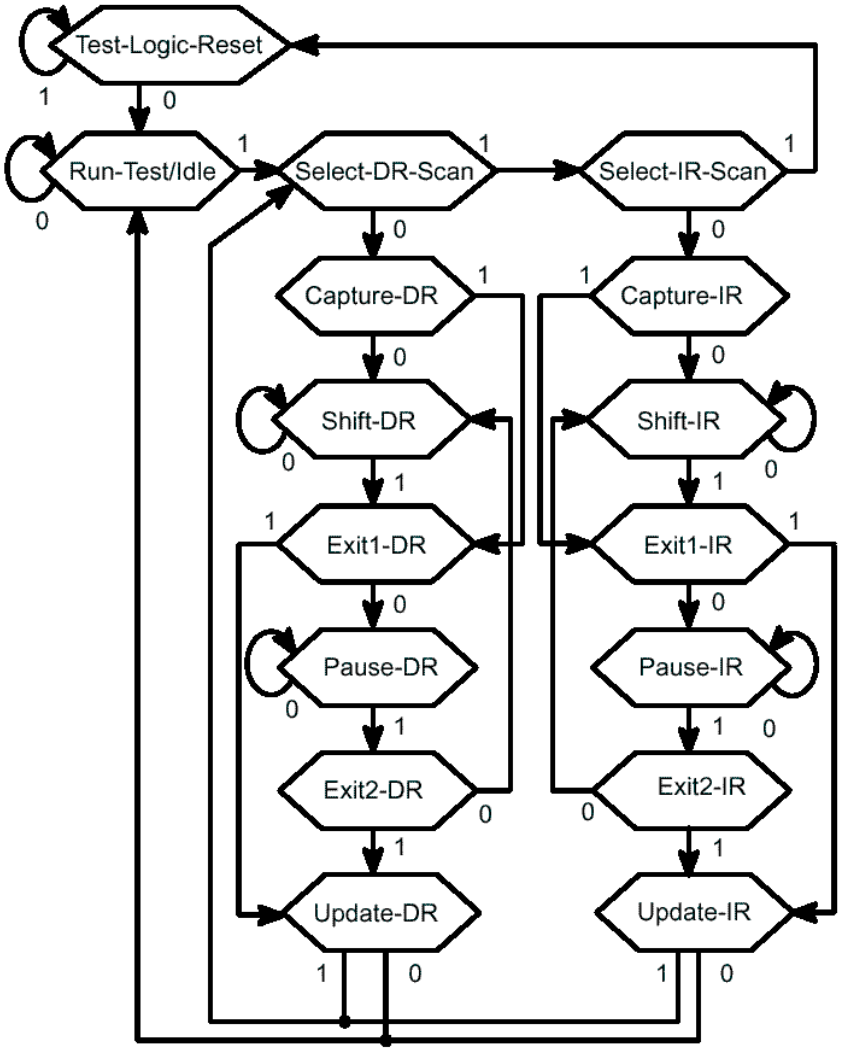

Figure 2: TAP Controller State Diagram

The main state diagram consists of six steady states: Test-Logic-Reset, Run-Test/Idle, Shift-DR, Pause-DR, Shift-IR, and Pause-IR. A unique feature of this protocol is that only one steady state exists for the condition when TMS is set high: the Test-Logic-Reset state. This means that a reset of the test logic can be achieved within five TCKs or less by setting the TMS input high. At power up, or during normal operation of the host IC, the TAP is forced into the Test-Logic-Reset state by driving TMS high and applying five or more TCKs. In this state, the TAP issues a reset signal that places all test logic in a condition that does not impede normal operation of the host IC. When test access is required, a protocol is applied via the TMS and TCK inputs, causing the TAP to exit the Test-Logic-Reset state and move through the appropriate states. From the Run-Test/Idle state, an instruction register scan or a data register scan can be issued to transition the TAP

through the appropriate states shown in **Figure 2**. The states of the data register scan and instruction register scan blocks are mirror images of each other, adding symmetry to the protocol sequences. The first action that occurs when either block is entered is a capture operation. For the data registers, the Capture-DR state is used to capture (or parallel load) the data into the selected serial data path. If the boundary scan register is the selected data register, the normal data inputs are captured during this state. In the instruction register, the Capture-IR state is used to capture status information into the instruction register. From the Capture state, the TAP transitions to either the Shift or Exit1 state. Normally, the Shift state follows the Capture state so that test data or status information can be shifted out for inspection and new data shifted in. Following the Shift state, the TAP either returns to the Run-Test/Idle state via the Exit1 and Update states or enters the Pause state via Exit1. The reason for entering the Pause state is to temporarily suspend the shifting of data through either the selected data register or instruction register while a required operation, such as refilling a tester memory buffer, is performed. From the Pause state, shifting can resume by re-entering the Shift state via the Exit2 state or be terminated by entering the Run-Test/Idle state via the Exit2 and Update states. Upon entering the data register scan or instruction register scan blocks, shadow latches in the selected scan path are forced to hold their present state during the capture and shift operations. The data being shifted into the selected scan path is not output through the shadow latch until the TAP enters the Update-DR or Update-IR state. The Update state causes the shadow latches to update (or parallel load) with the new data that has been shifted into the selected scan path.

