



# 1. Configuring Altera FPGAs

CF51001-2.2

## Introduction

Stratix® series, Cyclone™ series, APEX™ II, APEX 20K (including APEX 20KE and APEX 20KC), Mercury™, ACEX® 1K, FLEX® 10K (including FLEX 10KE and FLEX 10KA), and FLEX 6000 devices can be configured using one of seven configuration schemes. Table 1–1 shows which device families support which configuration schemes.

**Table 1–1. Configuration Scheme Device Family Support**

Configuration Scheme	Device Family													
	Stratix IV	Stratix III	Stratix II, Stratix II GX	Stratix, Stratix GX	Arria GX	Cyclone III	Cyclone II	Cyclone	APEX II	APEX 20K, APEX 20KE,	Mercury	ACEX 1K	FLEX 10K, FLEX 10KE,	FLEX 6000
Passive Serial (PS)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Active Serial (AS)	✓	✓	✓	—	✓	✓	✓	✓	—	—	—	—	—	—
Active Parallel (AP)	—	—	—	—	—	✓	—	—	—	—	—	—	—	—
Fast Passive Parallel (FPP)	✓	✓	✓	✓	✓	✓	—	—	✓	—	—	—	—	—
Passive Parallel Synchronous (PPS)	—	—	—	—	—	—	—	—	—	✓	✓	✓	✓	—
Passive Parallel Asynchronous (PPA)	—	—	✓	✓	✓	—	—	—	✓	✓	✓	✓	✓	—
Passive Serial Asynchronous (PSA)	—	—	—	—	—	—	—	—	—	—	—	—	—	✓
Joint Test Action Group (JTAG)	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	(1)

**Note to Table 1–1:**

- (1) Although you cannot configure FLEX 6000 devices through the JTAG pins, you can perform JTAG boundary-scan testing.

All configuration schemes use either an intelligent host or a configuration device(s) (Table 1–2).

<b>Configuration Scheme</b>	<b>Typical Use</b>
Passive Serial (PS)	Configuration with the enhanced configuration devices (EPC16, EPC8, and EPC4), EPC2, EPC1, EPC1441 configuration devices, serial synchronous microprocessor interface, the USB Blaster USB Port Download Cable, MasterBlaster™ communications cable, ByteBlaster™ II parallel download cable or ByteBlasterMV™ parallel port download cable.
Active Serial (AS)	Configuration with the serial configuration devices (EPCS1, EPCS4, EPCS16, EPCS64, and EPCS128).
Active Parallel (AP)	Configuration with the parallel flash memory which 16 bits of configuration data are loaded on every clock cycle. Sixteen times faster than AS.
Passive Parallel Synchronous (PPS)	Configuration with a parallel synchronous microprocessor interface.
Fast Passive Parallel (FPP)	Configuration with an enhanced configuration device or parallel synchronous microprocessor interface where 8 bits of configuration data are loaded on every clock cycle. Eight times faster than PPS.
Passive Parallel Asynchronous (PPA)	Configuration with a parallel asynchronous microprocessor interface. In this scheme, the microprocessor treats the target device as memory.
Passive Serial Asynchronous (PSA)	Configuration with a serial asynchronous microprocessor interface.
Joint Test Action Group (JTAG)	Configuration through the IEEE Std. 1149.1 (JTAG) pins. (1)

The following chapters discuss how to configure one or more Stratix series, Arria GX series, Cyclone series, APEX II, APEX 20K (including APEX 20KE and APEX 20KC), Mercury, ACEX 1K, FLEX 10K (including FLEX 10KE and FLEX 10KA), and FLEX 6000 devices. The following chapters should be used in conjunction with the following documents:

- Stratix IV (E and GX) Device Handbook
- Stratix III Device Handbook
- Cyclone III Device Handbook
- Arria GX Device Handbook
- Stratix II GX Device Handbook
- Stratix II Device Handbook
- Stratix Device Handbook
- Stratix GX Device Handbook

- Cyclone II Device Handbook
- Cyclone Device Handbook
- APEX II Programmable Logic Device Family Data Sheet
- APEX 20K Programmable Logic Device Family Data Sheet
- APEX 20KC Programmable Logic Device Data Sheet
- Mercury Programmable Logic Device Family Data Sheet
- ACEX 1K Programmable Logic Device Family Data Sheet
- FLEX 10K Embedded Programmable Logic Family Data Sheet
- FLEX 10KE Embedded Programmable Logic Family Data Sheet
- FLEX 6000 Programmable Logic Device Family Data Sheet

