

February 2002

Introduction

Lattice is the leading supplier of In-System Programmable (ISP™) devices and devices that are fully compliant with the IEEE-1149.1 testability standard. The Lattice product offering includes many devices that incorporate in-system programmability through an 1149.1 compliant test access port (TAP). The ispLSI® 1000EA, 2000VE, 2000VL, 5000V, 8000/V, ispMACH™ 4A, 4000B/C, 5000VG, MACH® 4 and 5, ispGDXV and ispGDX® device families implement 1149.1 testability and are fully compliant with the 1149.1 standard. The ispLSI 2000E, 2000V, ispGAL®22LV10, MACH 1SP, 2SP and ispPAC® device families offer in-system programmability, but do not include 1149.1 boundary scan registers and are therefore considered 1149.1 compatible rather than compliant.

In-system programming was developed to simplify the use of programmable devices packaged in fine pitch packaging (e.g. PQFP, TQFP and BGA). A manufacturing flow that doesn't take advantage of ISP devices requires additional handling, increasing the probability that delicate leads will be damaged and decreasing overall manufacturing yield. Over the past several years, the use of ISP devices has increased and all new devices can be programmed in-system.

There are very few limitations placed on the type of system can be used to execute an ISP device algorithm. Today, most programmable logic companies offer programming solutions that range from programming a single device through a simple cable attached to a computer, to programming multiple devices from different vendors as part of a board test program. Embedded programming, the ability to program devices using a microprocessor on the same board as the devices being programmed, is also readily available and gives users the ability to update the programming of a device in the field.

Benefits of ISP Through JTAG

In-system programming using a standard boundary scan test interface is necessary for compatibility with advanced board testing techniques. The IEEE 1149.1 boundary scan test interface standard, sponsored by the Joint Test Action Group (JTAG), was developed to test printed circuit board connections. The standard has been commonly referred to as JTAG. The standard also allows in-system programmable CPLDs to be programmed through the same interface used for test. The 1149.1 standard defines a simple, serial interface that allows for program and test of multiple devices using basic desktop tools. If a design incorporates 1149.1-ISP devices, then no separate programming interface is needed. All IEEE-1149.1 compatible or compliant devices (logic, interconnect and analog) can be used in the same scan chain.

ISP devices make design jobs easier by simplifying device configuration. Designers have the option of soldering parts directly on the board and then programming them through the TAP pins. In the design phase, ISP devices let designers implement redesigns within a few seconds by making changes directly to devices on the board. This speeds up the design process and reduces time to market.

ISP devices also offer benefits for manufacturing. Lower inventory cost is achieved because blank devices can be used for manufacturing and then programmed at test time. This eliminates the need to maintain a separate inventory part number for each programmed part and improves the manufacturing process by facilitating board connectivity testing. Once the design is finalized and the board assembled, manufacturing engineers can use testers for both board connectivity testing and programming. As a result, 1149.1-ISP eliminates the cost of separate programming stations, unnecessary manufacturing steps and excessive handling. This shortens production time, reduces scrap cost and increases reliability.

IEEE 1532 Programming Standard

IEEE 1532 refers to an industry standardization effort aimed at simplifying the challenge of programming In-System Configurable (ISC) devices using the industry standard IEEE 1149.1 Boundary Scan Test Access Port. The IEEE 1532 effort is focused on the standardization of the programming algorithm at the component level (silicon)

as well as the definition of the software required to describe the programming algorithm and associated programming data.

The silicon portion of the standard establishes a common device behavior during programming via the IEEE 1149.1 state machine. Any device which claims compatibility or compliance with this standard will behave predictably and consistently when given an ISC instruction.

The software portion of the IEEE 1532 standard defines a modified Boundary Scan Description Language file (ISC BSDL file) which has been expanded to cover the new ISC instructions. Additionally, there is a new data file (ISC data file) which contains all of the device and pattern-specific programming data.

