AN 933: Updating Intel® Stratix® 10 FPGA Firmware

Subscribe

Send Feedback

AN-933 | 2020.11.18

Latest document on the web: PDF | HTML

Contents

1. AN 933: Updating Intel® Stratix® 10 FPGA Firmware

Intel periodically updates Intel® Stratix® 10 FPGA firmware and provides these updates with Intel Quartus® Prime Pro Edition and Intel Quartus Programmer software. The firmware is inserted during the process of converting compiled primary device programming files, commonly known as SRAM Object Files (`.sof`), into secondary device programming files, such as `.jic` or `.rpd` files. As a result, it is possible to update the firmware without recompiling existing designs.

This application note describes the process of performing an update of the Intel Stratix 10 FPGA firmware. The document explains the process to create such an update, to verify the update, and special considerations for the firmware co-signing and Remote System Upgrade (RSU) features. You must then deploy the updated programming file to devices in your production environment and power cycle the Intel Stratix 10 device in order to load the new firmware.

The deployment process is user-dependent and outside the scope of this guide. After you deployed the update, Intel recommends that you follow the instructions to utilize the firmware anti-rollback feature.

Related Information

- Intel Stratix 10 Device Security User Guide
- Intel Stratix 10 Configuration User Guide
- Intel Stratix 10 Hard Processor System Remote System Update User Guide
- Intel Quartus Prime Pro Edition User Guide: Programmer

1.1. Updating Intel Stratix 10 Firmware in Programming Files

The Intel Quartus Prime Pro Edition or Intel Quartus Prime Programmer software installation contains the Intel Stratix 10 FPGA firmware. To update firmware, you must first install the latest available version of Intel Quartus Prime Pro Edition or Intel Quartus Prime Programmer.

1.2. Firmware Co-Signing

Intel Stratix 10 FPGA firmware co-signing is a feature that allows an Intel Stratix 10 device owner to require the FPGA to validate both, the Intel signature and an owner signature, prior to loading and executing firmware. A detailed description of the feature is available in the Co-Signing Device Firmware Overview section of the *Intel Stratix 10 Device Security User Guide*.

If you enabled firmware co-signing, you perform the following steps to co-sign firmware:

1. Locate the firmware file `Stratix_10.zip` and sign it with a signature chain that begins with your root key and ends with a code signing key with the `SIGN_CODE` permission and an appropriate key cancellation ID.
2. You need to specify the signed `Stratix_10.zip` in the Quartus Programming File Generator GUI, or in any Quartus Programming File Generator command line operation with the `-o fw_source=` option as you continue through the following sections.
3. You may use a new code signing key and advance the key cancellation ID in order to utilize both, the Intel and your key cancellation ID-based anti-rollback mechanisms. If you choose to use a new key cancellation ID, you need to cancel the your key cancellation ID assigned to the key that was previously used to sign firmware in addition to canceling the appropriate Intel key cancellation ID, after you have completed the firmware updates. Instructions to cancel key cancellation IDs are in the *Intel Stratix 10 FPGA Firmware Cancellation IDs* section.

Related Information

Intel Stratix 10 Device Security User Guide
Additional information on co-signing device firmware.

1.2.1. Non-RSU Flow

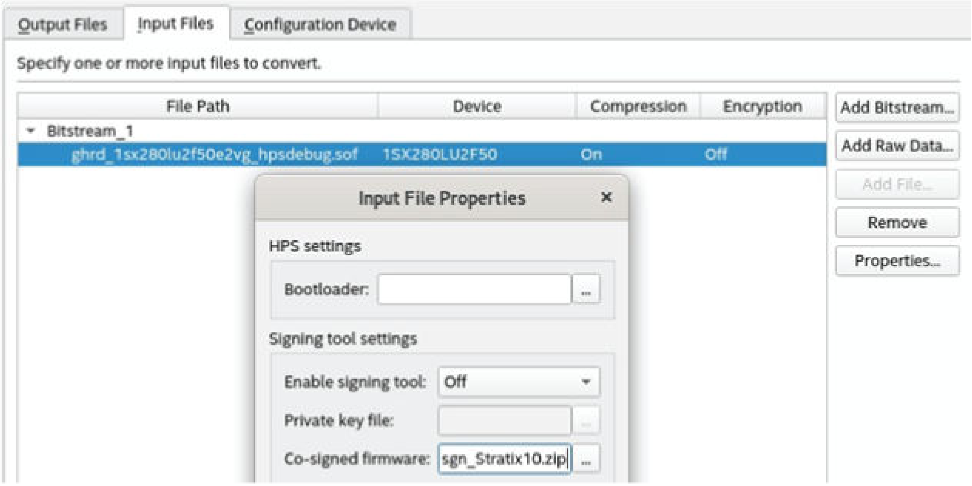
The non-RSU flow is appropriate for all configuration methods that do not involve RSU support. You must use your updated installation of Intel Quartus Prime Pro Edition or Intel Quartus Prime Programmer in order to use the latest firmware.

