Fusion Security

Introduction to Security in Fusion

The need for security on FPGA programmable logic devices has never been greater than today. If the contents of the FPGA can be read by an external source, the intellectual property (IP) of the system is vulnerable to unauthorized copying. Actel Fusion™ devices contain state-of-the-art circuitry to make the Flash-based devices secure during and after programming. The new Fusion Flash FPGAs consist of an FPGA array core, FlashROM and Flash Memory Blocks (FBs). The FlashROM, the FBs, and the FPGA core fabric can be securely programmed independently of each other, allowing the FlashROM or the FBs to be updated without changing the FPGA core fabric.

Actel has incorporated the Advanced Encryption Standard (AES) decryption core into the new Fusion devices and has also included the Actel Flash-based lock technology, FlashLock®. Together, they provide leading-edge security in a programmable logic device. Configuration data loaded into Fusion can be decrypted prior to being written to the FPGA core using the AES 128-bit block cipher standard. The AES encryption key is stored in on-chip, nonvolatile Flash memory.

Fusion devices have been designed with the most comprehensive programming logic design security in the industry. In the architecture of Fusion devices, security has been designed into the very fabric. The Flash cells are located beneath seven metal layers, and the use of many device design and layout techniques makes invasive attacks difficult. Since device layers cannot be removed without disturbing the charge on the programmed (or erased) Flash gates, devices cannot be easily deconstructed to decode the design. Fusion is unique in being reprogrammable and having inherent resistance to both invasive and noninvasive attacks on valuable IP. Secure remote in-system programming (ISP) is now possible with AES encryption capability for the programming file during electronic transfer. Figure 1 on page 2 shows a view of the AES decryption core inside Fusion devices, which is used to decrypt the encrypted programming file when programming.

This application note outlines the security features offered in Fusion devices, some applications and uses, as well as the different software settings for each application.

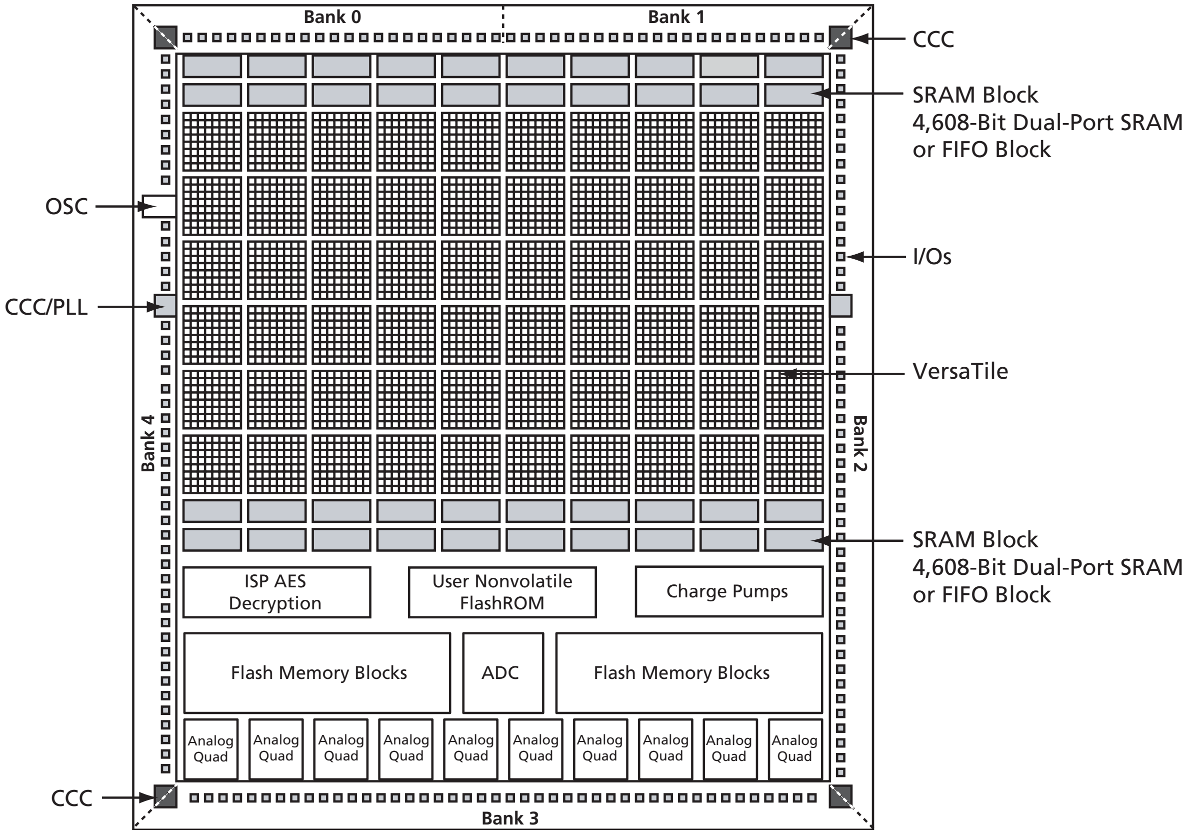


Figure 1 • **Block Representation of the AES Decryption Core in an AFS600 FPGA**

Fusion Security Features

Consider Fusion devices as containing three entities: FlashROM, FBs, and the FPGA core fabric. These parts can be programmed or updated independently with a STAPL programming file. The programming files can be AES-encrypted or plain text. This allows maximum flexibility in providing security to the entire device. Refer to the *Fusion FlashROM* application note for the FlashROM structure.

Unlike SRAM-based FPGA devices, which require a separate boot PROM to store programming data, Fusion devices are nonvolatile, and the secured configuration data is stored in on-chip Flash cells that are part of the FPGA fabric. Once programmed, this data is an inherent part of the FPGA array and does not need to be loaded at system power-up. SRAM-based FPGAs load the configuration bitstream upon power-up; therefore, the configuration is exposed and can be read easily.

The built-in FPGA core, FB, and FlashROM support programming files encrypted with the 128-bit AES (FIPS-192) block ciphers. The AES key is stored in dedicated, on-chip Flash memory and can be programmed before the device is shipped to other parties (allowing secure remote field updates).

AES Encryption of Programming Files

Fusion devices employ AES as part of the security mechanism that prevents invasive and noninvasive attacks. The mechanism entails encrypting the programming file with AES encryption then passing the programming file through the AES decryption core, which is embedded in the device. The file is decrypted there, and the device is successfully programmed.

The AES key is protected by a FlashLock security Pass Key that is also implemented in Fusion devices. This FlashLock Pass Key technology is exclusive to the Actel Flash-based device families. FlashLock Pass Key technology can also be implemented without the AES encryption option, providing a choice of different security levels.

In essence, security features can be categorized into the following three options:

- AES encryption with FlashLock Pass Key protection
- FlashLock protection only (no AES encryption)
- No protection

Each of the above options will be explained in more detail in the following sections with application examples and software implementation options.

Advanced Encryption Standard

AES is a cryptographic algorithm that complies with Federal Information Processing Standard (FIPS) Publication 192, used by U.S. government organizations to protect sensitive, unclassified information.

Actel has implemented the 128-bit AES (Rijndael) algorithm in Fusion devices. With this key size, there are approximately 3.4×10^{38} possible 128-bit keys. The DES standard has a 56-bit key size, which provides approximately 7.2×10^{16} possible keys. In their AES fact sheet, the National Institute of Standards and Technology (NIST) uses the following hypothetical example to illustrate the theoretical security provided by AES. If one were to assume that a computing system existed that could recover a DES key in a second, it would take that same machine approximately 149 trillion years to crack a 128-bit AES key. NIST continues to make their point by stating the universe is believed to be less than 20 billion years old.[1]

The AES key is securely stored on-chip in dedicated Fusion Flash memory and cannot be read out. In the first step, the AES key is generated and programmed into the device (for example, at a secure or trusted programming site). The Actel Designer software tool provides AES key generation capability. After the key has been programmed into the device, the device will only correctly decrypt programming files that have been encrypted with the same key. If the individual programming file content is incorrect, a Message Authentication Control (MAC) mechanism inside the device will fail in authenticating the programming file. In other words, when an encrypted programming file is being loaded into a device that has a different programmed AES key, the MAC will prevent this incorrect data from being loaded, preventing possible device damage. See Figure 2 on page 4 for a graphical representation of this process.

