



# アルテラ・フォールト・インジェクション IP コア のユーザーガイド

**UG-01173**  
**2016.10.31**



## 目次

---

<b>1 アルテラ・フォールト・インジェクション IP コアのユーザーガイド</b> .....	<b>3</b>
1.1 特長.....	3
1.2 サポートされるデバイス.....	3
1.3 リソース使用率およびパフォーマンス.....	3
1.4 IP コアのインストールとライセンス取得.....	4
1.5 IP コアのカスタマイズと生成.....	4
1.5.1 IP カタログとパラメーター・エディター.....	4
1.5.2 IP コアの生成出力 ( Quartus Prime プロ・エディション).....	6
1.6 フォールト・インジェクション IP コアのインスタンス化.....	9
1.6.1 EMR Unloader IP コアを使用する.....	9
1.6.2 Advanced SEU Detection IP コアを使用する.....	10
1.7 機能の説明.....	11
1.7.1 シングル・イベント・アップセット(SEU)の緩和.....	11
1.7.2 Fault Injection Debugger とフォールト・インジェクション IP コアを使用する.....	12
1.7.3 アルテラ・フォールト・インジェクション IP ピンの概要.....	13
1.8 アルテラ・フォールト・インジェクション IP コアのユーザーガイドのアーカイブ.....	14
1.9 改訂履歴.....	14



## 1 アルテラ・フォールト・インジェクション IP コアのユーザーガイド

Altera® フォールト・インジェクション IP コアは、FPGA デバイスのコンフィグレーション RAM (CRAM) にエラーを注入することができます。

この手順により、SEU (Single Event Upset) のために通常動作時に生じかねないソフトウェアをシミュレーションすることができます。SEU は稀にしか発生しないため、テストすることが困難です。フォールト・インジェクション IP コアをデザインにインスタンス化してデバイスをコンフィグレーションした後、Quartus® Prime Fault Injection Debugger のツールを使用することにより、FPGA で意図的にエラーを発生させて、これらのエラーに対するシステムの対応をテストできます。

### 関連情報

- [Debugging Single Event Upset Using the Fault Injection Debugger \(Quartus Prime Standard Edition Handbook Volume 3: Verification\)](#)
- [Debugging Single Event Upset Using the Fault Injection Debugger \(Quartus Prime Pro Edition Handbook Volume 3: Verification\)](#)

### 1.1 特長

- SEFI (Single Event Functional Interrupts) の緩和のためにシステムの応答を評価することができる
- インハウスの SEFI 特性評価が実行でき、システム全体のビームテストが不要になる。その代わりに、ビームテストをデバイスレベルの FIT (Failure In Time) または Mb 測定に制限することができる
- デザインのアーキテクチャーに関する SEFI 特性評価に応じて FIT レートを調整する。フォールト・インジェクションはデバイス全体を通じてランダムに分配するか、またはテストを迅速化するために特定の機能エリアに制限することができる
- デザインを最適化し、SEU によって生じる妨害を軽減できる

### 1.2 サポートされるデバイス

フォールト・インジェクション IP コアは、Arria® 10 および Stratix® V ファミリーのデバイスをサポートします。Cyclone® V ファミリーでは、フォールト・インジェクションは製品コードで末尾が -SC のデバイスをサポートします。末尾が -SC の Cyclone V デバイスの注文について、販売代理店にお問い合わせください。

### 1.3 リソース使用率およびパフォーマンス

Quartus Prime ソフトウェアは、Stratix A7 FPGA に対して以下のリソース推定値を生成します。他のデバイスの結果も同様です。

Intel Corporation. 無断での引用、転載を禁じます。Intel、インテル、Intel ロゴ、Altera、ARRIA、CYCLONE、ENPIRION、MAX、NIOS、QUARTUS および STRATIX の名称およびロゴは、アメリカ合衆国および/またはその他の国における Intel Corporation の商標です。インテルは FPGA 製品および半導体製品の性能がインテルの標準保証に準拠することを保証しますが、インテル製品およびサービスは、予告なく変更される場合があります。インテルが書面にて明示的に同意する場合を除き、インテルはここに記載されたアプリケーション、または、いかなる情報、製品、またはサービスの使用によって生じるいっさいの責任を負いません。インテル製品の顧客は、製品またはサービスを購入する前、および、公開済みの情報を信頼する前には、デバイスの仕様を最新のバージョンにしておくことをお勧めします。