Volume I covers how to configure Altera FPGAs, where each chapter covers a different device family. Each subsection describes how to configure the devices with the following configuration schemes:

- PS Configuration
  - Using a Configuration Device
  - Using a Microprocessor
  - Using a Download Cable
- AS Configuration
- AP Configuration
- FPP Configuration
  - Using an enhanced Configuration Device
  - Using a Microprocessor
- PPS Configuration
- PPA Configuration
- PSA Configuration
- JTAG Programming and Configuration

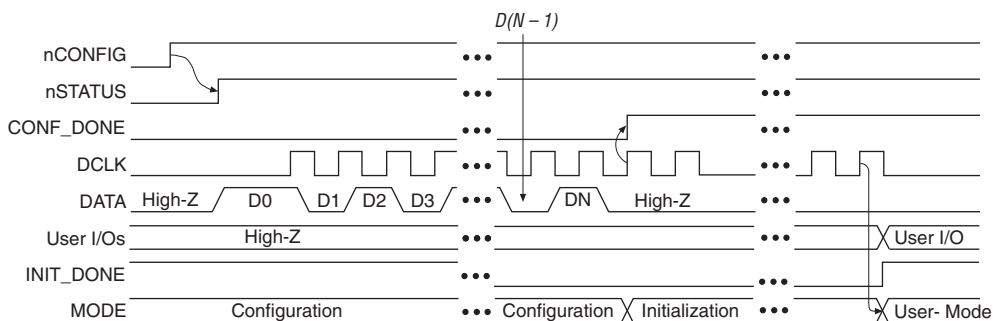
Volume II contains information that is relevant for all Altera FPGAs discussed in this handbook. Information about configuration devices and combining different Altera device families in the same configuration chain can be found in this volume.

## Device Configuration Overview for Passive Schemes

During device operation, Altera FPGAs store configuration data in SRAM cells. Because SRAM memory is volatile, the SRAM cells must be loaded with configuration data each time the device powers up. After the device is configured, its registers and I/O pins must be initialized. After initialization, the device enters user mode for in-system operation.

Figure 1-1 shows the waveform of the configuration pins during configuration, initialization, and user-mode.

Figure 1-1. Configuration Cycle Waveform



The low-to-high transition of `nCONFIG` on the FPGA begins the configuration cycle. The configuration cycle consists of 3 stages: reset, configuration, and initialization. While `nCONFIG` is low, the device is in reset. When the device comes out of reset, `nCONFIG` must be at a logic high level in order for the device to release the open-drain `nSTATUS` pin. Once `nSTATUS` is released, it is pulled high by a pull-up resistor and the FPGA is ready to receive configuration data. Before and during configuration all user I/O pins are tri-stated. Stratix series, Cyclone series, APEX II, APEX 20K, Mercury, ACEX 1K, and FLEX 10KE devices have weak pull-up resistors on the I/O pins which are on before and during configuration.

