



Intel® Quartus® Prime Pro Edition 用户指南

Programmer

针对 Intel® Quartus® Prime 设计套件的更新: **19.4**

本翻译版本仅供参考，如果本翻译版本与其英文版本存在差异，则以英文版本为准。某些翻译版本尚未更新对应到最新的英文版本，请参考[英文版本](#)以获取最新信息。



在线版本



发送反馈

UG-20134

ID: **683039**

版本: **2019.12.16**

内容

1. Intel® Quartus® Prime Programmer 用户指南.....	4
1.1. 生成器件主要编程文件.....	5
1.2. 生成辅助编程文件.....	6
1.2.1. 生成辅助编程文件 (Programming File Generator)	6
1.2.2. 生成辅助编程文件 (Convert Programming File Dialog Box)	10
1.3. 使能 Intel Stratix 10 器件的 Bitstream Security.....	16
1.3.1. 使能 Bitstream Authentication (Programming File Generator)	16
1.3.2. 指定其他物理安全设置 (Programming File Generator)	18
1.3.3. 使能比特流加密 (Programming File Generator)	19
1.4. 使能比特流加密或针对 Intel Arria 10 和 Intel Cyclone 10 GX 器件的压缩.....	21
1.5. 生成用于局部重配置的编程文件.....	22
1.5.1. 生成 PR 比特流文件.....	22
1.5.2. 部分重新配置比特流兼容性检查.....	24
1.5.3. 原始二进制编程文件字节序列传输实例.....	25
1.5.4. 生成由多个.pmsf 文件合并的.pmsf 文件.....	26
1.6. 脚本支持.....	26
1.6.1. quartus_pfg 命令行工具.....	26
1.6.2. quartus_cpf 命令行工具.....	27
1.7. 生成编程文件修订历史.....	27
2. 使用 Intel Quartus Prime Programmer.....	29
2.1. Intel Quartus Prime Programmer.....	29
2.2. 编程和配置模式.....	30
2.3. 器件配置基本步骤.....	30
2.4. 指定编程硬件设置.....	32
2.4.1. JTAG Chain Debugger 工具.....	33
2.4.2. 编辑未知器件的详细信息.....	34
2.4.3. 在 Linux 上运行 JTAG Daemon.....	34
2.5. 使用 Flash Loader 编程.....	34
2.5.1. 指定 Flash 分区.....	35
2.5.2. 完全擦除 Flash 存储器扇区.....	35
2.6. 使用 Project Hash 验证编程文件源.....	36
2.6.1. 获得 Intel Arria 10 器件的 Project Hash.....	36
2.7. 使用 PR 比特流安全验证 (Intel Stratix 10 和 Intel Agilex 设计)	36
2.8. 独立运行的 Programmer.....	38
2.8.1. Stand-Alone Programmer 存储器消耗.....	38
2.9. Programmer 设置参考.....	38
2.9.1. 器件&管脚选项对话框.....	38
2.9.2. 更多安全选项对话框.....	45
2.9.3. 输出文件选项卡设置 (Programming File Generator)	45
2.9.4. 输入文件选项卡设置 (Programming File Generator)	46
2.9.5. 比特流联合签字安全设置 (Programming File Generator)	46
2.9.6. 配置器件选项卡设置.....	47
2.9.7. 添加分区对话框 (Programming File Generator)	47

2.9.8. 转换编程文件对话框.....	48
2.9.9. 压缩和加密设置 (Convert Programming File)	48
2.9.10. SOF 数据属性对话框 (Convert Programming File)	49
2.9.11. 选择器件 (Flash Loader) 对话框.....	49
2.10. 脚本支持.....	49
2.10.1. jtagconfig 调试工具.....	50
2.11. 编程 Intel FPGA 器件修订历史.....	50
2.12. Intel Quartus Prime Pro 版用户指南: Programmer 文档存档.....	52
A. Intel Quartus Prime Pro Edition 用户指南.....	53

1. Intel® Quartus® Prime Programmer 用户指南

Intel® Quartus® Prime Programmer (下载配置工具) 允许编程和配置 Intel CPLD, FPGA 以及配置器件。经过完整设计编译, 可通过 **Assembler** 生成主要的器件编程文件, 然后使用 **Programmer** 将编程文件加载到器件。本用户指南详细介绍 Intel FPGA 编程文件的生成以及 Intel Quartus Prime Programmer 的使用。

图 1. Intel Quartus Prime Programmer



相关链接

- [生成器件主要编程文件](#) (第 5 页)
- [生成辅助编程文件](#) (第 6 页)
- [使能 Intel Stratix 10 器件的 Bitstream Security](#) (第 16 页)
- [使用 Intel Quartus Prime Programmer](#) (第 29 页)
- [使用 Flash Loader 编程](#) (第 34 页)

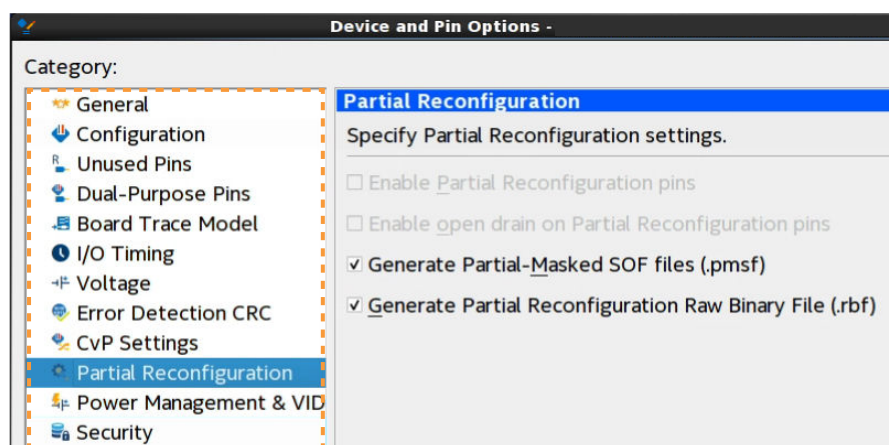
1.1. 生成器件主要编程文件

默认情况下，Compiler 的 Assembler 模块在完整编译结束时生成主要编程文件。或者，可在设计布局布线之后的任何时间单独启动 Assembler 以生成主要器件编程文件，例如用于配置 Intel FPGA 的 SRAM Object Files (.sof)。

按照以下步骤生成主要器件编程文件：

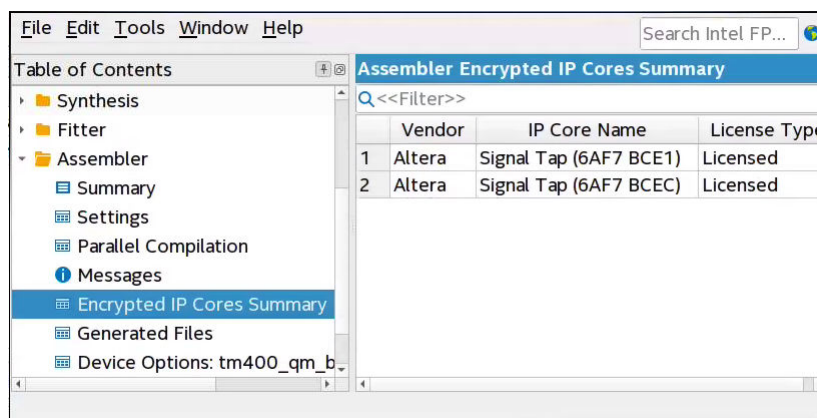
1. 指定使能主要器件编程文件中各功能的编程选项，例如 **Configuration**，**Error Detection CRC** 和器件 **Security** 选项，点击 **Assignments > Device > Device & Pin Options**。器件&管脚选项对话框 (第 38 页)介绍全部选项。⁽¹⁾

图 2. Device & Pin Options 对话框 (Intel Stratix® 10 设计)



2. 要生成主要器件编程文件，请点击 **Processing > Start > Start Assembler**，或在 Compilation Dashboard 上双击 **Assembler**。Assembler 按照指定选项生成编程文件。
3. 运行 Assembler 后，查看编程文件的详细信息，包括编程文件 Summary 和 Compilation Report 中 Assembler 文件夹的 Encrypted IP 信息。

图 3. Assembler 报告



⁽¹⁾ Intel Agilex™ 器件尚无安全选项。

1.2. 生成辅助编程文件

生成主要器件编程文件后，可选择生成 1 个或多个派生编程文件以用于备用器件配置，例如 flash 编程，局部重新配置，远程系统更新，Configuration via Protocol (CvP)，或硬核处理系统 (HPS) 内核配置。

可使用 **Programming File Generator** 或 **Convert Programming Files** 对话框生成辅助编程文件：

- **Programming File Generator** 支持高级编程功能，并尤其适用于 Intel Agilex，Intel Stratix® 10，Intel MAX® 10 和 Intel Cyclone® 10 LP 器件。
- **Convert Programming Files** 对话框支持先于 Intel Stratix 10 器件发布的所有器件。

表 1. 辅助编程文件生成器

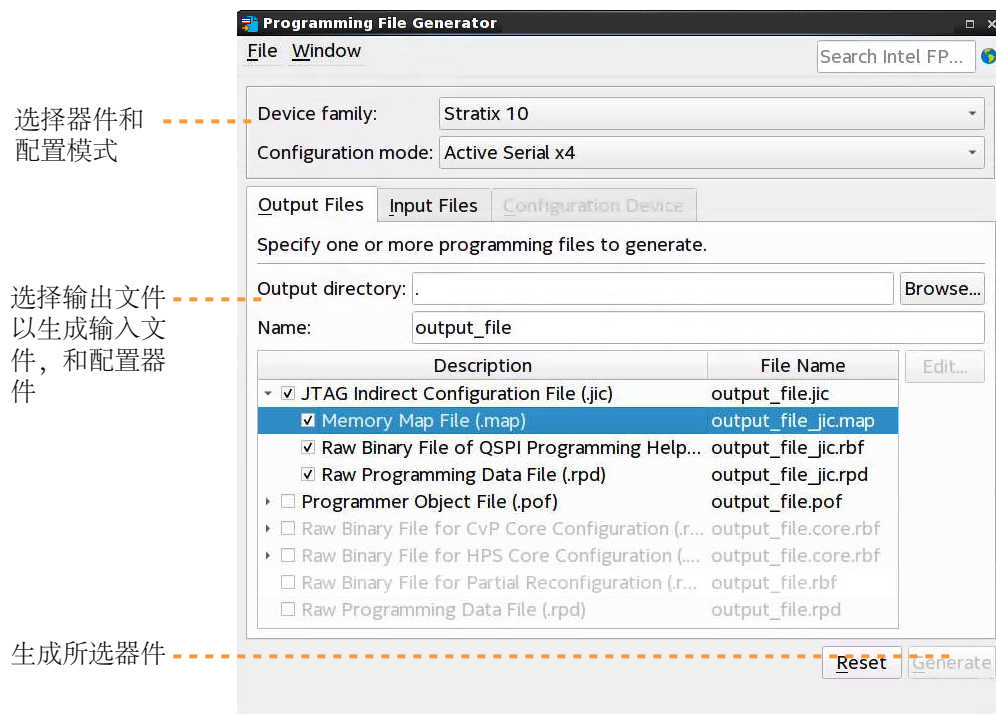
	编程文件生成器 (Programming File Generator)	转换编程文件	
器件支持	<ul style="list-style-type: none"> Intel Agilex Intel Stratix 10 Intel MAX 10 Intel Cyclone 10 LP 	<ul style="list-style-type: none"> Intel Arria® 10 Intel Cyclone 10GX 和 LP Intel MAX 10 	APEX20K, Arria II GX 和 GZ, Arria V, Cyclone, Cyclone II, Cyclone III 和 LS, Cyclone IV E 和 GX, Cyclone V, HardCopy® III, HardCopy II, HardCopy IV, MAX V, Stratix, Stratix II, Stratix III, Stratix IV, Stratix V

1.2.1. 生成辅助编程文件 (Programming File Generator)

按照如下步骤使用 **Programming File Generator** 生成辅助编程文件取得备用器件编程方式。

1. 为您的设计生成主要编程文件，如[生成器件主要编程文件](#) (第 5 页)所述。
2. 点击 **File > Programming File Generator**。
3. 对于 **Device family**，选择您的目标器件。**Programming File Generator** 中的可用选项会根据您的器件和配置模式选择进行动态更改。
4. 对于 **Configuration mode**，请选择您器件支持的 Active Serial 模式。[配置模式 \(Programming File Generator\)](#) (第 9 页)中介绍所有模式。
5. 在 **Output Files** 选项卡上，启用生成所需文件的复选框。此时，**Input Files** 选项卡可用。[辅助编程文件 \(Programming File Generator\)](#) (第 9 页)中介绍所有输出文件。
6. 指定生成文件的 **Output directory** 和 **Name**。[输出文件选项卡设置 \(Programming File Generator\)](#) (第 45 页)中介绍所有选项。

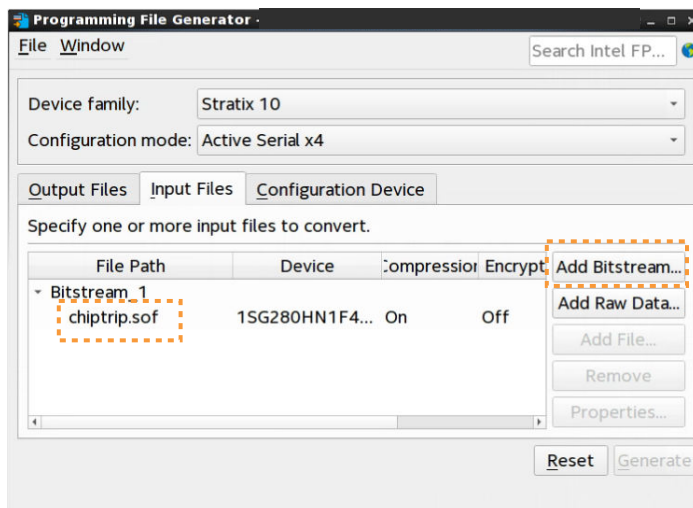
图 4. 编程文件生成器 (Programming File Generator)



7. 要指定 .sof 文件包含配置比特流数据，请在 **Input Files** 选项卡上，点击 **Add Bitstream**。要包含原始数据，请点击 **Add Raw Data** 并指定 Hexadecimal (Intel-Format) File (.hex)或 Binary (.bin)文件。
8. 为文件指定比特流验证授权或加密安全设置，请选择 .sof 并点击 **Properties**，如[使能 Bitstream Authentication \(Programming File Generator\)](#) (第 16 页)中所述。(2)

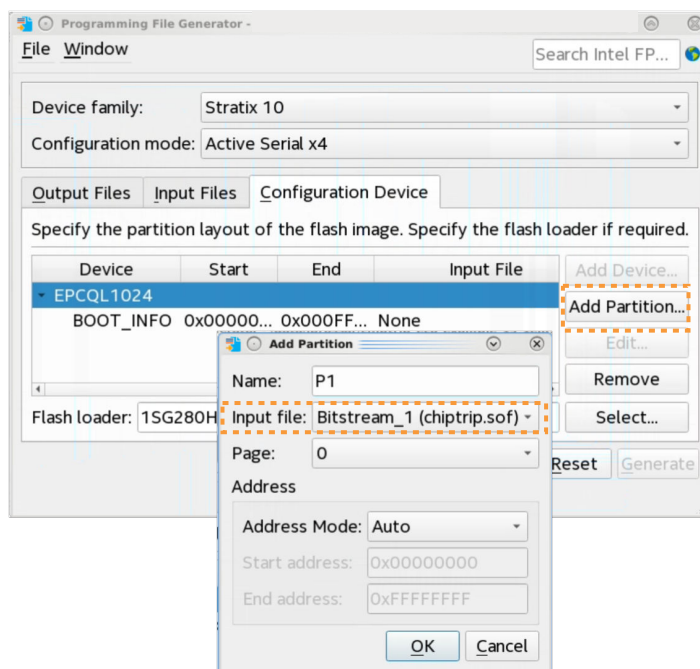
(2) Intel Agilex 器件尚无安全选项。

图 5. 输入文件选项卡



9. 指定占用 Flash 存储器分区的 .sof 文件，请在 **Configuration Device** 选项卡上点击 **Add Partition**。添加分区对话框 ([Programming File Generator](#)) (第 47 页)中介绍所有选项。

图 6. 添加 Flash 分区



10. 要选择支持的 Flash 存储器件并预定义编程流程，请在 **Configuration Device** 选项卡上点选 **Add Device**。或者，点击 **<<new device>>** 定义一个新的 Flash 存储器件和编程流程。[配置器件选项卡设置](#) (第 47 页) 介绍所有设置。
11. 单击 **Flash Loader** 的 **Select** 按钮并选择器件控制加载 Flash 存储器件。[选择器件 \(Flash Loader\) 对话框](#) (第 49 页) 中介绍所有设置。
12. 在 **Programming File Generator** 中指定全部选项后，**Generate** 按钮启用。点击 **Generate** 创建文件。

1.2.1.1. 配置模式 (Programming File Generator)

在 **Programming File Generator** 中从如下 **Configuration modes** 选择一个模式生成辅助编程文件：

表 2. Programming File Generator 配置模式

编程模式	说明	支持器件
Active Serial x4	用于非易失性存储器和 4-pin 接口的低成本串联配置器件中的配置数据储存。串行配置器件提供一个串行接口访问配置数据。器件配置期间， Intel Stratix 10 器件通过串行接口读取配置数据，必要时解压缩数据，并配置其 SRAM 单元。	Intel Agilex Intel Stratix 10
AVST x8	Avalon® -ST 配置模式使用外部主机，例如，微处理器或 Intel MAX 10 器件。外部主机控制外部存储器（如，Flash 存储器）向 FPGA 的配置数据传输。控制配置处理的设计位于外部主机。可使用 PFL II IP 核以及 Intel MAX 10 器件作为主机从配置 Intel Stratix 10 FPGA 的 Flash 存储器件读取配置数据。	
AVST x16		
AVST x32		
1-Bit Passive Serial	外部控制器通过串行数据流将配置数据传递到 1 个或多个 FPGA 器件。FPGA 器件是一个从器件，通过 5 线接口连接外部控制器。外部控制器可为智能主机（intelligent host），例如微控制器或 CPU。	Intel Cyclone 10 LP
Active Serial	用于非易失性存储器和 4-pin 接口的低成本串联配置器件中的配置数据储存。	
Internal Configuration	使用 .pof 文件，通过下载电缆线 Intel Quartus Prime Programmer 对 Intel MAX 10 器件的 Configuration Flash Memory (CFM)和 User Flash Memory (UFM)进行内部配置。	Intel MAX 10