\*その他の社名、製品名などは、一般に各社の表示、商標または登録商標です。

ISO  
9001:2008  
登録済

表 1. フォールト・インジェクション IP コアの FPGA パフォーマンスとリソース使用率

デバイス	ALM 数	ロジック・レジスタ数		M20K
		プライマリ	セカンダリー	
Stratix V A7	3,821	5,179	0	0

## 1.4 IP コアのインストールとライセンス取得

Quartus Prime ソフトウェアのインストールには、アルテラ FPGA IP ライブラリーが含まれます。このライブラリーは追加のライセンスなしで製品使用に役立つ IP コアの機能を提供しています。ライブラリーの一部の MegaCore<sup>®</sup> IP 機能では、製品に使用する場合に個別のライセンスを購入する必要があります。OpenCore<sup>®</sup> 機能により、Quartus Prime ソフトウェアでシミュレーションおよびコンパイル時にいずれのアルテラ FPGA IP コアを評価することができます。機能性とパフォーマンスにご満足の場合、Self Service Licensing Center ページで任意のアルテラ FPGA 製品のライセンス番号を取得できます。

Quartus Prime ソフトウェアはデフォルトで IP コアを以下のロケーションにインストールします。

図 -1: IP コアのインストールパス

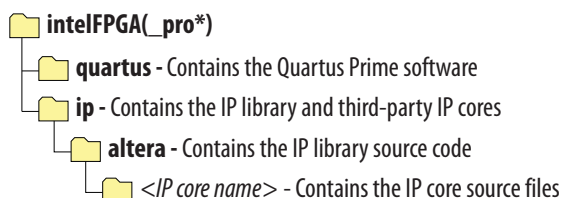


表 2. IP コアのインストール・ロケーション

ロケーション	ソフトウェア	プラットフォーム
<drive>:\intelFPGA_pro\quartus\ip\altera	Quartus Prime <sup>®</sup> プロ・エディション	Windows
<drive>:\intelFPGA\quartus\ip\altera	Quartus Prime スタンダード・エディション	Windows
<home directory>:/intelFPGA_pro/quartus/ip/altera	Quartus Prime プロ・エディション	Linux
<home directory>:/intelFPGA/quartus/ip/altera	Quartus Prime スタンダード・エディション	Linux

## 1.5 IP コアのカスタマイズと生成

IP コアをカスタマイズし、多種多様なアプリケーションのサポートが可能です。Quartus Prime IP Catalog とパラメーター・エディターの使用で、IP コアのポート、機能、および出力ファイルをすばやく選択し、コンフィグレーションすることができます。

### 1.5.1 IP カタログとパラメーター・エディター

IP カタログには、プロジェクトに使用可能な IP コアが表示されます。IP コアを探し出してカスタマイズするには、IP カタログの以下の機能を使用します。



- IP カタログを **Show IP for active device family** または **Show IP for all device families** でフィルタします。プロジェクトを開いていない場合には、IP カタログで **Device Family** を選択します。
- 検索フィールドに IP コアの完全な、または部分的な名称を入力し IP カタログ内を検索します。
- IP カタログで IP コアの名称を右クリックすることにより、サポートされるデバイスの詳細の表示、IP コアのインストレーション・フォルダーの開き、そして IP の資料へのリンク参照ができます。
- **Search for Partner IP** をクリックし、ウェブサイト上でパートナーの IP 情報へアクセスします。

パラメーター・エディターは、IP バリエーション名、オプションのポート、出力ファイルの生成オプションを指定するよう促します。また、パラメーター・エディターは、Quartus Prime プロ・エディション のプロジェクトの IP バリエーションに対しトップレベルの Quartus Prime IP ファイル (.ip) を生成します。

パラメーター・エディターは、Quartus Prime スタンダード・エディション のプロジェクトの IP バリエーションに対しトップレベルの Quartus IP ファイル (.qip) を生成します。これらのファイルはプロジェクトの IP バリエーションを表示し、パラメーター化情報を格納します。

図 -2: IP パラメーター・エディター (Quartus Prime プロ・エディション)

