

Allegro FPGA System Planner User Guide

Product Version 23.1

September 2023

Last Updated: January 2016

© 2023 Cadence Design Systems, Inc. All rights reserved.

Portions © Apache Software Foundation, Sun Microsystems, Free Software Foundation, Inc., Regents of the University of California, Massachusetts Institute of Technology, University of Florida. Used by permission. Printed in the United States of America.

Cadence Design Systems, Inc. (Cadence), 2655 Seely Ave., San Jose, CA 95134, USA.

Product Allegro FPGA System Planner contains technology licensed from, and copyrighted by: Apache Software Foundation, 1901 Munsey Drive Forest Hill, MD 21050, USA © 2000-2005, Apache Software Foundation. Sun Microsystems, 4150 Network Circle, Santa Clara, CA 95054 USA © 1994-2007, Sun Microsystems, Inc. Free Software Foundation, 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA © 1989, 1991, Free Software Foundation, Inc. Regents of the University of California, Sun Microsystems, Inc., Scriptics Corporation, © 2001, Regents of the University of California. Daniel Stenberg, © 1996 - 2006, Daniel Stenberg. UMFPACK © 2005, Timothy A. Davis, University of Florida, (davis@cise.ulv.edu). Ken Martin, Will Schroeder, Bill Lorensen © 1993-2002, Ken Martin, Will Schroeder, Bill Lorensen. Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, Massachusetts, USA © 2003, the Board of Trustees of Massachusetts Institute of Technology. vtkQt, © 2000-2005, Matthias Koenig. All rights reserved. Open SystemC, Open SystemC Initiative, OSCI, SystemC, and SystemC Initiative are trademarks or registered trademarks of Open SystemC Initiative, Inc. in the United States and other countries and are used with permission.

Trademarks: Trademarks and service marks of Cadence Design Systems, Inc. contained in this document are attributed to Cadence with the appropriate symbol. For queries regarding Cadence's trademarks, contact the corporate legal department at the address shown above or call 800.862.4522. All other trademarks are the property of their respective holders.

Restricted Permission: This publication is protected by copyright law and international treaties and contains trade secrets and proprietary information owned by Cadence. Unauthorized reproduction or distribution of this publication, or any portion of it, may result in civil and criminal penalties. Except as specified in this permission statement, this publication may not be copied, reproduced, modified, published, uploaded, posted, transmitted, or distributed in any way, without prior written permission from Cadence. Unless otherwise agreed to by Cadence in writing, this statement grants Cadence customers permission to print one (1) hard copy of this publication subject to the following conditions:

1. The publication may be used only in accordance with a written agreement between Cadence and its customer.
2. The publication may not be modified in any way.
3. Any authorized copy of the publication or portion thereof must include all original copyright, trademark, and other proprietary notices and this permission statement.
4. The information contained in this document cannot be used in the development of like products or software, whether for internal or external use, and shall not be used for the benefit of any other party, whether or not for consideration.

Disclaimer: Information in this publication is subject to change without notice and does not represent a commitment on the part of Cadence. Except as may be explicitly set forth in such agreement, Cadence does not make, and expressly disclaims, any representations or warranties as to the completeness, accuracy or usefulness of the information contained in this document. Cadence does not warrant that use of such information will not infringe any third party rights, nor does Cadence assume any liability for damages or costs of any kind that may result from use of such information. Cadence is committed to using respectful language in our code and communications. We are also active in the removal and/or replacement of inappropriate language from existing content. This product documentation may however contain material that is no longer considered appropriate but still reflects long-standing industry terminology. Such content will be addressed at a time when the related software can be updated without end-user impact.

Restricted Rights: Use, duplication, or disclosure by the Government is subject to restrictions as set forth in FAR52.227-14 and DFAR252.227-7013 et seq. or its successor

Contents

<u>Before you begin</u>	21
<u>Audience</u>	21
<u>Related Documentation</u>	21
 <u>1</u>	
<u>Overview to FPGA System Planner</u>	23
<u>FSP Work Flow</u>	26
<u>Work Flow Using FPGA System Planner</u>	27
<u>Advantages of the Workflow Using FSP</u>	27
 <u>2</u>	
<u>Getting Started with FPGA System Planner</u>	31
<u>Starting FPGA System Planner</u>	31
<u>The FPGA System Planner Start-Up Window</u>	32
<u>Open a Recent Project</u>	33
<u>Open an Existing Project</u>	33
<u>Create a New Project</u>	33
<u>Open a Sample Design</u>	33
<u>The FPGA System Planner Workspace</u>	34
<u>The Libraries window</u>	36
<u>The Messages window</u>	41
<u>The TCL Command Bar</u>	46
<u>The Pin Legend Window</u>	48
<u>The Design Board Canvas</u>	50
<u>The Canvas Zoom Preview</u>	80
<u>The Design Connectivity window</u>	81
<u>The Status Bar</u>	118
<u>The Properties Window</u>	118
<u>The Die View</u>	128
<u>The Power Connections Window</u>	132

<u>Menu Bar</u>	133
<u>Toolbars</u>	144
<u>Customizing Toolbars</u>	145
<u>Undoing and Redoing Changes</u>	147
<u>Configuring FPGA System Planner</u>	153
<u>FSP User Interface</u>	153
<u>Config.ini File</u>	153
 3	
Managing Environment Variables and Template Files	157
<u>The FSP Template Files</u>	158
<u>Customizing Template Files for a Site</u>	159
<u>Customizing Config.ini file for a Local Directory</u>	160
<u>Default Search Order of Template Files</u>	161
<u>Rules File Search Mechanism</u>	161
<u>Merging Attributes of Config.ini file</u>	162
<u>Modifying Template File Attributes</u>	163
<u>Assigning Footprint Models</u>	163
<u>Locating Footprints in Install Directory</u>	164
 4	
Project Creation and Setup	167
<u>Creating Design Projects</u>	168
<u>Creating a New Design Project</u>	168
<u>Creating a Project from a cpm file</u>	169
<u>Project Files</u>	171
<u>Setting up the Project</u>	174
<u>Setting up the Search Path to the Rules File</u>	175
<u>Updating Configuration File</u>	176
<u>Using the Rules File Path Editor</u>	176
<u>Setting up the Search Path to the Footprint File</u>	178
<u>Locating Sample Footprints in Install Directory</u>	180
<u>Setting Footprint Variables at Project Level</u>	181
<u>Setting Footprint Variables at Site-level</u>	182
<u>Setting Footprint Variables in a Local env File</u>	182

Allegro FPGA System Planner User Guide

<u>Setting up the Preference</u>	184
<u>Opening a Project</u>	185
<u>Selecting Schematic Environment</u>	186
<u>Saving a Project</u>	186
<u>Closing a Project</u>	187
<u>Archiving a Project</u>	187
<u>FSP Install Directory</u>	188

5

<u>Working with Libraries</u>	191
<u>Library Overview</u>	191
<u>Library Structure</u>	191
<u>Library Files</u>	192
<u>FPGA Rules File (.frf)</u>	192
<u>Interface Rules File (.lrf)</u>	193
<u>Adding Interface Rules File</u>	194
<u>Points to Remember</u>	194

6

<u>Working with Components</u>	197
<u>Understanding Component Placement</u>	198
<u>Creating FSP Design Using Real Components</u>	198
<u>Creating FSP Design Using Rules File or Virtual Components</u>	199
<u>Understanding Logical Mapping</u>	200
<u>The Logical Mapping File</u>	200
<u>Sample Mapping Files Location</u>	201
<u>Mapping Considerations</u>	202
<u>Mapping Scenarios</u>	203
<u>Creating Mapping File</u>	211
<u>Setting Up PTF for Component Selection</u>	216
<u>Adding Interface Component</u>	219
<u>Adding Interface Component (DEHDL)</u>	219
<u>Using Create>Select Component Rules and Mapping Information Wizard</u>	219
<u>Using Add Part dialog box</u>	244
<u>Adding Interface Component (OrCAD)</u>	255

Allegro FPGA System Planner User Guide

<u>Using Create>Select Component Rules and Mapping Information Wizard</u>	255
<u>Using Add Part dialog box</u>	263
<u>Adding Interface Component (Libraries)</u>	263
<u>Replacing Logical Reference Model of the Component</u>	264
<u>DEHDL</u>	265
<u>OrCAD</u>	269
<u>Re-referencing Logical and Mapping of the Instantiated Component</u>	270
<u>OrCAD</u>	271
<u>Converting Components to Real Components</u>	271
<u>DE-HDL</u>	272
<u>OrCAD</u>	273
<u>Converting Virtual Interface to Real Components</u>	274
<u>Viewing Components</u>	275
<u>Modifying Components</u>	276
<u>Adding Device Component</u>	277
<u>DE-HDL</u>	277
<u>OrCAD</u>	280
<u>Libraries</u>	280
<u>Linking Device Component to Front-End Symbol</u>	281
<u>DE-HDL</u>	282
<u>OrCAD</u>	282
<u>Replacing Device</u>	283
 <u>7</u>	
<u>Preparing Components for Connection</u>	285
<u>Overview</u>	286
<u>Modifying Instance Name Prefix</u>	286
<u>Creating Interface Protocol (Interface to Device Connection)</u>	288
<u>Creating Interface Protocol (Single Interface to Multiple Devices/Connectors)</u>	291
<u>Creating Device Protocol (Device to Device Connection)</u>	294
<u>The Edit Protocol Editor</u>	295
<u>Using Existing Rules File</u>	296
<u>Using Existing Protocol</u>	299
<u>Using Constraints/Pinouts File</u>	299
<u>Using CSV File</u>	301

<u>Creating Deep and Wide Connections</u>	304
<u>Creating Deep and Wide Common Group for Homogenous interfaces</u>	305
<u>Creating Deep and Wide Common Group for Heterogeneous interfaces</u>	308
<u>Creating Deep and Wide Buses for Homogenous interfaces</u>	308
<u>Creating System Ace Device Chain</u>	310
<u>Creating Daisy Chain Connections</u>	311
<u>Defining a Target Set for Connectors</u>	311
<u>Defining Connectivity for Target Sets</u>	313
<u>Establishing connectivity between target Sets and DUT</u>	313
<u>Re-Optimizing Connectivity</u>	314

8

<u>Working with Nets and Ports</u>	317
<u>About the Nets and Connectivity</u>	318
<u>About Bus Name</u>	318
<u>Net Naming Conventions</u>	318
<u>Assigning Net Name Template</u>	319
<u>Assigning Net Name Template on Interface Protocol</u>	320
<u>Specifying Net Names Manually</u>	322
<u>Resetting Net Names</u>	324
<u>Modifying Net Names</u>	324
<u>Re-routing Nets</u>	326
<u>Deleting Nets</u>	327
<u>Locking and Unlocking Nets</u>	327
<u>Using the Pop-up menu options in the canvas</u>	328
<u>Using the Pop-up menu option in the Design Connectivity</u>	328
<u>Using the Lock Nets dialog box</u>	330
<u>Showing and Hiding Nets</u>	332
<u>Using NetGroups</u>	332
<u>Naming NetGroups</u>	333
<u>Using RTL Port Names</u>	334
<u>Specifying Port Names for FSP Nets</u>	334

9

<u>Running Synthesis</u>	337
<u>Running Synthesis for Complete Design</u>	338
<u>Incremental Synthesis</u>	338
<u>Configuring and Monitoring Synthesis</u>	339
<u>Applying Synthesis Options</u>	340
<u>Applying Advanced Synthesis Options</u>	341
<u>Monitoring Synthesis Status</u>	342
<u>Saving and Importing Synthesis Details</u>	342
<u>Saving Synthesis Details</u>	343
<u>Importing Synthesis Details</u>	343
<u>Match Length for Group Nets</u>	344
<u>Smallest Fit Bank</u>	346
<u>Sorted Groups (Farthest First)</u>	347
<u>Sorted Groups (Nearest First)</u>	348

10

<u>Working with Power Regulators</u>	351
<u>Overview to the Power connections</u>	351
<u>Points to Remember Before Adding Power Regulators</u>	354
<u>Adding and Deleting Power Regulators</u>	354
<u>Adding Power Regulators from Schematic Symbols</u>	356
<u>Defining Power Mapping</u>	356
<u>Mapping Voltage Value</u>	357
<u>Mapping Power Pins</u>	358
<u>Adding Power Regulators Automatically</u>	360
<u>Mapping Power Regulators to Power Pins</u>	361
<u>Removing Power Regulators</u>	362
<u>Resetting and Mapping Power Regulators</u>	363
<u>Exporting Power Mapping Information</u>	363
<u>Importing Power Mapping Information</u>	364

11

<u>Working with Associated Components</u>	367
<u>Terminations</u>	368
<u>Termination Types</u>	368
<u>Mapping Terminations Ports</u>	372
<u>Defining Series Termination For Single Ended Signals</u>	373
<u>Defining Series Termination for Differential Ended Signals</u>	375
<u>Adding Differential Termination to Differential Ended Signals</u>	376
<u>Adding Pull Up Termination to Single and Differential Ended Signals</u>	378
<u>Modifying a Termination</u>	378
<u>Deleting a Termination</u>	378
<u>Applying Termination to Instance Pins</u>	379
<u>Applying Series Termination to Single Ended Pins (Interface/Device)</u>	379
<u>Applying Series Termination to Differential Pair Signals (Interface/Device)</u>	380
<u>Quick Options for Applying Terminations</u>	380
<u>Power Filters</u>	381
<u>Defining Power Filters</u>	382
<u>Applying Power Filters to Power Pins</u>	384
<u>Applying Power Filters to Instance Pins</u>	385
<u>Defining Decoupling Capacitors</u>	385
<u>Adding Decoupling Capacitor</u>	386
<u>Modifying Decoupling Capacitors</u>	389
<u>Defining External Connections</u>	389
<u>Applying an External Port</u>	391
<u>Defining External Connections for Virtual Interface</u>	392

12

<u>Exporting and Updating Design</u>	393
<u>Exporting Part and Pin Properties</u>	394
<u>The Export CSV Window</u>	394
<u>Exporting Views</u>	397
<u>Editing the External File</u>	398
<u>Importing Part and Pin Properties</u>	401
<u>The Update From CSV window</u>	401

<u>Importing Changed Properties</u>	406
13	
<u>Schematic in Design Entry HDL</u>	409
<u>Preparing the Design for Schematic</u>	410
<u>Enabling Name Space Checking</u>	411
<u>Name Space Checking at Various Stages</u>	411
<u>Setting Up Symbols</u>	416
<u>Specifying Directory Path to Generate Symbols</u>	416
<u>Customizing Schematic Symbols</u>	417
<u>Cross-Probing between Tree View and Graphics View</u>	418
<u>Generating DE HDL Symbols</u>	422
<u>Mapping Attributes Names to Properties</u>	423
<u>Assigning “No Connect” Power Pins</u>	425
<u>Steps to Generate DE-HDL Symbols</u>	425
<u>Enabling and Disabling DE-HDL Symbol Generation Options</u>	426
<u>Creating Split Symbols for Design Block</u>	427
<u>Points to Remember</u>	427
<u>Preserving the Split Symbol Changes</u>	427
<u>Cross-probing between Tree view and Graphics view of the Design Block Symbol Editor</u>	428
<u>Creating Split Symbols</u>	428
<u>Merging Split Symbols</u>	430
<u>Miscellaneous Operations</u>	430
<u>Working with Full Symbol</u>	431
<u>Generating DE HDL Schematics</u>	433
<u>Overview to Generating DE-HDL Schematics</u>	434
<u>Preparing the FSP Design to Generate the Schematics</u>	436
<u>Using CPM Directives</u>	438
<u>Reading Directives from CPM Files at Different Levels</u>	439
<u>Steps for Generating DE-HDL Schematics</u>	444
<u>Understanding DE-HDL Schematic Generation</u>	446
<u>Placements of Components</u>	448
<u>Capturing Connectivity in the DE-HDL Schematic</u>	450
<u>Representing Associated Components in the Schematic</u>	450

<u>Generating Schematic in Preserve Mode</u>	452
<u>Preserving Schematic in the Front and Back flow</u>	452
<u>General Features for Schematic Generation</u>	454
<u>Mapping Schematic Attributes to Properties</u>	455
<u>Files Created after Generating Schematics</u>	458
 14	
<u>Schematic in OrCAD Capture</u>	461
<u>Preparing the Design for Schematic</u>	462
<u>Enabling Name Space Checking</u>	462
<u>Setting Up Symbols</u>	462
<u>Specifying Directory Path to Generate Symbols</u>	462
<u>Customizing Symbols</u>	463
<u>Generating OrCAD Symbols</u>	464
<u>Steps to Generate OrCAD Symbols</u>	465
<u>Generating OrCAD Schematics</u>	467
<u>Understanding OrCAD Schematic generation</u>	467
<u>Create Top Level Design</u>	467
<u>Place Termination blocks in separate page</u>	468
<u>Generate / Specify FPGA symbol Location</u>	468
<u>Skip Unused Splits</u>	468
<u>Specify FPGA Symbol Location</u>	468
 15	
<u>Exporting the Logical Design to a Board</u>	471
<u>Overview</u>	472
<u>FPGA System Planner to Allegro PCB Editor Flow</u>	472
<u>Creating a Board Outline for Logical Design</u>	474
<u>Generating Board File and Placement Data</u>	476
<u>Updating the Board with the Changes in Logical Design</u>	477
<u>Updating the Logical Design with the Changes in the Board</u>	480
<u>Points to Remember Before Updating the Logical Design</u>	481
<u>Importing an FSP/Non-FSP Initiated Board</u>	483

16

<u>Generating Output</u>	485
<u>Generating Constraint Files</u>	485
<u>Generating Verilog Design Files</u>	486
<u>Generating Plan Ahead scripts</u>	486
<u>Generating CSV Files</u>	488
<u>Viewing Outputs and Their Location in Explorer</u>	488
<u>Allegro DE HDL Symbols</u>	488
<u>Allegro DE HDL Schematics</u>	489
<u>Allegro DE CIS Symbols</u>	489
<u>Allegro DE CIS Schematics</u>	489
<u>Constraint Files</u>	490
<u>Allegro PCB Placement</u>	490

17

<u>Synchronizing With FPGA Tools</u>	491
<u>Overview</u>	491
<u>Net Name Convention</u>	491
<u>FPGA Port Names</u>	492
<u>Design Flow starts with FSP</u>	492
<u>Design Flow does not start with FSP</u>	493
<u>Net Names</u>	494
<u>Defining FPGA Port Names for FSP Nets</u>	495
<u>Defining FPGA Port Names for Interface Signals</u>	495
<u>Defining FPGA Port Names for Protocol Signals</u>	495
<u>Defining FPGA Port Names for Virtual Interface Signals</u>	496
<u>Mapping Resources</u>	497
<u>About Resource Mapping Window</u>	497
<u>Rules for Mapping Resources</u>	498
<u>Steps to map the resources.</u>	498
<u>Mapping FPGA Port Names and Use Pins</u>	502
<u>Understanding different Mapping Port names and Use Pins Scenarios</u>	502
<u>Step by Step Instruction for Port Mapping</u>	502
<u>Mapping Port Names and Use Pins with Existing Net Names</u>	502

Allegro FPGA System Planner User Guide

<u>Mapping Port Names and Use Pins with Pin Names</u>	507
<u>Importing Constraints from an External File</u>	508
<u>About Constraint File</u>	509
<u>Step by Step Instruction for Importing Constraints</u>	509
<u>Importing Details</u>	511
<u>Points to Remember when Importing Constraint Files</u>	511
<u>Importing Incomplete Constraints Files</u>	512
<u>Optimizing with Constrained Settings</u>	513
<u>Constrained Optimization</u>	513
<u>Points to Remember when working with Use Pin</u>	513
<u>Start Optimizing Design</u>	513
<u>Exporting Constraints</u>	517
<u>Exporting All Constraints</u>	517
<u>Exporting Partial Constraints</u>	518
<u>Swapping Groups</u>	520
<u>Overview of Swapping Groups</u>	520
<u>Points to Remember Before Swap Groups</u>	521
<u>Swapping Groups Within an Interface</u>	521
<u>Swapping Groups Between Two Interfaces</u>	524

18

<u>Configuring Devices</u>	529
<u>Xilinx Devices</u>	529
<u>Types of Configuration Schemes</u>	529
<u>Select Map mode</u>	530
<u>Define PROM Chain</u>	531
<u>Build and Edit a Prom Chain</u>	532
<u>Define JTAG chain</u>	534
<u>Altera Devices</u>	536

19

<u>Managing Model and Design Versions</u>	537
<u>Overview</u>	537
<u>Comparing Different Versions</u>	537
<u>Managing v16.5 (or lower version) Design/Model in v16.6</u>	538

Allegro FPGA System Planner User Guide

<u>Opening v16.5 (or lower version) Design in v16.6</u>	538
<u>Editing v16.5 (or lower version) model in v16.6</u>	538
<u>Managing v16.6 Design/Model in v16.5</u>	538
<u>Opening v16.6 (or higher version) design in v16.5</u>	539
<u>Opening v16.6 design in lower version</u>	539
<u>Editing v16.6 Model in v16.5 or lower version</u>	539
<u>Migrating a v16.5 Design to v16.6</u>	540
<u>Opening a v16.5 Design in v16.6</u>	540
<u>Using TCL Command</u>	542
<u>Limitations of the Uprev Process</u>	542
<u>Migrating a Pre-16.6 Design to Current Version</u>	543
<u>Opening a Pre 16.6 Design</u>	545
<u>Using TCL Command</u>	546

20

<u>Creating Parts</u>	547
<u>Guidelines for Creating Parts</u>	548
<u>Using Rules Editor</u>	550
<u>Invoking Rules Editor</u>	550
<u>Creating Parts from DE-HDL Symbol</u>	551
<u>Understanding Symmetrical Symbol</u>	551
<u>Steps to Create Part from Symmetrical Symbol</u>	554
<u>Steps to Create Part from Asymmetrical Symbol</u>	557
<u>Creating Parts from OrCAD Symbol</u>	562
<u>Creating Part using Custom Footprint</u>	564
<u>Points to Remember While Creating the Part Using Custom Footprint</u>	564
<u>Creating Parts from Existing Interface Rules File</u>	568
<u>Creating Parts from External Files</u>	573
<u>Comma Separated File (CSV) Import Example</u>	574
<u>Points to Remember While Importing CSV File</u>	575
<u>Creating Parts Manually</u>	580
<u>Invoking Rules Editor</u>	580
<u>Adding Properties</u>	581
<u>Adding Signal</u>	582
<u>Adding Logical Group</u>	584

<u>Adding Pin Details</u>	585
<u>AutoMapping Diff Pair Pins</u>	586
<u>Adding Custom Attributes</u>	588
<u>Rearranging Pin Order</u>	589
<u>Editing Options</u>	590
<u>Verifying the Part</u>	592
<u>Saving the Part</u>	592
<u>Editing Parts</u>	594
<u>Editing the Logical Properties</u>	594
<u>Editing the Group Constraints</u>	594
<u>Editing the Pin Constraints</u>	597
<u>Updating Pin and Group Constraints from External File</u>	601

21

<u>Creating Virtual Interfaces</u>	603
<u>Overview to Virtual Interface</u>	604
<u>Creating Single Virtual Interface</u>	606
<u>Using Existing LRF</u>	606
<u>Using Constraints/Pinout File</u>	606
<u>Creating Multiple Virtual Interfaces (Using Verilog/VHDL file)</u>	610
<u>Editing Options</u>	612

22

<u>Creating Connectors</u>	615
<u>Types of Connectors</u>	616
<u>Creating Connector</u>	617
<u>Creating Connector Manually</u>	617
<u>Invoking Rules Editor</u>	617
<u>Adding Properties</u>	618
<u>Adding Signal</u>	619
<u>Adding Bank</u>	621
<u>Adding Pin Details</u>	622
<u>AutoMapping Diff Pair Pins</u>	623
<u>Defining Patterns for Connector Pins</u>	624
<u>Adding Custom Attributes</u>	631

Allegro FPGA System Planner User Guide

<u>Rearranging Pin Order</u>	632
<u>Editing Connector</u>	633
<u>Saving the Connector Model</u>	633
<u>Creating Testing Connector</u>	635
<u>Creating Testing Connector Manually</u>	636
 23	
<u>Automation in FSP</u>	641
<u>Working with FSP TCL</u>	642
<u>Using the TCL Command window</u>	642
<u>Executing a TCL Script</u>	643
 24	
<u>Dialog Box Descriptions</u>	645
<u>Add Bus for Groups</u>	646
<u>Add Bus for Bank</u>	647
<u>Add Group</u>	648
<u>Add Bank</u>	649
<u>Auto Detect Pin Pairs</u>	650
<u>Add Part (DE-HDL)</u>	654
<u>Add Part (OrCAD)</u>	656
<u>Auto Map FPGA Ports</u>	658
<u>Using Regular Expression</u>	658
<u>Auto Map Symbol Pins</u>	661
<u>Add Description</u>	663
<u>Add Regulator from Schematic Symbol</u>	664
<u>BUFR Nets for Device Instance</u>	665
<u>Bank Settings for Device Instance</u>	666
<u>Column Chooser</u>	667
<u>Create Part from PCB Footprint</u>	669
<u>Convert Rules File Instance to Real Part (DE-HDL)</u>	670
<u>Convert Rules File Instance to Real Part (OrCAD)</u>	671
<u>Changing Schematic Symbol File Reference(OrCAD)</u>	672
<u>Creating Part From Custom Footprint</u>	673
<u>Change Product Choices</u>	677

Allegro FPGA System Planner User Guide

<u>Modifying the Default License</u>	677
<u>Change Net Names</u>	679
<u>Custom Attribute</u>	681
<u>Create New Protocol</u>	682
<u>Create Virtual Interface(s) From Verilog/VHDL for Device Instance</u>	683
<u>Change FPGA</u>	685
<u>Create New Project</u>	686
<u>Connector Net Name and Pin Assignment Mapping for Instance</u>	688
<u>Filtering the Display of Entries</u>	690
<u>Create Target Sets</u>	692
<u>DCI Master Slave Banks</u>	693
<u>Define Termination</u>	694
<u>Deep and Wide Groups</u>	696
<u>Define Termination Mapping</u>	698
<u>Define PROM Chain</u>	700
<u>Define JTAG Chain</u>	702
<u>Define Virtual Interface for Device Instance</u>	703
<u>Define Decoupling Capacitors</u>	707
<u>Define Decoupling Capacitor Symbol</u>	708
<u>Design Comparison</u>	709
<u>Design Block Full Symbol Editor</u>	712
<u>Design Block Split Symbol Editor</u>	713
<u>Edit Rules (for OrCAD)</u>	714
<u>Edit Rules (for DE-HDL)</u>	716
<u>Edit Clock Group</u>	717
<u>Edit Group</u>	718
<u>Edit Bank</u>	719
<u>Edit Properties</u>	720
<u>Edit Interface Protocol</u>	721
<u>Edit Protocol</u>	723
<u>Export Design Constraints</u>	729
<u>Edit Symbol Settings</u>	730
<u>Export CSV</u>	731
<u>FPGA Port and Use Pin Mapping for Device Instance</u>	732
<u>Generate Allegro DE-HDL Symbols</u>	734
<u>Generate Allegro DE-HDL Schematics</u>	736

Allegro FPGA System Planner User Guide

<u>Hierarchy</u>	736
<u>Placement</u>	736
<u>General</u>	737
<u>Advanced Settings</u>	738
<u>Symbols tab</u>	738
<u>Power Tab</u>	740
<u>Placement Tab</u>	741
<u>Routing tab</u>	742
<u>Colors tab</u>	743
<u>Properties tab</u>	744
<u>Custom Attribute Tab</u>	744
<u>Generate OrCAD Symbols</u>	746
<u>Generate OrCAD Schematics</u>	748
<u>Group Settings For Interface Instance <inst_name></u>	751
<u>Generate/Update Layout Placement Data</u>	753
<u>Generate SCM Design</u>	754
<u>Import Signals from OrCAD Symbol</u>	756
<u>Import Verilog/VHDL Signals for Device Instance</u>	757
<u>Instance Symbol Editor</u>	759
<u>Import Constraint For Device Instance</u>	762
<u>Import from CSV</u>	763
<u>Lock Nets</u>	765
<u>Configure Multi-Voltage Pins</u>	767
<u>Multiple Device Connection Groups</u>	768
<u>Map Clock Groups</u>	769
<u>Map FPGA Resources For Device Instance</u>	770
<u>Manage Protocols</u>	771
<u>Net Removal Confirmation</u>	772
<u>Power Connections</u>	773
<u>Process Option Editor</u>	775
<u>Advanced</u>	781
<u>Properties</u>	783
<u>Preferences</u>	798
<u>Display Tab</u>	798
<u>Design tab</u>	800
<u>General Tab</u>	801

Allegro FPGA System Planner User Guide

<u>Remove Net Confirmation</u>	804
<u>Resolve Net Name Conflict</u>	805
<u>Rules File Path Editor</u>	807
<u>Rules Editor</u>	808
<u>Cross-Probing between Tree View (Editor) and Graphics View</u>	808
<u>Rules Viewer</u>	815
<u>Rules Instance Editor</u>	816
<u>Rules Signal Mapper</u>	817
<u>Select Instance</u>	819
<u>Select Rules File</u>	820
<u>Save Rules File</u>	821
<u>Swap Groups</u>	822
<u>Select Font</u>	823
<u>Settings</u>	824
<u>Project tab</u>	824
<u>Net Name Template tab</u>	826
<u>Design Clocks tab</u>	827
<u>Xilinx tab</u>	827
<u>Altera</u>	827
<u>NetGroup</u>	828
<u>Read Me tab</u>	829
<u>System ACE Dialog</u>	830
<u>Specify DE-HDL Symbols To Generate</u>	831
<u>Specify OrCAD Symbol To Generate</u>	832
<u>Setup OrCAD Symbols Data</u>	833
<u>Setup DE-HDL Symbols Data</u>	834
<u>Update Design from Allegro Board</u>	835
<u>Update Pin Locations from PCB Footprint</u>	836
<u>Workflow</u>	837
<u>What do you want to do....?</u>	841

Allegro FPGA System Planner User Guide

Before you begin

FPGA System Planner (FSP) is a FPGA-PCB co-design pin assignment tool. With FSP, you can synthesis the design based on user-specified, interface-based connectivity, FPGA pin assignment rules, and actual placement of FPGAs on PCB.

FSP is fully integrated with a number of different tool suites, such as OrCAD Capture CIS, Design Entry HDL, and Allegro PCB Editor. FSP tool is available only on Windows platform.

Audience

This user guide assumes that you are familiar with the FPGAs and FPGAs pin assignment rules, and have basic knowledge of development and design of printed circuit boards at the board level. This user guide is intended for the users having working knowledge of the following Cadence tools:

- Allegro Design Entry HDL
- Allegro PCB Editor
- OrCAD Capture CIS

Related Documentation

In addition to this guide, you can find technical product information on the Cadence website. The following table describes the types of technical documentation provided with FSP.

Document Name	Provide this...
FSP TCL Reference Guide	Concise descriptions including examples of each of the TCL commands available in FSP.
DE-HDL - FSP Flow Guide	Provides you comprehensive information for understanding the flow between Allegro FPGA System Planner, Allegro DE-HDL, and Allegro PCB Editor.
OrCAD - FSP Flow Guide	Provides you comprehensive information for understanding the flow between Allegro FPGA System Planner, OrCAD CaptureCIS, and Allegro PCB Editor.

Allegro FPGA System Planner User Guide

Before you begin

Overview to FPGA System Planner

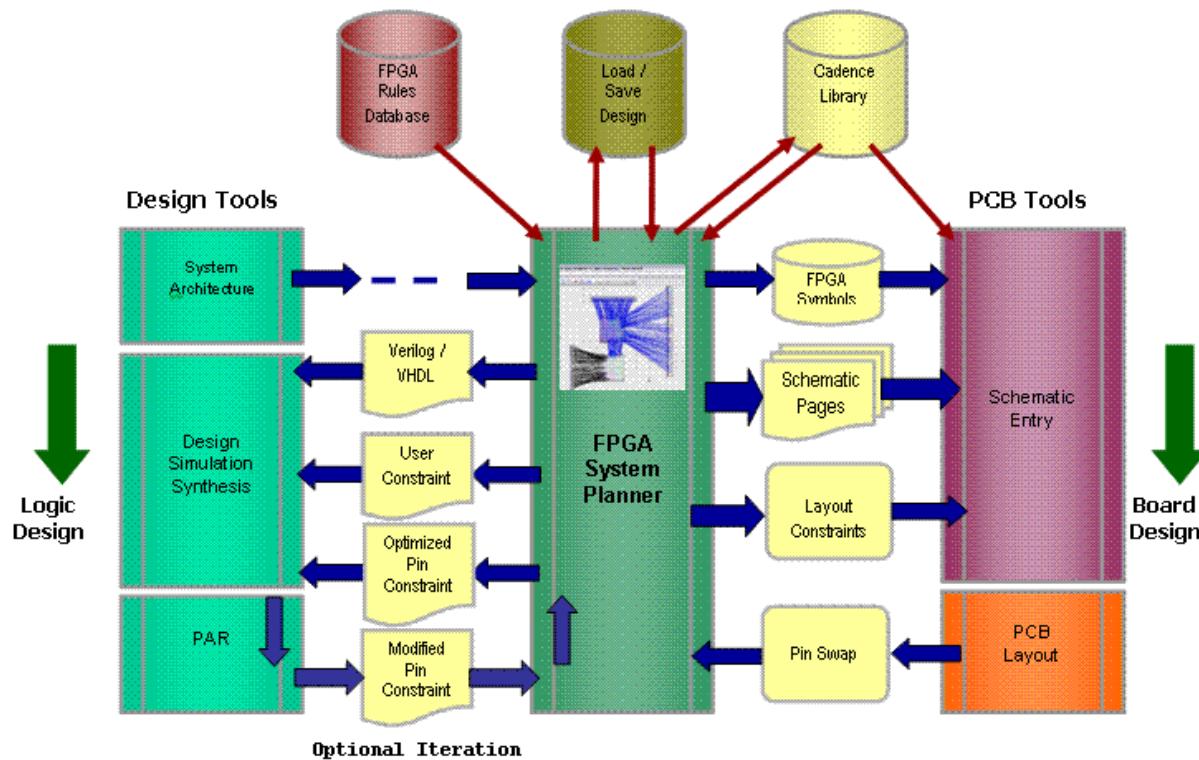
Designing a PC board with the latest FPGAs, containing more than 2000 configurable I/O pins, is quite complex if it is done manually. Completing a single FPGA project requires several rounds of communication between the logic designer, the system designer, and the PC board designers. Yet most of this process is actually routine and even redundant. It is just the FPGAs that make the whole process complex, because of their numerous, multidimensional I/O rules. These problems can be addressed through FPGA System Planner by automating all of these complex processes. The result is hassle-free and quick PCB designing for FPGA-based boards.

FSP fully automates the processes of assigning FPGA pins to other components, generating schematics, and interconnecting layouts—all while meeting the following constraints:

- Logical Constraints - The pin-out should satisfy the requirements for the protocol underlying the interface. For example, for a source-synchronous bus to capture data successfully, both the data and the corresponding clocks must be pinned out correctly.
- Electrical Constraints - Electrical constraints are related to FPGA I/O DRCs. FPGAs have complex banking structures and a detailed set of associated rules. The electrical signaling standards of the interface determine if it can use a particular bank.
- Physical Constraints - Physical constraints are related to how different devices are placed on the board. Pins should be selected to minimize the wire crossings and to reduce the number of layers required to route the board.

Allegro FPGA System Planner User Guide

Overview to FPGA System Planner



FPGA System Planner has the following features that make it unique. FPGA System Planner:

- Uses its rule-based engine and selects FPGA pins automatically in accordance with I/O DRC standards.
- Generates symbols and complete schematics, which can be supported for Allegro Design Entry HDL and OrCAD Capture.
- Completes power connections in accord with pin I/O standards, and it generates them as nets or symbol properties.
- Minimizes signal crossings to reduce the PCB layer count, thereby improving Signal Integrity (SI).
- Easily splits or merges project active, as well as inactive projects, so a team of designers can work systematically.
- Helps designers create and edit models based on frequently changing requirements.
- Presents a detailed log of messages so the designers know the exact status of the various actions performed.
- Lets designers view a library of models from various families.

Allegro FPGA System Planner User Guide

Overview to FPGA System Planner

- Creates custom parts.
- Creates custom business rules.
- Reduces crossovers, resulting in fewer vias and fewer PCB layers.
- Shortens design cycles.
- Reduces ECOs.
- Cuts down design, testing, manufacturing, and end-product costs remarkably.
- Improves product reliability.

FSP Work Flow

This topic compares the work flow of a traditional FPGA-based board design with that of FSP.

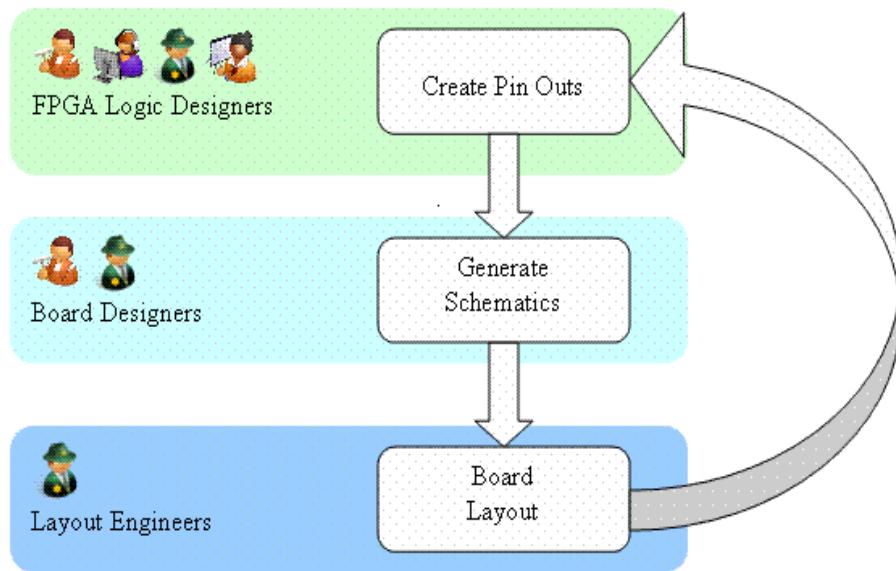


Figure 1-1 The traditional FPGA-Based Board Design Flow

- FPGA Logic Designers understand the rules and constraints for pinning out the device and completing pin selection. The data is transferred to Board Designers.
- Board Designers use this data for generating the symbols and schematics. The Board Designers need to ensure that the FPGA is connected to the appropriate voltage levels, since voltage connections depend upon the logic that is connected to different FPGA banks. The voltage level change if the logic connections changes. The generated schematics are then passed to Layout Engineers.
- Layout Engineers start designing the layout based on these schematics.
- Board Designers consult the Logic Designers if the pin-outs require changes to simplify the rats-nest of connections and make the design routable.

In this existing flow, there is a lot of manual work for Logic Designers, Schematic Designers, and Board Designers. Moreover, work flow proceeds by trial and error, taking several months to achieve the expected design.

Work Flow Using FPGA System Planner

This section describes how FPGA System Planner makes the same process of designing FPGA - based boards simple and hassle free.

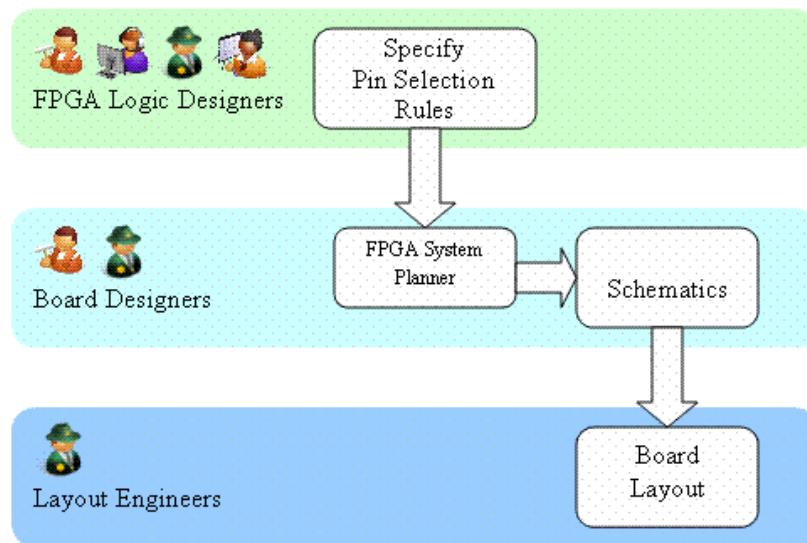


Figure 1-2 The work flow of designing a FPGA based PCB using FSP.

- The Logic Designer specifies the higher-level rules for connecting the pins of the various interfaces.
- FSP process the rules, specified by the Logic Designer, to automatically generate the pin-outs for the FPGA, the FPGA symbols, and the schematics.
- The generated schematics and symbols are read by schematic design tools to complete the rest of the board - design process.

Using FSP makes the symbols and schematic generation faster and easier, reducing cycle time from months to weeks.

Advantages of the Workflow Using FSP

FSP is used to you accelerate the process of designing an FPGA board. It is built ground up to work in an incremental mode, meaning that you can add more interfaces to an FPGA in your design while preserving the existing connections. It also offers complete flexibility in optimizing I/O pin selection to meet your timing and layout constraints.

Allegro FPGA System Planner User Guide

Overview to FPGA System Planner

Before selecting the pins, FSP runs through various optimization phases. It then displays a rats-nest view of the connections. At this stage, this interactive output has a huge influence on the results.

There are numerous advantages for PCB designers to use FSP instead of manual PCB designing. Indeed, FSP simplifies the following tasks for the three key designers:

- FPGA Logic Designers, by creating pin-outs
- Board Designers, by generating schematics
- Layout Designers, by generating layouts

Following are the advantages of using FSP

- Leverages a system-level understanding for a FPGA solution
FSP is the first tool to bring system-level understanding to FPGA based system design.
- Improves the productivity of board designers

With the FSP rule-based synthesis engine, designers save the enormous effort wasted on the cumbersome tasks of FPGA pin selection obeying DRC. With just minimal attention to board design, designers using FSP can improve the logic of the design, with greatly improved productivity as a result.

- Create symbols and generate schematics using existing libraries and create symbols for the missing ones

Before generating the schematics, FSP checks the symbol data for each part used in the design and it flags a warning if any symbols are missing in the library database. This prompts the user to generate the symbol data for the missing symbols and to regenerate symbols for the existing ones.

- Minimizes crossovers to reduce the PCB layer count and improve Signal Integrity (SI)

FSP automatically detects the most suitable pin assignment to effectively reduce the crossovers. This ultimately results in fewer PCB layers and improved Signal Integrity. As the number of board layers reduces, the cost also reduces radically.

- Optimizes usage of pins

Optimum usage of pins results in more space available on the board.

- Supports large number of standard components

FSP supports a large number of standard components that you can select and place on the board.

Allegro FPGA System Planner User Guide

Overview to FPGA System Planner

- Creates custom interface for components that are not supported

If a particular component is not supported, FSP provides a simple user interface to create the customized component interface that is needed.

- Develops multiple designs for a single project

With FSP, you can try out four to five different pin placements and decide on the best design.

- Reduces the design cycle from months to weeks

The process of automatically detecting and selecting components and their pin placements drastically reduces the overall time for the entire design cycle.

- Easily integrates with the existing design flow

FSP reads the existing logical, physical, and electrical constraints, and then generates the design with the most suitable pin connections.

- Improves collaboration between designers by using split and merge design features

When there are multiple FPGAs and other components, FSP helps split the design based on logic. Once the design is split, designers can work on their respective parts; later, they can merge them while maintaining the same constraints. This team work helps the team of designers to collaborate during the design cycles of a large project.

