MAX Series Configuration Controller Using Flash Memory

Altera's flash memory configuration controller provides an alternative configuration solution for high-density FPGA-based designs. With the flexibility to use a bigger flash memory to store more configuration data, designers can implement the flash memory controller in Altera's MAX® II, MAX 3000A, or MAX 7000 devices for use in Stratix series, Arria series, and Cyclone series FPGAs.

Introduction

Configuration bitstream sizes are increasing with the introduction of higher-density FPGAs. This increase requires larger configuration devices to store the data and configure these FPGAs. As an alternative to using larger configuration devices, designers can use flash memory to store configuration data. To use flash memory to perform configuration, designers must use a flash memory configuration controller, which also allows the implementation of a remote system upgrade configuration scheme in the design. This white paper shows how to implement the flash memory controller in Altera's MAX® II, MAX 3000A, or MAX 7000 devices.

Configuration Controller Features

Designers can use the MAX series configuration controller for the following functions:

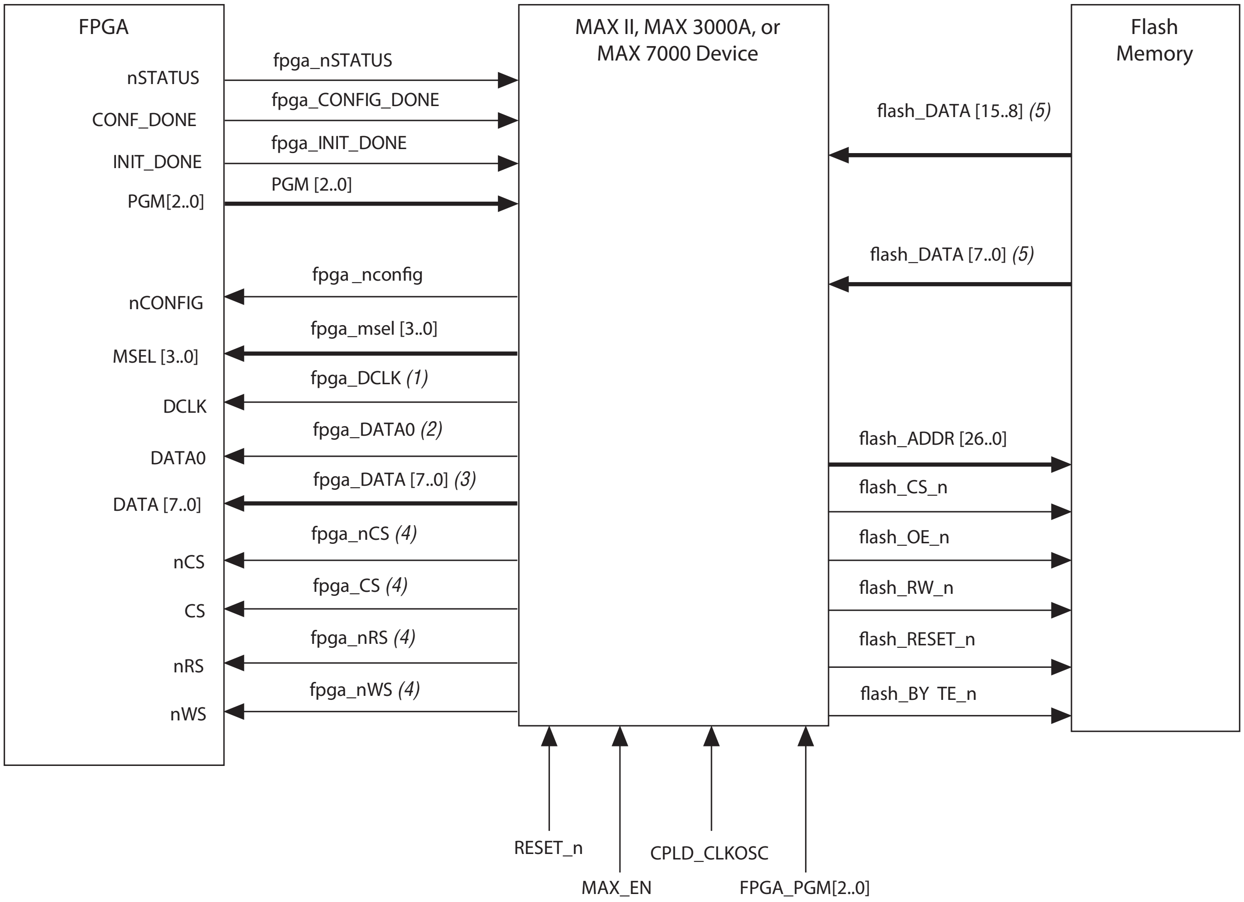
- Read configuration data from a flash memory
- Configure Altera® FPGAs
- Remote System Upgrade Configuration (only in Altera's Stratix® series, Arria® series, and Cyclone® series FPGAs)
- Configuration from multiple pages of configuration data. `FPGA_PGM` pins allow designers to choose one of the configuration pages to configure FPGAs