1.2.1.2. 辅助编程文件 (Programming File Generator)

生成主要器件编程文件后，可使用 **Programming File Generator** 生成如下辅助器件编程文件，用于备用器件配置模式：

表 3. Programming File Generator 输出文件类型

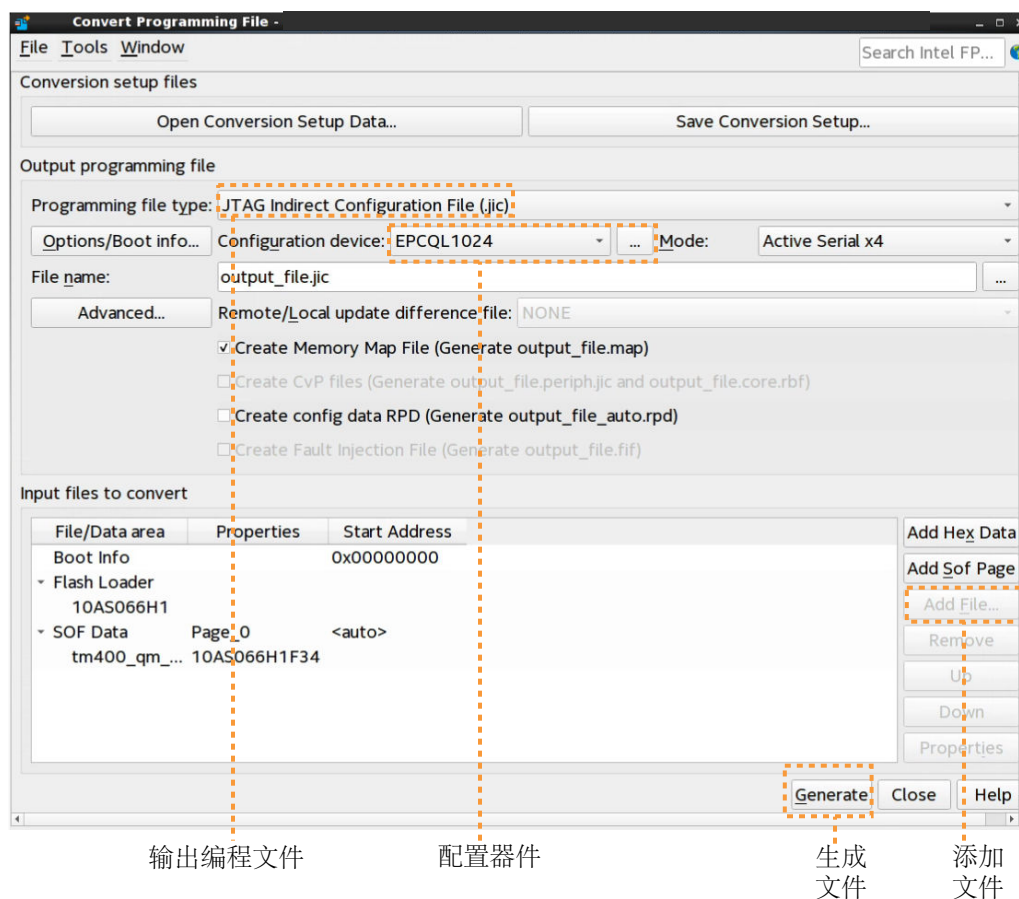
编程文件类型	扩展名	说明
SRAM 的十六进制 (Intel - 格式) 输出文件	.hexout	Intel 十六进制格式的 ASCII 文本文件包含用于对并行数据源（例如配置器件或大容量存储器件）进行编程的配置数据。并行数据源依次配置基于 SRAM 的 Intel 器件。
JTAG 间接配置文件	.jic	专利 Intel FPGA 文件类型存储通过 Intel FPGA JTAG 管脚进行编程的串行 Flash 编程数据。该方式仅支持 Active Serial 配置。对 Flash 进行编程之前，Programmer 首先通过 Serial Flash Helper Design 配置 FPGA。
映射文件	.map	文本文件，包含用于编程的配置器件中，存储器内存储的页面和数据的字节地址。
Programmer 目标文件	.pof	Programmer 使用的二进制文件，通过主动串行头 (header) 对 Flash 存储器件进行编程，或通过 Parallel Flash Loader Intel FPGA IP 对 Flash 存储器件进行编程。
继续...		

编程文件类型	扩展名	说明
原始二进制文件	.rbf	用于第三方数据源，局部重配置，或 HPS 数据源的配置比特流文件。支持 Passive Serial (PS)和 Avalon -Streaming (AVST)模式。
原始二进制文件用于 CvP 核配置	.rbf	二进制文件，包含由 CvP 第 2 阶段的配置 (CRAM) 编程的逻辑。该内核比特流为 .rbf 格式。
原始二进制文件用于 HPS 核配置		二进制文件，包含由 HPS 第 2 阶段的配置 (CRAM) 编程的逻辑。该内核比特流为 .rbf 格式。
原始编程数据文件	.rpd	为使用第三方编程硬件的配置储存数据。可从 .pof 或 .sof 生成 Raw Programming Data File (原始编程文件) 。.rpd 文件是 .pof 或 .jic 的子集，其仅包含用于 EPCS 或 EPCQ 串行配置器件 Active Serial 配置方案和远程系统更新的指定器件二进制编程数据。
表格文本文件	.ttf	TTF 包含 Raw Binary File (.rbf)的十进制等效项。

1.2.2. 生成辅助编程文件 (Convert Programming File Dialog Box)

可使用 **Convert Programming File** 对话框生成辅助编程文件获得备用器件编程方式。例如，生成 .jic 文件用于 Flash 编程，生成 .rbf 文件用于局部重新配置，或者 .rpd 文件用于第三方下载配置工具 (programmer) 的配置。

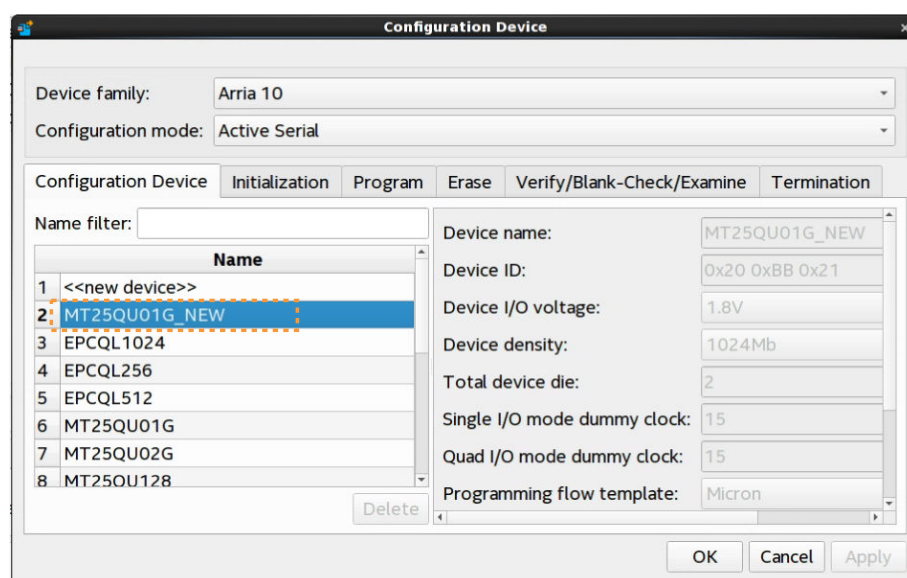
图 7. Convert Programming File Dialog Box (转换编程文件对话框)



Convert Programming File 对话框中的可用选项会根据您的器件和配置模式的选择动态更换。

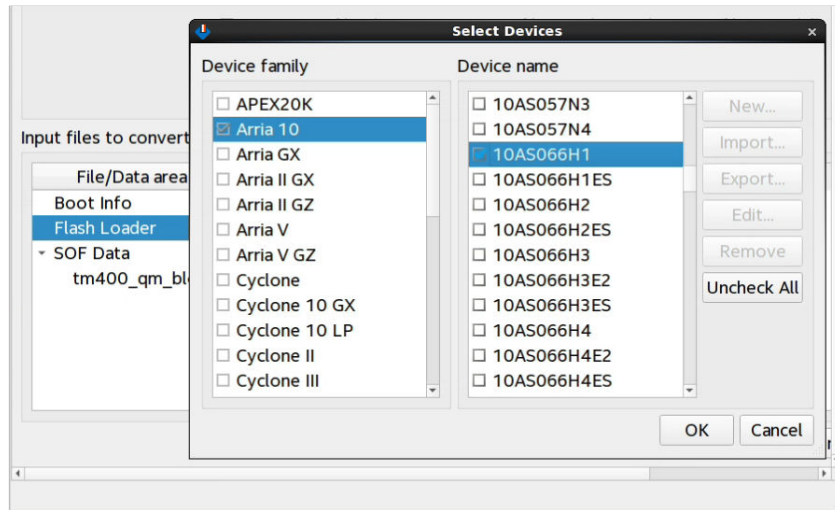
1. 为您的设计生成主要编程文件，如[生成器件主要编程文件](#)（第 5 页）中所述。
2. 点击 **File > Convert Programming Files**。
3. 在 **Output programming file** 下，选择需要生成的 **Programming file type**。[辅助编程文件（转换编程文件）](#)（第 12 页）中说明所有文件选项。
4. 为生成的文件指定 **File name** 和输出目录(…)。
5. 关于配置 **Mode**，请选择 **Active Serial x4** 或 **Active Serial**。[配置模式（Convert Programming Files）](#)（第 13 页）中介绍了所有模式。
注意： Intel Stratix 10 器件仅支持 **Active Serial x4**。
6. 指定 **Configuration device**，点击(…)选项选择支持的 Flash 存储器件和预定义编程流程。选择预定义器件后，不可修改任何设置。或者，点击<<new device>>定义一个新的 Flash 存储器件和编程流程。[配置器件选项卡设置](#)（第 47 页）中介绍所有设置。

图 8. Configuration Device Dialog Box（配置器件对话框）



7. **Input files to convert** 下，选择 **SOF Data** 项，然后点击 **Add File** 按钮。指定包含配置比特流数据的 .sof 文件。要包含原始数据，请点击 **Add Hex Data** 并指定一个 .hex 文件。
8. 要使能比特流压缩或加密安全设置，请选择 .sof 文件并点击 **Properties**，如[使能比特流加密或针对 Intel Arria 10 和 Intel Cyclone 10 GX 器件的压缩](#)（第 21 页）中所述。
9. 选择 **Flash Loader** 文本，并点击 **Add Device** 按钮。选择 ff 控制 Flash 器件加载的器件。

图 9. 选择 Flash Loader 器件



10. 在 **Convert Programming File** 对话框中指定所有选项后，点击 **Generate** 按钮创建文件。

1.2.2.1. 辅助编程文件（转换编程文件）

生成主要器件编程文件后，可使用 **Convert Programming Files** 对话框生成如下辅助器件编程文件，用于备用器件配置模式：

表 4. 输出文件类型

编程文件类型	扩展名	说明
CvP 文件	jic/.rbf	CvP 配置所需的文件
十六进制（Intel -格式）输出文件用于 SRAM	.hexout	Intel 十六进制格式的 ASCII 文本文件包含用于对并行数据源（例如配置器件或大容量存储器件）进行编程的配置数据。并行数据源依次配置基于 SRAM 的 Intel 器件。
JTAG 间接配置文件	.jic	专利 Intel FPGA 文件类型存储通过 Intel FPGA JTAG 管脚进行编程的串行 Flash 编程数据。该方式仅支持 Active Serial 配置。对 Flash 进行编程之前，Programmer 首先通过 Serial Flash Helper Design 配置 FPGA。
存储器映射文件	.map	包含存储在 EPC4，EPC8 或 EPC16 配置器件中的页面的字节地址和 HEX 数据。MAP File 存储 Main Block Data 和 Bottom Boot Data 项的页面起始和结束地址，以及 Main Block Data 项内部的页面起始和结束地址。
局部屏蔽（Masked）的 SRAM 目标文件	.pmsf	包含用于 PR 区域配置的局部屏蔽位。 .pmsf 文件包含创建 PR 比特流的所有信息。
合并掩码设置文件	.msf	包含用于 PR 设计中静态区域的掩码位。
编程器目标文件	.pof	二进制文件，包含对非易失性 Intel MAX 10，MAX V，MAX II，或者可配置 Intel FPGA 器件的 Flash 存储器件进行编程的数据。Programmer 由有效远程更新文件 .pof 和用于远程更新配置模式中编程配置器件的其他有效远程更新文件 .sof 组成。
继续...		

编程文件类型	扩展名	说明
原始二进制文件	.rbf	用于第三方数据源，局部重配置，或 HPS 数据源的配置比特流文件。支持 Passive Serial (PS) 和 Avalon -Streaming (AVST) 模式。
原始编程数据文件	.rpd	储存使用第三方编程硬件的配置数据。可从 .pof 或 .sof 生成 Raw Programming Data File (原始编程文件)。 .rpd 文件是 .pof 或 .sof 的子集，其仅包含用于 EPCS 或 EPCQ 串行配置器件 Active Serial 配置方案和远程系统更新的指定器件二进制编程数据。相比输出文件，.rpd 文件内容对换了一位。
Tabular Text File	.tff	TTF 包含 Raw Binary File (.rbf) 的十进制等效项。

1.2.2.2. 配置模式 (Convert Programming Files)

在 **Convert Programming Files** 中从如下 **Configuration modes** 选择一个模式生成辅助编程文件：

表 5. Convert Programming Files 配置模式

编程模式	说明
1-Bit/2-Bit/4-Bit/8-Bit Passive Serial	外部控制器通过串行数据流将配置数据传递到 1 个或多个 FPGA 器件。FPGA 器件是一个从器件，通过 5 线接口连接外部控制器。外部控制器可为智能主机，例如微控制器或 CPU，或 Intel Quartus Prime Programmer，EPC2 或 EPC16 配置器件。
Active Parallel	支持配置器件使用商用 16-bit 并行存储器控制配置接口。
Active Serial	用于具有非易失性存储器的低成本串联配置器件中的配置数据储存。串行配置器件提供一个串行接口访问配置数据。器件配置期间，器件通过串行接口读取配置数据，必要时解压缩数据，并配置其 SRAM 单元。
Active Serial x4	
AVST x8/x16/x32	Avalon -ST 配置模式使用外部主机，如微处理器或 Intel MAX 10 器件。外部主机控制外部存储器（如，Flash 存储器）向 FPGA 的配置数据传输。控制配置处理的设计位于外部主机。可使用 PFL II IP 核以及 Intel MAX 10 器件作为主机从配置 FPGA 的 Flash 存储器件读取配置数据。
Passive Parallel Synchronous	外部控制器，如 CPU，通过公共数据总线将设计数据加载到器件。数据被器件锁存于由 CPU 驱动的时钟信号首个上升沿处。接下来的 8 个下降时钟沿串化器件内锁存的数据。器件将数据的后 8 位字节锁存于时钟信号的每第 8 个上升沿直到器件完全配置。
Passive Parallel Asynchronous	外部控制器（如，CPU），通过公共数据总线将设计数据加载到器件。器件接受并行字节输入数据。外部控制器和器件之间的智能通信允许外部控制器配置器件。
Internal Configuration	使用 .pof 文件，通过下载电缆线 Intel Quartus Prime Programmer 对 Intel MAX 10 器件的 Configuration Flash Memory (CFM) 和 User Flash Memory (UFM) 进行内部配置。

1.2.2.3. 调试配置 (Convert Programming Files)

在 **Convert Programming Files** 对话框中单击 **Advanced** 选项，以调试文件转换配置。仅选择适用于设计中目标 Intel FPGA 器件的高级设置。

Advanced Options 对话框中的变化会影响 .pof，.jic，.rpd 和 .rbf 文件的生成。

下表说明 **Advanced Options** 设置：

表 6. 高级选项设置

选项设置	说明	值
Disable EPCS/EPCQ ID check	指示 FPGA 跳过 EPCS/EPCQ 芯片 ID 验证。	默认设置为 ON (EPCS/EPCQ ID 检查已启用)。
继续...		

选项设置	说明	值
	适用于所有器件上的单个和多个器件 AS 配置模式。	
Disable AS mode CONF_DONE error check	指示 FPGA 跳过 CONF_DONE 错误检查。 适用于所有器件上的单个和多个器件 AS 配置模式。	默认设置为 OFF (AS 模式 CONF_DONE 错误检查开启)。
Program Length Count adjustment	指定可应用于整个比特流已计算 PLC 的偏移。 适用于所有 FPGA 器件的单个和多个器件 (AS) 配置模式。	整数 (默认值 = 0)
Post-chain bitstream pad bytes	指定附加到整个比特流末尾的填充字节数。	如果最后一个器件的比特流未经压缩, 则默认值为 0。 否则, 默认值为 2
Post-device bitstream pad bytes	指定附加到器件整个比特流末尾的填充字节数。 适用于所有 FPGA 器件的单个器件配置模式。	0 或正整数。 默认为 0
Bitslice Padding Value	指定用于准备 bitslice (位片) 配置比特流的填充值, 以便所有 bitslice (位片) 配置链同步接收其最终配置数据位。 当使用已使能解压缩功能的 EPC 器件时, 请仅用于 2、4 和 8-bit PS 配置模式。 适用于所有支持加强型配置器件的 FPGA 器件。	0 或 1 默认值是 1。

下表列出失败配置可能出现的症状, 并说明高级选项对配置调试的必要性。

故障症状	禁用 EPCS/ EPCQ ID 检查	禁用 AS 模式 CONF_DONE 错误检查	PLC 设置	Post-Chain Bitstream 填充 字节	Post-Device Bitstream 填充 字节	Bitslice 填充值
配置周期后出现配置故障。	—	Yes	Yes	Yes (3)	Yes (4)	—
解压缩功能开启。	—	Yes	Yes	Yes (3)	Yes (4)	—
加密功能开启。	—	Yes	Yes	Yes (3)	Yes (4)	—
配置周期后, CONF_DONE 保持低 电平。	—	Yes	Yes (5)	Yes (3)	Yes (4)	—
配置周期后, CONF_DONE 即刻变 为高电平。	—	Yes	Yes (6)	—	—	—
即使 CONF_DONE 变 为高电平, FPGA 也未 进入用户模式。	—	—	—	Yes (3)	Yes (4)	—
配置周期开头出现配 置故障。	Yes	—	—	—	—	—
EPCS128	Yes	—	—	—	—	—
压缩功能使能时, 使 用 Intel Quartus Prime Convert	—	—	—	—	—	Yes
继续...						