Allegro FPGA System Planner User Guide

Overview to FPGA System Planner

Getting Started with FPGA System Planner

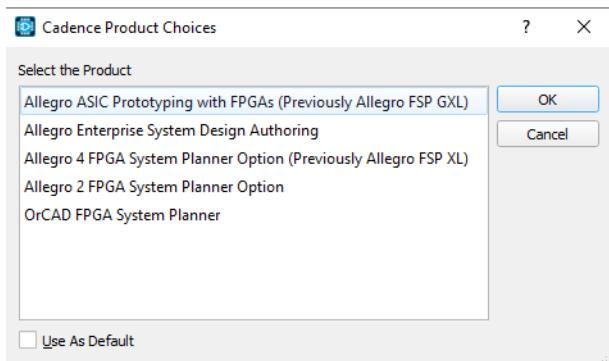
Starting FPGA System Planner

To launch FSP on Windows platform, do one of the following:

1. Using the Windows Start menu: Choose *Start – Cadence PCB <release version>– FPGA System Planner <release version>*.

Alternatively, type `fpgasysplanner` in the windows command prompt:

2. In the Cadence Product Choices window, select one of the available licenses and click OK.

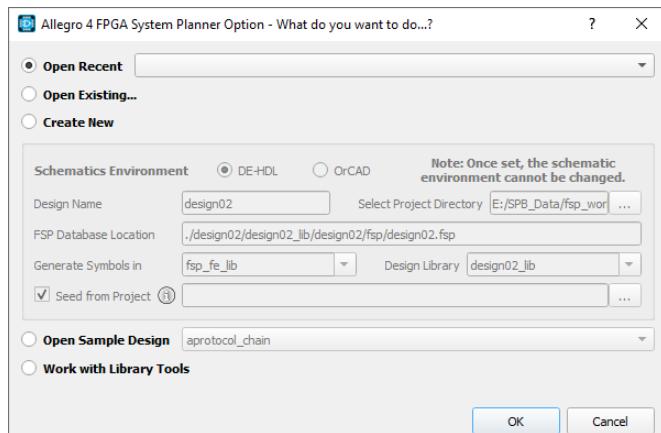


The Cadence Product Choices lists all available license from which you can invoke FSP.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

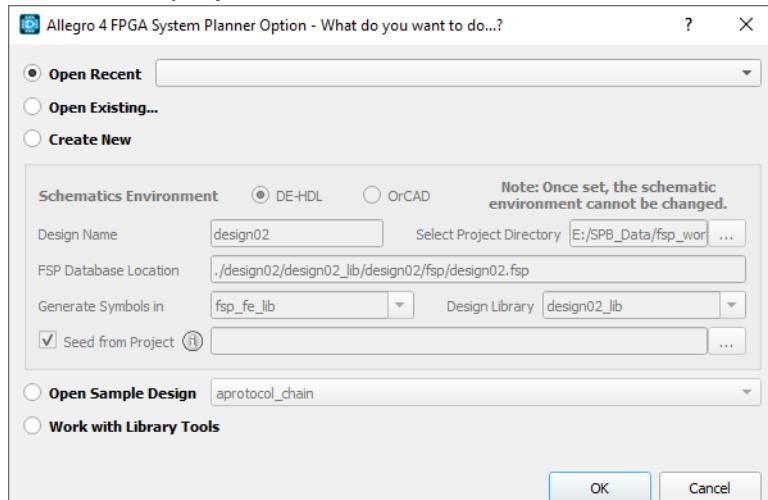
The FSP start-up window is displayed.



Note: The number of FPGAs to be used in a design depends on the selected license. For instance, you cannot open a project with lower license, if the project is previously created with a higher license such as Allegro 4 FPGA System Planner license or higher. When you try to open a project, an error message is displayed in the Messages window.

The FPGA System Planner Start-Up Window

The start-up window is displayed each time you launch FSP. This window includes quick options to open and create projects.



Use this window to do open a recent or existing project, to create a new project, open a sample design, or to work with library tools.

Open a Recent Project

To open a recent project, do the following:

- Click the drop-down list and select an entry from the list.
- Click *OK*.

Open an Existing Project

To open an existing project, do the following:

- Select the *Open Existing* option and click *OK*.
The Open Project window is displayed.
- Select the project file (.fsp) and click *Open*.
The project is opened in FSP.

Create a New Project

To create a new project, do the following:

1. Select the *Create New* option.
2. Specify the project name in the *Name* field.
3. Specify the path in the *Path* field or click *Browse* to browse to the directory where you want to save the project.
4. Select the schematic environment.
5. Click *OK*.

Open a Sample Design

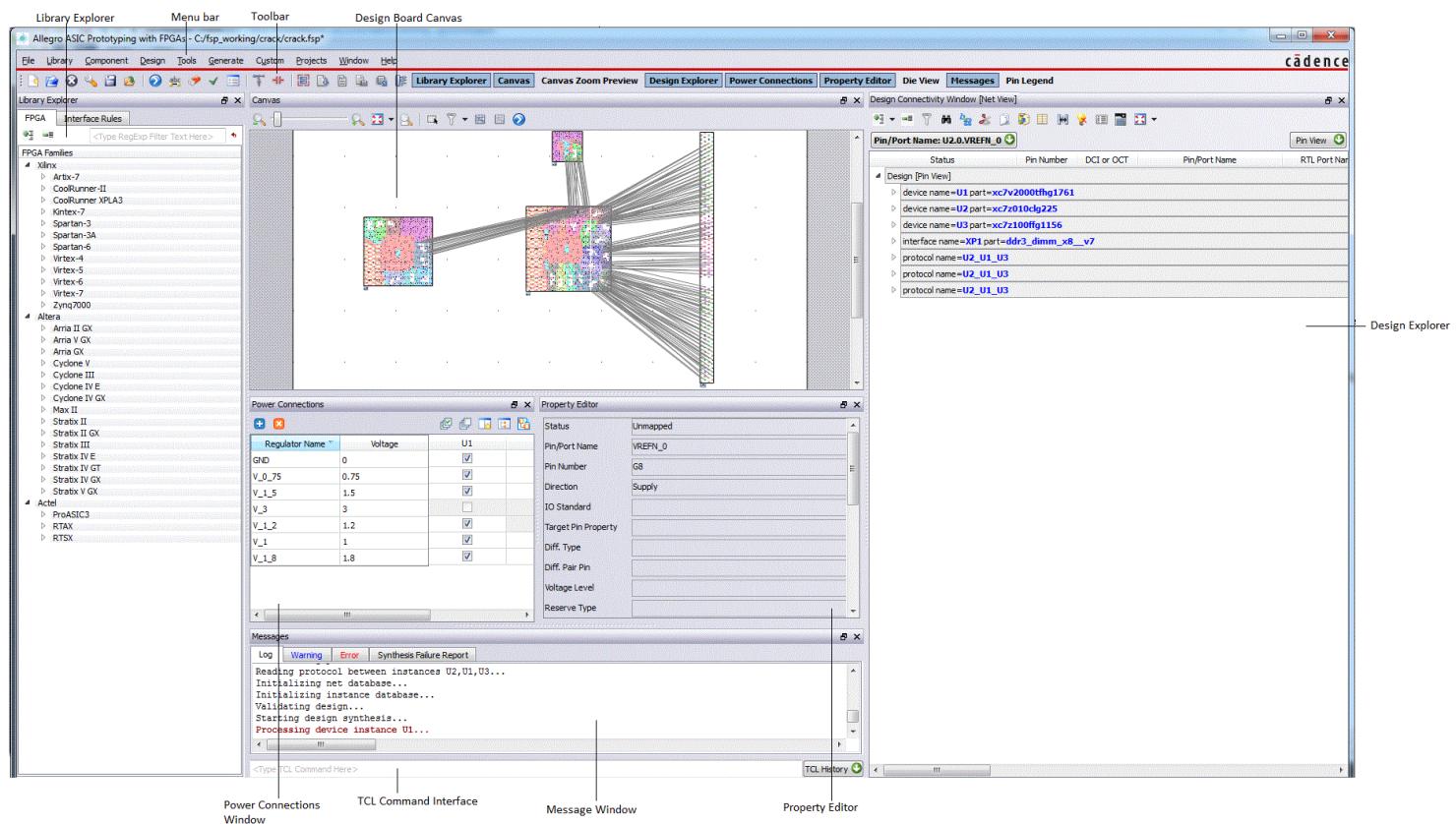
To open a sample design, do the following:

1. Select the *Open Sample Design* option.
2. Select a design name from the drop-down list.
3. Click *OK*.

For more information on the UI of the window, see [What do you want to do....?](#)

The FPGA System Planner Workspace

FSP is a new graphical user interface-based application that provides board designers the flexibility of capturing their designs in multiple ways. The application offers you a wide range of user-friendly tools and features that help you easily capture your design. When you open an existing project or create a new project in FSP, the FSP workspace appears.



Note: The workspace view displayed in the above figure is not the default view. The workspace view appears based on the view that you saved the last time.

Libraries provides a hierarchical Windows explorer-type access to the libraries in the design.

Design Board Canvas provides a board layout view, in which you can easily place components on the canvas and create connections between the components. The canvas provides the look and feel of a Printed Circuit Board (PCB).

The Design Connectivity window provides you a consistent and quick way to view, create, and edit information about the components and their connectivity information that required to build a design.

Allegro FPGA System Planner User Guide

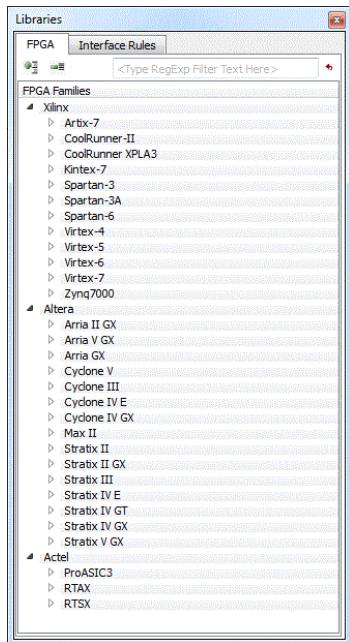
Getting Started with FPGA System Planner

The Message window displays the application error and warning messages.

This chapter covers the following topics:

- [The Libraries window](#)
- [The Messages window](#)
- [The TCL Command Bar](#)
- [The Pin Legend Window](#)
- [The Design Board Canvas](#)
- [The Canvas Zoom Preview](#)
- [The Properties Window](#)
- [The Design Connectivity window](#)
- [The Status Bar](#)
- [The Die View](#)
- [The Power Connections Window](#)
- [Menu Bar](#)
- [Toolbars](#)
- [Customizing Toolbars](#)
- [Undoing and Redoing Changes](#)

The Libraries window



The Libraries window always appears at the left side in the FSP workspace frame whenever you open or create a new project. You use the Libraries window to browse and manage the libraries that you need for your project. The Libraries includes two tabs:

- **FPGA tab**
- **Interface tab**

FPGA tab

The FPGA tab displays a list of FPGA vendors, supported device family names, and part names in tree view structure.

Interface tab

The Interface tab displays a list of interface library names and the associated model names in tree view structure. The interface library names are organized based on their logical characteristics such as memories, connectors, and microprocessors.

For detailed information on the library structure and library files, see the [Working with Libraries](#) chapter.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

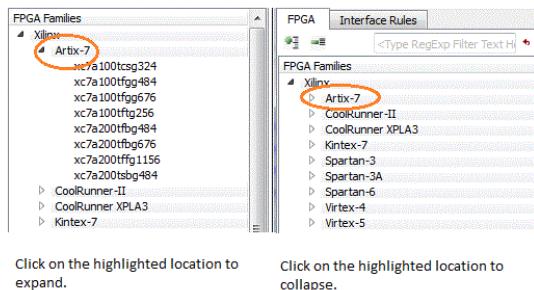
This section covers the following topics:

- [Browsing Parts](#)
- [Searching Part Names](#)
- [Refreshing the Display](#)
- [Context Menu Options](#)

Browsing Parts

Within the Libraries window, the FPGA and interface library files are displayed in the tree view structure. You can expand or collapse the structure by clicking the plus or minus signs to the left of the names. A plus sign indicates that you can expand the device family/interface library name to view its contents. A minus sign indicates that the content is visible and listed below.

Note: You can also click the names to the right of the plus and minus sign to expand or collapse the structure. The following figure shows an example of expanding and collapsing the structure.



The Libraries window also provides Expand All and Collapse All toolbar buttons at the top-left side. Click the *Exand All* button to expand the entire structure and the *Collapse All* button to collaspe the structure.

Searching Part Names

The Libraries window provides a powerful filter tab with which you can filter or search for a specific part or model name. In the Filter tab, you can either enter the complete string or specific alpha-numeric characters that includes in the part you seek. The Regular Expressions search feature is also supported in the Filter tab to make the search more easier. You can use the question marks (?), asterisks (*), underscore (_), and other metacharacters in the search string. FSP searches all the part names available in the Libraries window to find the required part name that matches the specified string.

A few quick examples to search the strings:

- To search for all parts targeted to virtex5, use either V5 or 5\$.
- To search for all parts reference containing DDR, you can use either DDR or ^DDR.
- To search for a part containing specific string such as flash, zdok, bga, tsop, you can enter the keyword.

Examples to search a part or a model name without using Regular Expressions

This section shows a few examples to perform search in the *Libraries* window without using regular expressions.

For more information, see the [Using Regular Expression](#) section.

Example 1

To search for the part name xc7k160tfg676, click the *FPGA* tab and specify any one of the following strings in the Filter tab.

- xc7k
- tfg
- k160
- bg676

FSP displays a list of part names containing xc7k or tfg or k160, or bg676. This list includes the part name xc7k160tfg676. To obtain best results, specify a unique or specific keyword from the desired part name which does not matches with none of the part names in the FPGA libraries. However, it is not mandatory that the desired part name must start or end with the string you enter.

Example 2

To search for the model name qdriip_sram_x18_b14_165bga_v6, click the *Interface Rules* tab and enter any one of the followings:

- qdriip_sram
- qdriip_sram_x18
- qdriip_sram_x18_b14
- 165bga_v6

FSP displays a list of model names containing the strings listed above. The list includes the desired model name qdriip_sram_x18_bl4_165bga_v6.

Note: It is not recommended to use sram as search string, as it may display more results.

Examples to search a part or a model name using Regular Expressions

This section shows a few examples to perform search in the *Libraries* with using regular expression.

Example 1

To search for all parts with reference containing a DDR or DDR followed by any numeric 2 and 3 and followed by alpha-numeric or special characters, use the search string DDR(2|3)_(.*) .

'.' matches any alpha-numeric or special character expect newline character (\n), such as ddr2_sdram_x16_sd_84bga_v5, ddr3_sram_x9_cio_bl4_165bga_v4 and more.

Example 2

To search for all parts with reference containing a SO or U followed by a string DIMM and ends with V5, use the search string ^(SO|U)DIMM_(.*)5\$.

The regex matches following strings:

sodimm_200pin_64b_ddr2_x8_v5

sodimm_200pin_64b_ddr2_x16_v5

udimm_184pin_72b_ddr_x8_v5

udimm_240pin_72b_ddr2_x8_v5

^ character indicates the search string should start with either SO or U, such as sodimm and udimm.

'.' matches all alpha-numeric characters till '_' underscore character before 5, such as _200pin_64b_ddr2_x16_v.

5\$ indicates the search string must ends with numeric character 5, such as V5.

Refreshing the Display

The Libraries window lets you refresh or update the interface libraries. This is helpful when you add or remove libraries in the *Rules File Path Editor* window and want the changes to reflect immediately in Libraries.

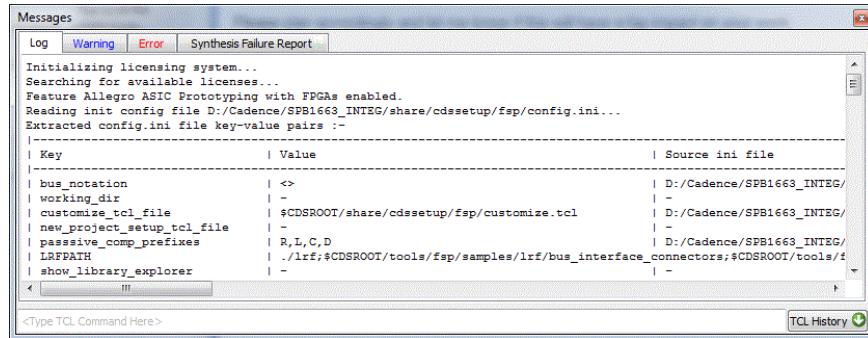
To refresh interface libraries in the Libraries, click the *Interface Rules* tab and click  to refresh.

Context Menu Options

You can access various context menu options in the *Libraries*. Right-click the part name or model to access these context menu options. The following table describes the usage of the context menu options:

Menu Name	Description
Place Component	Use this option to place the selected component on canvas. This option is available in both the FPGA and Interface Rules tabs.
Open Containing Dir	Opens the directory where the selected component library file is present. This option is available in both the FPGA and Interface Rules tabs.
Refresh Libraries	Refreshes the list. This option is available in both the FPGA and Interface Rules tabs.
Edit Component	Opens the Rules Editor form. You can use this form to perform logical changes in the component. This option is available in the Interface Rules tab.
View Component	Opens the Rules Viewer form. The Rules Viewer form is a read-only form. This option is available in the Interface Rules tab.
Edit Rules File Path	Opens the Rules File Path Editor dialog box. You can use this dialog box to add, modify, remove, or set working directory. This option is available in the Interface Rules tab.

The Messages window



The *Messages* window displays the events, process, and design reports that occur during designing in FSP. This window also includes the results and messages that occur from the execution of options available in the *Tools* menu. When you generate an output file using the *Tools* menu options, the *Messages* window displays a successful message along with the path of directory where the output files are generated. In such cases, FSP always displays the directory path with a hyperlink. You can click the link to open the directory.

FSP also reports the errors and warnings in the *Messages* window. Errors are displayed in *red* and warnings in *blue*. The errors that occur during synthesis are displayed in detail in the *Synthesis Failure Report* tab.

During the project session, the messages are gradually saved in the `<project_name>.log` file located in the current project directory. You can also save the messages to any location on your system by right-click in the *Log* tab and choose *Save Log Messages*.

The *Messages* window contains the following tabs:

- The Log View

Displays the design reports, errors, warnings, and log messages.

- The Warning View

Displays the warning messages.

- The Error View

Displays the error messages.

- The Synthesis Failure Report

Displays a log of messages that occur during synthesis. The report is displayed in a tree view structure. You can expand or collapse the branches of the tree by clicking the + and

- buttons beside them, respectively. The tree structure illustrates the contextual flow of the FSP synthesis engine. The contextual path of each message helps you understand the content under which the message was thrown.

The Synthesis Failure Report tab

After running the FSP synthesis, if there are any failed connections in the design, you would be able to view the reasons for these failures in the *Synthesis Failure Report* tab of the *Messages window*. These failures could be due to Vcco/Vref voltage mismatch, insufficient pins in the target device bank, incompatible IO standards, and several other reasons. The *Synthesis Failure Report* lets you browse, view, and analyze the reasons for the failed connections.

The Synthesis Failure Report tab is displayed in a tree view structure. You can expand or collapse the branches of the tree by clicking the + or – buttons. The tree structure illustrates the contextual flow of the FSP synthesis engine. The contextual path of each message helps you understand the context under which the message was thrown. Through the tree structure, the report is presented structurally in a sequential fashion in which the synthesis engine attempts to make connections. This kind of presentation helps you understand the reasons why each of these attempts failed and lets you take corrective actions to fix the failures. After you resolve the failed connections, you can synthesize the design again and check the failures. You can repeat the process as often as necessary.

Typically, you can see the Synthesis Failure Report tab for a failed interface pin or protocol port by clicking the *Failed* hyperlink against it on the *Status* column of the Design Connectivity window. When the pin/port is a part of a logical group with a group constraint such as Same Clock Region, Same Bank etc., the report corresponding to the entire logical group is presented as the failure of any pin/port in the logical group will lead to the failure of the entire group. Similarly, when a pin/port is a part of an interface or protocol segment with a model level constraint such as Same MMCM Region or Same SLR Region etc., the report corresponding to the entire interface/protocol is presented.

Viewing the Violation Report

After synthesis is complete, if there are any failed connections, the Design Connectivity window automatically switches to the *Synthesis Failure Pins* view. In addition, the connection status for each failed pin/port is displayed in the *Status* column as *Failed*. Also, the connection status of successfully connected pins is displayed as *Allocated*.

Note: In *Status* column, the *Allocated* text is displayed in blue color and the *Failed* text in red color.

FSP facilitates you with the option to view the detailed report of the reasons for the failed connection. In the Design Connectivity window, the *Failed* status is displayed with hyperlink. When you click the hyperlink, FSP automatically zoomfits the instance to which the selected pin/port belongs to and highlights the pin on the Canvas. In addition, it displays a detailed report that corresponds to the selected pin/port in the *Synthesis Failure Report* tab in *Messages* window.

To view the synthesis failure report of any pin/port, perform the following step:

- In *Status* column in the *Design Connectivity window* window, click *Failed* against the corresponding pin/port.

The report is displayed in the *Synthesis Failure Report* tab of the *Messages* window.

To browse the synthesis failure report, perform the following steps:

- In the *Synthesis Failure Report* window, do the following:
 - a. click + to expand the tree structure.
 - b. Click - to collapse the tree structure.

Cross-Probing from the Synthesis Failure Tab

Sometimes a large number of violation messages makes it difficult to quickly analyze, identify, and debug the problem.

In the *Synthesis Failure Report* tab, you can cross-probe to elements such as instance, bank, group, and pin/port by clicking on the corresponding texts in the message. To understand whether you can cross-probe a particular element hover the mouse pointer over the message. The mouse pointer turns into hand icon and selected text is displayed in blue color for the items on which cross-probing can be done. The blue colored text indicates that the selected text is navigable.

Toolbar options

The following table describes the toolbar options available in the *Synthesis Failure Report* window and their usage.

Toolbar name...	Description...
------------------------	-----------------------

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Expand All	Use this option to expand the complete tree view structure of violation message.
Collapse All	Use this option to collapse the complete tree view structure of violation message.
Verbose Report	<p>Use this option to turn off or turn on the verbose mode.</p> <ul style="list-style-type: none">■ Turn on this mode to display a more detailed log of events that subsequently lead to failure.■ Turn off this mode to display a one line reason for the failure.
Save Report	<p>Use this option to save the failure report in a text file.</p> <p>Click to invoke the <i>Save To File</i> dialog box. Browse to the directory where you want to save the text file. Specify the name for the text file and click <i>Save</i>.</p>

This section contains the following topics:

- [Searching in the Messages window](#)
- [Context Menu Options](#)

Searching in the Messages window

In the *Messages* window, you can search for specific text, keyword, numeric character, or line. The *Find* bar lets you easily locate the text or keyword in the *Messages* window. You can either enter the complete string or specific alpha-numeric character that includes in the text you seek. The *Find* bar is applicable to all the four tabs in the *Messages* window. By default, the *Find* bar is a hidden option. You can invoke the bar by choosing any one of the following:

- Right-click in the *Log* window and choose *Find*.
- Press **Ctrl + F**

The *Find* bar provides various options which make your searching operation easy. The following table describes the usage of each option.

Name	Description
X	Click to exit.
Previous	Displays the previous matched string.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Next	Displays the next matched string.
Highlight All	Click to highlight the keyword that you are searching for.
Match Case	Click to match the case (Keyword).
Whole Words	<p>Use this option if you want to search for the whole keyword.</p> <p>For example, if you uncheck this option and search for XP23, FSP highlights the words which contain or start with XP such as XP11, XP22, XP_DDR2 and so on.</p> <p>If you check this option, FSP highlights the keyword XP23.</p>

Context Menu Options

You can access various context menu options in the *Messages window*. Right-click in the *Log* view to access the context menu options. The following table describes the usage of the context menu options:

Note: The following menu options are available in the *Log*, *Error*, and *Warning* views.

Menu Name	Description
Copy	Copies the text in the clipboard.
Copy Link Location	
Select All	Selects all the text in the <i>Log</i> window.
Open File	This option is enabled when you select the path to the directory and file name. After selecting, right-click on the selected line and choose this option to open the selected file name. For example, select the line C:/working/project5/output/dehdl/project5.cpm, right-click and choose <i>Open project5.cpm</i> .
Open Directory	This option is enabled when you select the any directory path displayed in the <i>Messages window</i> . After selecting, right-click the selected line, and choose this option to open the selected directory. For example, select the path C:/fsp_working/project5/output/, right-click and choose <i>Open Directory output</i> .

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Word Wrap	Select this option to continue the words on a new line when a line is full, such that each line fits in the <i>Messages window</i> . This lets you read from top to bottom without any horizontal scrolling.
Show Colors	Displays the messages text in different colors.
Show Time	Displays the time in hour:minute:sec format. The time text is displayed at the start of the message line. When you select this option, FSP displays the time from the next message.
Find	Displays the <i>Find</i> bar.
Increase Font Size +	Increases the font size of the <i>Messages window</i> .
Decrease Font Size -	Decreases the front size of the <i>Messages window</i> .
Save Log Messages	Use this option if you want to save the messages at a different location on your system. After clicking, the <i>Save To File</i> window is displayed. Browse to the directory where you want to save the file, specify the file name, and click <i>Save</i> .
Clear Log Messages	Deletes the message from the <i>Log</i> view.
Clear All Messages	Deletes the messages from the <i>Log</i> view, error messages from the <i>Error</i> view, and warning messages from the <i>Warning</i> view.

The TCL Command Bar



The Messages window includes the TCL Command bar. The TCL Command bar is located at the bottom of the *Messages window*. You use this command bar to execute a TCL command. After executing the TCL commands, the results, errors, or warning messages are displayed in the *Log* view.

This section covers the following topics:

- [Entering TCL Commands](#)
- [Using TCL Help](#)
- [Using TCL History](#)

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

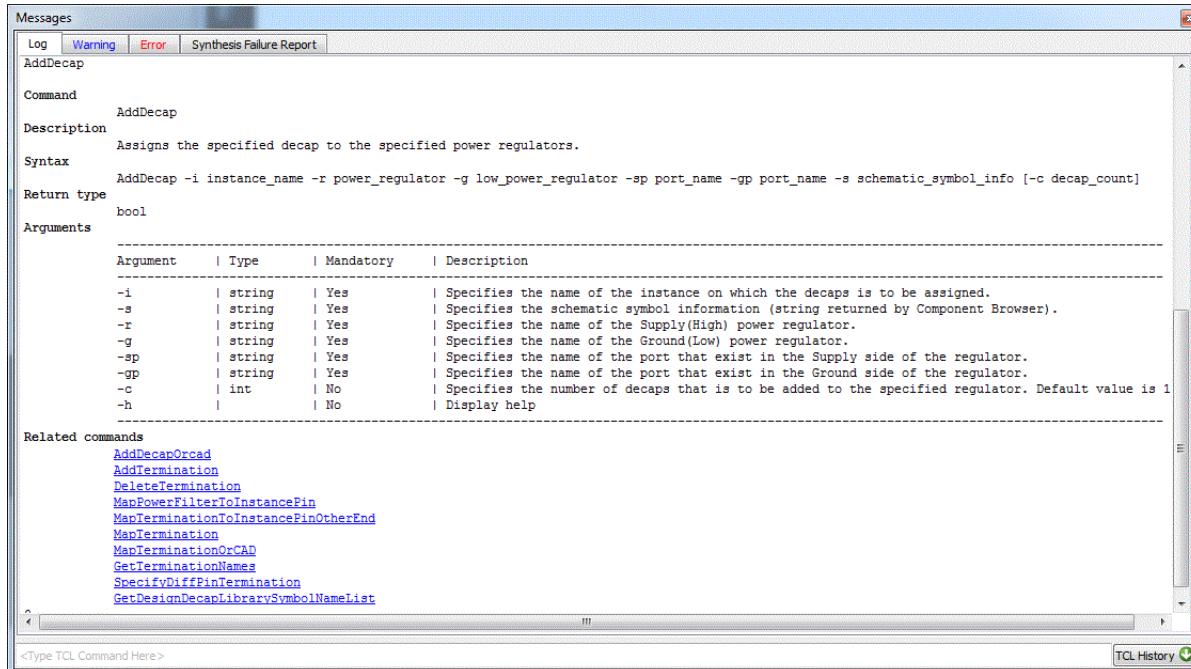
Entering TCL Commands

You can run a TCL command from the *TCL Command* bar. As you type a command, FSP automatically filters and suggests complete commands.

Using TCL Help

FSP provides a detailed help for all the TCL commands. Use the syntax `help <command name>` to access more help a specific command.

For example, type `help addnet` and press *Enter*. The following help is displayed in the *Log* view.



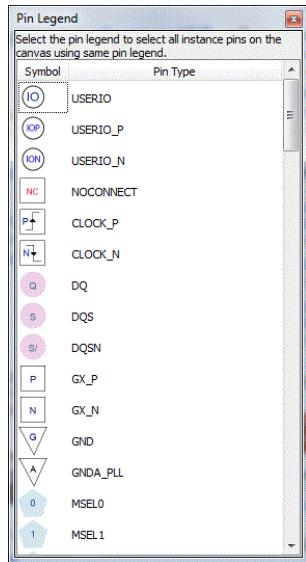
You can also find the complete detail about all FSP's TCL commands in the [FSP TCL Reference](#) guide.

Using TCL History

When you execute a command in the *TCL Command* line, the associated command and the arguments are registered and saved within the TCL interpreter. You can access the previously executed commands using the *TCL History* option in the *TCL Command* line. Click the *TCL*

History option and select any one of the command from the list. The selected command and the associated arguments are displayed in the *TCL Command* line. Press *Enter* to execute the command.

The Pin Legend Window



The Pin Legend window provides a legend of icons of the device and interface pins. This window allows you to view and select the legends. The *Pin Legend* window is useful during re-routing nets. During re-routing, you can refer to the *Pin Legend* window, if you are not sure on which type of pin you want to re-route the net.

When you select a legend in the *Pin Legend* window, FSP highlights the legends on the instances (in Canvas) that match with the selected legend.

The following image illustrates an example of highlighting legends.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

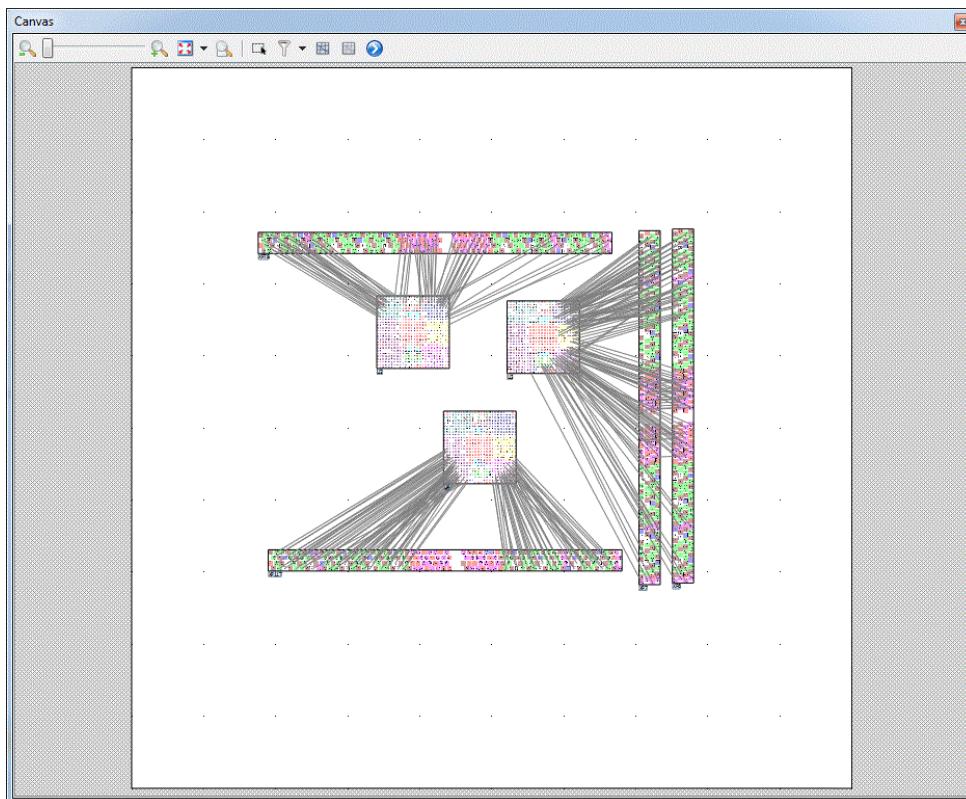


To display the *Pin Legend* window, perform the following steps:

- Choose *Window – Pin Legend*.

By default, the *Pin Legend* window is displayed on the right side of the workspace. You can dock or un-dock the window by using a set of controls available at the upper top corner of the *Pin Legend* window. You can maximize, minimize, or close the window using these controls.

The Design Board Canvas



The design board canvas appears in middle of the FSP frame whenever you open or create a new project. The design board canvas is where you create your design. It provides a board layout view, on which you can easily place components and create connections between them. The canvas displays the connections and physical placement of components as they will appear on Printed Circuit Board (PCB). FSP also supports various keyboard shortcut keys and right mouse button options on canvas to execute frequently performed tasks.

This section contains the following topics:

- [Using View Controls](#)
- [Customizing Board Preference](#)
- [Moving around in the Canvas](#)
- [Using Toolbar](#)
- [Using Mouse Scroll Button](#)
- [Controlling Components in the Canvas](#)

- [Working with Nets in the Canvas](#)
- [Manipulating Instance Data](#)
- [RMB Menu Options](#)
- [Keyboard Shortcuts](#)

Using View Controls

The design board canvas provides a set of controls to dock and un-dock. These controls are available at top corner in the *Toolbar*. You can undock or close the windows using these controls. You can undock the canvas window by using the *maximize* icon. After you undock the canvas from the display docking area, the canvas window appears in a separate floating window. You can move or float the canvas window around the screen. In some cases, the canvas window may overlap with the other windows, you can move it by dragging the banner using left mouse click. You can also resize the canvas view independently by dragging the outer frame using the left mouse click. You can also close the canvas window at any time during the design using the *close* icon.

Customizing Board Preference

FSP supports two colors, *white* and *black* for the canvas. By default, the design board canvas appears in white. You can switch to any color at any time during the design. FSP also allows you to customize the board extents, height, and width. When you create a new project, by default the board extents are set to 10.00 X 10.00 inches.

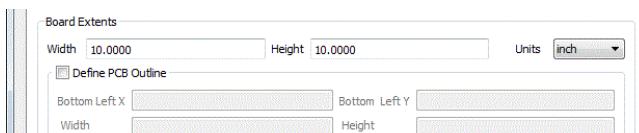
To customize the board extents, perform the following steps:

1. Choose *File – Settings*.

The *Settings* dialog box is displayed.

2. In the *Board Extents* section of the *Project* tab, do the following:

- a. Specify a new value in the *Width* field.
 - b. Specify a new value in the *Height* field.



3. Click *OK* to save the settings.

To change the color of the design board canvas, perform the following steps:

1. Choose *File – Preferences*.

The *Preferences* dialog box is displayed.

2. In the *Color Scheme* section of the *Display* tab, choose *black* option.

3. Click *OK* to save the settings.

The design board canvas appears in black color.

Moving around in the Canvas

Scrolling

In design board canvas, you can scroll up and down, or to the left or the right, to focus on a different portion of the design.

The following are various methods and options to perform scrolling in design board canvas:

- On left side of the canvas, click the *Up* and *Down* arrow button to move up and down one grid unit in the corresponding direction.
- On down side of the canvas, click the *Left* and *Right* arrow button to move right and left one grid unit.
- Roll the mouse wheel on the left side scroll bar to scroll vertically and on the down side scroll bar to scroll horizontally.
- Press *Ctrl + Page Up* to scroll up and *Ctrl + Page Down* to scroll down.
- Left-click anywhere in the canvas and drag the horizontally and vertically using to scroll dynamically.

Zooming

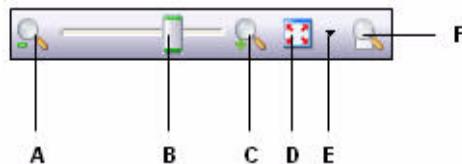
In FSP, to adjust the view of the canvas or to closely look at a particular area of the design you can zoom in using the various zooming functions. These zooming functions are available in following ways:

- [Using Toolbar](#)
- [Using Mouse Scroll Button](#)

Using Toolbar

The following figure depicts the zooming options available in toolbar.

Zoom options in tool bar



The following table lists the zoom toolbar options and the description of each option.

Zoom Option..	Description..
----------------------	----------------------

A (Zoom Out)	Click this button to zoom out the view.
--------------	---

Note: You can press *O* continuously to zoom out.

B (Zoom Slider)	Click and press the bar button and keep slowly move to right to zoom in and left to zoom out.
-----------------	---

C (Zoom In)	Click this button to zoom in the view.
-------------	--

Note: You can press *I* continuously to zoom in.

D (Zoom Fit)	Use the option to see or fit the whole design on canvas.
--------------	--

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

E (Zoom Fit) Several options are available to enhance the zoom fit view. A list of sub menu is displayed when you click this option.

- All Instances – Use this option to see or fit all the instances available in the design on canvas.
- Selected Instances – Use this option to zoom in the particular instance. Left-click on the instance and click this option or press Shift+T.
- PCB Outline – Use this option to zoom in the area inside the PCB outline or press L.

Note: The below options help you to quickly find the instance of your choice to zoom in or zoom out.

- All Instances – A list of options are displayed when you click this option.
 - All Instances - Use this option to see or fit all the instances available in the design on canvas.
 - <Instance_name1> - Click to zoom in the selected <instance_name1>.
 - <Instance_name2> - Click to zoom in the selected <instance_name2>.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

- Instance-wise - List of instances available on the canvas are organized in following ways. You can click the following options to zoom in the particular instance.
 - Device – <device_instance_name> list.
 - Interface – <interface_instance_name> list.
 - Connector – <connector_instance_name> list.
- Part Wise - List of interface instances available on the canvas are organized in part name basis. You can click the following instance name to zoom in the particular interface instance.
 - Part Name – <instance_name1>...
- Family Wise - List of instance available on the canvas are organized in family basis.
 - Connector – <connector_instance_name1>..
 - Normal Interface – <interface_name1>..
 - <device_family_name> – <device_instance_name>..

F (Zoom Fit)

You can also zoom into a specific object/area on the canvas by using the left mouse button. Click this option and click on a blank area, keep the mouse button pressed, and drag the area over the part you want to zoom into. The area is zoomed into as soon as you release the mouse button.

Click to see rats.



Click to hide rats.



Provides the following options:

- Pin - Lets you select the pins on the device.
- Nets - Lets you select the nets on the Canvas.



Selection Filter

Enables the net navigation.





You can add additional details on the Canvas, such as following:

- Title (name of the design)
- Engineer name (name of the engineer who created the design)
- Date (date of creation)

After clicking this option, you will notice a text <Add Info here> on the Canvas. Double-click on the label text and enter your own text.

Note: To delete the notes, double-click on the notes and press *Backspace* or *Delete*.

Using Mouse Scroll Button

You can also use the mouse scroll button to zoom in, zoom out, and move across the design board canvas. The mouse scroll zooming functionality depends on the position of the mouse cursor relative to its location and when you first click the left mouse button. The clock wise direction of the scroll button determines the zoom in and anti-clock wise direction determines the zoom out. By default, if none of the object is selected on canvas by default FSP zoom in the canvas center.

To zoom in using scroll button, perform the following steps:

1. Select an area or instance on canvas and scroll the mouse button in clock wise direction to zoom in.
2. Scroll the mouse button in anticlock wise direction to zoom out.

Controlling Components in the Canvas

Moving Instance

You can easily move or change the location of instances in the canvas. FSP supports both the keyboard shortcut keys and mouse button options to move instance. When you perform a move operation on an instance, you see a vertical and horizontal lines intersecting at centre point of the instance and a move icon. This indicates that you are ready to move instance.

To move an instance, perform the following steps:

1. Do one of the following:
 - Left-click on the instance and press *M*.

- Right-click on the instance and choose *Move*.
2. Drag the instance to a new location and left-click to place the instance.

The Messages window displays the following message with the new X and Y location values:

```
moving instance <instance_name> (part/instance name) at x = <value>  
inch, y = <value> inch.
```

Note:

- Press *Esc* to abort the move operation. After you press *Esc*, the instance automatically moves back to the old location.
- Moving an instance does not break any connections.

Copying and Pasting Instance

FSP supports the standard copy and paste functions. You can copy and paste instances, instances information, and net names across the canvas. You can copy text from other Windows applications and paste it into instance info or instance name text box. You can also copy a instance information from canvas and paste to another Windows application.

FSP supports two types of paste commands, *Paste* and *Paste Special*. The following table provides more detail on these commands:

Paste	Paste Special
Use the <i>Paste</i> command to paste an instance on the canvas if you want the instance information such as instance name, part name, and instance information to be pasted along with the instance.	The <i>Paste Special</i> command is used when you paste an instance on the canvas and if you want the pin and group/bank level properties to be pasted along with the instance. However, the paste special command is useful for interface instances.
The <i>Paste</i> command is active by default.	By default, the paste special is not active when you invoke FSP for the first time. You must enable the paste special command through the <i>Preference</i> window to use it.

To copy a instance, perform the following steps:

1. Select the instance.
2. Right-click on the instance and choose *Copy* or press `Ctrl + C`.

To copy more than one instance, perform the following steps:

1. Press and hold `Ctrl`, and select the instances using the left mouse button.
2. Right-click and choose *Copy* or press `C`.

To paste a instance or instances:

- Right-click the canvas and choose *Paste* or press `Ctrl + V`.

To paste a instance or instances using the paste special command, perform the following steps:

1. Choose *File – Preferences*.
The *Preferences* dialog box is displayed.
2. Click the *Design* tab.
3. Select the *Copy pin and group/bank level properties while copying instance* option.
4. Click *OK* to save the settings.
5. Right-click on the instance and choose *Paste Special* or press `Ctrl + V`.

Note: The *Paste* command will not available through shortcut key `Ctrl + V`, after selecting the *Copy pin and group/bank level properties while copying instance* option in the preference window.

Rotating Instance

Instances on the canvas can be rotated by any angle. Rotating instances helps in improving routability by minimizing crossovers. When you first time place the component on canvas the rotation angle of the instance is 0 degree. You can then repeatedly click to rotate the instance 90 degree clockwise or any angle you required.

Note: When you rotate the instance, the instance info text string is not rotated and is continued to be displayed at bottom of the instances.

When you place the mouse pointer on this option following options are displayed:

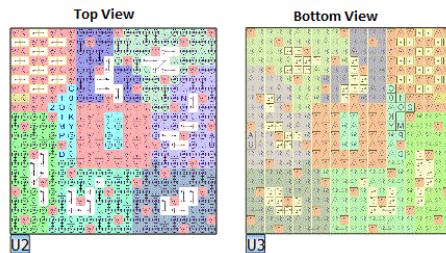
Name	Description
0 Degrees	By default this is checked. Click this to rotate the instance at 0 degree.
90 Degrees	Click this to rotate the instance at 90 degree.
180 Degrees	Click this to rotate the instance at 180 degree.
270 Degrees	Click this to rotate the instance at 270 degree.
Custom...(0)	Use this option to customize the angle rotation of an instance. When you click this option <i>Rotate Instance</i> dialog box is displayed. Specify the angle as required and click <i>OK</i> . For example, specify 30 degree and click <i>OK</i> , the instance on canvas rotates to 30 degree angle.

Flipping Instance

You can flip an instance on the canvas. When you place a component on the canvas, by default the top view of the instance is displayed. You can flip the instance to view the bottom view of the instance. After flipping, the banks at the left side are displayed in the right side and the banks at the right side are displayed in the left side.

Note:

- When you flip an instance, the instance info text string is not flipped and is continued to be displayed at bottom of the instances.
- The top view of an instance is displayed in thick bank colors, where as the bottom view of the canvas is displayed in light shade bank colors, as depicted in the below figure.



To flip an instance from *Top* to *Bottom* view, perform the following steps:

1. Select an instance.
2. Right-click and choose *Flip – Bottom* or press *Ctrl + F*.

To flip more than one instance, perform the following steps:

1. Press and hold Ctrl, and select the instances using the left mouse button.
2. Right-click and choose *Flip – Bottom* or press F.

Selecting and De-selecting Instance

You can perform frequently used operations such as move, rotate, cut, copy, paste on the instance. You can also perform these operations on multiple instances if they are all in selection mode. Instances that are selected appear in the selection color.

To select an instance:

- Position the mouse pointer on the instance and click the left mouse button.

The instance appears in the selection color.

To select multiple instances, perform the following steps:

1. Press Ctrl and left click on the instances.
Or
2. Click the *Selection Rect* icon in the toolbar and drag over the instances that you want to select.

The instances appears in the selection color.

To deselect instances:

- Click an area in the canvas where there is no instance, net, or instance information/name field, or press Esc key.

Deleting Instance

You can delete one or multiple instances from the canvas. The instances must be in the selection mode if you want to delete.

To delete instances, perform the following step:

- Right-click and choose *Delete Instance* or press the Delete key.

Aligning Instances

Aligning instances can be a time-consuming task if it has to be done manually such as moving instances. FSP provides a functionality which helps in speeding up the process of placing the instances in a more systematic manner and achieve a better looking canvas or PCB with little manual effort.