It is important to note that the user decides what level of protection will be implemented for the device. When AES protection is desired, the FlashLock Pass Key must be set. The AES key is a content protection mechanism, whereas the FlashLock Pass Key is a device protection mechanism. When the AES key is programmed into the device, the device still needs the Pass Key to protect the FPGA and FlashROM contents and the security settings, including the AES key. Using the FlashLock Pass Key prevents modification of the design contents by means of simply programming the device with a different AES key.

AES Decryption in Fusion Devices

Fusion has a built-in 128-bit AES decryption core, which decrypts the encrypted programming file and performs a MAC check that authenticates the file prior to programming. This will ensure the following:

- Correct decryption of the encrypted programming file
- Prevention of erroneous or corrupted data being programmed during the programming file transfer
- Correct bitstream passed to the device for decryption

Figure 2 on page 4 shows the use of AES in the Fusion.

[1] *National Institute of Standards and Technology, "ADVANCED ENCRYPTION STANDARD (AES) Questions and Answers," 28 January 2002, <http://csrc.nist.gov/CryptoToolkit/aes/aesfact.html> (10 January 2005).*

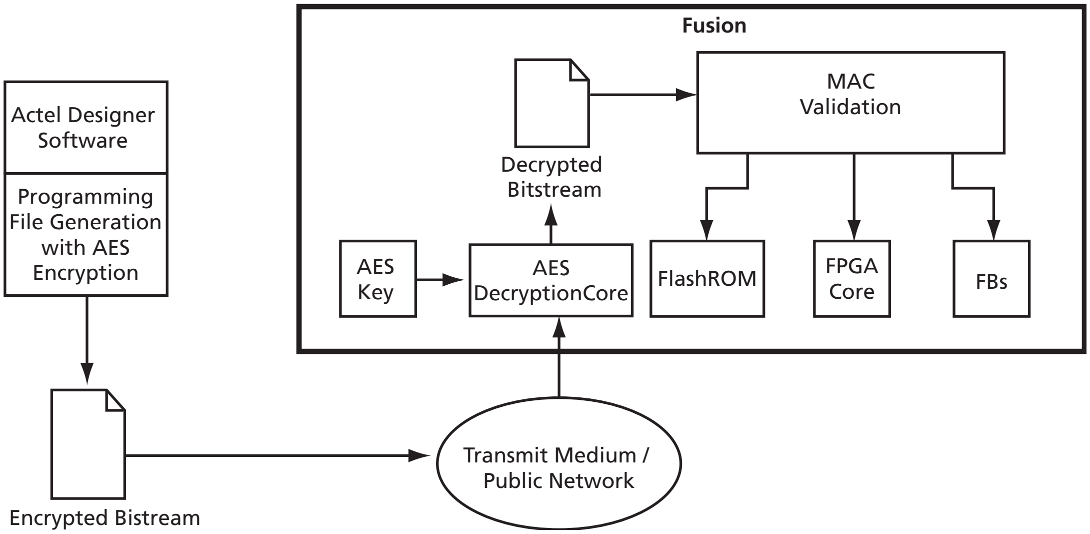


Figure 2 • **Example Application Scenario Using AES in Fusion Devices**

FlashLock

The 128-bit Flash-based FlashLock feature in Fusion works via a key mechanism. The user locks or unlocks the device with a user-defined FlashLock Pass Key. When the device is locked, there is no access to the FPGA without the correct FlashLock Pass Key. By default, functions such as device write, verify, and erase are disabled. Figure 3 shows the application scenarios and the corresponding security settings selected during the programming file generation step.

Security In Action

This section illustrates some applications of the security advantages of Fusion devices (Figure 3).

