

External Memory Interfaces Intel[®] Agilex[™] 7 F-Series and I-Series FPGA IP Design Example User Guide

Updated for Intel[®] Quartus[®] Prime Design Suite: 23.1

IP Version: 2.7.0



ID: 683162 Version: 2023.04.03

UG-20219

intel.

Contents

1. About the External Memory Interfaces Intel [®] Agilex [™] 7 F-Series and I-Series FPGA IP	3
1.1. Release Information	3
2. Design Example Quick Start Guide for External Memory Interfaces Intel Agilex [™] 7 F-Series and I-Series FPGA IP	4
 2.1. Creating an EMIF Project 2.2. Generating and Configuring the EMIF IP	4 7 7 8 11 13 14 14 14 16 17 17 17 18
3. Design Example Description for External Memory Interfaces Intel Agilex 7 F- Series and I-Series FPGA IP	. 23
3.1. Synthesis Design Example3.2. Simulation Design Example3.3. Example Designs Interface Tab	. 23 . 24 . 24
4. External Memory Interfaces Intel Agilex 7 F-Series and I-Series FPGA IP Design Example User Guide Archives	. 25
5. Document Revision History for External Memory Interfaces Intel Agilex 7 F-Series and I-Series FPGA IP Design Example User Guide	. 26





1. About the External Memory Interfaces Intel[®] Agilex[™] 7 F-Series and I-Series FPGA IP

1.1. Release Information

IP versions are the same as the Intel[®] Quartus[®] Prime Design Suite software versions up to v19.1. From Intel Quartus Prime Design Suite software version 19.2 or later, IP cores have a new IP versioning scheme.

The IP versioning scheme (X.Y.Z) number changes from one software version to another. A change in:

- X indicates a major revision of the IP. If you update your Intel Quartus Prime software, you must regenerate the IP.
- Y indicates the IP includes new features. Regenerate your IP to include these new features.
- Z indicates the IP includes minor changes. Regenerate your IP to include these changes.

Item	Description
IP Version	2.7.0
Intel Quartus Prime	23.1
Release Date	2023.04.03

Intel Corporation. All rights reserved. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Intel warrants performance of its FPGA and semiconductor products to current specifications in accordance with Intel's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Intel assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Intel. Intel customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services. *Other names and brands may be claimed as the property of others.



An automated design example flow is available for Intel Agilex[™] 7 F-Series and I-Series external memory interfaces.

The **Generate Example Designs** button on the **Example Designs** tab allows you to specify and generate the synthesis and simulation design example file sets which you can use to validate your EMIF IP.

You can generate a design example that matches the Intel FPGA development kit, or for any EMIF IP that you generate. You can use the design example to assist your evaluation, or as a starting point for your own system.

Figure 1. General Design Example Workflows



2.1. Creating an EMIF Project

For the Intel Quartus Prime software version 17.1 and later, you must create an Intel Quartus Prime project before generating the EMIF IP and design example.

1. Launch the Intel Quartus Prime software and select **File ➤ New Project Wizard**. Click **Next**.

Intel Corporation. All rights reserved. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Intel warrants performance of its FPGA and semiconductor products to current specifications in accordance with Intel's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Intel assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Intel. Intel customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services. *Other names and brands may be claimed as the property of others.





2. Specify a directory (*<user project directory>*), a name for the Intel Quartus Prime project (*<user project name>*), and a top-level design entity name (*<user top-level instance name>*) that you want to create. Click **Next**.

Ne	ew Project Wizard		
Directory, Name, Top-Level Entity	/		
What is the working directory for this project?			
user project directory			
What is the name of this <u>p</u> roject?			
user project name]
What is the name of the top-level design entity fo entity name in the design file.	or this project? This name is ca	se sensitive and mus	at exactly match the
user top level instance name			
This project uses a Partition Database (.qdb) fil	le for the root partition		

3. Verify that Empty Project is selected. Click Next two times.





New Project Wizard	—		×
Project Type			
Select the type of project to create.			
Empty project			
Create new project by specifying project files and libraries, target device tool settings.	family and devi	ce, and E	DA
O Project template			
Create a project from an existing design template. You can choose from with the Quartus Prime software, or download design templates from th	design templat ne <u>Design Store</u> .	es instal	led
< Back Next > Finish	Cancel	Hel	р

- 4. Under Family, select Intel Agilex 7.
- 5. Under Name filter, type the device part number.
- 6. Under **Available devices**, select the appropriate device.