Note: The align functionality doesn't work on nets. During the align operations, connectivity is always maintained. If the instances are already connected, the nets are stretched to maintain the connectivity.

You can align instances, either vertically or horizontally. The Align functionality aligns a selected set of instances with respect to a common axis.

To align a set of instances, you need to select similar type of instances and then run the *Align* commands. The Alignment functionality aligns instances with respect to a common axis. You first need to select the instances to be aligned to activate the Align menu icons on the Canvas toolbar. Depending on the alignment option you choose, a common axis is calculated. You can align instances in the following manner:

- Align Horizontal (Center)
- Align Horizontal (Top)
- Align Horizontal (Bottom)
- Align Vertical (Center)
- Align Vertical (Left)
- Align Vertical (Right)



Important
When you align instances, they may stack on top of each other. Therefore, before aligning objects you must ensure that the objects are positioned relative to one another in the desired manner.

Alignment Option	Description
Align Horizontal (Center)	Aligns objects horizontally through the centers of the objects. The objects are moved in such a way that they lie on same x coordinate as that of component placed at center.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Align Horizontal (Top)	The instances move with reference to the top edge of the topmost instance, such that the top edge of each instance is aligned, that is has the same y coordinate value.
Align Horizontal (Bottom)	The instances move with reference to the bottom edge of the bottom-most instance, such that the bottom edge of each instance is aligned, that is has the same y coordinate value
Align Vertical (Center)	Aligns objects vertically through the centers of the objects. The objects are moved in such a way that they lie on same x coordinate as that of component placed at center.
Align Vertical (Left)	The instances move with reference to the left edge of the leftmost instance, such that the left edge of each instance is aligned, that is has the same x coordinate value.
Align Vertical (Right)	The instances move with reference to the right edge of the right-most instance, such that the right edge of each instance is aligned, that is has the same x coordinate value.

Working with Nets in the Canvas

For more information, see the *Working with Nets and Ports* chapter.

Manipulating Instance Data

Note: In this section, the instance name, information, and part name are referred as *Instance Info*.

The instance info in the canvas are not a part of any logical connections or components. They do not have any effect on the files you generate from FSP. The instance info just provides a way for you to document your design without affecting its connectivity.

Displaying Instance Data

When you place a component on the canvas, the instance name is displayed at the bottom left of the instance. FSP lets you customize the display of the instance information.

To hide and unhide the instance info, perform the following steps:

1. Choose *File – Preferences*.

The *Preferences* dialog box is displayed.

2. Select the options in the *Display* tab, that you want to display on the canvas.

For example, select *Part Name* to display the part name.

3. Click *OK* to save the settings.

The instance data is displayed at bottom left of the instance.

Controlling Text Size

When you select the instance info option in the *Preference* window, by default FSP displays <Add Info Here> text at below of the instance name. This indicates that you must enter some information. You can also increase or decrease the text size of the instance info to have better readability.

To modify the instance info text display, perform the following steps:

1. Choose *File – Preferences*.

The *Preferences* dialog box is displayed.

2. Select a appropriate font size options from the *Design* tab.
3. Click *OK* to save the settings.

RMB Menu Options

FSP Canvas GUI provides bunch of right mouse button menus which provides a quick and easy way to access the most commonly used dialog boxes and tasks. The menus are context-sensitive so different menus appear depending on the instance you select.

The following table outlines the right mouse button options available for both device and interface when the instances are on canvas.

For Device	For Interface	Functionality
Note: The below options are available only when you right click on device instance.	Note: The below options are available only when you right click on interface instance.	

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

-	Target To Device --> < <i>Device_Name</i> >	Use this option to target the interface to device.
Instance Properties	Instance Properties	Opens the Property Editor.
Constraints – Map FPGA Ports and Pins.	-	Opens the FPGA Port Mapping Wizard for Device Instance wizard. You can use this wizard to map the HDL ports names from HDL files to device instance. You can also map the FPGA pin location constraints to the HDL ports from constraint files.
Constraints – Map FPGA Resources	-	Invokes Map Resources for Device Instance window.
Edit Connector Part	-	Opens the Edit Connector dialog box. You can use this dialog box to modify the connector model. Note: This option is available for only connector device.
Constraints – Import Constraints	-	Opens the Import Constraints Wizard For Device Instance dialog box. You can use this dialog box to import constraints completely or partially from constraint file.
Constraints – Export Constraints	-	Opens the Export Design Constraint dialog box. You can use the Export design Constraint dialog box to export the constraints.
Update Instance Footprint		Opens the Update Pin Locations from PCB Footprint dialog box. Use this dialog box to update the device pin locations using external PCB footprint (dra) file.
Link to Schematic Symbols (Allegro/OrCAD)	-	Updates the DE-HDL and OrCAD symbols.
Update Hierarchical Schematic Pages (Allegro/OrCAD)	-	Updates the hierarchical schematic pages of the FPGA.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

	Rules – View	Displays a spreadsheet view of the logical and electrical constraints of the interface rules file. This form can only be used to verify whether you have assigned the correct pin and group information details in Rules File editor.
	Rules – Edit	Opens Rules Editor form. The Rules Editor provides an spreadsheet editor view through which all the logical information necessary for creating part can be imported from existing rule file and can be edited later.
	Rules – Change	This option lets you replace selected rules file with another rules file from the Libraries. The <i>Select Rules File</i> dialog box is displayed when you click this option. Select a rules file name from the list and click <i>OK</i> .
	Rules – Change Rules/Mapping File Reference	Invokes Change Rules/Mapping File Reference dialog box. Note: This option is available only when you place the component on canvas using Component Browser.
	Convert to Rules Interface	In DE-HDL schematic environment invokes component browser and in OrCAD schematic environment Convert Rules File Instance to Real Part dialog box. Note: This option is available only when you place the component on canvas using Libraries.
	Update Instance Footprint	Invokes Update Pin Locations From PCB Footprint dialog box. This dialog box allows you to update the X and Y locations of the current interface rules file. Note: This option is available only when you place the component on canvas using Libraries.
Constraints – Show BUFR Nets	-	Opens the BUFR Nets for Device Instance dialog box.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Change FPGA	-	Open the Change FPGA dialog box. You can use this option to replace the device on canvas with same existing library part model.
Update Symbol		
Report	Report	Generates the design report in: <ul style="list-style-type: none">■ Log window■ Text file
Run	Run	Generates the connections between the components on canvas.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Protocol	-	<p>Displays the following three sub menus:</p> <ul style="list-style-type: none">■ Create<p>Lets you to create a new device protocol between two or more devices.</p><p>This menu displays a list of device instance names present on the canvas plus an additional sub menu called Multi Point.</p><p>To create a two point device protocol select any one of the device instance name from the list. The Edit Protocol editor is displayed after clicking the device instance name.</p><p>Multi Point option lets you to create protocol between multiple devices. After selecting this option the Create New Protocol dialog box is displayed for device chain selection followed by Edit Protocol editor.</p>■ Edit<p>Lets you to edit the definition existing device protocol.</p><p>This menu displays a list of device protocol names of the current design.</p><p>To edit the device protocol definition select any one of the device protocol name. After selecting the protocol name the Edit Protocol editor is displayed. You can continue to edit the existing protocol details as required.</p>■ Delete<p>Lets you to delete the existing device protocol from the current design.</p><p>This menu displays a list of device protocol names of the current design.</p><p>To delete the device protocol select a device protocol name. A warning message is displayed. Click Yes to delete the protocol.</p>
September 2023 © 2023	67	Product Version 23.1 Click Yes to delete

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Virtual Interface	-	<p>Displays two options:</p> <ul style="list-style-type: none"> ■ Create New Virtual Interface You can use this option to create virtual interface. ■ Create New Virtual Interface from HDL You can use this option to create multiple virtual interface with HDL.
-	Rerun	<p>Reruns the complete design.</p> <p>Note: Delete all the nets and starts synthesis.</p>
-	Connect Pin <Part Name> to <Instance name>>>	<p>You can use this option to connect a interface pin to device pin. To target a single interface pin to device pin choose right-click instance pin - <device instance name> - <bank name> - <device pin name>.</p> <p>Note: This option is enabled only when you target the interface to device.</p>
Re optimize Protocol	-	<p>You can use this option to re-assign pins for specific protocol either for a given device or for the whole design protocol chain.</p>
Re-Assign Net on Pin I/O_<Pin Name>	-	<p>You can use this option to manually swap the connection of one pin to another pin.</p> <p>Right-click device instance pin and select this option. A thick dotted line is displayed. Take the line to other device pin with which you want to swap the connections. Double-click on the device pin.</p> <p>If the pin is not suitable, an error message is displayed in Message Log section. FSP does not expel the connection unless you select a suitable pin.</p> <p>Note: This option is available only for device instance.</p>

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Move	Move	You can select this option to move the instance to another location on the Canvas. When you select this option, the mouse pointer changes to a suggestion that you move the instance. Once you select the new location, click the mouse button to place the instance. You press the Esc button to abort the operation any time during the movement.
Align	Align	Select the following options to align the instance, <i>Center</i> --> Aligns the instance Centrally. <i>Horizontally (Top)</i> --> Aligns the instance Horizontally Top. <i>Horizontally (Bottom)</i> --> Aligns the instance Horizontally Bottom. <i>Vertically (Right)</i> --> Aligns the instance Vertically Right. <i>Vertically (Left)</i> --> Aligns the instance Vertically Left.
Copy	Copy	Copies an instance.
Paste	Paste	Pastes a new instance on the canvas with default settings. If the option <i>Copy pin and group/bank level properties while copying instance</i> in the <i>Preference</i> dialog box is selected, the <i>Paste</i> command pastes a new instance on the canvas with instance-level settings.
Paste Special	Paste Special	Pastes a new instance on the canvas with instance-level settings.
Send it back	Send it back	Sends the instance back to another location.
Bring it front	Bring it front	Moves the instance to the front to another location.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Rotate	Rotate	<p>This option helps you to rotate the instances on canvas. Rotating instances helps in improving routability by minimizing crossovers. By default when you first time place the component on canvas the rotation angle of the instance is 0 degree. You can then repeatedly click to rotate the instance 90 degree clockwise or any angle you required.</p> <p>Note: When you rotate the instance, the instance info text string is not rotated and is continued to be displayed at bottom of the instances.</p> <p>When you place the mouse pointer on this option following options are displayed:</p> <ul style="list-style-type: none">■ 0 Degrees – By default this is checked. Click this to rotate the instance at 0 degree.■ 90 Degrees – Click this to rotate the instance at 90 degree.■ 180 Degrees– Click this to rotate the instance at 180 degree.■ 270 Degrees – Click this to rotate the instance at 270 degree.■ Custom...(0) – Use this option to customize the angle rotation of an instance. <p>The <i>Rotate Instance</i> dialog box is displayed when you click this option. Specify the angle as required and click <i>OK</i>. For example, specify 30 degree and click <i>OK</i>, the instance on canvas rotates to 30 degree angle.</p>
Flip		Flips the instance.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Delete Nets	Delete Nets	<p>Deletes the nets selected to pin(s) or to delete all the nets to the instance. Following more options are available:</p> <ul style="list-style-type: none"> ■ Delete All Nets to Instance – Deletes all the nets associated with the selected instance. ■ Delete Nets to Selected Pins – Deletes nets associated with the selected pin of the instance. ■ Delete Nets to Protocol – Deletes the interface and device protocols. ■ Delete Nets to Bank (for a device) – Deletes the nets belongs to single bank. ■ Delete Nets to Group (for a interface) – Deletes the nets belonging to same interface group.
Instance Nets	Instance Nets	Displays the options, Show, Hide, Lock, and Unlock. Select the appropriate one.
Highlight Connected Pins	Highlight Connected Pins	Highlights the connected pins on both FPGA's and interfaces.
Bank< <i>Bank_Name</i> >	Group< <i>Group_Name</i> >	<p>Displays the following options:</p> <ul style="list-style-type: none"> ■ Select All Pins ■ Select All Nets ■ Show All Nets ■ Hide All Nets ■ Lock All Nets ■ UnLock All Nets ■ Lock All Constraint Pin Nets ■ Unlock All Constraint Pin Nets
Delete Instance	Delete Instance	Deletes the instance from the canvas.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

When you right-click on a net, the following options are displayed on the shortcut menu		
Selectable Items		<p>Place the mouse pointer on this option and select both or anyone of the following options.</p> <ul style="list-style-type: none">■ Pin – Select this option to enable the pin selection on the canvas.■ Nets – Select this option to enable the net selection on the canvas.
Select All		Selects all items on the canvas. For example, instance and nets.
Highlight Connected Pins		Highlights the connected pins on both FPGA's and interfaces.
Design Properties		Invokes the Properties window for the design.
Reset Instance Name Positions		Resets the instance name positions for all the instances. A confirmation window appears when you click on the option. Click Yes to proceed.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Net	<p>Displays the following options:</p> <ul style="list-style-type: none">■ Selected<ul style="list-style-type: none">Displays the following options:<ul style="list-style-type: none"><input type="checkbox"/> Show in Design Connectivity Window<ul style="list-style-type: none">Displays the following items:<ul style="list-style-type: none">○ Name of the selected net.○ Pin names of the instances between which the selected net is connected.When you click on any one of the options above, FSP navigates you to the selected item on the Design Connectivity window.<input type="checkbox"/> Update NetGroup<ul style="list-style-type: none">Lets you specify and update a NetGroup name to all the nets. When you click this option, the Create or Update NetGroup window appears. Specify a name you want to use it as a NetGroup name and click OK.<input type="checkbox"/> Hide<ul style="list-style-type: none">Hides the selected nets on the canvas view.<input type="checkbox"/> Hide Rest (Un-Selected)<ul style="list-style-type: none">Hides all the nets except the selected nets.<input type="checkbox"/> Lock<ul style="list-style-type: none">Locks the selected nets.<input type="checkbox"/> Unlock<ul style="list-style-type: none">Unlock the selected nets.<input type="checkbox"/> Delete<ul style="list-style-type: none">Deletes the selected nets in the design. You can multi-select the nets by pressing and holding <code>Ctrl</code> button to delete them at once.
-----	---

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

When you right-click the blank space in Canvas view following options are displayed.	
Selectable Items	<p>When you place the mouse pointer on this option and select both or anyone of the following options:</p> <ul style="list-style-type: none"> ■ Pin – Select to enable the pin selection on the canvas. ■ Nets – Select to enable the net selection on the canvas.
Select All	Selects all the components and nets on the canvas.
Highlight Selected Pins	Highlights the connected pins on both FPGA's and interfaces.
Paste Instance	<p>You can use this option to paste the instance on the FSP canvas.</p> <p>Note: This options is available only when you have copied the instance.</p>
Hide All Net	Hides the entire design nets.
Show All Net	<p>Displays the entire design nets.</p> <p>Note: This option is available only when all the nets are hidden.</p>
Add Notes (Text)	<p>You can add additional details on the Canvas, such as following:</p> <ul style="list-style-type: none"> ■ Title (name of the design) ■ Engineer name (name of the engineer who created the design) ■ Date (date of creation) <p>After clicking this option, you will notice a text <Add Info here> on the Canvas. Double-click on the label text and enter your own text.</p> <p>Note: To delete the notes, double-click on the notes and press <i>Backspace</i> or <i>Delete</i>.</p>

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Keyboard Shortcuts

There are several keyboard shortcuts available to perform the tasks quickly and easily.

Note: * signifies applicable in the schematic and in the abstract view.

The following is a list of keyboard shortcuts available on canvas.

Action	Key Combination
Extend this selection while selecting pin,net and instance	Ctrl + lmb
Extend this selection while selecting a pin in the group/bank and net.	Shift + lmb
Display the RMB menu on any of the already selected objects.	rmb
Copy the selected instance	Ctrl + C
Paste the selected instance	Ctrl + V
Delete all selected instances or selected nets	Delete
Rotate the selected component by +90 degree	R
Rotate the selected component anti-clock wise by 90 degree	shift + R
Flipping selected component	F
Move selected component	M, Alt + Left click
Select all components and nets on the canvas	Ctrl + A
Zoom In	I *
Zoom Out	O *
Zoom Fit	T *
Zoom Fit (<i>for the selected instance</i>)	Shift + T *
Zoom Rect Mode	Z *

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Selection Rect Mode	S
Move selected instances to northward.	Up
Move selected instances to southward.	Down
Move selected instances to westward.	Left
Move selected instances to eastward	Right
Cancel move mode, or net re-route mode, or zoom rect mode, or selection rect mode	Esc

The following is a list of FSP menu bar keyboard shortcuts.

Action	Key Combination
Create a new project.	Ctrl + N
Open an existing project.	Ctrl + O
Close a project.	Alt + Q
Define project settings	Ctrl + T
Save the changes	Ctrl + S
Save Project As	Ctrl + Shift + A
Split a project	Ctrl + L
Merge the current project	Ctrl + M
Merge different projects	Ctrl + D
Quit from FSP	Ctrl + Q
Run the design	Ctrl + R
Import Allegro Design	Ctrl + A
Update the Design From the Board File	Ctrl + U
Import the Spreadsheet	Ctrl + I
Import the Constraint	Ctrl + Shift + I

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Preference Ctrl + P

Process Ctrl + P

The following is a list of keyboard shortcuts available in Messages window.

Action	Key Combination
Copy	Ctrl + C
Select All	Ctrl + A
Open File	Shift + F
Open Directory	Shift + D
Word Wrap	Ctrl + Z
Show Colors	Ctrl + K
Show Time	Ctrl + E
Find	Ctrl + F

The following is a list of keyboard shortcuts available while working with nets and pins.

Action	Key Combination
Selection of Single Net	Ctrl + Right Click
Selection of Group (Nets)	Ctrl + Shift +Right Click
Selection of whole Bank on the canvas	Shift +Right Click

The following is a list of keyboard shortcuts available in Process Option window.

Action	Key combination
Apply To All	Ctrl + A
Reset To Default	Ctrl + D
Check All	Ctrl + C

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

UnCheck All	Ctrl + U
Expand All	Ctrl + E
Collapse All	Ctrl + R
Set Second Pass For All	Ctrl + P
Reset Second Pass For All	Ctrl + Shift + All
Save Run Configuration	Ctrl + S

The following is a list of keyboard shortcuts available in the Lock Nets window

Action	Key Combination
Lock All Nets	Ctrl + A
UnLock All Nets	Ctrl + shift + A
Lock All Clocks	Ctrl + C
UnLock All Clocks	Ctrl + Shift + C
Lock Selected Nets	Ctrl + S
UnLock Selected Nets	Ctrl + Shift + S
Import Locked Nets	Ctrl + I
Export Locked Nets	Ctrl + E
Find	Ctrl + F

The following is a list of keyboard shortcuts available in Configure pin, Die view, and Package view.

Action	Key Combination
Find	Ctrl + F
Clear Selection	Ctrl + L
Expand All	Ctrl + Q
Collapse All	Ctrl + W
Paste	Ctrl + V

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

The following is a list of keyboard shortcuts available in the Change Signal Form.

Action	Key combination
Copy	Ctrl + C
Paste	Ctrl + V

The following is a list of keyboard shortcuts available in Logical tab of the Rules Editor.

Action	Key Combination
Add a new Group	Ctrl + A
Edit the Model	Ctrl + E
Edit Custom Attributes	Ctrl + T
Auto Detect Pin Pairs	Ctrl + D
Edit the selected Group	Ctrl + G
Paste	Ctrl + V

The following is a list of keyboard shortcuts available in Schematic tab of the Rules Editor.

Action	Key combination
Merge All Splits	Ctrl + N
Add Split Symbols	Ctrl + A
Edit Symbol Settings	Ctrl + T
Auto Detect Pin Pairs	Ctrl + D
Merge Split Name	Ctrl + E

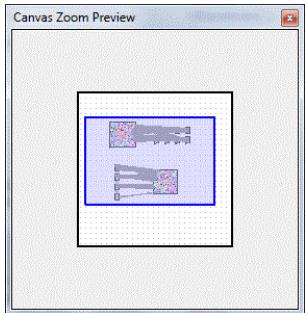
The following is a list of keyboard shortcuts available in Layout (Pin Footprint) tab of Rules Editor.

Action	Key Combination
Add Pins.	Ctrl + A
Edit Symbol Settings.	Ctrl + T
Edit Properties.	Ctrl + E

The following is a list of keyboard shortcuts available in the xml Editor.

Action	Key Combination
Selects the current cell to be the first cell in the group	Ctrl + Pgup
Selects the current cell to be the last cell in the group	Ctrl + PgDn
Selects the current cell to be the last cell of the whole column	Ctrl + End
Selects the current cell to be the first cell in the whole column	Ctrl + Home

The Canvas Zoom Preview



The *Canvas Zoom Preview* feature lets you navigate around the entire design area. It also enables a normal pan of the viewed design area. This feature is applicable when you are working in the canvas. The Zoom preview reflects the zoom area and the selected instances for the active view. The below figure displays the overall design view which is currently zoomed into the area navigated by the rectangle. You can select and drag the navigation rectangle to reposition the displayed area in the canvas. In addition, the rectangle moves as you zoom in and out, or move across in the canvas.

When you invoke the canvas zoom preview window, it displays the design as per the current zoom settings. You can use the zoom in and zoom out icons in the toolbar to zoom the particular area in the canvas. The Zoom preview window can be repositioned anywhere in the tool. Click the *Maximize* button in the upper right corner of the window and drag anywhere on the perimeter of the tool. When you resize and reposition the Zoom preview window, it remains at the size and position you have established whenever you close and reopen the design.

To invoke the Zoom preview window, perform the following step:

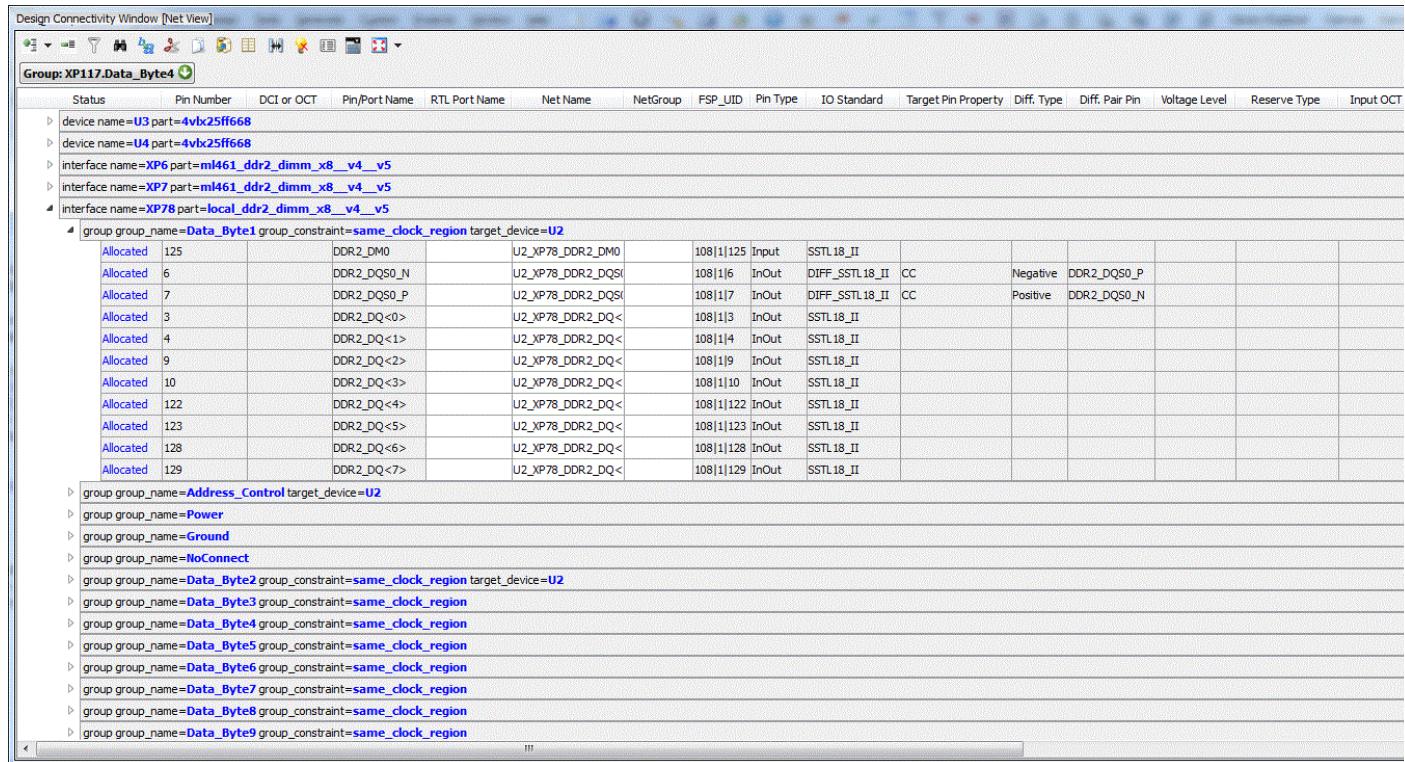
Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

■ Choose Window – *Canvas Zoom Preview*.

The *Preview* window is displayed at the right side of the canvas.

The Design Connectivity window



FSP provides you a new design environment, *Design Connectivity* window that helps you to capture the design in a spreadsheet-based view. This view allows you to quickly capture or modify connectivity information in the design. With the spreadsheet interface you can quickly work with the properties and constraints across your design. The Design Connectivity window is very effective for capturing designs with high pin count components and devices.

There are many other features in the Design Connectivity window that can be used to modify and maintain the connectivity of complex and huge designs in a fast and simple manner. Using the Design Connectivity window, you also have the flexibility of modifying the properties of the components that are used in the current design.

Besides providing a spreadsheet view of the design, the Design Connectivity window also maintains association with the other windows such as Canvas, Properties, Die View, and Power Connections. While working with the Design Connectivity window, you can customize

and put all the windows together to perform various tasks simultaneously. Putting together all the other windows with the Design Connectivity window gives a more efficient way of capturing the design. When the windows are put together with the Design Connectivity window, FSP maintains and displays the changes immediately across all the open windows.

This section covers the following topics:

- [The Design Connectivity User Interface](#)
- [Browsing in Design Connectivity](#)
- [Moving around in the Design Connectivity window](#)
- [Searching and Filtering in the Design Connectivity window](#)
- [The Design Connectivity Window Toolbar](#)
- [Organizing the Workspace for Design Connectivity window](#)
- [Cross-Probing between Design Connectivity and Other Windows](#)
- [Context-Sensitive Menus](#)
- [Accessing Mini Toolbar](#)
- [Switching Views in Design Connectivity](#)

The Design Connectivity User Interface

When FSP starts up, the Design Connectivity window appears by default along with the other windows such as Canvas, Libraries.

You can hide or unhide the Design Connectivity window by choosing *Window – Design Connectivity*. The Design Connectivity window provides a spreadsheet user interface. The spreadsheet editor is the key interface of FSP. You can use the spreadsheet to view or modify the information about components and their connectivity information. The Design Connectivity window lets you quickly connect component pins to signals, apply termination, add pullups and pulldowns, add and remove components and many more.

The following table provides the description of each column that appears in the Design Connectivity window. However, some of the columns appear based on the selected view. For more information on the views, see the [Switching Views in Design Connectivity](#) section. Each row in the Design Connectivity window displays the properties and connectivity information for a pin of an instance and protocol signals placed on the canvas.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Column Name	Description
Status	<p>Displays the connection status of the pin.</p> <p>For example,</p> <p><i>Allocated</i> is displayed, if the pin is connected to a signal after synthesis.</p> <p><i>Unallocated</i> is displayed, if the pin is not targeted or connected to any signal.</p> <p><i>Failed</i> is displayed, if the pin failed to connect to a signal after synthesis.</p> <p><i>Unmapped</i> is displayed, if the power pin is not mapped to any regulators.</p> <p>Note: Click on the <i>Failed</i> text to find more information on the failure reasons.</p> <p>Note: This is a read-only column.</p>
Pin/Port Name	<p>Displays the logical pin names of the interface instance or port names of the device instance.</p> <p>Note: This is a read-only field.</p>
Pin Type	<p>Displays the pin direction of the pins of the instance.</p> <p>Note: This is a read-only column.</p>
Diff.Pair Pin	<p>Displays the differential pair pin of the pins of the instance.</p> <p>Note: This is a read-only column.</p>
Serial IO TX/RX Pin	<p>Displays the serial IO TX/RX pair of the pins of the instance.</p> <p>Note: This is a read-only column.</p>
Diff.Type	<p>Displays the differential type of the pins of the instance.</p> <p>For example, for positive clock pin name CLK_P it displays <i>Negative</i> and <i>Positive</i> for negative clock pin name CLK_N.</p> <p>Note: This is a read-only column.</p>

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Reserve	Type	<p>The cells under this column are enabled for Altera devices. When you click on a cell, the following values are displayed:</p> <ul style="list-style-type: none">■ AS BIDIRECTIONAL■ AS INPUT TRI-STATE■ AS OUTPUT DRIVING AN UNSPECIFIED SIGNAL■ AS OUTPUT GROUND■ AS DRIVING VCC <p>Note: The values are available for the pins of an interface that is targeted to an Altera device.</p> <p>The values lets you specify the reserve state of all the unused pins on the Altera device. For example, select AS INPUT TRI-STATE to reserve the pins as tri-state input pins or AS OUTPUT DRIVING AN UNSPECIFIED SIGNAL to reserve the pins as output pins and drive any signal.</p> <p>FSP outputs these values to the quartus settings file (.qsf).</p>
IO Standard		Displays the name of the IO standards for the interface pins and for Device instance it displays for the pins to which the IO standards are connected.
Connected Pin Number		For interface pin, displays the pin number of the device pin to which the interface pin is connected. For device pin, displays the pin number of the device pin to which the device pin is connected through device protocol.
Connected Pin Name		For interface pin, displays the pin name of the device pin to which the interface pin is connected. For device pin, displays the pin name of the device pin to which the device pin is connected through device protocol.
Connected Bank Name		For interface pin, displays the bank name of the device to which the interface pin is connected. For device pin, displays the bank name of the device to which the device pin is connected through device protocol.
DCI or OCT		Displays a list of DCI standards for the interface pins that are supported by the targeted device. Also, displays a list of OCT values supported by the targeted device.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Input OCT	This column is applicable for Altera devices. Displays a list of input OCT values supported by the Altera device. Select a value to apply on the pin.
Output OCT	This column is applicable for Altera device. Displays a list of output OCT values supported by the Altera device. Select a value to apply on the pin.
Connection Type	<p>Use this column to define the external ports for the pins and nets. The cell of this column provides a list of pre-defined external ports. However, this list varies based on the instance, instance pin, and the connection.</p> <p>The following options are available:</p> <ul style="list-style-type: none">■ Extend as External Port This option is available for the interface pins (connected or not connected).■ Do Not Connect This option is available for the interface pins (not connected).■ Fixed External Port This option is available for interface pins (not connected) and device power pins (not mapped) .■ Fixed Internal Connection This option is available for interface pins (not connected) and device power pins (not mapped) .■ Preserve Pin This option is available for device pins.
Net Name	Displays the name of the net connected to the pin.
RTL Port Name	Displays the RTL port name for the pin. This name is automatically generated when you export the constraints.
Pin Termination	Lets you add the termination to the interface pins. The termination names defined in the <i>Define Termination</i> dialog box are displayed in this column cells. Double-click on a cell and select a termination name from the drop-down list.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

FPGA Ext.Termination	Displays a list of termination names that are defined in the <i>Define Termination</i> dialog box. Double-click on a cell and select a termination name from the drop-down list.
Global Clock Route	This column is applicable for Spartan family devices.
Assigned to Pin	This column is applicable for interface pins. When you click on a cell for an interface pin, a list of pin numbers of the device is listed to which the interface is targeted. Note: When you import constraints from an external file, the values in these cells are automatically updated.
Net Group	Displays the NetGroups. NetGroups are automatically defined under this column. You select the second option in the <i>NetGroup</i> tab in <i>Settings</i> dialog box and place the interface component on the canvas. NetGroups are created based on the interface logical group properties. Under this column, you can modify, redefine, or remove NetGroups.
Custom Attribute	Lets you to add attributes and properties to the pins. Double-click on the cell and click <i>browse (...)</i> to invoke the <i>Custom Attribute</i> window.

Browsing in Design Connectivity

The Design Connectivity window, displays the entire design for the instances you placed on canvas and their properties in a spreadsheet view. By default, the properties of instance groups, banks, and, pins are displayed in non-expanded mode in the Design Connectivity window. The Design Connectivity window lets you browse the entire spreadsheet-based design in detail with the press of a button. You can expand the non-expanded mode using mouse clicks to see the properties. This makes it easy to find, select, and edit properties.

For example, the interface group and pin properties row are displayed in collapsed mode.

To display the group properties of the selected instances, click the > icon next to the interface name.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Status	Pin Number	DCI or OCT	Pin/Port Name	RTL Port Name	Net Name	NetGroup	FSP_UID	Pin Type	IO Standard	Target Pin Property	Diff. Typ
Design [Pin View]											
interface name=XP6 part=ml461_ddr2_dimm_x8_v4_v5											
group group_name=Data_Byt1 group_constraint=same_clock_region											
group group_name=Address_Control											
group group_name=Power											
group group_name=Ground											
group group_name=NoConnect											
group group_name=Data_Byt2 group_constraint=same_clock_region											
group group_name=Data_Byt3 group_constraint=same_clock_region											
group group_name=Data_Byt4 group_constraint=same_clock_region											
group group_name=Data_Byt5 group_constraint=same_clock_region											
group group_name=Data_Byt6 group_constraint=same_clock_region											
group group_name=Data_Byt7 group_constraint=same_clock_region											
group group_name=Data_Byt8 group_constraint=same_clock_region											
group group_name=Data_Byt9 group_constraint=same_clock_region											

To display the bits of vector or scalar pins, click the > icon next to the group name.

Status	Pin Number	DCI or OCT	Pin/Port Name	RTL Port Name	Net Name	NetGroup	FSP_UID	Pin Type	IO Standard	Target Pin Property	Diff. Type
Design [Pin View]											
Interface name=XP6 part=ml461_ddr2_dimm_x8_v4_v5											
group group_name=Data_Byt1 group_constraint=same_clock_region											
Unallocated 125 DDR2_DMO XP6_DDR2_DMO 6 1 125 Input SSTL18_II											
Unallocated 6 DDR2_DQS0_N XP6_DDR2_DQS0_N 6 1 6 InOut DIFF_SSTL18_II CC Negative											
Unallocated 7 DDR2_DQS0_P XP6_DDR2_DQS0_P 6 1 7 InOut DIFF_SSTL18_II CC Positive											
Unallocated 3 DDR2_DQ<0> 6 1 3 InOut SSTL18_II											
Unallocated 4 DDR2_DQ<1> 6 1 4 InOut SSTL18_II											
Unallocated 9 DDR2_DQ<2> 6 1 9 InOut SSTL18_II											
Unallocated 10 DDR2_DQ<3> 6 1 10 InOut SSTL18_II											
Unallocated 122 DDR2_DQ<4> 6 1 122 InOut SSTL18_II											
Unallocated 123 DDR2_DQ<5> 6 1 123 InOut SSTL18_II											
Unallocated 128 DDR2_DQ<6> 6 1 128 InOut SSTL18_II											
Unallocated 129 DDR2_DQ<7> 6 1 129 InOut SSTL18_II											
group group_name=Address_Control											
group group_name=Power											

Each instance property appears as a first-level row heading in the Design Connectivity window. When you step down to second-level by clicking >, each instance groups or banks properties is displayed. When you further step down by clicking >, the pins that are available in the selected group is displayed.

In the Design Connectivity window, you can sort the data displayed in the columns. You can sort the column lists according to various criteria and locate element within lists. When the rows are in non-expanded mode, the data is sorted in alpha-numeric order. For example, the instance names at the first-level row heading are in the following order:

U1, U2, U10, XP117, U99, XP6, U123

To sort the data alpha-numerically, click the *Status* column heading name. The instance names will be sorted in the following order:

U1, U2, U10, U123, U99, XP117, XP6

The Design Connectivity window also provides many sorting options. These options depends on the column you choose. The following section illustrates few examples about how the lists are sorted.

■ Pin/Port Name

Sorts the lists according to the pin names. For example, IO_DIFFIO_TX6p, IO_DIFFIO_TX7p, Input_CLK3n, Input_CLK3p, VCCIO1, VCCIO1.

■ Pin Number

Sorts the lists according to the pin numbers. For example, 1, 2, 3 or A1, A2, A3...

■ FSP_UID

Sorts the list according to the ID numbers.

Moving around in the Design Connectivity window

In the Design Connectivity window, you can scroll up or down, or to the left or the right, to focus on a different portion of the pin properties. Not all the columns and pins are displayed in the Design Connectivity window by default. You must scroll up and down, and left and right to see them.

The following are various methods to perform scrolling:

- On the right side of the Design Connectivity window, click and drag the vertical scroll bar to move up and down, and at the bottom side click and drag the horizontal scroll bar to move right and left.
- Click on the up, down, right, or left arrow to scroll one grid unit in the corresponding direction.
- Drag the horizontal or vertical scroll button to scroll the window dynamically.
- Press Page Up to scroll the panning distance up.
- Press Page Down to scroll the panning distance down.
- Roll the mouse wheel up and down anywhere in the Design Connectivity window, to scroll through vertically.
- Roll the mouse wheel up and down on horizontal scroll bar at the bottom side of the Design Connectivity window to move left and right.
- Roll the mouse wheel up and down on vertical scroll bar at the right side of the Design Connectivity window to move up and down.

Note: When you scroll left and right, the instances row headers and the columns that are freezed will not move.

Moving to a Location

You can use the *Find* or *Find and Replace* dialog boxes to move to specific locations in the Design Connectivity window. This feature saves your time from browsing and scrolling.

Searching and Filtering in the Design Connectivity window

Searching Text

In the Design Connectivity window, you can search for specific text in the cells or you can search for a pin by name or by any of its property values. Using the *Find* and *Find and Replace* features, you can locate a text or pin name in the Design Connectivity window. In the *Find* toolbar, you enter a string and specify the column name in which you want to find the string. FSP searches all the strings in the specified column to find string that matches with the specified string.

The *Find* search bar is regular expression enabled. You can use the alpha-numeric characters, underscore (_), question marks, or asterisks to perform search operation. You must select the *Reg Exp* option in the *Find* search bar to enable the regular expression search feature.

The *Find* functionality bar in the Design Connectivity window contains the following options:



- 1 Enter the text to search.

By default, this text box is a combo text box and displays a list of values of *Status* column. You can also select a value from the drop-down list if you want to search any one of the listed value.

You can also use the alpha-numeric characters, underscore(_), asterisks (*) to perform regular expression based search operation. To enable the regular expression, select *Reg Ex* option.

- 2 Select a column name from the drop-down list in which you want to search the desired text.

For example, type DDR_CS in the search text box and select *Pin/Port Name* from this drop-down list, FSP searches the DDR_CS and highlight the matched text in the *Pin/Port Name* column.

- 3 Displays a list of instance names present on the canvas. Select a instance name from the drop-down list. When you select a instance name, the entered text is searched only in the pin information displayed in the Design Connectivity window. This option is useful to check whether the enter pin name or text exists in the instance.

For example, you want to search for a pin IO_L3N_15 of U15 instance. To search, type IO_L3N_15 in the search text box, select *All Columns* in first combo-box, and select *U15* in the second combo-box.

- 4 Select this option to enable the regular expression based search.

- 5 This option is not displayed by default. This option is displayed when you enter text in search box.

Click to see the previous item in the search list. The previous item will be selected and highlighted in the Design Connectivity window.

- 6 This option is not displayed by default. This option is displayed when you enter text in search box.

Click to see the next item in the search list. The next item will be selected and highlighted in the Design Connectivity window.

- 7 Displays the count of number of text or cell(s) matched.

For example, displays 50 if fifty texts or cells are matched in the Design Connectivity window. *No Matches* is displayed if no matches are found.

- 8 Click to exit the functionality bar.

The *Find* functionality in the Design Connectivity window allows you to search at different levels, that is, column level and instance level.

To locate a text at column level, perform the following steps:

1. Press Ctrl + F or click *Find* in Design Connectivity window toolbar.

The *Find* bar is displayed at bottom of the Design Connectivity window.

2. In the first *Combo-box* select a column name from the drop-down list in which you want to search the specified string.

3. In the second *Combo-box* select *All Instance* from the drop-down list.

Note: Selecting *All Instance* from the drop-down list indicates that the text will be searched in all the instances pin details displayed in the Design Connectivity window.

4. Select the *RegEx* option, if you have used alpha-numeric characters or wildcard characters.

5. In the text box, enter the string that you want to search.

To locate a text at instance level, perform the following steps:

1. In the first *Combo-box* select *All Columns* from the drop-down list.
2. In the second *Combo-box*, select an instance name in which you want to search the text.
3. Select the *RegEx* option, if you have used alpha-numeric characters or wildcard characters.
4. In the text box, enter the string that you want to search.

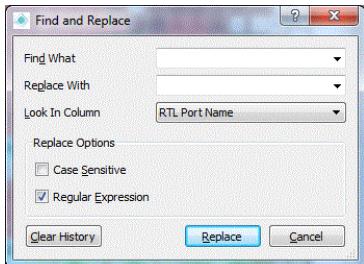


The fields in the *Find* functionality bar must be carefully set. You may get different results or no results for incorrect set.

The following table distinguishes the difference between incorrect and correct scenarios. for searching a pin name Input_CLK in the instance U3.

Case	Incorrect	Correct
1	First combo-box: <i>All Columns</i> Second combo-box: <i>U1</i>	First combo-box: <i>All Columns</i> Second combo-box: <i>U3</i>
2	First combo-box: <i>Status</i> Second combo-box: <i>U3</i>	First combo-box: <i>Port/Pin Name</i> Second combo-box: <i>All Instances</i>
3	First combo-box: <i>Port/Pin Name</i> Second combo-box: <i>U1</i> All the above three scenarios does not provide you the desired result.	First combo-box: <i>Port/Pin Name</i> Second combo-box: <i>U3</i> You may use any one of the scenario from the above three scenarios to search the text.

Search and Replace Text



The *Find and Replace* dialog box provides you a quick way to search for a specific pin name or text and replace all matching text(s) with the new text. You can replace the text(s) across all columns in the Design Connectivity window. However, the *Find and Replace* dialog box will not replace text(s) in read-only cells or columns. The *Find and Replace* dialog box is also regular expression enabled.

To search and replace a text, perform the following steps:

1. Press **Ctrl+H** or click the *Find and replace* icon in the toolbar of the Design Connectivity window.

The Find and Replace dialog box is displayed.

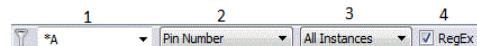
2. In the *Find What* field, enter the pin name or text that you want to search. You can also select an old entry that you have previously searched from the list box.
3. In the *Replace With* field, enter the text or value that you would like to replace the searched text with.
4. In the *Look In Column* field, select a column name from the drop-down list in which you want to search and replace the text.
5. Select *Case Sensitive* option to specify the search as case-sensitive.

When selected, FSP only searches for text that match the exact capitalization as in the text entered in the *Find What* field.

6. Select *Regular Expression* option, if you have used aplha-numeric character, underscore (_), or asterisks (*) in the *Find What* field.
7. Click *Clear History*, to clear the old entries that you previously searched and entered in the *Find What* field.
8. Click *Replace* to replace the text.
9. Click *Cancel* to exit.

The matched text is replaced with the new text.

Filtering the Display of Information in the Design Connectivity window



The Design Connectivity window also provides a filter bar that lets you to filter the display of information according to the following columns values:

- Status
- Pin/Port Name
- Pin Number
- Pin Type
- IO Standard
- Target Pin Property
- Diff.Type
- Diff.Pair Pin
- Serial IO TX/RX Pin

Filtering the display of information lets you focus on working on a specific set of instances or pins in the design. For example, you can filter the display of information in the *Status* column to display the pins that are not allocated or failed to connect. This will give you an opportunity to focus only on those pins which are failed to connect. This is especially useful when you are working with large pin-count devices.

The *Filter* bar is regular expression enabled. You can use the alpha-numeric characters, underscore (_), question marks, or asterisks to filter the display of information. You must select the *Reg Exp* option in the *Filter* bar to enable the regular expression.

The *Filter* bar provides three drop-down combo boxes that shows all the valid values and column names which you can use to filter out the information.

The following table describes the usage of each drop-down combo boxes:

-
- 1 Enter the text that need to be matched and displayed in the Design Connectivity window.

By default, this text box is a combo text box and displays a list of values of *Status* column. You can also select a value from the drop-down list if you want to search any one of the listed value.

You can also use the alpa-numeric characters, underscore(_), asterisks (*) to perform regular expression based search operation. To enable the regular expression, select *Reg Ex* option.