The Boundary Scan Register

To get an idea how to implement a boundary scan register, eight pins are connected to the IC (two input pins, two bi-directional pins and four output pins (two can be tristated)). The scan chain implementation consists of 14 boundary scan cells (bits). It is fully IEEE 1491.1 compliant. This 14-bit register is connected between TDI and TDO when **extest** or **sample/preload** instructions are selected. It is used for capturing signal pin data on the input pins, forcing fix values on the output signal pins and selecting the direction and drive characteristics (a logic value or high impedance) of the bi-directional and three-state signal pins. Figures 3, 4 and 5 depict various types of cells (input call, output cell and control cell).

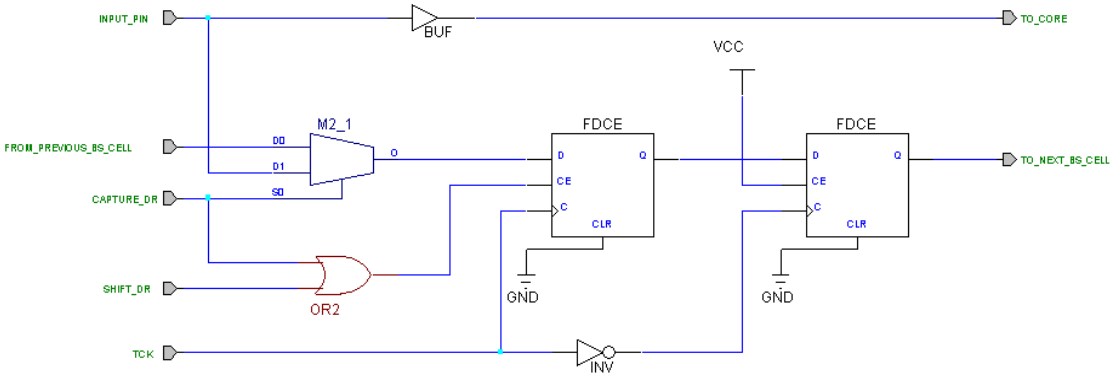

Figure 3: Input Cell

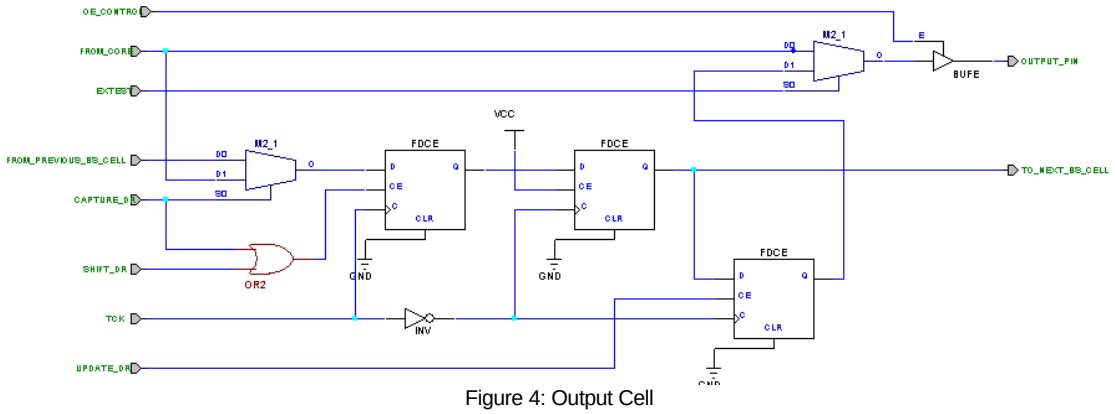

Figure 4: Output Cell

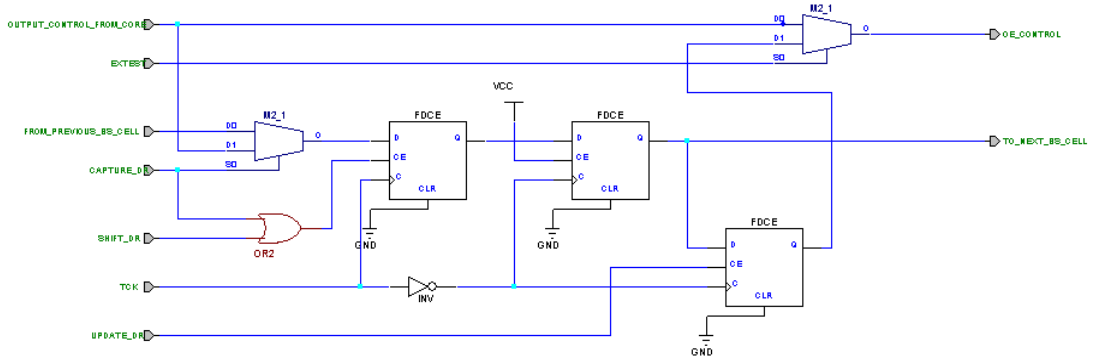

Figure 5: Control Cell

Control Cell controls the Output Cell's drive characteristics (a logic value or high impedance). Control Cells can control one or more Output Cells. Bi-directional Cells are built from an Input Cell, Output Cell and a Control Cell. It is very important to know the boundary scan bit order and the pins that are associated with them. Next table shows the Boundary Scan Bit Definition. There are 6 columns in the table: bit (cell) position within the boundary scan register, cell name, pin name, cell type (input, output or control), pin type (input, output, tristated output, bi-directional) and control cell that controls the associated output or bi-directional pin. The bit closest to the TDO is marked with number 0. The last bit in the table is the one closest to the TDI.

Bit	Cell Name	Pin Name	Cell Type	Pin Type	Output Control Cell
0	BS0	InputPin[0]	I	I	-
1	BS1	InputPin[1]	I	I	-
2	BS2	Output3Pin[0]	O	O	BS3
3	BS3	-	CTRL	-	-
4	BS4	Output3Pin[1]	O	O	BS5
5	BS5	-	CTRL	-	-
6	BS6	BidirectionalPin[0]	I	I/O	-
7	BS7	BidirectionalPin[0]	O	I/O	BS8
8	BS8	-	CTRL	-	-
9	BS9	BidirectionalPin[1]	I	I/O	-
10	BS10	BidirectionalPin[1]	O	I/O	BS11
11	BS11	-	CTRL	-	-
12	BS12	Output2Pin[0]	O	O	-
13	BS13	Output2Pin[1]	O	O	-

Table 1: Boundary Scan Bit Definition

The Instruction Register

IC's implementation includes all IEEE 1149.1 required instruction (bypass, extest, sample/preload) and optional instructions (idcode). More instructions are to be included in the next release of Boundary-Scan Architecture.

Instruction	Code
	IR[3:0]
EXTEST	0000
SAMPLE/PRELOAD	0001
IDCODE	0010
BYPASS	1111

Table 2: Instruction Decoding

EXTEST Instruction