(3) 仅用于多器件链

(4) 仅用于单器件链

(5) 从 PLC 设置的正偏移开始

(6) 从 PLC 设置的负偏移开始

故障症状	禁用 EPCS/ EPCQ ID 检查	禁用 AS 模式 CONF_DONE 错误检查	PLC 设置	Post-Chain Bitstream 填充 字节	Post-Device Bitstream 填充 字节	Bitslice 填充值
Programming File Utility 的 EPC 器件中 .pof 生成过程中出现故障。						

1.3. 使能 Intel Stratix 10 器件的 Bitstream Security

Intel Stratix 10 器件提供灵活而强健的安全性能，以保护敏感数据，知识产权和器件硬件免受物理性和远程攻击。Intel Stratix 10 器件体系结构支持比特流授权验证和加密安保功能。但凡使用授权验证和加密时，Assembler 会自动应用比特流压缩以减小文件大小。

- **Bitstream Authentication**—验证配置比特流和固件来源是否可靠。使能其他 co-signing（联合签字）器件固件授权验证以确保 HPS 或 FPGA 上仅运行已签字固件，并授权 HPS JTAG 进行调试。通过对 **Quartus Key File** 选项（**Device and Pin Options** 对话框）指定第一级签字链文件（.qky）使能授权安全性，如使能 **Bitstream Authentication**（**Programming File Generator**）（第 16 页）中所述。⁽⁷⁾
- **Bitstream Encryption**—使用 Advanced Encryption Standard (AES) 256-bit 或 384-bit 密钥保护专有或敏感数据不被查看或从配置比特流中提取。通过加密功能还可提供防止非侵入性攻击的侧边通道保护。可将所有者的 AES 密钥密钥储存在 eFuses 或 BBRAM 中。开启 **Enable programming bitstream encryption** 选项（**Device and Pin Options** 对话框）使能加密功能，如使能 **比特流加密**（**Programming File Generator**）（第 19 页）中所述。

表 7. Intel Stratix 10 Bitstream Authentication Files

专用术语	说明	扩展名
First Level Signature Chain Key File	生成的文件指定根密钥（.pem）和 1 个或多个设计签字密钥（.pem），是使用验证和加密时对比特流签字和允许访问 FPGA 的必要内容。	.qky
Root Key File	该生成文件将第一级签名链锚定到已知根密钥。FPGA 计算根条目的散列并检查其是否符合预期散列。Assembler 将根密钥附加到编程文件并将密钥存储在 eFuse 中。	.qky
Design Signing Key File	生成文件并附加到验证 SDM 中比特流的根密钥，以允许对带有未决比特流的器件进行配置。请对 FPGA 和 HPS 使用单独的设计签名密钥，以获得最高安全性。	.pem
Firmware Co-signing Key File	<install> \common\devinfo\programmer\firmware 中提供的文件，包含所有者签字和固件文件，可用于对固件签字以便在 FPGA 或 HPS 上运行。	.zip
Signed HPS Certificate File	指定一个安全的 HPS 调试证书，该证书允许通过访问 JTAG 接口调试 HPS。安全的 HPS 调试证书一直有效，直到掉电或重新配置器件。	.cert

注意: Intel Arria 10 和 Intel Cyclone 10 GX 器件不支持比特流加密。

相关链接

Intel Stratix 10 器件安全用户指南

了解有关器件安全性配置步骤的详细信息。

1.3.1. 使能 Bitstream Authentication（Programming File Generator）

比特流验证需要生成包括根密钥和 1 个或多个设计签字密钥的一级签字链（.qky）。根密钥使能基本安全功能，并通过公共签字链验证设计签字密钥。根密钥将密钥的 SHA-256 或 SHA-384 散列存储于 eFuses。可选择使能固件联合签字功能，以要求对器件上运行的配置固件版本进行签字。随后 FPGA 器件仅可加载已签字的固件。

⁽⁷⁾ 比特流验证（Bitstream authentication）仅适用于包含 AS（Advanced Security）订购代码的 Intel Stratix 10 器件和所有 Intel Stratix 10 DX 器件。

注意: 请参阅 *Intel Stratix 10 器件安全用户指南* 了解逐步生成一级签字密钥的说明。(8)

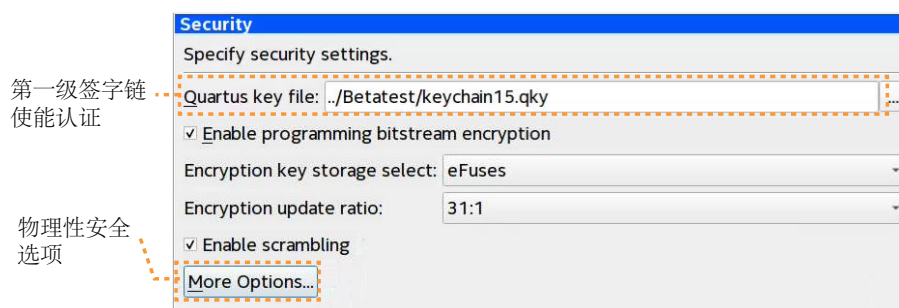
在 Assembler 中指定 .qky 后, Assembler 将一级签名链附到您生成的 .sof 生成文件。

使用 **Programming File Generator** 生成 .sof 文件的已签字配置比特流。JTAG Indirect Configuration File (.jic) 和 Raw Programming Data File (.rpd) 格式适用于 Active Serial (AS) 配置。Programmer Object File (.pof) 和 Raw Binary File (.rbf) 适用于 Avalon Streaming 配置。

按照以下步骤使能比特流验证:

1. 生成包含根密钥和 1 个或多个设计签字密钥的第一级签字链 (.qky), 如 *Intel Stratix 10 器件安全用户指南* 中说明。
2. 将一级签字链添加到配置比特流, 点击 **Assignments > Device > Device and Pin Options > Security**, 然后对 **Quartus key file** 选项指定第一级签字链 .qky。
3. 为使能更多物理器件安全选项, 请在 **Security** 页面点击 **More Options** 按钮。更多安全选项对话框 (第 45 页) 中介绍了所有选项。

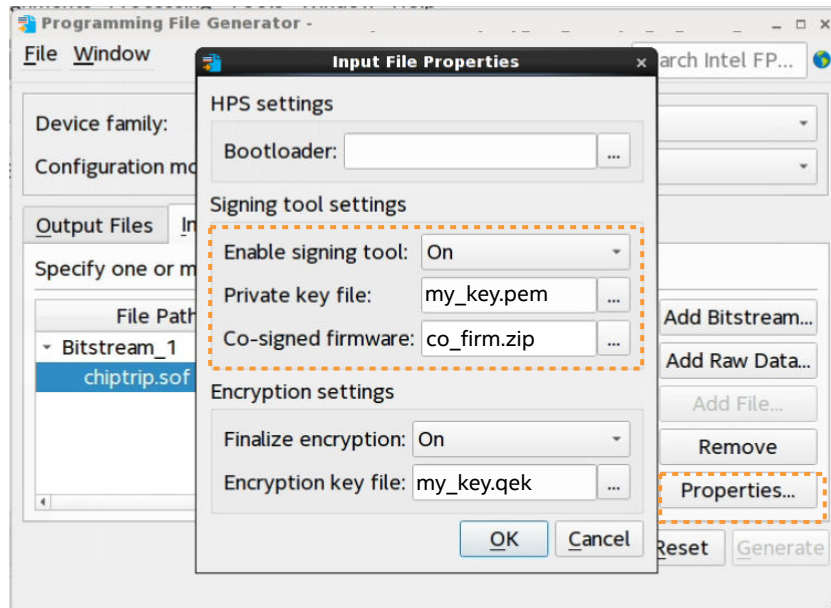
图 10. 安全选项卡 (器件和管脚选项)



4. 在 Assembler 中生成主要器件编程文件, 如 [生成器件主要编程文件](#) (第 5 页) 中所述。现在, 主要器件编程文件包含使能第一级验证的数据。
5. 要选择使能联合签字器件固件验证, 请使用如下选项生成 .jic 或 .rbf 辅助编程文件, 如 [生成辅助编程文件 \(Programming File Generator\)](#) (第 6 页) 中所述:
 - a. 在 **Programming File Generator** 中, 点击 **Properties** 按钮。出现 **Input File Properties** 对话框。

(8) Intel Agilex 器件尚无安全选项。

图 11. 使能联合签字器件固件验证 (Intel Stratix 10 器件)



- b. 将 **Enable signing tool** 设置为 **On**。
 - c. 对于 **Private key file**，指定一个设计签字密钥 Privacy Enhanced Mail Certificates 文件 (.pem) 用于固件联合签字。该密钥可从 FPGA 设计签字密钥中分离出来。
 - d. 对于 **Co-signed firmware**，请指定 Quartus Co-Signed Firmware 文件 (.zip)。
 - e. 点击 **OK**。
6. 使用 Programmer 通过 .jic 或 .rbf 配置器件。

相关链接

- [器件&管脚选项对话框](#) (第 38 页)
- [指定其他物理安全设置 \(Programming File Generator\)](#) (第 18 页)
- [Intel Stratix 10 器件安全用户指南](#)
了解有关生成器件安全密钥的详细信息。

1.3.2. 指定其他物理安全设置 (Programming File Generator)

Intel Stratix 10 器件可将安全和其他配置储存于 eFuses 中。可使能 eFuses 中的其他物理安全设置，以扩展器件安全保护级别。

指定其他物理器件安全设置，请按如下步骤：

1. 点击 **Assignments > Device > Device and Pin Options > Security**。
2. 在 **Security** 选项卡上，指定包含根密钥和 1 个或多个设计签字密钥的 First Level Signature Chain .qky 文件，以进行 **Quartus key file** 设置。
3. 点击 **More Options** 按钮，指定如下任意一项

图 12. 更多安全选项对话框

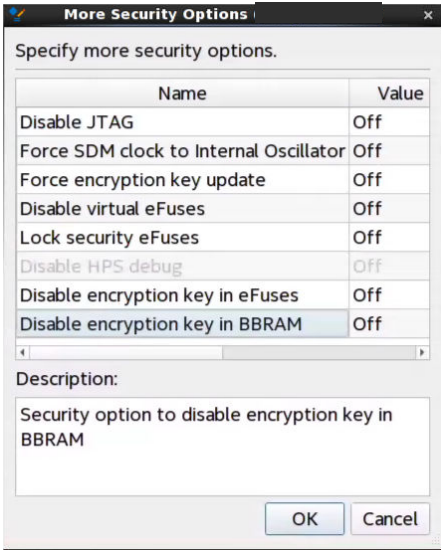


表 8. 更多安全选项对话框设置

选项	说明	值
Disable JTAG	禁用 JTAG 命令和器件配置。设置该选项将消除 JTAG 用于入侵模式，但同时消除了边界扫描功能。	<ul style="list-style-type: none">• Off—无效• On—开启，直到擦除包含的设计• On sticky—有效，直到下个 POR。• On check—检查相应的熔断熔丝
Force SDM clock to internal oscillator	禁用 SDM 的外部时钟源。SDM 必须使用内部振荡器 (oscillator)。使用内部振荡器比允许使用外部时钟源进行配置更为安全。	
Force encryption key update	指定加密密钥必须由您在 Encryption update ratio 选项中指定的频率进行更新。默认配比为 31:1。最多支持 20 个中间密钥加密。	
Disable virtual eFuses	禁用 eFuse 虚拟编程功能。	
Lock security eFuses	如果 eFuse CRC 不符合计算值，导致 eFuse 失败。	
Disable HPS debug	禁用通过 JTAG 接口访问 HPS 而进行的调试。	
Disable encryption key in eFuses	指定不可使用 eFuse 中存储的 AES 密钥的器件。然而可通过 BBRAM 中存储的 AES 密钥提供更高级别的安全性。	
Disable encryption key in BBRAM	指定不可使用 BBRAM 中存储的 AES 密钥的器件。然而可通过使用 eFuses 中存储的 AES 密钥提供更高级别的安全性。	

4. 点击 **OK**。

相关链接

使能 [Bitstream Authentication \(Programming File Generator\)](#) (第 16 页)

1.3.3. 使能比特流加密 (Programming File Generator)

要使能比特流加密，必须首先生成第一级签名链 (.qky) 使能 GUI 中的加密选项。随后，在 Assembler 中生成加密的配置比特流。最后，生成辅助编程文件，指定比特流解密的 AES Encryption Key 文件 (.qek)。

请按如下步骤使能比特流加密：

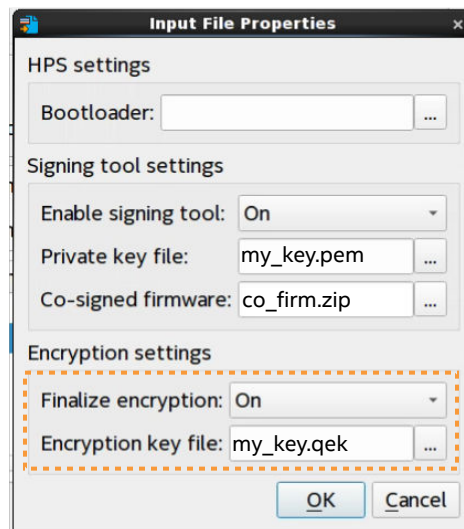
1. 生成包含根密钥和 1 个或多个设计签字密钥的 First Level Signature Chain，如 *Intel Stratix 10 器件安全用户指南* 中说明。
2. 点击 **Assignments > Device > Device and Pin Options > Security**。
3. 对于 **Quartus key file** 设置，指定包含根密钥和 1 个或多个设计签字密钥的第一级签字链 .qky。
4. 开启 **Enable programming bitstream encryption**，并指定如下一个或多个选项：

表 9. Assembler 加密安全设置

选项	说明
Encryption key storage select	指定存储 .qek 密钥文件的位置。可选择使用 Battery Backup RAM 或 eFuses 存储。
Encryption update ratio	指定配置比特与比特流解密需要的密钥更新次数的比率。可选择 31:1 （该密钥每 31 位变更 1 次）或 Disabled （无需更新）。最多支持 20 个中间密钥加密。
Enable scrambling	加扰配置比特流。
More Options	打开 More Security Options 对话框，指定其他物理安全选项。

5. 在 Assembler 中生成主要器件配置文件，如 [生成器件主要编程文件](#)（第 5 页）中所述。
6. 生成 .jic 或 .rbf 辅助编程文件，如 [生成辅助编程文件（Programming File Generator）](#)（第 6 页）中所述：
 - a. **Programming File Generator** 中，在 **Input Files** 选项卡上选择 .sof 文件。
 - b. 单击 **Properties** 按钮。弹出 **Input File Properties** 对话框。

图 13. 输入文件属性



- c. 将 **Finalize encryption** 设置为 **On**。
 - d. 器件配置之前，请指定对 SDM 中比特流解密的 AES 256-bit 或 384-bit **Encryption** 密钥文件 (.qek)。
7. 点击 **OK**。

相关链接

输入文件选项卡设置 (Programming File Generator) (第 46 页)

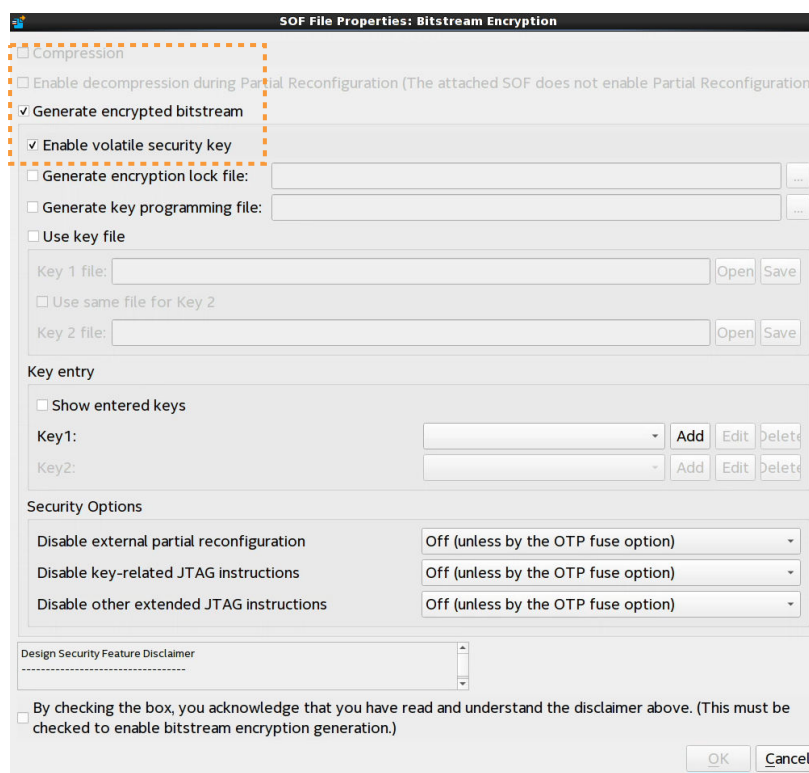
1.4. 使能比特流加密或针对 Intel Arria 10 和 Intel Cyclone 10 GX 器件的压缩

可选择使能比特流文件加密，但需要用户定义的 256-bit 安全密钥来访问配置比特流。或者，使能比特流压缩减小编程文件的大小，最大程度减少传输和存储要求。通过压缩可将配置文件缩小 30% 至 55%（具体取决于设计）。Intel Arria 10 和 Intel Cyclone 10 GX 器件中文件压缩和加密选项为互斥访问。

按照如下步骤使能 Intel Arria 10 和 Intel Cyclone 10 GX 器件的比特流文件压缩或加密：

1. 如本文档中说明生成 .jic 文件，进行 Flash 编程。
2. **Convert Programming File** 对话框中，在 **Input files to convert** 下选择 .sof 文件。
3. 单击 **Properties** 按钮。弹出 **SOF File Properties: Bitstream Encryption** 对话框。

图 14. 使能比特流压缩或加密(Intel Arria 10 和 Intel Cyclone 10 GX 设计)



4. 使能压缩，开启 **Compression** 选项。所有加密选项都禁用，由于压缩和加密选项互斥。
5. 使能比特流文件加密：
 - a. 关闭 **Compression** 选项。
 - b. 开启 **Generate encrypted bitstream** 选项。

- c. 指定编程文件密钥解密选项，和 **Security Options**，如[压缩和加密设置 \(Convert Programming File\)](#) (第 48 页)中所述。
6. 点击 **OK**。