	Board											
elect the	ne family and device y rmine the version of t	you want to he Intel Qu	target for compilat artus Prime softwar	ion. You ca e in which y	n install addition your target devic	al device su e is suppor	upport with the the ted, refer to the ted, refer to the ted ted ted ted ted ted ted ted ted te	he Instal he <u>Devic</u>	l Devices comman e Support List web	d on the To page.	ols menu.	
evice f	amily					Show in	'Available de	evices' lis	t			
<u>E</u> amil	y: Agilex (F-Series/I	-Series)			*							
Devic	e: All				*	Pac <u>k</u> a	ge:	Any				•
	- 6					Pin co	ount	Any				*
arget c	levice					Core	speed grade:	Any				*
O o	ther: n/a	•				IV S <u>⊦</u>	iow advance	d device:	5			
○ o	Name Name	• Tile	Core Voltage	ALMs	Total I/Os	GPIOs	ow advances HSSI Ch	d device: annels	Memory Bits	мгок	DSP Blocks	I/O P
ilter :	Name Name SFA012R24A1E1V	Tile	Core Voltage VID	ALMs 399500	Total I/Os 924	GPIOs 768	HSSI Ch	d device: annels	Memory Bits	M20K 5900	DSP Blocks 3743	I/O P 24
O Q ilter : AC	Name Name 5FA012R24A1E1V 5FA012R24A111V	 Tile P-Tile P-Tile 	Core Voltage VID VID	ALMs 399500 399500	Total I/Os 924 924	GPIOs 768 768	HSSI Ch 32 32	d device: annels	Memory Bits 120832000 120832000	M20K 5900 5900	DSP Blocks 3743 3743	I/O P 24 24
	Name Name 5FA012R24A1E1V 5FA012R24A1E1V 5FA012R24A2E2V	 Tile P-Tile P-Tile P-Tile 	Core Voltage VID VID VID	ALMs 399500 399500 399500	Total I/Os 924 924 924	GPIOs 768 768 768	HSSI Ch 32 32 32	d device: annels	Memory Bits 120832000 120832000 120832000	M20K 5900 5900 5900	DSP Blocks 3743 3743 3743	I/O P 24 24 24
0 0 ilter : AG 2 AG 3 AG	ther: n/a Name SFA012R24A1E1V SFA012R24A1E1V SFA012R24A2E2V SFA012R24A2E2V SFA012R24A2E3E	 Tile P-Tile P-Tile P-Tile P-Tile 	Core Voltage VID VID VID VID VID	ALMs 399500 399500 399500 399500	Total I/Os 924 924 924 924 924	GPIOs 768 768 768 768 768	HSSI Ch 32 32 32 32 32	d device: annels	Memory Bits 120832000 120832000 120832000 120832000 120832000	M20K 5900 5900 5900 5900	DSP Blocks 3743 3743 3743 3743 3743	I/O P 24 24 24 24 24
	ther: n/a Name SFA012R24A1E1V SFA012R24A1E1V SFA012R24A2E2V SFA012R24A2E3E SFA012R24A2E3V	 Tile P-Tile P-Tile P-Tile P-Tile P-Tile P-Tile 	Core Voltage VID VID VID VID VID VID	ALMs 399500 399500 399500 399500 399500	Total I/Os 924 924 924 924 924 924	GPIOs 768 768 768 768 768 768	HSSI Ch 32 32 32 32 32 32 32 32	d devices	Memory Bits 120632000 120832000 120832000 120832000 120832000 120832000	M20K 5900 5900 5900 5900 5900	DSP Blocks 3743 3743 3743 3743 3743 3743	I/O P 24 24 24 24 24 24 24
	Name Name SFA012R24A1E1V SFA012R24A1E1V SFA012R24A2E2V SFA012R24A2E3E SFA012R24A2E3V SFA012R24A2E3V SFA012R24A2E3V SFA012R24A2E3V SFA012R24A2E3V	 Tile P-Tile P-Tile P-Tile P-Tile P-Tile P-Tile P-Tile 	Core Voltage VID VID VID VID VID VID	ALMs 399500 399500 399500 399500 399500 399500	Total I/Os 924 924 924 924 924 924 924	GPIOs 768 768 768 768 768 768 768 768	HSSI Ch 32 32 32 32 32 32 32 32 32 32	d device:	Memory Bits 120832000 120832000 120832000 120832000 120832000 120832000 120832000	M20K 5900 5900 5900 5900 5900 5900	DSP Blocks 3743 3743 3743 3743 3743 3743 3743	I/O P 24 24 24 24 24 24 24 24
	ther: n/a Name Name FA012R24A1E1V SFA012R24A1E1V SFA012R24A2E3V SFA012R24A2E3V SFA012R24A2E3V SFA012R24A2E3V SFA012R24A2E3V SFA012R24A2E3E	Tile P-Tile P-Tile P-Tile P-Tile P-Tile P-Tile P-Tile P-Tile P-Tile	Core Voltage VID VID VID VID VID VID VID	ALMs 399500 399500 399500 399500 399500 399500 399500	Total I/Os 924 924 924 924 924 924 924 924 924	 ✓ 51 GPIOs 768 	HSSICh 32 32 32 32 32 32 32 32 32 32 32	d device:	Memory Bits 120832000 120832000 120832000 120832000 120832000 120832000 120832000	M20K 5900 5900 5900 5900 5900 5900 5900	DSP Blocks 3743 3743 3743 3743 3743 3743 3743 374	I/O P 24 24 24 24 24 24 24 24 24

7. Click Finish.





2.2. Generating and Configuring the EMIF IP

The following steps illustrate how to generate and configure the EMIF IP. This walkthrough creates a DDR4 interface, but the steps are similar for other protocols. (These steps follow the **IP Catalog** (standalone) flow; if you choose to use the **Platform Designer** (system) flow instead, the steps are similar.)

1. In the IP Catalog window, select External Memory Interfaces Intel Agilex 7 FPGA IP. (If the IP Catalog window is not visible, select View ➤ IP Catalog.)



2. In the **IP Parameter Editor**, provide an entity name for the EMIF IP (the name that you provide here becomes the file name for the IP) and specify a directory. Click **Create**.

4	New IP Variant	×
Look <u>I</u> n: 🗖	example	- A C BB -
dp 🗖		
File <u>N</u> ame:	<user instance="" name=""></user>	
Files of <u>Typ</u> e:	IP Variation Files (*.ip)	-
		Create Cancel

3. The parameter editor has multiple tabs where you must configure parameters to reflect your EMIF implementation.

2.2.1. Intel Agilex 7 F-Series and I-Series EMIF Parameter Editor Guidelines

This topic provides high-level guidance for parameterizing the tabs in the Intel Agilex 7 F-Series and I-Series EMIF IP parameter editor.



Parameter Editor Tab	Guidelines				
General	Ensure that the following parameters are entered correctly:The speed grade for the device.The memory clock frequency.The PLL reference clock frequency.				
Memory	 Refer to the data sheet for your memory device to enter the parameters on the Memory tab. You should also enter a specific location for the ALERT# pin. (Applies to DDR4 memory protocol only.) 				
Mem I/O	 For initial project investigations, you may use the default settings on the Mem I/O tab. For advanced design validation, you should perform board simulation to derive optimal termination settings. 				
FPGA I/O	 For initial project investigations, you may use the default settings on the FPGA I/O tab. For advanced design validation, you should perform board simulation with associated IBIS models to select appropriate I/O standards. 				
Mem Timing	 For initial project investigations, you may use the default settings on the Mem Timing tab. For advanced design validation, you should enter parameters according to your memory device's data sheet. 				
Controller	Set the controller parameters according to the desired configuration and behavior for your memory controller.				
Diagnostics	You can use the parameters on the Diagnostics tab to assist in testing and debugging your memory interface.				
Example Designs	The Example Designs tab lets you generate design examples for synthesis and for simulation. The generated design example is a complete EMIF system consisting of the EMIF IP and a driver that generates random traffic to validate the memory interface.				

EMIF Parameter Editor Guidelines Table 1.

For detailed information on individual parameters, refer to the appropriate chapter for your memory protocol in the External Memory Interfaces Intel Agilex 7 F-Series and I-Series FPGA IP User Guide

2.3. Generating the Synthesizable EMIF Design Example

For the Intel Agilex 7 F-Series and I-Series development kit, it is sufficient to leave most of the Intel Agilex 7 F-Series and I-Series EMIF IP settings at their default values. To generate the synthesizable design example, follow these steps:

1. On the **Example Designs** tab, ensure that the **Synthesis** box is checked.





 If you are implementing single interface example design, configure the EMIF IP and click **File> Save** to save the current setting into the user IP variation file (<user instance name>.ip).