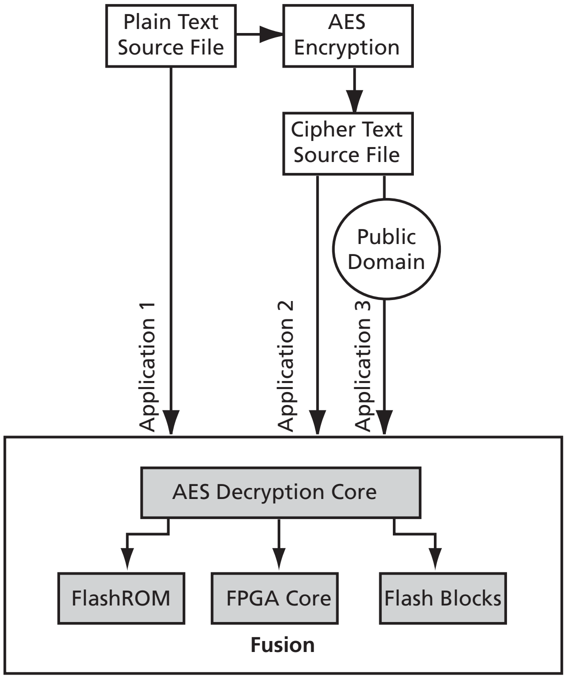


Figure 3 • **Security Options in Fusion**

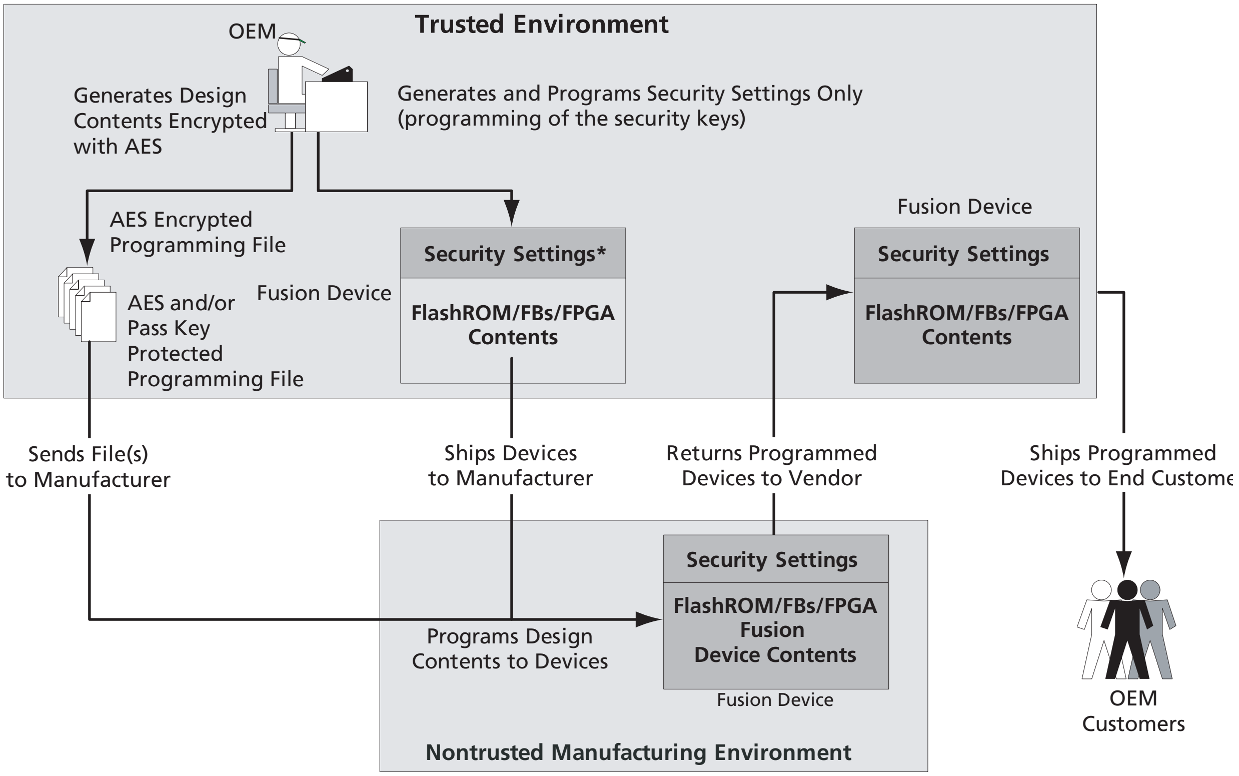
Application 1: Trusted Environment

As illustrated in Figure 4, this application allows the programming of devices at design locations where research and development take place. Therefore, encryption is not necessary and is optional to the user. This is often a secure way to protect the design, since the design program files are not sent elsewhere. In situations where production programming is not available at the design location, programming centers (such as Actel in-house programming) provide a way of programming designs at an alternative, secure, and trusted location. In this scenario, the user will generate a STAPL programming file from the Designer software in plain text format containing information on the entire design or the portion of the design to be programmed. The user can choose to employ the FlashLock Pass Key feature with the design. Once the design is programmed to unprogrammed devices, the design will be protected by this FlashLock Pass Key.

Application 2: Nontrusted Environment – Unsecured Location

Often, programming of the devices is not performed in the same location as actual design implementation to reduce manufacturing cost. Overseas programming centers and contract manufacturers are examples of this scenario.

To achieve security in this case with Fusion, the AES key and the FlashLock Pass Key can be initially programmed in-house (trusted environment). This is done by generating a programming file with only the security settings and no design contents. The design FPGA core, FlashROM, and/or FB contents are generated in a separate programming file. This programming file must be set with the same AES key that was used to program to the device previously so the device will correctly decrypt this encrypted programming file. As a result, the encrypted design content programming file can be safely sent off-site to nontrusted programming locations for design programming. Figure 4 shows a more detailed flow for this application.


Trusted Environment
OEM
Generates Design Contents Encrypted with AES
Generates and Programs Security Settings Only (programming of the security keys)
AES Encrypted Programming File
AES and/or Pass Key Protected Programming File
Fusion Device
Security Settings*
FlashROM/FBs/FPGA Contents
Fusion Device
Security Settings
FlashROM/FBs/FPGA Contents
Sends File(s) to Manufacturer
Ships Devices to Manufacturer
Returns Programmed Devices to Vendor
Ships Programmed Devices to End Custome
Security Settings
FlashROM/FBs/FPGA Fusion Device Contents
Programs Design Contents to Devices
Fusion Device
Nontrusted Manufacturing Environment
OEM Customers

Note: ******Programmed portion indicated with gray.*

Figure 4 • **Implementing Device Programming in a Nontrusted Environment**

Application 3: Nontrusted Environment – Field Updates/Upgrades

Programming or reprogramming of devices may occur at remote locations. Reconfiguration of devices in consumer products/equipment through public networks is one example. Typically, the remote system is already programmed with particular design contents. When design update (FPGA array contents update) and/or data upgrade (FlashROM and/or FB contents upgrade) is necessary, an updated programming file with AES encryption can be generated, sent across public networks, and transmitted to the remote system. Reprogramming can then be done using this AES-encrypted programming file, providing easy and secure field upgrades. Fusion devices support this secure ISP using AES. The detailed flow for this application is shown in Figure 5.

To prepare devices for this scenario, the user can initially generate a programming file with the available security setting options. This programming file is programmed into the devices before shipment. Actel recommends that the programming file use encryption with an AES key, especially when ISP is done via public domain.

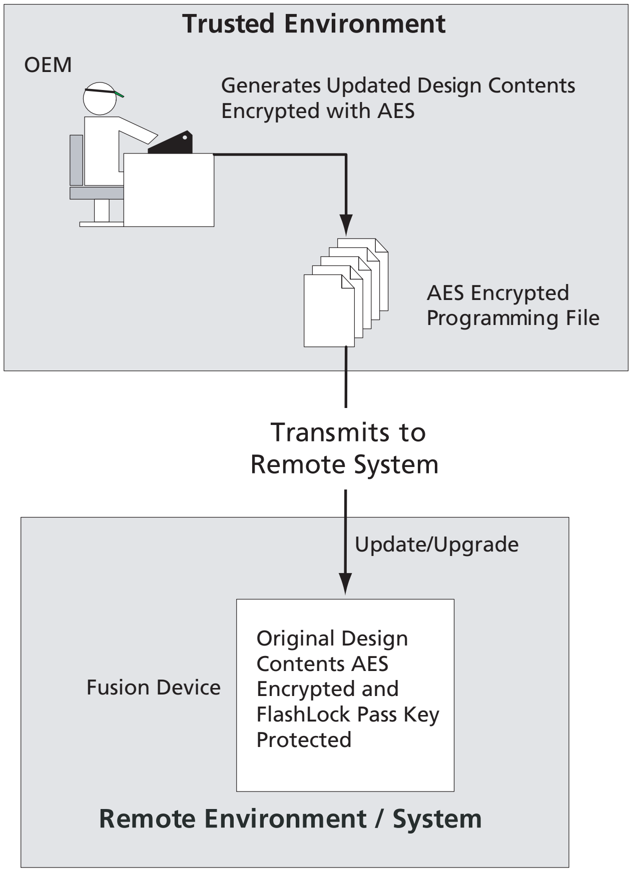

Trusted Environment
OEM
Generates Updated Design Contents Encrypted with AES
AES Encrypted Programming File
Transmits to Remote System
Update/Upgrade
Fusion Device
Original Design Contents AES Encrypted and FlashLock Pass Key Protected
Remote Environment / System

Figure 5 • **Remote Update in Nontrusted Environment – Field Updates/Upgrades**

Fusion FlashROM Security Use Models

Each of the subsequent sections describes in detail the available selections in Actel Designer software as an aid to understanding security applications and generating appropriate programming files for those applications. Before proceeding, it is helpful to review Figure 3 on page 4, which gives a general overview of the programming file generation flow within the Designer software as well as what occurs during the device programming stage. Specific settings are discussed in the following sections.