- 2 Select a column name from the drop-down list in which you want to search the desired text.

For example, type DDR_CS in the search text box and select *Pin/Port Name* from this drop-down list, FSP displays only those rows whose *Pin/Port Name* column cells contains DDR_CS text.

- 3 Displays a list of instance names present on the canvas. Select a instance name from the drop-down list. When you select a instance name, the entered text is matched only in the pin information displayed in the Design Connectivity window. This option is useful to check whether the enter pin name or text exists in the instance.

For example, if you want to display all the pins of an instance U15 that contain IO_L3N. To filter, type IO_L3N in the filter text box, select *All Columns* in the first combo-box, and select *U15* in the second combo-box.

- 4 Select this option to enable the regular expression based search.
-

The Design Connectivity Window Toolbar

The Design Connectivity window includes the following toolbar options that provide shortcuts to most of the commonly used features:

Name	Description
Expand Tree	Click to expand all the collapsed rows in the Design Connectivity window.
Collapse Tree	Click to expand all the expanded rows in the Design Connectivity window.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Filter Rows	Click to display the <i>Filter</i> bar at the bottom of the Design Connectivity window.
Find	Click to display the <i>Find</i> bar at the bottom of the Design Connectivity window. The <i>Find</i> dialog box lets you search text, pin name, or strings.
Find and Replace	Click to invoke the <i>Find and Replace</i> dialog box.
Cut	Use this command to cut the text from a cell. The text remains in a paste clipboard until a subsequent cut or paste special command is used.
Copy	Use this command to copy the text from a cell. The text remains in clipboard until a paste command is used.
Paste	Use this command to paste the copied text in a cell.
Show and Hide columns	Click to invoke the <i>Column Chooser</i> dialog box. You can use this dialog box to hide and unhide columns.
Reset Width	Resizes the width of all columns according to the content present in the cell(s).
Clear Highlight	Click to clear the text highlighted in the Design Connectivity window.
Import CSV	Click to invoke the <i>Update from CSV</i> dialog box. The <i>Update from CSV</i> dialog box lets you update the existing design by importing the properties of the instances and pins that are stored in the comma separated values (csv) file.
Export CSV	Click to invoke the <i>Export from CSV</i> dialog box. The <i>Export from CSV</i> dialog box lets you output the complete information that you see in the Design Connectivity window, such as properties of instances and pins to the csv file. In addition, this dialog box also facilitates you with the options, either export the information for the complete design, or export the information for specific instance or protocol.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Canvas Zoom

Use this option to set the canvas zoom settings while browsing through Design Connectivity window. The following options are available:

- To Highlighted Instance

When you select this option and click on any row in the Design Connectivity window. The entire instance is fit in the canvas view to which the selected row belongs. In addition, if the selected row is a pin, the respective pin is highlighted in the instance and if it is a group, all the pins that belongs to the selected group is highlighted in the instance. This is applicable for both device and interface.

- To Highlighted Group/Bank

When you select this option and click on the group or bank row in the Design Connectivity window. The entire group or bank is fit in the canvas view to which the selected row belongs.

- Do Not Change

Select this option, if you do want to highlight or zoom fit the instances or groups/banks.

Organizing the Workspace for Design Connectivity window

In FSP, every window you open is a separate window. These windows displays the design information in different types. For example, canvas displays the design in instances and blocks views, the Design Connectivity window displays the design in spreadsheet view, and Properties displays the properties of each instances and their groups. You can independently control the windows and their views with respect to size, visibility, and location. If you are working simultaneously with several features such as editing properties, adding and deleting nets, or applying externs and power filters and more, it is recommended and useful to have all the respective windows open.

Generally, the windows Design Connectivity and Canvas requires more screen space. These windows can be made full size, floated, or split by using the options provided in the top right corner of the respective windows. After you set, you can also display different areas of the window at different zoom scales.

Note: Zooming areas is not available in the Design Connectivity window.

FSP maintains and displays the selection set across all the windows. For example, anything you select in the Design Connectivity window, the respective object is cross-selected in the rest of the opened windows. Further, in the Design Connectivity window you can select the selection commands to zoom in and highlight the selected pins and groups in the Canvas window.

Each opened window in FSP, can be re-arranged or moved from its current location to another to better utilize the viewing area. You can select the window and drag the view. A rectangular box displays at the background that indicates that the windows is moved from its current location. Dropping one window onto another existing window places the tabs at the bottom of the window. You can switch or activate the windows by selecting the appropriate tabs.

Opening the Windows

Most windows can be invoked using the *Window* menu from the main menu or using keyboard shortcuts.

To open a window:

- From the *Window* menu, choose the window name that you want to open.
For example, to invoke *Properties* choose *Window – Properties* or press *Ctrl + 6*.

Setting the Windows

FSP provides options to position, move, and orient the different windows in FSP workspace. This is a useful feature when you are working with multiple windows simultaneously.

For example, you might need to have the Canvas, the Design Connectivity, and the Properties together. That means, you want to place the Canvas on the left, the Design Connectivity window in the middle, and the Properties at the right of FSP workspace.

To allow you to move and arrange the FSP workspace, you can dock the windows and set them floating across different screens. You can access these options from the top-right corner of the respective windows. You can also do it manually by selecting and dragging the windows.

To dock and move the window, perform the following step:

1. Perform any one of the following:
 - Click *Maximize* icon at top-right corner of the window.
- Or

- Left click on the top of the window and drag the window.

The light blue outline appears at back of the window. This outline serves as a guide.

2. Drag the window to a location you want and release the left mouse button.

Cross-Probing between Design Connectivity and Other Windows

In complex designs, a large number of signals may be aliased to each other on the same instance or different instances present in the Canvas. This makes it difficult to quickly identify the signals and view their connectivity for debugging purpose. FPGA System Planner lets you select a net in the Design Connectivity window and view all synonyms of the selected across the opened windows. You can select any row in the Design Connectivity window such as device, banks, interface, and groups, to view it in the Canvas, Properties, and Die View.

When you select an instance or a net in the Canvas, the details of the selected objects are displayed in the Design Connectivity window and in the Properties. In the same way, when you select any row in the Design Connectivity window, the respective instance to which the selected row belongs to, is highlighted in the canvas; however, these highlighted instances can be small and difficult to see in the Canvas because of the current zoom-level of the canvas or the screen size.

FSP provides a *Canvas Zoom* options in the toolbar in Design Connectivity window. You can set the zoom factor of the canvas by selecting any one of the canvas zoom options. When the zoom factor is set, FSP automatically corrects the zoom factor of the canvas and, fits and highlight the selected instance in the canvas. You can also deselect these options if you do not wish to highlight the selected instance, nets or pins in the canvas.

The section contains the following topics:

- Navigating Signal
- Navigating Design
- Navigating Instances
- Navigating Group and Banks
- Navigating Pins

Important

To navigate the pins, signals, device and interface instances across the opened windows, it is recommended that you organize your workspace as explained in the [Organizing the Workspace for Design Connectivity window](#) section.

Navigating Signals

The Design Connectivity window lets you quickly navigate the signals to view its connectivity in the Canvas. You can navigate to the signals by clicking on a pin row in the Design Connectivity window. To navigate to a signal, make sure that you click on a pin that is connected or has *Allocated* in the Status column. If you click on a unrouted pin in the Design Connectivity window, the corresponding pin is highlighted and zoomfit in the Canvas. Besides highlighting the signals in the Canvas, FSP also highlights the instances and the pins to which the signal is connected.

You can also navigate the signals in the following way:

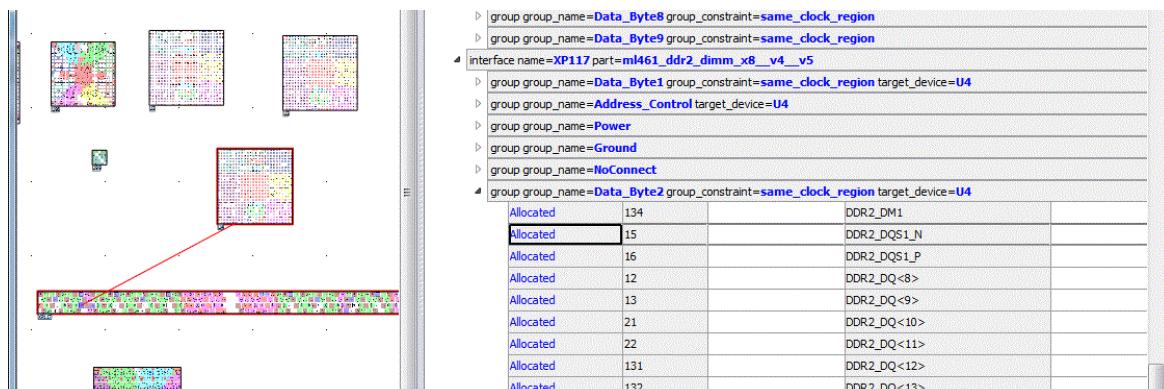
- A group of signals from different interface groups
- A group of signals from the same group
- A group of signals from one or more groups
- A complete interface signals

To start navigating signals, you must enable the net navigation mode. The enable net navigation mode is available as one of the context-sensitive menu option. You can click anywhere over the Design Connectivity window to access and enable this mode. After you enable the net navigation mode and you can start clicking on the rows to navigate the signals. When you enable the net navigation mode, FSP automatically hides the entire design nets. Now when you start clicking on the rows in Design Connectivity, FSP displays the corresponding net(s) in the Canvas.

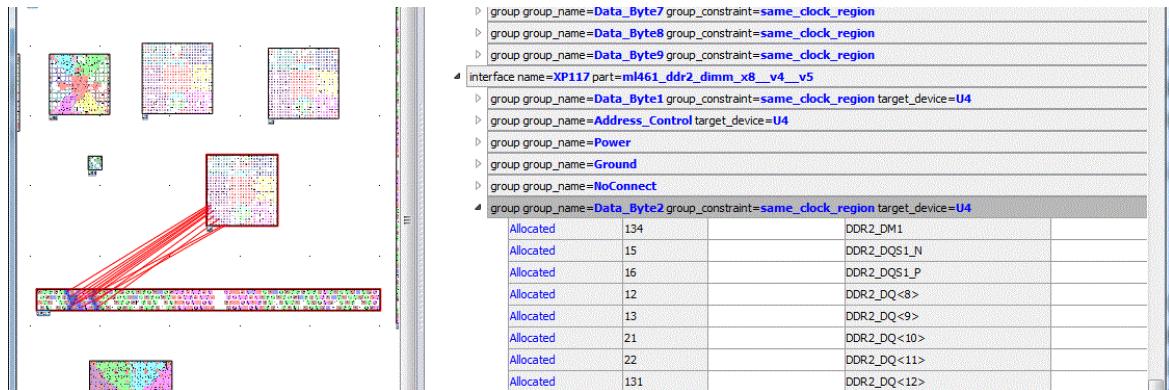
You can disable the net navigation mode at any time during the design.

The following section demonstrate examples of navigating a signal and navigating a group of signals.

Navigating a signal



Navigating a group of signals



Navigating Design

Note: To navigate the complete design, you must select any one from the options, *To Highlighted Instances* and *To Highlighted Group/Bank* in the *Canvas zoom* toolbar option.

When you select the design level node in the Design Connectivity window, all the instances present in the design are highlighted and zoomfit in the canvas. In addition, the complete information about each instances and the design description is displayed in the Properties.

Navigating Instance(s)

Note: To navigate an instance(s), you must select the option *To Highlighted Instances* in the *Canvas Zoom* toolbar option.

Note: During navigating the instance(s), if you do not want the signals to be highlighted, disable the net navigation by right-clicking anywhere over the Design Connectivity and choose *Disable Net Navigation Mode*.

To highlight a instance, perform the following step:

- Select a device or interface instance row in the Design Connectivity.

To highlight instances, perform the following steps:

- Press and hold the Ctrl and left-click on the device or interface instance rows.

The instances with the nets and connected pins are highlighted and zoomfit simultaneously.

Navigating Group(s) and Bank(s)

Note: To highlight group(s) or bank(s), you must select the option *To Highlighted Group/Bank* in the *Canvas Zoom* toolbar option.

Note: During navigating the instance(s), if you do not want the signals to be navigated, disable the net navigation by right-clicking anywhere over the Design Connectivity and choose *Disable Net Navigation Mode*.

To highlight a group:

- Click on a group row.

To highlight the groups:

- Press and hold the Ctrl and left-click on the groups rows.

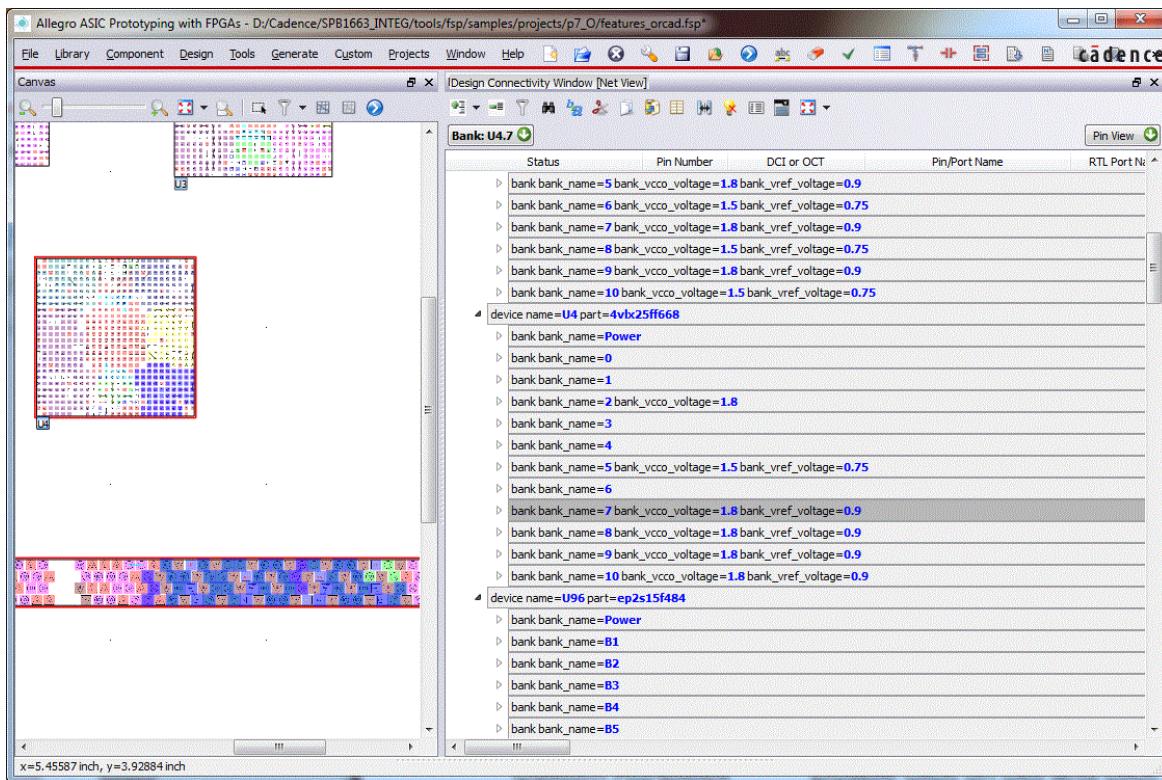


To highlight a bank:

- Click on the bank row. See the following figure.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner



To highlight the banks, perform the following step:

- Press and hold Ctrl and left-click on the banks rows.

Navigating Pin(s)

Note: To highlight the pin(s), make sure that the any one of the option, *To Highlighted Instances* and *To Highlighted Group/Bank* in the *Canvas zoom* toolbar is selected.

Note: During navigating the instance(s), if you do not want the signals to be navigated, disable the net navigation. Right-clicking anywhere over the Design Connectivity and choose *Disable Net Navigation Mode*.

To highlight a pin:

- Click on a pin row. See the below figure.

To highlight pins:

- Press and hold Ctrl and left-click on the pin rows.

Or

- a. Click on a pin row.
- b. Press Shift and click on the pin rows.

Context-Sensitive Menus

The Design Connectivity window offers various context-sensitive menus. Infact, every node in the Design Connectivity has a set of context-sensitive menu attached to it. These menus are useful to perform certain operations that are relevant to the selected node. In addition, these menus helps you work faster. You can access these menus by right-clicking on the node or by clicking the button at the top-left corner of the Design Connectivity window. The menu's list in this button varies based on the row you select in the Design Connectivity window.

Note: The context-sensitive menus are similar in both in the Design Connectivity window and in the Canvas.

The following section outlines the description of each context-sensitive menus and steps to access:

Header Column

To access the context-sensitive menus of the design, perform the following steps:

- Right-click on the *Status* header.

Menu Name	Description
-----------	-------------

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Freeze upto column #<row number>(column name)	This option is useful, when you want to keep a column of the Design Connectivity window visible while you scroll to another column of the Design Connectivity window. You can lock specific columns that you do want to scroll left and right. For example, you might want to keep <i>Pin Number</i> and <i>Pin/Port Name</i> column visible as you scroll. After you freeze columns, a thick line appears. This thick line indicates that the <i>Pin Number</i> and <i>Pin/Port Name</i> column is frozen to keep columns in place when you scroll. To lock rows, do the following: <ol style="list-style-type: none">1. Select the column to the right of the column or columns that you want to keep visible when you scroll and right-click.2. Choose <i>Freeze upto column #<row number>(column name)</i> option. The option displays the selected row number followed by the name of the selected column. For example, Freeze upto column #2<Pin Number>.
Scroll Column	Displays a list of column names. Select a column name to quickly navigate to the selected column.
Unfreeze All Columns	Click to unfreeze the locked columns.
Hide Current Column	Lets you hide the column that you do want to see in the <i>Design Connectivity</i> window. This option is not available for the <i>Status</i> column.

Design

To access the context-sensitive menus of the design, perform the following steps:

- In the Design Connectivity window, do any one of the following:
 - Right-click on the design level node.
 - Left-click on the device level node and click the button at top-left corner.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Menu Name	Description
Run	Invokes the <i>Process Options Editor</i> dialog box. Use this dialog box to run the complete design.
Project Settings	Invokes the <i>Project Settings</i> dialog box.
Check Consistency	Use this option to check the consistency in the design. For example, checks part definition for instances present in the Canvas and whether schematic symbol exists for the instance. After checking, displays a design consistency report in the Log window.
Abstract View	Invokes the Abstract view.
Disable/Enable Net Navigation Mode	Use this option to enable and disable the net navigation mode.
Show All Pins (In Design Connectivity)	This option lets you display all the pins and its details in the Design Connectivity window irrespective of view you are working in. For example, the <i>Power Pins</i> view displays all the power pins and there details of the instances present in the Canvas. When you click this option, FSP displays the entire pins and their details in the Design Connectivity window.
Nets	Displays a list of options. You can use these options to perform certain operations on complete nets of the current design. For example, click <i>Hide</i> to hide all the nets of the design, click <i>Lock</i> to lock all the nets, and more.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Create NetGroups Displays the following sub menus when you place the mouse pointer.

■ Group Wise

Defines NetGroups for interface protocol in `<interface_instance_name>_<group_name_interface>` order and for device protocol in `<device_instance_name1>_<device_instance_name2>_<group_name>` order.

For example, for an interface U1 as instance name with group1, group2 and group3 as group names, the NetGroups name order would be as following:

- `<U1>_<group1>` for all the pins of group1
- `<U1>_<group2>` for all the pins of group2
- `<U1>_<group3>` for all the pins of group3

■ Interface /Protocol Wise

Defines NetGroups using interface instance name for interface component and device protocol name for device protocol.

For example,

- NetGroups for interface components are defined as `<instance_name>_0, <instance_name>_1...` so on.
- NetGroups for device protocols are defined as `<device_protocol_name>_0, <device_protocol_name>_1...` so on.

Note: “`_0, _1..`” are applied based on the maximum NetGroups size specified in the NetGroups tab of the Settings dialog box. For example, assume that you have 32-bit size of protocol and you specify 16 as maximum NetGroup size, then the *Interface/Protocol wise* option defines the NetGroups as:

- `<instance_name>_0` for first 16-bit.
- `<instance_name>_1` for second 16-bit.

■ Swappable Pins

Assigns the same NetGroups on the pins which are swappable.

Example,

Following is one of the case:

CQ_P - U1_readgroup0

CQ_N - U1_readgroup0

Q_<0> - U1_readgroup1

Q_<1> - U1_readgroup1

Q_<2> - U1_readgroup1

From example above, since CQ_N and CQ_P are swappable pins same NetGroups are assigned.

Q_<0>..<2> are swappable pins and same NetGroups are assigned to them.

Device Instance

To access the context-sensitive menus for device instance, perform the following steps:

- In the Design Connectivity window, do any one of the following:
 - Right-click on the device level node.
 - Left-click on the device level node and click the button at top-left corner.
- In the Canvas, do the following:
 - Right-click on the device instance.

Menu Name	Description
Device <Instance_name>	Displays the instance name of the selected device.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Run	<p>Generates the connection between the selected device and the interface targeted to this device.</p> <p>When you select this option, a confirmation window is displayed prompting you about processing the selected instance. Click <i>Yes</i> to run the selected instance and <i>No</i> to cancel.</p>
Bank Settings	<p>Invokes the <i>Bank Settings for Device Instance <Inst_name></i> dialog box. This dialog box provides you the pin count details in each bank of the selected device. You also have the option to preserve the dedicated pins such as VREF, VRP/VRN, RUP/RDN, and more, using this dialog box.</p>
Manage Protocols	<p>Invokes the <i>Manage Protocols</i> dialog box. This dialog box lets you create a new protocol between the selected device and the other devices, modify or delete the existing device protocol.</p>
Protocols	<p>This option offers you a quick way to create a protocol between the selected device and the other devices.</p> <p>When you place the mouse pointer on this option, a list of device instance names present in the canvas is displayed. Select a instance name from the list to create protocol. The <i>Edit Protocol</i> dialog box appears, when you select an instance name.</p> <p>One more option <i>Multi Point</i> appears at the end of the device instance names list. Select this option to create a multi point protocol between the selected device and the other devices.</p>
Virtual Interface	<p>Displays the following options:</p> <ul style="list-style-type: none">■ Create New Virtual Interface<p>Select this option to invoke the <i>Define Virtual Interface for Device Instance <Inst_Name></i> dialog box. Use this dialog box to create virtual interface.</p>■ Create New Virtual Interface from HDL<p>Select this option to invoke the <i>Create Virtual Interface(s) from Verilog/VHDL for Device Instnace <Inst_Name></i> dialog box. Use this dialog box to create multiple virtual interface with HDL.</p>

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Constraints	Displays the following features list: <ul style="list-style-type: none">■ Map FPGA Ports and Pins■ Map FPGA Resources■ Import Constraints■ Export Constraints■ Show BUFR Nets
Update Instance Footprint	Invokes the <i>Update Pin Locations from PCB Footprint</i> dialog box. Use this dialog box to update the device pin locations using external PCB footprint (dra) file.
Link to Schematic Symbol	If you are in DE-HDL schematic environment, this option invokes the <i>Component Browser</i> . If you are in OrCAD schematic environment, this option invokes the <i>Add Part</i> dialog box.
Update ' <instance_name>' Hierarchical block</instance_name>	Updates the hierarchical schematic pages of the selected device. When you select this option, a confirmation window is displayed prompting you about updating the hierarchical symbol for selected instance. Click <i>Yes</i> to update and <i>No</i> to cancel.
Report	Displays the following options: <ul style="list-style-type: none">■ To Log Window Select this option to generate the design report for selected instance in the <i>Log</i> window.■ To Text File Select this option to output the design report for the selected instance in a text file.
Change FPGA	Invokes the <i>Change FPGA</i> dialog box. Use this dialog box to replace the selected device with any one of the device from the same device family.
Delete Instance	Deletes the selected instance from the Canvas.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Export to CSV	Invokes the <i>Export CSV</i> dialog box. Use this dialog box to export the pin and group constraints of the selected device. For more information, see the Exporting Part and Pin Properties section.
Nets	<p>Displays the following options:</p> <ul style="list-style-type: none">■ Show■ Hide■ Lock■ Unlock■ Lock Constraint Pins■ Unlock Constraint Pins■ Lock Clocks■ Unlock Clocks■ Delete All■ Select All <p>Note: The above options are applicable to the nets of the selected device instance.</p>
Disable Net Navigation Mode	Click on this option to disable the net navigation mode. After selecting this option, FSP stops highlighting and zoom fit the instance on canvas through Design Connectivity.
Show Only Modified Pins	Click this to display a list of pins that are modified. After selecting this option, the Design Connectivity window hides all the pins details of the components expect the modified pins.
Show All Pins	Click this to display the pin details of all pins of the components present in the canvas.
Expand	Click to expand all hidden rows of the selected device instance.
Collapse	Click to collapse all unhide rows of the selected device.

Device Instance Bank

To access the context-sensitive menus for device instance's bank, perform the following steps:

- In the Design Connectivity window, do any one of the following:
 - Right-click on the device bank node.
 - Left-click on the device bank node and click the button at top-left corner.

Menu Name	Description
Bank <Inst_Name>. <Bank_N ame>	Displays the instance name followed by name of the selected bank.
Nets	For more information, see the Device Instance section. Note: These options are applicable only to the nets of the selected bank.
Disable/Enable Net Navigation Mode	Click on this option to disable or enable the net navigation mode.
Show Only Modified Pins	Click to see the pins that are modified.
Show All Pins	Click to see all the pins.
Expand	Click to expand all hidden rows of the selected device instance.
Collapse	Click to collapse all unhide rows of the selected device.

Device Instance Pin

To access the context-sensitive menus for device instance's pin, perform the following steps:

- In the Design Connectivity window, expand the device instance and the bank nodes and do any one of the following:
 - Right-click on any row.
 - Left-click on a row and click the button at top-left corner.

Menu Name	Description
Pin/Port Name:<instance_name> >.<bank_name>.<pin_name>	Displays the name of the selected interface, selected bank, and the selected pin.
Lock Selected Nets	Click to lock the selected net. This option is useful only when the net is connected to the selected pin.
Unlock Selected Nets	Click to unlock the selected net. This option is useful only when the net is connected to the selected pin.
Delete Selected Nets	Click to delete the selected net. This option is useful only when the net is connected to the selected pin.
Disable/Enable Net Navigation Mode	Click on this option to disable or enable the net navigation mode.
Select In	Displays the following options. You can select any one of the options to select the pins. <ul style="list-style-type: none"> ■ Bus <bus_name in the current group> ■ Group <group name> ■ Interface <name>
Find <name>	Displays the value of the selected cell. For example, if you right-click on a cell with the “Unallocated” value under the <i>Status</i> column, you will see <i>Find “Unallocated”</i> . Lets you search for the displayed value, that is, <i>Find “value”</i> , in all the columns. After you click this option, all the cells that contains the displayed value is highlighted.
Filter Allocated	Use this option to filter and display the cells that contains the displayed vale.

Interface Instance

To access the context-sensitive menus for device instance, perform the following steps:

- In the Design Connectivity window, do any one of the following:
 - Right-click on the interface node.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

- Left-click on the interface node and click the button at top-left corner.
- In the Canvas, do the following:
 - Right-click on the interface instance.

Menu Name	Description
Interface:<instance _name>	Displays the name of the selected interface instance.
View Rules	Click to invoke the Rules Editor. This Rules Editor is a read-only editor.
Edit Rules	Click to invoke the Rules Editor.
Run	Click to make the connections. After you click this option, a confirmation message is displayed prompting you about making connections for the selected interface.
Re-Run	Click to re-run the interface.
Group Settings	Click to invoke the <i>Groups Settings For Interface Instance <inst_name></i> dialog box.
Update Instance Footprint	Invokes the <i>Update Pin Locations From PCB Footprint</i> dialog box.
Convert to Real Interface	Invokes the <i>Convert Rules File Instance to Real Part</i> dialog box.
Report	Use this option to generate summary of a report about the selected instance.
Delete Instance	Click to delete the selected instance. After you select this option, the selected instance will also be removed from the Canvas.
Export to CSV	Click to invoke <i>Export to CSV</i> dialog box.
Nets	For more information, see the <u>Device Instance</u> section.
Disable/Enable Net Navigation Mode	Click to disable or enable the net navigation mode.
Show Only Modified Pins	Click to see only those pins that were modified during importing csv process.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Show All Pins	Click to see all the pins. Pins that are modified and not modified.
Expand	Click to expand all hidden rows of the selected device instance.
Collapse	Click to collapse all unhide rows of the selected device.

Interface Instance Group

To access the context-sensitive menus for interface instance group, perform the following steps:

- In the Design Connectivity window, expand the interface node and do any one of the following:
 - Right-click on the interface group node.
 - Left-click on the interface group node and click the button at top-left corner.

Menu Name	Description
Group : <Instance_Nam e>.<group_name>	Displays the name of the selected interface instance followed by the selected group name.
Swap Groups	Click to invoke the <i>Swap Group Virtual</i> dialog box.
Nets	For more information, see the <u>Device Instance</u> section.
Disable/Enable Net Navigation Mode	Click to disable or enable the net navigation mode.
Show Only Modified Pins	Click to see only those pins that were modified during importing csv process.
Show All Pins	Click to see all the pins. Pins that are modified and not modified.
Expand	Click to expand all hidden rows of the selected device instance.
Collapse	Click to collapse all unhide rows of the selected device.

Interface Instance Pin

To access the context-sensitive menus for interface instance's pin, perform the following steps:

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

- In the Design Connectivity window, expand the interface instance and the group nodes and do any one of the following:
 - Right-click on any row.
 - Left-click on a row and click the button at top-left corner.

Menu Name	Description
Pin/Port Name:<instance_name> >.<bank_name>.<pin_name>	Displays the names of the selected instance, selected bank, and selected pin.
Lock Selected Nets	Click to lock the selected net. This option is useful only when the net is connected to the selected pin.
Unlock Selected Nets	Click to unlock the selected net. This option is useful only when the net is connected to the selected pin.
Delete Selected Nets	Click to delete the selected net. This option is useful only when the net is connected to the selected pin.
Disable/Enable Net Navigation Mode	Click on this option to disable or enable the net navigation mode.
Select In	Displays the following options. You can select any one of the options to select the pins. <ul style="list-style-type: none">■ Bus <bus_name in the current group>■ Group <group name>■ Interface <name>
Find <name>	Displays the value of the selected cell. For example, if you right-click on a cell with the “Unallocated” value under the <i>Status</i> column, you will see <i>Find “Unallocated”</i> .
	Lets you search for the displayed value, that is, <i>Find “value”</i> , in all the columns. After you click this option, all the cells that contains the displayed value is highlighted.
Filter Allocated	Use this option to filter and display the cells that contains the displayed vale.

Accessing Mini Toolbar

The Design Connectivity window provides a shortcut toolbar. This toolbar is useful to perform certain operations that are relevant to the selected node. In addition, this toolbar helps you work faster. You can access this toolbar by clicking on the node. The icons in the toolbar may vary based on the row you select in the Design Connectivity window.

The following table displays few examples of accessing shortcut toolbar at different nodes.

This toolbar appears when you click on the bank or group level node.



This toolbar appears when you click on the pin level node.

Allocated	A20	Output	HSTL_II
Allocated	B20	Output	HSTL_II
Allocated	A5	Input	HSTL_II
Allocated	B5	Input	HSTL_II
Allocated	C5	Input	HSTL_II
Allocated			-_II
Allocated		✖️	-_II
Allocated	A7	Input	HSTL_II
Allocated	D7	Input	HSTL_II

Switching Views in Design Connectivity

The Design Connectivity window provides a spreadsheet-based user interface that allows you to capture and maintain the connectivity information in the design. The Design Connectivity window also provides multiple views to capture the design. The Design Connectivity window is divided into different functional views. Each view represents the different aspects of the design information. For example, *Pin View* lets you modify and maintain the pin details of all the components that are present in the Canvas and *Net View* lets you modify and maintain the connectivity information of the design. You can also modify and maintain the connectivity information in the *Pin View*. However, if the design is dominated by large pin-count devices, you may have to spend more amount of time to work on a single spreadsheet, that is, in the *Pin View*.

The Design Connectivity window provides a more efficient way of capturing the design information in different views. Using these different views in the Design Connectivity window allow you to concurrently modify different aspects of the design and synchronize the information as and when required.

The Design Connectivity window hide and unhide the pins (or rows) based on the view you are working on. For example, the *Net View* displays the pins that are used for connections, the *Preserve Pins View* displays the pins that are preserved, the *Power Mapping View* displays the power pins and so on. You can use a *view* button at the top-right corner of the window to switch between the views. You can switch to any view during the design.

The Default Pin View

By default, all the columns are displayed in the *Pin View*. When you switch to a different view, not all the columns are displayed. The columns which are useful to work in the selected view are displayed and remaining are hide. In all the views, you have the option to hide and unhide the columns using the *Column Chooser* dialog box. Any appearance changes you make in the view, such as hiding and unhiding columns, resetting width of the columns, freezing columns, and more, are saved when you close the *Design Connectivity* or FSP or switch to different views. The next time you open FSP or invoke the *Design Connectivity*, or switch to the view, you will see your last settings.

The following table describe the usage of each views:

View Name	Description
Pin View	The <i>Pin View</i> is displayed by default, when you first time invoke the Design Connectivity window. The <i>Pin View</i> displays the complete details of all the components that are present in the Canvas. You can use the <i>Pin View</i> to modify any properties of the instances or information of the design.
The Net View	Displays the pins that are used for connections or routed and hides rest of the pins.
Synthesis Failure Pins	Displays the pins that are failed to connect during synthesis.
Externs and Interns	Displays the pins that are marked as external port in the <i>Connection Type</i> column.
Power Pins	Displays the power pins (mapped or unmapped) of all the instances present in the Canvas.
Clock Nets	Displays the clock pins of all the instances present in the Canvas. Note: Displays a blank window, in case none of the instances contain clock pins.

High Speed Signals	Displays the properties of the high speed pins of all the device instances present in the canvas.
	Note: Displays a blank window, in case the devices does not contains high speed pins.
Pin With Termination	Displays the properties of the pins to which the terminations are applied.

To switch to a different view, perform the following steps:

1. Click  at the top-right corner of the Design Connectivity window.
A list of views are displayed.
2. Select any one of the view from the list.

The pin details are displayed in the Design Connectivity window as per the selected view.

The Status Bar

The Status Bar is available at the bottom by default. The Status Bar provides you a quick information about a pin or a net present on the Canvas. For instance, hover the mouse pointer on the pins, the Status Bar shows the name of the pin on the left side (Pin:) . If the pin is connected to a net, net name is displayed at the right side (Net:). You can also hover the mouse pointer on the nets to see the net name in the Status Bar.

The Properties Window

The Properties window lets you view and edit logical properties of the instance in a project.

Using the Properties Window, you can edit:

- Device properties
- Device bank properties
- Device pin properties
- Interface properties
- Interface group properties
- Interface pin properties
- Nets

- Ports
- Terminations

The *Properties* window appears at the right side of the tool when you click the *Properties* button in the toolbar. The Properties window can be repositioned anywhere in the tool. Click the *Maximize* button in the upper right corner of the window and drag anywhere on the perimeter of the tool. When you resize and reposition the Properties window, it remains at the size and position you have established whenever you close and reopen the design.

In the Properties window , each row is a property. Each row contains a property name and the property value. The property names and values are displayed in grey color. The grey color indicates that you cannot edit the properties. However, not all the property values are displayed in grey color. Some of them appear in white color. The white color indicates that you can edit, add, and select a value from the drop-down list.

Note: Some of the property cells in white color provides drop-down selection option. This means you can select a value from the drop-down list.

Note: The display of number of rows in the Properties can be customized using the *Show Hide/Columns* option. This option is available in the toolbar in Design Connectivity.

The properties that appear in the Properties depends on the items selected in the Design Connectivity window. Also, these properties depend on the item selected in the Canvas. For example, if you click on a pin, the properties of the selected pin appear in the Properties. When you modify the properties in the Properties, the changes are immediately reflected in the Design Connectivity window and in Canvas. For example, if you modify a net name in the *Properties*, the revised net name is updated in *Design Connectivity* and *Canvas*. To see the change in the *Canvas*, click on a net using the left mouse button. The *Net Info* displays the revised net name. In addition, if you perform any changes in the Design Connectivity window or in Canvas, these changes are also reflected in the Properties.

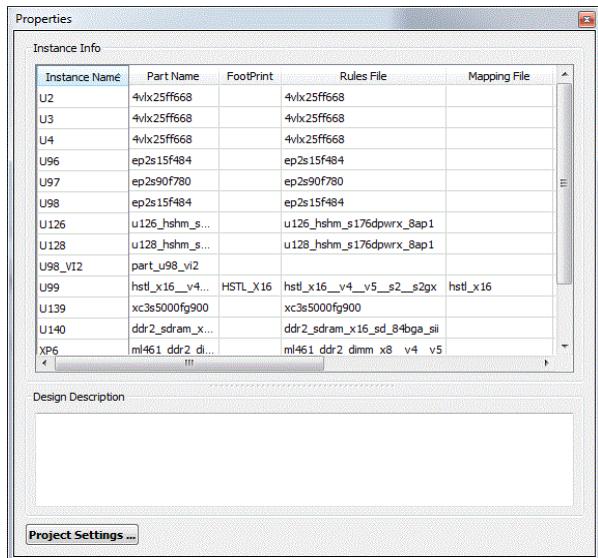
Note: When you add, edit, or delete property values using any forms in FSP, the changes are immediately reflected in *Design Connectivity*, *Canvas*, and *Properties*. For example, if you modify the net name using *Change Net Names* dialog box. The revised net name is update in Design Connectivity and Canvas.

The following section illustrates what appears in the Properties window when you click or select in Design Connectivity and in Canvas.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Design Information Page



This is the default view of the Properties. When you first time invoke the Properties or click the *Design [view name]* node, you see this page. This page displays the instance information and design description of the current design. The *Instance Info* displays a list of instance name and its associated information in a spreadsheet view. The *Instance Info* pane is a read-only pane. When you perform changes in the canvas or in any forms in FSP such as adding or deleting instance, editing instance names, rotating or flipping instances, or modifying the schematic symbol path, these changes are immediately reflected in the *Instance Info* pane.

The following table provides the description of each column in the *Instance Info* pane.

Name	Description
Instance Name	Displays a list of instance names that are present on the canvas.
Part Name	Displays the part name of the instance.
FootPrint	Displays the footprint name.
Rules File	Displays the rules file name of the instance.
Mapping File	Displays the mapping file name of the instance.
Schematic Symbol	Displays the <i>lib:cell:view</i> structure of the instance.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

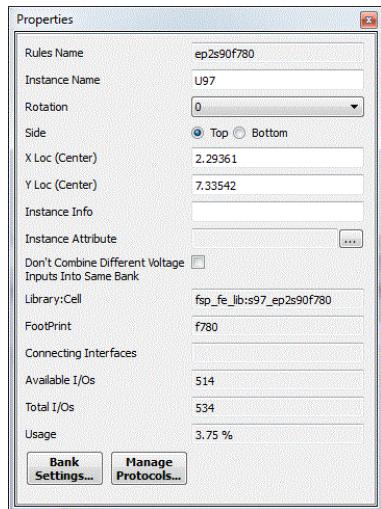
Rotation	Displays the current rotation angle of the instance
Flip	Displays the flip status of the instance. Displays <i>No</i> if the instance is not flipped and <i>Yes</i> if flipped.

The *Design Description* pane lets you add information. The design information can include any text information such as parts information, board dimensions, and so on. This design information is saved in the project database so that you can see the information at any time when you invoke the *Properties*.

The Properties provides options to quickly access the *Project Settings* and *User Preferences* dialog boxes. These options are available at bottom left of the Properties.

Note: The two options, *Project Settings* and *User Preferences* options are not available when you are editing properties of the instances and pins in the Properties.

Device Instance Properties



This page appears when you do any one of the following:

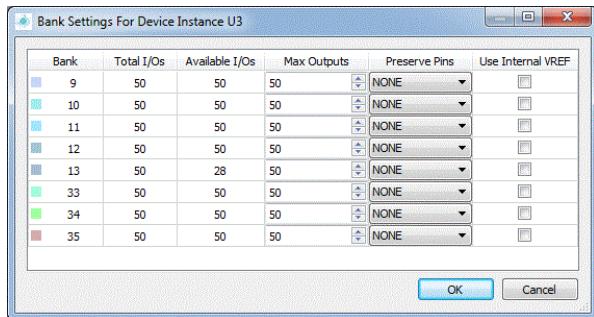
- Click the *Device Name = <inst_name> part = <name>* row in the *Design Connectivity* window.
- In Canvas, right-click on the device instance and choose *Instance Properties*.

You can use this page to modify the properties of the device instance. For example, you can change the instance name in the *Instance Name* field, add information in the *Instance Info* field, flip the device instance by selecting the options in the *Side* field, and more.

Allegro FPGA System Planner User Guide

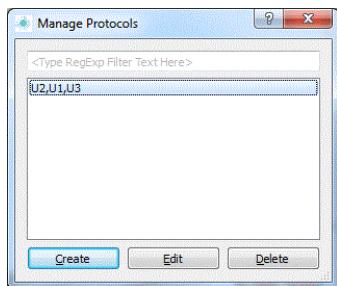
Getting Started with FPGA System Planner

There are two more options, *Bank Settings* and *Manage Protocol* available in this page. Click *Bank Settings* to invoke the *Bank Settings for Device Instance <inst_name>* dialog box.



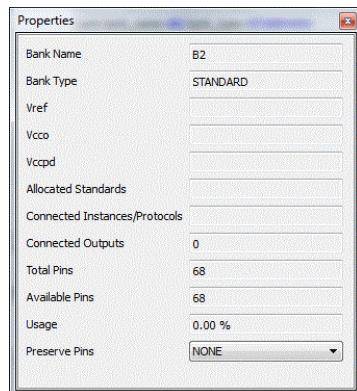
This dialog box displays the bank details such as names of the banks, number of I/Os in each banks, and available I/Os of the selected device instance. In this dialog box, you can also set the maximum outputs and specify the power pins to preserve.

Click *Manage Protocols* to invoke the *Manage Protocols* dialog box.



This dialog box lets you to create a protocol between the devices, modify or delete the existing protocol.

Device Instance Bank Properties

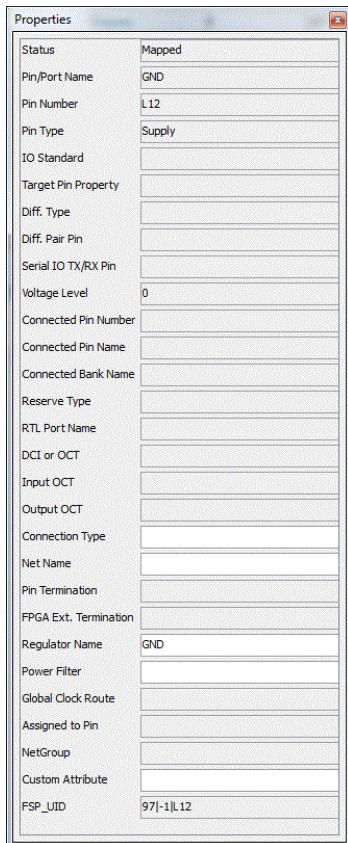


This page appears by performing any one of the following:

- In the Design Connectivity window, expand the device level by clicking the plus sign (+) to the left of the row and click the *bank bank_name = <name>* node.
- In the Canvas, press Shift and click on a pin from the bank that you want to select.

This page provides you the details of the selected bank. For example, reference voltage (VREF) is displayed in the VREF field, name of the I/O standard allocated to this bank appears in the *Allocated Standards* field and more.

Device Instance Pin Properties

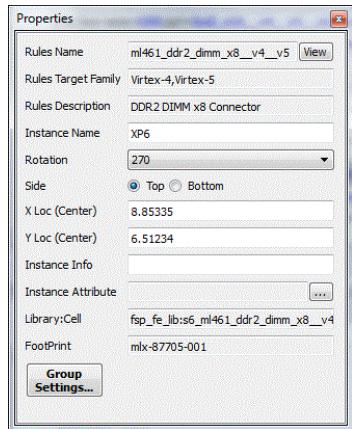


This page appears by performing any one of the following:

- In the Design Connectivity window, expand the bank level by clicking the plus sign (+) to the left of the node and click any pin from the list.
- In the Canvas, double-click on a pin of a device instance.

This page provides you the liberty to modify the properties of the selected pin. However, not all the fields are editable. The columns that are editable in the Design Connectivity window, is also available as editable in this page. For example, RTL port name can be modified in *RTL Port Name* column in the Design Connectivity window and in *RTL Port Name* cell in this page.