相关链接

- [Intel Arria 10 内核架构和通用 I/O 手册](#)
了解有关器件安全性配置步骤的详细信息。
- [Intel Cyclone 10 GX 内核架构和通用 I/O 手册](#)
了解有关器件安全性配置步骤的详细信息。

1.5. 生成用于局部重配置的编程文件

以下部分介绍局部重配置的比特流和其他文件的生成。

1.5.1. 生成 PR 比特流文件

对于 Intel Stratix 10 设计，**Assembler** 在编译结束时自动生成配置 .rbf。对于 Intel Arria 10 和 Intel Cyclone 10 GX 设计，请使用如下方法处理 PR 比特流并生成用于重配置的 Raw Binary File (.rbf) 文件。

编译期间生成 PR 比特流

编译过程中生成 .rbf 文件的步骤如下：

1. 将以下约束添加到 .qsf 修订版中以在编译后自动生成需要的 PR 比特流：

```
set_global_assignment -name GENERATE_PR_RBF_FILE ON set_global_assignment -  
name ON_CHIP_BITSTREAM_DECOMPRESSION OFF
```

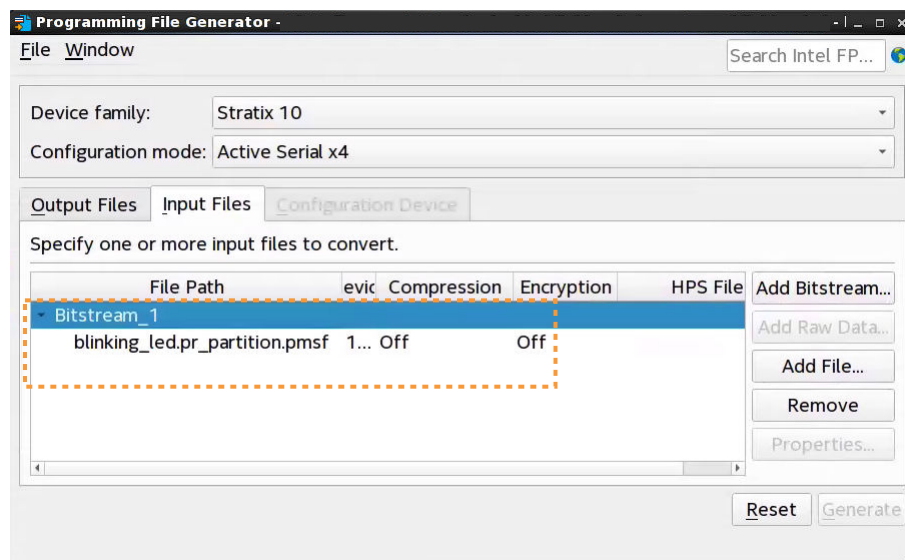
2. 要编译修订内容并生成 .rbf，可点击 **Processing > Start Compilation**。

使用 Programming File Generator 生成 PR 比特流

按照如下步骤使用 **Programming File Generator** 生成用于 PR 编程的 .rbf 文件。

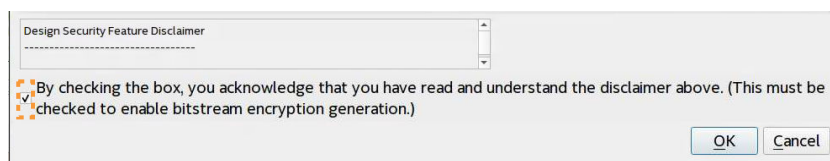
1. 点击 **File > Programming File Generator**。**Programming File Generator** 弹出。
2. 指定进行局部重新配置的目标 **Device family** 和 **Configuration mode**。
3. 在 **Output File** 选项卡上，指定 **Output directory**，文件 **name**，并启用 **Raw Binary File for Partial Reconfiguration (.rbf)** 文件类型。
4. 为添加输入 .pmsf 文件进行转换，可点击 **Input Files** 选项卡，点选 **Add Bitstream**，并指定在 **Assembler** 中生成的 .pmsf。

图 15. 添加比特流文件



5. **Input Files** 选项卡上，选择比特流 .pmsf 文件，并点击 **Properties**。为 .rbf 指定如下选项：
 - **Enable compression**—生成压缩的 PR 比特流文件将文件缩小。
 - **Enable encryption**—为基础映像和 PR 映像生成加密的独立比特流。即使基础映像无加密，也可为 PR 映像加密。PR 映像可具有单独的加密密钥文件（.ekp），也可指定其他 **Security settings**。
 - 如果开启 **Enable encryption**，则必须通过选中复选框确认 **Design Security Feature Disclaimer**（设计安全功能免责声明）。

图 16. Design Security Feature Disclaimer



6. 点击 **OK**。
7. 在 **Programming File Generator** 中，点击 **Generate**。PR 比特流文件将按您的规定生成。

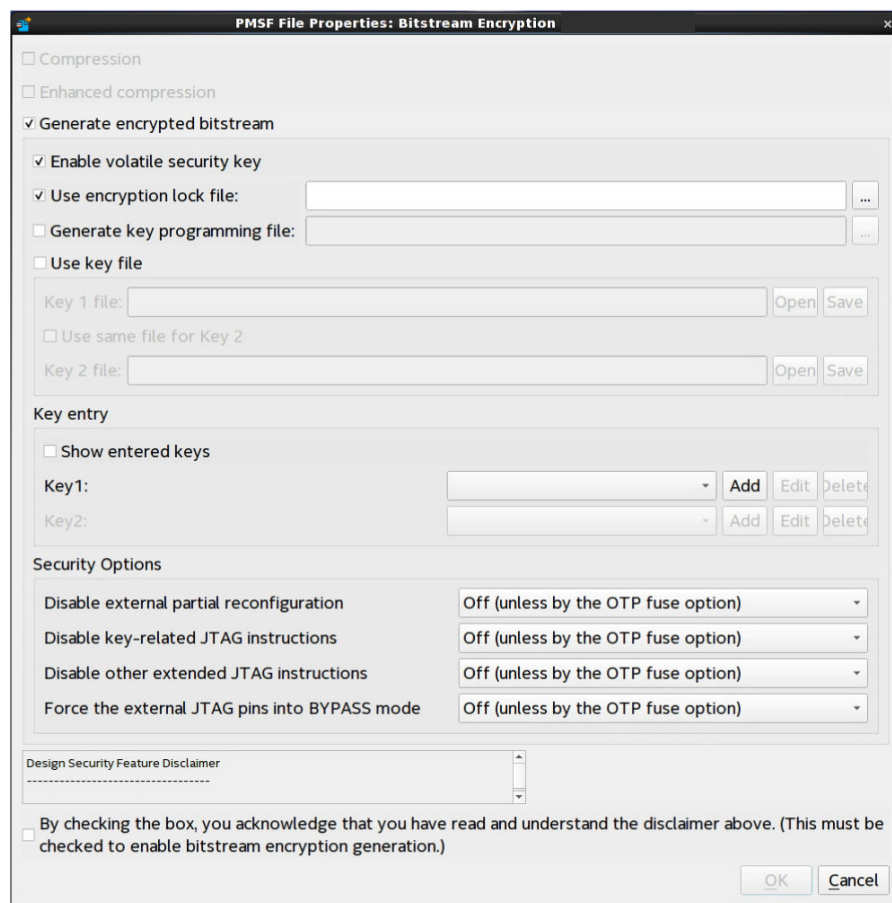
使用 **Convert Programming Files** 对话框生成 PR 比特流

按照下列步骤，通过 **Convert Programming Files** 对话框生成 .rbf：

1. 点击 **File > Convert Programming Files**。**Convert Programming Files** 对话框弹出。
2. 指定输出文件名称，并将 **Programming file type** 指定为 **Raw Binary File for Partial Reconfiguration (.rbf)**。
3. 添加输入 .pmsf 文件进行转化，点击 **Add File**。

4. 选择新添加的 .pmsf 文件，并点击 **Properties**。

图 17. PMSF 文件属性比特流加密

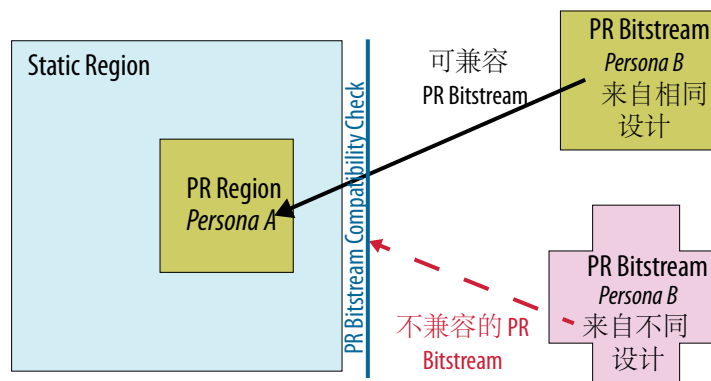


5. 使能或禁用以下选项并点击 **OK**:
 - **Compression**—使能 PR 比特流压缩。
 - **Enhanced compression**—使能 PR 比特流增强压缩
 - **Generate encrypted bitstream**—对基础映像和 PR 映像生成加密独立比特流。可以加密 PR 映像，即使您的基础映像并未加密。PR 映像可具有单独的加密密钥文件 (.ekp)。如果使能 **Generate encrypted bitstream**，就可使能或禁用 **Enable volatile security key**，**Use encryption lock file** 和 **Generate key programming file** 选项。
6. 点击 **Generate**。PR 比特流文件按照您的规定生成。

1.5.2. 部分重新配置比特流兼容性检查

部分重新配置比特流兼容性检查可验证重新配置的比特流兼容性，以避免配置不兼容的 PR 比特流。

图 18. PR 比特流兼容性检查



如下部分说明 PR 比特流兼容性检查支持。

Intel Stratix 10 和 Intel Agilex PR 比特流兼容性检查

Intel Stratix 10 和 Intel Agilex 设计中，PR 比特流兼容性检查已在 Compiler 和 Secure Device Manager (SDM) 固件中默认开启。如果开启了 PR 比特流兼容性检查，则如下限制适用于 PR 设计：

- Compiler 允许的 PR 区域最多达到 255 个。
- Compiler 允许父 PR 区域最多具有 15 个子 PR 区域。
- Compiler 允许的分层局部重配置层次最多达到 6 个。

PR 比特流兼容性检查启用后，如果 PR 设计超出上述限制，则 Compiler 会生成错误。

如果所需 PR 区域超出上述限制，或者希望禁用 PR 比特流兼容性检查，则可将以下约束添加到 .qsf 文件：

```
set_global_assignment -name ENABLE_PR_POF_ID OFF
```

Intel Arria 10 和 Intel Cyclone 10 GX PR 比特流兼容性检查

Intel Arria 10 和 Intel Cyclone 10 GX 设计中，可在例化 IP Catalog 中的 Partial Reconfiguration Controller Intel Arria 10 /Cyclone 10 FPGA IP 时，通过开启 **Enable bitstream compatibility check** 选项来使能或禁用 PR 比特流兼容性检查。

然后，软件验证局部重新配置 PR Bitstream 文件 (.rbf)。如果软件检测到不仅安装的特流，则 PR 操作将停止，且 status 输出报错。PR .pof ID 编码为 PR 比特流的第 71 个字。

开启 **Enable bitstream compatibility check** 时，PR Controller IP 核创建 **PR bitstream ID** 并在配置对话框中显示该比特流 ID。

1.5.3. 原始二进制编程文件字节序列传输实例

原始二进制编程文件 (.rbf) 包含小端 (little-endian) 原始二进制格式的器件配置数据。以下实例显示 .rbf 字节序列 02 1B EE 01 以 x32 模式发送：

表 10. x32 模式下写入 PR 控制模块或 SDM

x32 模式下，文件中的首个字节是配置双字的最低有效字节，而第 4 个字节为最高有效字节。

双字= 01EE1B02			
LSB: BYTE0 = 02	BYTE1	BYTE2 = EE	MSB: BYTE3 = 01
D[7..0]	D[15..8]	D[23..16]	D[31..24]
0000 0010	0001 1011	1110 1110	0000 0001

1.5.4. 生成由多个.pmsf 文件合并的.pmsf 文件

使用单个 .rbf 文件同时重新配置 2 个 PR 区域。合并 2 个或多个 .pmsf 文件：

1. 开启 **Convert Programming Files** 对话框。
2. 指定输出文件名称，并将编程文件类型指定为 **Merged Partial-Mask SRAM Object File (.pmsf)**。
3. 在 **Input files to convert** 对话框中，选择 **PMSF Data**。
4. 添加输入文件，请点击 **Add File**。必须指定 2 个或多个文件进行合并。
5. 生成合并文件，点击 **Generate**。

或者，要合并来自 Intel Quartus Prime shell 的 2 个或多个 .pmsf 文件，可键入如下命令：

```
quartus_cpf --merge_pmsf=<number of merged files> <pmsf_input_file_1> \
<pmsf_input_file_2> <pmsf_input_file_etc> <pmsf_output_file>
```

例如，合并 2 个 .pmsf 文件，键入如下命令：

```
quartus_cpf --merge_pmsf=<2> <pmsf_input_file_1> <pmsf_input_file_2> \
<pmsf_output_file>
```

1.6. 脚本支持

Intel Quartus Prime 软件允许从命令行生成编程文件。可将这些命令合并于脚本化流程中。

1.6.1. quartus_pfg 命令行工具

Programing File Generator 也可作为 quartus_pfg 可执行文件。可在命令行中指定转换设置或通过 FPFG 设置文件 (.pfg) 进行设置。该功能对于需要多个图像或多用户数据文件 (HEX/RBF) 的高级设计很有用，因为只要在 GUI 中定义了设置，就可导出以供命令行中后续使用。

将 PFG 设置导入 .pfg 文件，点击 **File > Save**。Programming File Generator 仅保存持续不变的设置。

有关 quartus_pfg 可执行文件的更多信息，请在命令行中键入以下内容：

```
quartus_pfg --help
```

GUI 和 Command Line Tool 之间的差异

命令行工具仅支持单图像转换。

1.6.2. quartus_cpf 命令行工具

Convert Programming Files 工具也可作为 quartus_cpf 命令行可执行文件使用。可在命令行中指定转换设置或通过转换设置文件(.cof)进行转换。

要发挥 quartus_pfg 可执行文件的作用，请在命令行中键入以下内容：

```
quartus_pfg --help
```

1.6.2.1. 使用 Mask Settings File 和 SRAM Object File 生成 Partial-Mask SRAM Object File

- 要生成带有 quartus_cpf 可执行文件的.pmsf 文件，请键入以下命令行：

```
quartus_cpf -p <pr_revision.msf> <pr_revision.sof> <new_filename.pmsf>
```

注意：

The -p option is available for designs targeting Intel Arria 10 和 Intel Cyclone 10 GX device families.

相关链接

[Intel Quartus Prime Pro Edition 用户指南：局部重新配置](#)

In *Intel Quartus Prime Pro Edition 用户指南：局部重新配置*

1.7. 生成编程文件修订历史

文档版本	Intel Quartus Prime 版本	修订内容
2019.09.30	19.3.0	<ul style="list-style-type: none"> 添加了“使能 Intel Stratix 10 器件比特流安全性”的新主题。 添加了“使能 Bitstream Authentication (Programming File Generator)”的新主题。 添加了“指定其他物理安全设置(Programming File Generator)”的新主题。 添加了“使能 Bitstream Encryption(Programming File Generator)”的新主题。 将“Authentication and Encryption”选项卡名称更新为“Security”选项卡。 添加有关 Intel Agilex 器件编程文件支持的脚注。 介绍了新对话框 More Security Settings 对话框。
2019.06.10	19.1.0	<ul style="list-style-type: none"> 为 <i>通用 Flash Programmer 用户指南</i> 添加了链接。 为“生成辅助编程文件”添加 Flash 编程详情，并为 Programming File Generator 和 Convert Programming Files 对话框创建了单独的主题。 添加了“使能比特流加密或联合签字(Programming File Generator)”的新主题。 添加了“使能比特流压缩或加密(Convert Programming File)”的新主题。 截屏更新为最新的 GUI
2019.04.01	19.1.0	<ul style="list-style-type: none"> 重列标题并重组主题以改善信息流。 添加了“Programming File Generator 配置模式”主题 添加了“转换编程文件配置模式”主题 添加“生成用于局部重配置的编程文件”内容。 添加了“生成 PR 比特流文件”。 添加了“局部重新配置比特流兼容性检查”。 添加了“原始二进制编程文件字节序列传输实例”。 添加了“生成由多个.pmsf 文件合并的.pmsf 文件”。
继续...		