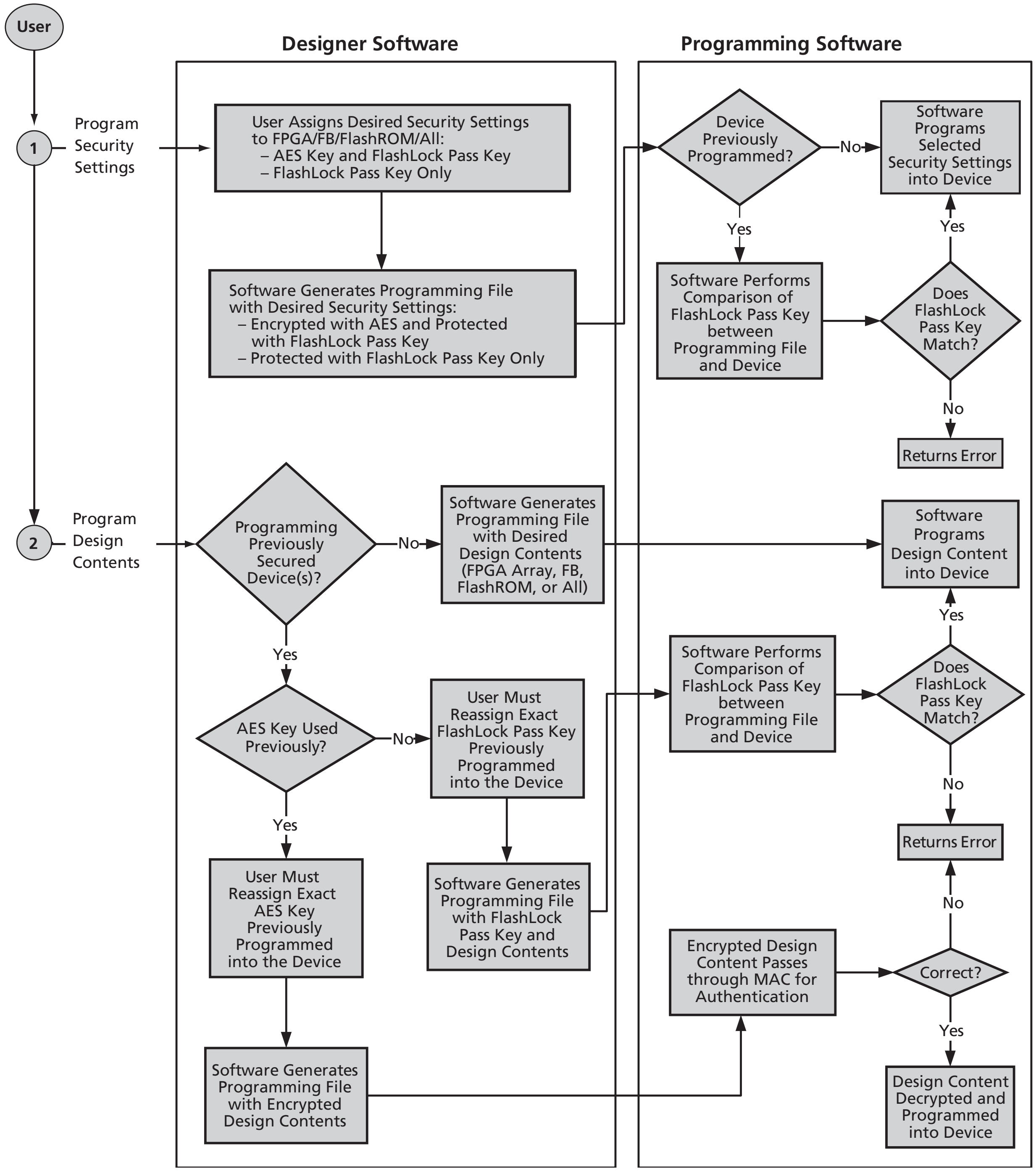
In Figure 3 on page 4, the flow consists of two sub-flows. Sub-flow 1 describes programming security settings to the device only, while sub-flow 2 describes programming the design contents only.

In Application 1, described on page 5, the user does not need to generate separate files but can generate one programming file containing both security settings and design contents. Then programming of the security settings and design contents is done in one step. Both sub-flow 1 and sub-flow 2 are used.

In Application 2, described on page 5, the trusted site should follow sub-flows 1 and 2 separately to generate two separate programming files. The programming file from sub-flow 1 will be used at the trusted site to program the device(s) first. The programming file from sub-flow 2 will be sent off-site for production programming.

In Application 3, described on page 6, typically only sub-flow 2 will be used because only updates to the design content portion are needed and no security settings need to be changed.

In the event that update of the security settings is necessary, see the "Reprogramming Devices" section on page 14 for details.


User
Designer Software
Programming Software
1
Program Security Settings
User Assigns Desired Security Settings to FPGA/FB/FlashROM/All:
– AES Key and FlashLock Pass Key
– FlashLock Pass Key Only
Software Generates Programming File with Desired Security Settings:
– Encrypted with AES and Protected with FlashLock Pass Key
– Protected with FlashLock Pass Key Only
Device Previously Programmed?
No
Software Programs Selected Security Settings into Device
Yes
Software Performs Comparison of FlashLock Pass Key between Programming File and Device
Does FlashLock Pass Key Match?
Yes
No
Returns Error
2
Program Design Contents
Programming Previously Secured Device(s)?
No
Software Generates Programming File with Desired Design Contents (FPGA Array, FB, FlashROM, or All)
Software Programs Design Content into Device
Yes
AES Key Used Previously?
No
User Must Reassign Exact FlashLock Pass Key Previously Programmed into the Device
Software Performs Comparison of FlashLock Pass Key between Programming File and Device
Does FlashLock Pass Key Match?
Yes
No
Returns Error
Yes
User Must Reassign Exact AES Key Previously Programmed into the Device
Software Generates Programming File with FlashLock Pass Key and Design Contents
Software Generates Programming File with Encrypted Design Contents
Encrypted Design Content Passes through MAC for Authentication
Correct?
No
Yes
Design Content Decrypted and Programmed into Device

Note: *If programming the Security Header only, just perform Step 1.*
If programming design content only, just perform Step 2.

Figure 6 • **Fusion Security Programming Flows**

Generation of the Programming File in a Trusted Environment – Application 1

As discussed in the "Application 1: Trusted Environment" section on page 5, in a trusted environment the user can choose to program the device with plain text bitstream content. It is possible to use plain text for programming even when the FlashLock Pass Key option has been selected. In this application, it is not necessary to employ AES encryption protection. For AES encryption settings, refer to the next sections.

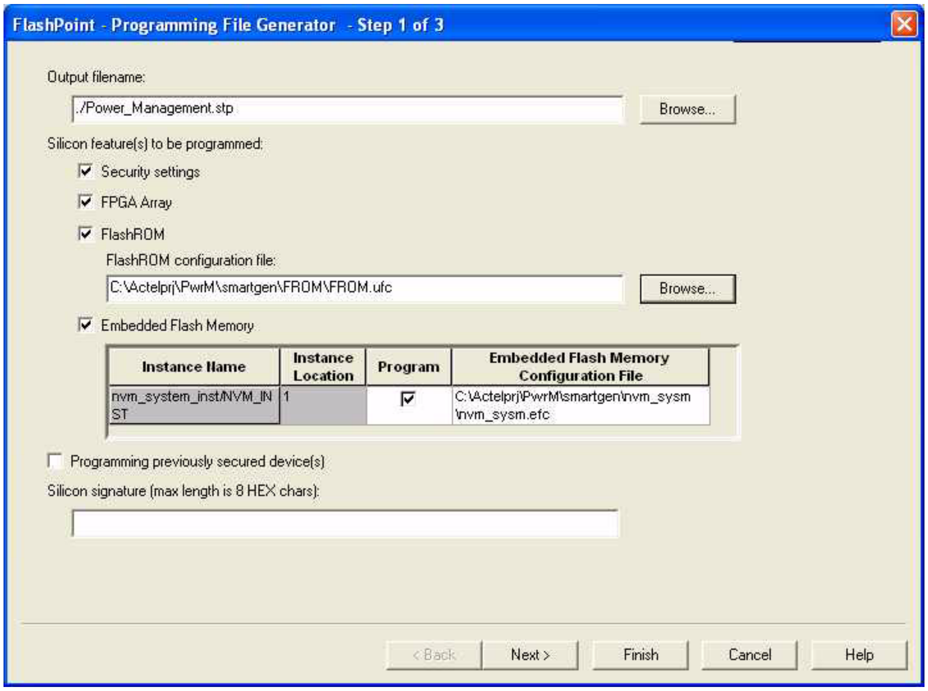
The generated programming file will include the security setting (if selected) and the plain text programming file content for the FPGA Array, FlashROM, and/or FB. These options are indicated in Table 1.

Table 1 • **Plain Text Security Options**

Security Protection	**FlashROM Only**	**FPGA Core Only**	**FB Core Only**	**All**
No AES / No FlashLock	✓	✓	✓	✓
FlashLock	✓	✓	✓	✓
AES and FlashLock	–	–	–	–

For this scenario, generate the programming file as follows:

1. Select the **Silicon features to be programmed** (Security Settings, FPGA Array, FlashROM, Flash Memory Block), as shown in Figure 7. Click **Next**.
   If **Security Settings** is selected (i.e., the FlashLock security Pass Key feature), the following step will be displayed to prompt you to select the security level setting. If no security setting is selected, you will be directed to Step 3.