The MAX series configuration controller supports the following configuration modes:

- Fast Passive Parallel (FPP) Mode (with and without decompression)
- Passive Serial (PS) Mode (with and without decompression)
- Passive Parallel Asynchronous (PPA) Mode
- Remote System Upgrade (only in Stratix series, Arria series, and Cyclone series FPGAs)

Figure 1 shows the flash memory controller block diagram.

Figure 1. Flash Memory Configuration Controller Block Diagram



Notes:

(1) Not used in PPA mode.

(2) Not used in FPP or PPA mode.

(3) Not used in PS mode.

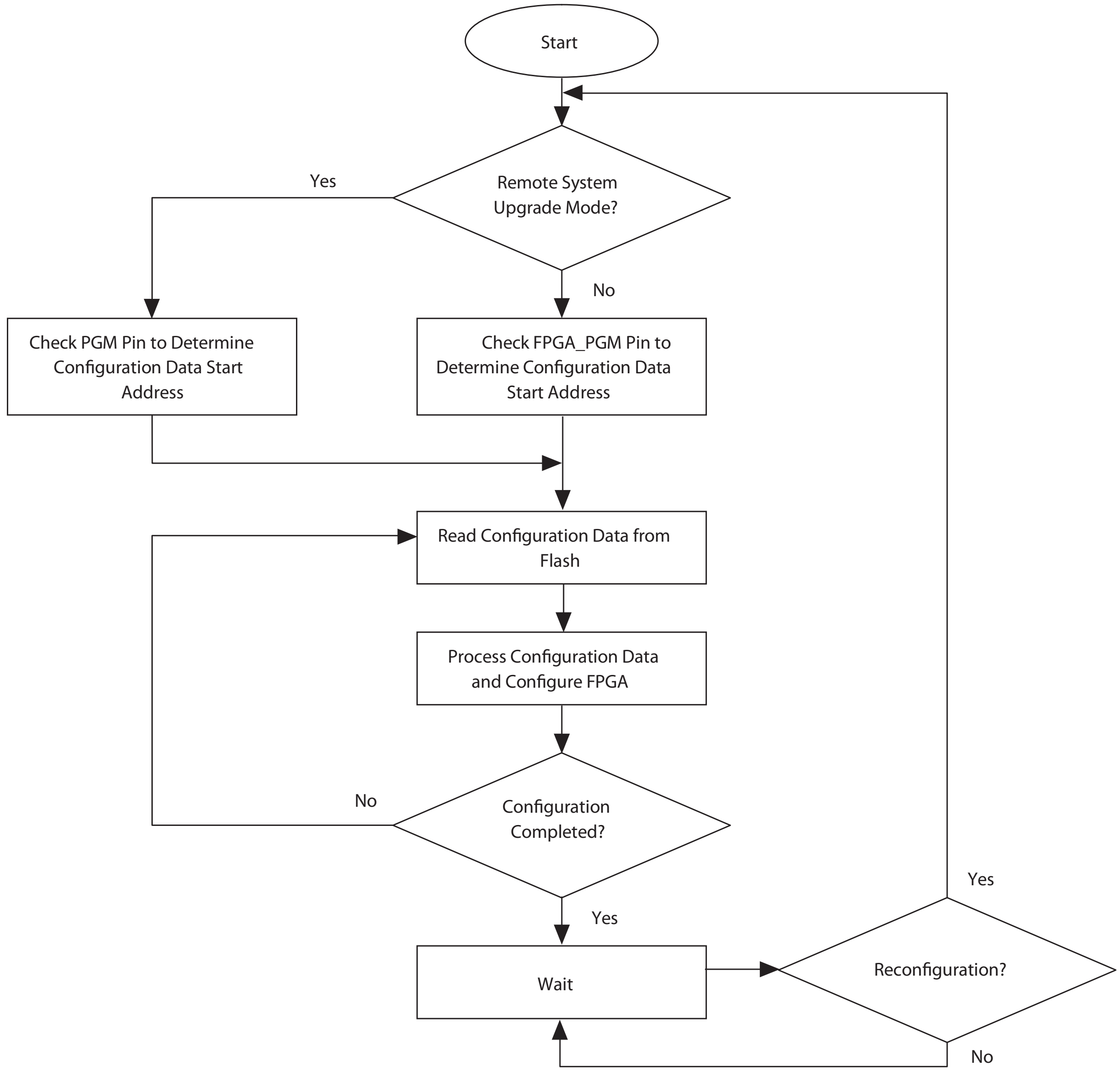
(4) Not use in PS or FPP mode.