You use the **Programming File Generator** to convert your existing compiled primary device programing (`.sof`) file into a new secondary device programming file that is appropriate for your configuration method, such as `.jic` for JTAG configuration or `.pof` for Avalon® streaming interface configuration. Intel Stratix 10 FPGA firmware is backwards compatible with compiled bitstreams. For example, you may use firmware from Intel Quartus Prime Pro Edition software version 20.2 with a bitstream compiled with Intel Quartus Prime Pro Edition software version 20.1 or earlier.

Follow the instructions in the Generating Secondary Programming Files section of the *Intel Quartus Prime Pro Edition User Guide: Programmer* document to create the appropriate secondary programming file, and check the box to additionally generate the Raw Binary File (`.rbf`) equivalent. You need the `.rbf` file as you proceed to the next step,

If you are using the firmware co-signing feature, after you specify the bitstream (`.sof`) as the input file to **Programming File Generator**, you must click the bitstream row and click the **Properties...** button, and specify the path to the co-signed firmware `.zip` file, as shown in the *Specifying a Co-Signed Firmware File in Programming File Generator* figure. Programming File Generator then inserts the co-signed firmware into the generated secondary programming files. You must also enable other device security features, such as authentication or encryption, during this step as appropriate. Refer to the *Intel Stratix 10 Device Security User Guide* for more information.

Figure 1. Specifying a Co-Signed Firmware File in Programming File Generator



Specify one or more input files to convert.

File Path	Device	Compression	Encryption
Bitstream_1			
ghrd_1sx280lu2f50e2vg_hpsdebug.sof	1SX280LU2F50	On	Off

Input File Properties

HPS settings

Bootloader:

Signing tool settings

Enable signing tool: Off

Private key file:

Co-signed firmware: sgn_Stratix10.zip

1.2.2. RSU Flow

The RSU flow is appropriate for the Active Serial (AS) configuration mode with the RSU support. Updating Intel Stratix 10 FPGA firmware for RSU involves more steps as there are multiple instances of the firmware in an RSU flash layout.

A firmware instance is included in all application images, the factory image, and each copy of decision firmware. You may choose to deploy updates to the different instances of firmware in stages, but all instances of firmware, including the decision firmware, must be updated prior to utilizing the firmware anti-rollback feature. All steps in this section must be performed with the updated installation of Intel Quartus Prime Pro Edition or Intel Quartus Prime Programmer.

First, follow the instructions in the Generating an Application Image section of the *Intel Stratix 10 Configuration User Guide* to generate updates for your application images. You should check the box to additionally generate the Raw Binary File (`.rbf`) for each application image. If you are using firmware co-signing, specify the co-signed firmware file in the properties of each input `.sof` file as shown in the *Specifying a co-signed firmware file in Programming File Generator*.

You must also enable other device security features, such as authentication or encryption, during this step as appropriate. For each application image generated, you may verify the presence of updated firmware using the steps in the *Verifying Firmware Updates* section of this document.

The Intel Stratix 10 FPGA firmware implements a robust and reliable procedure to update the factory image, including the factory image firmware, and the copies of the decision firmware, all with one update image. After you created your updated application images, follow the instructions in the Generating a Factory Update Image section of the *Intel Stratix 10 Configuration User Guide*. If you are using firmware co-signing, specify the co-signed firmware in the `.sof`properties as you turn on the **Enable remote system firmware upgrade** option. You must also enable other device security features, such as authentication or encryption, during this step as appropriate.

Related Information

- Intel Stratix 10 Configuration User Guide
- Intel Quartus Prime Pro Edition User Guide: Programmer

1.2.3. Verifying Firmware Updates

The Intel Quartus Programming File Generator software provides a way to inspect and verify the integrity of signature chains of secondary programming files in the Raw Binary File (`.rbf`) format. This procedure allows you to confirm the version of the firmware present in the programming file and, if you are using firmware co-signing, verify that the your signature chain for the firmware is correct.

The first signature chain on firmware, identified by the **Programming File Generator** tool as **Signature Chain #0** is always the signature provided by Intel. The version of firmware can be identified by the key cancellation ID that the signature chain can be canceled by. Refer to the Canceling Intel Firmware ID section of the *Intel Stratix 10 Device Security User Guide* for information on the key cancellation ID that corresponds to a given version of firmware.

Signature Chain #1 is reserved for use by Intel.

If you are using firmware co-signing, your signature chains are placed in entries **Signature Chain #2** and **Signature Chain #3**, where you can verify the key cancellation ID, root public key hash fuse value, and public key X/Y contents.