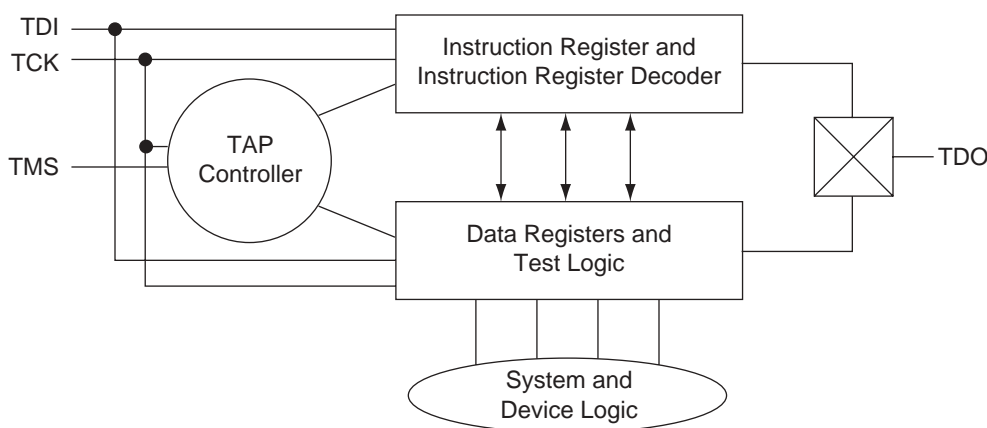
History of the 1149.1 Standard

For years, many companies have used proprietary test methodologies implemented with boundary scan registers to reduce test complexity at the board and system level. In the late 1980s, a group of European companies formed a group with the purpose of standardizing a method for implementing and performing boundary scan testability. The composition of this group included representatives from board-test companies, system design companies and semiconductor manufacturers. A year after this group was formed, additional companies from both Asia and the United States joined this group and continued to work on a standard to be voted on by the IEEE. In 1990, this standard was passed as standard IEEE 1149.1-1990. This standard included a definition for a Test Access Port (TAP), a group of both mandatory and optional test registers, a control mechanism and timing for both the registers and TAP, and a set of both mandatory and optional test instructions. In 1993, corrections and additions were made to the standard, including a language that can be used to describe its implementation in a given device. This language, called the Boundary Scan Definition Language (BSDL), is a subset of VHDL (another IEEE standard). The 1149.1 working group continues to meet on a regular basis and constantly works to improve the standard.

What is IEEE-1149.1?

In its simplest form, the 1149.1 standard is implemented using a four-pin, dedicated test access port, a 16-state, synchronous state machine and a group of data registers. The data registers include the bypass register and a boundary scan register that is used to control the inputs and outputs of the device being tested. It also needs an instruction register and instruction register decoder used to control the data registers. Figure 1 shows a top-level diagram of a basic implementation of the 1149.1 standard.

Figure 1. IEEE 1149.1 Block Diagram



There are four pins that make up the Test Access Port (TAP): TDI (Test Data Input), TMS (Test Mode Select), TCK (Test Clock), and TDO (Test Data Output). An additional pin defined by the standard, TRST (Test ReSeT), can be used to asynchronously reset both the TAP controller and the instruction register. All registers, along with the TAP controller, are clocked using the TCK pin.

Boundary Scan TAP Controller

The TAP Controller is a synchronous, finite state machine that controls both the TAP and the instruction and various data registers. It controls whether a device is in reset mode, where the core logic has full control of the device, if it is receiving an instruction, receiving and/or transmitting data, or is in an idle state. The state machine, as illustrated in Figure 2, is controlled by TMS and clocked by TCK. The value of TMS is located next to each transition in Figure 2.

Boundary Scan Test Instructions

While the TAP controller is the heart of any 1149.1 implementation, the instruction register and instruction register decoder can be thought of as the brains. The instruction register stores information concerning which test register or test circuitry is active. For any instruction code selected, an associated register and/or test circuit is also selected. This is one of the requirements stated in the 1149.1 standard. Instructions are shifted into the instruction register when the TAP controller is in the SHIFT-IR state and become active when the controller enters the UPDATE-IR state.