文档版本	Intel Quartus Prime 版本	修订内容
2018.10.09	18.1.0	<ul style="list-style-type: none"> 将 MAX V 添加到 Programming File Generator 工具支持的器件列表中。 添加表格: <i>Convert Programming Files</i> 工具支持的器件系列。
2018.09.24	18.1.0	<ul style="list-style-type: none"> 添加主题: <i>quartus_cpf</i> 命令行工具。 阐明 Convert Programming Files 对话框为 legacy 工具, 支持旧版器件系列的文件转换。 主题: <i>输出文件类型</i>中, 指定包含文件类型的列表由 Converting Programming Files 工具生成。
2018.08.07	18.0.0	将文档标题还原为 <i>Programmer</i> 工具用户指南: <i>Intel Quartus Prime Pro Edition</i> 。
2018.06.27	18.0.0	<ul style="list-style-type: none"> 按照 <i>编程器件</i>章节中的信息创建新章节。 包含了有关 Programming File Generator 工具的信息。

2. 使用 Intel Quartus Prime Programmer

使用 Intel Quartus Prime Programmer（下载配置工具）和所连接的通信电缆对 Intel CPLD，FPGA 和配置器件进行编程或配置。本章节说明如何建立和使用 Intel Quartus Prime Programmer。

2.1. Intel Quartus Prime Programmer

在 Intel Quartus Prime 软件中，点击 **Tools > Programmer** 访问完整 Programmer。

图 19. Intel Quartus Prime Programmer



进行编程和配置之前，需要先生成并指定主要编程文件，设置编程硬件，并在 Programmer 中设置配置模式。

2.2. 编程和配置模式

当前版本的 Intel Quartus Prime Programmer 支持 Programmer **Mode** 列表中的编程和配置模式，具体如下。选择一个配置模式以建立并运行该类型。

表 11. 编程和配置模式

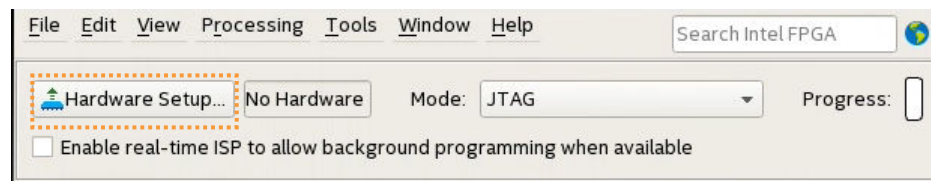
编程或配置模式	说明
JTAG	通过连接 Test Action Group (JTAG) Boundary-Scan Test (BST) 电路来配置 1 个或多个器件的配置方法。
In-Socket Programming	通过 Altera Programming Unit (APU) 的配置器件编程和测试。
Passive Serial	外部控制器通过串行数据流将配置数据传递到 1 个或多个配置器件。该器件作为通过 5 线接口连接外部控制器的从器件使用。外部控制器可为智能主机，例如微控制器或 CPU，或 Intel Quartus Prime Programmer。外部控制器还可作为串行配置器件。
Active Serial Programming	有效串行存储器接口块将设计数据加载到 1 个或多个器件。有效串行存储器块控制配置进程，并使用 EPCS1, EPCS4, EPCS16, EPCS64, EPCQ, EPCQL, 和第三方 QSPI 串行配置器件中存储的配置数据对链中器件进行配置。

2.3. 器件配置基本步骤

通过 JTAG 的基本 FPGA 器件配置需要开启 Intel Quartus Prime Programmer，连接开发套件或开发板上的器件，并将配置 SRAM Object File (.sof) 加载到 FPGA 的 SRAM 中。以下步骤说明基本 JTAG 器件配置流程：

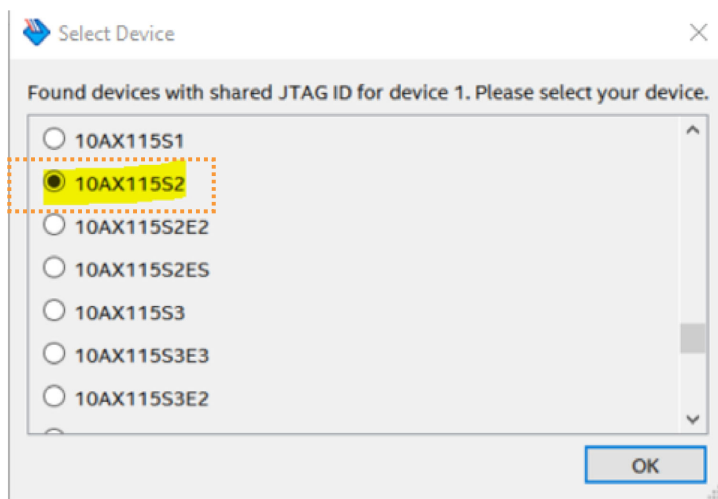
1. 运行 Assembler 生成主要编程文件，点击 **Processing > Start > Start Assembler**。Compiler 将运行必要阶段并按照您的规定生成编程文件，如[生成器件主要编程文件](#) (第 5 页) 中说明。
2. 要开启 Programmer，请点击 **Tools > Programmer**。
3. 连接电路板线缆。对于 JTAG 器件配置，将 JTAG USB 线缆连接到电路板，并将电路板的电源线连接电源。
4. 接通板上电源。
5. 在 Programmer 中，选择 **JTAG** 作为 **Mode**，如[编程和配置模式](#) (第 30 页) 中说明。
6. 点击 **Hardware Setup**。在 **Hardware** 列表中，选择已连接的编程硬件，如[指定编程硬件设置](#) (第 32 页) 中说明。

图 20. 硬件设置



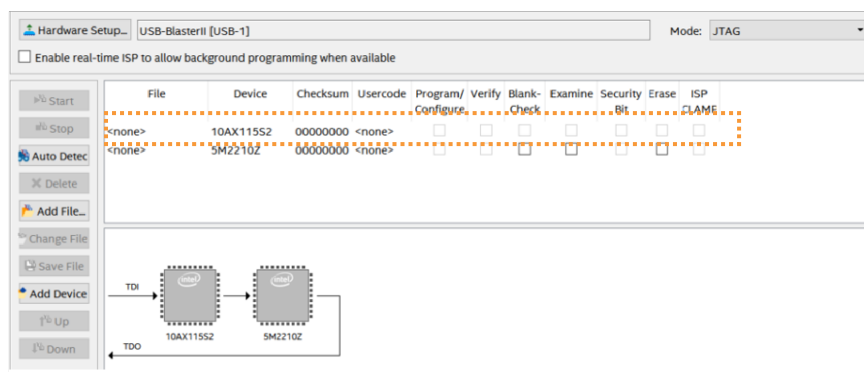
7. 在 **Found Devices** 列表中，选择符合您设计的器件，并点击 **OK**。

图 21. 选择器件



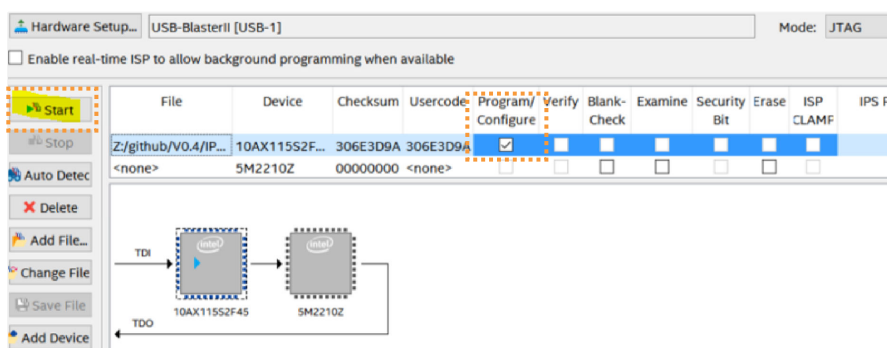
8. 右键单击文件列表中的行，然后点击 **Change File**。

图 22. Programmer 窗口



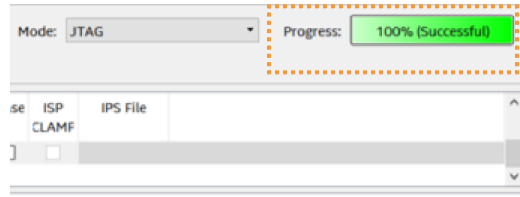
9. 浏览并选择 .sof 文件。
10. 对该行启用 **Program/Configure** 选项。

图 23. 编程/配置选



11. 点击 **Start**。器件配置完成后进度条达到 100%。该器件现已完成配置并在运行中。

图 24. 编程成功



注意: 如果器件编程失败, 请确认您选择用于配置的器件与指定用于 .sof 文件生成中的器件相吻合。

2.4. 指定编程硬件设置

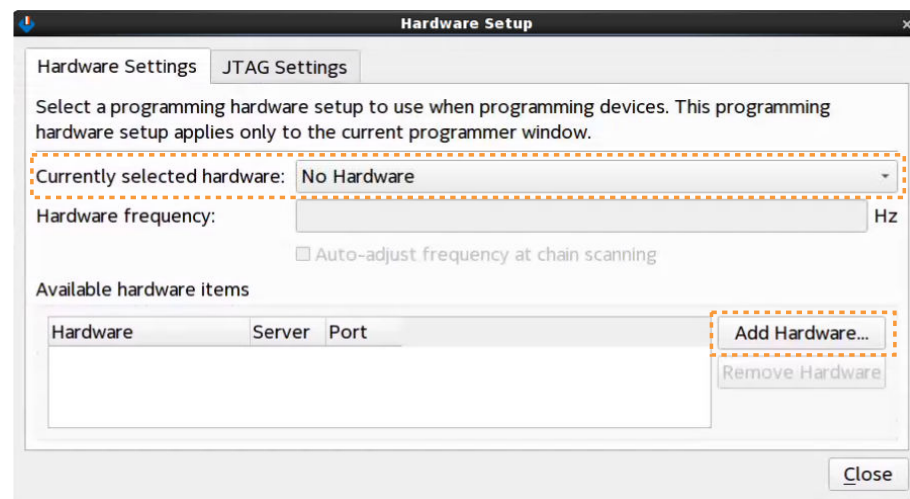
在能够编程或配置器件之前, 必须指定适当的硬件设置。Programmer 的 **Hardware Setup** 对话框支持从当前编程设置添加和删除编程硬件或 JTAG 服务器。可为器件编程或配置指定硬件设置, 或配置局部 JTAG 服务器。

JTAG 服务器允许 Intel Quartus Prime Programmer 通过远程计算机的 JTAG 服务器访问与该计算机相连接的 JTAG 编程硬件。JTAG 服务器支持通过远程位置的其他计算机来控制单台计算机中器件的编程和配置。JTAG 服务器使用 TCP/IP 通信协议。

选择器件编程硬件

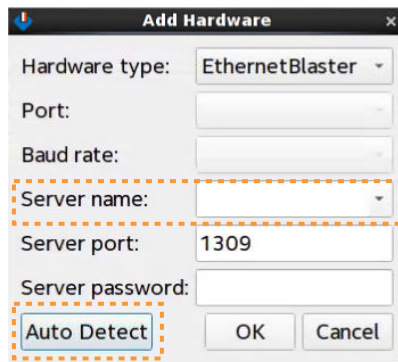
按如下步骤在 Programmer 中选择器件编程硬件:

1. 在 Programmer 中, 单击 **Hardware Setup** (硬件设置)。

图 25. **Hardware Setup** 对话框

2. 要添加新的编程硬件, 在 **Hardware Settings** 选项卡上单击 **Add Hardware**。Add Hardware 对话框中, 单击 **Auto Detect** 可删除您的编程硬件, 或指定编程硬件的属性。

图 26. 添加新硬件



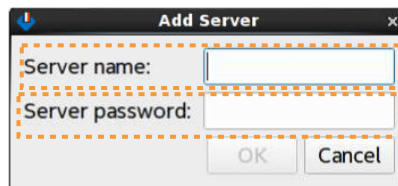
3. **Hardware Settings** 选项卡上，在 **Currently selected hardware** 中选择已连接的编程硬件。在连接并将编程硬件添加到系统之前，该列表为空。
4. 使能或禁用 **Auto-adjust frequency at chain scanning**，以根据链扫描时的频率自动调整 **Hardware frequency**。
5. 单击 **Close**。该设置显示为当前硬件设置。

选择器件编程的 JTAG 服务器

按以下步骤在 Programmer 中选择用于器件编程的 JTAG 服务器：

1. 在 Programmer 中，单击 **Hardware Setup**。
2. 在 **JTAG Settings** 选项卡上，单击 **Add Server**。 **JTAG Settings** 对话框中，指定 **Server name** 和 **Server password**。

图 27. JTAG 设置



3. **JTAG Servers** 下，选择需要访问以进行编程的 JTAG 服务器。
4. 单击 **Close**。该设置显示为当前硬件设置。

2.4.1. JTAG Chain Debugger 工具

JTAG Chain Debugger 工具允许测试 JTAG 链的完整性并检测 JTAG 链的间歇性故障。在 Intel Quartus Prime 软件中单击 **Tools > JTAG Chain Debugger** 访问该工具。

此外，该工具允许通过 JTAG 接口移入 JTAG 指令和数据，并逐步通过测试访问端口（TAP）控制器状态机进行调试。

2.4.2. 编辑未知器件的详细信息

当 Intel Quartus Prime Programmer 自动检测到具有共享 JTAG ID 的器件时，Programmer 提示指定的器件。如果 Programmer 未提示指定器件，就必须手动将 JTAG 链中的每个器件添加到 Programmer，并定义每个器件的指令寄存器长度。

请按如下步骤编辑未知器件的详细信息：

1. 双击器件栏中列出的未知器件。
2. 单击 **Edit**。
3. 更改器件 **Name**。
4. 指定 **Instruction register length**。
5. 单击 **OK**。
6. 保存 .cdf 文件。

2.4.3. 在 Linux 上运行 JTAG Daemon

JTAGD daemon（守护程序）是 Linux 版的 JTAG 服务器。JTAGD daemon 允许远程机器通过网络对与 Linux 主机连接的电路板进行编程或配置。JTAGD daemon 还允许程序共享 JTAG 资源。

运行 JTAGD daemon 可避免：

- JTAGD 服务器在空闲两分钟后退出。
- JTAGD 服务器不接受来自远程机器的连接，从而导致间歇性故障。

将 JTAGD 作为 daemon（守护程序）运行，需要：

1. 创建 /etc/jtagd 目录。
2. 设置关于该目录和目录中文件的权限，以允许进行读/写访问。
3. 从 quartus/bin 目录执行 jtagd（无自变量）。

JTAGD 守护程序正在运行中，直到您退出系统（log off）时才会终止。

2.5. 使用 Flash Loader 编程

并行和串行配置器件不支持 JTAG 编程接口。但可使用 Flash 加载程序（loader）通过 JTAG 接口对器件进行在系统编程配置。Flash 加载程序允许 FPGA 作为 JTAG 接口和配置器件间的桥接运行。Intel Quartus Prime 软件支持各种并行和串行 Flash 加载程序进行编程比特流存储和通过 Flash 器件进行配置。

请参阅以下文档，了解 flash 编程说明。

相关链接

- [通用串行 Flash 接口 Intel FPGA IP 核用户指南](#)
- [Intel 并行 Flash Loader IP 核用户指南](#)
- [通用 Flash Programmer 用户指南](#)
- [可定制 Flash Programmer 用户指南](#)

2.5.1. 指定 Flash 分区

Flash 分区允许存储比特流或原始数据。

注意: **Programming File Generator** 仅支持为 .jic 或 .pof 编程文件定义 Flash 分区。

在 **Configuration Devices** 选项卡中创建 Flash 分区:

1. 选择器件, 并点击 **Add Partition**。
2. **Add Partition** 对话框中, 定义如下参数, 并点击 **OK**:

表 12. 添加分区对话框设置

设置	说明
Name	您给定的分区名称。
Input file	编程到 Flash 分区中的输入文件。
Page	配置器件可将多个配置比特流存储于 flash 存储器 (称作, 页面) 中。CFI 配置器件最多可存储 8 个配置比特流。Intel Hyperflex™ 器件可最多存储 4 个配置比特流, 包括出厂映像。 Intel Hyperflex 器件中, 在启用远程系统更新功能时, Page 代表奇偶性。
Address Mode	选项包括: <ul style="list-style-type: none">• Auto—在 Flash 器件中自动分配一个块以存储数据。• Block—指定 Flash 分区的起始和结束地址。• Start—指定分区的起始地址。该工具基于输入数据的大小分配分区结束地址。
Start address	指定分区的起始地址。仅当 Address Mode 为 Block 或 Start 时启用。
End address	指定分区的结束地址。仅当 Address Mode 为 Block 时启用。

与器件关联的分区会出现在器件列表中。

3. 如果需要更改分区参数, 请点击该分区, 然后点击 **Edit**。
4. 如果需要删除分区, 请点击该分区, 然后点击 **Remove**。
5. 指定所有 Flash 分区设置后, 点击 **Generate**。

2.5.2. 完全擦除 Flash 存储器扇区

通过 JTAG 和 .jic 文件执行 flash 存储器擦除操作时, Intel Quartus Prime Programmer 仅擦除 .jic 指定的 flash 存储器扇区。

例如, 如果指定 .jic 文件仅包含 EPCQ64A 器件中 13.6Mbits FPGA 映像, 则 Programmer 仅擦除底部 13.6Mbits, 而不会擦除其余 50.4Mbits 数据。

要擦除整个 flash 存储器件的内容, 就请勿为 flash 编程指定 .jic 文件。或者将 flash 器件手动添加到相关 FPGA 器件链, 具体步骤如下:

1. Programmer 中, 右键点击目标 FPGA 器件, 然后单击 **Edit > Attach Flash Device**。
2. 从列表中选择合适的 flash 器件。Factory Default Serial Flash Loader 自动为 FPGA 加载。
3. Programmer 中, 使能 **Erase** 复选框, 并点击 **Start** 以开启擦除操作。

2.6. 使用 Project Hash 验证编程文件源

Intel Quartus Prime 编程文件支持工程散列 (hash) 属性, 该属性可识别生成编程文件的源工程。

编译器件, Intel Quartus Prime 软件生成唯一工程 hash, 并将此散列值嵌入编程文件 (.sof)。可通过匹配工程和编程文件的散列值来验证编程文件的来源。

该工程 hash 不因 Intel Quartus Prime 软件的不同或安装软件的更新版而改变。但是, 如果使用其他版本或补丁更新任何 IP, 则工程 hash 会发生改变。

2.6.1. 获得 Intel Arria 10 器件的 Project Hash

为获得设计的 .sof 编程文件的工程散列值, 请通过 --project_hash 选项, 使用 quartus_asm 命令行可执行文件 (Windows 中的 quartus_asm.exe)

```
quartus_asm --project_hash <sof-file>
```

实例-1: Project Hash 命令输出:

本实例中, 编程文件为 worm.sof。

```
Info: *****
Info: Running Quartus Prime Assembler
Info: Version 17.0.0 Build 288 04/12/2017 SJ Pro Edition
Info: Copyright (C) 2017 Intel Corporation. All rights reserved.
Info: Your use of Intel Corporation's design tools, logic functions
Info: and other software and tools, and its AMPP partner logic
Info: functions, and any output files from any of the foregoing
Info: (including device programming or simulation files), and any
Info: associated documentation or information are expressly subject
Info: to the terms and conditions of the Intel Program License
Info: Subscription Agreement, the Intel Quartus Prime License Agreement,
Info: the Intel MegaCore Function License Agreement, or other
Info: applicable license agreement, including, without limitation,
Info: that your use is for the sole purpose of programming logic
Info: devices manufactured by Intel and sold by Intel or its
Info: authorized distributors. Please refer to the applicable
Info: agreement for further details.
Info: Processing started: Fri Apr 14 18:01:47 2017
Info: Command: quartus_asm -t project_hash.tcl worm.sof
Info: Quartus(args): worm.sof
Info: 0x1ffdc3f47c57bbe0075f6d4cb2cb9deb
Info: (23030): Evaluation of Tcl script project_hash.tcl was successful
Info: Quartus Prime Assembler was successful. 0 errors, 0 warnings
Info: Peak virtual memory: 1451 megabytes
Info: Processing ended: Fri Apr 14 18:01:56 2017
Info: Elapsed time: 00:00:09
Info: Total CPU time (on all processors): 00:00:04
```