Interface Instance Properties



This page appears when you perform any one of the following steps:

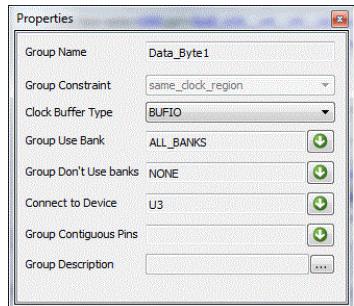
- In the Design Connectivity window, click the *interface name = <inst_name> part = <name>* node.
- In the Canvas, right-click on the interface instance and choose *Instance Properties*.

This page provides you the information about the selected interface. For example, rules name in the *Rules Name* field, device name to which the selected interface is targeted in the *Rules Target Family* field and more. You can also use this page to modify the properties of the instance. For example, modify the instance name in the *Instance Name* field, flip the selected interface by selecting options in the *Side* field.

This page also lets you modify the settings of the selected interface groups. Click *Group Settings* to invoke the *Group Settings For Interface Instance <inst_name>* dialog box. This dialog box provides the following options:

- Connect to Device
 - Use this option to target the group to device instance.
- Use Banks
 - Use this option to target the group to a specific bank of the targeted device.
- Don't Use Bank
 - Use this option to specify the bank of the targeted device to which you do not want the group to be connected.

Interface Instance Group Properties

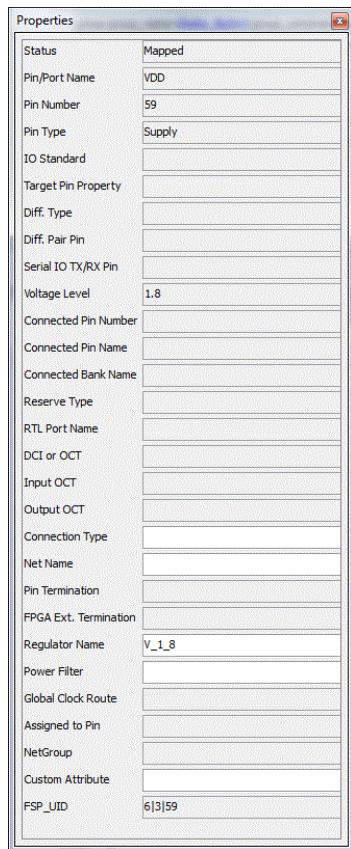


This page appears when you perform any one of the following step:

- In the Design Connectivity, when you expand the interface level by clicking the plus sign (+) to the left of the node and click the *group group_name = <name>* row.
- In the Canvas, press and hold Shift key, and click on a pin of interface instance.

This page lets you modify the properties of the selected group. For example, specify the bank name to which the group you want to target in the *Group Use Bank* field, add description about the group in the *Group Description* field and more.

Interface Instance Pin Properties



This page appears when you perform any one of the following step:

- In the Design Connectivity window, expand the interface group level by clicking the plus sign (+) to the left of the row and click any pin from the list.
- In the Canvas, double-click on a pin of the interface instance.

This page lets you modify the properties of the selected pin. However, not all the fields are editable. The columns that are editable in the Design Connectivity window, is also available as editable in this page. For example, RTL port name can be modified in *RTL Port Name* column in the Design Connectivity window and in *RTL Port Name* cell in this page.

The Die View

The Die view provides you a graphical interface which is used for design analysis. You can also use the Die view to plan I/O assignments early and validate the pin assignments after synthesis.

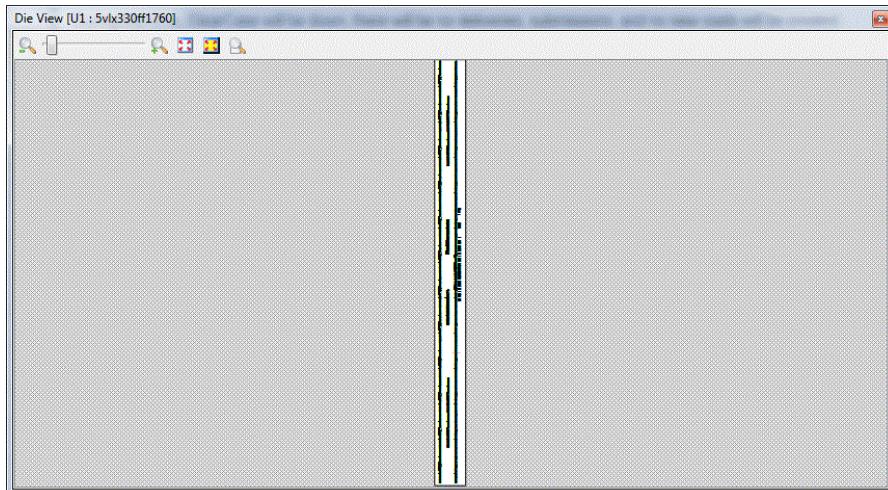
The Die view displays the I/O resources available in a selected device. The I/O resources display in the Die view is based on the device-specific FPGAs. For example, for Xilinx FPGAs the I/O pins are displayed in column wise and for Altera FPGA's, the device view is broken into smaller rectangles called *Tiles*. These pins and tiles are displayed based on the I/O pads information provided by the respective FPGA vendors.

Note: The *Die view* is unsupported for Actel FPGA's.

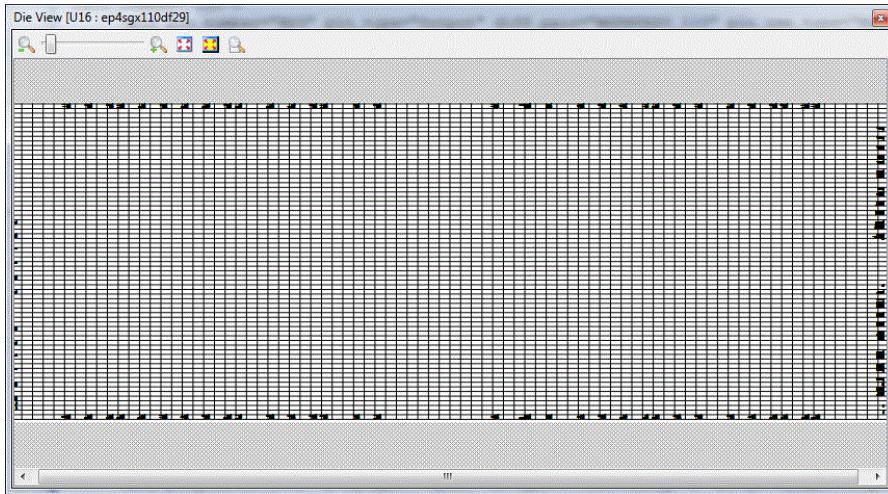
The following figure shows the die view of Xilinx and Altera FPGAs.

Note: When you hover mouse over the pins, the other pins in the bank to which the selected pin belong are highlighted.

For Xilinx FPGA



For Altera FPGA



When you first time invoke the Die view, it displays a blank window. You must select a device in the Canvas or in the *Design Connectivity* window to see the I/O resources. When displayed these I/O resources are not visible until you zoom. The Die view provides various zoom options at the upper left corner. Use these options to increase or decrease the zoom level of the Die view.

By default, the pin numbers and pin names are displayed with the I/Os in the *Die view*. The Die view gives you the control to turn off and on the display of the pin informations. You can also access other pin informations such as bank name, net name and so on, using the right mouse button options. The following table describes the usage of each right mouse button options:

Name	Description
Show Pin Name	By default this option is selected when you first time invoke the Die view. However, you can uncheck this option to hide the pin names.
Show Pin Number	By default this option is selected when you first time invoke the Die view. However, you can uncheck this option to hide the pin numbers.
Show Bank Name	Select this option to display the bank names and unselect to hide the bank names.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Show Net Name	Select this option to display the net names and unselect to hide the net names.
Show FPGA Port Name	Select this option to display the FPGA port names and unselect to hide the FPGA port names.
Show Clock Region Wise	Select this option to view the pins in clock region wise. After selecting, the pins are displayed in different colors other than bank colors.
Show Bank Wise	Select this option to view the pins in bank wise. After selecting, the pins are displayed in bank colors.
Select Bank	This option is available when pin number or pin name is selected earlier and you right-click on the pin. Select this option to select all the pins in a bank to which the selected pin belongs to.
Select Group Nets	This option is available when pin number or pin name is selected earlier and you right-click on the pin. Select this option to select the group nets.

When you select a pin in the *Die View*:

- The properties of the selected pin is displayed in the *Properties* and *Design Connectivity*.
- The selected pin and bank to which the pin belongs to is highlighted in red color in the canvas.

To invoke the Die view, perform the following step:

- Click *Die View* in the toolbar or choose *Windows – Die View* or press **Ctrl + 7**.

To select pins in the Die view, perform the following step:

- Press and hold **Ctrl** and select the pins using left mouse button.

Note: Pin selection is enabled only when the pin numbers and pin names are displayed.

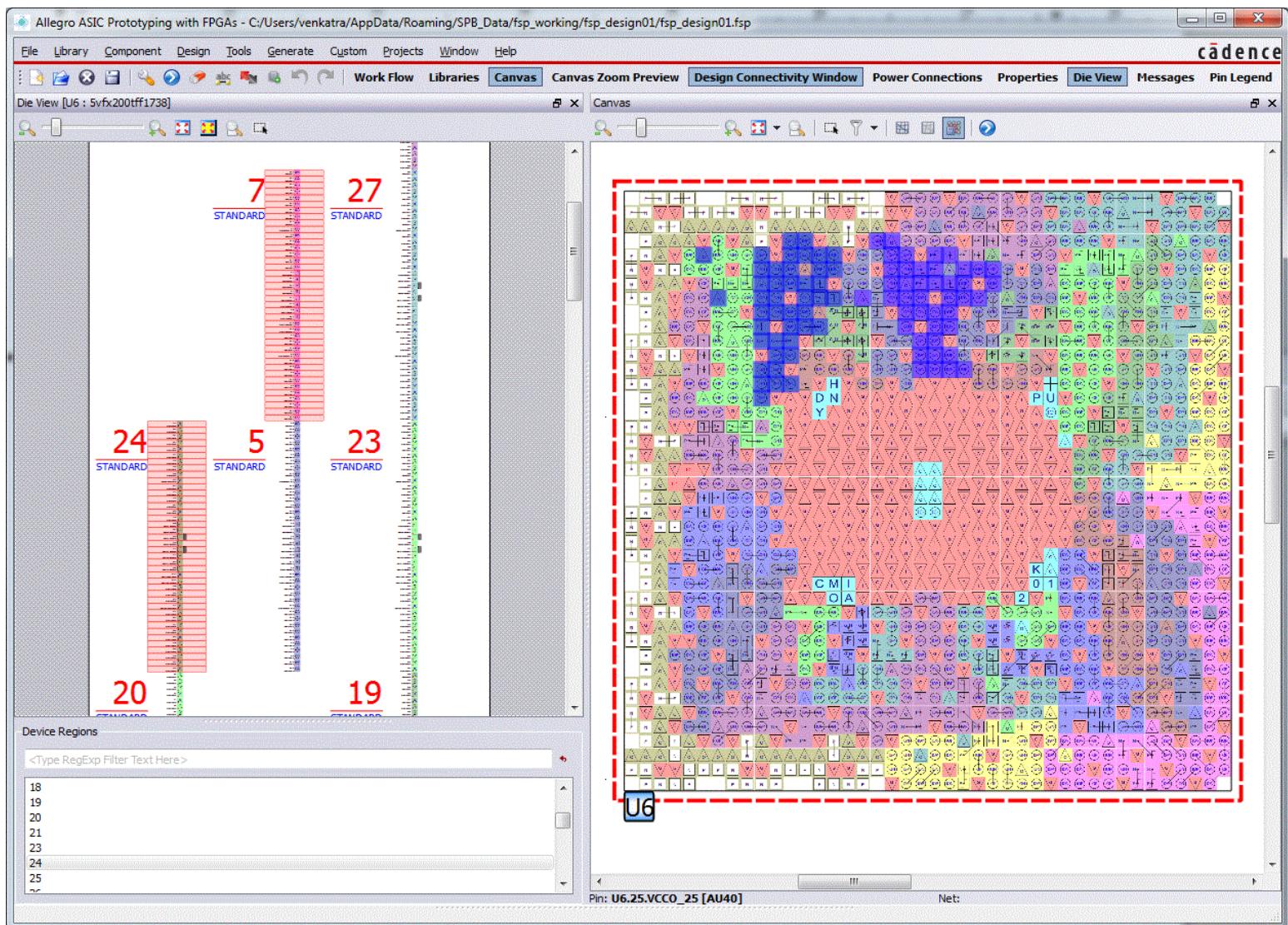
Searching Device Regions

The Die view provides a powerful filter and search pane, *Device Region*, with which you can filter or search for a specific bank or region name. The *Device Region* pane also provides a list of bank numbers and region names of the device that is placed on the Canvas. You can click on the region names to navigate to the pins that are present in the region you have selected.

Allegro FPGA System Planner User Guide

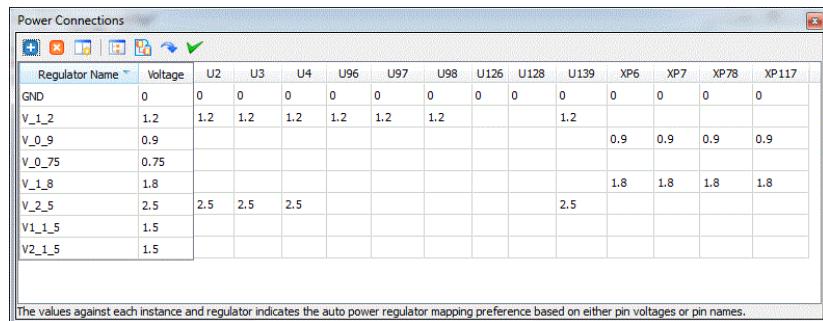
Getting Started with FPGA System Planner

The following figure displays an example of cross-probing between the Die view and Canvas. When you click on a bank or a region name in the *Device Regions* pane, the corresponding logic areas, such as x4, x8 data groups, regions, VREF groups, HROWS are highlighted in both Die view and Canvas.



In the Filter tab, you can either enter the complete string or specific alpha-numeric characters that includes in the part you seek. The Regular Expressions search feature is also supported in the Filter tab to make the search more easier. You can use the question marks (?), asterisks (*), underscore (_), and other metacharacters in the search string.

The Power Connections Window



The screenshot shows the 'Power Connections' window with a grid of data. The columns represent regulator names and their values, followed by multiple columns for different FPGA instances (U2, U3, U4, U96, U97, U98, U126, U128, U139, XP6, XP7, XP78, XP117). The rows list various regulators like GND, V_1_2, V_0_9, etc. Most cells in the first two columns are fixed values, while the remaining columns contain check boxes.

Regulator Name	Voltage	U2	U3	U4	U96	U97	U98	U126	U128	U139	XP6	XP7	XP78	XP117
GND	0	0	0	0	0	0	0	0	0	0	0	0	0	0
V_1_2	1.2	1.2	1.2	1.2	1.2	1.2				1.2				
V_0_9	0.9										0.9	0.9	0.9	0.9
V_0_75	0.75													
V_1_8	1.8										1.8	1.8	1.8	1.8
V_2_5	2.5	2.5	2.5	2.5					2.5					
V1_1_5	1.5													
V2_1_5	1.5													

The values against each instance and regulator indicates the auto power regulator mapping preference based on either pin voltages or pin names.

FSP includes a *Power Connections* window that allows you to define, add, delete, or reset the power connections in the current project.

The first column lists the regulator names and second column lists the regulator values. These two columns are freezed columns, that is, you cannot move the columns around the window. The cells under these two columns are editable. You can manually edit the regulator names and values using mouse clicks. The remaining columns are displayed based on the number of instances present in the canvas. These columns contain check boxes. These check boxes are displayed in enabled or disabled mode. The enable check box appears if the instance contains a IO standard of the selected voltage connected and disable check box if the instance does not contain any IO standard power by the selected regulator value.

Generally, power connections is performed after synthesizing the design. When you invoke the power connections window, you see a blank window. This indicates that no power regulators have been defined and mapped. You can both manually and automatically define the power regulators in the *Power Connections* window.

The power connections window includes the following options:

- Auto Add Regulator
- Map Missing Regulator
- Reset and Map Regulator

To invoke the power connections window, perform the following step:

- Choose *Windows – Power Connections*.

The Power connections window is displayed.

Menu Bar

FPGA System Planner user interface includes the following menu bars that provide access to most of the used commands in the FSP.

- [File menu](#)
- [Library Menu](#)
- [Component Menu](#)
- [Design Menu](#)
- [Tools menu](#)
- [Generate menu](#)
- [Custom menu](#)
- [Projects menu](#)
- [Window menu](#)
- [Help menu](#)

File menu

The options in the *File* menu are described below.

Name	Description
New	Create a new project.
Open	Opens an existing project saved on your computer or network.
Close	Close the active project.
Design Compare	Invokes the Design Comparison dialog box. For more information, see the <u>Design Comparison</u> section.
Project Settings	Opens the Settings dialog box. You use this dialog box to apply the project settings. For example project settings, Allegro DE-HDL, OrCAD integration settings, data-generation properties and net name settings.
Preferences	Opens the Preference dialog box. You use this dialog box to set the preferences for the PC board canvas view and instance display.
Customize Toolbar	Invokes the <i>Customize Toolbar</i> window. Use this window to hide and unhide the toolbar icons in the workspace.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Open Project Directory	Open the current project directory.
Save	Saves the current project.
Save As	Saves the current project with a new name in specified directory. Note: FSP does not create any directory while saving the design. Also the newly created project file .fsp reuse the existing cds.lib and cpm file.
Update Design from Board	Invokes Update Design from Board dialog box. Lets you synchronize the FSP design with the changes available in the Allegro board.
Create Project Copy for Allegro..	Lets you to create a copy of the current design in any directory. Click this option and specify a name and directory where you want to save the design in the Specify Design Copy Path dialog box. Note: An option dialog box is displayed prompting you about design modification state, if the design is unsaved mode and you click the <i>Create Design Copy</i> . Click <i>Save</i> to save the design with the changes made or click <i>Discard</i> to ignore the changes.
Print	Prints the current view of the canvas.
Print Preview	Opens the Print Preview dialog box. You can use this dialog box to see the current view of the canvas before printing.
Export to PDF	Click this to export the current canvas and abstract view to PDF. Note: This option does not work for Files and Schematics view. A warning message is displayed when you click this option.
Change Product	Opens the Cadence Product Choices dialog box. You can use this dialog box to select different product. You must close the project before you use this option.
Archive Design	Lets you save the project file and other associated project files in a specified directory and create a zip archive of this directory.
Recent Projects	Displays the list of projects which was recently opened by you. Select the project names as per required.
Quit	Exits from FSP.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Library Menu

The options in the *Library* menu are described below.

Name	Description
Create Part	Displays two sub menus as: <ul style="list-style-type: none">■ Interface : Invokes the Rules Editor dialog box to create interface rules file.■ Connector: Invokes the Rules Editor dialog box to create connector.■ Tester Connector: Invokes the Rules Editor dialog box to create tester connector1.
Edit Part	Displays two sub menus: <ul style="list-style-type: none">■ Linked to DE-HDL Symbol In DE-HDL environment, invokes Component Browser and in OrCAD environment invokes Edit Rules dialog box.■ Rules File Invokes Select Rules File dialog box.
Edit Rules File Path	Invokes the Edit Rules File Path dialog box.
Edit PSM Path	Invokes the User Preferences Editor dialog box.

Component Menu

The options in the *Component* menu are described below.

Name	Description
Add Part	In DE-HDL schematic environment, invokes Component Browser and in OrCAD schematic environment invokes Create and Select Rules/ Mapping File wizard.
Add FPGA	Invokes the Rules File dialog box.
Add Interface Rules	Invokes the <i>Select Rules File</i> dialog box.
Target Groups	Invokes the <i>Select Instance</i> dialog box.
Bank Settings	Invokes the <i>Select Instance</i> dialog box.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Add Protocol	Invokes the <i>Select Instance</i> dialog box.
Add Virtual Interface	Invokes the <i>Select Instance</i> dialog box.
Map FPGA Port or Pin Assignment	Invokes the <i>Select Instance</i> dialog box.
Import Constraint/ Pin Assignment	Invokes the <i>Select Instance</i> dialog box.
Export Constraint/ Pin Assignment	Invokes the <i>Select Instance</i> dialog box.
Map FPGA Resources	Invokes the <i>Select Instance</i> dialog box.

Design Menu

The options in the *Design* menu are described below.

Name	Description
Run Design	Opens the Run Design dialog box. Use this dialog box to make the connections.
Lock Nets	Opens the Lock Nets dialog box. Use this dialog box to lock, unlock the interface and device protocols.
Check Design Consistency	Checks the following points and displays a design consistency summary report in Log window. <ul style="list-style-type: none">■ Part definition for the instances on canvas■ Existing schematic symbol for the instances.
Report Design	Generates report design in: <ul style="list-style-type: none">■ Log window■ Text file
Change Net Names	Opens the Change Net Names dialog box. Use this dialog box to modify the signal names or apply the net name templates.
Update FPGA Port Names	Updates the FPGA Port Names with existing net names in Pin Property of Interface Instance window.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Check Power Connections	After you finish power mapping, use this option to check the power mapping.
Undo	Use this option to step back and recover a recent changes(s).
Redo	Use this option to reverse an undo operation, you can use the <i>Redo</i> command immediately after using Undo.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Create NetGroups

Displays the following sub menus when you place the mouse pointer.

■ Group Wise

Defines NetGroups for interface protocol in `<interface_instance_name>_<group_name_interface>` order and for device protocol in `<device_instance_name1>_<device_instance_name2>_<group_name>` order.

For example, for an interface U1 as instance name with group1, group2 and group3 as group names, the NetGroups name order would be as following:

- `<U1>_<group1>` for all the pins of group1
- `<U1>_<group2>` for all the pins of group2
- `<U1>_<group3>` for all the pins of group3

■ Interface /Protocol Wise

Defines NetGroups using interface instance name for interface component and device protocol name for device protocol.

For example,

- NetGroups for interface components are defined as `<instance_name>_0, <instance_name>_<1>...` so on.
- NetGroups for device protocols are defined as `<device_protocol_name>_<0>, <device_protocol_name>_<1>...so on.`

Note: “`<>_0, <>_1..`” are applied based on the maximum NetGroups size specified in the NetGroups tab of the Settings dialog box. For example, assume that you have 32-bit size of protocol and you specify 16 as maximum NetGroup size, then the *Interface/Protocol wise* option defines the NetGroups as:

- `<instance_name>_<0>` for first 16-bit.
- `<instance_name>_<1>` for second16-bit.

■ Swappable Pins

Assigns the same NetGroups on the pins which are swappable.

Example,

Following is one of the case:

CQ_P - U1_readgroup0

CQ_N - U1_readgroup0

Q_<0> - U1_readgroup1

Q_<1> - U1_readgroup1

Q_<2> - U1_readgroup1

From example above, since CQ_N and CQ_P are swappable pins same NetGroups are assigned.

Q_<0>..<2> are swappable pins and same NetGroups are assigned to them.

Clear Cached Net Names..	Removes the net names from Cached Net Names column in Pin Property of Interface Instance window.
Delete All Nets	Opens the Remove Net Confirmation dialog box. You can use this dialog box to remove all nets or remove the power nets.
Abstract View	Invokes the <i>Abstract View</i> form.

Tools menu

The options in the *Tools* menu are described below.

Name	Description
Terminations and Power Filters	Opens the Define Termination dialog box. Use this dialog box define termination in your design.
Deep and Wide Groups	Opens the Define Deep and Wide Groups dialog box. You can use this dialog box to create Deep and Wide groups.
JTAG Chain	Opens the Define JTAG Chain dialog box. Use this dialog box define JTAG chain.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

PROM Configuration	Opens the Define PROM Chain dialog box. Use this dialog box to define PROM chain.
Power Connections	Opens the Power Mapping Editor. You can use this dialog box to map the power regulators with power pins in the design. You can manually map the regulators or use the Map Regulators options to automatically power map the regulators.
Decoupling Capacitors	Invokes the Define Decoupling Capacitor dialog box. This dialog box helps you define decoupling capacitors for power signals.
Create Target Sets	Invokes the Create Target Sets dialog box.

Generate menu

The options in the *Generate* menu are described below.

Name	Description
Constraints Files	Generates constraints files such as .tcl, .ucf, and .verilog.
PlanAhead Scripts	Generates planahead scripts for xilinx devices.
Verilog Board Description files	Generates verilog design files of the current design.
Design Net List.	Generate the complete design net list file.
Schematic Generation Wizard	Opens the Schematic Generation Wizard. Use this wizard for the following: <ul style="list-style-type: none">■ Setup symbol data form for DE-HDL and OrCAD symbols.■ Generating DE-HDL or OrCAD symbols and schematics.

In DE-HDL Environment

Set up Symbol Data	Opens the Setup DE-HDL Symbols to Use/Generate dialog box. You can use this dialog box to select the existing symbols, foot prints and customize the symbol.
Symbols	Opens the Generate Allegro DE-HDL Symbols dialog box. You can use this dialog box to generate or regenerate the DE-HDL symbols.
Schematics	Opens the Generate Allegro Schematics dialog box. You can use this dialog box to generate the DE-HDL schematics.

In OrCAD Environment

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Set up OrCAD Symbol Data	Opens the Setup OrCAD Symbols to Use/Generate dialog box. You can use this dialog box to select the existing symbols, foot prints and customize the symbol.
Symbols	Opens the Generate OrCAD Symbols dialog box. You can use this dialog box to generate or regenerate the OrCAD symbols.
Schematics	Opens the Generate OrCAD Schematics dialog box. You can use this dialog box to generate the OrCAD schematics.
SCM Design	Opens the Generate SCM Design dialog box. You can use this dialog box to generate the SCM design. You can then open the exported design using SCM and continue the design process. The following files are needed while opening the design using SCM: <ul style="list-style-type: none">■ <project_file_name>.cpm■ cds.lib■ fsp_fe_lib directory■ worklib
Edit Design Block Symbol	Lists the following options: <ul style="list-style-type: none">■ Full<ul style="list-style-type: none">Click to open the Design Block Full Symbol Editor dialog box.■ Splits<ul style="list-style-type: none">Click to open the Design Block Split Symbol Editor dialog box.
Open	Lets you invoke and access the other Cadence tools. Lists the following tools names: <ul style="list-style-type: none">■ Project Manager■ DE-HDL■ SCM■ Capture■ Allegro PCB Editor

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Custom menu

The options in the *Custom* menu are described below.

Name	Description
Generate CSV Files	Generates .csv files.
Archive Design	<p>Use this option to save all the files related to your current project in the directory for archival and zip the directory into a single zip file.</p> <p>Archiving design process is useful when if you need to send a design to another person residing in a remote location, you will have to send the design along with the libraries. Archiver identifies every library part used in the project and then creates a local library containing only those parts. Any archived project can be used independent of reference libraries and other reference data.</p>

Projects menu

The options in the *Project* menu are described below.

Name	Description
Open Projects	Displays the list of all open projects. You may choose any one of the projects to view.
Snapshots	<p>Displays the list of snapshots automatically captured by FSP after every specific interval (specified in the Preference window).</p> <p>If you want to open the design which is saved at a particular interval then select the snapshot<time> as required.</p>

Window menu

The options in the *Window* menu are described below.

Name	Description
Work Flow	Invokes and displays the <i>Work Flow</i> window at the left side of the FSP canvas by default.
Libraries	Opens the Libraries pane. You can use this pane to view the FSP Library lists, schematics, and Project files.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Canvas	Displays the Canvas view.
Canvas Zoom Preview	Displays the Canvas Zoom Preview.
Design Connectivity	Displays the Design Connectivity window.
Power Connections	Displays the Power Connections window.
Properties	Displays the Properties at right side of the Canvas.
Die View	Displays the Die view window.
Messages	Opens the Messages window at the bottom of the tool.
Pin Legend	Opens the Pin Legend pane. Displays the pin legends available in FSP.

Help menu

The options in the *Help* menu are described below.

Name	Description
Examples	Displays the Cadence example designs. Select any one of the option to open the design.
Help(HTML)	Displays the topics available in FSP help.
User Guide (PDF)	Opens the PDF version of the FPGA System Planner user guide (PDF).
What's New	Opens the PDF version of the What's New guide (PDF).
Known Problems and Solution	Opens the PDF version of the KPNS document (PDF).
Web Resources	Use this option to access the following web pages: <ul style="list-style-type: none">■ cdnsUsers.org■ Online Support■ Web Collaboration■ Education Services
About	Displays the release and build version of FSP.

Toolbars

FPGA System Planner user interface includes the following toolbars that provide shortcuts to most of the commonly used commands in FSP.

Icon	Name	Description
	New	Creates a new project.
	Open	Opens an existing project.
	Close	Closes the current project.
	Settings	Opens the Settings dialog box. You can use this dialog box to apply the project settings.
	Save	Saves the changes made in the project.
	Run Design	Makes the connection between the components on canvas.
	Report Design	Generates the detailed project report in Messages window.
	Change Signal	Opens the Change Signal dialog box. You can use this dialog box to modify the signal names or apply the net name template to signals.
	Define Power Regulators	Opens the Define Power Regulator dialog box. You can use this dialog box to add power regulators in the design.
	Define Power Mapping	Opens the Define Power Mapping dialog box. You can use this dialog box to map the power regulators with power pins.
	Clear All Nets	Opens the Remove Net Confirmation dialog box. You can use this dialog box to remove the power connections or complete design nets.

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

	FSP User Guide	Displays the topics available in FSP html help.
	Canvas view	Displays the Canvas view.
	Zoom Options	Use this options to zoom in / zoom out the workspace.
	Preview	Opens the Preview pane.
	Add Part	In DE-HDL schematic environment, invokes Component Browser and in OrCAD schematic environment invokes Create and Select Rules/ Mapping File wizard.
	Design Compare	Opens the Design Comparison dialog box.

Customizing Toolbars

You can customize the toolbars in capture to alter the look and feel of the toolbar buttons or as per your comfort to work with the tool. You can also add and remove buttons from existing toolbars.

This section contains the following topics:

- [Docking Toolbars](#)
- [Add Buttons to Toolbars](#)
- [Remove Buttons from Toolbars](#)
- [Hide Buttons from Toolbars](#)

Docking Toolbars

The toolbars in FSP can be docked or made floating. This gives the flexibility of placing the toolbar anywhere on screen. You can even place a floating toolbar outside the tool perimeter.

To make a toolbar floating, position the mouse pointer on the area which is shown in the below figure.



The mouse pointer changes to . Click and keep the left mouse button pressed, and drag the toolbar.

To dock a toolbar, position the mouse pointer on the area as shown in the below figure.



The mouse pointer changes to . Click and keep the left mouse button pressed, move the toolbar and release the left mouse button.

Add Buttons to Toolbars

To create a new toolbar, perform the following steps:

1. Choose *File – Customize Toolbar*.

The Customize Toolbar dialog box is displayed.

2. Click on a toolbar button in the right side pane that you want to display in toolbar area.
3. Click to move the selected toolbar button to the left side pane.
4. Do the following if you want to insert a separator between the toolbar buttons:
 - a. Click <SEPARATOR> line in the right side pane.
 - b. Click to move the <SEPARATOR> to the left side button.
5. Click *OK* to save the settings.

Note: Before clicking *OK* make sure that the *Toolbar* option at bottom left of the window is selected.

Remove Buttons from Toolbars

To remove button(s) from toolbars, perform the following steps:

1. Choose *File – Customize Toolbar*.

The Customize Toolbar dialog box is displayed.

2. Click on a toolbar button in the left side pane that you do not want to display in the toolbar area.
3. Click to move the selected toolbar button to the right side pane.
4. Do the following if you do not want to insert a separator between the toolbar buttons:

- a. Click <SEPARATOR> line in the right side pane.
 - b. Click  to move the <SEPARATOR> to the left side button.
5. Click *OK* to save the settings.

Note: Before clicking *OK* make sure that the *Toolbar* option at bottom left of the window is selected.

Hide Buttons from Toolbars

To hide button(s) from toolbars, select the *Do Not Show* option in the Customizing Toolbar window.

Restoring the Default Settings

To restore the default toolbar settings, click *Restore Defaults* in the Customizing Toolbar window.

Undoing and Redoing Changes

You can use the *Undo* command to step back and recover a recent change(s). To reverse an undo operation, you can use the *Redo* command immediately after using *Undo*.

To save a new version of the design, choose the *File – Save* command.

Note: Undo/Redo commands are also available after you save the designs or switch between the designs.

Undo and Redo functionality is available for:

- All the spreadsheet editors, such as *Rules Editor*, *Rules Instance Editor*, *Design Connectivity*, *Protocol Editor*, *External Virtual Interface*, *LMF Mapper*, *Instance Symbol Editor*.

Note: Undo is not available for net renaming, that is if you change the name of a net, it cannot be undone.

- Canvas

Note: You can undo/redo the changes (except design changes) that are made through TCL commands. Consider you rotate an instance by 90 degrees by using the tcl command “RotateInstance”. You can use undo command to return the previous state.

You can use Undo/Redo in all the spreadsheet editors for:

- Creating and deleting pins and bus signals
- Manipulating pins and bus signals, for example
 - naming
 - copying and pasting names
 - moving selected pins or bus
 - moving all pins or bus
 - removing selected pins or bus
 - merging bus of one group with other group's bus
- Creating and deleting group
- Manipulating group information, for example
 - naming group
 - specifying group constraint
 - specifying group color
 - merging one group with other group



Important
You cannot use Undo/Redo once you update the pin definitions after updating pin definitions using the options in the *Create Form* menu in any spreadsheet editor.

Note: All the changes are applied to the design when you close the spreadsheet editor. Therefore, you cannot undo if you reopen the spreadsheet editor.

You can use Undo/Redo on the Canvas for:

- Adding and removing instance
- Manipulating instance details, for example adding and removing instance names and information
- Manipulating instance, for example
 - rotating
 - flipping

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

- moving
- copying and pasting

You can use Undo/Redo in the Instance Property Editor for:

- Manipulating device instance, for example
 - naming
 - rotating
 - flipping (side)
 - X and Y location values
 - instance information
- Manipulating interface instance, for example
 - naming
 - rotating
 - flipping(side)
 - X and Y location values
 - Instance information

You can use Undo/Redo in the Design Connectivity window for:

Column Name	Use Redo and Undo for
Net Name	<ul style="list-style-type: none">■ lock and unlock nets■ manipulation in the <i>Net Name</i> column, for example<ul style="list-style-type: none"><input type="checkbox"/> adding and removing net names<input type="checkbox"/> clear selected cell contents<input type="checkbox"/> clear text for current – Group, interface, column

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

Custom Attribute	<ul style="list-style-type: none"> ■ Adding and removing attribute details ■ Apply to all rows ■ Clear selected cell contents ■ Clear text for current – Group, interface, column
RTL Port Name	<ul style="list-style-type: none"> ■ Adding and removing attribute details ■ Apply to all rows ■ Clear selected cell contents ■ Clear text for current – Group, interface, column
Net Group	<ul style="list-style-type: none"> ■ Rename NetGroup <Group_Name> ■ Create and Assign New NetGroup ■ Clear selected cell contents ■ Clear text for current – Group, interface, column
Connection Type	<ul style="list-style-type: none"> ■ Specifying and removing the cell values
Pin Termination	<ul style="list-style-type: none"> ■ Clear selected cell contents
FPGA Ext Termination	<ul style="list-style-type: none"> ■ Clear text for current – Group, interface, column
Power Filter	
Assigned to Pin	
Regulator Name	<ul style="list-style-type: none"> ■ Specifying and removing the cell values
Voltage Value	<ul style="list-style-type: none"> ■ Apply to All Rows ■ Clear Selected Cell Contents ■ Clear Text For Current ■ Map Missing Regulators and Reset Map Regulators <p>Note: You can also undo/redo these two changes <i>Map Missing Regulators</i> and <i>Reset Map Regulators</i> if applied from the Power Connections dialog box.</p>

Note: You can perform the undo and redo changes mentioned above on the Canvas as well.

Undo Changes

To undo the most recent changes, perform any one of the following:

- Press Ctrl + Z, choose *Design – Undo*, or click the *Undo* icon in the toolbar.

Redo Changes

To redo the most action that you undid, perform any one of the following:

- Press Ctrl + Y, choose *Design – Redo*, or click the *Redo* icon in the toolbar.

Note: By default, the Undo and Redo icons are not available in the toolbar. You can use the *Customize Toolbars* window to make the Undo and Redo icons display in the toolbar.

Multiple Undo/Redo Changes

As you perform operations in the design, you can use the *Undo* command (or Press Ctrl + Z) repeatedly to undo every change you made since the design was last opened. You can undo the changes in the sequence in which they occurred.

To undo several changes, do the following:

- Press Ctrl + Z repeatedly until the change that you want to undo.

To redo several changes, do the following:

- Press Ctrl + Y repeatedly until the change that you want to redo.

Points to Remember

The section lists a few important you should remember when using the Undo/Redo commands.

- You cannot undo and redo any design related changes. For example,

- Deleting and adding nets.
 - FPGA swapping

Consider the following changes

- a. Place xc7vx550tffg1927 FPGA.
- b. Swap xc7vx550tffg1927 with the xc7v2000tflg1925 using the *Change <Instance_Name>* dialog box.

You cannot use undo to get the previous xc7vx550tffg1927 FPGA. You must manually delete the xc7v2000tflg1925 and place the xc7vx550tffg1927 FPGA on the Canvas.

- Interface groups and device bank settings
For example, Connect to Device, Use Banks, Don't Use Banks.
 - Import and export constraints
 - Map FPGA Ports and Pins and Map resources
 - Run design
 - Create JTAG chain
 - Create PROM chain
 - Create Deep and Wide groups
 - Define and Edit Termination
 - Define Decap
 - Virtual interface creation and deletion
 - Device protocol creation and deletion
- You cannot undo/redo any changes that are applied to the design from the following dialog boxes.
- Project Settings
 - Preference Settings
 - Customize Toolbar
 - Run Design
 - Change Net Names
 - Update FPGA Port Names
 - Clear Cached Nets
 - Edit Rules
 - Link FPGA to Schematic Symbols
 - Convert VI to Real Part
 - Edit VI and Protocol

- Re-assign Net
- Swap Groups

Configuring FPGA System Planner

FSP User Interface

FSP provides different levels of configuration. These different levels determine the scope of each configuration settings.

- Create default settings for new designs with the *Settings* dialog box. These settings stay with the design even if it is moved to another system with different preferences and settings.
- Customize the working environment specific to your system with the *Preferences* dialog box.

The settings in the *Preferences* dialog box determine how FSP works on your system. These settings must persist from one design to the next. The preference settings are system dependent. This means that you can set the colors, grid display, enable instance information and more, based on your desire. These preference on your system will not change, even if you work on a design which is created on another system with different preference settings.

Once you begin working on a new design, you can customize the preference settings even in between the design.

Config.ini File

When FSP invokes, it uses a pre-defined set of default properties values for the application settings. These default properties are defined in the configuration (config.ini) file.

Location of Config.ini file

By default, the capture.ini file is copied at the location <release>/share/cdssetup/fsp during installation. Different projects may require different settings and it might not always be possible to modify the configuration files at the installation level. Also, if a config.ini file pre-exists in the installation location, the properties in the existing config.ini file may be overridden by the new config.ini file properties. You also have the option to move config.ini to another

location. To ensure that FSP uses the config.ini file from a different location, you need to set the location as a value for the environment variable CDSSITE.

Config.ini Variable

The config.ini file contains a large set of properties used by FSP. As this is a text-based file, you have the option to add, modify, or delete the properties and sections in this file. However, any changes you make to the config.ini file located in the installation level can cause unexpected behaviour in FSP. So you are advised to only make changes in the site-level. After modifying the properties, you are advised to restart FSP to see the changes.

The following table lists the names and description of the properties available in the config.ini file:

Property Name	Description
bus_notation	<p>The bus_notation property determines the individual bits for vector signals. The signals will always be saved within the specified braces. For example, if you specify bus_notation = “ [] ”, the individual bits would be represented as Data_[1], Data_[2], ... so on.</p> <p>By default, the angular braces “ <> ” is set as bus notation. You can manually specify other braces as bus notation using any text editor. The valid braces are <>, {}, (), and [].</p> <p>Note: Each time you make a change in the bus_notation property, you need to save, close, and reopen the tool.</p>

Allegro FPGA System Planner User Guide

Getting Started with FPGA System Planner

working_dir	<p>The <code>working_dir</code> property determines working directory of FSP. By default, this property is not set. When FSP starts up, FSP reads the value of this property. If the value is specified or found, it is set as default working directory and will be displayed in the <i>What do you want to do?</i> and <i>Create New Project</i> windows each time you create new project or invoke FSP.</p> <p>If the value is not found, you can manually specify any location where you want to save project files as a value to this variable.</p> <p>Note: Each time you make a change in the <code>working_dir</code> property, you need to save, close, and reopen the tool.</p> <p>Note: You must use relative path to specify directory path as a value. For example, <code>working_dir = "\$CDSROOT/projects"</code>.</p> <p>If you do not specify the <code>working_dir</code> property's value, FSP looks for <code>FSP_WORKING_DIR</code> environment variable. If this environment variable is set, the value of this variable is considered as default working directory. The <code>FSP_WORKING_DIR</code> is an alternative to the <code>working_dir</code> property.</p> <p>If the <code>FSP_WORKING_DIR</code> environment variable is not set, FSP reads the <code>HOME</code> environment variable to locate the working directory.</p> <p>Note: You must use relative path while specifying <code>FSP_WORKING_DIR</code> and <code>HOME</code> variables values.</p>
passive_com_prefixes	<p>This section determines the prefix for the components that you want to treat as passive components while importing the board file in FSP.</p> <p>By default, R, L, C, D are set as default prefixes. You can manually specify other prefixes within the double quotes.</p> <p>Note: Prefixes are case-sensitive.</p>

`lrfpath` Displays a list of paths where the interface rules files are copied during installation. Whatever you set here will be displayed in the *Libraries*. You can also manually add paths of your own rules file in this field. After you set your own rules file paths, each time you invoke FSP, the rules file added by you will be displayed in the *Libraries*.

Note: You can use both relative and absolute paths to specify the library paths.

Schematic Settings

`capture_ini_file` Displays the path of the directory where the capture settings file exists. This file is useful and internally used by FSP during capture schematic generation.

`max_symbol_pins_left_right` This section determines that the maximum number of pins allowed for each symbol on left and right, and top and bottom side.

`max_symbol_pins_top_bottom` For example, if you define as depicted below.

```
max_symbol_pins_left_right = 100  
max_symbol_pins_top_bottom = 100
```

When you generate symbols, FSP will generate the symbol blocks with each block including 100 pins on left and right, and 100 pins on top and bottom.

Managing Environment Variables and Template Files

This chapter contains the following topics:

- [The FSP Template Files](#)
- [Customizing Template Files for a Site](#)
- [Customizing Config.ini file for a Local Directory](#)
- [Default Search Order of Template Files](#)
- [Merging Attributes of Config.ini file](#)
- [Modifying Template File Attributes](#)

The FSP Template Files

This section describes the templates files that are processed by FSP from installation directory. The template files are located at `.. /<release_name>/share/cdssetup/fsp`. FSP provides four different files and uses these template files to communicate with other design-capture tools, such as Design Entry HDL, PCB Editor, and Capture. FSP also uses these template files to support features, such as create project, generate schematic/symbols or perform name space validation.

The CDSROOT environment variable, stores the location of the Cadence Software Installation path. For example, if you install the Cadence software at `/main/<release_name>`, set the CDSROOT variable to `..<release_name>/main/spb166`. The FSP template files are located at `.. /<release_name>/share/cdssetup/fsp`.

The following table provides a brief description of the files types associated with FSP.

File	Description	Used For
config.ini	Contains the tool configuration settings information.	<p>The config.ini file contains the following settings:</p> <ul style="list-style-type: none">■ Rules files directory paths For example, <code>\$CDSROOT/tools/fsp/samples/lrf/bus_interface_connectors, \$CDSROOT/tools/fsp/samples/lrf/configuration_proms</code> and more.■ Bus notations Angle brackets “<code><></code>” are set as default bus notations.■ Different pin types for symbols For example, <pre>input_pin_dir="Left" output_pin_dir="Right"</pre>

custom_attribute.tx	Contains the default values for the custom attributes that can be customized for FSP entities.	Adding custom properties to instance during schematic generation.
---------------------	--	---

Customizing Template Files for a Site

Different projects may require different settings and it might not always be possible to modify the configuration files at the installation-level. It is recommended that you do not modify the template files at the installation-level. Copying the configuration files to new directory and creating an environment variable CDS_SITE that points to new location helps you from modifying the template files at the installation level.