Figure 2. TAP Controller

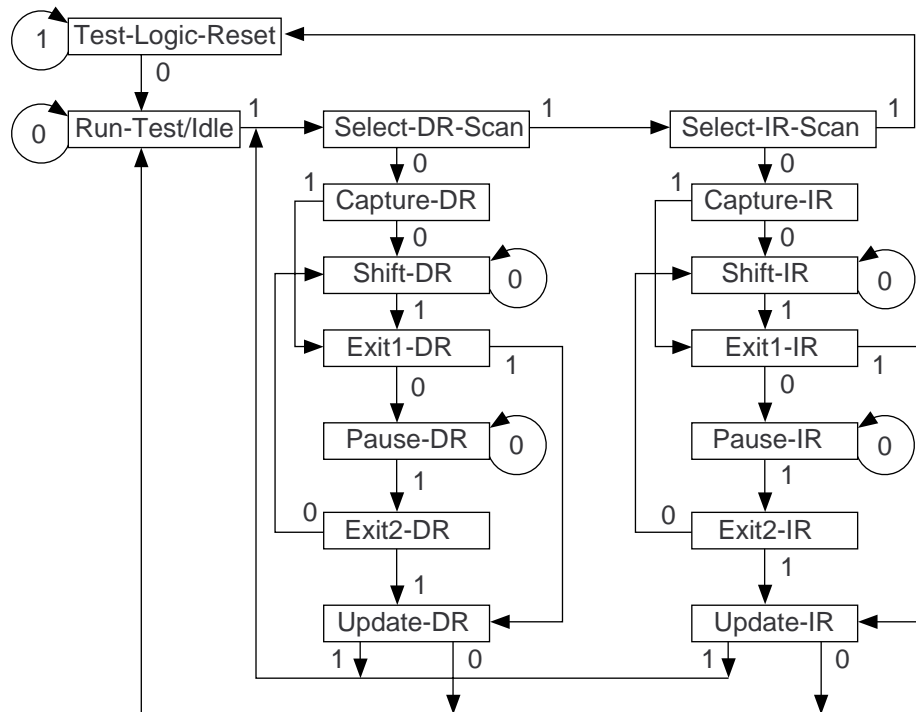


Table 1 shows the boundary scan test instructions supported by the Lattice device families that use the IEEE 1149.1 TAP controller state machine for programming. Only the ispLSI 2000E, 2000V, ispPAC and ispGAL22LV10 device families do not have boundary scan registers to support test.

All of the above instructions have a unique, 5- or 8-bit code that is shifted into the instruction register. The exception to this is the BYPASS instruction that will turn on whenever its own code is selected or when an invalid code is selected. Individual test instruction codes can be found in the device BSDL file.

Lattice ispLSI 1000EA, 2000VE/VL, 5000V, 8000, ispMACH 4A, 4000B/C, 5000VG, MACH 4 and 5, ispGDX and ispGDXV devices support the three mandatory instructions defined by the boundary scan definition: SAMPLE/PRELOAD, EXTEST and BYPASS. The following paragraphs describe each of these instructions. A shift register is defined within the devices to implement the instruction shift register. For the devices that do not support boundary scan test, loading the SAMPLE/PRELOAD and EXTEST instructions results in the BYPASS function.

Table 1. Summary of Boundary Scan Test Instruction Support

Boundary Scan Test Instruction	Required	Optional	ispLSI 1000EA	ispLSI 2000E	ispLSI 2000VE	ispLSI 2000V	ispLSI 2000VL	ispLSI 5000V	ispLSI 8000V	ispMACH 4A/MACH 4	MACH 5	ispMACH 4000B/C	ispMACH 5000VG	ispGDXV	ispGDX	ispGAL22LV10	ispPAC
BYPASS	✓		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
EXTEST	✓		✓		✓		✓	✓	✓	✓	✓	✓	✓	✓	✓		
SAMPLE	✓		✓		✓		✓	✓	✓	✓	✓	✓	✓	✓	✓		
HIGHZ		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓	
IDCODE		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
USERCODE		✓	✓	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓		
Instruction Register Length			5	5	5	5	5	5	5	6	6	8	8	5	5	5	5

The SAMPLE/PRELOAD instruction is used to take a snapshot of the device as it is in normal functional operation. This instruction does not interfere with the normal operation of the device. The SAMPLE part of this instruction occurs during Capture-DR. The DRs are loaded with the state of the pins on the rising edge of TCK while in Capture-DR. The DR can then be shifted out TDO for examination. The PRELOAD part of this instruction is simply loading the DRs during Shift-DR. It is typically used before the EXTEST instruction to define the initial state of the pins.