FlashPoint - Programming File Generator - Step 1 of 3

Output filename: ./Power_Management.stp

Silicon feature(s) to be programmed:

☑ Security settings
☑ FPGA Array
☑ FlashROM

FlashROM configuration file: C:\Actelprj\PwrM\smartgen\FROM\FROM.ufc

☑ Embedded Flash Memory

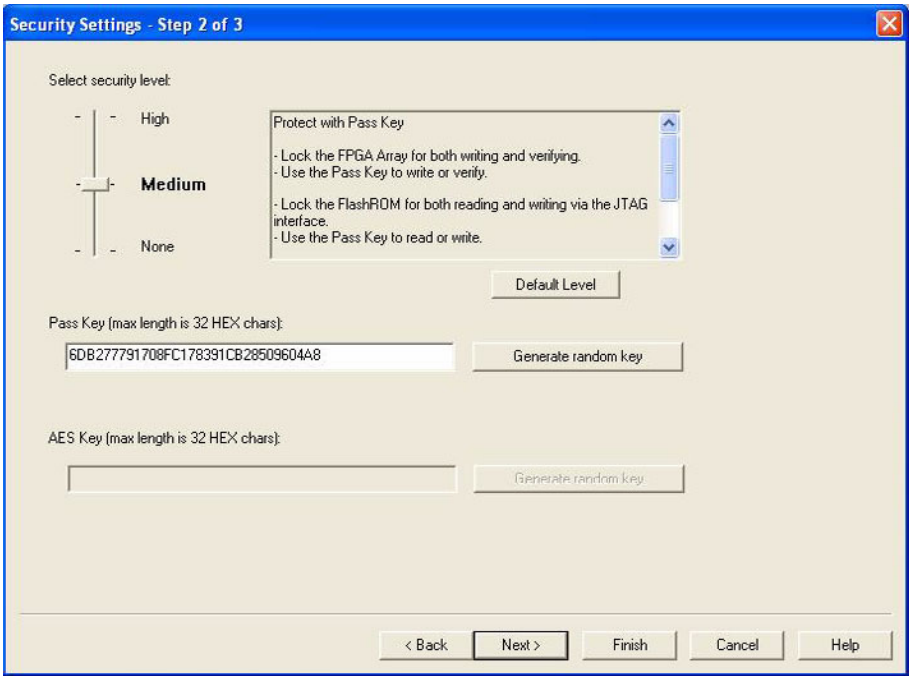
Instance Name	Instance Location	Program	Embedded Flash Memory Configuration File
nvm_system_inst/NVM_INST	1	☑	C:\Actelprj\PwrM\smartgen\nvm_sysm\nvm_sysm.efc

☐ Programming previously secured device(s)

Silicon signature (max length is 8 HEX chars):

Figure 7 • **All Silicon Features Checked**

2. Choose the appropriate security level setting and enter a FlashLock Pass Key. The default is the **Medium** security level (Figure 8). Click **Next**.



Security Settings - Step 2 of 3

Select security level:

High

Medium

None

Protect with Pass Key

- Lock the FPGA Array for both writing and verifying.
- Use the Pass Key to write or verify.

- Lock the FlashROM for both reading and writing via the JTAG interface.
- Use the Pass Key to read or write.

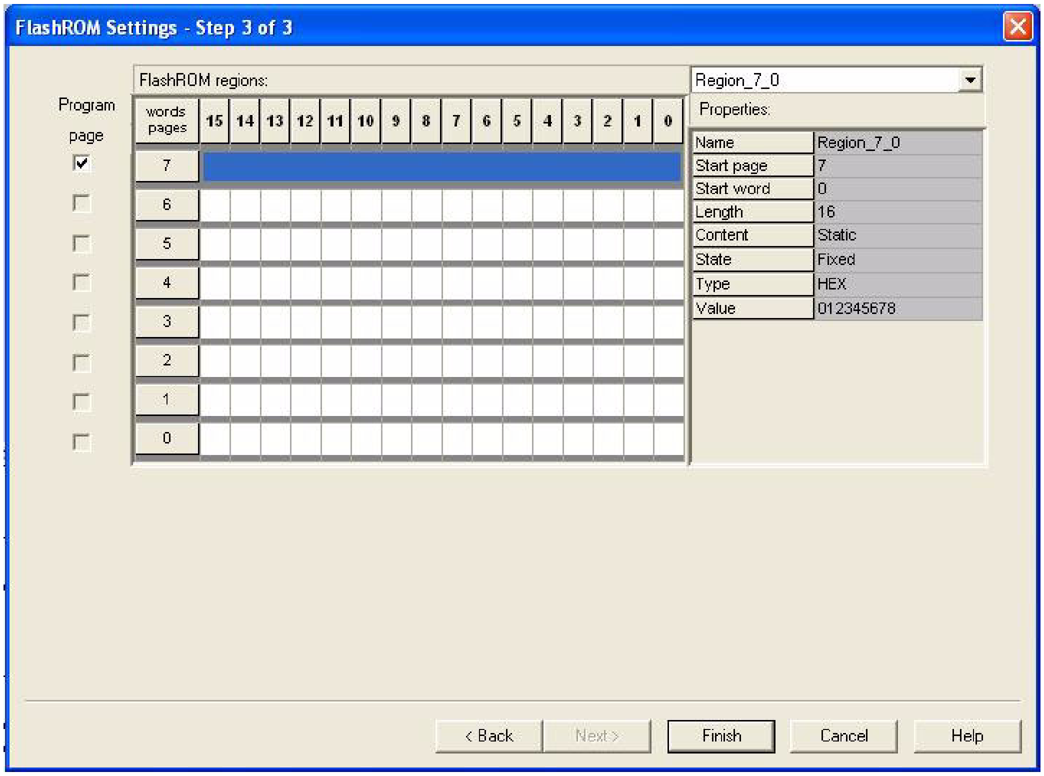
Pass Key (max length is 32 HEX chars):

6DB277791708FC178391CB28509604A8

AES Key (max length is 32 HEX chars):

Figure 8 • **Medium Security Level Selected**

3. Choose the desired settings for the FlashROM configurations to be programmed (Figure 9). Click **Finish** to generate the STAPLE programming file for the design.



FlashROM Settings - Step 3 of 3

FlashROM regions:

Program page	words pages	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
☑	7																
□	6																
□	5																
□	4																
□	3																
□	2																
□	1																
□	0																

Region_7_0

Properties:

Name	Region_7_0
Start page	7
Start word	0
Length	16
Content	Static
State	Fixed
Type	HEX
Value	012345678

Figure 9 • **FlashROM Configuration Settings**

Generation of Security Header Programming File Only – Application 2

As mentioned in the "Application 2: Nontrusted Environment – Unsecured Location" section on page 5, the designer may employ FlashLock Pass Key protection or FlashLock Pass Key with AES encryption on the device before sending it to a nontrusted or unsecured location for device programming. To achieve this, the user needs to generate a programming file containing only the security settings desired (Security Header programming file).

Note: If AES encryption is configured, FlashLock Pass Key protection must also be configured.