CDS_SITE is a user-defined variable which lets you customize the FSP supplied environment by overriding the default site location CDSROOT. The CDS_SITE environment ensures that initially FSP look at the files stored at the location pointed to the CDS_SITE variable.

The FPGA System Planner, v16.6, ships with a new interface library database schema version and directory structure. The new library database improves the component naming convention as per the logical rules of the component and JEDEC type. The config.ini file located at `..</release_name>/share/cdssetup/fsp` directory contains the new lrf paths of the rules files. If you currently have designs and libraries developed in a previous version of FSP and uses the old config.ini file settings from CDS_SITE level, then you must update the new lrf paths settings in the old config.ini file located at CDS_SITE level. Updating the new lrf paths settings in the config.ini file helps you open the old design in FPGA System Planner, v16.6 without loosing the old design database.

To set up CDS_SITE environment variable perform the following tasks:

1. Create a directory structure as `D:/my_local/cdssetup/fsp`.
2. Copy the template files from installation directory to `D:/my_local/cdssetup/fsp` folder.
3. Right-click My Computer icon and choose *Properties*.

The System Properties window is displayed.

4. Click the *Advanced* Tab.
5. Click *Environment Variables*.
6. Click *New* In System Variables.

The New System Variable dialog box is displayed.

7. Specify the Variable Name as CDS_SITE.
8. Specify the Variable Value as D:/my_local/cdssetup/fsp.
9. Click *OK*.

Customizing Config.ini file for a Local Directory

Site customization through the environment variable CDS_SITE lets you customize the Cadence supplied FSP template files. In addition to the CDS_SITE variable, you can set a variable FSP_CONFIG_FILE for individual designs. This FSP_CONFIG_FILE environment variable is available for only config.ini file. FSP_CONFIG_FILE variable lets you to locate the config.ini file outside the standard default location such as CDSROOT and CDS_SITE. This helps you to personalize the configuration settings specific to particular design.

For example, for a particular design, you may want to use a bus notation that is different from the default notation specified in the config.ini file at CDSROOT and CDS_SITE level. This can be implemented by using a local config.ini file and the FSP_CONFIG_FILE environment variable that points to the local config.ini file. To use your required bus notations in your design copy the config.ini file outside the standard location and create FSP_CONFIG_FILE environment variable points to new location.

To setup FSP_CONFIG_FILE environment variable, perform the following steps:

1. Copy the config.ini file from ../<release_name>/share/cdssetup/fsp to your local directory.

For example D: /my_lib/sample.

2. Right-click My Computer icon and choose Properties.

The System Properties window is displayed.

3. Click the Advanced Tab.

4. Click *Environment Variables*.

5. Click *New* in System Variables.

The New System Variable dialog box is displayed.

6. Specify the Variable Name as FSP_CONFIG_FILE.

7. Specify the Variable Value as D:/my_lib/sample.

8. Click **OK**.

Default Search Order of Template Files

When FSP launches, it looks for the template files with following order:

- \$FSP_CONFIG_FILE (Individual setting, local to a machine or user)
- \$CDS_SITE (Site - level setting)
- \$CDSROOT (Default setting)

Note: If you have not defined the \$FSP_CONFIG_FILE variable, FSP starts searching for the template files from site-level.

FSP supports all the attributes depending on the site level settings CDSROOT and CDS_SITE. When attributes is specified in more than one template files located at different site-level, the \$CDS_SITE template file attributes have precedence over the \$CDSROOT template file attributes. This implies FSP reads the attributes only from \$CDS_SITE template files.

Note: For config.ini file, the attributes of config.ini file are read by FSP from \$FSP_CONFIG_FILE directory.

Rules File Search Mechanism

Rules file directory path are defined in config.ini file and <project_name>.fsp. By default the config.ini files are located in share/cdssetup/fsp directory.

The rules files directory path can be specified through Settings or Edit Rules File Path dialog box and saved in the project file <project_name>.fsp. To read the <project_name>.fsp and config.ini file for rules file directory paths from more than one location, FSP supports a search mechanism. The search mechanism reads for the rules file directory paths in the <project_name>.fsp and config.ini file in a prescribed search order so that you can use the rules files consistently.

Searching begins from project level, the <project_name>.fsp file given first preference over the config.ini file in site level and then config.ini file present in root level. For detailed information on the searching mechanism of config.ini file, see the [Merging Attributes of Config.ini file](#) section.

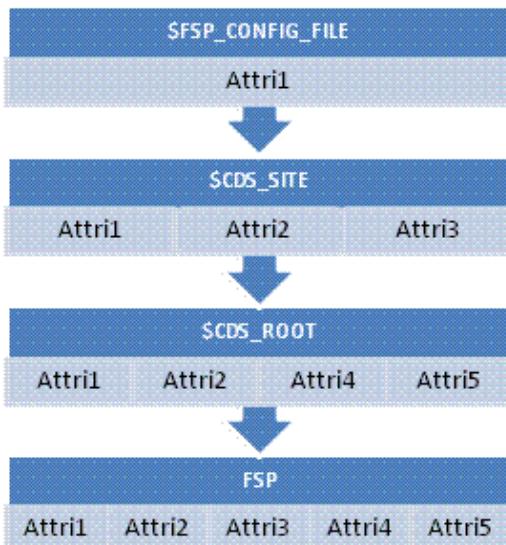


The rules file paths specified in the config.ini file are processed from any one of the search locations based on order in which it comes first. For example, FSP stops searching for the config.ini file if the rules file path are found in the config.ini file located in the \$CDSSITE site level.

Merging Attributes of Config.ini file

This section describes how attributes of the config.ini file located at different site-level are read and merged. When attribute is specified in more than one config file located at different site-level, then the attributes specified in the \$FSP_CONFIG_FILE environment variable is given first preference.

The following figure depicts how the attributes are read and merged from template files located at different site-level settings.



The attributes are read and merged as following:

- Attr1 is read from the FSP_CONFIG_FILE site, since FSP_CONFIG_FILE site has precedence over the other sites.
- Attr2 is read from the CDS_SITE site, since it is not available in the FSP_CONFIG_FILE site and CDS_SITE has the second preference over the remaining sites. The Attr2 in CDS_ROOT and FSP sites is ignored.

- Attr3 is read from the CDS_SITE site, since it is not available in the FSP_CONFIG_FILE site and CDS_SITE has the second preference over the remaining sites. The Attr3 in the FSP site is ignored.
- Attr4 and Attr5 are read from the CDS_ROOT directory, since these are not available in FSP_CONFIG_FILE and CDS_SITE sites, and CDS_ROOT has the precedence over the FSP directory.

The attributes of config.ini located at different site-levels are merged into single level. After FSP launching, the common and missing attributes of config.ini file are used altogether.

Modifying Template File Attributes

When you try to modify the template files it is recommended you modify the files that are located in CDS_SITE and FSP_CONFIG_FILE. For more information about why you must not modify the template files in the installation directory, see the [Customizing Template Files for a Site](#) section.

The following table depicts the changes you can do in each template files.

File	Changes
config.ini	The attributes can be changed as required. All the unknown attributes are ignored.
custom_attribute.txt	Custom attributes are user-defined. Adding new entities are not supported. You can either modify the entity value or can specify a value when none was specified.

Assigning Footprint Models

To map a FSP logical model to the physical board environment you need to specify the footprint information for all the design libraries that are to be included in the PCB Editor. You set an environment variable directly from a graphical user interface in FSP. The footprint files are useful at the time of placing the components on the FSP canvas using Component Browser. After selecting rules and symbol files, the Component Browser automatically determines a valid footprint from the path set in the environment variable for the selected rules file.

Locating Footprints in Install Directory

The footprint files gets installed at the time of Cadence Software installation. Cadence ships a library of footprints and padstacks for FSP interface and device libraries. By default footprints and padstacks libraries are copied at <install dir>/tools/fsp/samples/dra. The /dra has three types of files of different format, i.e., .dra, .psm and .pad. But this path is not set in the environment variable when you first time invoke the FPGA System Planner. Before starting your design or invoking Component Browser, you need to include this location of footprints and padstacks in psmpath and padpath variables. Setting the psmpath and padpath variables ensures that the footprints and padstacks are accessible during Component Browser. Both variable are also read by Allegro PCB Editor to display FSP components footprint on the Allegro canvas.

In case you wish to move custom footprints or padstacks to another directory, you can copy them to another directory or site level respectively. After copying to another directory make sure that the new directory paths are set in the environment variables.

To set the psmpath and padpath variables perform the following steps:

1. Choose *Library – Edit PSM Path*.

The User Preferences Editor dialog box is displayed.

2. Select *Paths - Library* from the Categories list.

3. Click *browse (...)* under *Value* column of *padpath* Preference.

The padpath Items dialog box is displayed.

4. Click *New(insert)* icon to add a new row to the dialog box.

5. Paste the directory path in this row where the pad files exists and click *OK*.

6. Click *browse (...)* under *Value* column of *psmpath* Preference.

The psmpath Items dialog box is displayed.

7. Click *New(insert)* icon to add a new row to the dialog box.

8. Paste the directory path in this row where the dra files exists and click *OK*.



FSP always looks for dra files instead of psm files using psmpath variable.

9. Click *OK* of the User Preferences Editor dialog box.

Allegro FPGA System Planner User Guide

Managing Environment Variables and Template Files

Once you have finalized the footprint location you are ready to place the components on the canvas using Component Browser.

Allegro FPGA System Planner User Guide

Managing Environment Variables and Template Files

Project Creation and Setup

This chapter describes the following sections:

- [Creating Design Projects](#)
- [Project Files](#)
- [Project Directory Structure \(DE-HDL Schematic Environment\)](#)
- [Setting up the Project](#)
- [Setting up the Search Path to the Rules File](#)
- [Setting up the Search Path to the Footprint File](#)
- [Setting up the Preference](#)
- [Opening a Project](#)
- [Saving a Project](#)
- [Closing a Project](#)
- [FSP Install Directory](#)

Creating Design Projects

You can create a new design project using the Create New Project window or from an existing cpm file.

See the following topics to create a new design project:

- [Creating a New Design Project](#)
- [Creating a Project from a cpm file](#)

Creating a New Design Project

When you create a new design project, you need to select the schematic environment in which you need to work, that is DE-HDL or OrCAD.

The procedure to create a project is the same (or similar) for both DE-HDL and OrCAD.

1. Choose *File – New* or press **Ctrl + N**.

The *Create New Project* dialog box is displayed.

2. Select DE-HDL or OrCAD as the schematic environment.
3. In the Design Name field, type the project name.

Note: Use lowercase letters, numbers, and the underscore (_) character in project names. Using mixed-case project names might cause problems when you move your design across platforms.

4. In the Select Project Directory field, specify the path to the directory in which you want to create the project. You can also click *Browse*, select a folder in the *Select Project Path* dialog box, and then click *Select Folder*.

Note: If you want to create the project in a directory that does not exist in the path, add the name of the new directory to the path(for example: \project1). The folder name is added to the path.

5. In the *Generate Symbols in* field, specify the library name.

Note: Symbols are generated in this folder. By default, `fsp_fe_lib` is displayed as the library name. You can also choose an existing library name from the drop-down list or specify a new name.

6. In the *Design Library* field, specify the working directory name.

This field represents the working directory name. All the design data, FSP project file, and view-related log files are stored in this directory. By default, <design_name>_lib name appears in this field. You can also choose an existing name from the drop-down list or type a new name.

7. In the *Configuration File* field, specify the path to the folder in which the capture.ini file exists followed by the capture.ini file.

Or

Click *browse(...)*, select a folder in the *Select Capture Configuration File Path* dialog box, and then click *OK*.

Note: If you do not specify the capture.ini file path, then the default config.ini file is read from the %CDSROOT%\share\cdssetup\fsp path.

Note: The *Configuration File* field is available when you select OrCAD as the schematic environment.

8. Click *OK*.

Creating a Project from a cpm file

You can create an FSP project from an existing CPM file. A new project file (.fsp) is created at the following location <project_name>/<project_name>_lib/<project_name>/fsp.

Note: This section is only available for the projects created in the DE-HDL environment.

You can create a project from an existing cpm file using one of the following methods:

- [From the Create New Project Window](#)
- [From the Run Window](#)

From the Create New Project Window

To create a new project from a cpm file:

1. Choose *File – Open* or press Ctrl + O.
The *Open Project* dialog box appears.
2. Browse to the project directory, select .cpm file, and click *Open*.

The *Open Existing Project* directory appears with the following fields values:

The *Design Name* field displays the design name.

The *Select Project CPM File* field displays the path to the folder followed by the cpm file name.

The *FSP Database Location* field displays the path to the folder followed by the *.fsp* file name.

3. In the *Generate Symbols In* field, you can either continue with the current library name or specify a new library name.

If you specify a new library name, a folder with the specified name is created in the current project directory.

The *Design Library* field displays the design library name.

4. Click *OK* to open the design on the Canvas.

From the Run Window

To create a project from a cpm file from the command prompt:

1. Choose *Start – Run*.

The *Run* window appears.

2. Type *fpgasysplanner<space><-proj><space><path and file name.cpm>*.
3. Click *OK* to open the design on the Canvas.

Project Files

When you create a project in FSP (using the DE-HDL Schematic environment), a folder with the same name as the project is created. The project folder contains the following files and directories:

- A project file (<project_name>.cpm)

This is a DE-HDL project file. The project file is used by FSP to invoke Component Browser and generate DE-HDL symbols and schematic. By default, the <project_name>.cpm file contains the following minimum settings required for a project:

- The name of the current design and design library.
- The list of DE-HDL symbol libraries.
- The location of the temporary directory where tool generates temporary files.

You can also point a <project_name>.cpm file of the master board schematics in FSP. While specifying the cpm file, the *cds.lib* file with all the library entries (which are listed in the cpm file) must be available in the same directory as the cpm file.

- A *cds.lib* file

The *cds.lib* file contains the list of DE-HDL symbol libraries which you can use in FSP. The file contains logical names of the DE-HDL symbol libraries and their physical locations. The *cds.lib* file also contains the local library name, *fsp_fe_lib* pointing to the local library of the current project and the design library name, <design_name>*_lib* used as design library.

- *fsp_fe_lib*

The *fsp_fe_lib* directory is a local directory which is used to store symbol files. You need to specify the local library name in the Settings dialog box to generate the symbols. See [Setting up the Project](#).

fsp_fe_lib is the default directory name. You can also specify any other name.

- A <project_name>*_lib* directory

When you generate a schematic, all the schematic files are generated in the <project_name> directory. This directory also stores the design you capture in FSP in the <project_name> subdirectory, which is referred to as a cell representing the entire design or a part of it.

The cell (<project_name>) contains a set of subdirectories called cell views. By default, *fsp* is the only cell view present in the cell. It stores the following files and folders:

Allegro FPGA System Planner User Guide

Project Creation and Setup

- ❑ <project_name>.fsp

The FSP project file.

The project file contains the setup information specified for the current project in the Settings dialog box. For example, lrf paths, net_name_template, schematic environment, schematic generation output directory, and the DE-HDL project file name and location.

- ❑ lrf

This is a rules file directory. During the design any new rules or mapping files you create or which are generated by default are saved in this project directory. By default, the lrf directory is set as the working directory. Rules file in the lrf directory is also displayed in the Libraries window.

- ❑ output

The output directory contains a subdirectory, constraints. In the constraints subdirectory, the constraint files are generated within the folder as <instance_name>. For example output/constraints/u1, u1 is the name of the instance on canvas for which you have generated the constraint files.

Note: This directory is created only when you generate constraints files.

- <project_name>.log

The log file contains all the messages, warnings, and errors displayed in the Messages window.

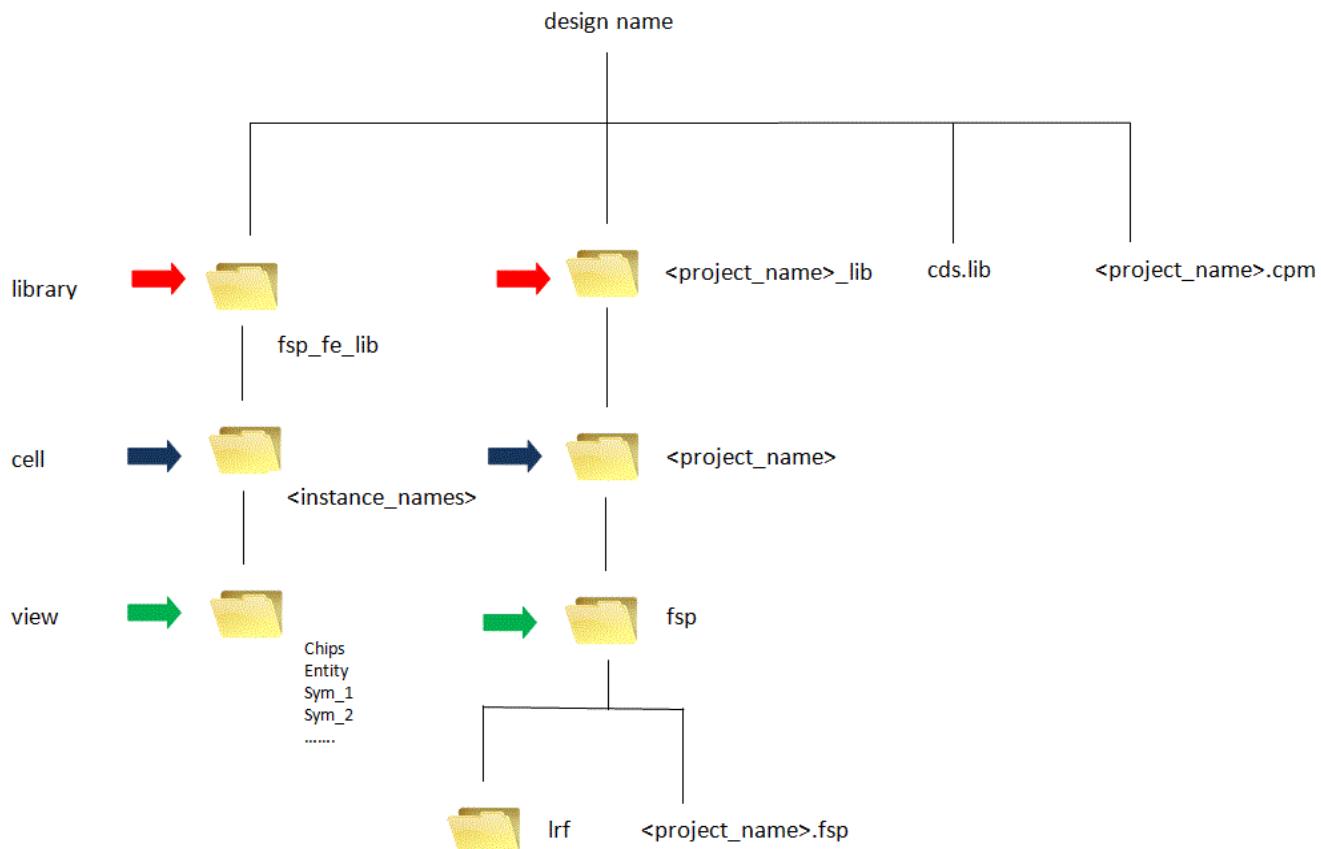
- A temp directory(temp)

Temporary files created by FSP are placed in the temp directory. You can delete the contents of the directory.

Allegro FPGA System Planner User Guide

Project Creation and Setup

Project Directory Structure (DE-HDL Schematic Environment)



When you create a project in FSP in the OrCAD schematic environment, a folder with same name as the project is created. The project folder contains the following files and directories:

Files/Folder Name Description

<project_name>.fsp	This is an FSP project file. For more information, see the <project_name>.fsp topic in the Project Files section.
--------------------	---

output	The output directory contains two subdirectories, constraints and OrCAD. In the constraints subdirectory, the constraint files are generated within the folder as <instance_name>. For example output/constraints/u1, u1 is the name of the instance on canvas for which you have generated the constraint files.
lrf	The OrCAD subdirectory, contains the OrCAD schematic files you generate from FSP. For more information, see the lrf topic in the Project Files section.

Setting up the Project

After clicking OK in Create New Project window, the Settings dialog box is automatically displayed. Although you can change them at any time during your project work.

To setup the settings for the project complete the following steps:

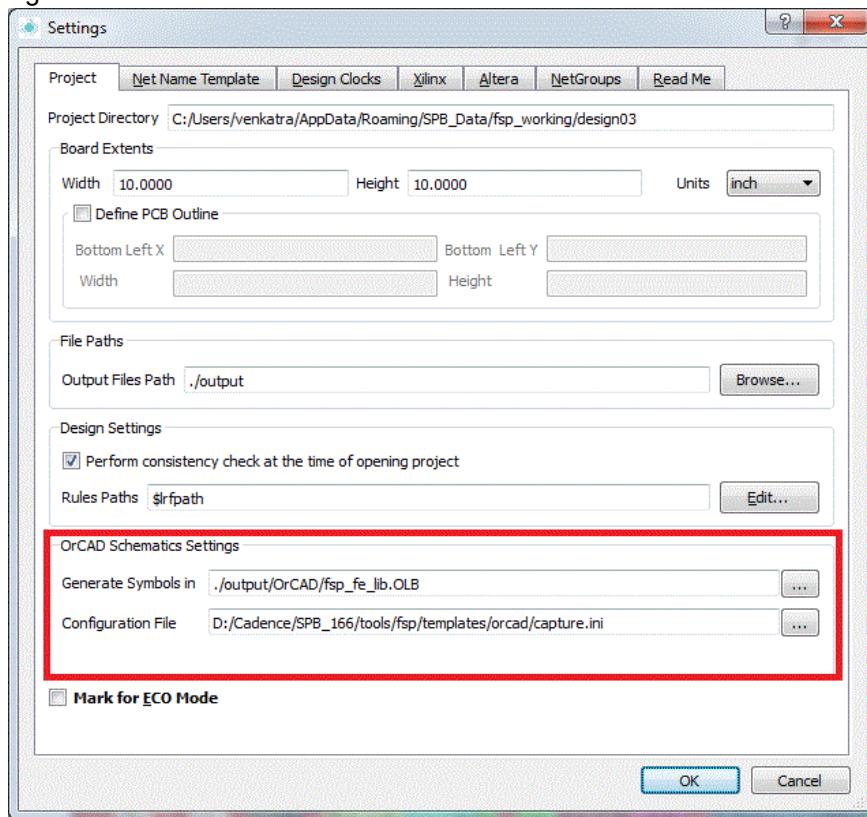
1. Choose *File – Settings*.

The Settings dialog box is displayed. The Settings dialog box has six tabs: Project, Net Name Template, Design Clocks, Xilinx, Altera, and NetGroups.

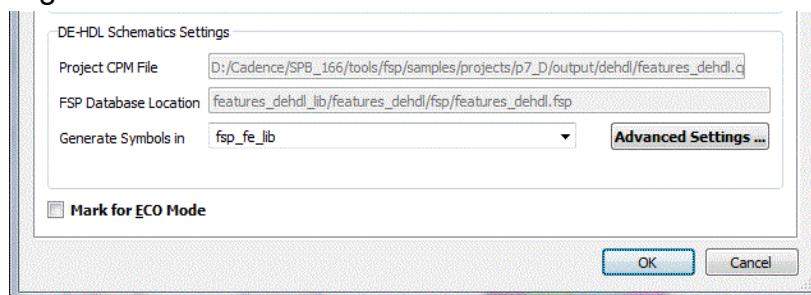
Allegro FPGA System Planner User Guide

Project Creation and Setup

OrCAD Settings



DE-HDL Settings



2. In each tab of the Settings dialog box, specify the settings options for the project.

For more information on options in each tab of the Settings dialog box, see [Settings](#) section.

3. Click **OK**, to save the settings and close the settings dialog box.

Setting up the Search Path to the Rules File

There are two ways to add search path to the rules file:

- [Updating Configuration File](#)
- [Using the Rules File Path Editor](#)

Updating Configuration File

The config.ini configuration file located at \$CDSROOT/cdssetup/fsp defines FSP's logical library names and physical storage locations for each installed library. The config.ini file contains the following variable for the installed library:

```
lrfpath = "./lrf;$CDSROOT/tools/fsp/samples/lrf/  
bus_interface_connectors;...."
```

FSP reads the config.ini file, locates the logical libraries, and displays the library names in Libraries. In Libraries, you can view and browse the libraries. You can edit the config.ini file and add library locations as a value of the lrfpath variable to include your own logical libraries. You can update the config.ini file using any text editor. You can add libraries in FSP by specifying their logical names and physical locations in the config.ini using any text editor. Do not modify the config.ini file located at \$CDSROOT. Instead, create a site config.ini file and then do the neccessary changes. For more information on how to create a site config.ini file, see the [Customizing Template Files for a Site](#) section.

Using the Rules File Path Editor

The *Rules File Path Editor* dialog box lets you add, modify, or delete the libraries of the project. For any changes you make in the *Rules File Path Editor*, the changes are reflected in Libraries. The logical library names and their locations are fetched from the config.ini and displayed in grey color in the *Rules File Path Editor*. These grey colored paths are read-only fields. FSP's rules and mapping files search mechanism reads the library paths in the dialog box in a search order prescribed by you. The order in which the libraries are listed in the *Rules File Path Editor* determines their search order. Libraries are searched starting at the top of the list. You can change the search order to determine which library need to be searched first.

Note: The project directory is the last location where FSP searches for the rules file if it is not found in the specified library location.

Note: While searching rules file, FSP also search for the associated mapping file (.lmf) in the lib:cell:fsp directory of the associated schematic symbol for DE-HDL designs and .OLB directory path for OrCAD designs. If not found mapping file is searched in the order in which the libraries are listed in the *Rules File Path Editor*.

Both the rules and mapping files are fetched only from the directories specified in the *Rules File Path Editor*. This ensures that the designs are portable across different systems and the search mechanism is aligned with the rest of the Cadence products.

To add a library or change the search order, perform the following steps:

1. Choose *Library – Edit Rules File Path*.

The Rules File Path Editor is displayed.

2. Click *Add* to add a new library folder in the *Rules File Path Editor*.
3. Browse to a folder where rules file is located and click *Select Folder* in the *Select Rules File Directory* dialog box.

The library name with path is displayed.

4. Select a row and click *Move Up* to move a library one level up.
5. Select a row and click *Move Down* to move a library one level down.

6. Click *OK* to save the settings.

Setting the Working Directory

A working directory is the directory to which the rules and mapping files generated by FSP are added during the design.

When you create a project, FSP creates a default `lrf` directory in the project directory and sets it as the working directory. You can set any existing library listed in the *Rules File Path Editor* dialog box as the working directory. A working directory is also included as a part of rules file search mechanism. You may move the working directory up or down to determine your search order. Working directory is displayed in bold black in the *Rules File Path Editor*.

To set the working directory:

1. Choose *Library – Edit Rules File Path*.

The Rules Files Path Editor is displayed.

2. Select a library row and click *Set Working Dir* to set as the working directory.
3. Click *OK* to save the settings.

Modifying Library Entries

You can modify or remove libraries from FSP using the *Rules File Path Editor*. You can use both relative and absolute paths while modifying the logical libraries locations.

The read-only library path entries in the *Rules File Path Editor* can be modified only by manually editing the config.ini file at the CDS_SITE level. Whenever lrfpath variable value is modified in the config.ini file, the library paths are automatically updated for the design. This allows you to control the design portability by managing the site-level or system specific paths.

The read-write library path entries in the *Rules File Path Editor* can be modified at any time during the design in the *Rules File Path Editor*.

To modify the library entries, perform the following steps:

1. Double-click on a non-read-only row to modify the path.
2. Select a non-read-only row and click *Delete* to delete a library from the design.
3. Click *OK* to save the settings.

The library is removed from Libraries.

Setting up the Search Path to the Footprint File

To map an FSP logical model to the physical board environment, you need to specify the footprint information for all the design libraries that are to be included in PCB Editor. You set an environment variable directly from a graphical user interface in FSP. The process of specifying and fetching the footprint information from the footprint file (dra) varies based on the following schematic environment.

- [DE-HDL](#)
- [OrCAD](#)

DE-HDL

The footprint files are useful at the time of placing the components on the FSP canvas using Component Browser. After selecting the rules and symbol files, the Component Browser automatically determines a valid footprint from the path set in the footprint variables for the selected rules file. The FSP library structure is closely aligned with the library:cell:view structure of DE-HDL, to ensure that the correct footprint file is picked when selecting a schematic symbol. The footprint files are selected based on the

Allegro FPGA System Planner User Guide

Project Creation and Setup

JEDEC_TYPE property. If the JEDEC_TYPE information is specified in the PTF file, corresponding footprint is automatically used for the part. The footprint variables, psmpath and padpath are fetched from the cpm file located at different levels. Search begins at the local level, so that the preference files stored locally or in your home directory take precedence over the preference files with the same name located elsewhere.

The following table describes the precedence FSP follows to read the footprint variables from cpm file located at different levels.

File	Description
<project>.cpm	<p>This is the first cpm file to be read. The <project>.cpm file contains settings that are local to a project.</p> <p>To add the footprint files specific to your project, the following section needs to be added in the cpm file.</p> <pre>START_ALLEGRO padpath '<path of the directory where the .pad files are located>' psmpath '<path of the directory where the .dra files are located>' END_ALLEGRO</pre> <p>The pathname (or variable value) is a directory search list. The FSP and layout editor looks for data in the order listed in the path.</p> <p>Note: Both relative and/or absolute paths are supported.</p> <p>When a project is loaded, the variables in the <project>.cpm file is honored.</p>
\$CDS_SITE (site.cpm)	<p>The second cpm file in the read process is from CDS_SITE. This file contains site-level settings.</p> <p>You can add the footprint variables in this file also. The footprint variables in the site.cpm file are honored if they are not specified in <project>.cpm. The variables in this cpm file will be applicable for all the projects.</p>

Allegro FPGA System Planner User Guide

Project Creation and Setup

\$CDSROOT (cds.cpm)	The third cpm file in the read process is from installation directory, that is, CDSROOT. This file contains install-level settings. You can add the variables in this file also. The footprint variables in the cds.cpm file are honored if they are not specified in <project>.cpm or site.cpm. The variables in this cpm file will be applicable for all the projects.
	<p>Note: It is not recommended to modify the cds.cpm file, as it is overwritten when you install an ISR over the standard installation.</p>
\$HOME (env)	This is the Allegro User Preference ENV file. After CDSROOT and CDS_SITE, the env file is fetched. This file contains the user-level settings. The psmpath and padpath variables defined in the User Preference Editor are saved as variables in this file. The variables defined in the env file are honored if they are not specified in the <project>.cpm, site.cpm, or cds.cpm files.

OrCAD

The footprint files are useful at the time of placing the components on the FSP canvas. After selecting the rules and schematic symbol files, FSP automatically determines a valid footprint from the path set in the footprint variables. During component browsing, the footprints are read only from the path variables defined using the User Preference Editor (Allegro User Preference Env file located at \$HOME/pcbenv/env).

Locating Sample Footprints in Install Directory

FSP's sample footprint files are installed along with the Cadence Software installation. Cadence ships a library of footprints and padstacks for FSP sample interface and device libraries. By default, footprints and padstacks libraries are copied at <install dir>/tools/fsp/samples/dra. The dra directory contains three types of files of format, .dra, .psm, and .pad. But this path is not set in the environment variable when you launch FSP for the first time. Before starting your design or invoking Component Browser, you need to set the location of footprints and padstacks in psmpath and padpath variables. Setting the psmpath and padpath variables ensures that the footprints and padstacks are accessible during component browsing. Both the variables are also read by Allegro PCB Editor to display FSP components footprint on the Allegro canvas.

In case you wish to move custom footprints or padstacks to another directory, you can copy them to another directory or at site-level. After copying to another directory make sure that the new directory paths are set in the footprint variables.

Setting Footprint Variables at Project Level

To integrate a project into HDL-based design flows, you may wish to specify search path to the dra files at the project level. The dra path variables can be defined in the `<project>.cpm` file. If the path variables are not set in the `<project>.cpm` file, FSP automatically reads the variables defined in the cpm file from other levels.

You can set project based path variables by:

- Editing `<project>.cpm`
- Using the `enved` command or running *Tool Setup* in Project Manager

To set the path variables, perform the following steps:

1. Enter the `projmgr` command in the *Run* window to invoke *Project Manager*.
The Cadence Product Choices dialog box is displayed.
2. Select an appropriate product and click *OK*.
3. Click *Open Project* and specify `<project>.cpm` file.
4. Click *Setup*.
5. Click *Tools* tab.
6. Click the *Setup* for PCB Editor.
7. Select *Paths - Library* from the Categories list.
8. Click *browse (...)* under *Value* column of *padpath* Preference.
The padpath Items dialog box is displayed.
9. Click the *New(insert)* icon to add a new row to the dialog box.
10. Paste the directory path in this row where the pad files exists and click *OK*.
11. Click *browse (...)* under *Value* column of *psmpath* Preference.
The psmpath Items dialog box is displayed.
12. Click the *New(insert)* icon to add a new row to the dialog box.

- 13.** Paste the directory path in this row where the dra files exists and click *OK*.

 *Important*

FSP always looks for dra files using the psmpath variable.

- 14.** Select the *CPM* check boxes for both padpath and psmpath.

Note: Selecting *CPM* check boxes implies that the padpath and psmpath variables will be saved in the <project>.cpm file.

- 15.** Click *OK* in the User Preferences Editor dialog box.

Once you have finalized the footprint location, you are ready to place the components on the canvas using Component Browser.

Setting Footprint Variables at Site-level

You can copy a cpm file containing footprint variables that will be accessed for all the projects that you create in FSP. You can customize the variables settings in site.cpm file by using the methods explained in earlier section.

Setting Footprint Variables in a Local env File

You can also set the path variables in your local environment file that can be accessed for all the projects. This file is located at \$HOME/pcbenv/env. You can set the path variables by using the User Preferences Editor in FSP.

Note: You can follow the steps explained in the [Setting Footprint Variables at Project Level](#) section to define the path variables using envedit command except one step. In this case, do not select the *CPM* check boxes in the *User Preferences Editor*.

This section describes the steps to set the path variables using the *User Preferences Editor* in FSP.

To set the path variables, perform the following steps:

1. Choose *Library – Edit PSMPATH*.

The User Preferences Editor is displayed.

2. Select *Paths – Library* from the Categories list.

3. Click *browse (...)* under *Value* column of *padpath* Preference.

Allegro FPGA System Planner User Guide

Project Creation and Setup

The padpath Items dialog box is displayed.

4. Click *New(insert)* icon to add a new row to the dialog box.
5. Paste the directory path in this row where the pad files exists and click *OK*.
6. Click *browse (...)* under *Value* column of psmpath Preference.

The psmpath Items dialog box is displayed.

7. Click *New(insert)* icon to add a new row to the dialog box.
8. Paste the directory path in this row where the dra files exists and click *OK*.



FSP always looks for dra files using psmpath variable.

9. Click *OK* in the User Preferences Editor dialog box.

Once you have finalized the footprint location, you are ready to place the components on the canvas using Component Browser.

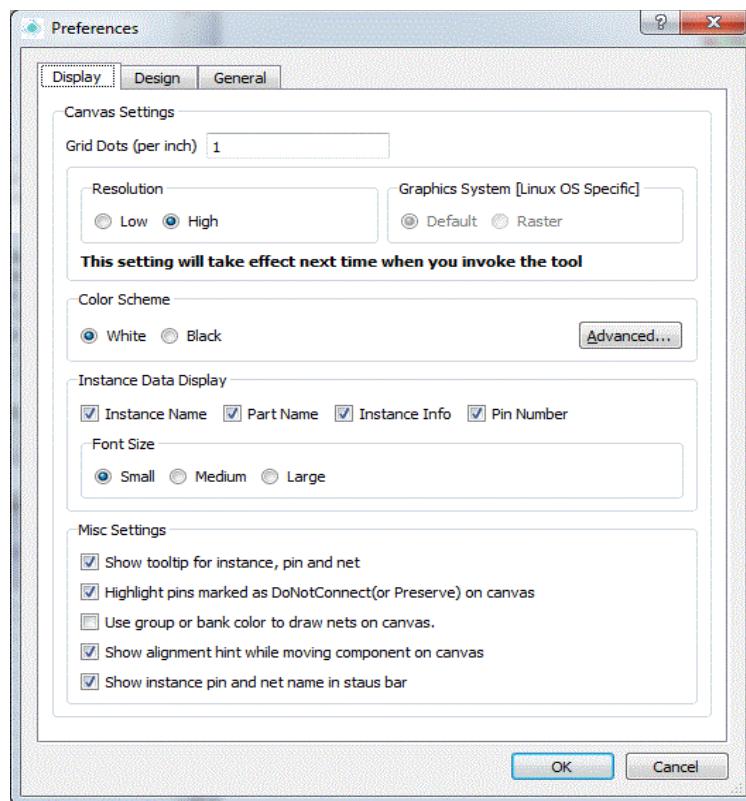
Setting up the Preference

FSP allows you to set the preferences for the PCB canvas view, instance display and more.

To specify the preferences for your project complete the following steps:

1. Choose *File – Preferences*.

The Preferences dialog box is displayed. The Preference dialog box contains four tabs.



2. In each tab of the Preference dialog box, specify the options for the canvas view.

For more information on options in each tab of the Preference dialog box, see [Preferences](#) section.

3. Click *OK*, to save the settings and close the Preference dialog box.

Opening a Project

You can open an existing project in FSP. When you open a project, the interface part definition is verified in both the project library and the central library. While opening the design, if any part with modified logical information is found, the connections of all the instances is removed. This ensures that the project's library database is in sync with the central library. In case no logical information of the part is modified, the part is referred from the Central Library.

You can open an existing project created in the earlier versions (for example 16.6 QIR5, QIR4) of FSP. You can open the project from the .cpm file or .fsp file. When you open a project from the .cpm file, the associated .fsp file in the <project_name>_lib/<project_name>/fsp directory is searched. If the fsp project file is not present in the fsp folder, a blank new project is created.



A project created in the current release cannot be opened in the earlier versions of FSP.

To open an existing project using an .fsp file complete the following steps:

1. Choose *File – Open*.
The Open Project dialog box appears.
2. Locate the <project_name>.fsp file and click *Open*.

To open an existing project using the .cpm file complete the following steps:

1. Choose *File – Open*.
The Open Project dialog box appears.
2. Locate the <project_name>.cpm file and click *Open*.
The Open Existing Project dialog box is displayed.
3. Click *OK*.

To open a recently opened project:

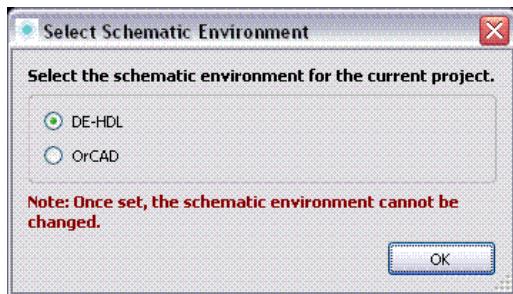
1. Choose *File – Recent Projects*.
List of recent projects appears.
2. Select the project to open.

A lock file is created every time you open a project. The lock file is located in the same folder where the project is saved. This lock file prevents others from making changes to the project. This file is removed when you close the project.

Note: If you try to open a project that is already open, a system-generated warning message appears, indicating that the project is already open. The project is then opened in the read-only mode.

Selecting Schematic Environment

After creating a new project, during opening an unsaved blank FSP design or opening a FSP sample design example, a small dialog box called Select Schematic Environment is displayed.



Select any one of the options that you want to work on and click the OK button.

Note: You cannot skip or cancel this window. You must select an option; and based on this selection of environment, some of the Generate menu is disabled also after a right-click on the device, the instance options are also disabled.

For example, if you select the DE-HDL environment and click OK, all the options and menus related to OrCAD will be disabled; i.e., Generate ->Allegro CIS and a right click on device -> Update Symbol and Update Hierarchical Schematics for OrCAD are disabled.

Saving a Project

You may save a project by using its current name and location, or you may save a copy of the project by using a different name or location.

When you make any change in the project, an asterisk (*) appears on the title bar.

[C:/Cadence/SPB_16.2/tools/fsp/examples/projects/DeepNWide/DeepNWide.scp*: Design Board]

An asterisk indicates unsaved changes.

To save a project:

1. Choose *File – Save*.
2. Click *Save*.

To save the project with a new name in different location:

1. Choose *File – SaveAs*.
2. Specify the project name and location.
3. Click *Save*.

Closing a Project

After completing the design, you can close the project. You can save and close your project as you quit.

To close a project, perform the following steps:

1. Choose *File – Close* or click *close icon* in the tool bar.

A confirmation dialog box is displayed, asking if you want to save your designs before closing the project.

2. Click *OK* to save and close the project.

Archiving a Project

You can save the project (.fsp) and all the related files (design, library, and output files) in a different directory and also create a zip file of this directory for archival purposes. You can use the *ArchiveProject* TCL command or choose *File – Archive Project* to archive your project. This command will allow you to save all the files related to your project in the directory you specify for archival and zip the directory into a single zip file, which will have a .zip extension. You can use the WinZip software to unzip zip archives created using Capture.

To archive a project, do the following:

1. Make sure that the project you want to archive is active.
2. Choose *File – ArchiveProject*.

A confirmation window appears in case the design is not saved.

3. Click *Yes* to save the design.

The *Select Project Archive Directory* dialog box appears.

4. Find and select the directory in which you want your project archived or if required, create the directory and click *Select Folder*.

FSP archives your project with all the project files to the specified directory and displays information/error messages in the *Log* window.

FSP Install Directory

The following table details the FSP install directory (%cdsroot%\tools\fsp) structure. The table below also provides a brief description of the files types associated with FSP.

Directory Names	Description
bin	Contains the fsp.exe executable file and tcl folder.
frf	Contains the Cadence-supplied FPGA library directory names. For example, V4 – Contains virtex 4 frf files. V5 – Contains virtex 5 frf files. CYIII – Contains cyclone 3 frf files.
legends	Contains the pin legends for interfaces and FPGAs.

Allegro FPGA System Planner User Guide

Project Creation and Setup

samples	<p>Contains the following directories:</p> <ul style="list-style-type: none">■ dehdl – Contains the dehdl symbol files for the Cadence-supplied interface libraries required by FSP.■ dra – Contains the footprint files such as .dra, .pad, and .psm for the Cadence-supplied interface libraries required by FSP.■ lrf – Contains the Cadence-supplied interface library directory names. <p>For example,</p> <ul style="list-style-type: none">memory_protocols – Contains memory component lrf files.connectors – Contains connector lrf files. <ul style="list-style-type: none">■ orcad – Contains the mapping file for the interface libraries.■ projects – Contains sample designs.
scripts	Contains TCL scripts used by FSP.
templates	<p>Contains the following directories and files:</p> <ul style="list-style-type: none">■ allegro – Contains FSP supported termination type files.■ orcad – Contains OrCAD capture library file for FSP supported termination.■ project – Contains FSP project template file.■ custom_attributes.txt – Custom attribute file required by FSP.■ fsp_char_support.txt – NMP checking file required by FSP.■ sch_prop_config.xml – Schematic property configuration file required by FSP.

Allegro FPGA System Planner User Guide

Project Creation and Setup

Working with Libraries

Library Overview

FSP library contains the parts that enable you to make your design. The parts are stored in form of files. A library file commonly stores the logical information of the part required by FSP. You can view or manage the libraries from Libraries. FSP supplies both FPGA and Interface models. Interface libraries are basically fixed-pin components such as memories, DSP's and microprocessors. The FPGA libraries are created and maintained by Cadence. They are delivered and installed as part of FSP.