Figure 2.

🔝 IP Paral	neter Editor Pro - ddr4* (/nfs/site/disks/psg_data_46/jmartinb/check/ddr4.ip) (on scfc12	2370
<u>File Edit System Generate View Tools H</u> elp	_	
New System Ctrl-N		
New IP Variant	- U	
New Component		
Select Quartus Project	Details	
Open Ctrl-O	Generate Example Design	Т
Save Ctrl-S		
New BSP	value is	
Save BSP As		
Export BSP as Tcl script		
Export System as Platform Designer script (.tcl)		
Save As		
Refresh System F5		
Refresh System and Reload All Components Ctrl-F5		
Synchronize IP File References		=
Archive System		
Restore Archived System	▼	
Browse Project Directory		
Recent Projects		
Exit Alt Ed	-	
Memory Clock Frequency: 1600.0	MHz	
Memory Davice Preset Selection		
Han Mamon Provide Preset Selection		

- If you are implementing an example design with multiple interfaces, specify **Number of IPs** to the desired number of interfaces. You can see the total number of **EMIF ID** same as the selected **Number of IPs**. Follow these steps to configure each interface:
 - a. Select the **Cal-IP** to specify the connection of the interface to the Calibration IP.
 - b. Configure the EMIF IP accordingly in all the Parameter Editor Tab.
 - c. Return to **Example Design** tab and click **Capture** on the desired **EMIF ID**.
 - d. Repeat step **a** to **c** for all **EMIF ID**.
 - e. You may click the **Clear** button to remove the captured parameters and repeat step **a** to **c** to make changes to the EMIF IP.
 - f. Click **File≻ Save** to save the current setting into the user IP variation file (<user instance name>.ip).

External Me	amony Interfaces Intel Agi	IN FOCA ID	Details
altera_emif_fm	inory interfaces inter Agi		Generate Example Design
Memory Prote	ocol		·
Protocol: DDR	4 🔻		
General Me	emory Mem I/O FPGA I/O M	Aem Timing Y Board Y Controller Y Diagnostics Y Example De	signs
* Example De	sign with Multi-IPs		
	🖌 Simulatio	on 🕑 Synthesis 🔲 Signal Integrity 🔲 Spyglass CDC S	imulation HDL format: Verilog 💌
-			
Number of IPs	² 4	·	_
These sectors	Online the ID and the dealers. Could	where the state in the second state state of the state of	
There are two Please select	Calibration IPs on the device. Each of for each EMIF IP, which of the two C.	of the EMIF IP must be connected to either of the two Cal-IPs. CAL-IPs should the EMIF IP be connected to.	
There are two Please select. 	Calibration IPs on the device. Each (for each EMIF IP, which of the two C	of the EMIF IP must be connected to either of the two Cal-IPs. CAL-IPs should the EMIF IP be connected to.	
There are two Please select, EMIF ID	Calibration IPs on the device. Each i for each EMIF IP, which of the two C.	of the EMF IP must be connected to either of the two Cal-IPs. 2AL-IPs should the EMF IP be connected to.	
There are two Please select EMIF ID EMIF #0	Calibration IPs on the device. Each i for each EMIF IP, which of the two C CAL-IP CaL-IP: CaLIP: CALIP_0	of the EMFIP must be connected to either of the two Cal-IPs. CAL-IPs should the EMFIP be connected to. ernal Memory Interfaces Intel [®]	[®] Agilex [™] 7 F-Series and I-Seri
There are two Please select EMIF ID EMIF #0 FMIF #1	Calibration IPs on the device. Each i for each EMF IP, which of the two C CAL-IP Cal-IP: CALIP_O	of the EMF IP must be connected to either of the two Cal-IPs. CAL-IPs should the EMF IP be connected to.	[®] Agilex [™] 7 F-Series and I-Seri Design Example
There are two Please select EMIF ID EMIF #0 EMIF #1	Calibration IPs on the device. Each : tor each EMIF IP, which of the two C CAL-IP CaL-IP: CALIP_0 CaLIP: CALIP_0 CaLIP: CALIP_0 CaLIP: CALIP_0 CaLIP.	of the EMFIP must be connected to either of the two Cal-IPs.	[®] Agilex [™] 7 F-Series and I-Seri Design Example I
There are two Please select, EMIF ID EMIF #0 EMIF #1 EMIF #2	Calibration IPs on the device. Each : for each EMF IP, which of the two C CALIP CaLIP: CALIP_0 CALIP: CALIP_0 CaLIP: CALIP_0 CaLIP: CALIP_0 CaLIP: CALIP_0 CALIP	of the EMF IP must be connected to either of the two Cal-IPs. AL-IPs should the EMF IP be connected to. Capture Clear Capture Clear Capture Clear	9 Agilex [™] 7 F-Series and I-Serie Design Example_I

Figure 3.

Send Feedback



2. Click **Generate Example Design** in the upper-right corner of the window.



3. Specify a directory for the EMIF design example and click **OK**. Successful generation of the EMIF design example creates the following fileset under a qii directory.

4	Select Example Design Directo	ory	×
Select	the directory for the example design.		
Path:	<design directory="" example=""></design>		
		Cancel	OK

- Click File ➤ Exit to exit the IP Parameter Editor Pro window. The system prompts, *Recent changes have not been generated. Generate now?* Click No to continue with the next flow.
- 5. To open the example design, click File ➤ Open Project, and navigate to the <project_directory>/<design_example_name>/qii/ed_synth.qpf and click Open.
 - *Note:* For information on compiling and programming the design example, refer to Compiling and Programming the Intel Agilex 7 EMIF Design Example.

Figure 4. Generated Synthesizable Design Example File Structure







For information on constructing a system with two or more external memory interfaces, refer to Creating a Design Example with Multiple EMIF Interfaces, in the *External Memory Interfaces Intel Agilex 7 F-Series and I-Series FPGA IP User Guide*.

For information on debugging multiple interfaces, refer to Enabling the EMIF Toolkit in an Existing Design, in the External Memory Interfaces Intel Agilex 7 F-Series and I-Series FPGA IP User Guide.

- *Note:* If you don't select the **Simulation** or **Synthesis** checkbox, the destination directory contains only Platform Designer design files, which are not compilable by the Intel Quartus Prime software directly, but which you can view or edit in the Platform Designer. In this situation you can run the following commands to generate synthesis and simulation file sets.
 - To create a compilable project, you must run the

quartus_sh -t make_qii_design.tcl
script in the destination directory.

• To create a simulation project, you must run the quartus_sh -t make_sim_design.tcl

script in the destination directory.

