# The Basics of ISP

## Introduction

This section describes the details of programming with Lattice's In-System Programmable (ISP™) devices. It is organized into three sections. The first section summarizes the ISP design flow. The next section describes ISP hardware interface basics, including discussions on both key issues required to get started with ISP quickly, as well as detailed ISP information for those interested in a thorough understanding of ISP at the device level. The final section focuses on ISP software, which summarizes all the development tools available to support easy implementation of the Lattice ISP solutions.

## ISP Design Flow

As with other Programmable Logic Devices (PLDs), the ISP design flow includes design entry using CAD software, compiling and fitting the design, generating a JEDEC standard fuse map file, and programming the device (Figure 1).

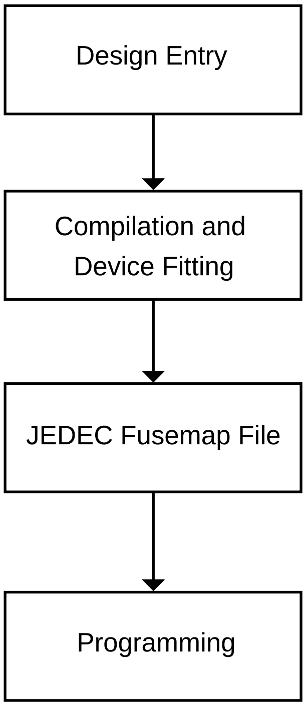
### Creating a JEDEC Fuse map File

As part of any PLD design flow, the logic design must be entered through any combination of VHDL, schematic, Boolean equation, state machine, or truth table design entry. Lattice has various third-party and proprietary software packages which support these design methodologies. Each of these design packages will take a design and generate a standard JEDEC fuse map for programming. Up to this point in the logic design process, ISP devices share the same design flow as standard programmable logic devices.

### Programming

Programming consists of converting the JEDEC fuse map file into a serial data format and shifting that serial data into the device. The ISP programming software automatically converts the JEDEC fuse map file into the serial data format which is programmed into the ISP device. However, a JEDEC file is made up of ASCII characters which use a relatively large amount of space, especially in environments in which storage space for the fuse map information is limited. In order to support these storage-critical environments, Lattice has defined an ispSTREAM™ data format which represents each fuse location with a single bit instead of an ASCII character. The ISP programming software also accepts this ispSTREAM format for programming.

**Figure 1. ISP Design Flow**



Once the fuse map is ready, it is just a matter of serially shifting the data into the device along with the appropriate addresses and commands. The basic ISP interface uses four wires to shift the JEDEC fuse map data into the device. An additional fifth wire is used by ispLSI devices, employing an active low ISP Enable ($\overline{\text{ispEN}}$) signal as a mode control to put the device into programming or normal operation mode.

ISP programming is controlled internally by a state machine. The user has the option of programming through a Lattice proprietary programming state machine or an IEEE 1149.1 standard Test Access Port (TAP) programming interface. Both options control the state machine for programming through MODE/TMS, SDI/TDI and SCLK/TCK signals. The programming data is shifted in and out from the SDI/TDI and SDO/TDO signals. ISP commands such as Program, Bulk Erase, Verify, Data Shift and Address Shift are executed through the device's instruction registers and ISP interface input pins.

# The Basics of ISP

Traditionally, programmable logic devices have been programmed on PLD/PROM programmers which require that all programming signals and algorithms be generated by the programmer. The programmer also generates the external super voltage or high voltage required by non-ISP devices (typically 12-14 volts). This super voltage requirement is one of the main reasons dedicated programmers are used to program conventional PLDs.

However, with ISP devices, the ISP programming super voltage is generated within the device from the 3.3 or 5-volt power supply. This internal super voltage generation teamed with Lattice's unique serial ISP programming interface enables designers to program any ISP device using a simple four- or five-wire interface in which all the programming signals are driven by standard TTL logic levels.

The details of device programming are transparent to the user if ISP programming software such as ISP Daisy Chain Download and ispCODE™ C Source Code are used. These software tools drive the four or five ISP programming signals in accordance with the programming specifications and the state machine requirements. The ISP Daisy Chain Download software generates the ISP signals with proper timing through the use of the PC parallel port. ispCODE can be ported to any hardware platform required and can be used to generate ISP programming signals using whatever hardware is available.

Lattice also supports the use of other programming interfaces such as Automatic Test Equipment (ATE). Specifically for testers, Lattice provides JEDEC file conversion routines, such as ispATE, to tester-acceptable formats. In addition, testers which accept high-level languages can be programmed using ispCODE C routines as a model for structuring test programs.

These and other topics are covered in the next two sections, Hardware Basics and Software Basics.

Lattice Semiconductor Corporation (LSC) products are made under one or more of the following U.S. and international patents: 4,761,768 US, 4,766,569 US, 4,833,646 US, 4,852,044 US, 4,855,954 US, 4,879,688 US, 4,887,239 US, 4,896,296 US, 5,130,574 US, 5,138,198 US, 5,162,679 US, 5,191,243 US, 5,204,556 US, 5,231,315 US, 5,231,316 US, 5,237,218 US, 5,245,226 US, 5,251,169 US, 5,272,666 US, 5,281,906 US, 5,295,095 US, 5,329,179 US, 5,331,590 US, 5,336,951 US, 5,353,246 US, 5,357,156 US, 5,359,573 US, 5,394,033 US, 5,394,037 US, 5,404,055 US, 5,418,390 US, 5,493,205 US, 0194091 EP, 0196771B1 EP, 0267271 EP, 0196771 UK, 0194091 GB, 0196771 WG, P3686070.0-08 WG. LSC does not represent that products described herein are free from patent infringement or from any third-party right.

The specifications and information herein are subject to change without notice. Lattice Semiconductor Corporation (LSC) reserves the right to discontinue any product or service without notice and assumes no obligation to correct any errors contained herein or to advise any user of this document of any correction if such be made. LSC recommends its customers obtain the latest version of the relevant information to establish, before ordering, that the information being relied upon is current.

LSC warrants performance of its products to current and applicable specifications in accordance with LSC's standard warranty. Testing and other quality control procedures are performed to the extent LSC deems necessary. Specific testing of all parameters of each product is not necessarily performed, unless mandated by government requirements.

LSC assumes no liability for applications assistance, customer's product design, software performance, or infringements of patents or services arising from the use of the products and services described herein.

LSC products are not authorized for use in life-support applications, devices or systems. Inclusion of LSC products in such applications is prohibited.

LATTICE SEMICONDUCTOR CORPORATION
5555 Northeast Moore Court
Hillsboro, Oregon 97124 U.S.A.
Tel.: (503) 681-0118
FAX: (503) 681-3037
http://www.latticesemi.com

November 1996