(5) Controller works with 8- and 16-bit flash devices.

Configuration Controller Operation

Figure 2 shows how the configuration controller executes the basic operation when it is powered up.

Figure 2. Flash Memory Configuration Controller Basic Operation Flow Chart


Start
Remote System Upgrade Mode?
Yes
No
Check PGM Pin to Determine Configuration Data Start Address
Check FPGA_PGM Pin to Determine Configuration Data Start Address
Read Configuration Data from Flash
Process Configuration Data and Configure FPGA
No
Configuration Completed?
Yes
Wait
Yes
Reconfiguration?
No

Page Selection for Configuration Controller

Flash memory can store multiple configuration pages in different addresses. The configuration controller allows the designer to select which configuration page in the flash memory to load during the configuration. To determine which page to load, the controller reads the `FPGA_PGM` pin in non-remote upgrade mode or reads the `PGM` pin in remote upgrade mode. A Stratix series, Arria series, or Cyclone series FPGA controls the `PGM` pins through the FPGA's remote system upgrade block. Designers can control `FPGA_PGM` pins using DIP switches or other devices.

Read and Process Configuration Data

The configuration controller reads configuration data through the `flash_DATA [7..0]` data bus, and optionally through the `flash_DATA [15..8]` data bus. In PS mode, the configuration controller sends the serial configuration bitstream through the `fpga_DATA0` pin. In FPP, FPP with decompression, and PPA mode, the configuration controller sends the configuration data through the `fpga_DATA[7..0]` data bus.

During the configuration process, the configuration controller executes the following processes:

PS Mode

- Reads one byte (eight bits) or two bytes (16 bits) of configuration data from flash memory and serializes the data
- Generates the `DLCK` signal and sends one bit of configuration data for every `DLCK` signal
- Reads the next byte(s) of configuration data from flash memory after 8 or 16 `DLCK` signals

FPP Mode

- Reads one byte (eight bits) or two bytes (16 bits) of configuration data from flash memory
- Generates the `DLCK` signal and sends one byte (eight bits) of configuration data for every `DLCK` signal
- Reads the next byte(s) of configuration data from flash memory after one or two `DLCK` signals

FPP Mode with Decompression

- Reads one byte (eight bits) or two bytes (16 bits) of configuration data from flash memory
- Generates the `DLCK` signal and sends one byte (eight bits) of configuration data for every four `DLCK` signals
- Reads the next byte(s) of configuration data in flash memory after every four or eight `DLCK` signals

PPA Mode

- Reads one byte (eight bits) or two bytes (16 bits) of configuration data from flash memory
- Sends one byte (eight bits) or two bytes (16 bits) of configuration data to the FPGA and generates the control signals (`nWS`, `nRS`, `CS`, and `nCS`) to regulate the data transfer
- Reads the next byte(s) of configuration data in flash memory after sending one or two bytes of configuration data

For more information on the configuration modes, refer to the *Configuration Handbook*.

For more information on the configuration controller, refer to *AN 386: Using the Parallel Flash Loader with the Quartus II Software*.

Reconfiguration

The configuration controller reconfigures the FPGA if there is an error (`nSTATUS` goes low) during the configuration state.

The FPGA can initiate reconfiguration in remote update mode. The remote update block can update the `PGM` pin and initiate reconfiguration through the FPGA `core_nconfig`. The `CONF_DONE` pin goes low when the FPGA initiates `core_nconfig` in user mode. The configuration controller checks the `CONF_DONE` pin and reconfigures the FPGA after `CONF_DONE` goes low.

For more information about remote system upgrades, refer to the *Remote Update Circuitry (ALTREMOTE_UPDATE) Megafunction User Guide*.