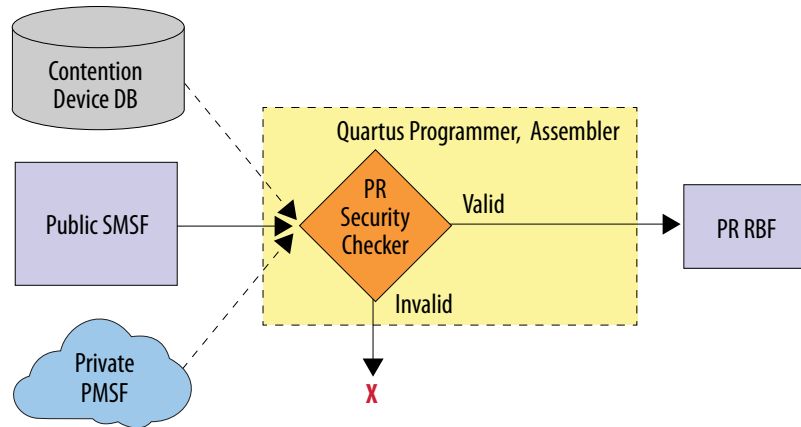
2.7. 使用 PR 比特流安全验证 (Intel Stratix 10 和 Intel Agilex 设计)

Pr 比特流安全验证需要单独的许可和 .qsfc 设置才能启用。在许可并使能 .qsfc 设置才能启用。在许可并使能 PR 比特流验证后, Compiler 在基础编译过程中为每个 PR 区域生成公开的 Secure Mask Settings File (.smsf) 和私密 Partially Masked Settings File (.pmsf)。

.pmsf 中包含 Programmer 生成 Client 区域 PR 比特流所需的全面信息，包括实际 bit 设置，区域掩码和所有辅助 bit 掩码。.smsf 包含区域所有权掩码和综合信息，以检测 PR 区域中用户实例 (persona) 的取数或存数 (peek 或 poke) 攻击。

此后，Programmer 需要私密 .pmsf 和公开 .smsf 来生成该 PR 区域的 PR 比特流，从而确保此 PR 用户实例仅可更改其拥有的 bit。Platform Owner 可能会也可能不会将 .smsf 文件发布给作为 PR 区域并行部分的第三方客户端。Platform Owner 使用 .smsf，通过 Programmer 从客户端 .pmsf 中为该 PR 区域生成 PR 比特流。

图 28. Programmer 中的 PR 比特流安全验证



按照如下步骤许可，使能，并使用 PR 比特流安全验证：

1. 获取许可证文件，以便在基础编译过程中为 PR 区域生成 .smsf 文件，并在 Programmer 中的 PR 比特流生成期间执行 PR 比特流安全验证。要获取许可，可先登录或注册 My-Intel 账号，然后提交 1 个 Intel Premier Support 案例请求许可证密钥。
2. 将许可证文件添加到 Intel Quartus Prime Pro Edition 软件，可点击 **Tools > License Setup** 并指定 **License File** 功能。
3. 要使能 PR 安全验证功能，可将如下行添加到工程 .qsf：

```
set_global_assignment -name PR_SECURITY_VALIDATION on
```

4. 编译基础版本
5. 基础编译后，查看 Assembler 报告检查针对每个 PR 区域的比特流生成而需要的 .smsf 文件生成情况。
6. Platform Owner 可将 .smsf 文件发布给作为 PR 区域并行部分的第三方客户端 (client)。Client 向 Platform Owner 提供私密 .pmsf 以验证 PR Persona (用户实例) 配置的 PR 安全性并生成经验证的 PR 比特流。
7. 为验证 Client 的 .pmsf PR 安全性，Platform Owner 在 Programmer 命令行指定 .smsf 和相应 .pmsf 文件以生成经验证的 PR 比特流：

```
quartus_cpf -c --smsf=<smsf_file> <pmsf_file> <output_file>
```

相关链接

- [Intel Quartus Prime Pro Edition 用户指南：局部重新配置](#)
In *Intel Quartus Prime Pro Edition 用户指南：局部重新配置*

- [My- Intel .com](#)

2.8. 独立运行的 Programmer

提供免费的 Stand-Alone Programmer（独立运行的下载编程工具），与 Intel Quartus Prime Programmer 具有相同的完整功能性。

Stand-Alone Programmer 有助于在无 Intel Quartus Prime 软件许可证的工作站上进行器件编程。Stand-Alone Programmer 不需要单独的 Intel Quartus Prime 软件许可证。可从 Intel 网站的 Download Center（下载中心）下载 Stand-Alone Programmer。

相关链接

[FPGA 下载中心](#)

2.8.1. Stand-Alone Programmer 存储器消耗

以下操作会增加 Stand-Alone Programmer 中的存储使用量：

- 自动检测
- 向 Flash 存储器添加编程文件
- Programmer 中手动连接 Flash

Windows 系统中，Stand-Alone Programmer 具有以下存储器限制：

表 13. 独立 Programmer 存储器限制

应用程序	最大 Flash 器件大小	使用 PFL 的 Flash 器件操作
64-bit 独立 Programmer	最高达到 2 Gb	多 Flash 器件

2.9. Programmer 设置参考

以下主题介绍影响编程和编程文件生成的 Intel Quartus Prime 设置。

2.9.1. 器件&管脚选项对话框

以下表格说明影响主要和辅助编程文件生成的 **Device & Pin Option** 设置。要访问该设置，请点击 **Assignments > Device > Device & Pin Options**。

表 14. 通用选项

允许指定独立于特定配置方案的基本器件配置选项。要访问这些设置，请点击 **Assignments > Device > Device and Pin Options > General**。

选项	说明
<p>选项</p> <p>注意: 不支持 Intel Agilex 或 Intel Stratix 10 器件。</p>	<ul style="list-style-type: none"> • Auto-restart configuration after error—如果出现数据错误，将自动重新启动配置进程。如果该选项已关闭，则必须在出现错误时从外部命令器件重新启动配置进程。该选项适用于被动串行和主动串行配置方案。 • Release clears before tri-states—先释放继寄存逻辑单元上的清除信号，然后才释放三态缓冲器上的输出使能重写 (override)。如果该选项关闭，则先释放输出使能信号，然后才释放清除重写。 • Enable user-supplied start-up clock (CLKUSR)—使用由用户提供，并位于 CLKUSR 管脚的时钟进行初始化。关闭后，需要外部电路为 Passive Serial 和 Passive Parallel Synchronous 配置方案中的该 DCLK 管脚提供初始化时钟；Passive Parallel Asynchronous 配置方案中，器件使用内部初始化时钟。 • Enable device-wide reset (DEV_CLRn)—使能 DEV_CLRn 管脚，允许外部源复位器件的所有寄存器。如果关闭该选项，则 DEV_CLRn 管脚可用作用户 I/O 管脚，但器件在用户模式下运行时为禁用。 • Enable device-wide output enable (DEV_OE)—器件处于用户模式时，使能 DEV_OE 管脚。如果该选项开启，片上所有输出均正常运行。该管脚禁用时，所有输出均为三态。如果该选项关闭，则 DEV_OE 管脚可用作用户 I/O 管脚，但器件运行于用户模式下运行时为禁用。 • Enable INIT_DONE output—使能 INIT_DONE 管脚，从而在完成初始化且器件处于用户模式时允许从外部监控。如果关闭该选项，则 INIT_DONE 管脚可用作用户 I/O 管脚，但器件运行于用户模式时为禁用。 • Enable JTAG Pin Sharing—使能 JTAG 管脚共享功能。JTAGEN 管脚使能后，成为用户模式中的专用输入管脚。JTAGEN 管脚拉低时，JTAG 管脚 (TDO, TCK, TDI 和 TMS) 管脚可用作测试管脚。JTAGEN 管脚为高电平有效时，JTAG 管脚用作专用管脚。如果关闭该选项，则 JTAGEN 管脚为可用作用户 I/O 管脚，但器件运行于用户模式时为禁用。JTAG 管脚保留为专用 JTAG 管脚。 • Enable nCONFIG, nStatus, and CONF_DONE pins—用户模式下使能主要配置管脚，nCONFIG, nSTATUS 和 CONF_DONE 管脚。如果关闭此选项，则 nCONFIG, nSTATUS 和 CONF_DONE 管脚可用作用户 I/O 管脚，但器件在用户模式下运行时为禁用。 • Enable OCT_DONE—使能 OCT_DONE 管脚，其控制 INIT_DONE 管脚是否由 OCT_DONE 管脚门控。如果关闭此选项，则 INIT_DONE 管脚不由 OCT_DONE 管脚门控。 • Enable security bit support—使能安全位支持，可阻止获取本器件中的数据并用于对其他器件编程。本选项使用于支持的器件 (MAX II 和 MAX V) 系列。 • Set unused TDS pins to GND—在管脚中将为使用的温度传感二极管 TSD 管脚，TEMPDIODEp 和 TEMPDIODEn 设置为 GND。默认情况下，TSD 管脚可用于连接外部温度传感器件；但在未连接时，必须手动将管脚连接到 GND。启用后，该选项将更新 .pin 文件中的信息，但不影响 FPGA 行为。 • Enable CONFIG_SEL pin—用户模式下使能 BOOT_SEL 管脚。如果关闭该选项，则 BOOT_SEL 管脚可用作用户 I/O 管脚，但在器件运行于用户模式时为禁用。 • Enable nCEO pin—使能 nCEO 管脚。当编程多个器件时，应将该管脚与后续器件的 nCE 连接。如果关闭该项，则 nCEO 管脚可用作用户 I/O 管脚，但器件在用户模式下运行时为禁用。 • Enable autonomous PCIe HIP mode—器件内核配置完成之前，先在外设配置完成后释放 PCIe HIP。该选项仅在 CvP 模式禁用时生效。 • Enable the HPS early release of HPS IO—IOCSR 编程完成后释放 HPS 共享 I/O 块。
Auto usercode	设置 JTAG 用户代码以匹配器件编程文件校验和值。非易失器件的编程文件是 .pof，而基于 SRAM 的器件使用 .sof。如果开启该选项，则 JTAG user code 选项不可用。
JTAG user code	为当前 Compiler 设置中选择的器件指定十六进制数。JTAG 用户代码时本选项寄存器的扩展。可使用 JTAG USERCODE 读取该数据。如果开启 Auto usercode ，则该选项不可用。
继续...	

选项	说明
In-system programming clamp state	允许在对使用的管脚进行“在系统编程”时，为没有“在系统编程”钳位状态约束的管脚指定可选取的状态。未使用管脚和专用管脚在“在系统编程”中必须始终保持三态。使用的管脚在“在系统编程”过程中默认为三态，因而将该器件与板上的其他器件电气隔离。但是有时为了避免系统损坏，可能需要为使用的管脚指定“在系统编程”器件的逻辑电平。可用设置如下： <ul style="list-style-type: none"> • Tri-state—管脚均为三态。 • High—管脚驱动 VCCIO。 • Low—管脚驱动 GND。 • Sample and Sustain—管脚驱动 SAMPLE/PRELOAD JTAG 指令期间捕获的电平。
配置时钟源	指定用于器件初始化的时钟源（CONF_DONE 信号变高和 INIT_DONE 信号变高之前的持续时间）。对于 AS x1 或 AS x4 配置模式，仅可选择 Internal Oscillator 或 CLKUSR 管脚。对于 AS 模式，DCLK 管脚是非法选项。14 nm 器件系列中，仅 Internal scillator 或 OSC_CLK_1 管脚可用。
Device initialization clock source	指定用于器件初始化的时钟源（CONF_DONE 信号变高和 INIT_DONE 信号变高之前的持续时间）。对于 AS x1 或 AS x4 配置模式，仅可选择 Internal Oscillator 或 CLKUSR 管脚。对于 AS 模式，DCLK 管脚是非法选项。14 nm 器件系列中，仅 Internal scillator 或 OSC_CLK_1 管脚可用。

表 15. 配置选项

允许使用您的编程比特流为后续器件配置指定配置方案，配置器件和管脚选项，串行时钟源，以及其他选项。访问这些设置，请点击 **Assignments > Device > Device and Pin Options > Configuration**。禁用了不适用于当前器件或配置模式的选项。

选项	说明
Configuration scheme	指定生成正确的主要和辅助编程文件的配置方案，例如 Active Serial x4 。仅适用于当前 Configuration Scheme 的选项可用。
Configuration Device	允许为储存和加载配置数据的外部配置器件指定选项。 <ul style="list-style-type: none"> • Configuration device I/O voltage—为目标器件当前配置方案的配置管脚指定 VCCIO 电压。该选项可用于支持的器件系列。 • Force VCCIO voltage to be compatible with configuration I/O voltage—强制配置管脚的 VCCIO 电压与配置器件 I/O 电压相同。如果关闭该选项，则配置管脚的 VCCIO 电压可能会依据包含配置管脚的 I/O bank 中使用的 I/O 标准而有所不同。此选项可用于支持的器件系列。
Configuration Pin Options	使能或禁用用于状态监控，SEU 错误检测，CvP 和其他配置管脚选项的特定器件配置管脚的操作。
Generate compressed bitstreams	生成压缩的比特流并使能目标器件中的比特流解压缩。
Active serial clock source	为 Active Serial 编程指定配置时钟源。选项范围为 12.5 MHz 到 100 MHz。
VID Operation Mode	在已选定操作模式的目标器件中使能 Voltage Identification 逻辑。可用选项为 PMBus Master 或 PMBus Slave 。
HPS/FPGA configuration order	对于硬核处理器系统（HPS）配置，请指定 HPS 和 FPGA 之间的配置顺序。可用选项为 HPS First ， After INIT_DONE 和 When requested by FPGA 。
HPS debug access port	<ul style="list-style-type: none"> • Disabled—HPS JTAG 未使能。 • HPS Pins—HPS JTAG 路由到 HPS 专用 I/O。 • SDM Pins—HPS JTAG 链接到 FPGA JTAG。
Disable Register Power-Up Initialization	指定 Assembler 是否通过寄存器上电初始化生成比特流。

表 16. 未使用的管脚选项

允许将器件上所有未使用管脚指定为保留状态。通过点击 **Assignments > Device > Device and Pin Options > Unused Pins** 访问。禁用了不适用于当前器件或配置模式的选项。

选项	说明
Reserve all unused pins	<ul style="list-style-type: none"> As input tri-stated—该管脚保留为三态输入管脚。 As output driving ground—该管脚保留为输出管脚并驱动接地信号。 As output an unspecified signal—该管脚保留为输出管脚并驱动各信号。 As input tri-stated with bus-hold circuitry—该管脚通过总线保持电路保留为三态输入管脚。 As input tri-stated with weak pull-up—该管脚通过弱上拉电阻器保留为三态输入管脚。

表 17. 两用管脚选项

允许指定已关联的两用管脚是否保留，以及保留目的。**Assignments > Device > Device and Pin Options > Unused Pins** 访问。禁用了不适用于当前器件或配置模式的选项。

选项	说明
Dual-purpose pins	<ul style="list-style-type: none"> Use as regular I/O—不保留两用管脚。除非 I/O 管脚处于用户模式下。 Use as programming pin—nCEO 管脚保留为专用编程管脚。 As input tri-stated—两用管脚保留为输入管脚。 As output driving ground—两用管脚保留为输出管脚并驱动接地信号。 As output driving an unspecified signal—两用管脚保留为输出管脚并驱动各种信号。 Compiler configured—鉴于当前配置方案以及管脚是否仅用于配置，Compiler 会自动选择两用管脚的最佳保留设置。如果您的设计使用 Active Parallel 配置方案，而 Programmer 未直接与用户模式下的通信，则应按照 Compiler 配置，保留与并行 flash 器件连接的所有两用管脚。

表 18. 电路板走线模型选项

仅 Intel Cyclone 10 GX 设计，允许指定每个 I/O 标准的电路板走线，端接和电容性负载参数。电路板走线模型参数适用于所有依据 I/O 标准约束的输出或双向管脚。如果将电路板走线模型参数约束到任何非输出或双向管脚，则这些参数不可用。可在 Pin Planner 中为单个输出或双向管脚创建电路板走线模型约束。可点击 **Assignments > Device > Device and Pin Options > Board Trace Model** 进行访问。禁用了不适用于当前器件或配置模式的选项。

选项	说明
I/O standard	指定支持的 I/O 标准，例如 Differential 1.8-V SSTL Class II 。
Board trace model	罗列电路板走线模型参数，以及单位，和指定 I/O standard 的值。可更改每个参数值。电路板走线模型约束适用于所有已得到指定 I/O 标准分配的输出和双向管脚。

表 19. I/O 时序选项

允许在输出 I/O 时序端接处指定节点。可点击 **Assignments > Device > Device and Pin Options > Unused Pins** 进行访问。禁用了不适用于当前器件或配置模式的选项。

选项	说明
Default timing I/O endpoint	指定 Near end 或 Far end 。

表 20. 电压选项

允许指定目标器件上管脚的默认 I/O 块电压。还显示器件的内核电压或其他内部电压信息。可通过 **Assignments > Device > Device and Pin Options > Voltage** 进行访问。禁用了不适用于当前器件或配置模式的选项。

选项	说明
Default I/O standard	可指定为 1.2 V 、 1.5 V 、 1.8 V 、 2.5 V 、 3.0 LVTTTL 或 3.0 LVCMOS 。

表 21. 错误检测 CRC 选项

允许指定是否使用错误检测循环冗余校验 (error detection cyclic redundancy check, CRC) 以及指定划分当前所选器件错误检测频率的值。通过点击 **Assignments > Device > Device and Pin Options > Error Detection CRC** 进行访问。禁用了不适用于当前器件或配置模式的选项。

选项	说明
Enable Error Detection CRC_ERROR pin	使能目标器件的错误检测 CRC 和 CRC_ERROR 管脚。该检测会确定器件中编程数据的有效性。器件运行时数据中的任何变更都会生成错误。 <i>注意:</i> 不适用于 Intel Agilex 或 Intel Stratix 10 器件。
Enable Open Drain on CRC Error pin	将 CRC_ERROR 管脚设置为漏极开路管脚。该操作会解耦 VCCIO 电压中 CRC_ERROR 管脚的电压电平。开启该选项时, 必须将上拉电阻器连接到 CRC_ERROR 管脚。 <i>注意:</i> 不适用于 Intel Agilex 或 Intel Stratix 10 器件。
Enable error detection check	使能错误检测 CRC 检查以验证器件中编程数据的有效性, 并在器件运行过程中报告数据中的变更情况。
Minimum SEU interval	指定对同一比特位 2 次检查之间的最小时间间隔。设置为 0 表示尽可能频繁的检查。设置为较大值时可节省功耗。间隔时间的单位为毫秒 (millisecond)。允许的最大间隔数值为 10000。
Enable internal scrubbing	指定使用内部擦拭以在器件运行时纠正正在内核配置存储器内部检测到的单个错误或双邻错误。
Generate SEU sensitivity map file	生成 Single Event Upset Sensitivity Map 文件。该文件支持使能 Advanced SEU 检测功能。
Allow SEU fault injection	允许注入故障码型以测试 SEU。