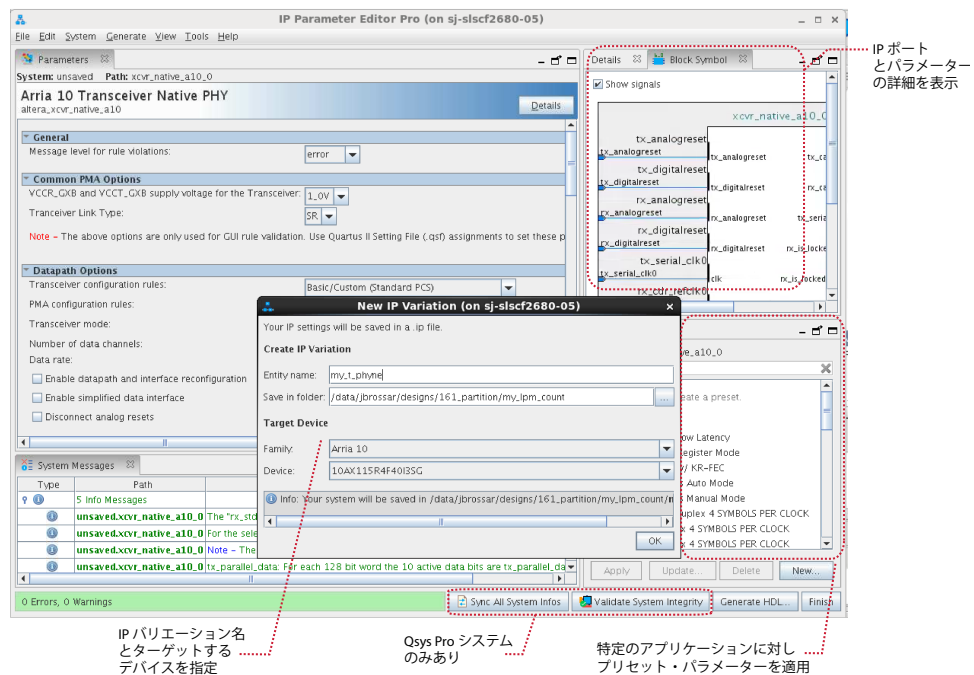
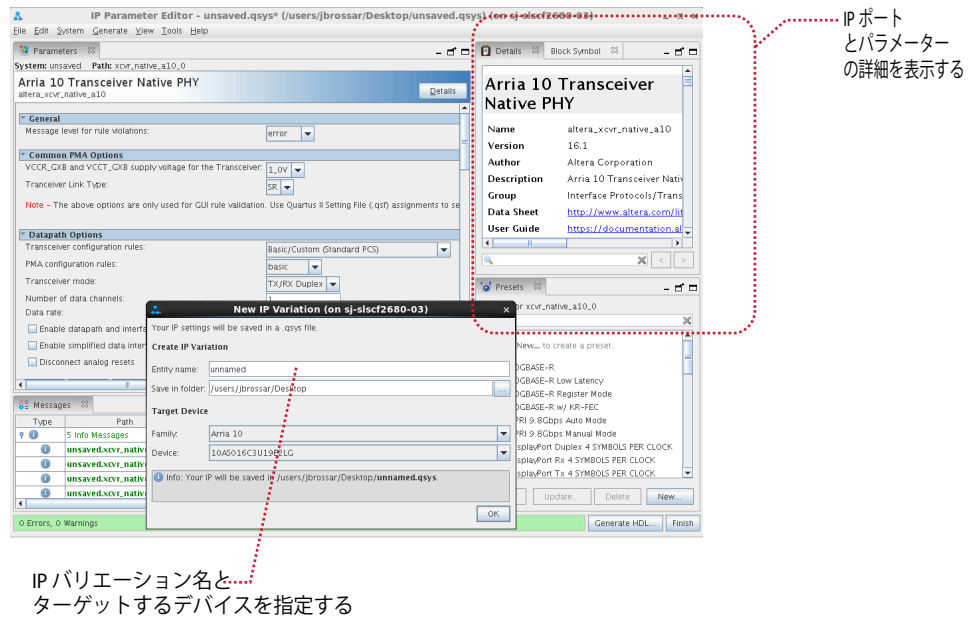


図 -3: IP パラメーター・エディター (Quartus Prime スタンダード・エディション)