Configuration Modes

The configuration controller supports the following configuration modes:

- FPP Mode
- PS Mode
- PPA Mode
- Remote System Upgrade

FPP Mode

Stratix series, Arria series, selected Cyclone series, and APEX™ II devices support the FPP configuration mode.

During FPP configuration, configuration data is transferred from a 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. For FPP without decompression or the design security feature, the configuration data is transferred at a rate of one byte per clock cycle. For FPP with decompression and the design security feature, the configuration data is transferred at a rate of one byte every four clock cycles.

PS Mode

Stratix series, Arria series, Cyclone series, APEX II, APEX 20K, Mercury™, ACEX® 1K, FLEX® 10K, and FLEX 6000 devices support the PS configuration mode.

During PS configuration, configuration data is transferred from flash memory to the FPGA on the DATA (FLEX 6000 devices) or DATA0 (Stratix series, Arria series, Cyclone series, APEX II, APEX 20K, Mercury, ACEX 1K, and FLEX 10K devices) pin. This configuration data is latched into the FPGA on the rising edge of DCLK. Configuration data is transferred at a rate of one bit per clock cycle.

PPA Mode

Stratix series, Arria series, APEX II, APEX 20K, Mercury, ACEX 1K, and FLEX 10K devices support the PPA configuration mode.

During PPA configuration, configuration data is transferred from a storage device, such as a configuration device or flash memory to the FPGA on DATA[7..0] pins. Since this configuration mode is asynchronous, control signals (nWS, nRS, CS, and nCS) regulate the configuration cycle.

For more information about the FPP, PS, and PPA configuration modes, refer to the *Configuration Handbook*.

Remote System Upgrade

The Remote Update Circuitry megafunction supports remote system upgrade using FPP, PS, or PPA configuration modes to configure Stratix series, Arria series, and Cyclone series FPGAs.

The configuration controller determines which page to load in remote system upgrade mode by reading the PGM pin on the FPGA. The configuration controller monitors the nSTATUS pin to detect any error during configuration or to initiate reconfiguration after configuration mode. The FPGA's remote system upgrade block initiates reconfiguration to change the page.

For more information about remote system upgrades, refer to the *Remote Update Circuitry (ALTREMOTE_UPDATE) Megafunction User Guide*.

Flash Memory

Designers must convert the configuration data from SRAM Object Files (**.sof**) to a HEXOUT file (**.hex** or **.hexout**) and program it into the flash memory. Designers can generate HEXOUT files by choosing the **Convert Programming Files** in the Quartus® II design software's file menu.

For information on the HEXOUT file format, refer to Intel's *Hexadecimal Object File Format Specification*.

Prior to placing it on a board, designers can program the flash memory with standard programming equipment or in-system using test equipment. Because different flash memories have different algorithms, it is recommended that the flash memory data sheet is read for accurate programming information.

A full list of supported flash memories can be found in *AN 386: Using the Parallel Flash Loader with the Quartus II Software.*

Source Code

The configuration controller megafunction source code is available in Verilog HDL and VHDL code. The same source code can be compiled to support the following four configuration modes:

- FPP
- FPP Decompression
- PS
- PPA

The configuration controller megafunction source code is written for MAX II, MAX 3000A, and MAX 7000 devices. The code reads from the flash memory and configures the FPGA. Designers can customize and modify the megafunction according to other hardware requirements.

For more details about the configuration controller megafunction source code, refer to *AN 386: Using the Parallel Flash Loader with the Quartus II Software.*

Conclusion

The flash memory configuration controller provides an alternative configuration solution for a design that uses high-density FPGAs. It offers the flexibility to use a bigger flash memory to store more configuration data. Designers can use the Parallel Flash Loader and ALTREMOTE_UPDATE megafunctions to design remote upgrade systems for Stratix series, Arria series, and Cyclone series FPGAs in FPP, PS, and PPA modes.

Further Information

- *Configuration Handbook:*
  www.altera.com/literature/lit-config.jsp
- *AN 386: Using the Parallel Flash Loader with the Quartus II Software:*
  www.altera.com/literature/an/an386.pdf
- *Remote Update Circuitry (ALTREMOTE_UPDATE) Megafunction User Guide:*
  www.altera.com/literature/ug/ug_altremote.pdf
- *Intel Hexadecimal Object File Format Specification*:
  http://microsym.com/editor/assets/intelhex.pdf

ALTERA®

101 Innovation Drive
San Jose, CA 95134
www.altera.com