Note: If you have generated a design example and then make changes to it in the parameter editor, you must regenerate the design example to see your changes implemented. The newly generated design example does not overwrite the existing design example files.

2.4. Generating the EMIF Design Example for Simulation

For the Intel Agilex 7 F-Series and I-Series development kit, it is sufficient to leave most of the Intel Agilex 7 F-Series and I-Series EMIF IP settings at their default values. To generate the design example for simulation, follow these steps:

- 1. On the **Example Designs** tab, ensure that the **Simulation** box is checked. Also choose the required **Simulation HDL format**, either Verilog or VHDL.
- Configure the EMIF IP and click File ➤ Save to save the current setting into the user IP variation file (<user instance name>.ip).
- 3. Click **Generate Example Design** in the upper-right corner of the window.





A		IP Param	neter Editor Pro - ddr4* (/nfs/site/disks/psg_data_46/jmartinb/check/ddr4.ip) (on scfc12	2370
<u>F</u> ile	<u>E</u> dit <u>S</u> ystem <u>G</u> enerate <u>V</u> iew <u>T</u> ools	<u>H</u> elp		
Nev	v System	Ctrl-N		_
Nev	v IP Variant			
Nev	v Component			
Sele	ect Quartus Project		Details	
Ope	en	Ctrl-O	Generate Example Design	Т
Sav	e	Ctrl-S		
Nev	v BSP		value is	-
Sav	e BSP As			
Exp	ort BSP as Tcl script			
Exp	ort System as Platform Designer script (.tcl).			
Sav	e As			
Ref	resh System	F5		
Ref	resh System and Reload All Components	Ctrl-F5		
Syn	chronize IP File References			=
Arc	hive System			
Res	tore Archived System		-	
Bro	wse Project Directory		▼	
Rec	ent Projects	Þ		
E <u>x</u> it		Alt-F4		
	Memory Clock Frequency:	1600.0	MHz	
	Memory Device Preset Selection			
	Use Memory Device Preset from file:	alse		•

4. Specify a directory for the EMIF design example and click **OK**. Successful generation of the EMIF design example creates multiple file sets for various supported simulators, under a sim/ed_sim directory.

4	Select Example Design Directory	×
Select	the directory for the example design.	
Path:	<design directory="" example=""></design>	
	Cancel	ОК

5. Click **File** ➤ **Exit** to exit the **IP Parameter Editor Pro** window. The system prompts, *Recent changes have not been generated. Generate now?* Click **No** to continue with the next flow.



intel



Figure 5. Generated Simulation Design Example File Structure

- *Note:* The External Memory Interfaces Intel Agilex 7 F-Series and I-Series FPGA IP currently supports only the VCS, ModelSim/QuestaSim, and Xcelium simulators. Additional simulator support is planned in future releases.
- *Note:* If you don't select the **Simulation** or **Synthesis** checkbox, the destination directory contains only Platform Designer design files, which are not compilable by the Intel Quartus Prime software directly, but which you can view or edit in the Platform Designer. In this situation you can run the following commands to generate synthesis and simulation file sets.
 - To create a compilable project, you must run the

quartus_sh -t make_qii_design.tcl
script in the destination directory.

To create a simulation project, you must run the

quartus_sh -t make_sim_design.tcl
script in the destination directory.

Note: If you have generated a design example and then make changes to it in the parameter editor, you must regenerate the design example to see your changes implemented. The newly generated design example does not overwrite the existing design example files.

2.5. Simulation Versus Hardware Implementation

For external memory interface simulation, you can select either skip calibration or full calibration on the **Diagnostics** tab during IP generation.

EMIF Simulation Models

This table compares the characteristics of the skip calibration and full calibration models.



Table 2. EMIF Simulation Models: Skip Calibration versus Full Calibration

Skip Calibration	Full Calibration					
System-level simulation focusing on user logic.	Memory interface simulation focusing on calibration.					
Details of calibration are not captured.	Captures all stages of calibration.					
Has ability to store and retrieve data.	Includes leveling, per-bit deskew, etc.					
Represents accurate efficiency.						
Does not consid	der board skew.					

RTL Simulation Versus Hardware Implementation

This table highlights key differences between EMIF simulation and hardware implementation.

Table 3. EMIF RTL Simulation Versus Hardware Implementation

RTL Simulation	Hardware Implementation
Nios [®] initialization and calibration code execute in parallel.	Nios initialization and calibration code execute sequentially.
Interfaces assert cal_done signal simultaneously in simulation.	Fitter operations determine the order of calibration, and interfaces do not assert cal_done simultaneously.

You should run RTL simulations based on traffic patterns for your design's application. Note that RTL simulation does not model PCB trace delays which may cause a discrepancy in latency between RTL simulation and hardware implementation.

2.6. Simulating External Memory Interface IP With ModelSim

This procedure shows how to simulate the EMIF design example.

- Launch the Mentor Graphics* ModelSim software and select File ➤ Change Directory. Navigate to the sim/ed_sim/mentor directory within the generated design example folder.
- 2. Verify that the **Transcript** window is displayed at the bottom of the screen. If the **Transcript** window is not visible, display it by clicking **View ➤ Transcript**.
- 3. In the **Transcript** window, run **source msim_setup.tcl**.
- 4. After **source msim_setup.tcl** finishes running, run **ld_debug** in the **Transcript** window.
- 5. After **Id_debug** finishes running, verify that the **Objects** window is displayed. If the **Objects** window is not visible, display it by clicking **View ➤ Objects**.
- 6. In the **Objects** window, select the signals that you want to simulate by rightclicking and selecting **Add Wave**.
- 7. After you finish selecting the signals for simulation, execute **run -all** in the **Transcript** window. The simulation runs until it is completed.
- 8. If the simulation is not visible, click **View > Wave**.

2.7. Pin Placement for Intel Agilex 7 F-Series and I-Series EMIF IP

This topic provides guidelines for pin placement.





Overview

Intel Agilex 7 F-Series and I-Series FPGAs have the following structure:

- Each device contains up to 8 I/O banks.
- Each I/O bank contains 2 sub-I/O banks.
- Each sub-I/O bank contains 4 lanes.
- Each lane contains 12 general-purpose I/O (GPIO) pins.

General Pin Guidelines

The following are general pin guidelines.