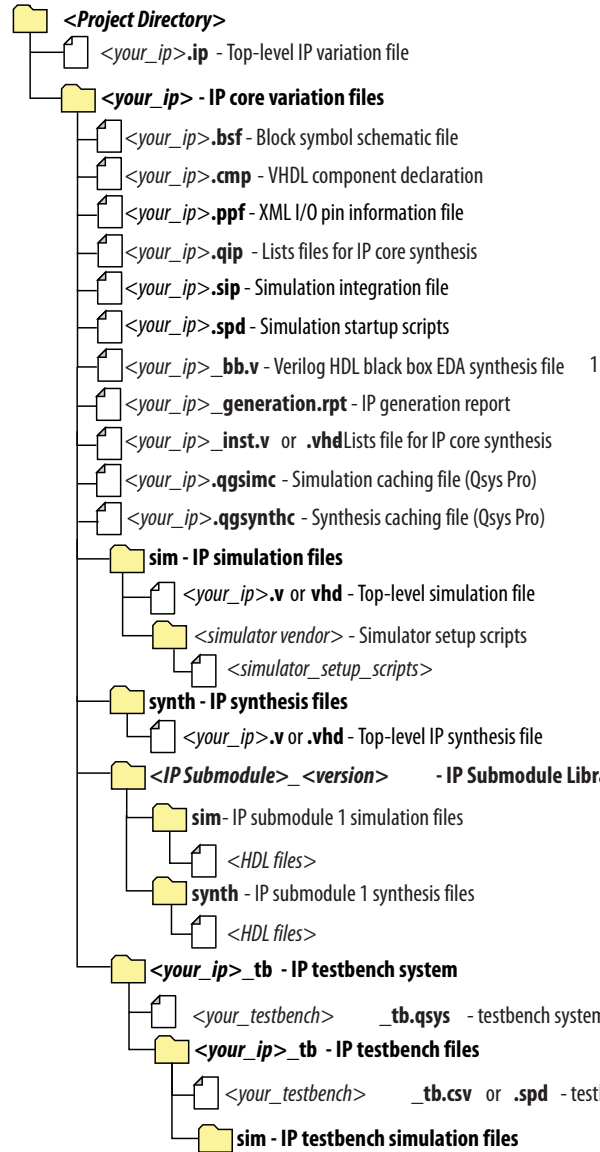


### 1.5.2 IP コアの生成出力 (Quartus Prime プロ・エディション)

Quartus Prime ソフトウェアは、Qsys システムの一部ではない個々の IP コアにおいて以下の出力ファイル構造を生成します。



図 -4: 個々の IP コアでの生成出力 ( Quartus Prime プロ・エディション)



1. IP コア・バリエーションに対しサポートされてイネーブルされる場合。

表 3. IP コアで生成されるファイル

ファイル名	説明
<my_ip>.ip	プロジェクトでの IP コアのパラメーター化を含むトップレベルの IP バリエーションのファイルです。IP バリエーションが Qsys Pro システムの一部の場合は、パラメーター・エディターは .qsys ファイルも生成します。
<my_ip>.cmp	.cmp (VHDL Component Declaration) ファイルは、VHDL デザインファイルで使用するローカル・ジェネリックおよびポート定義を含むテキストファイルです。
<my_ip>_generation.rpt	IP または Qsys 生成のログファイルで、IP 生成中のメッセージの要約を示します。
<i>continued...</i>	



ファイル名	説明
<my_ip>.qgsimc (Qsys Pro システムのみ)	.qsys と .ip ファイルを Qsys Pro システムと IP コアの現在のパラメーター化で比較するシミュレーション・キャッシング・ファイルです。この比較は、Qsys Pro が HDL の再生成をスキップできるかどうかを決定します。
<my_ip>.qgsynth (Qsys Pro systems only)	.qsys と .ip ファイルを Qsys Pro システムと IP コアの現在のパラメーター化と比較する合成キャッシング・ファイルです。この比較は、Qsys Pro が HDL の再生成をスキップできるかどうかを決定します。
<my_ip>.qip	IP コンポーネントを統合し、コンパイルするためのすべての情報を含みます。
<my_ip>.csv	IP コンポーネントのアップグレード・ステータスに関する情報を含みます。
<my_ip>.bsf	ブロック図ファイル (.bdf) で使用する IP バリエーションの表記です。
<my_ip>.spd	サポートされるシミュレーターのシミュレーション・スクリプトの生成のために ip-make-simscrip で必要な入力ファイルです。.spd ファイルは、シミュレーション用に生成するファイルのリスト、および初期化可能なメモリーについての情報を含みます。
<my_ip>.ppf	Pin Planner ファイル (.ppf) は、Pin Planner での使用のために作成した IP コンポーネントのポートとノードの割り当てを格納しています。
<my_ip>_bb.v	Verilog blackbox (_bb.v) ファイルは、ブラックボックスとして使用する空のモジュール宣言として使用します。
<my_ip>.sip	IP コンポーネントの NativeLink シミュレーションに必要な情報を含んでいます。サポートされるデバイスで NativeLink を有効にするには、.sip ファイルを Quartus Prime スタンド・エディション・プロジェクトに追加します。Quartus Prime プロ・エディション・ソフトウェアでは、NativeLink シミュレーションをサポートしていません。
<my_ip>_inst.v または _inst.vhd	HDL 例のテンプレート・インスタンスです。IP バリエーションのインスタンス化には、このファイル内容をコピーして、HDL ファイルに張り付けます。
<my_ip>.regmap	IP がレジスター情報を含む場合、Quartus Prime ソフトウェアは .regmap ファイルを生成します。.regmap ファイルは、マスターおよびスレーブ・インターフェイスのレジスターマップ情報を記述しています。このファイルは、システムに関するより詳しいレジスター情報を提供することで .sopcinfo ファイルを補完します。このファイルで、システムコンソールでレジスター・ディスプレイビューとユーザーのカスタマイズ・スタティックが可能になります。
<my_ip>.svd	Qsys Pro システム内で HPS に接続しているペリフェラルのレジスターマップを HPS System Debug ツールで表示できるようにします。 合成中、Quartus Prime ソフトウェアは、デバッグセッションに .sof ファイルにシステム・コンソール・マスターに表示されるスレーブ・インターフェイスの .svd ファイルを格納します。システムコンソールはこのセクションをリードし、これにより Qsys Pro がレジスターマップ情報を照会します。システムスレーブに対しては、Qsys Pro は名称によりそのレジスターにアクセスします。
<my_ip>.v <my_ip>.vhd	合成またはシミュレーション向けに各サブモジュールまたは IP コアをインスタンス化する HDL ファイルです。
mentor/	シミュレーションの設定と実行のための ModelSim® スクリプト msim_setup.tcl を含んでいます。
aldec/	シミュレーションの設定と実行のための Riviera-PRO スクリプト rivierapro_setup.tcl を含んでいます。
/synopsys/vcs /synopsys/vcsmx	VCS®/シミュレーションの設定と実行のためのシェルスクリプト vcs_setup.sh を含んでいます。 VCS MX シミュレーションの設定および実行のためのシェルスクリプトである vcsmx_setup.sh および synopsys_sim.setup ファイルを含んでいます。
/cadence	NCSIM シミュレーションの設定と実行のためのシェルスクリプト ncsim_setup.sh、およびその他の設定ファイルを含んでいます。
/submodules	IP コア・サブモジュールの HDL ファイルを含んでいます。
<IP submodule>/	生成された各 IP サブモジュール・ディレクトリー Qsys Pro において、/synth と /sim サブ・ディレクトリーを生成します。