The required EXTEST instruction places the IC 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 register is accessed to drive test data off-chip via the boundary outputs and receive test data off-chip via the boundary inputs. The bit code of this instruction is defined as all zeroes by IEEE Std 1149.1 .

SAMPLE/PRELOAD Instruction
The required SAMPLE/PRELOAD instruction allows the IC to remain in its functional mode and selects the boundary-scan register to be connected between TDI and TDO. 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 IC. This instruction is also used to preload test data into the boundary-scan register before loading an EXTEST instruction. The bit code for this instruction is defined by the vendor.

IDCODE Instruction
The optional IDCODE instruction allows the IC to remain in its functional mode and selects the optional device identification register to be connected between TDI and TDO. The device identification register (see Figure 3-8) is a 32-bit shift register containing information regarding the IC manufacturer, device type, and version code. Accessing the device identification register does not interfere with the
operation of the IC. Also, access to the device identification register should be immediately available, via a TAP data-scan operation, after power-up of the IC or after the TAP has been reset using the optional TRST* pin or by otherwise moving to the Test-Logic-Reset state. The bit code of this instruction is defined by the vendor.

HIGHZ Instruction (not implemented)
The optional HIGHZ instruction sets all outputs (including two-state as well as three-state types) of an IC to a disabled (high-impedance) state and selects the bypass register to be connected between TDI and TDO. During this instruction, data can be shifted through the bypass register from TDI to TDO without affecting the condition of the IC outputs. The bit code of this instruction is defined by the vendor.

BYPASS Instruction
The required BYPASS instruction allows the IC to remain in a functional mode and selects the bypass register to be connected between TDI and TDO. The BYPASS instruction allows serial data to be transferred through the IC from TDI to TDO without affecting the operation of the IC. The bit code of this instruction is defined as all ones by IEEE Std 1149.1 .

BSDL file (Boundary-Scan Description Language)

BSDL file can be obtained from the OpenCores homepage (www.opencores.org). The BSDL file and the Boundary Scan Architecture were tested with the JTAG Technologies testing equipment and are fully operational. Detailed description of the bsdl file is within the file itself.