- *Note:* For more detailed pin information, refer to the *Intel Agilex 7 FPGA EMIF IP Pin and Resource Planning* section in the protocol-specific chapter for your external memory protocol, in the External Memory Interfaces Intel Agilex 7 F-Series and I-Series FPGA IP User Guide.
 - Ensure that the pins for a given external memory interface reside within the same I/O row.
 - Interfaces that span multiple banks must meet the following requirements:
 - The banks must be adjacent to one another. For information on adjacent banks, refer to the EMIF Architecture: I/O Bank topic in the External Memory Interfaces Intel Agilex 7 F-Series and I-Series FPGA IP User Guide.
 - All address and command and associated pins must reside within a single subbank.
 - Address and command and data pins can share a sub-bank under the following conditions:
 - Address and command and data pins cannot share an I/O lane.
 - Only an unused I/O lane in the address and command bank can contain data pins.

Table 4.General Pin Constraints

Signal Type	Constraint
Data Strobe	All signals belonging to a DQ group must reside in the same I/O lane.
Data	Related DQ pins must reside in the same I/O lane. For protocols that do not support bidirectional data lines, read signals should be grouped separately from write signals.
Address and Command	Address and Command pins must reside in predefined locations within an I/O sub-bank.

Adjacent Banks

For banks to be considered adjacent, they must reside in the same I/O row. To determine if banks are adjacent, refer to the EMIF Architecture: I/O Bank topic in the *External Memory Interfaces Intel Agilex 7 F-Series and I-Series FPGA IP User Guide*.

Send Feedback



Pin Assignments

To determine locations for all EMIF I/O pins you should refer to the pin table for your device. When referring to the pin table, the bank numbers, I/O bank indices, and pin names are provided. You can find the pin indices for address and command pins in the Intel Agilex 7 EMIF Pin Table at the following location: https://www.intel.com/ content/www/us/en/programmable/support/literature/lit-dp.html.

You can perform pin assignments in a variety of ways. The recommended approach is to manually constrain some interface signals and let the Intel Quartus Prime Fitter handle the rest. This method consists of consulting the pin tables to find legal positions for some of the interface pins and assigning them through the .qsf file that is generated with the EMIF design example. For this method of I/O placement, you must constrain the following signals:

- CK0
- One DQS pin per group
- PLL reference clock
- RZQ

Based on the above constraints, the Intel Quartus Prime Fitter rotates pins within each lane as necessary.

2.8. Compiling and Programming the Intel Agilex 7 F-Series and I-Series EMIF Design Example

After you have made the necessary pin assignments in the .qsf file, you can compile the design example in the Intel Quartus Prime software.

- 1. Navigate to the Intel Quartus Prime folder containing the design example directory.
- 2. Open the Intel Quartus Prime project file, (.qpf).
- 3. To begin compilation, click **Processing > Start Compilation**. The successful completion of compilation generates a .sof file, which enables the design to run on hardware.
- 4. To program your device with the compiled design, open the programmer by clicking **Tools** > **Programmer**.
- 5. In the programmer, click **Auto Detect** to detect supported devices.
- 6. Select the Intel Agilex 7 device and then select Change File.
- 7. Navigate to the generated ed_synth.sof file and select **Open**.
- 8. Click Start to begin programming the Intel Agilex 7 device. When the device is successfully programmed, the progress bar at the top-right of the window should indicate 100% (Successful).

Refer to Using the Design Example with the EMIF Debug Toolkit for guidance in debugging your external memory interface.





2.9. Generating a Design Example with the Calibration Debug Option

You can use the External Memory Interface (EMIF) Debug Toolkit to assist in debugging your external memory interface. The toolkit provides access to data collected by the Nios II sequencer during memory calibration, and analysis tools to evaluate the stability of the calibrated interface.

The available task and analysis capabilities include the following:

- Requesting calibration of the memory interface. •
- Reading probe data or writing source data to the In-System Sources and Probes (ISSPs) instances in the design.
- Viewing the delay setting on any pin in the selected interface and changing it if necessary.
- Rerunning the traffic generator in the design example.
- Running driver margining on the interface. ٠
- Calibrating or changing termination settings. ٠
- 1. Navigate to the **Diagnostics** tab of the EMIF parameter editor.
- 2. Click Intel Quartus Prime EMIF Debug Toolkit/On-Chip Debug Port > Add **EMIF Debug Interface**.
- 3. If you are planning to use Driver Margining or ISSPs, ensure that the Enable In-System Sources and Probes option is enabled.
- 4. After you have completed parameterizing your interface, click Generate Example Design.

The resulting generated design example has the debug toolkit enabled and all the necessary components connected as required for a single interface.

2.10. Generating a Design Example with the TG Configuration Option

The generated EMIF design example includes a traffic generator block (TG). By default, the design example uses a simple TG block (altera tg avl) which can only be reset in order to relaunch a hard-coded traffic pattern. If necessary, you may choose to enable a configurable traffic generator (TG2) instead.

In the configurable traffic generator (TG2) (altera tg avl 2), you can configure the traffic pattern in real time through control registers—meaning that you do not have to recompile the design to change or relaunch the traffic pattern. This traffic generator provides fine control over the type of traffic that it sends on the EMIF control interface. Additionally, it provides status registers that contain detailed failure information.

2.10.1. Enabling the Traffic Generator in a Design Example

You can enable the configurable traffic generator from the **Diagnostics** tab in the EMIF parameter editor.

To enable the configurable traffic generator, turn on Use configurable Avalon traffic generator 2.0 on the Diagnostics tab.





Figure 6.

* Traffic Generator (settings only applicable for exar	nple design)	
✓ Use configurable Avalon traffic generator 2.0		
Enable default traffic pattern (pattern configured duri	ng compile-time)	
Enable user-configured traffic pattern (pattern configu	ired during run-time)	
TG2 default traffic duration:	Short	-
TG2 Configuration Interface Mode:	JTAG	-

- You may choose to disable the **default traffic pattern** stage or the **userconfigured traffic** stage, but you must have at least one stage enabled. For information on these stages, refer to Default Traffic Pattern and User-configured Traffic Pattern in the External Memory Interfaces Intel Agilex 7 F-Series and I-Series FPGA IP User Guide.
- The **TG2 test duration** parameter applies only to the default traffic pattern. You may choose a test duration of short, medium, or infinite.
- You may choose either of two values for the TG2 Configuration Interface Mode parameter:
 - JTAG: Allows use of a GUI in the system console. For more information, refer to Traffic Generator Configuration User Interface in the External Memory Interfaces Intel Agilex 7 F-Series and I-Series FPGA IP User Guide.
 - *Export*: Allows use of custom RTL logic to control the traffic pattern.