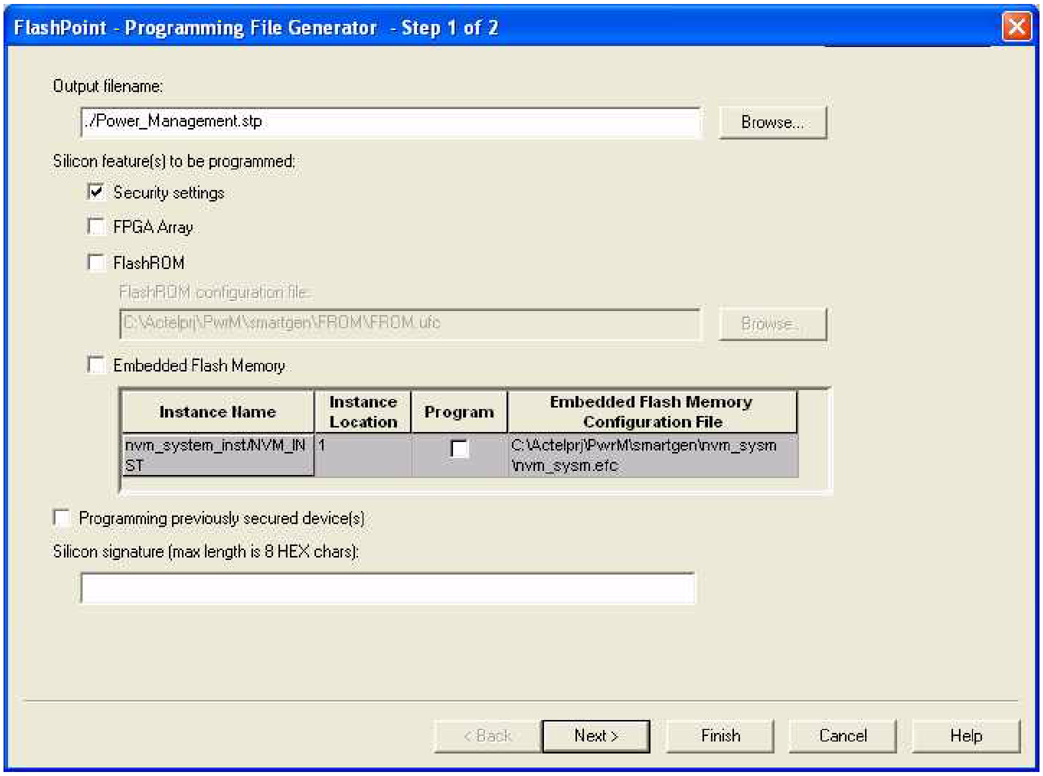
The available security options are indicated in Table 2.

Table 2 • **FlashLock Security Options**

Security Option	**FlashROM Only**	**FPGA Core Only**	**FB Core Only**	**All**
No AES / No FlashLock	–	–	–	–
FlashLock	✓	✓	✓	✓
AES and FlashLock	✓	✓	✓	✓

For this scenario, generate the programming file as follows:

1. Select only the **Security settings** option, as indicated in Figure 10. Click **Next**.



FlashPoint - Programming File Generator - Step 1 of 2

Output filename:

./Power_Management.stp

Silicon feature(s) to be programmed:

☑ Security settings

□ FPGA Array

□ FlashROM

FlashROM configuration file:

C:\Actelprj\PwrM\smartgen\FROM\FROM.ufc

□ Embedded Flash Memory

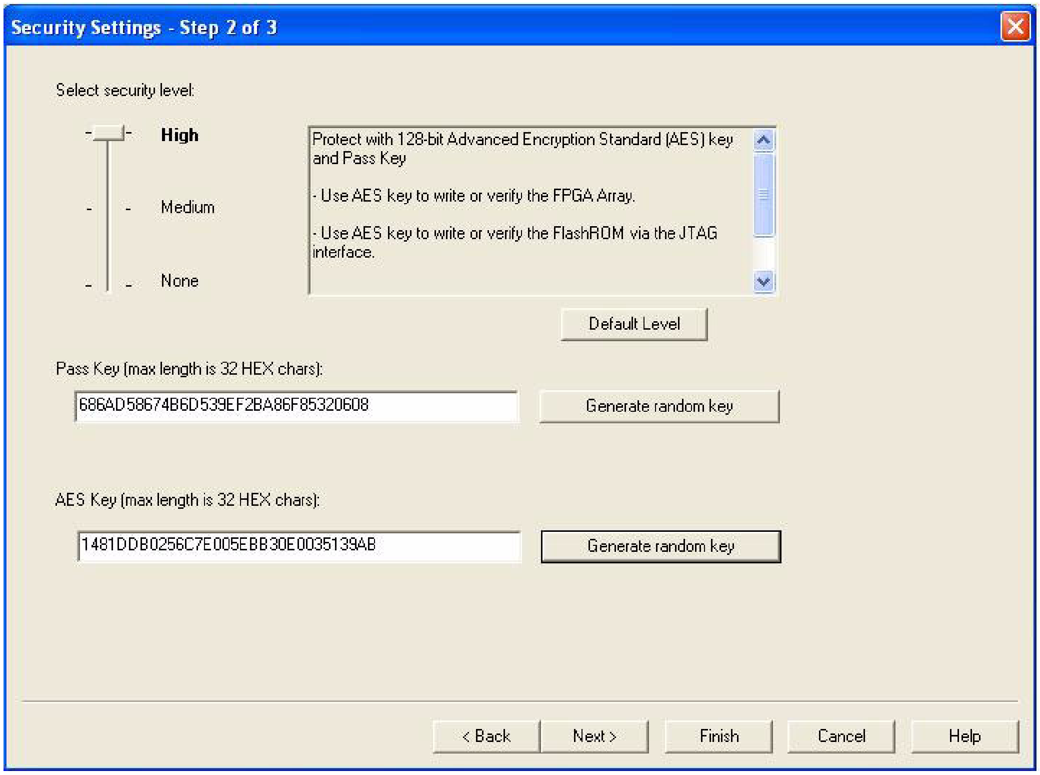
Instance Name	Instance Location	Program	Embedded Flash Memory Configuration File
nvm_system_inst/NVM_INST	1	□	C:\Actelprj\PwrM\smartgen\nvm_sysm\nvm_sysm.efc

□ Programming previously secured device(s)

Silicon signature (max length is 8 HEX chars):

Figure 10 • **Security Settings**

2. Choose the desired security level setting and enter the key(s) (Figure 11).
   – The **High** security level employs FlashLock Pass Key with AES Key protection.
   – The **Medium** security level employs FlashLock Pass Key protection only.



Security Settings - Step 2 of 3

Select security level:

High

Medium

None

Protect with 128-bit Advanced Encryption Standard (AES) key and Pass Key

- Use AES key to write or verify the FPGA Array.

- Use AES key to write or verify the FlashROM via the JTAG interface.

Pass Key (max length is 32 HEX chars):

686AD58674B6D539EF2BA86F85320608

AES Key (max length is 32 HEX chars):

1481DDB0256C7E005EBB30E0035139AB

Figure 11 • **High Security Level to Implement Both FlashLock Pass Key and AES Key Protection**

Table 3 shows all available options.

3. When done, click **Finish** to generate the Security Header programming file.

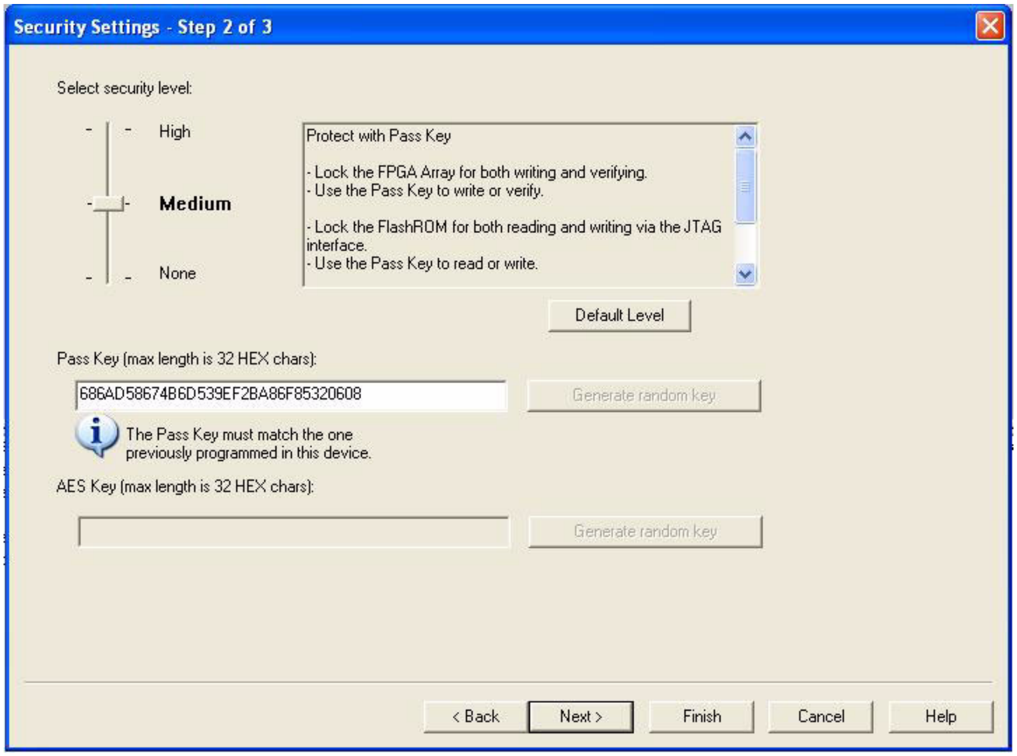
Table 3 • **Header File Security Options**