The EXTEST instruction drives the external pins with the values present in the DR. Functional logic is interrupted while running this instruction. This instruction is used to test external devices or board-level connections. The values of the pins are loaded in the DR on the rising edge of TCK while in Capture-DR. New values are shifted into the DR during Shift-DR. The new values are latched to the pins during Update-DR.

The BYPASS instruction is used to bypass any device that is not accessed during any part of the test. The definition of the BYPASS instruction allows TDI not to be driven during Shift-IR. In order to shift in the correct instruction code, the TDI pin has an internal pull-up to drive logic high. A bypassed boundary scan device has a single bypass register between TDI and TDO while in Shift-DR.

The optional instructions, HIGHZ, IDCODE and USERCODE, are supported by some Lattice devices. For the devices that do not support these instructions, loading them results in the BYPASS function. The following paragraphs discuss these instructions in more detail.

The HIGHZ instruction is used to disable the functional I/O pins. The I/O pins tri-state on the rising edge of TCK in Update-IR with HIGHZ loaded. The BYPASS register is selected in Shift-DR when this instruction is loaded. The I/O pins are re-enabled on the rising edge of TCK in Update-IR with any other standard boundary scan test instruction loaded.

The IDCODE instruction gives access to the hard-wired 32-bit boundary scan IDCODE. There are two ways to access the IDCODE register, one by state and one by instruction. By IEEE 1149.1, moving directly from Test-Logic-Reset to Shift-DR will select the IDCODE register. In addition, loading the IDCODE instruction in Shift-IR will select the IDCODE register in Shift-DR. All IDCODEs can be found in the individual device BSDL file.

The USERCODE instruction gives access to the user programmable 32-bit boundary scan USERCODE. The USERCODE has proven to be useful for version control, especially with embedded applications. The user can program a version number into the USERCODE register. At power up, the on-board processor or control unit can verify that the version is correct in the USERCODE register. Since the USERCODE can be read while the device is functional, no interruption in the functional logic is necessary.

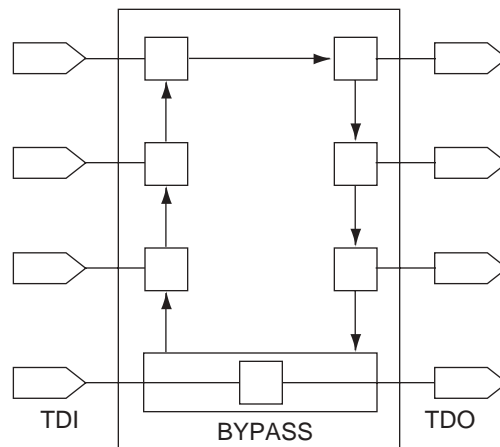
BSDL Support Files

Boundary Scan Description Language (BSDL) files are used to describe an individual device's boundary scan hardware configuration. Each BSDL file shows the instructions supported and the different data shift registers lengths and configurations. Individual BSDL files can be found on the Lattice web site at www.latticesemi.com.

Data Registers

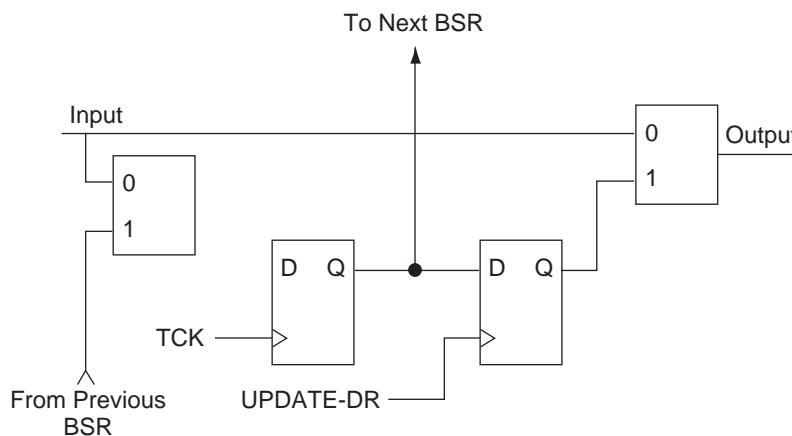
Two data registers have been defined by the 1149.1 standard and are considered a requirement. These are the BYPASS register and the boundary scan register (BSR). The BYPASS register is a single-bit register used to shift data from TDI to TDO without affecting other circuitry. Figure 3 illustrates the BYPASS register.