2.11. Using the Design Example with the EMIF Debug Toolkit

Before launching the EMIF Debug Toolkit, ensure that you have configured your device with a programming file that has the EMIF Debug Toolkit enabled. To launch the EMIF Debug Toolkit, follow these steps:

- 1. In the Intel Quartus Prime software, open the System Console by selecting **Tools** > System Debugging Tools > System Console.
- 2. [Skip this step if your project is already open in the Intel Quartus Prime software.] In the System Console, load the SRAM object file (.sof) with which you programmed the board (as described in Prerequisites for Using the EMIF Debug Toolkit, in the External Memory Interfaces Intel Agilex 7 F-Series and I-Series FPGA IP User Guide).
- 3. Select instances to debug.
- 4. Select EMIF Calibration Debug Toolkit for EMIF calibration debugging, as described in Generating a Design Example with the Calibration Debug Option. Alternatively, select EMIF TG Configuration Toolkit for traffic generator debugging, as described in *Generating a Design Example with the TG* Configuration Option.
- 5. Click **Open Toolkit** to open the main view of the EMIF Debug Toolkit.





Toolkit Explorer 🛛 System Explorer 🕮	– d =
>> Show all instances	
Instances	References
✓ № ed_synth.sof ✓ ♣ AGF FPGA Development Kit on pg ✓ ♣ AGFB014(F25AIR24AR0)@1	
🔻 🛄 ed_synth_inst	emif_cal_dbg_ag_2.3.0
t⊕= tg	emif_tg_cfg_1.0 3) Select Instance
Details	Collections
"EMIF Calibration Debug Toolkit" 4) Select Toolkit	No collections created 5) Open Toolkit
	Open Toolkit





Toolkit Explorer 🛛 System Explorer 😂	- đ 🗆
>> Show all instances	
Euau Design	
Instances	References
✓ Ge ed_synth.sol ✓ AGF FPGA Development Kit on pg	
→ AGFB014(F25A R24AR0)@1 → Milliod cumth inct	amit and dha na 220
	emif_tg_cfg_1.0
	3) Select Instance
Details	Collections
EMIF TG Configuration Toolkit	
(1) Salast Taalkit	1
4) Select Toolkit	No collections created
	5) Open Toolkit
	5) Open Toolkit

6. If there are multiple EMIF instances in the programmed design, select the column (path to JTAG master) and memory interface ID of the EMIF instance for which to activate the toolkit.



7. Click Activate Interface to allow the toolkit to read the interface parameters and calibration status.







8. You must debug one interface at a time; therefore, to connect to another interface in the design, you must first deactivate the current interface.

Welcome 😂 Colle	ction_1 🔀 Collection_2 🔀
ed_synth_inst	
Toolkit: "EMIF Calibra Systems: AGF FPGA I	ion Debug Toolkit"(<i>emif_cal_dbg_ag 2.3.0</i>) Development Kit on [USB-1]/AGFB014(F25A/R24AR0)@1[ed_synth_inst
De-Activate Interface	JTAG Master HPATH: 👩 synth instjernif callernif callitag master 🖓 Memory Interface ID: 🚺 🖉 Export Populated Tables

The following are examples of reports from the EMIF Calibration Debug Toolkit and the EMIF TG Configuration Toolkit:, respectively

Welcor	ne 🛛 Coll		_1 🕮 🕛	Collection_2	8						
ed_syr	nth_inst										
Toolki	t: "EMIF Calibra	ation D	ebug Tooli	kit" <i>(emif_ca</i>	_dbg_	r_ag 2.3.0) ger com intel com ILISB-11160	ER014/E25AIR	748P0\@illod_evn	th inct		
393emis, AGP PEGA Development At on pg-ennipger genotip.inter.com (OSB-1)/AGPB014(F25A)/C24AR0/@1]ed_synoinst											
De-A	De-Activate Interface JTAG Master HPATH: @d_synth_inst]emif_cal]emif_cal]itag_master Memory Interface ID: 0 Export Populated Tables							iles			
Men	nory Configurat	tion	Calibration	Calibratio	n Repor	Calibrate Termination	Vref Margining	Driver Margining	ISSP	Pin Delay Settings	
► C	DT Settings Ir	n Effec	t								
	Calibration Chai										
	alibration Sta	lus	D4 Devite O								_
	Check Parity	DD	R4 Parity S	tatus: 🔵							
P	arameter				Value	e					
er	ror_code				SUCCI	CESS					
er	ror_stage				NIL						
er	ror_group				0x000	000000					
<				11	1		>				