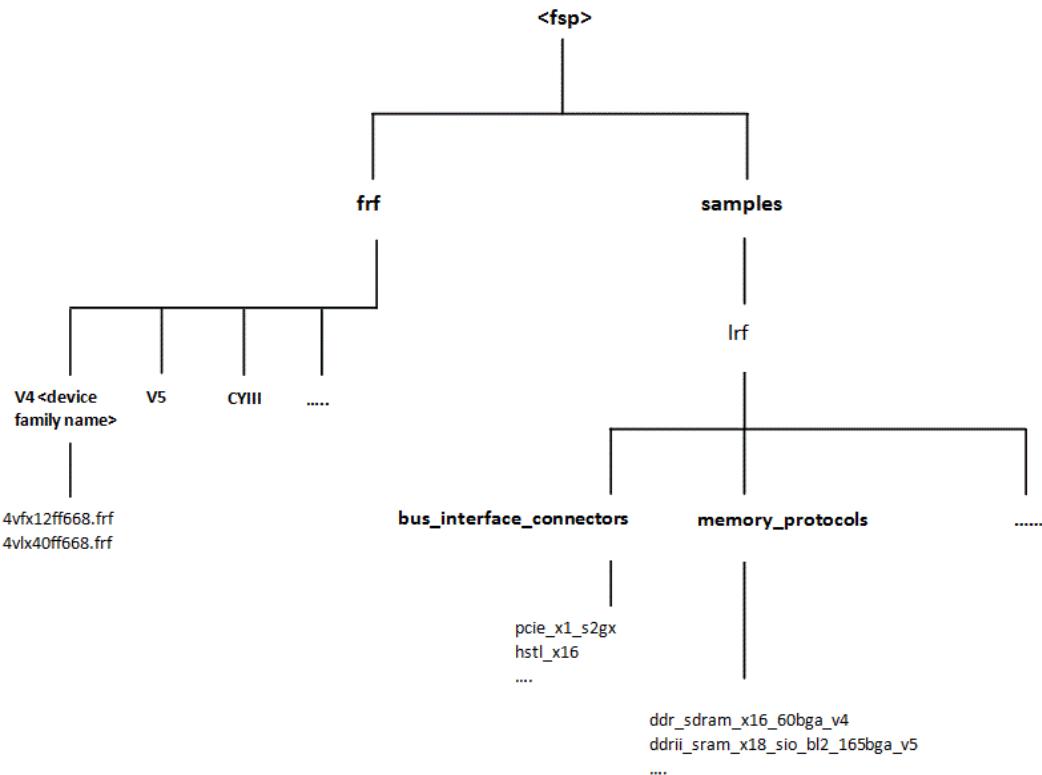
Library Structure

The libraries get installed at the time of Cadence Software installation. FSP supplies both FPGA and interface libraries. By default interface libraries are copied at <install_directory>/tools/fsp/samples/lrf and device libraries are copied at <install_directory>/tools/fsp/frf. The device libraries are further organised into directories based on the device family names. Each device family directory contains many files one for each of the parts, such as 4vfx12ff668, 4vlx60ff668. The interface libraries are organised into different directories such as custom_interface_models, memory_protocols, configuration_proms, and bus_interface_connectors.

The following figure depicts the physical representation of FSP library directory structure.

Allegro FPGA System Planner User Guide

Working with Libraries



Library Files

The FPGA models are created and maintained by Cadence. They are delivered and installed as part of FSP. There are no mechanisms within FSP, from which you can create the FPGA models.

FSP library directory contains the following library files:

- [FPGA Rules File \(.frf\)](#)
- [Interface Rules File \(.lrf\)](#)

FPGA Rules File (.frf)

The FPGA rules file is a device library file. Each device is divided into set of banks. Bank information such as bank names, pin names, pin numbers, and pin types are stored in the .frf file located in frf directory. You are not allowed to modify the device information using any forms in the FSP. To modify the FPGA rules file contact Cadence Customer Support.

Allegro FPGA System Planner User Guide

Working with Libraries

The following family devices are supported in FSP.

Vendors Name	Device Family Name
Xilinx	<ul style="list-style-type: none">■ Virtex-4, Virtex-5, Virtex-6, Virtex-7■ Spartan-3, Spartan-3A, Spartan-6■ Kintex-7■ CoolRunner-II, CoolRunner XPLA3■ System ACE
Altera	<ul style="list-style-type: none">■ Stratix II, Stratix II GX, Stratix III, Stratix IV E, Stratix IV GT, Stratix IV GX, Stratix V GX■ Arria GX, Arria II GX, Arria V GX, Arria 10■ Cyclone III, Cyclone IV E, Cyclone IV GX, Cyclone V■ Max II
Actel	<ul style="list-style-type: none">■ ProASIC3

Interface Rules File (.lrf)

The interface rules file is a logical representation of an interface model, which defines the electrical characteristics of the device as well as the logical relationships and grouping between the signals required by FSP. The pin information such as pin names, pin numbers, IO standards, pin voltages are stored in .lrf file located at `fsp/samples/lrf` directory.

Each interface rules file contains different model information, depending on which FPGA family the interface will be connected to. For example, a memory component can have several models: one for Stratix IV, another for Virtex-5 and yet another for Virtex-6. Each FPGA from vendors and even from different FPGA families from same vendor, comes up with different terminology and techniques for IO standards, clocking regions etc. The interface rules file directory contains the separate model for all the interface models, depending on which type of FPGA the interface will be connected to. The example below shows a micron component with different models targeted to different FPGA devices.

- `ddr_sdram_x16_60bga_v4.lrf`
- `ddr_sdram_x16_60bga_v5.lrf`
- `ddr_sdram_x16_66tsop2_v4.lrf`

- ddr_sdram_x16_60bga_aiigx.lrf
- ddr_sdram_x16_60bga_cyiii.lrf

Adding Interface Rules File

The interface rules files are accessed by FSP, by using a path pointer specified in the config.ini file which is automatically copied whenever Cadence tool is installed. The config.ini is the key file used by FSP to identify the libraries that can be used in your design. The file maps the interface rules file names to physical directory paths. The file contains lrf paths as variable name and location of the interface rules file as variable values. See the example below:

```
lrfpath = "<path of the interface rules file path>;"
```

You can add any number of your own interface rules file in FSP Libraries by manually adding directory paths pointer in config.ini file. Open the config.ini file in any text editor and specify the rules file directory paths in general settings section of the config.ini file. See the following example:

```
lrfpath = "<path of the interface rules file path1>;<path of the interface rules file path2>;<path of the interface rules file path3>;"
```

After specifying the rules file directory paths, all the interface rules files present in the three directories path1, 2, 3 are displayed in the Libraries.

You can also specify or modify the interface rules file search path through Settings or Edit Rules File Path dialog box.

Points to Remember

You must remember the following rules before specifying the pointer in config.ini file:

- Rules file paths should be separated by only semi colon (;
 - Any number of rules files directory path is allowed. For example,
- ```
lrfpath = <path of the interface rules file 1>; lrfpath = <path of the interface rules file 2>;
```
- Rules file directory paths should not contain any special characters except underscore “\_”. For example,

```
lrfpath = <D:\tools\fsp\samples\lrf\bpc$> and lrfpath = <D:\tools\fsp\samples\lrf\bp c>, lrfpath = <D:\tools\fsp\samples\lrf\micr@on>
```

- Keywords are case sensitive

## **Allegro FPGA System Planner User Guide**

### Working with Libraries

---

- You can enter both relative or absolute file paths.

## **Allegro FPGA System Planner User Guide**

### Working with Libraries

---

---

## **Working with Components**

---

The topics covered in this chapter are:

- [Understanding Component Placement](#)
- [Understanding Logical Mapping](#)
- [The Logical Mapping File](#)
- [Setting Up PTF for Component Selection](#)
- [Adding Interface Component](#)
- [Replacing Logical Reference Model of the Component](#)
- [Re-referencing Logical and Mapping of the Instantiated Component](#)
- [Converting Components to Real Components](#)
- [Converting Virtual Interface to Real Components](#)
- [Viewing Components](#)
- [Modifying Components](#)
- [Adding Device Component](#)
- [Linking Device Component to Front-End Symbol](#)
- [Replacing Device](#)

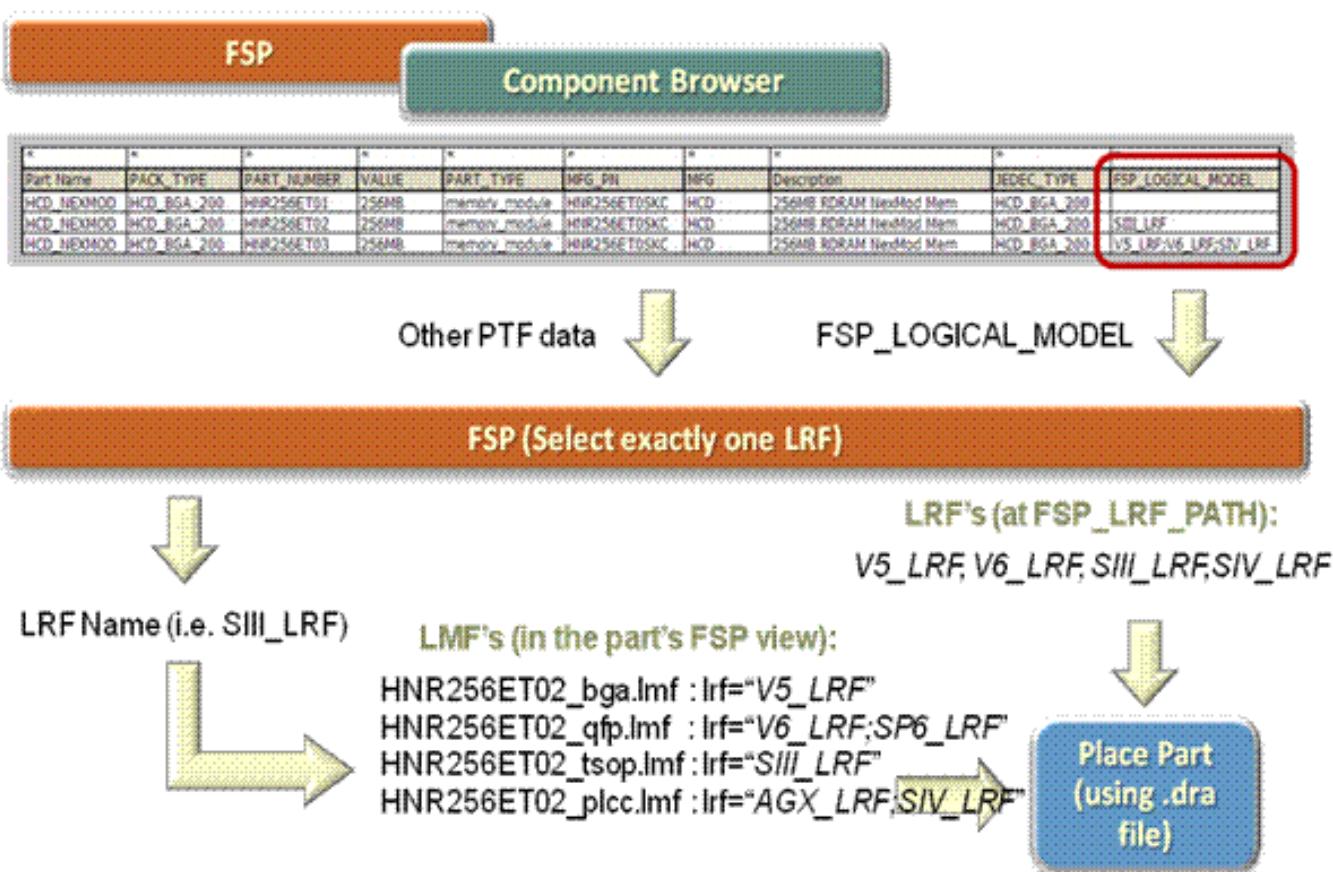
## Understanding Component Placement

The following methodologies are involved in the component placement flow. The task level details involved in both the methodologies are covered later in this chapter.

- [Creating FSP Design Using Real Components](#)
- [Creating FSP Design Using Rules File or Virtual Components](#)

### Creating FSP Design Using Real Components

The section describes the flow where you create your design by selecting and placing the symbols for your models through Component Browser on the FSP canvas.



Component Browser is invoked from FSP to select a symbol from the central library. If the PTF property is set for the symbols then available interface rules file for the selected symbol is

displayed. You select a specific PTF row for the symbol. The selected PTF row is used to derive a proper rules file for the chosen part/primitive. After selecting FSP automatically determines and select the appropriate mapping file and rules file combination for the selected symbol. Once the exact interface rules file is found the mapping file is searched in the `symbols library:cell:view` directory. The following conditions are expected at this stage:

- If only single mapping file is found in `library:cell:view` directory then by default the mapping file in `fsp` folder is selected and part is placed on canvas.
- If multiple mapping files are found then you are provided with the option to choose one out of them.
- If no mapping file is found in `library:cell:view` directory you are allowed to select the file from outside the central library. You are also allowed to create a new mapping file if you do not have a mapping file at this stage. Any new mapping file can be stored locally or can be promoted to the `fsp` folder of the `library:cell:view` structure of the cell.

Once the interface rules file and mapping file combination is selected the associate dra file is used to draw the footprint on the canvas.

## **Creating FSP Design Using Rules File or Virtual Components**

When you do not have the write permission to access the central library or cannot find the required symbol using Component Browser you can continue to use Libraries to place the rules file on canvas. You can even create a virtual interface if you still do not find your desired part in Libraries. Note that the rules file placed using Libraries or the virtual interface is just a logical rules file which means they are not mapped with any of the front-end symbol or footprint.

You can continue your design by capturing the connectivity. Once you complete your design you can convert the interface rules file or virtual interface to real component using mapping file before generating schematics.

## Understanding Logical Mapping

FSP interface model are created independent of front-end symbol pin names which exists in central library (OrCAD and DE-HDL symbol libraries). The interface rules files (.lrf) consists of logical grouping of signals, constraints and IO standards information which is required by FSP to perform IO synthesis. In order to use the front-end symbol from central library for FSP schematic generation it is required to provide mapping between interface rules file signal names and schematic symbol pin names. This mapping is achieved with the help of mapping file called logical mapping file(.lmf).

Generally mapping file is generated after mapping symbol pin names to interface rules file pin names. You may promote the mapping file to central library by copying it to the fsp folder in the `library:cell:view` structure of the cell. This will help FSP to automatically determine and select the appropriate mapping file for given primitive and interface rules file. The combination (primitive and interface rules file) for the component that your trying to place on the canvas. Since Component Browser is now used in FSP to select a part from corporate library, you can added the interface rules file name to the PTF database to obtain the proper interface rules file for the chosen part/primitive. Once the interface rules file is determined through Component Browser, the mapping file is parsed in the fsp folder of the `library:cell:view` structure to find the mapping file which contains binding information of the selected interface rules file to the primitive. The following conditions are expected at this stage:

- If only single mapping file is found in `library:cell:view` directory then by default the mapping file in fsp folder is selected and part is placed on canvas.
- If multiple mapping files are found then you are provided with the option to choose one out of them.

If no mapping file is found in `library:cell:view` directory you are allowed to select the file from outside the central library. You are also allowed to create a new mapping file if you do not have a mapping file at this stage. Any new mapping file can be stored locally or can be promoted to the fsp folder of the `library:cell:view` structure of the cell.

### The Logical Mapping File

Logical Mapping File (.lmf) is a text file, which contains mapping information of signal names of interface rules file to symbol names of front-end symbol. Sample mapping file is shipped as part of FSP installation which contains the mapping information of the supplied interface rules files and frontend symbol files. You may also use the supplied mapping files for your own components. The mapping information of the supplied mapping files can also be changed as applicable for your design, but usually not recommended.

If you do not find a desired mapping file from FSP installation directory you can create your own mapping file as required for your design. To create a new mapping file you use Rules Signal Mapper form. Rules Signal Mapper form helps you to create links between the rules file pin names and symbol pin names. This mapping file can be saved anywhere in your local disk and can be promoted later.

The sections in the mapping file (.lmf) are briefly described below:

### **Primitive Mapping Section**

Firstly in mapping file, the interface rules file name is mapped to the primitive name of the symbol.

```
<lmf lrf="connectors_ddr2_dimm_x4_s2_s2gx_agx"
primitive="s6_my_ddr2_dimm_x4_s2_s2gx_agx">
```

### **Pin Mapping Section**

The Pin Mapping section captures the mapping information of interface rules file signal names and port names of primitive.

The basic form for a pin map entry is:

```
<pin signal_name= [pin name]>" symbol_pin_name="[symbol port name]>"
```

pin name is the name of the interface rules file pin. The syntax of the interface rules file pin name is the same syntax defined in the logical rule file.

port name is the name of the symbol port. The syntax of the symbol port name is the same syntax defined in the front-end symbol file.

### **Sample Mapping Files Location**

Sample mapping files are supplied and copied along with the library files at the time of installation. These mapping files are organized in FSP installation directory based on the following flow:

- DE-HDL
- OrCAD

## DE-HDL

The mapping files for DE-HDL symbols are copied to <install\_directory>tools/fsp/samples/de-hdl. The dehdl directory is further organized into library:cell:view architecture. The default location of the mapping file in library:cell:view structure is fsp folder of the cell. Each DE-HDL library has cell and each cell has mapping file associated with it. The logical mapping file can be used while placing the component on canvas through Component Browser.

## OrCAD

The mapping files for OrCAD symbols are copied to <install\_directory>tools/fsp/samples/orcad. The orcad directory is further organized into sub directories based on the fsp interface library names. And each interface library sub directory contains symbol file and mapping files.

## Mapping Considerations

The following points are the key points that you need to remember before creating a new mapping file:

- The following data are fetched from symbol and footprint file:
  - Pin Name
  - Pin Number
  - Pin Type
  - X loc
  - Y loc
- Number of pins in both symbol file (chips.prt or .olb) and Rules file need not to be same.
- Number of pins in interface rules file should not be more than pins in symbol file (chips.prt or .olb).
- Number of pins in symbol can be more in count than the pins in Rules File. The additional pins will be declared as No Connect pins after mapping.  
  
For example, if a symbol has ten pins and interface rules file has six pins, then while mapping the remaining four pins will be set as No Connect pins.
- Pin names and pin numbers need not to be same in both chips.prt and interface rules file.

- The pin types (input, output, inout) in rules file are overridden by symbol pin types. The supply and nc pin types cannot be override in lrf. The following table summarizes how FSP handles the different pin type:

| <b>Symbol Pin Type</b> | <b>Rules file Signal Type</b> | <b>Action</b>                   |
|------------------------|-------------------------------|---------------------------------|
| IO                     | IO                            | Rules file signal type is used. |
| IO                     | Supply                        | Rules file signal type is used  |
| IO                     | NC                            | Rules file signal type is used  |

- Symbol pin numbers is used for symbols and schematic generation after mapping.
- The following pins are not considered as part of mapping process:
  - JTAG pins
  - No Connect
  - Supply pins
- In Rules Editor dialog box, the Target Pin Property column value of the unmapped schematic symbol pins (Reset or NC pins) are set as Do Not Connect.
- In Rules Editor dialog box, the Target Pin Property column value of other miscellaneous primitive pins not present in the symbol mapping file are set as Do Not Connect signals in the logical model.
- JTAG signal names is used with same name as schematic pin names.
- Mapping file does not contain No Connect pins.
- There is only one entry per power pin for the symbol pin name VDD in mapping file.
- In chips.prt file all the power pins defined under GLOBAL section are displayed for mapping in LMF Mapper dialog box.

## Mapping Scenarios

This section describes various mapping scenarios possible while creating mapping files. If you have the following:

- Interface rules file and chips.prt file
- Two logical models targeted to same family devices and a chips.prt file

■ Logical Model and Homogeneous part

**Note:** Read the Mapping Considerations section before you start mapping.

### **Interface Rules file and Chips.prt file**

In this example the chips.prt file has two primitive sections. Both the primitives has different set of pins and pin names.

#### **The chips.prt file**

##### **Symbol Pin Names**

```
mem_clock_bar
mem_clock
mem_address1
mem_address2
reset
tdo
tdi
```

##### **Primitives**

```
prim1: pin number set (A1,A2,A3, J1,J2, B1, D1,D2,D3, N1)
prim2: pin number set (A1,A2,A3, J1,J2, C1, E1,E2,E3, N1)
```

```
primitive 'prim1';
pin
 'mem_clock_bar':
 PIN_NUMBER='(A1,0,0)';
 BIDIRECTIONAL='TRUE';
 'mem_clock':
 PIN_NUMBER='(A2,0,0)';
 BIDIRECTIONAL='TRUE';
 'reset':
 PIN_NUMBER='(A3,0,0)';
 BIDIRECTIONAL='TRUE';
 'tdi_pin':
 PIN_NUMBER='(J1,0,0)';
 BIDIRECTIONAL='TRUE';
 'tdo_pin':
 PIN_NUMBER='(J2,0,0)';
 BIDIRECTIONAL='TRUE';
```

# Allegro FPGA System Planner User Guide

## Working with Components

---

```
'VREF':
 PIN_NUMBER='(B1,0,0)';
 PINUSE='POWER';
end_pin;
body
 POWER_PINS='(VDD:D1,D2,D3)';
 NC_PINS='(N1,N2)';
end_body;
end_primitive

primitive 'prim2';
pin
 'mem_clock_bar':
 PIN_NUMBER='(A1,0,0)';
 BIDIRECTIONAL='TRUE';
 'mem_clock':
 PIN_NUMBER='(A2,0,0)';
 BIDIRECTIONAL='TRUE';
 'mem_address1':
 PIN_NUMBER='(A2,0,0)';
 BIDIRECTIONAL='TRUE';
 'mem_address2':
 PIN_NUMBER='(A2,0,0)';
 BIDIRECTIONAL='TRUE';
 'reset':
 PIN_NUMBER='(A3,0,0)';
 BIDIRECTIONAL='TRUE';
'VREF':
 PIN_NUMBER='(C1,0,0)';
 PINUSE='POWER';
end_pin;
body
 POWER_PINS='(VIN:E1,E2,E3)';
 NC_PINS='(N1,N2)';
end_body;
end_primitive
```

## Interface Rules File

```
<lrf type="interface" name="x4ddr32" target_family="s2"
<group name="g1" constraint="dqs_group"
```

## Allegro FPGA System Planner User Guide

### Working with Components

---

```
<signal name="ddr_cq_n" type="inout" diff="ddr_cq" dontcon="true" />
<signal name="ddr_cq" type="inout" diff="ddr_cq_n" />
/group>
/lrf>
```

Few points from example above:

- Number of pins in logical model and primitive1 are same.
- Number of pins in primitive2 and logical model are different. In addition the pins in primitive2 is more than the pins in logical model.

Considering the above points, you need to create two mapping files.

The following is the first logical mapping file:

#### ***Mapping File 1***

```
<lmf primitive="prim1" lrf="x4ddr32__S2"
 <map pin="mem_clock_bar" signal="ddr_cq_n" />
 <map pin="mem_clock" signal="ddr_cq_p" />
 <map pin="reset" />
 <power name="VDD" voltage="1.8" />
 <power name="VREF" voltage="0.9" />
/lmf>
```

The following pins are mapped:

---

<b>Prim1 Pin Name</b>	<b>Rules File Pin Name</b>
mem_clock_bar	ddr_cq_n
mem_clock	ddr_cq_n
reset	will be set as do not connect

---

#### ***Mapping File 2***

```
<lmf primitive="prim2" lrf="x4ddr32__S2"
 <map pin="mem_clock_bar" signal="ddr_cq_n" />
 <map pin="mem_clock" signal="ddr_cq_p" />
 <map pin="mem_address1" />
 <map pin="mem_address2" />
 <map pin="reset" />
 <power name="VDD" voltage="1.8" />
```

## Allegro FPGA System Planner User Guide

### Working with Components

---

```
<power name="VREF" voltage="0.9" />
/lmf>
```

The following pins are mapped:

---

Prim2 Pin Name	LRF Pin Name
mem_clock_bar	ddr_cq_n
mem_clock	ddr_cq_n
mem_address1	will be set as do not connect
mem_address2	will be set as do not connect
reset	will be set as do not connect

---

### Two interface rules file (targeted to same family devices) and a chips.prt file

In this example the chips.prt file has only primitive section.

#### Symbol Pin Names

```
sdram_A
sdram_B
sdram_C
sdram_D
sdram_Y
reset
tdo
tdi
```

#### Primitives

```
prim1: pin number set (A1,A2,A3,A4,A5 J1,J2, B1, D1,D2,D3, N1)
```

```
primitive 'prim1';
pin
'sdram_A':
 PIN_NUMBER='(A1,0,0)';
 BIDIRECTIONAL='TRUE';
'sdram_B':
 PIN_NUMBER='(A2,0,0)';
 BIDIRECTIONAL='TRUE';
'sdram_C':
 PIN_NUMBER='(A3,0,0)';
```

# Allegro FPGA System Planner User Guide

## Working with Components

---

```
BIDIRECTIONAL='TRUE';
'sdram_D':
 PIN_NUMBER='(A4,0,0)';
 BIDIRECTIONAL='TRUE';
'sdram_Y':
 PIN_NUMBER='(A5,0,0)';
 BIDIRECTIONAL='TRUE';
'tdi_pin':
 PIN_NUMBER='(J1,0,0)';
 BIDIRECTIONAL='TRUE';
'tdo_pin':
 PIN_NUMBER='(J2,0,0)';
 BIDIRECTIONAL='TRUE';
'VREF':
 PIN_NUMBER='(B1,0,0)';
 PINUSE='POWER';
end_pin;
body
 POWER_PINS='(VDD:D1,D2,D3)';
 NC_PINS='(N1,N2)';
end_body;
end_primitive
```

### **Interface Rules File 1**

```
<lrf type="interface" name="x4ddr32" target_family="V4__V5"
 <group name="g1" constraint="same_clock_region" clock="cq"
 <signal name="cq_n" type="inout" diff="cq" />
 <signal name="cq" type="inout" diff="cq_n" />
 /group>
/lrf>
```

### **Interface Rules File 2**

```
<lrf type="interface" name="x4ddr32" target_family="sp6"
 <group name="g1" constraint="sp6_tile" clock="cq"
 <signal name="cq_n" type="inout" diff="cq" />
 <signal name="cq" type="inout" diff="cq_n" />
 /group>
/lrf>
```

From example above, the following points need to be considered:

- Both the interface rules files 1 and 2 are same since they are targeted to same Xilinx family devices Virtex-4 and Virtex-5(V4\_V5) and other Spartan-6(SP6).
- Pin names of interface rules files 1 and 2 are also identical.
- Number of pins in primitive 1 and interface rules file 1 are equal.
- Number of pins in primitive 2 is more than the pins in interface rules file 2.

A single mapping file can be used to capture the mapping information. You can also capture the mapping information in two separate mapping files. One for primitive1 and interface rules file1 and another for primitive1 and interface rules file 2. But this creates unnecessary duplication of mapping files across cells.

### **Mapping File**

```
<lmf primitive="prim1" lrf="x4ddr32__V4__V5, x4ddr32__SP6"
 <map pin="sdram_A" signal="cq_n" />
 <map pin="sdram_B" signal="cq_n" />
 <map pin="sdram_C" />
 <map pin="sdram_D" />
 <map pin="sdram_Y" />
 <map pin="reset"/>
 <power name="VDD" voltage="1.8" />
 <power name="VREF" voltage="0.9" />
 <jtag pin="tdo_pin" property="tdo" />
 <jtag pin="tdi_pin" property="tdi" />
 /lmf>
```

The following pins are mapped:

---

<b>Prim1 Pin Name</b>	<b>Interface Rules File 1/2 Pin Name</b>
sdram_A	cq_n
sdram_B	cp_p
sdram_C	Will be set as Do not connect
sdram_D	Will be set as Do not connect
sdram_Y	Will be set as Do not connect
reset	Will be set as Do not connect

---

## Interface Rules File and Homogeneous part

Homogenous parts are also represented multiple identical logical parts, each part represents a different logic. In case of a homogenous parts, each of the pins will be flattened separately.

To map such parts, the pins in chips.prt file is first read by FSP. Then all the pins are prefixed with following name template to generate a unique pin name.

S(\$split\_number)

For detailed information on how FSP converts the pin names of a homogenous part see Creating Part from DE-HDL symbol section.

An example of an homogeneous part is displayed below:

```
FILE_TYPE=LIBRARY_PARTS;
TIME=' COMPILE ON THU JAN 10 14:52:02 1991 ';
primitive '74LS241','74LS241_DIP';
pin
'A1'<0>;
OUTPUT_LOAD='(24.0,-15.0)';
INPUT_LOAD='(-0.02,0.02)';
OUTPUT_TYPE='(TS,TS)';
PIN_NUMBER='(12,14,16,18,0,0,0,0)';
'B1'<0>;
INPUT_LOAD='(-0.2,0.02)';
PIN_NUMBER='(8,6,4,2,0,0,0,0)';
'Y1':
INPUT_LOAD='(-0.2,0.02)';
PIN_NUMBER='(1,1,1,1,0,0,0,0)';
'Y0'<0>;
OUTPUT_LOAD='(24.0,-15.0)';
INPUT_LOAD='(-0.02,0.02)';
OUTPUT_TYPE='(TS,TS)';
PIN_NUMBER='(0,0,0,0,3,5,7,9)';
'E0':
INPUT_LOAD='(-0.2,0.02)';
PIN_NUMBER='(0,0,0,0,19,19,19,19)';
'AE1'<0>;
INPUT_LOAD='(-0.2,0.02)';
PIN_NUMBER='(0,0,0,0,17,15,13,11)';
end_pin;
body
```

## Allegro FPGA System Planner User Guide

### Working with Components

---

```
POWER_PINS='(VCC:20;GND:10)';
FAMILY='LSTTL';
PART_NAME='74LS241';
BODY_NAME='LS241';
DEFAULT_SIGNAL_MODEL='SN74LS241N TI';
JEDEC_TYPE='DIP20_3';
CLASS='IC';
TECH='74LS';
end_body;
end_primitive;
```

The mapping file is automatically generated by FSP if you map the above chips.prt file with interface rules file. See below.

```
<lmp lrf="homogeneous_part" primitive="homogeneous_part" >
<pin pin_voltage="1.5" symbol_pin_name="VCC" />
<pin pin_voltage="0" symbol_pin_name="GND" />
<pin signal_name="S1_A1" symbol_pin_name="S1_A1" />
<pin signal_name="S1_B1" symbol_pin_name="S1_B1" />
<pin signal_name="S1_Y1" symbol_pin_name="S1_Y1" />
<pin signal_name="S1_Y0" symbol_pin_name="S1_Y0" />
<pin signal_name="S1_E0" symbol_pin_name="S1_E0" />
<pin signal_name="S1_AE1" symbol_pin_name="S1_AE1" />
<pin signal_name="S2_A1" symbol_pin_name="S2_A1" />
<pin signal_name="S2_B1" symbol_pin_name="S2_B1" />
<pin signal_name="S2_Y1" symbol_pin_name="S2_Y1" />
<pin signal_name="S2_Y0" symbol_pin_name="S2_Y0" />
<pin signal_name="S2_E0" symbol_pin_name="S2_E0" />
<pin signal_name="S2_AE1" symbol_pin_name="S2_AE1" />
</lmp>
```

## Creating Mapping File

The logical mapping file is created using Rules Signal Mapper. You use the Rules Signal Mapper for creating mapping files at following stages:

- Placing interface rules file through Add Part dialog box
- Converting interface rules file to real component

## **Placing Interface Rules File through Component Browser**

After creating a new project, you choose Add Part dialog box to select and place the interface rules file. While browsing and selecting interface rules file, you will be asked to select existing mapping file or create a new mapping file. For creating a new mapping file you invoke Rules Signal Mapper. The Rules Signal Mapper helps you to automatically or manually map the names. Once the mapping file creation is completed, the consistency between the pin names of interface rules file and front-end symbol is validated and all the pin information are processed by FSP. Once it successfully done the interface rules file is ready for schematic generation.

The following steps demonstrate only the steps that are required while working with Rules Signal Mapper form. The Rules Signal Mapper form is available in the Convert Rules File Instance to Real Part dialog box.

1. Click *Define Mapping* to create a new logical mapping file, in the Convert Rules File Instance to Real Part dialog box.

The Rules Signal Mapper dialog box is displayed. The Part Symbol Data Page displays the port names and pin types of the front-end symbol. The Rules File Signals page displays the pin names and pin types of the interface rules file. For more information on fields and buttons of Rules Signal Mapper dialog box see [Rules Signal Mapper](#) section.

2. Do the following to manually map the pin names and symbol pin names:

- ❑ In the *LRF Signals* pane select a single logical pin and drag to the associated *Signal Name* column cell and drop.

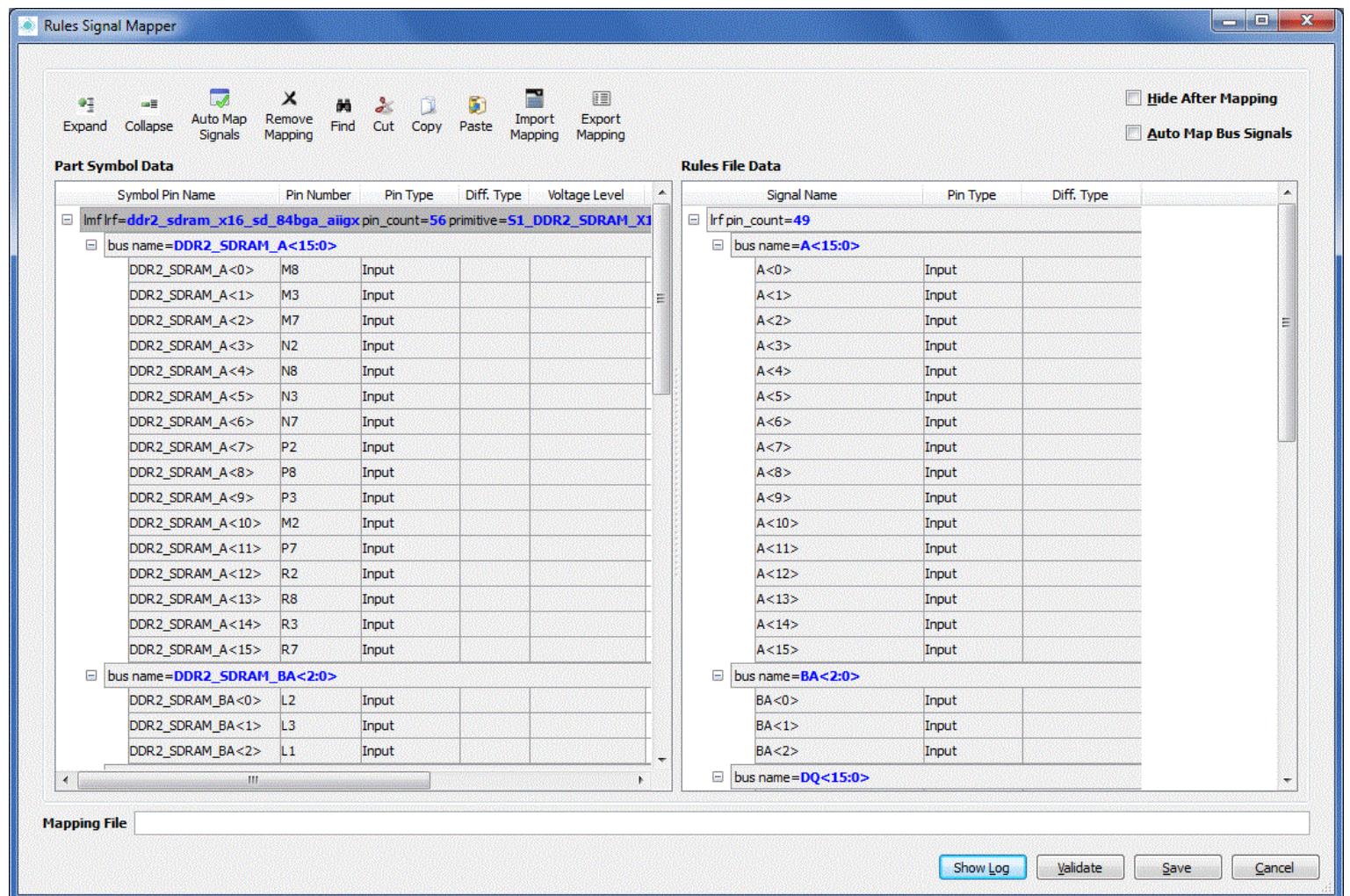
The Logical Pin Name is copied in the cell.

- ❑ For Multi selection, in the *Rules File Signals* pane select group of signals and drag to the *Signal Name* column and drop.

# Allegro FPGA System Planner User Guide

## Working with Components

---



3. Click *Auto Map Signals* tool bar icon to auto map the pins.

The Auto Map Symbol Pins dialog box is displayed.

4. Specify the pattern to be excluded in respective text boxes and click *Fetch*.

After clicking Fetch, the symbol pin names and associated signal names are displayed.

5. Click *Check All*.

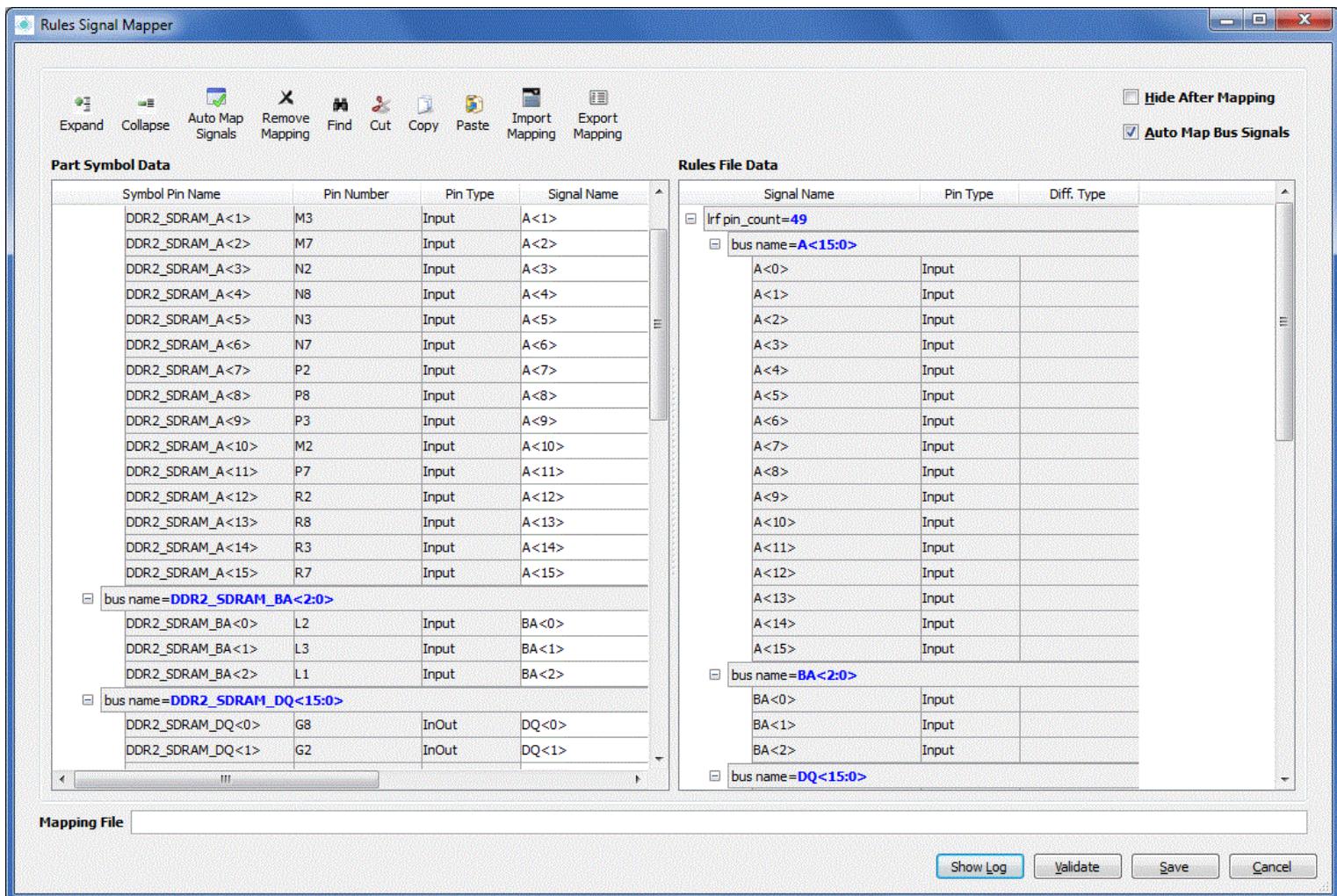
6. Click *Map*.

All the signal names of the interface rules file is copied (mapped) to the Signal Name column.

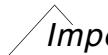
# Allegro FPGA System Planner User Guide

## Working with Components

---

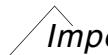


7. Specify the mapping file name and path in *Mapping File* text box or click ... to browse to the location where you want to save the mapping file.
8. Click *Validate* to verify the consistency between symbol pin names and interface rules file signal names.
9. Enter the name for the mapping file that you are creating in the *Mapping File* field and click *Save*.

 *Important*

By default the mapping file is saved in the working directory. It is recommended not to use absolute and relative paths while saving the mapping file. Saving the mapping file using relative or absolute paths may lead to error prone state when you next time open the project.

This completes the creation of logical mapping file.

 *Important*

The above methods are also applicable while working in OrCAD schematic environment.

### **Converting Interface Rules File to Real Component**

It is not necessary to map the pin and signal names at starting of the design. It can be done at later stage also. Once the design is complete, you may decide to convert the selected interface rules file to real component.

For detailed information see [Converting Components to Real Components](#) for more information.

## Setting Up PTF for Component Selection

Generally, the rules file are controlled by rules path setup and symbols are controlled by cds.lib file and project.cpm file setup. In order to auto link the specific symbol to rules file, you have a choice to update the part PTF.

You can also place the part on canvas with given symbol and rules file without updating the PTF file. In such cases, you need to provide required rules file and mapping information in PTF file before placing the component on canvas.

Besides the part models, FSP installation also includes the PTF files. The DE-HDL symbol library directory is organized into `library:cell:view` architecture. The default location of the PTF file in `library:cell:view` structure is PTF folder of the cell. Each DE-HDL library has cell and each cell has PTF associated with it. These PTF files contains the physical information for the parts defined in the chips.prt and the rules file name associated with the front-end symbol (chips.prt). Component Browser displays the list of symbols which has the properties added in the PTF file. Two different types of PTF properties are supported for both interface and device components. These properties helps Component Browser to identify the components as FSP components.

- [FSP\\_LOGICAL\\_MODEL](#)
- [FSP\\_FPGA](#)

### ***FSP\_LOGICAL\_MODEL***

The `FSP_LOGICAL_MODEL` property in the PTF file contains the name (s) of the interface rules file (s) that are available for the selected symbol. If the selected symbol has the property then available interface rules file (s) for the symbol is displayed in Component Browser.

Consider an example,

Each FSP interface component contains different models depending which FPGA family the interface is targeted to. For example a micron component has following models:

- `micron_mt46v32m16_CYIII.lrf`
- `micron_mt46v32m16_CYIV.lrf`
- `micron_mt46v32m16_S2_S2GX_AGX.lrf`
- `micron_mt46v32m16_S3.lrf`

In order to link any one out of these models to the specific symbol at the time of placing the component through Create>Select Component Rules and Mapping File wizard you need to

# **Allegro FPGA System Planner User Guide**

## Working with Components

add the above model names as values of the FSP\_LOGICAL\_MODEL property. The FSP\_LOGICAL\_MODEL property value should be specified in semi colon (;) separated values.

FSP will read the `FSP_LOGICAL_MODEL` property to determine the appropriate interface rules file for the chosen part/primitive. The following conditions are expected at this stage:

- If FSP\_LOGICAL\_MODEL contains a single value then by default the interface rules file is selected.
  - If FSP\_LOGICAL\_MODEL contains multiple value then you have the option to choose one out of them.
  - If FSP\_LOGICAL\_MODEL is empty or does not exist, then you are allowed to select the file from outside the central library. If no interface rules file are found in your local directory then you can create a new interface rules file.

**Note:** A new interface rules file requires a new mapping file.

FSP FPGA

For device component selection the `FSP_FPGA` property in the PTF file is used. To support selection of device component from the corporate library using Component Browser this PTF value must be defined. The `FSP_FPGA` property enables the Component Browser to recognize the symbol as device component. See below example,

```
:PART_NAME = FSP_FPGA ;
{=====
=====}
ff1152 = 4vfx100ff1152
```

The following conditions are expected while browsing the device component using Component Browser:

- If FSP\_FPGA contains a value the selected device footprint pin numbers are checked against FSP device model for consistency.
    - If the pin numbers are consistent the FPGA footprint from central library is used to place the device component onto the FSP canvas.

## **Allegro FPGA System Planner User Guide**

### Working with Components

---

- ❑ If the pin numbers are inconsistent, then you must use the Libraries to select the correct FPGA.
- If FSP\_FPGA value is empty or incorrect you will be prompted with warning message to use Libraries to place the FPGA.