表 22. CvP 设置

指定 Configuration via Protocol (CvP) 的配置模式。可通过点击 **Assignments > Device > Device and Pin Options > CvP Settings** 访问。禁用了不适用于当前器件和配置模式的选项。

选项	说明
Configuration via protocol	Initialization and update 模式中, 外设映像储存在外部配置器件中, 并且通过常规配置方案将映像加载到 FPGA 中。内核映像储存在主机存储器中, 并通过 PCIe 链接加载到 FPGA 中。 Core initialization 模式中, 外设映像储存在外部配置器件中, 并通过常规配置方案加载到 FPGA 中。内核映像储存在主机存储器中, 且通过 PCIe 链接加载到 FPGA 中。 Core update 模式下, 先通过将外部本地配置器件中的完整配置映像加载到 FPGA 来初始化 FPGA 器件。可通过此模式使用 PCIe 链接执行 1 个或多个 FPGA 核映像更新。在 Off 模式下, CvP 关闭。
Enable CvP_CONFDONE pin	显示器件已完成 Configuration via Protocol 模式下的内核编程。如果关闭该选项, 则 CvP_CONFDONE 管脚可用作用户 I/O 管脚, 但器件运行于用户模式下时为禁用。 <i>注意:</i> 不适用于 Intel Agilex 或 Intel Stratix 10 器件。
Enable open drain on CvP_CONFDONE pin	使能 CvP_CONFDONE 管脚上的开漏。 <i>注意:</i> 不适用于 Intel Agilex 或 Intel Stratix 10 器件。

表 23. 局部重配置选项

指定局部重新配置需要的辅助编程文件的生成。可通过点击 **Assignments > Device > Device and Pin Options > Partial Reconfiguration** 进行访问。禁用了不适用于当前器件或配置模式的选项。

选项	说明
Enable partial reconfiguration pins	支持使能 PR_REQUEST, PR_READY, PR_ERROR, PR_DONE, DCLK 和 DATA[31..0] 管脚。需要这些管脚来支持通过外部主机的局部重新配置 (PR)。外部主机使用 PR_REQUEST 管脚请求局部重新配置, 使用 PR_READY 管脚确定器件是否已准备接收编程数据, 通过 PR_ERROR 管脚监控是否存在外部编程错误, PR_DONE 管脚显示器件是否完成编程。如果关闭该选项, 则两用编程管脚可用作用户 I/O 管脚, 但当器件运行于用户模式时, 这些管脚将不可用做 PR 管脚。
继续...	

选项	说明
	注意: 不适用于 Intel Agilex 或 Intel Stratix 10 器件。
Enable open drain on partial reconfiguration pins	支持在 PR_READY, PR_ERROR, PR_DONE Partial Reconfiguration 管脚上指定漏极开路。 注意: 不适用 Intel Agilex 或 Intel Stratix 10 器件。
Generate Partial-Masked SOF files	生成 Partial-Masked SRAM Object 文件 (.pmsf), 包含可用于重新配置器件区域的配置数据和区域定义。如果开启该选项, 则将生成 .pmsf, 而非 Mask Settings 文件 (.msf)。
Generate Partial Reconfiguration RBF	生成 Partial Reconfiguration Raw Binary File (.rbf), 外部智能控制器可使用该文件中包含的配置数据重新配置目标器件的分区。

表 24. 电源管理& VID 选项

仅用于 Intel Stratix 10 和 Intel Agilex 器件, 可指定电源管理的选项, 例如总线速度模式, 以及 PMBus Master 模式下, 电压调节器从地址。可通过点击 **Assignments > Device > Device and Pin Options > Power Management & VID Options** 进行访问。禁用了不适用于当前器件或配置模式的选项。

选项	说明
Bus speed mode	生成 Partial-Masked SRAM Object 文件 (.pmsf), 包含可用于重新配置器件区域的配置数据和区域定义。如果开启该选项, 则将生成 .pmsf, 而非 Mask Settings 文件 (.msf)。
Slave device type	生成 Partial Reconfiguration Raw Binary File (.rbf), 外部智能控制器可使用该文件中包含的配置数据重新配置目标器件的分区。
Device address in PMBus Slave mode	PMBus Slave 模式下, 指定起始 00 器件地址。
PMBus device 0 slave address through PMBus device 7 slave address	指定 7-bit 十六进制值 (无前导前缀 0x)。例如, 将 7F 指定为电压调节器的从地址。必须指定一个非零地址。
Voltage output format	指定 PMBus Master 模式下的 Auto discovery , Direct format 或 Linear format 输出电压格式。
Direct format coefficient (m,b,R)	指定 PMBus Master 模式下的直接格式系数 m, b, 或 R。有符号整数介于 -32768 和 32767 之间。系数 m 是斜率系数。系数 b 是偏移。系数 R 是指数。请参阅 PMBus 器件制造商的产品文档获取这些值。当 PMBus 器件的输出电压格式为 Direct format 或 Auto discovery 格式时, 必须设置该参数。在 PMBus 器件的输出电压格式为 Direct format 时, 必须指定一个非零地址。
Linear format N	PMBus Master 模式时, 指定线性格式 N。有符号整数介于 -16 和 15 之间。这是 VOUT 格式设置为 Linear format 时, 输出电压相关命令的尾数。请参阅 PMBus 器件制造商的产品文档获取这些值。必须为 Linear format 指定一个非零值。
Translated voltage value unit	指定 PMBus Master 模式时, Volts 或 Millivolts 输出电压格式。
Enable PAGE command	FPGA PMBus 主接口使用 PAGE 命令设置所有已寄存稳压器模块上的所有输出通道以响应 VOUT_COMMAND。

表 25. Assembler 安全性选项

对于 Intel Stratix 10 器件, 可指定编程比特流验证, 加密, 加扰和其他启用 eFuse 安全选项的设置。通过点击 **Assignments > Device > Device and Pin Options > Security** 进行访问。禁用了不适用于当前器件或配置模式的选项。

选项	说明
Quartus Key File	指定您生成的第一级签字链文件 (.qky)。该链包含的根密钥 (.pem) 和 1 个或多个设计签字密钥 (.pem), 是使用验证和加密时, 对比特流进行签字和允许访问 FPGA 所需要的密钥。
Encryption key storage select	指定存储 .qek 密钥文件的位置。可选择使用 Battery Backup RAM 或 eFuses 存储。
继续...	

选项	说明
Encryption update ratio	指定配置比特与比特流解密需要的密钥更新次数的比率。可选择 31:1 （该密钥每 31 位变更 1 次）或 Disabled （无需更新）。最多支持 20 个中间密钥加密。
Enable scrambling	加扰配置比特流。
更多选项	打开 More Security Options 对话框指定其他物理安全选项。

表 26. 配置 PIN 对话框

对于 Intel Stratix 10 器件，允许使能或禁用特定的配置管脚。例如，可使能 CVP_CONF_DONE 管脚，以指示 Configuration via Protocol 模式中的器件已完成内核编程。可通过点击 **Assignments > Device > Device and Pin Options > Configuration Pin Options** 进行访问。禁用了不适用于当前器件或配置模式的选项。

选项	值	说明
USE PWRMGT_SCL output	SDM_100 SDM_IO14	当 VID 运行模式为 PMBus Master 或 PMBus Slave 模式时，这是电源管理所需要的 PMBus 接口。 Disable this pin for a non-SmartVID device. Intel 建议此功能使用 SDM_IO14 管脚。
Use PWRMGT_SDA output	SDM_1011 SDM_1012 SDM_1016	当 VID 运行模式为 PMBus Master 或 PMBus Slave 模式时，这是电源管理所需要的 PMBus 接口。 Disable this pin for a non-SmartVID device. Intel 建议该功能使用 SDM_IO11 管脚。
Use PWRMGT_ALERT output	SDM_100 SDM_1012	电源管理所需的 PMBus 接口，仅用于 PMBus Slave 模式。 对于非 SmartVID 器件，则请禁用此管脚。 Intel 建议此功能使用 SDM_IO12 管脚。
USE CONF_DONE output	SDM_100, SDM_1010 - SDM_1016	使用适当的配置管脚资源实现 CONF_DONE。
USE INIT_DONE output	SDM_100, SDM_1010 - SDM_1016	使能 INIT_DONE 管脚，从而在完成初始化且器件处于用户模式时允许从外部进行监控。如果关闭该选项，则 INIT_DONE 管脚可用作用户 I/O 管脚，但器件运行于用户模式时为禁用。
USE CVPCONF_DONE output	SDM_100, SDM_1010 - SDM_1016	使能 CVP_CONF_DONE 管脚，以显示 Configuration via Protocol 模式中器件已完成内核编程。如果关闭该选项，则 CVP_CONF_DONE 管脚可用作用户 I/O 管脚，但器件运行于用户模式时为禁用。
USE SEU_ERROR output	SDM_100, SDM_1010 - SDM_1016	使能用于单事件翻转错误检测的 SEU_ERROR 管脚。
USE UIB CATTRIP output	SDM_100, SDM_1010 - SDM_1016	使能 UIB_CATTRIP 输出以显示因使用 UIB 而导致的极端超高温情况。
USE HPS cold nreset	SDM_100, SDM_1010 - SDM_1016	可选的复位输入仅冷复位 HPS，并被配置用于双向操作。
Direct to factory image	SDM_100, SDM_1010 - SDM_1016	如果该管脚置位，则器件在引导后会加载出厂映像作为首个映像，而不加载任何应用程序映像。
USE DATA LOCK output	SDM_100, SDM_1010 - SDM_1016	输出以显示同一封装中两个芯片上的 DIB 均已准备号进行数据传输。

(9)

[相关链接](#)[使能 Bitstream Authentication \(Programming File Generator\)](#) (第 16 页)⁽⁹⁾ Intel Agilex 器件尚无安全选项。

2.9.2. 更多安全选项对话框

表 27. 更多安全选项对话框

对于 Intel Stratix 10 器件，指定其他配置比特流物理安全设。可通过点击 **Assignments > Device > Device and Pin Options > Security > More Settings** 按钮访问这些设置。禁用了不适用于当前器件或配置模式的选项。

选项	说明	值
Disable JTAG	禁用 JTAG 命令和器件配置。设置该选项将消除 JTAG 用于入侵模式，但同时消除了边界扫描功能。	<ul style="list-style-type: none"> Off—无效 On—有效，直到擦除所包含的设计 On sticky—有效，直到下个 POR On check—检查相应熔丝是否熔断
Force SDM clock to internal oscillator	禁用 SDM 的外部时钟源。SDM 必须使用内部振荡器（oscillator）。使用内部振荡器比允许使用外部时钟源进行配置更为安全。	
Force encryption key update	指定加密密钥必须由您在 Encryption update ratio 选项中指定的频率进行更新。默认配比为 31:1。最多支持 20 个中间密钥加密。	
Disable virtual eFuses	禁用 eFuse 虚拟编程功能。	
Lock security eFuses	如果 eFuse CRC 不符合计算值，则导致 eFuse 故障。	
Disable HPS debug	禁用通过 JTAG 接口访问 HPS 而进行的调试。	
Disable encryption key in eFuses	指定器件不可使用 eFuses 中存储的 AES 密钥。然而可通过使用 BBRAM 中存储的 AES 密钥提供更高级别的安全性。	
Disable encryption key in BBRAM	指定不可使用 BBRAM 中存储的 AES 密钥的器件。然而可通过使用 eFuses 中存储的 AES 密钥提供更高级别的安全性。	

(10)

2.9.3. 输出文件选项卡设置 (Programming File Generator)

Output Files 选项卡允许指定需要通过 **Programming File Generator** 生成（输出）的辅助编程文件类型。**Programming File Generator** 将主要编程文件（例如，.sof）转换成使用备用编程方式的编程文件（例如，用于 Flash 编程的 .jic）。**Output Files** 选项卡和选项根据您的选择动态更换。

可使用的输出文件选项如下：

表 28. 输出文件选项

设置	说明
Device family	指定进行配置的目标 FPGA 器件系列。 Programming File Generator 仅支持 Intel Agilex，Intel Stratix 10，Intel MAX 10 和 Intel Cyclone 10 LP 器件。
Configuration mode	指定 FPGA 配置方式，例如 Active Serial x4 ， AVST x8 ， AVST x16 或 AVST x32 。Generic Flash Programmer 仅支持 Active Serial x4 。
Output directory 和 Name	指定生成文件的名称和位置。默认情况下，其位于顶层工程目录中。
文件类型	允许使能需要生成的辅助编程文件类型。通用 Flash Programmer 仅支持 JTAG Indirect Configuration File (.jic) 。可用选项包括： <ul style="list-style-type: none"> JTAG Indirect Configuration File (.jic) Programmer Object File (.pof) Raw Binary File for CvP Core Configuration (.rbf)

继续...

(10) Intel Agilex 器件尚无可用的安全选项。

设置	说明
	<ul style="list-style-type: none"> • Raw Binary File for HPS Core Configuration (.rbf) • Raw Binary File for Partial Reconfiguration (.rbf) • Raw Programming Data File (.rpd)

2.9.4. 输入文件选项卡设置 (Programming File Generator)

Input Files 选项卡支持指定包含所需配置比特流数据 的 .sof, .pmsf, 或 .rbf 文件以生成 1 个或多个辅助编程文件。**Input Files** 选项卡和选项根据您的 **Output Files** 选项卡选择动态更换。

可使用的输入文件选项如下：

表 29. 输入文件设置：

设置	说明
Add Bitstream	点击该按钮指定 .sof, .pmsf 或 .rbf 作为您在 Output Files 中选择的辅助编程文件生成的输入。针对各目标器件，Intel Quartus Prime 软件可允许添加多个 SOF 文件。
Add Raw Data	点击该按钮指定包含原始编程数据的 .hex 或 .bin 文件，作为您在 Output Files 中选择的辅助编程文件生成的输入。
Remove	删除从 Input Files 选项卡中选择的文件。
Properties	显示 Input Files 选项卡中所选项目的属性。

相关链接

使能比特流加密 (Programming File Generator) (第 19 页)

2.9.5. 比特流联合签字安全设置 (Programming File Generator)

表 30. 输入文件属性对话框 (Programming File Generator)

允许指定比特流验证，联合签字和加密安全的选项。请访问 **Programming File Generator**，在 **Input files** 选项卡里选择 .sof 或 .rbf，并点击 **Properties**。

选项	说明
Bootloader	指定一个 Intel 十六进制格式的 ASCII 文本文件，该文件包含用于对并行数据源（例如配置器件或大容量存储器件）进行编程的配置数据。并行数据源依次配置基于 SRAM 的 Intel 器件。
Enable signing tool	使能签字工具，检查是否具有用于 Co-signed firmware 选项的 Private key file 和 Quartus Co-Signed Firmware 文件 (.zip) 所需的 Privacy Enhanced Mail Certificates 文件 (.pem)。
Private key file	指定使用签字工具时，对配置比特流签字所需的专用 .pem 文件。如果您的 .pem 受密码保护，则会提示输入密码。
Co-signed firmware	指定所需的固件源 (.zip)，以便将已签字固件包含于配置比特流。
Finalize encryption	最终确定配置比特流加密。
Encryption key file	指定解密配置比特流文件所需要的 Encryption Key File (.gek)。

(11)

(11) 安全选项不适用于 Intel Agilex 器件。

2.9.6. 配置器件选项卡设置

Configuration Device 选项卡支持为需要编程的现有或新的 Flash 存储器件指定属性。点击 **Add Device** 为预定义的 Flash 存储器选择编程模板，或点击 **<<new device>>**，然后定义一个新的 Flash 存储器件。

可用设置如下：

表 31. 配置器件选项卡设置

选项	说明
Device name	为 flash 指定一个 Name 栏中没有的特定名字。 Name 不得包含任何空字符串（比如空格）或特殊符号（“_”除外）。
Device ID	指定 3-byte ID，Programmer Auto-Detect 操作通过其检测 flash 编程器件，例如 0x20 0xBB 0x21。
Device I/O voltage	指定 1.8V 或 3.0/3.3V 以匹配您的存储器件规格。
Device density	选择与 Flash 存储器件大小相应的总密度。
Total device die	指定堆栈式器件的芯片总数（适用时）。
Single I/O mode dummy clock	为单 I/O 协议下的 Flash 器件指定 Fast Read 虚时钟周期。编程文件生成使用该设置确定配置是否需要位移动（bit shifting）以对 Active Serial 配置期间的实际虚时钟周期进行补偿。
Quad I/O mode dummy clock	为 Quad I/O 协议下的 Flash 器件指定 Fast Read 虚时钟周期（dummy clock cycle）。编程文件生成使用该设置确定配置是否需要位移动（bit shifting）以对 Active Serial 配置期间的实际虚时钟周期进行补偿。
Custom database directory	为保留 Flash 存储器件定义的 .xml 文件指定位置。 注意：将 Custom database directory 的位置指定到非默认文件夹时，请把 .sof 和 .jic 文件放置于与 .xml 文件相同的文件夹，以免丢失已定义的 Flash 数据库或 .jic 文件损坏。

2.9.7. 添加分区对话框 (Programming File Generator)

Programming File Generator 中打开，点击 **Configuration Device** 选项卡，从列表中选择器件，并点击 **Add Partition**。

允许指定新分区的属性。可用设置如下：

表 32. 添加分区对话框设置

设置	说明
Name	您给定的分区名称。
Input file	编程到 Flash 分区中的输入文件。
Page	配置器件可将多个配置比特流存储于 flash 存储器（称作，页面）中。CFI 配置器件最多可存储 8 个配置比特流。Intel Hyperflex 器件可最多存储 4 个配置比特流，包括出厂映像。 Intel Hyperflex 器件中，在启用远程系统更新功能时， Page 代表奇偶性。
Address Mode	选项包括： <ul style="list-style-type: none"> Auto—在 Flash 器件中自动分配一个块以存储数据。 Block—指定 Flash 分区的起始和结束地址。 Start—指定分区的起始地址。该工具基于输入数据的大小分配分区结束地址。
Start address	指定分区的起始地址。仅当 Address Mode 为 Block 或 Start 时启用。
End address	指定分区的结束地址。仅当 Address Mode 为 Block 时启用。