Run the following command to use Programming File Generator to verify the secondary programming file:

```
quartus_pfg --check_integrity updated_bitstream.rbf
```

An example of the output of the command is below. The bold font emphasizes the important values to inspect.

```
Info: Command: quartus_pfg --check_integrity signed_bitstream.rbf
Integrity status: OK

Section
Type: CMF
Signature Descriptor ...
Signature chain #0 (entries: 3, offset: 96)
Entry #0
Fuse: A1B9545C CAC4152D 9511A9AB 321778ED 1180A280 6DC58F2C 5607433E \
02A872E3 F52B2AE5 F7B8BDE0 53FA000D 8FC7AC04
Generate key ...
Curve : secp384r1
X     :
FC28C88662DF1437DD98E61336467DC9CDA788F22F949D8F488DA755A9F8CC11AEC10006E2649
0B3EAB8148E6C8AA8A1
Y     :
95D1EA0FF4C7374B350FDF39CFAE3AD8D0AEA9451EA66B5B1DFD4084DA68BC4DAD3AF5CF378D7
C6FB62A10BA7C512276
Entry #1
Generate key ...
Curve : secp384r1
X     :
35D8FD7138328F1CE56AE5DD7B6A528FF01CFD1493E75064931CB71D90CE87AA56219AF0AC5C9
8096B939BE23AE7AD51
Y     :
81830A069C8831E39817F4D193091D7F829A6DE904A50274A2282F644F618B9B19CBE1CDBCC8D
F79DC1206E6B054FAE8
Entry #2
Keychain permission: SIGN_CODE
Keychain can be cancelled by ID: 7
Signature chain #1 (entries: 0, offset: 0)
Signature chain #2 (entries: 3, offset: 640)
Entry #0
Fuse: 0E0EC654 BDA9944E 86B5E3D8 C0EAE5D0 7FA202DD 523AE96F 8E4639BC AA02F142

Generate key ...
Curve : prime256v1
X     :
DD12BDA1116B67E53C01B4433B08C0FFC8BBF3E6367146E38E327C8AC6254B3A
Y     :
B586065A814E23A649E8B5B4DD35F3E8D39328AF387DFF0336B4CD278E4503EB
Entry #1
Generate key ...
Curve : prime256v1
X     :
BDD025BE705CF1D58AC922EDF5BC156F0BB435D6309200CBC1AA46174E16EBEC
Y     :
56DA0651C1C3C76E1BB7174DE918752A2591B2138BB76D740D3A26C369763619
Entry #2
Keychain permission: SIGN_CODE
Keychain can be cancelled by ID: 1
Signature chain #3 (entries: 0, offset: 0))
```

Related Information

Intel Stratix 10 Device Security User Guide

1.3. FPGA Firmware Cancellation IDs

The Intel Stratix 10 FPGA implements an anti-rollback feature to help you control which versions of firmware can be loaded onto the FPGA.

The anti-rollback feature uses key cancellation IDs stored in signature chains as well as corresponding values in eFuses. By programming the eFuse that corresponds to a given key cancellation ID, you invalidate the signature chain that contains the key with that ID. This mechanism prevents the FPGA from loading the signed object, which can be firmware, an FPGA bitstream, an HPS first-stage boot loader, or a certificate that validates a command to the FPGA. Intel Stratix 10 FPGA firmware is distinct from the other objects as it is always signed by Intel, and there is an entire set of cancellation fuses for the Intel signature separate from the cancellation fuses that correspond to owner signatures. Therefore, you may utilize Intel Stratix 10 FPGA firmware anti-rollback without using any other device security features.

1.3.1. Updating Firmware Cancellation ID Fuses

To use the Intel Stratix 10 FPGA firmware anti-rollback feature, you issue a command to the FPGA to program the appropriate **Intel key cancellation** fuses prior to power cycling your device. If you are using firmware co-signing, you may program the appropriate **Owner key cancellation** fuses to cancel the signature chain used to sign a prior version of firmware in the same step. Refer to the Canceling eFuses section of the *Intel Stratix 10 Device Security User Guide* for detailed instructions on programming the key cancellation fuses, as well as the Canceling Intel Firmware ID section to choose the ID fuses to program. For example, to use the anti-rollback feature for all FPGA firmware prior to Intel Quartus Prime version 20.2, the correct Intel key cancellation line in the fuse file is:

```
Intel key cancellation    = "0,1,2,3,4,5,6"
```

After you program your **Intel key cancellation** fuses, you must power cycle your device prior to programming an AES root key.

Related Information

Intel Stratix 10 Device Security User Guide

1.4. Document Revision History for AN 933: Updating Intel Stratix 10 FPGA Firmware

Document Version	Changes
2020.11.18	Updated document part number and title from *AN 923: Updating Intel Stratix 10 FPGA Firmware* to *AN 933: Updating Intel Stratix 10 FPGA Firmware*.
2020.06.22	Initial release.