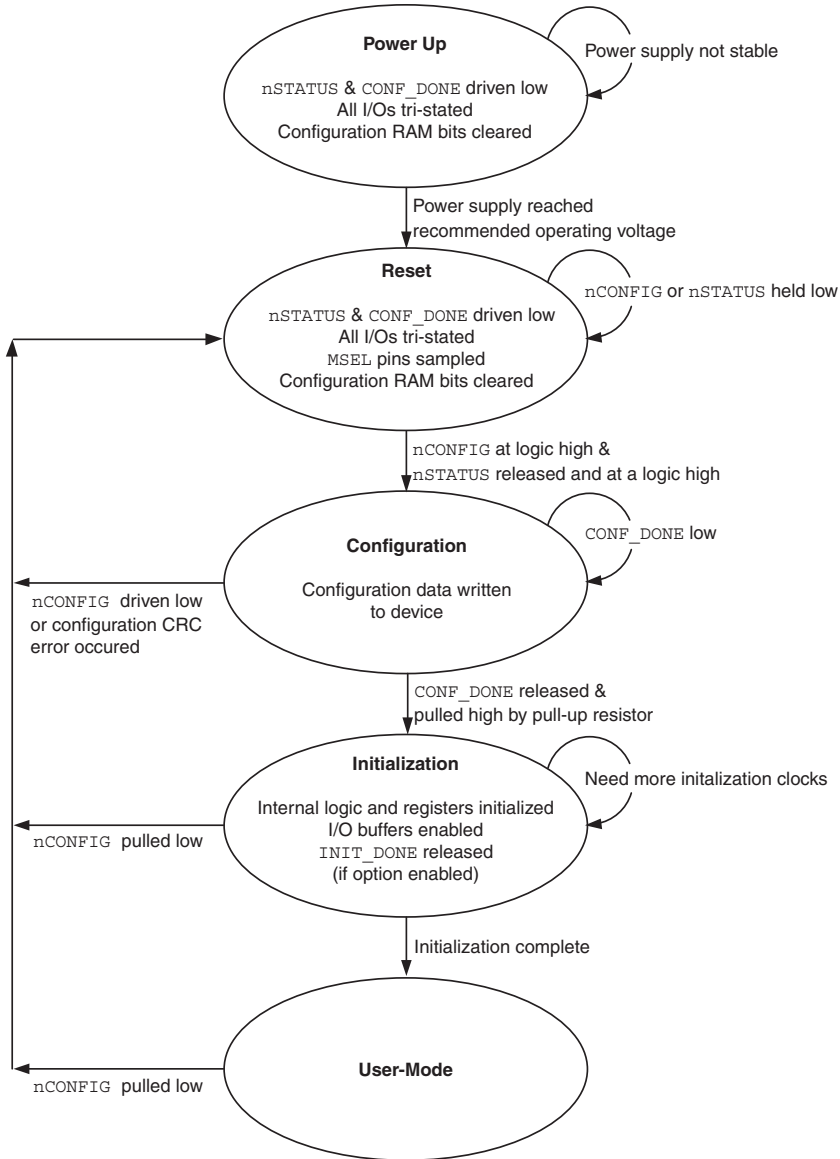
`nCONFIG` and `nSTATUS` must be at a logic high level in order for the configuration stage to begin. Configuration can be delayed by holding the `nCONFIG` low. The device receives configuration data on its `DATA` pin(s) and (for synchronous configuration schemes) the clock source on the `DCLK` pin. Configuration data is latched into the FPGA on the rising edge of `DCLK`. After the FPGA has received all configuration data successfully it releases the `CONF_DONE` pin, which is pulled high by a pull-up resistor. A low to high transition on `CONF_DONE` indicates configuration is complete and initialization of the device can begin.

An optional `INIT_DONE` pin is available, which signals the end of initialization and the start of user-mode. During initialization, internal logic, internal and I/O registers are initialized and I/O buffers are enabled. When initialization is finished, the `INIT_DONE` pin is released and pulled high by an external pull-up resistor. Once in user-mode, the user I/O pins will no longer have a weak pull-up and will function as assigned in your design. The `DCLK`, `DATA` (FLEX 6000), and `DATA0` (Stratix series, Cyclone series, APEX II, APEX 20K, Mercury, ACEX 1K, and FLEX 10KE) pins should not be left floating after configuration; they should be driven high or low, whichever is convenient, on your board.

A reconfiguration is initiated by toggling the `nCONFIG` pin from high to low and then back to high. When `nCONFIG` is pulled low, `nSTATUS` and `CONF_DONE` are also pulled low and all I/O pins are tri-stated. Once `nCONFIG` and `nSTATUS` return to a logic high level, configuration begins.

Figure 1–2 shows a simple state diagram of the configuration process.

Figure 1–2. Configuration Cycle State Machine



## Selecting a Configuration Scheme

The configuration data for Altera devices can be loaded using an active, passive or JTAG configuration scheme. When using an active configuration scheme with a serial configuration device, the target FPGA generates the control and synchronization signals. When both devices are ready to begin configuration, the serial configuration device sends data to the FPGA.

When using any passive configuration scheme, the Altera device is incorporated into a system with an Altera configuration device or an intelligent host, such as a microprocessor, that controls the configuration process. The configuration device or host supplies configuration data from a storage device (a configuration device(s), a hard disk, RAM, or other system memory). When using passive configuration, you can change the target device's functionality while the system is in operation by reconfiguring it.

Altera devices support a number of configuration schemes. Not all device families support all configuration schemes. [Table 1-1](#) and the individual device family sections should be referenced to determine if your target device family supports your intended configuration scheme. Once you have decided on the appropriate configuration scheme for your system, you will need to drive the dedicated mode select control pins, *MSEL*, of the FPGA to set the configuration mode.