Figure 3. BYPASS Register



The boundary scan register (BSR) is used to capture or send data from the I/O and input pins. Each boundary scan register is composed of two registers. The first is used to either capture data from the pin or shift data into and out of it from the TAP. The second is used to drive data from the first register onto an input or I/O pin. Figure 4 shows the structure of a typical BSR.

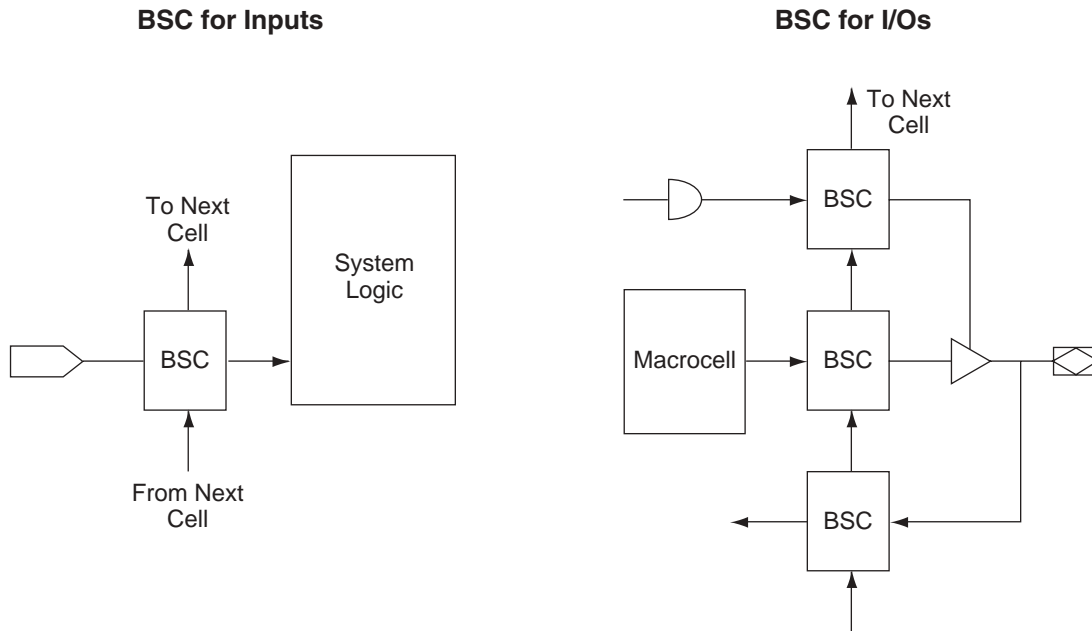
Figure 4. Boundary Scan Register



A common boundary scan register implementation uses three boundary scan control registers attached to it. The first is for input, the second for output and the third for the output enable signal. By looking at these three registers, test software can determine what is happening at a particular I/O. If the output enable is a '1' then the I/O pin will match the value of the output cell. If the output enable is '0,' the I/O pin is configured as an input with the value of the data in the input BSR. An input or clock pin would only have a single BSR and would not have the output tied to

anything, as it is used for observation only. Figure 5 shows the BSR configurations for this implementation for both the input pin and the I/O pin.

Figure 5. Common BSR Configurations



For a device to be considered 1149.1 compliant, it must have the TAP, TAP controller, a boundary register and the instructions BYPASS, SAMPLE and EXTEST. A device that only has the TAP and TAP controller may be compatible with the 1149.1 standard and may work in a scan chain, but it will not be considered compliant. Any device that does not have a boundary scan register cannot be tested using the TAP because there is no means of controlling and accessing the I/O and input pins other than a direct connection. All the ispLSI 1000EA, 2000VE, 5000V, 8000/V, ispMACH 4A, 4000B/C, 5000VG, MACH 4 and 5, and ispGDX/V families are compliant with the 1149.1 standard while the ispLSI 2000E, 2000V, ispGAL22LV10 and ispPAC families are compatible with it.

Technical Support Assistance

Hotline: 1-800-LATTICE (Domestic)
 1-408-826-6002 (International)
 e-mail: techsupport@latticesemi.com