## 1.6 フォールト・インジェクション IP コアのインスタンス化

フォールト・インジェクション IP コアでは、パラメーターを設定する必要はありません。IP コアを使用するには、新しい IP インスタンスを作成し Qsys システムに含み、信号を適宜に接続します。

**注意:** フォールト・インジェクション IP コアは、必ず EMR (Error Message Register) Unloader IP コアとともに使用してください。

フォールト・インジェクション IP コアと EMR Unloader IP コアは、Qsys および IP カタログで利用可能です。Verilog HDL, System Verilog, または VHDL を使用して、オプションでこれらの IP コアを RTL デザインに直接にインスタンスすることもできます。

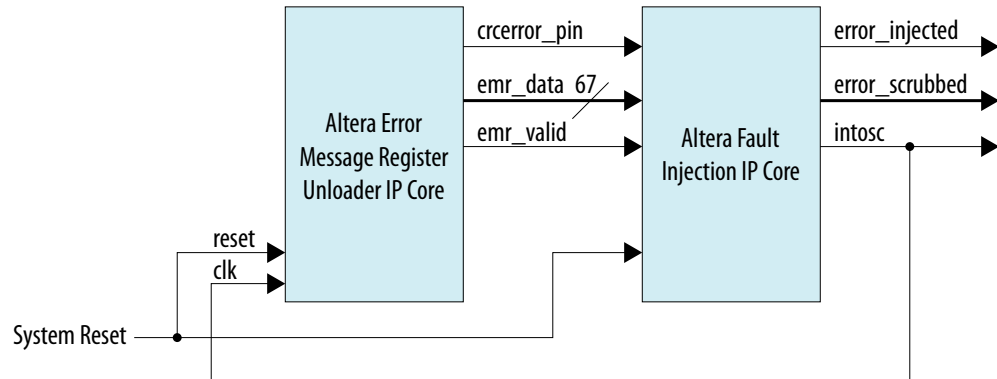
### 1.6.1 EMR Unloader IP コアを使用する

EMR Unloader IP コアは EMR へのインターフェイスを提供しています。EMR は、CRAM ビット CRC でソフトウェアエラーをチェックするデバイスの EDCRC によって継続的に更新されています。

図 -5: フォールト・インジェクション IP コアと EMR Unloader IP コアを含む Qsys システムの例

Use	Connections	Name	Description	Export
<input checked="" type="checkbox"/>		clock_bridge_0	Clock Bridge	
		in_clk	Clock Input	Double-click to export
		out_clk	Clock Output	clock_bridge_0_out_clk
<input checked="" type="checkbox"/>		reset_bridge_0	Reset Bridge	
		clk	Clock Input	Double-click to export
		in_reset	Reset Input	reset_bridge_0_in_reset
		out_reset	Reset Output	Double-click to export
<input checked="" type="checkbox"/>		emr_unloader_0	Altera Error Message Register Unloader	
		clock	Clock Input	Double-click to export
		reset	Reset Input	Double-click to export
		rcrcerror_pin	Conduit	Double-click to export
		rcrcerror	Conduit	emr_unloader_0_rcrcerror
		emr_read	Conduit	emr_unloader_0_emr_read
		avst_emr_src	Avalon Streaming Source	Double-click to export
<input checked="" type="checkbox"/>		fault_injection_0	Altera Fault Injection	
		rcrcerror_pin	Conduit	Double-click to export
		avst_emr_snk	Avalon Streaming Sink	Double-click to export
		reset	Reset Input	Double-click to export
		error_injected	Conduit	fault_injection_0_error_injected
		error_scrubbed	Conduit	fault_injection_0_error_scrubbed
		intosc	Clock Output	Double-click to export