Security Option	**FlashROM Only**	**FPGA Core Only**	**FB Core Only**	**All**
No AES / No FlashLock	✓	✓	✓	✓
FlashLock	✓	✓	✓	✓
AES and FlashLock	✓	✓	✓	✓

Reprogramming Devices

Previously programmed Fusion devices can be reprogrammed using the steps in the "Generation of the Programming File in a Trusted Environment – Application 1" section on page 9 and "Generation of Security Header Programming File Only – Application 2" section on page 11. In the case where a FlashLock Pass Key has been programmed previously, the user must generate the new programming file with a FlashLock Pass Key that matches the one previously programmed into the device. The software will check the FlashLock Pass Key in the programming file against the FlashLock Pass Key in the device. The keys must match before the device can be unlocked to perform further programming with the new programming file.

Figure 7 on page 9 shows the option **Programming previously secured device(s)**, which the user should check before proceeding. Upon going to the next step, the user will be notified that the same FlashLock Pass Key needs to be entered, as shown in Figure 12.



Protect with Pass Key

- Lock the FPGA Array for both writing and verifying.
- Use the Pass Key to write or verify.

- Lock the FlashROM for both reading and writing via the JTAG interface.
- Use the Pass Key to read or write.

Pass Key (max length is 32 HEX chars): 686AD58674B6D539EF2BA86F85320608

The Pass Key must match the one previously programmed in this device.

AES Key (max length is 32 HEX chars):

Figure 12 • **FlashLock Pass Key**

It is important to note that when the security settings need to be updated, the user also needs to select the **Security settings** check box in Figure 7 on page 9 to modify the security settings. The user must consider the following:

- If only a new AES key is necessary, the user must re-enter the same Pass Key previously programmed into the device in Designer and then generate a programming file with same Pass Key and a different AES key. This ensures the programming file can be used to access and program the device.
- If a new Pass Key is necessary, the user can generate a new programming file with a new Pass Key (with the same or a new AES key if desired). However, for programming, the user must first load the original programming file with the Pass Key that was previously used to unlock the device. Then the new programming file can be used to program the new security settings.

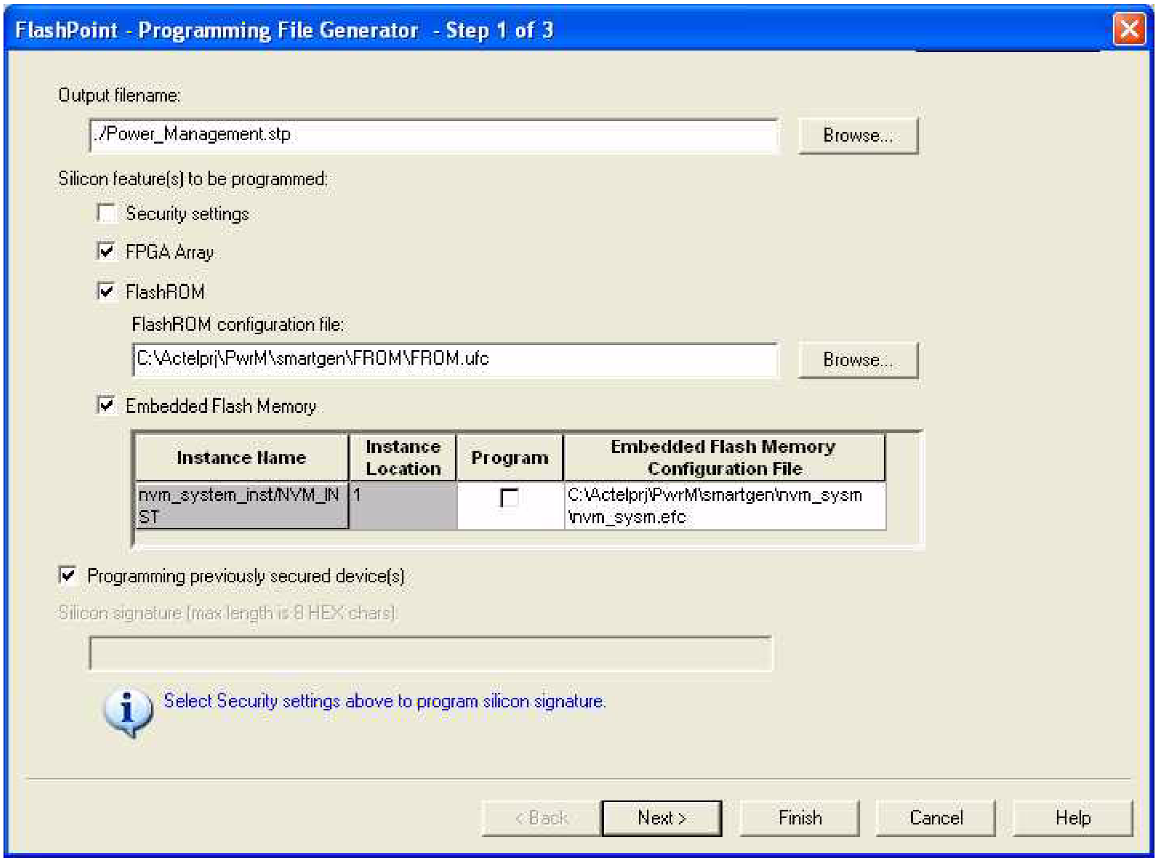
Generation of Programming Files with AES Encryption – Application 3

This section discusses how to generate design content programming files needed specifically at unsecured or remote locations to program device(s) with a security header (FlashLock Pass Key and AES Key) already programmed ("Application 2: Nontrusted Environment – Unsecured Location" section on page 5 and "Application 3: Nontrusted Environment – Field Updates/Upgrades" section on page 6). In this case, the encrypted programming file must correspond to the AES key already programmed into the device. If AES encryption was previously selected to encrypt the FlashROM, the FB, and the FPGA array, then AES encryption must be set when generating the programming file for them. AES encryption can be applied to the FlashROM only, the FB only, the FPGA array only, or all. The user must ensure both the FlashLock Pass Key and the AES key match those already programmed to the device(s), and all security settings must match what was previously programmed. Otherwise, the encryption and/or device unlocking will not be recognized when attempting to program the device with the programming file.

The generated programming file will be AES-encrypted.

In this scenario, generate the programming file as follows:

1. Select the **Security settings** and the portion of the device to be programmed (Figure 13). Select **Programming previously secured device(s)**. Click **Next**.