D ikit: "E MIF : AGF FPG/	TG Configura A Developmer	tion Toolkit" (é it Kit on pg-en	e <i>mif_tg_cfg 1.0</i> niffpga1.gar.co) rp.intel.com (USB-	1]JAGFB014((F25A R24AR0)	@1 ed_synth_	inst tg tg
Instruction I	Pattern Addr	ess Pattern	Data Pattern	TG Status Report	Configurati	ion and Status	Registers	
Export TG	Status Report	1						
	-							
IG Status:	•							
Number of a	Avalon Read (Commands is	sued: 196	Cle	ar			
						_		
Pass Not Fa	ail Signal			Cle	ar Failed Bits	в		
					0		(h 10	heat7
DQ PIN	beat0	beat1	beat2	beat3	beat4	beats	pearo	Deat
DQ PIN	beat0 pass	beat1	beat2 pass	pass	pass	pass	pass	pass
DQ PIN D	beat0 pass pass	beat1 pass pass	pass pass	pass pass	pass pass	pass pass	peats pass pass	pass pass
DQ PIN D I 2	beat0 pass pass pass	pass pass pass pass	pass pass pass pass	pass pass pass	pass pass pass	pass pass pass	pass pass pass pass	pass pass pass
DQ PIN D I 2 3	beat0 pass pass pass pass	beat1 pass pass pass pass	pass pass pass pass pass	pass pass pass pass pass	pass pass pass pass pass	pass pass pass pass pass	pass pass pass pass pass	pass pass pass pass
DQ PIN D 1 2 3 4	beat0 pass pass pass pass pass	pass pass pass pass pass pass	pass pass pass pass pass pass	pass pass pass pass pass pass	pass pass pass pass pass pass	pass pass pass pass pass pass	pass pass pass pass pass pass	pass pass pass pass pass pass
DQ PIN 0 1 2 3 4 5	beat0 pass pass pass pass pass pass	beat1 pass pass pass pass pass pass	pass pass pass pass pass pass pass pass	pass pass pass pass pass pass pass	pass pass pass pass pass pass pass	pass pass pass pass pass pass pass pass	pass pass pass pass pass pass pass	pass pass pass pass pass pass pass
DQ PIN 0 1 2 3 4 5 5 5	beat0 pass pass pass pass pass pass pass	beat1 pass pass pass pass pass pass pass	pass pass pass pass pass pass pass pass	pass pass pass pass pass pass pass pass	beat4 pass pass pass pass pass pass pass pas	pass pass pass pass pass pass pass pass	peats pass pass pass pass pass pass pass	pass pass pass pass pass pass pass pass
DQ PIN 0 1 2 3 4 5 6 7	beat0 pass pass pass pass pass pass pass pas	beat1 pass pass pass pass pass pass pass pas	beat2 pass pass pass pass pass pass pass pas	pass pass pass pass pass pass pass pass	pass pass pass pass pass pass pass pass	pass pass pass pass pass pass pass pass	peato pass pass pass pass pass pass pass pas	pass pass pass pass pass pass pass pass
DQ PIN 0 1 2 3 4 5 5 5 7 7 3	beat0 pass pass pass pass pass pass pass pas	beat1 pass pass pass pass pass pass pass pas	pass pass pass pass pass pass pass pass	pass pass pass pass pass pass pass pass	pass pass pass pass pass pass pass pass	pass pass pass pass pass pass pass pass	peato pass pass pass pass pass pass pass pas	pass pass pass pass pass pass pass pass
DQ PIN D 1 2 3 4 5 6 7 8 9	beat0 pass pass pass pass pass pass pass pas	beat1 pass pass pass pass pass pass pass pas	pass pass pass pass pass pass pass pass	pass pass pass pass pass pass pass pass	pass pass pass pass pass pass pass pass	pass pass pass pass pass pass pass pass	pass pass pass pass pass pass pass pass	pass pass pass pass pass pass pass pass
DQ PIN D 1 2 3 4 5 5 6 6 7 8 9 9 10	beat0 pass pass pass pass pass pass pass pas	pass pass pass pass pass pass pass pass	pass pass pass pass pass pass pass pass	peat3 pass pass pass pass pass pass pass pas	pass pass pass pass pass pass pass pass	pass pass pass pass pass pass pass pass	pass pass pass pass pass pass pass pass	pass pass pass pass pass pass pass pass
DQ PIN 0 1 2 3 4 5 6 6 7 7 8 8 9 10 11	beat0 pass pass pass pass pass pass pass pas	pass pass pass pass pass pass pass pass	beat2 pass pass pass pass pass pass pass pas	peat3 pass pass pass pass pass pass pass pas	beat4 pass pass pass pass pass pass pass pas	beab pass pass pass pass pass pass pass pa	Dearbo Dearbo pass pass pass pass pass pass pass pas	pass pass pass pass pass pass pass pass
DQ PIN 0 1 2 3 4 5 5 6 7 7 8 9 9 10 11 12	beat0 pass pass pass pass pass pass pass pas	beat1 pass pass pass pass pass pass pass pas	beat2 pass pass pass pass pass pass pass pas	peat3 pass pass pass pass pass pass pass pas	beat4 pass pass pass pass pass pass pass pas	pass pass pass pass pass pass pass pass	Dearbo Description pass pass pass pass pass pass pass pas	pass pass pass pass pass pass pass pass
DQ PIN 0 1 2 3 3 4 4 5 5 6 6 7 8 9 9 10 11 11 12 13	beat0 pass pass pass pass pass pass pass pas	beat1 pass pass pass pass pass pass pass pas	beat2 pass pass pass pass pass pass pass pas	peat3 pass pass pass pass pass pass pass pas	beat4 pass pass pass pass pass pass pass pas	pass pass pass pass pass pass pass pass	pass pass pass pass pass pass pass pass	pass pass pass pass pass pass pass pass
DQ PIN 0 1 2 3 3 4 4 5 6 6 7 7 8 8 9 9 10 11 12 13 14	beat0 pass pass pass pass pass pass pass pas	beat1 pass pass pass pass pass pass pass pas	beat2 pass pass pass pass pass pass pass pas	peat3 pass pass pass pass pass pass pass pas	beat4 pass pass pass pass pass pass pass pas	Deab Dass Dass Dass Dass Dass Dass Dass Da	DBAR DBS DBSS DBSS DBSS DBSS DBSS DBSS DBSS	pass pass pass pass pass pass pass pass
DQ PIN 0 1 2 3 4 5 5 6 6 7 8 9 10 11 12 13 14 15	beat0 pass pass pass pass pass pass pass pas	beat1 pass pass pass pass pass pass pass pas	beat2 pass pass pass pass pass pass pass pas	peat3 pass pass pass pass pass pass pass pas	beat4 pass pass pass pass pass pass pass pas	beab pass pass pass pass pass pass pass pa	DEARS DASS DASS DASS DASS DASS DASS DASS D	pass pass pass pass pass pass pass pass
DQ PIN 0 1 2 3 3 4 5 5 6 6 7 7 8 9 9 10 11 11 12 13 14 15 15	beat0 pass pass pass pass pass pass pass pas	beat1 pass pass pass pass pass pass pass pas	beat2 pass pass pass pass pass pass pass pas	peat3 pass pass pass pass pass pass pass pas	beat4 pass pass pass pass pass pass pass pas	pass pass pass pass pass pass pass pass	Dearbo Description pass pass pass pass pass pass pass pas	pass pass pass pass pass pass pass pass

- Note: For details on calibration debugging, refer to Debugging with the External Memory Interface Debug Toolkit, in the External Memory Interfaces Intel Agilex 7 F-Series and I-Series FPGA IP User Guide.
- Note: For details on traffic generator debugging, refer to Traffic Generator Configuration User Interface, in the External Memory Interfaces Intel Agilex 7 F-Series and I-Series FPGA IP User Guide.





3. Design Example Description for External Memory Interfaces Intel Agilex 7 F-Series and I-Series FPGA IP

When you parameterize and generate your EMIF IP, you can specify that the system create directories for simulation and synthesis file sets, and generate the file sets automatically.

If you select **Simulation** or **Synthesis** under **Example Design Files** on the **Example Designs** tab, the system creates a complete simulation file set or a complete synthesis file set, in accordance with your selection.

3.1. Synthesis Design Example

The synthesis design example contains the major blocks shown in the figure below.

- A traffic generator, which is a synthesizable Avalon[®]-MM example driver that implements a pseudo-random pattern of reads and writes to a parameterized number of addresses. The traffic generator also monitors the data read from the memory to ensure it matches the written data and asserts a failure otherwise.
- An instance of the memory interface, which includes:
 - A memory controller that moderates between the Avalon-MM interface and the AFI interface.
 - The PHY, which serves as an interface between the memory controller and external memory devices to perform read and write operations.