図 -6: アルテラ・フォールト・インジェクション IP コアと EMR Unloader IP コアのブロック図の例



**関連情報**

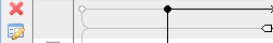


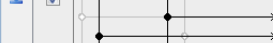
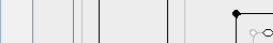


[Altera Error Message Unloader IP Core User Guide](#)

**1.6.2 Advanced SEU Detection IP コアを使用する**

ASD(Advanced SEU Detection)IP コアは、SEU 許容が設計上の懸念である際に使用できます。

EMR Unloader IP コアは ASD IP コアとともに使用する必要があります。そのため、同じデザインで ASD IP とフォールト・インジェクション IP を使用する場合は、IP が Avalon-ST スプリッター・コンポーネントを介して EMR Unloader の出力を共用する必要があります。以下の図は Avalon-ST スプリッターが EMR コンテンツを ASD IP コアとフォールト・インジェクション IP コアに分配する、Qsys システムを示しています。

図 -7: 同じ Qsys システムにおける ASD IP とフォールト・インジェクション IP の使用

Use	Connections	Name	Description	Export	Clock
<input checked="" type="checkbox"/>		<b>clock_bridge_intosc</b>	Clock Bridge	<i>Double-click to export</i>	<b>fault_injection_intosc</b>
		in_clk	Clock Input		clock_bridge_intosc_out_clk
		out_clk	Clock Output		intosc
<input checked="" type="checkbox"/>		<b>reset_bridge</b>	Reset Bridge	<i>Double-click to export</i>	<b>fault_injection_intosc</b>
		clk	Clock Input	<i>Double-click to export</i>	[clk]
		in_reset	Reset Input	<i>Double-click to export</i>	[clk]
		out_reset	Reset Output	<i>Double-click to export</i>	[clk]
<input checked="" type="checkbox"/>		<b>emr_unloader</b>	Altera Error Message Register Unloader	<i>Double-click to export</i>	<b>fault_injection_intosc</b>
		clock	Clock Input	<i>Double-click to export</i>	[clock]
		reset	Reset Input	<i>Double-click to export</i>	[clock]
		crerror_pin	Conduit	<i>Double-click to export</i>	crerror
		crerror	Conduit	<i>Double-click to export</i>	[clock]
		emr_read	Conduit	<i>Double-click to export</i>	emr_read
		emr_read	Conduit	<i>Double-click to export</i>	[clock]
		avst_emr_src	Avalon Streaming Source	<i>Double-click to export</i>	[clock]
<input checked="" type="checkbox"/>		<b>emr_splitter</b>	Avalon-ST Splitter	<i>Double-click to export</i>	<b>fault_injection_intosc</b>
		clk	Clock Input	<i>Double-click to export</i>	[clk]
		reset	Reset Input	<i>Double-click to export</i>	[clk]
		in	Avalon Streaming Sink	<i>Double-click to export</i>	[clk]
		out0	Avalon Streaming Source	<i>Double-click to export</i>	[clk]
		out1	Avalon Streaming Source	<i>Double-click to export</i>	[clk]
<input checked="" type="checkbox"/>		<b>fault_injection</b>	Altera Fault Injection	<i>Double-click to export</i>	<b>fault_injection_intosc</b>
		crerror_pin	Conduit	<i>Double-click to export</i>	crerror
		avst_emr_snk	Avalon Streaming Sink	<i>Double-click to export</i>	fault_injection_intosc
		reset	Reset Input	<i>Double-click to export</i>	[clk]
		error_injected	Conduit	<i>Double-click to export</i>	error_injected
		error_scrubbed	Conduit	<i>Double-click to export</i>	error_scrubbed
		intosc	Clock Output	<i>Double-click to export</i>	fault_injection_intosc
<input checked="" type="checkbox"/>		<b>reset_to_asd</b>	Reset Bridge	<i>Double-click to export</i>	<b>fault_injection_intosc</b>
		clk	Clock Input	<i>Double-click to export</i>	[clk]
		in_reset	Reset Input	<i>Double-click to export</i>	adv_seu_reset
		out_reset	Reset Output	<i>Double-click to export</i>	[clk]
<input checked="" type="checkbox"/>		<b>adv_seu</b>	Altera Advanced SEU Detection	<i>Double-click to export</i>	<b>fault_injection_intosc</b>
		clock	Clock Input	<i>Double-click to export</i>	[clock]
		reset	Reset Input	<i>Double-click to export</i>	[clock]
		cache_comparison_off	Conduit	<i>Double-click to export</i>	cache_comparison_off
		cache_comparison_off	Conduit	<i>Double-click to export</i>	[clock]
		avst_emr_snk	Avalon Streaming Sink	<i>Double-click to export</i>	[clock]
		avst_emr_snk	Avalon Streaming Sink	<i>Double-click to export</i>	[clock]
		asd_sp_master	Avalon Memory Mapped Master	<i>Double-click to export</i>	[clock]
		errors	Conduit	<i>Double-click to export</i>	errors
		errors	Conduit	<i>Double-click to export</i>	[clock]