For further details on how to set the *MSEL* pins for your target device, refer to the appropriate device family chapters.

Below is a brief description of each configuration scheme. For detailed information, consult the appropriate sections.



For supported devices for each configuration scheme, refer [Table 1-1 on page 1](#).

### Passive Serial Configuration

PS configuration can be performed by using an Altera download cable, an Altera enhanced configuration device or configuration device, or an intelligent host, such as a microprocessor. During PS configuration, configuration data is transferred from a storage device, such as a configuration device or flash memory, to the FPGA on the *DATA* (FLEX 6000) or *DATA 0* (Stratix series, Cyclone series, APEX II, APEX 20K, Mercury, APCEX 1K, and FLEX 10K) pin. This configuration data is latched into the FPGA on the rising edge of *DCLK*. Configuration data is transferred one bit per clock cycle.

### Active Serial Configuration

AS configuration can be performed by using an Altera Serial Configuration device. During AS configuration, the FPGA device is the master and the configuration device is the slave. Configuration data is transferred to the FPGA on the `DATA 0` pin. This configuration data is synchronized to the `DCLK` input. Configuration data is transferred one bit per clock cycle.

### Active Parallel Configuration

In the AP configuration scheme, Altera FPGA devices are configured using commodity 16-bit parallel flash memory. These external non-volatile configuration devices are industry standard microprocessor flash memories. During AP configuration, the FPGA device is the master and the parallel flash memory is the slave. Configuration data is transferred to the Altera device on the `DATA [15:0]` pins. This configuration data is synchronized to the `DCLK` input. Configuration data is transferred at a rate of 16 bits per clock cycle. The `DCLK` frequency driven out by the Altera device during AP configuration is approximately 40 MHz.

### Passive Parallel Synchronous Configuration

PPS configuration can be performed by using an intelligent host, such as a microprocessor. During PPS configuration, configuration data is transferred from a storage device, such as flash memory, to the FPGA on the `DATA [7..0]` pins. This configuration data is synchronized to the `DCLK` input. On the first rising edge of `DCLK`, a byte of configuration data is latched into the FPGA. The next 8 falling edges of `DCLK` are needed to internally serialize the data in the FPGA.

### Fast Passive Parallel Configuration

FPP configuration can be performed by using an Altera enhanced configuration device, or an intelligent host, such as a microprocessor. During FPP configuration, configuration data is transferred from a storage device, such as an enhanced configuration device or flash memory, to the FPGA on the `DATA [7..0]` pins. This configuration data is latched into the FPGA on the rising edge of `DCLK`. Configuration data is transferred one byte per clock cycle.

## Passive Parallel Asynchronous Configuration

PPA configuration can be performed by using an intelligent host, such as a microprocessor. During PPA configuration, configuration data is transferred from a storage device, such as a configuration device or flash memory, to the FPGA on the `DATA [7 . . 0]` pins. Since this configuration scheme is asynchronous, control signals are used to regulate the configuration cycle.

## Passive Serial Asynchronous Configuration

PSA configuration can be performed by using an intelligent host, such as a microprocessor. During PSA configuration, configuration data is transferred from a storage device, such as a configuration device or flash memory, to the FPGA on the `DATA` pin. Since this configuration scheme is asynchronous, control signals are used to regulate the configuration cycle.

## JTAG Configuration

JTAG configuration uses the IEEE Std 1149.1 JTAG interface pins and supports the JAM STAPL standard. JTAG configuration can be performed by using an Altera download cable or an intelligent host, such as a microprocessor.

## Document Revision History

Table 1–3 shows the revision history for this document.

Date & Document Version	Changes Made	Summary of Changes
October 2008 v2.2	<ul style="list-style-type: none"> <li>• Updated Table 1–1 and Table 1–2.</li> <li>• Updated Figure 1–2.</li> <li>• Updated “Introduction”, “Device Configuration Overview for Passive Schemes”, “Selecting a Configuration Scheme”, “Passive Serial Configuration”, “Active Serial Configuration”, “Passive Parallel Synchronous Configuration”, “Fast Passive Parallel Configuration”, “Passive Parallel Asynchronous Configuration”, “Passive Serial Asynchronous Configuration”, and “JTAG Configuration” sections.</li> <li>• Added “Active Parallel Configuration” and “Document Revision History” sections.</li> </ul>	—