Figure 7. Synthesis Design Example



Synthesis Example Design

Intel Corporation. All rights reserved. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Intel warrants performance of its FPGA and semiconductor products to current specifications in accordance with Intel's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Intel assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Intel. Intel customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services. *Other names and brands may be claimed as the property of others.

ISO 9001:2015 Registered



If one or more of the PLL Sharing Mode, DLL Sharing Mode, or OCT Sharing Note: **Mode** parameters are set to any value other than **No Sharing**, the synthesis design example will contain two traffic generator/memory interface instances. The two traffic generator/memory interface instances are related only by shared PLL/DLL/OCT connections as defined by the parameter settings. The traffic generator/memory interface instances demonstrate how you can make such connections in your own designs.

3.2. Simulation Design Example

The simulation design example contains the major blocks shown in the following figure.

- An instance of the synthesis design example. As described in the previous section, the synthesis design example contains a traffic generator, calibration component, and an instance of the memory interface. These blocks default to abstract simulation models where appropriate for rapid simulation.
- A memory model, which acts as a generic model that adheres to the memory protocol specifications. Frequently, memory vendors provide simulation models for their specific memory components that you can download from their websites.
- A status checker, which monitors the status signals from the external memory interface IP and the traffic generator, to signal an overall pass or fail condition.



Figure 10. Simulation Design Example

3.3. Example Designs Interface Tab

The parameter editor includes an **Example Designs** tab which allows you to parameterize and generate your design examples.





4. External Memory Interfaces Intel Agilex 7 F-Series and I-Series FPGA IP Design Example User Guide Archives

IP versions are the same as the Intel Quartus Prime Design Suite software versions up to v19.1. From Intel Quartus Prime Design Suite software version 19.2 or later, IPs have a new IP versioning scheme.

For the latest and previous versions of this user guide, refer to https://www.intel.com/ content/www/us/en/docs/programmable/683162/. If an IP or software version is not listed, the user guide for the previous IP or software version applies.

Intel Corporation. All rights reserved. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Intel warrants performance of its FPGA and semiconductor products to current specifications in accordance with Intel's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Intel assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Intel. Intel customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services. *Other names and brands may be claimed as the property of others.

ISO 9001:2015 Registered

intel

5. Document Revision History for External Memory Interfaces Intel Agilex 7 F-Series and I-Series FPGA IP Design Example User Guide

Document Version	Intel Quartus Prime Version	IP Version	Changes
2023.04.03	23.1	2.7.0	• Updated product family name to Intel Agilex 7.
2021.06.21	21.2	2.4.2	 In the Design Example Quick Start chapter: Added a note to the Compiling and Programming the Intel Agilex EMIF Design Example topic. Modified the title of the Generating a Design Example with the Calibration Debug Option topic. Added the Generating a Design Example with the TG Configuration Option and Enabling the Traffic Generator in a Design Example topics. Modified steps 2, 3, and 4, updated several figures, and added a note, in the Using the Design Example with the EMIF Debug Toolkit topic.
2021.03.29	21.1	2.4.0	 In the Design Example Quick Start chapter: Added a note to the Generating the Synthesizable EMIF Design Example and Generating the EMIF Design Example for Simulation topics. Updated the File Structure diagram in the Generating the EMIF Design Example for Simulation topic.
2020.12.14	20.4	2.3.0	 In the Design Example Quick Start chapter, made the following changes: Updated the Generating the Synthesizable EMIF Design Example topic to include multi-EMIF designs. Updated the figure for step 3, in the Generating the EMIF Design Example for Simulation topic.
2020.10.05	20.3	2.3.0	 In the Design Example Quick Start Guide chapter, made the following changes: In Creating an EMIF Project, updated the image in step 6. In Generating the Synthesizable EMIF Design Example, updated the figure in step 3. In Generating the EMIF Design Example for Simulation, updated the figure in step 3. In Simulation Versus Hardware Implementation, corrected a minor typo in the second table. In Using the Design Example with the EMIF Debug Toolkit, modified step 6, added steps 7 and 8.

Intel Corporation. All rights reserved. Intel, the Intel logo, and other Intel marks are trademarks of Intel Corporation or its subsidiaries. Intel warrants performance of its FPGA and semiconductor products to current specifications in accordance with Intel's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Intel assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Intel. Intel customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services. *Other names and brands may be claimed as the property of others.

ISO 9001:2015 Registered



Document Version	Intel Quartus Prime Version	IP Version	Changes
2020.04.13	20.1	2.1.0	 In the About chapter, modified the table in the Release Information topic. In the Design Example Quick Start Guide chapter: Modified step 7 and the associated image, in the Generating the Synthesizable EMIF Design Example topic. Modified the Generating the Design Example with the Debug Option topic. Modified the Using the Design Example with the EMIF Debug Toolkit topic.
2019.12.16	19.4	2.0.0	 In the Design Example Quick Start chapter: Updated the illustration in step 6 of the Creating an EMIF Project topic. Updated the illustration in step 4 of the Generating the Synthesizable EMIF Design Example topic. Updated the illustration in step 4 of the Generating the EMIF Design Example for Simulation topic. Modified step 5 in the Generating the EMIF Design Example for Simulation topic. Modified the General Pin Guidelines and Adjacent Banks sections of the Pin Placement for Intel Agilex EMIF IP topic.
2019.10.18	19.3		 In the <i>Creating an EMIF Project</i> topic, updated the image with point 6. In the <i>Generating and Configuring the EMIF IP</i> topic, updated the figure with step 1. In the table in the <i>Intel Agilex EMIF Parameter Editor Guidelines</i> topic, changed the description for the <i>Board</i> tab. In the <i>Generating the Synthesizable EMIF Design Example</i> and <i>Generating the EMIF Design Example for Simulation</i> topics, updated the image in step 3 of each topic. In the <i>Generating the EMIF Design Example for Simulation</i> topic, updated the <i>Generated Simulation</i> topic, updated the <i>Generated Simulation</i> Design Example File Structure figure and modified the note following the figure. In the <i>Generating the Synthesizable EMIF Design Example</i> for <i>Simulation</i> topic, updated the <i>Generated Simulation Design Example File Structure</i> figure and modified the note following the figure.
2019.07.31	19.2	1.2.0	 Added About the External Memory Interfaces Intel Agilex FPGA IP chapter and Release Information. Updated dates and version numbers. Minor enhancement to the Synthesis Design Example figure in the Synthesis Design Example topic.
2019.04.02	19.1		Initial release.