## 関連情報

[Altera Advanced SEU Detection \(ALTERA\\_ADV\\_SEU\\_DETECTION\) IP Core User Guide](#)

## 1.7 機能の説明

アルテラ・フォールト・インジェクション IP コアによって、設計者はインハウスの SEFI 特性評価を実行し、SEFI 特性評価に応じて FIT レートを調整し、SEU の影響を軽減するためにデザインを最適化できます。

### 1.7.1 シングル・イベント・アップセット(SEU)の緩和

集積回路および FPGA などのプログラマブル・ロジック・デバイスは SEU に対する感受性が強いです。SEU は、アルファ粒子と宇宙線からの中性子という 2 つの主要な放射線源に起因する、ランダムで非破壊的な現象です。放射線は、論理レジスタ、エンベデッド・メモリー・ビット、または CRAM(コンフィグレーション RAM)ビットのいずれかの状態を反転させ、デバイスの予期しない動作を引き起こす可能性があります。

Arria V、Cyclone V、Stratix V および新しいデバイスは、以下の CRAM 機能を持っています。

- EDCRC(Error Detection Cyclical Redundance Checking)
- CRAM の SEU の自動的な訂正(スクラブ)
- CRAM の SEU 状態を作る機能(フォールト・インジェクション)

アルテラ デバイスにおける SEU の緩和について詳しくは、該当するデバイスのハンドブックで *SEU の緩和* の章を参照してください。

#### 関連情報

アルテラの Web サイト: [Single Event Upsets](#)

### 1.7.2 Fault Injection Debugger とフォールト・インジェクション IP コアを使用する

Fault Injection Debugger はフォールト・インジェクション IP コアと一緒に動作します。まず、デザインで IP コアをインスタンス化してコンパイルし、結果となるコンフィギュレーション・ファイルをデバイスにダウンロードします。次に、Quartus Prime ソフトウェアまたはコマンドラインから Fault Injection Debugger を実行して、ソフトエラーをシミュレーションします。

Fault Injection Debugger は、JTAG インターフェイスを介してフォールト・インジェクション IP コアと通信します。デバッグは、Quartus Prime ソフトウェアの Fault Injection Debugger またはコマンドライン・インターフェイスを使用して実行します。

- Fault Injection Debugger を使用することにより、フォールト・インジェクション実験を対話的にまたはバッチコマンドで実行できる。また、デザインでフォールト・インジェクションのための論理領域を指定できる
- コマンドライン・インターフェイスは、スクリプトを介してデバッグを実行するのに便利

フォールト・インジェクション IP は、JTAG インターフェイスからコマンドを受け取り、JTAG インターフェイスを介してステータスをレポートします。

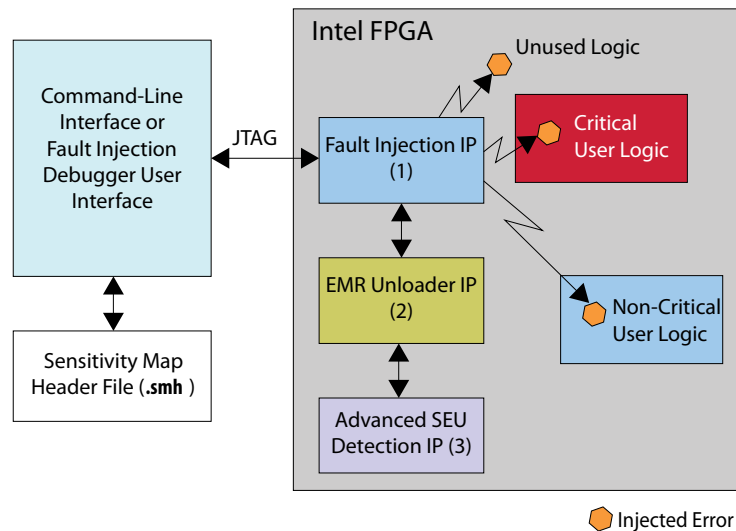
#### 注意:

フォールト・インジェクション IP コアは、デバイスのソフトロジックで実装されています。したがって、デザインでこのロジックの使用を考慮する必要があります。1 つの手法は、ラプでの SEU に対するデザインの応答を特性評価し、最終的に展開されたデザインから IP コアを省略することです。

フォールト・インジェクション IP コアは、以下の IP コアとともに使用します。

- アルテラ デバイスのハード化されたエラー検出回路からデータを読み取り、格納する EMR (Error Message Register) Unloader IP コア
- デバイス動作中にシングルビット・エラー位置を感度マップに比較して、ソフトエラーが動作に影響するかどうかを判断する、オプションの ASD (Advanced SEU Detection) IP コア

図 -8: Fault Injection Debugger の概要ブロック図



## 注意:

1. フォールト・インジェクションIPはターゲットロジックのビットを反転させます。
2. Fault Injection Debugger とAdvanced SEU Detection IP は、同じEMR Unloader インスタンスを使用します。
3. Advanced SEU Detection IP コアはオプションです。

## 関連情報

- [ダウンロードセンター](#)
- [AN 539: Test Methodology or Error Detection and Recovery using CRC in アルテラ FPGA Devices](#)
- [Understanding Single Event Functional Interrupts in FPGA Designs White Paper](#)
- [アルテラ・フォールト・インジェクション IP コアの利用ガイド](#)
- [Altera Error Message Unloader IP Core User Guide](#)
- [Altera Advanced SEU Detection \(ALTERA\\_ADV\\_SEU\\_DETECTION\) IP Core User Guide](#)

## 1.7.3 アルテラ・フォールト・インジェクション IP ピンの概要

アルテラ・フォールト・インジェクション IP コアは、以下の I/O ピンを備えています。

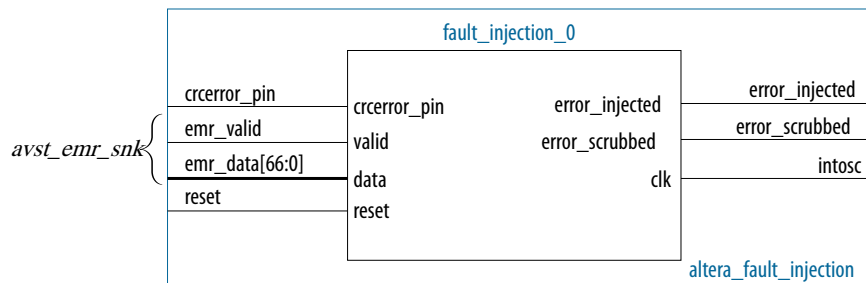
表 4. アルテラ・フォールト・インジェクション IP コアの I/O ピン

ピン名	ピンの方向	ピンの概要
crcerror_pin	入力	アルテラの Error Message Register Unloader IP からの入力です。この信号は、CRC エラーがデバイスの EDCRC によって検出された際にアサートされます。
emr_data	入力	EMR (Error Message Register) のコンテンツです。EMR フィールドについて、該当するデバイスのハンドブックを参照してください。

continued...

ピン名	ピンの方向	ピンの概要
		この入力、Avalon Streaming のデータ・インターフェイス信号に準拠しています。
emr_valid	入力	emr_data 入力に有効なデータが含まれていることを示します。これは Avalon Streaming の有効なインターフェイス信号です。
Reset	入力	モジュールのリセット入力です。
error_injected	出力	JTAG インターフェイスを介してコマンドされた通り、エラーが CRAM に注入されたことを示します。
error_scrubbed	出力	JTAG インターフェイスを介してコマンドされた通り、デバイスのスクラブが完了したことを示します。
intosc	出力	オプションの出力です。アルテラ・フォールト・インジェクション IP は、このクロックを使用して、例えば、EMR_unloader ブロックのクロックを供給します。

図 -9: アルテラ・フォールト・インジェクション IP ピン図



## 1.8 アルテラ・フォールト・インジェクション IP コアの利用ガイドのアーカイブ

バージョンが記載されていない場合には、以前のバージョン向けのマニュアルが当てはまります。

IP コアバージョン	ユーザーガイド
15.1	<a href="#">Altera Fault Injection IP Core User Guide</a>
15.0	<a href="#">Altera Fault Injection IP Core User Guide</a>

## 1.9 改訂履歴

表 5. 改訂履歴

日付	バージョン	変更内容
2016.10.31	16.1	デバイスのサポートを更新
2015.12.15	15.1	<ul style="list-style-type: none"> <li>Quartus II を Quartus Prime ソフトウェアに変更</li> <li>自己参照の関連リンクを修正</li> </ul>
2015.05.04	15.0	初版