FlashPoint - Programming File Generator - Step 1 of 3

Output filename:

./Power_Management.stp

Silicon feature(s) to be programmed:

□ Security settings

☑ FPGA Array

☑ FlashROM

FlashROM configuration file:

C:\Actelprj\PwrM\smartgen\FROM\FROM.ufc

☑ Embedded Flash Memory

Instance Name	Instance Location	Program	Embedded Flash Memory Configuration File
nvm_system_inst/NVM_INST	1	□	C:\Actelprj\PwrM\smartgen\nvm_sysm\nvm_sysm.efc

☑ Programming previously secured device(s)

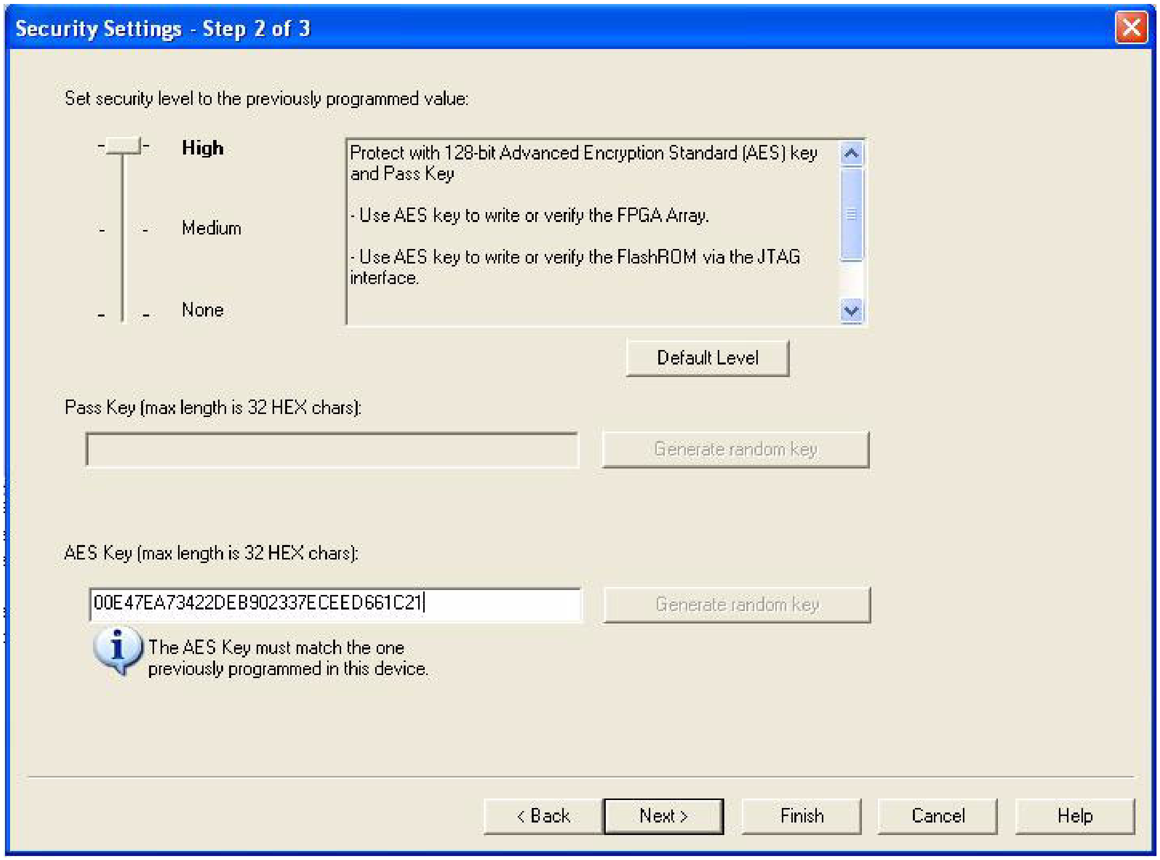
Silicon signature (max length is 8 HEX chars):

Select Security settings above to program silicon signature.

Note: *The settings in this figure are used to show the generation of an AES-encrypted programming file for both the FPGA array and FlashROM contents. One or both locations may be selected for encryption.*

Figure 13 • **FlashLock Pass Key and AES Key Protected Programming Files Settings**

Choose the **High** security level to employ both FlashLock Pass Key and AES Key protection (Figure 14). Enter both keys and click **Next**.



Security Settings - Step 2 of 3

Set security level to the previously programmed value:

Protect with 128-bit Advanced Encryption Standard (AES) key and Pass Key

- Use AES key to write or verify the FPGA Array.

- Use AES key to write or verify the FlashROM via the JTAG interface.

Pass Key (max length is 32 HEX chars):

AES Key (max length is 32 HEX chars): 00E47EA73422DEB902337ECEED661C21

The AES Key must match the one previously programmed in this device.

Figure 14 • **Security Level Set High for FlashLock Pass Key and AES Key Protection**

Programming with this file is intended for an unsecured environment. The AES key encrypts the programming file with the same AES key already used in the device and utilizes it to program the device.

Generated Programming File Header Definition

In each STAPL programming file generated, there will be information about how the AES key and the FlashLock Pass Key are configured. Table 4 shows the header definitions in STAPL programming files for different security levels.

Table 4 • **STAPL File Header Definitions for Various Security Levels**

Security Level	STAPL File Header Definition
No security (no FlashLock Pass Key or AES key)	NOTE "SECURITY" "Disable";
FlashLock Pass Key with no AES key	NOTE "SECURITY" "KEYED ";
FlashLock Pass Key with AES key	NOTE "SECURITY" "KEYED ENCRYPT ";

Conclusion

The new and enhanced security features offered in Actel Fusion devices provide state-of-the-art security to designs programmed into these Flash-based devices. Actel Fusion devices employ the encryption standard used by NIST and the U.S. government – AES using the 128-bit Rijndael algorithm.

The combination of an on-chip AES decryption engine and Actel FlashLock technology provides the highest level of security against invasive attacks and design theft, implementing the most robust and secure ISP solution. These security features protect IP within the FPGA and protect the system from cloning, wholesale "black box" copying of a design, invasive attacks, and explicit IP or data theft.

Glossary

Term	Explanation
Security Header Programming File	Programming file used to program the AES key and/or FlashLock Pass Key into the FlashROM, FB, and/or FPGA core of the device.
AES (Encryption) Key	128-bit key defined by the user when the AES encryption option is set in the Actel Designer software when generating the programming file.
FlashLock Pass Key	128-bit key defined by the user when the FlashLock option is set in the Actel Designer software when generating the programming file. The user must enter the FlashLock Pass Key to the programming software before programming (and actual decryption if the AES option is configured) is done.
FlashLock	Security features that protect the device content from attacks. This is achieved by the following: • Flash technology that does not require an external bitstream to program the device • FlashLock Pass Key that secures device content and prevents access to the device as necessary • AES key that allows secure device reprogrammability

References

National Institute of Standards and Technology. "ADVANCED ENCRYPTION STANDARD (AES) Questions and Answers." 28 January 2002. <http://csrc.nist.gov/CryptoToolkit/aes/aesfact.html> (10 January 2005).

Related Documents

Application Notes

Fusion FlashROM

http://www.actel.com/documents/Fusion_FROM_AN.pdf

Actel

http://www.actel.com

Actel Corporation

2061 Stierlin Court
Mountain View, CA
94043-4655 USA
Phone 650.318.4200
Fax 650.318.4600

Actel Europe Ltd.

Dunlop House, Riverside Way
Camberley, Surrey GU15 3YL
United Kingdom
Phone +44 (0) 1276 401 450
Fax +44 (0) 1276 401 490

Actel Japan
www.jp.actel.com
EXOS Ebisu Bldg. 4F
1-24-14 Ebisu Shibuya-ku
Tokyo 150 Japan
Phone +81.03.3445.7671
Fax +81.03.3445.7668

Actel Hong Kong
www.actel.com.cn
Two Pacific Place
88 Queensway, Admiralty
Hong Kong
Phone +852.2185 6460
Fax +852.2185 6488

51900130-0/4.06