2.9.8. 转换编程文件对话框

允许转换或合并支持备用器件配置方案的 1 个或多个辅助编程文件，如 flash 编程，部分重配置，或远程系统更新。

表 33. 转换编程文件对话框设置

设置	说明
Programming file type	允许指定用于主要编程文件转换的辅助编程文件格式。Generic Flash Programmer 仅支持 .jic 文件类型。
Configuration device	允许选择预定义的配置器件或定义新的配置器件。点击 (...) 按钮定义新的器件和编程流程。
Mode	选择器件配置的方法。Generic Flash Programmer 仅支持 Active Serial 或 Active Serial x4 模式。
Output file	指定 Convert Programming File 生成文件的位置。默认情况下该位置在工程目录顶层。
Input files to convert	指定 1 个或多个主要编程文件，用于转换或合并 1 个或多个辅助编程文件作为替代编程方式。

2.9.9. 压缩和加密设置 (Convert Programming File)

压缩和加密设置允许为用于器件配置的 SRAM Object File (.sof) 指定压缩和加密密钥安全选项。要访问该设置，可从 **Convert Programming File** 对话框中的 **Input files to convert** 清单选择 .sof，并点击 **Properties**。

表 34. SOF 文件属性：比特流加密对话框 ((Convert Programming Files)

允许为用于器件配置的 SRAM Object File (.sof) 指定压缩和加密密钥安全性选项。请访问 **Convert Programming Files** 对话框，在 **Input files to convert** 列表中选择 .sof，然后点击 **Properties**。

选项	说明
Compression	压缩比特流，可减小编程文件的大小。Intel Quartus Prime Assembler 可生成一个压缩的比特流图像，从而将配置文件减小 30% 至 55%（具体取决于设计）。FPGA 器件接收压缩的配置比特流，然后可在配置过程中实时解压数据。使能 Generate encrypted bitstream 时，该选项不可用。
Enable decompression during partial reconfiguration	使能该选项位，以便 Partial Reconfiguration 期间解压压缩比特流解。
Generate encrypted bitstream	生成加密比特流配置映像。然后生成并指定用于器件配置的加密密钥文件 (.ekp)。使能 Compression 时，该选项不可用。
Enable volatile security key	允许对 .sof 文件使用易失（已使能）或非易失（禁用）安全密钥进行加密。
Generate encryption lock file	对 Convert Programming Files 生成的加密锁定文件 (.elk) 指定名称。
Generate key programming file	对 Convert Programming Files 生成的加密锁定文件 (.elk) 指定名称。
Use key file	<ul style="list-style-type: none"> Key 1 file—指定 Key 1 .key 文件的名称。 Key 2 file—指定 Key 2 .key 文件的名称。
Key entry	指定比特流解密的密钥。
Security options	以下选项允许使能或禁用影响配置比特流器件安全性的功能。

继续...

选项	说明
	<ul style="list-style-type: none"> • Disable partial reconfiguration—禁用对比特流局部重配置。 • Disable key-related JTAG instructions—禁用与比特流密钥相关的 JTAG 指令。 • Disable other extended JTAG instructions—禁用对比特流使用其他其他 JTAG 指令。 • Force the external JTAG pins into BYPASS mode—强制外部 JTAG 管脚进入 BYPASS 模式。可指定为 Off (关闭)，Turns On Until the Next Full Configuration (开启直到下一次完全配置)，Turns on until the next Power-On-Reset event (开启直到下一次 Power-On-Reset 事件)，Turns on by blowing the corresponding fuses (通过相应熔丝熔断而开启)。
Design Security Feature Disclaimer	确认需要接受 Design Security Disclaimer。

2.9.10. SOF 数据属性对话框 (Convert Programming File)

允许定义存储配置数据的 flash 存储器页面。可通过 **Convert Programming File** 对话框，点击 **SOF Data** 项，然后点击 **Properties** 按钮进行访问。

可用设置如下：

表 35. SOF 数据属性对话框设置

设置	说明
Pages	配置器件可将多个配置比特流存储于 flash 存储器（称作，页面）中。CFI 配置器件最多可存储 8 个配置比特流。某些 Intel FPGA 器件可存储多个配置比特流，包括出厂映像。
Address mode for selected pages	选项包括： <ul style="list-style-type: none"> • Auto—在 Flash 器件中自动分配一个块以存储数据。 • Block—指定 Flash 分区的起始和结束地址。 • Start—指定分区的起始地址。该工具基于输入数据的大小分配分区结束地址。
Start address	指定分区的起始地址。仅当 Address Mode 为 Block 或 Start 时启用。
End address	指定分区的结束地址。仅当 Address Mode 为 Block 时启用。

2.9.11. 选择器件 (Flash Loader) 对话框

允许选择一个器件，控制对 Flash 存储器件的配置数据加载。要从 **Programming File Generator** 进行访问，可以在 **Configuration Device** 选项卡中为 **Flash loader** 点选 **Select** 按钮。要从 **Convert Programming File** 对话框进行访问，请选择 **Flash Loader** 项并点击 **Add Device**。

可用设置如下：

表 36. Flash Loader (选择器件对话框)

选项	说明
Device family	指定 Flash 加载程序器件的系列。
Device name	指定 Flash 加载程序器件的名称。

2.10. 脚本支持

除 Intel Quartus Prime Programmer GUI 以外，还可从命令行和 Intel Quartus Prime 命令行可执行文件 `quartus_pgm.exe` (或 Linux 中的 `quartus_pgm`) 接入编程配置工具的功能。

以下命令编程器件：

```
quartus_pgm -c usbblasterII -m jtag -o bpv;design.pof ←
```

其中：

-c usbblasterII 指定 Intel FPGA Download Cable II

-m jtag 指定 JTAG 编程模式

-o bpv 代表 blank-check（空白检查），编程和验证操作

design.pof 代表 .pof 包含设计逻辑

Programmer 在对器件编程之前自动执行擦除操作。

对于 Linux 终端，请使用：

```
quartus_pgm -c usbblasterII -m jtag -o bpv\;design.pof
```

相关链接

[Intel Quartus Prime 脚本](#)

Intel Quartus Prime Help 中

2.10.1. jtagconfig 调试工具

可使用 jtagconfig 命令行实用程序检查 JTAG 链中的器件以及用户指定器件。jtagconfig 命令行实用程序类似于 Intel Quartus Prime Programmer 中的自动检测操作。

有关 jtagconfig 实用程序的更多信息，请使用命令提示处的可用帮助：

```
jtagconfig [-h | --help]
```

注意： 此帮助切换并不涉及 -n 切换。jtagconfig -n 命令显示每个 JTAG 器件的每个节点。

相关链接

[Command Line Scripting](#)

In Intel Quartus Prime Pro Edition 用户指南：脚本

2.11. 编程 Intel FPGA 器件修订历史

表 37. 文档修订历史

日期	Intel Quartus Prime 版本	修订内容
2019.09.30	19.3.0	<ul style="list-style-type: none"> 更新了“器件&管脚选项”主题，以反映新的 Security 设置选项卡。 更新了“配置期间选项卡”主题，以反映 Custom 数据库目录选项。 引用了对 Intel Agilex 器件的编译支持。
继续...		

日期	Intel Quartus Prime 版本	修订内容
		<ul style="list-style-type: none"> • 添加了“更多安全选项对话框”主题。 • 添加了新的“比特流联合签字设置”主题。 • 更新了“SOF 文件属性: 比特流加密对话框”主题。 • 在“完全擦除 Flash 存储器扇区”主题中添加了新步骤。
2019.06.10	19.1.0	<ul style="list-style-type: none"> • 更新了“使用 Flash Loaders 编程”主题, 以反映新的 Generic Flash Programmer。 • 删除了有关淘汰的 32-bit 独立 Programmer 的参考内容。 • 添加了“擦除 Flash 存储器扇区”主题, 介绍完全擦除 flash 存储器。 • 添加了新的“Programmer 设置参考”部分, 其中包括以下新的 GUI 参考主题: <ul style="list-style-type: none"> — “器件&管脚选项对话框” — “输入文件选项卡设置 (Programming File Generator)” — “输出文件选项卡设置 (Programming File Generator)” — “配置器件选项卡设置 (Programming File Generator)” — “添加分区对话框 (Programming File Generator)” — “比特流压缩, 认证和加密设置 (Programming File Generator)” — “转换编程文件对话框” — “比特流压缩和加密设置 (Convert Programming File)” — “SOF 数据属性对话框” — “选择器件 (Flash Loader) 对话框”
2019.04.01	19.1.0	<ul style="list-style-type: none"> • 添加了新的“使用 PR 比特流安全验证”主题。 • 添加了新的“基础器件配置步骤”主题。 • 添加了“编程和配置模式”主题 • 重列标题并重组主题以改善信息流。 • 将 Programmer 的增强图添加到“Intel Quartus Prime Programmer”主题。 • 更新了截图。
2018.10.09	18.1.0	<ul style="list-style-type: none"> • 创建了主题: <i>Stand-Alone Programmer Memory Limitations</i>, 按照主题: <i>Stand-Alone Programmer</i> 中的内容进行创建。 • 删除了过期的支持信息。
2018.08.07	18.0.0	经文档标题还原为 <i>Programmer 用户指南: Intel Quartus Prime Pro Edition</i> 。
2018.06.27	18.0.0	<ul style="list-style-type: none"> • 将有关编程文件生成程序的信息移至新章节: <i>生成编程文件</i>。
2018.05.07	18.0.0	<ul style="list-style-type: none"> • 首次作为独立 <i>Programmer 用户指南</i> 中的一部分发布。
2017.05.08	17.0.0	<ul style="list-style-type: none"> • 添加了 Project Hash (工程哈希) 功能。
2016.10.31	16.1.0	<ul style="list-style-type: none"> • 实现品牌更名为 Intel。
2015.11.02	15.1.0	将 Quartus II 更改成 Intel Quartus Prime。
2015.05.04	15.0.0	添加了 Conversion Setup File (.cof) 说明和实例。
2014 年 12 月	14.1.0	更新了 Scripting Support 部分, 以包括编程器件的 Linux 命令。
2014 年 6 月	14.0.0	<ul style="list-style-type: none"> • 添加运行 JTAG Daemon。 • 删除了 Cyclone III 和 Stratix III 器件参考内容。 • 删除了 MegaWizard Plug-In Manager 的参考内容。 • 更新了“辅助编程文件”部分, 以添加有关 Quartus II Programmer 对 .rbf 文件的支持。
2013 年 11 月	13.1.0	<ul style="list-style-type: none"> • 转换成 DITA 格式。 • 为“局部重新配置”添加了“JTAG 调试模式”, 并在“JTAG 调试模式”部分中添加了“配置局部重新配置比特流”。
继续...		

日期	Intel Quartus Prime 版本	修订内容
2012 年 11 月	12.1.0	<ul style="list-style-type: none"> 更新了第 18-6 页的表 18-3、第 18-8 页的表 18-4。 在第 18-10 页添加了“转换编程文件进行部分重配置”；在第 18-10 页添加了“生成使用 .msf 和 .sof 的 .pmsf”；在第 18-12 页添加了“为使用 .pmsf 的 Partial Reconfiguration 生成 .rbf”；在第 18-14 页添加了“使能 Decompression during Partial Reconfiguration 选项” 更新了第 18-15 页中的“脚本支持”。
2012 年 6 月	12.0.0	<ul style="list-style-type: none"> 更新了第 18-8 页中的表 18-5。 更新了第 18-3 页的“Quartus II Programmer GUI”。
2011 年 11 月	11.1.0	<ul style="list-style-type: none"> 更新了第 18-5 页中的“配置模式”。 在第 18-6 页添加了“可选编程或配置文件”。 更新了第 18-5 页中的表 18-2
2011 年 5 月	11.0.0	<ul style="list-style-type: none"> 为 Quartus II “帮助” 添加了链接。 更新了第 21-4 页的“硬件设置”和“JTAG Chain Debugger 工具”
2010 年 12 月	10.1.0	<ul style="list-style-type: none"> 更换为新文档模板。 更新了第 20-4 页的“JTAG Chain Debugger 实例”。 为 Quartus II “帮助” 添加了链接。 重新组织章节。
2010 年 7 月	10.0.0	<ul style="list-style-type: none"> 为 Quartus II “帮助” 添加了链接。 删除了屏幕截图。
2009 年 11 月	9.1.0	内容无变化。
2009 年 3 月	9.0.0	<ul style="list-style-type: none"> 在表 21-4 中添加了一行。 将参考内容从“JTAG Chain 调试”更改为“JTAG Chain 调试器”。 更新了数字。

2.12. Intel Quartus Prime Pro 版用户指南：Programmer 文档存档

如有未列出的 Intel Quartus Prime 版本，可应用之前 Intel Quartus Prime 版本的用户指南。

Intel Quartus Prime 版本	用户指南
19.3.0	Intel Quartus Prime Pro 版用户指南：Programmer
19.1.0	Intel Quartus Prime Pro 版用户指南：Programmer
18.1.0	Intel Quartus Prime Pro 版用户指南：Programmer

A. Intel Quartus Prime Pro Edition 用户指南

请参阅以下用户指南获得关于 Intel Quartus Prime Pro Edition FPGA 设计流程中所有阶段的综合性信息。

相关链接

- [Intel Quartus Prime Pro Edition 用户指南：入门](#)
介绍 Intel Quartus Prime Pro Edition 软件的基本功能，文件和设计流程，包括管理 Intel Quartus Prime Pro Edition 工程和 IP，初始设计布局考量以及从软件先前版本进行工程移植。
- [Intel Quartus Prime Pro Edition 用户指南：平台设计程序](#)
说明使用 Platform Designer 创建和优化系统，该系统集成工具可简化工程中自定义 IP 核聚合。Platform Designer 自动生成互连逻辑以连接知识产权（IP）功能和子系统。
- [Intel Quartus Prime Pro Edition 用户指南：设计建议](#)
介绍使用 Intel Quartus Prime Pro Edition 软件进行 FPGA 设计时的最佳设计实践。HDL 代码样式和同步设计时间可显著影响设计性能。以下建议的 HDL 代码样式可确保 Intel Quartus Prime Pro Edition 将设计在硬件中最佳实现。
- [Intel Quartus Prime Pro Edition 用户指南：设计编译](#)
说明了 Intel Quartus Prime Pro Edition Compiler 从建立，运行到优化的全部阶段。生成器件编程文件之前，Compiler 对设计进行综合，布局和布线。
- [Intel Quartus Prime Pro Edition 用户指南：设计优化](#)
介绍可用于实现 Intel FPGA 中最高设计性能的 Intel Quartus Prime Pro Edition 设置，工具和技术。可使用的技术包括优化设计网表，解决限制重定时和时序收敛的关键链以及优化器件资源使用，器件平面布局规划以及实现工程变更命令（ECO）。
- [Intel Quartus Prime Pro Edition 用户指南：Programmer](#)
说明 Intel Quartus Prime Pro Edition Programmer 的运行，并通过连接 Intel FPGA 下载电缆配置 Intel FPGA 器件，编程 CPLD 和配置器件。
- [Intel Quartus Prime Pro Edition 用户指南：基于块的设计](#)
说明基于块的设计流程，亦称为模块化或分层式设计流程。这些高级流程可将设计块（或是包含分层型设计实例的逻辑）保留在工程中，并在其他工程中重复使用设计块。
- [Intel Quartus Prime Pro Edition 用户指南：局部重新配置](#)
介绍 Partial Reconfiguration 高级设计流程，其支持动态重配置 FPGA 某一部分的同时其余 FPGA 设计继续运行。将部分设计区域定义为多重角色时，并不影响其他区域中的操作。
- [Intel Quartus Prime Pro Edition 用户指南：第三方仿真](#)
说明通过 Aldec*，Cadence*，Mentor Graphics* 和 Synopsys* 为第三方仿真工具提供 RTL-和门级设计仿真支持，从而允许在器件编程之前验证设计行为。包括仿真器支持，仿真流程和仿真 Intel FPGA IP。
- [Intel Quartus Prime Pro Edition 用户指南：第三方综合](#)
说明通过 Mentor Graphics* 和 Synopsys*，第三方综合工具为设计中选择性综合部分提供支持。包括设计流程步骤，生成的文件说明和综合指导。

- [Intel Quartus Prime Pro Edition 用户指南：第三方逻辑等效检查工具](#)
说明通过 OneSpin*，第三方 LEC 工具为设计提供选择性逻辑等效检查（LEC）支持。
- [Intel Quartus Prime Pro Edition 用户指南：调试工具](#)
介绍为设计进行实时验证的 Intel Quartus Prime Pro Edition 在系统设计调试工具文件夹。这些工具通过将设计中的信号路由选择（或“分接”）到调试逻辑来提供可视性。具体工具包括，System Console，Signal Tap logic analyzer，Transceiver Toolkit，In-System Memory Content Editor 和 In-System Sources 以及 Probes Editor。
- [Intel Quartus Prime Pro Edition 用户指南：Timing Analyzer](#)
解释基本静态时序分析原则和 Intel Quartus Prime Pro Edition Timing Analyzer 的使用。其作为功能强大的 ASIC 式时序分析工具，通过使用行业标准的约束，分析和报告方法验证设计中所有逻辑的时序性能。
- [Intel Quartus Prime Pro Edition 用户指南：功耗分析和优化](#)
说明 Intel Quartus Prime Pro Edition Power Analysis 工具支持准确估算器件功耗。估算器件功耗以开发功率预算和设计电源，稳压器，散热器和冷却系统。
- [Intel Quartus Prime Pro Edition 用户指南：设计约束](#)
说明影响 Compiler 如何实现设计的时序和逻辑，例如，管脚约束，器件选项，逻辑选项和时序约束。使用 Interface Planner 原型开发接口实现，规划时钟并迅速定义合法器件平面图。使用 Pin Planner 在目标器件的图形呈现中可视化，修改和验证所有 I/O 约束。
- [Intel Quartus Prime Pro Edition 用户指南：PCB 设计工具](#)
说明通过 Mentor Graphics* 和 Cadence* 实现对可选第三方 PCB 设计工具的支持。还包括信号集成分析和使用 HSPICE 和 IBIS 模型进行仿真的信息。
- [Intel Quartus Prime Pro Edition 用户指南：脚本](#)
说明使用 Tcl 和命令行编制脚本进行 Intel Quartus Prime Pro Edition 软件控制并广泛执行各种功能，例如管理工程，指定约束，运行编译或时序分析，以及生成报告。