## Adding Interface Component

Interface component placement procedure varies based on the selected schematic environment. The placement procedure is classified into following flows:

- [Adding Interface Component \(DEHDL\)](#)
- [Adding Interface Component \(OrCAD\)](#)
- [Adding Interface Component \(Libraries\)](#)

### Adding Interface Component (DEHDL)

In DE-HDL schematic environment, the Component Browser form is used to browse the front-end symbol libraries which are in the installation directory. You can also make the Component Browser to access your local design libraries or from site-level. After selecting the front-end symbol you will be walked through a wizard called Create>Select Component Rules and Mapping Information. This wizard will guide you through a series of steps and tasks that are required to link the selected symbol to interface rules files and place it on canvas. There is another option to place the interface rules file on canvas in order to avoid the wizard steps called Add Part dialog box. To use the Add Part dialog box first you must hide the wizard by selecting *Show wizard while placing part on canvas* option in Preferences window. After selecting this option, the Add Part dialog box is invoked inplace of wizard. The Add Part dialog box allows you to quickly select the rules file and mapping file and place it on canvas.

To place the interface component on canvas you can choose any of the following methods: .

- [Using Create>Select Component Rules and Mapping Information Wizard](#)
- [Using Add Part dialog box](#)

**Note:** The Component Browser is available as primary interface for front-end symbol selection in both the methods above.

### Using Create>Select Component Rules and Mapping Information Wizard

Following are the four scenarios possible while placing the interface rules file through wizard:

#### Scenario   Rules File   Mapping File   Procedure

1	Yes	Yes	Read <a href="#"><u>Steps Required for Scenario 1</u></a> section.
---	-----	-----	--------------------------------------------------------------------

## Allegro FPGA System Planner User Guide

### Working with Components

---

2	Yes	No	Read <a href="#">Steps Required for Scenario 2</a> section.
3	No	Yes	Read <a href="#">Steps Required for Scenario 3</a> section.
4	No	No	Read <a href="#">Steps Required for Scenario 4</a> section.

#### **Note:**

- To enable the wizard invoke, select *Show Wizard while placing part on canvas* option in Preference Window.
- In every page of the wizard Finish & Place option is common. This option helps you to place the interface rules file on canvas at any time with the information provided so far. If you have both interface rules file and mapping file, then you can use the Finish & Place option at any time. Once you click the option, the interface rules file and its associated mapping file are automatically detected by FSP.

#### **Steps Required for Scenario 1**

1. To invoke Component Browser click *Add Part* icon in .  
The Component Browser is displayed.
2. Select a library name in Library pane, whose component you want to bind with the FSP logical model.
3. Select a cell name in *Cells* pane or enter the name of cell in *Cells* text box.
4. In *Search Results* pane, click the row corresponding to the physical part you want to add.

**Note:** For more information on assigning `FSP_LOGICAL_MODEL` property in PTF file see [Setting Up PTF for Component Selection](#) section.

The symbol and footprint name for the component is displayed in *Part Name* tab.

5. Select a symbol view in the *Symbol* drop-down list.

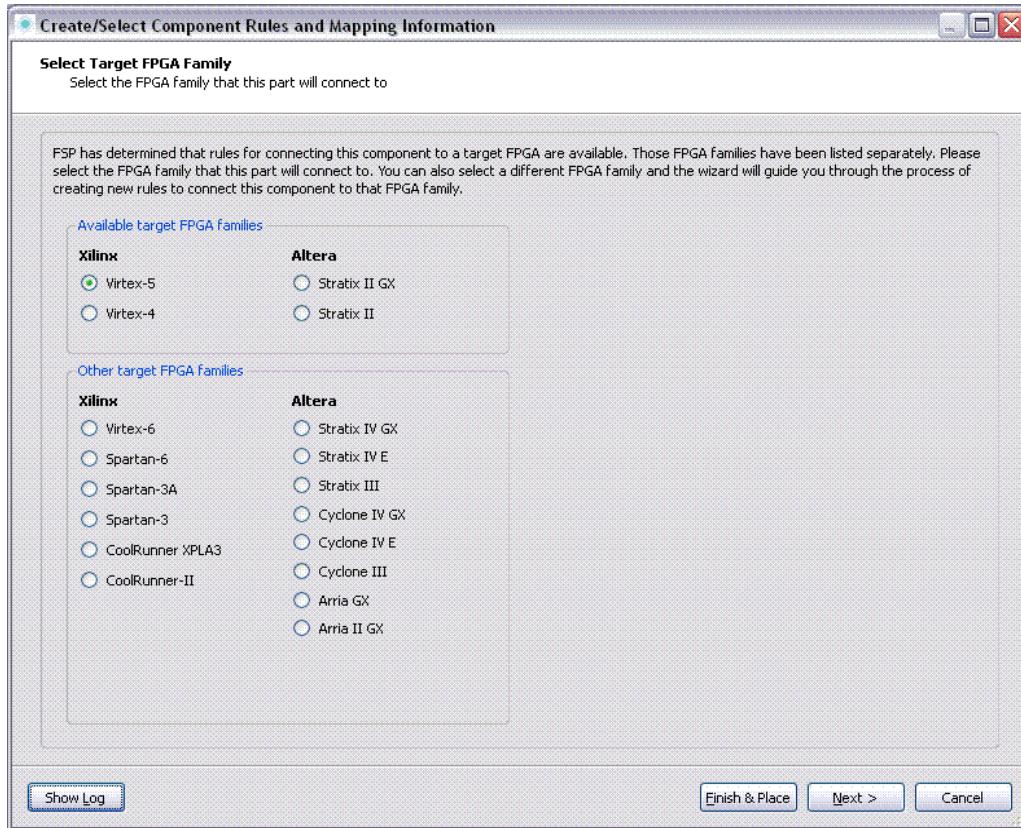
**Note:** If a symbol has one or more symbol views, you can select any one of the views of your choice. The selected view will be used during DE-HDL schematic generation.

6. Click *Select*.

The Select Target FPGA Family page of Create/Select Component Rules and Mapping Information wizard is displayed. By default an FPGA family name is selected.

# Allegro FPGA System Planner User Guide

## Working with Components

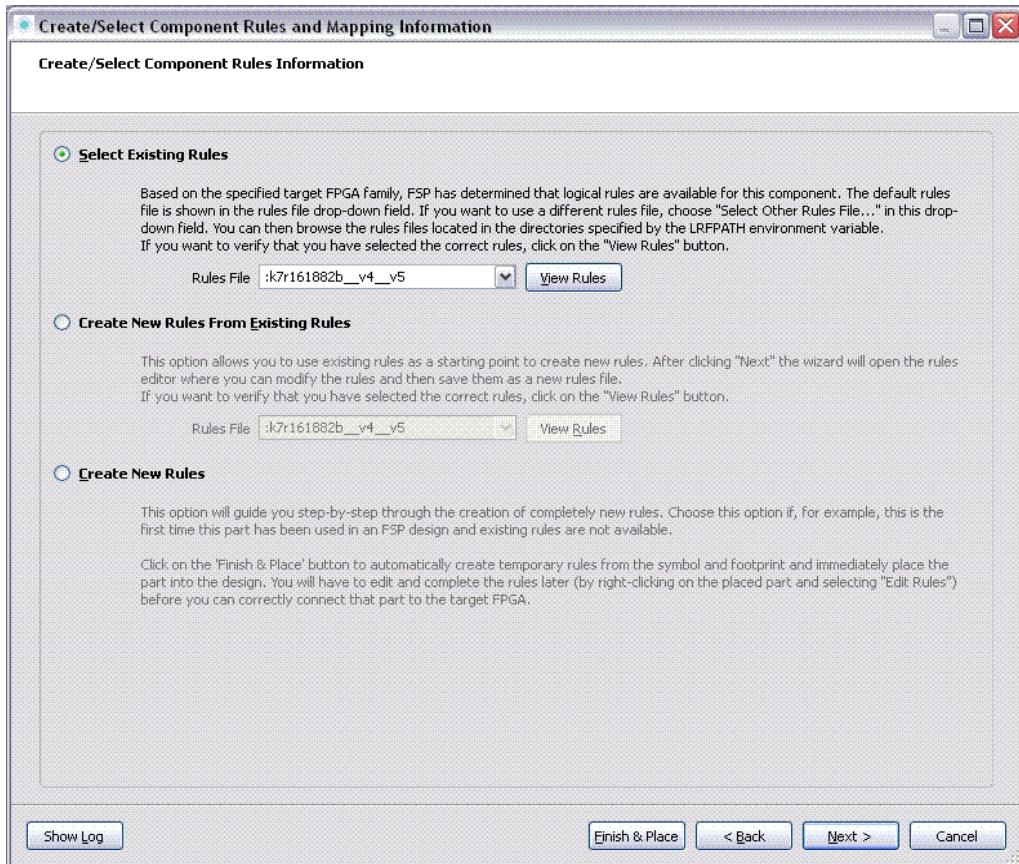


7. Click *Next* to advance to Create/Select Component Rules Information page.

The Create/Select Component Rules Information page is displayed. By default the Select Existing Rules option is selected.

# Allegro FPGA System Planner User Guide

## Working with Components

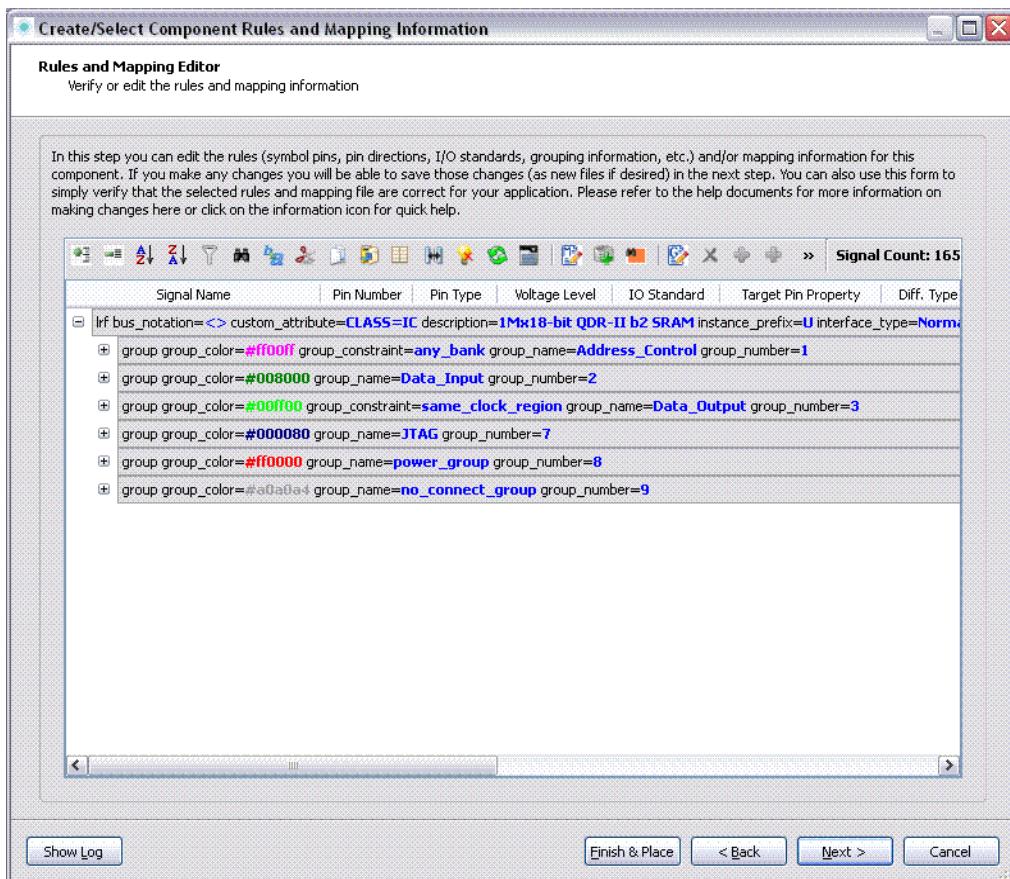


8. Click the Rules file drop down list to select a different rules file or you can proceed with the displayed one.
9. Click *Next* to advance to the Rules and Mapping Editor page, or you can click *Finish & Place* to place the component on canvas.

The Rules and Mapping Editor page is displayed.

# Allegro FPGA System Planner User Guide

## Working with Components

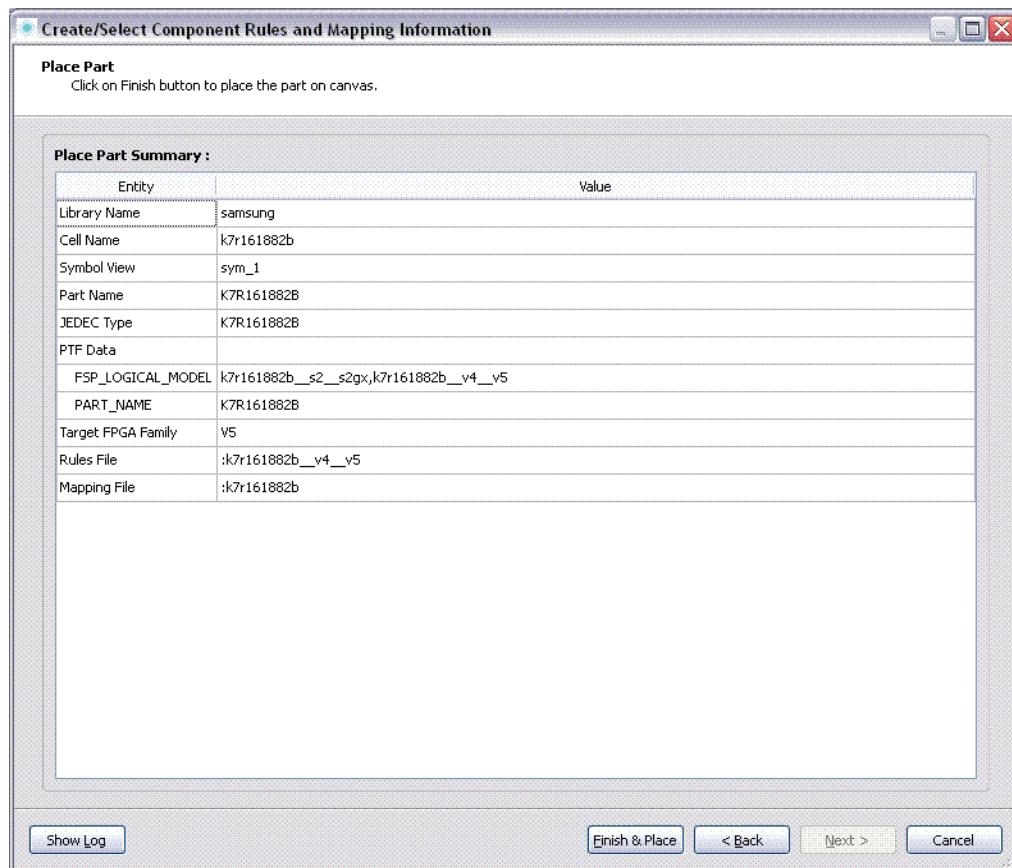


### 10. Modify the constraints as required and click *Next*.

The Place Part page is displayed. This page displays the summary of the part and its details.

# Allegro FPGA System Planner User Guide

## Working with Components



11. Click *Finish & Place* to place the part on canvas.

After clicking Finish & Place, a graphical view of component is displayed. Left click to drop the component on canvas and right click to disable the graphical view.

### Steps Required for Scenario 2

1. To invoke Component Browser click *Add Part* icon in .  
The Component Browser is displayed.
2. Select a library name in Library pane, whose component you want to bind with the FSP logical model.
3. Select a cell name in Cells pane or enter the name of cell in Cells text box.
4. In Search Results pane, click the row corresponding to the physical part you want to add.

**Note:** For more information on assigning FSP\_LOGICAL\_MODEL property in PTF file see [Setting Up PTF for Component Selection](#) section.

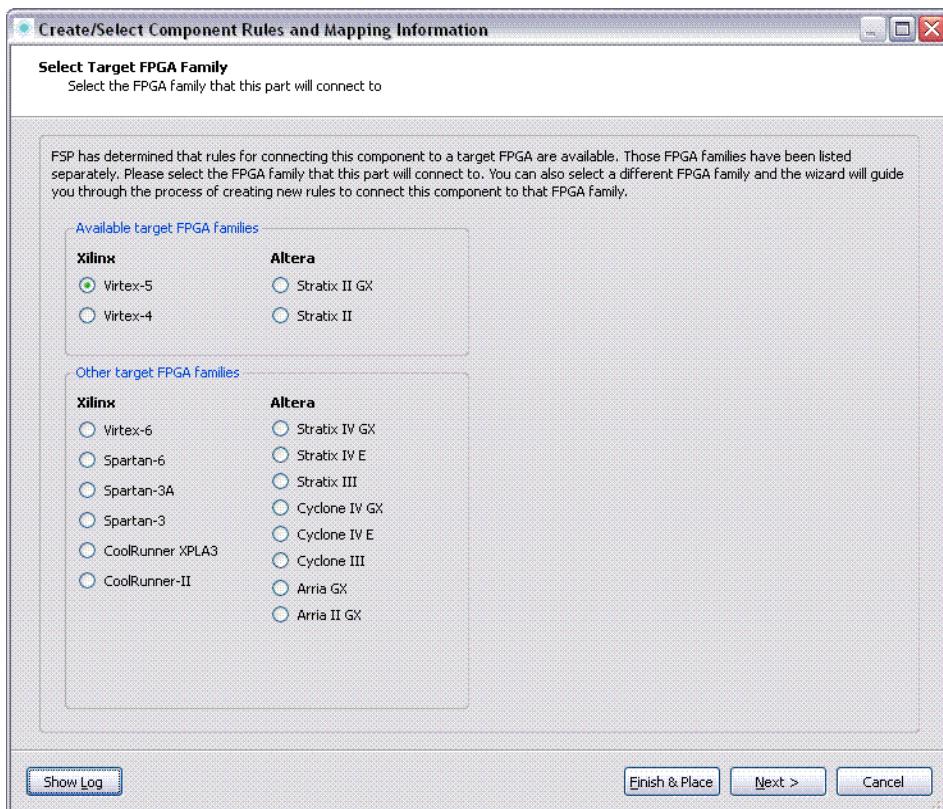
## Allegro FPGA System Planner User Guide

### Working with Components

The symbol and footprint name for the component is displayed in *Part Name* tab.

5. Select a symbol view in the *Symbol* drop-down list.
6. Click *Select*.

The Select Target FPGA Family page of Create/Select Component Rules and Mapping Information wizard is displayed. By default an FPGA family name is selected.

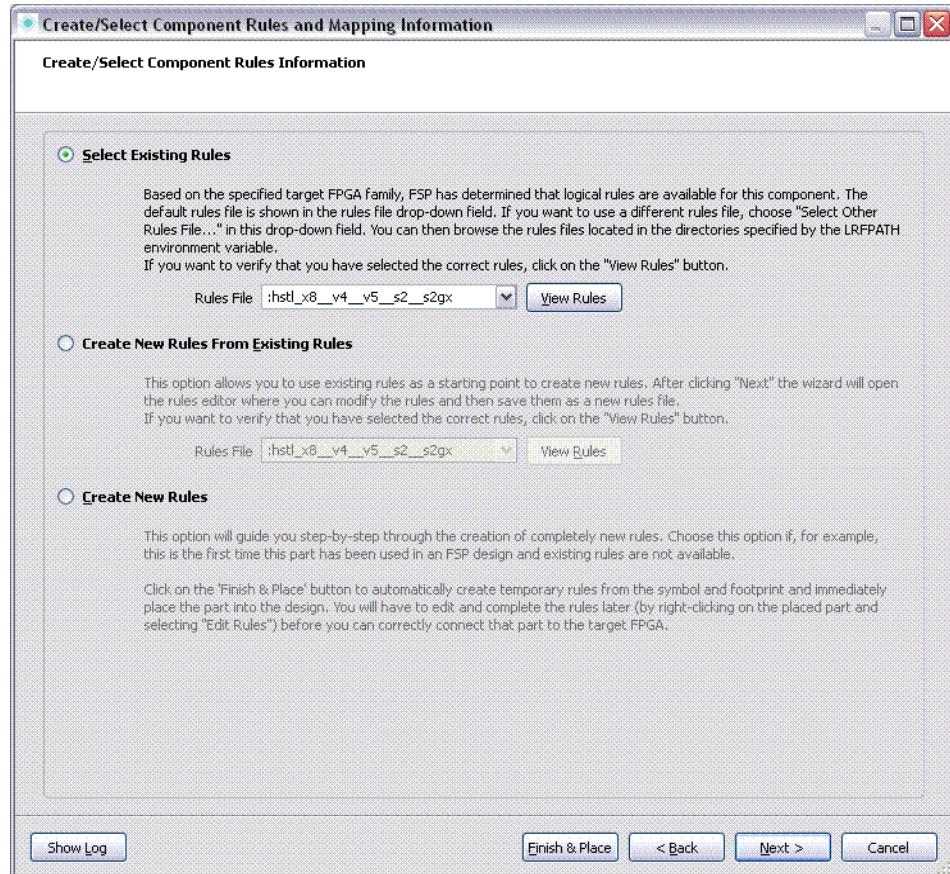


7. Click *Next* to advance to Create/Select Component Rules Information page.

The Create/Select Component Rules Information page is displayed. By default the Select Existing Rules option is selected.

# Allegro FPGA System Planner User Guide

## Working with Components

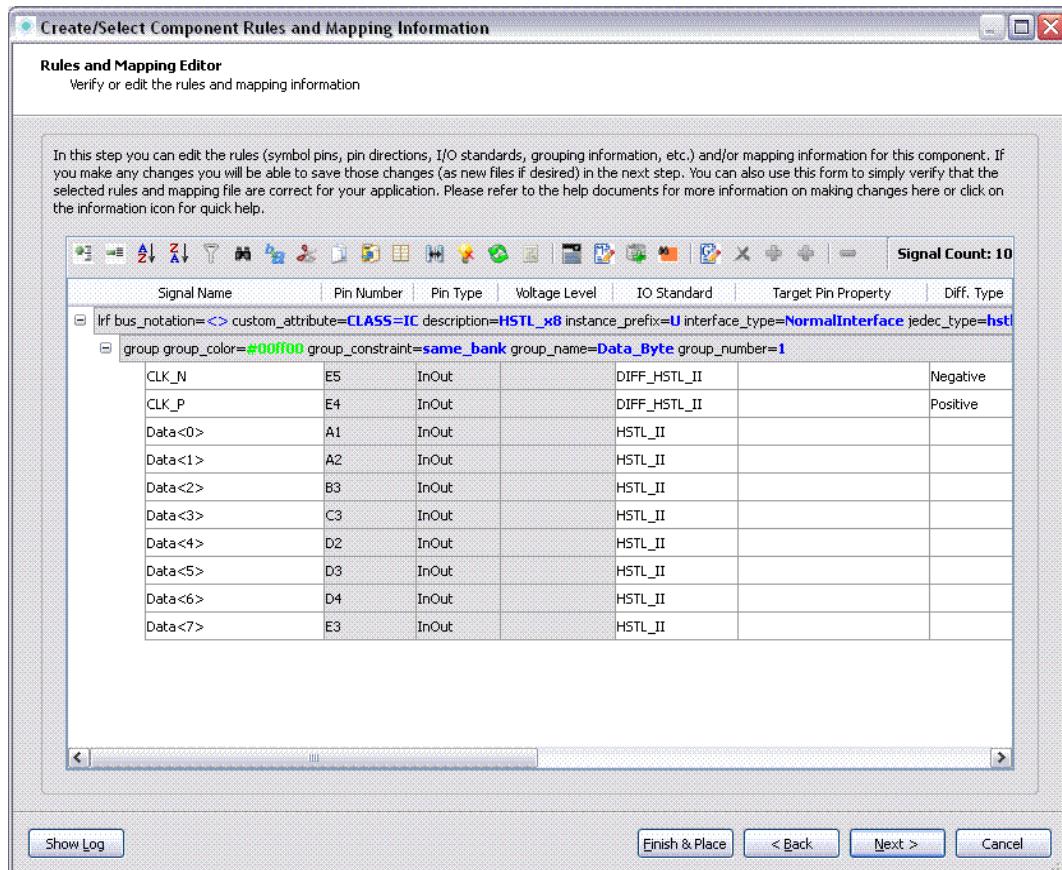


8. Click the Rules file drop down list to select a different rules file or you can proceed with the displayed one.
9. Click *Next* to advance to the Rules and Mapping Editor page, or you can click *Finish & Place* to place the component on canvas.

The Rules and Mapping Editor page is displayed.

# Allegro FPGA System Planner User Guide

## Working with Components

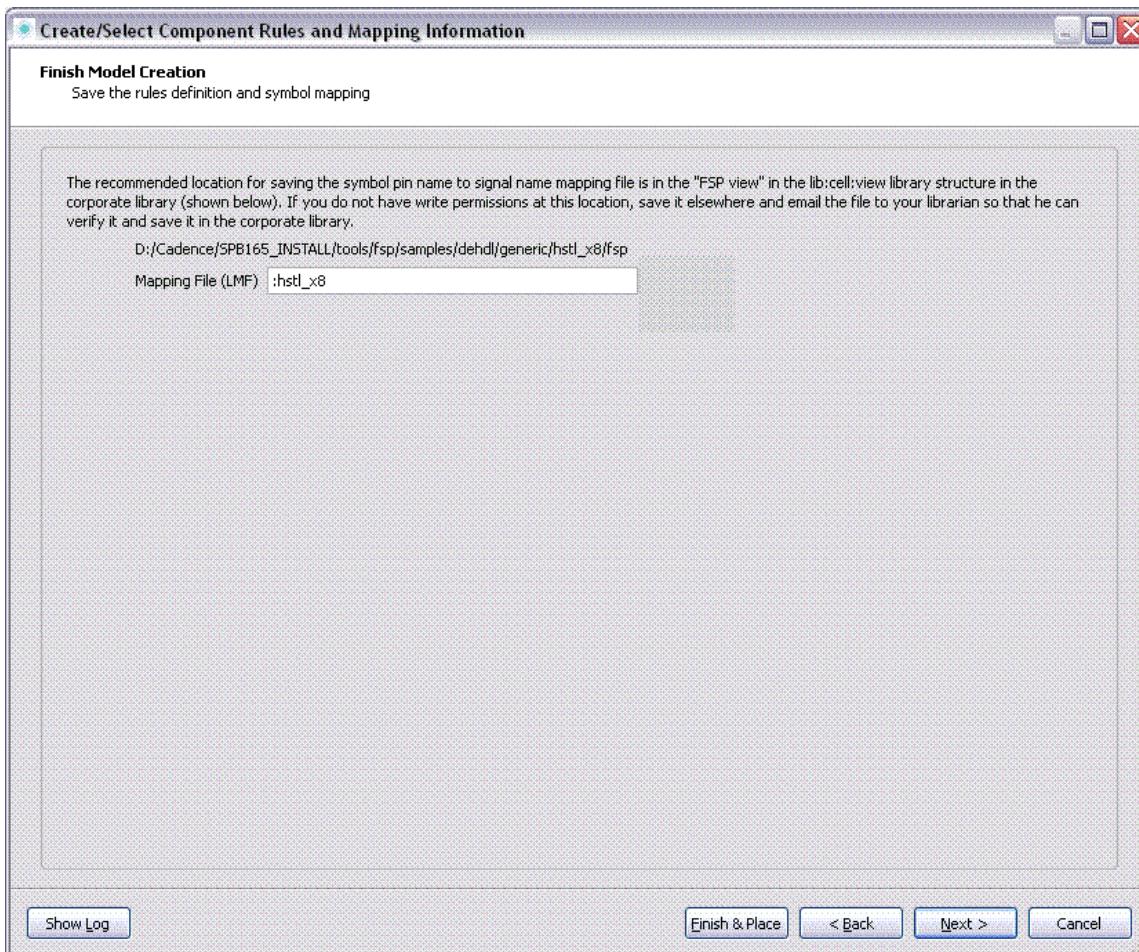


10. Modify the constraints as required and click *Next*.

The Finish Model Creation is displayed.

# Allegro FPGA System Planner User Guide

## Working with Components



11. Enter the name for the mapping file in the *Mapping File (LMF)* field.

**Note:** By default the mapping file is saved in the working directory. It is recommended not to use absolute and relative paths while saving the mapping file. Saving the mapping file using relative or absolute paths may lead to error prone state when you next time open the project.

12. Click *Next*.

The Place Part page is displayed. The Place Part page displays the summary of the part and its details.

13. Click *Finish & Place* to place the part on canvas.

After clicking *Finish & Place*, a graphical view of component is displayed on canvas. Left click to drop the component on canvas and right click to disable the graphical view.

### Steps Required for Scenario 3

1. To invoke Component Browser click *Add Part* icon in .  
The Component Browser is displayed.
2. Select a library name in Library pane, whose component you want to bind with the FSP logical model.
3. Select a cell name in *Cells* pane or enter the name of cell in *Cells* text box.
4. In *Search Results* pane, click the row corresponding to the physical part you want to add.

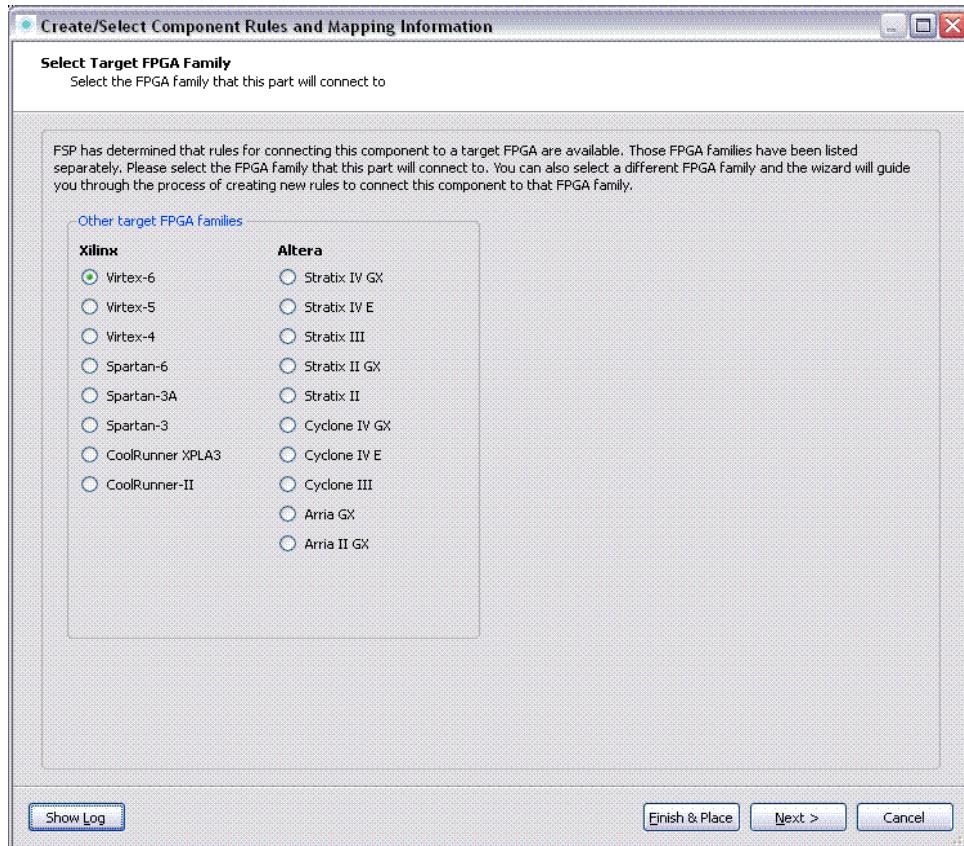
**Note:** For more information on assigning `FSP_LOGICAL_MODEL` property in PTF file see [Setting Up PTF for Component Selection](#) section.

- The symbol and footprint name for the component is displayed in *Part Name* tab.
5. Select a symbol view in the *Symbol* drop-down list.
  6. Click *Select*.

The Select Target FPGA Family page of Create/Select Component Rules and Mapping Information wizard is displayed. Since you do not have a rules file the list of Available target FPGA families is disabled. By default an FPGA family name is selected.

# Allegro FPGA System Planner User Guide

## Working with Components

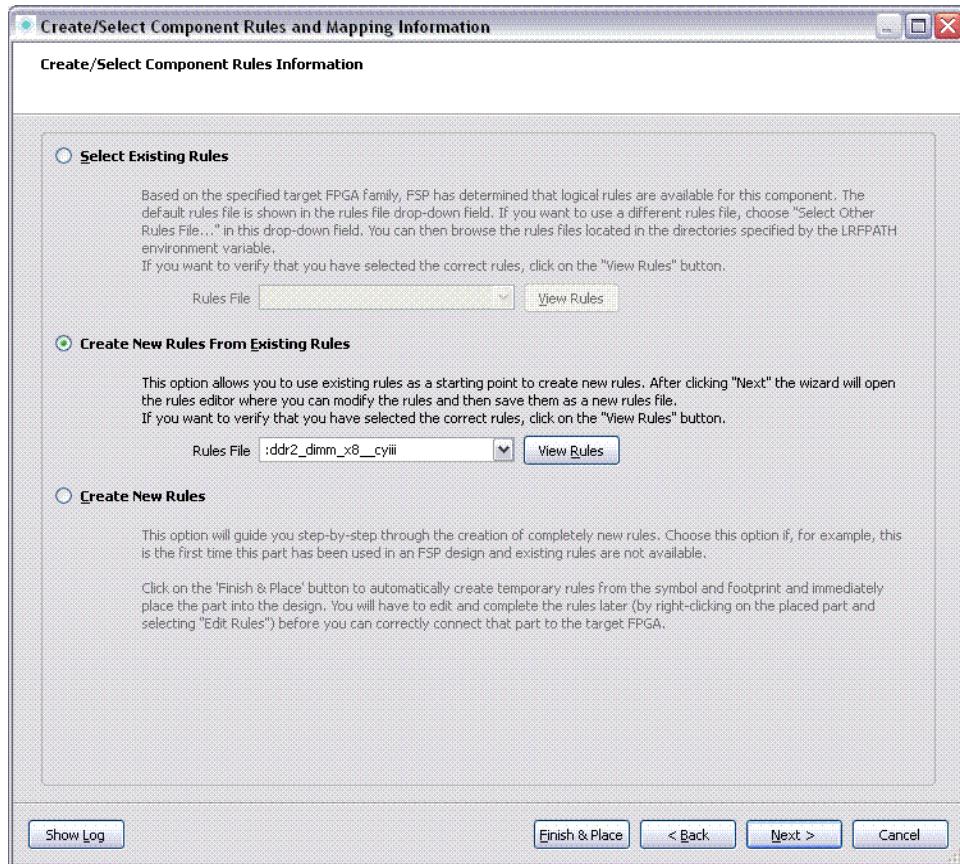


7. Select a FPGA family name to which you want to target the interface.
8. Click *Next* to advance to Create/Select Component Rules Information page.

The Create/Select Component Rules Information page is displayed. By default the Create New Rules From Existing Rules is selected. You are recommended to proceed with this option since you already have mapping file and have to create a new rules file according to it.

# Allegro FPGA System Planner User Guide

## Working with Components



9. Click *Next* to advance to the Rules and Mapping Editor page.

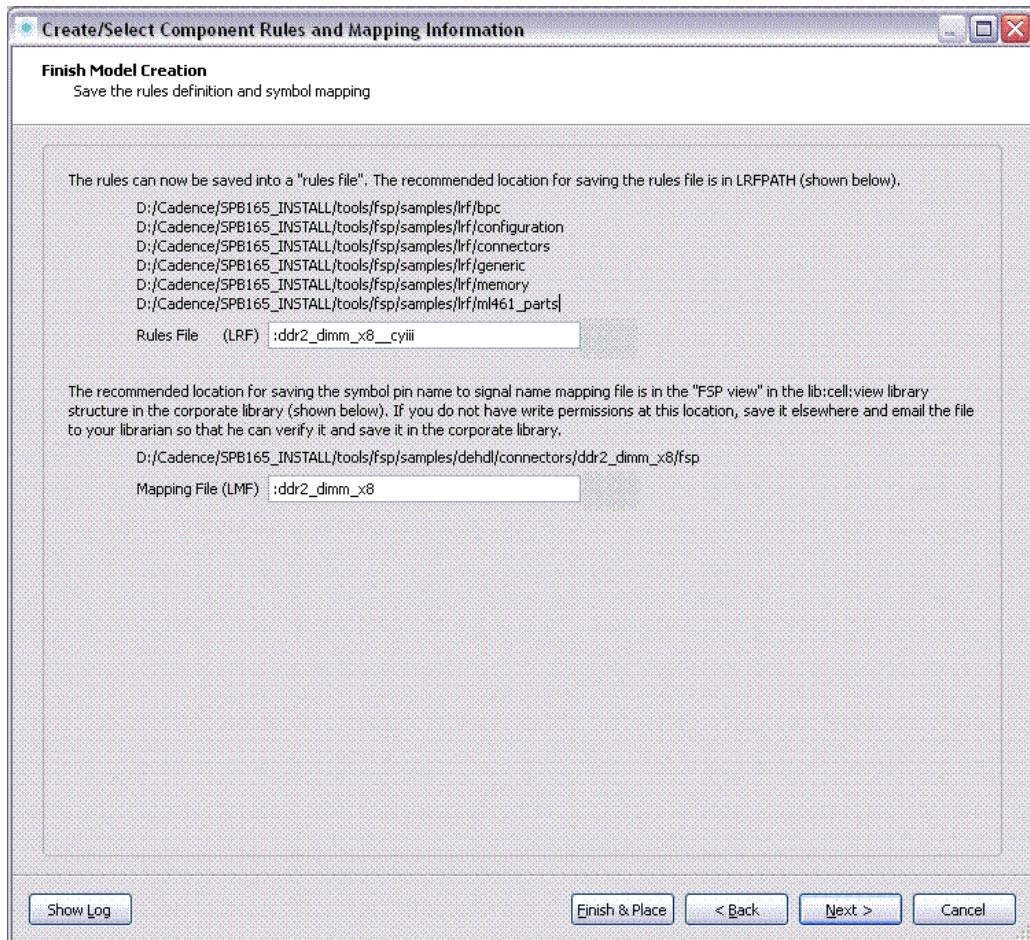
The Rules and Mapping Editor page is displayed.

10. Modify the constraints as required and click *Next*.

The Finish Model Creation page is displayed. By default the rules file name and mapping file name is displayed in their respective fields.

# Allegro FPGA System Planner User Guide

## Working with Components



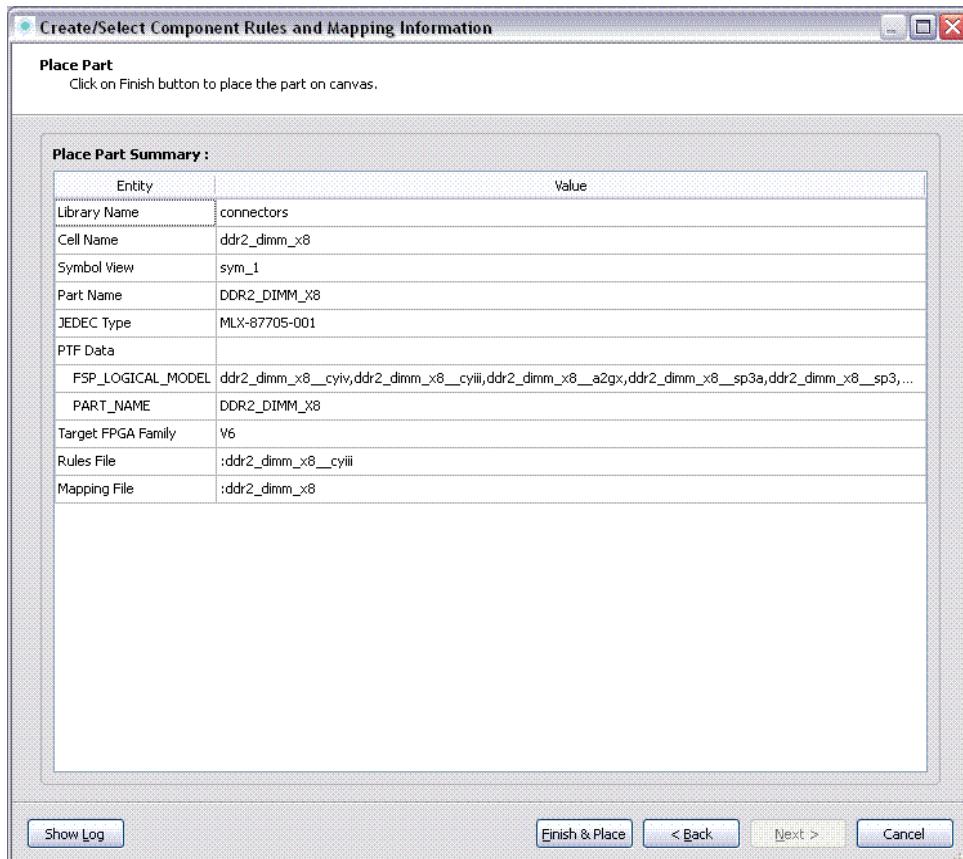
11. Enter the name for the rules file in the *Rules File (LRF)* field.
12. Enter the name for the mapping file in the *Mapping File (LMF)* field.
13. Click *Next* to advance to the Place Part page.

**Note:** When you click *Next*, by default the mapping file is saved in the working directory. It is recommended not to use absolute and relative paths while saving the mapping file. Saving the mapping file using relative or absolute paths may lead to error prone state when you next time open the project.

The Place Part page is displayed.

# Allegro FPGA System Planner User Guide

## Working with Components



14. Click *Finish & Place* to place the component on canvas.

After clicking Finish & Place, a graphical view of component is displayed. Left click to drop the component on canvas and right click to disable the graphical view.

### Steps Required for Scenario 4

1. To invoke Component Browser click *Add Part* icon in .  
The Component Browser is displayed.
2. Select a library name in Library pane, whose component you want to bind with the FSP logical model.
3. Select a cell name in Cells pane or enter the name of cell in Cells text box.
4. In *Search Results* pane, click the row corresponding to the physical part you want to add.

**Note:** For more information on assigning FSP\_LOGICAL\_MODEL property in PTF file see [Setting Up PTF for Component Selection](#) section.

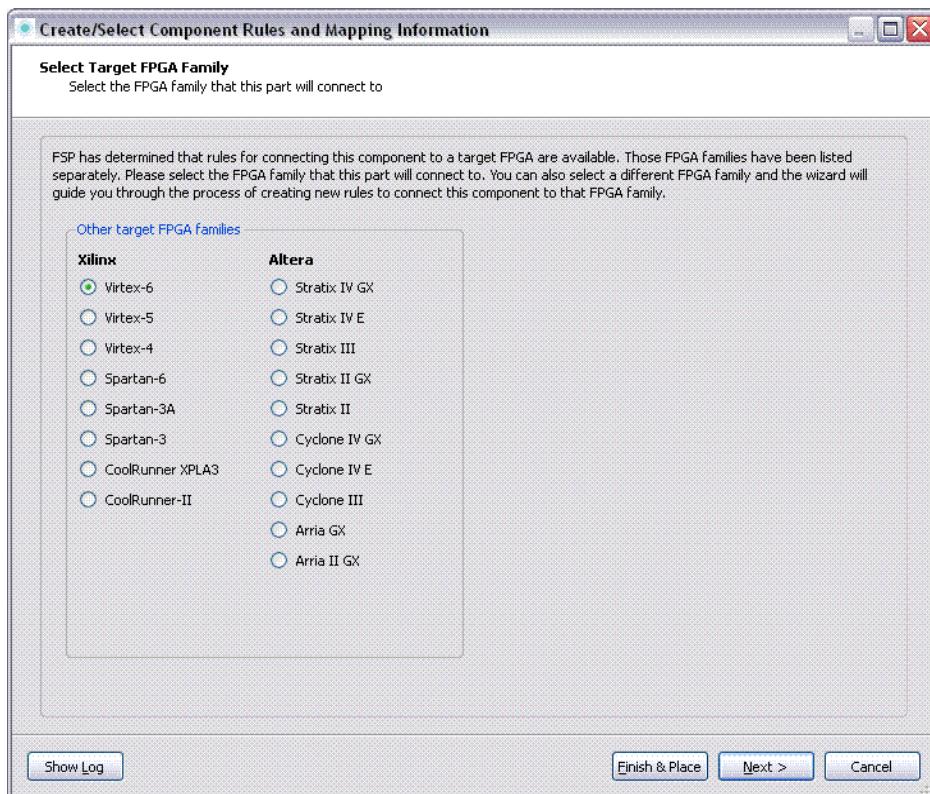
## Allegro FPGA System Planner User Guide

### Working with Components

The symbol and footprint name for the component is displayed in *Part Name* tab.

5. Select a symbol view in the *Symbol* drop-down list.
6. Click *Select*.

The Select Target FPGA Family page of Create/Select Component Rules and Mapping Information wizard is displayed. Since you do not have a rules file the list of Available target FPGA families is disabled. By default an FPGA family name is selected.

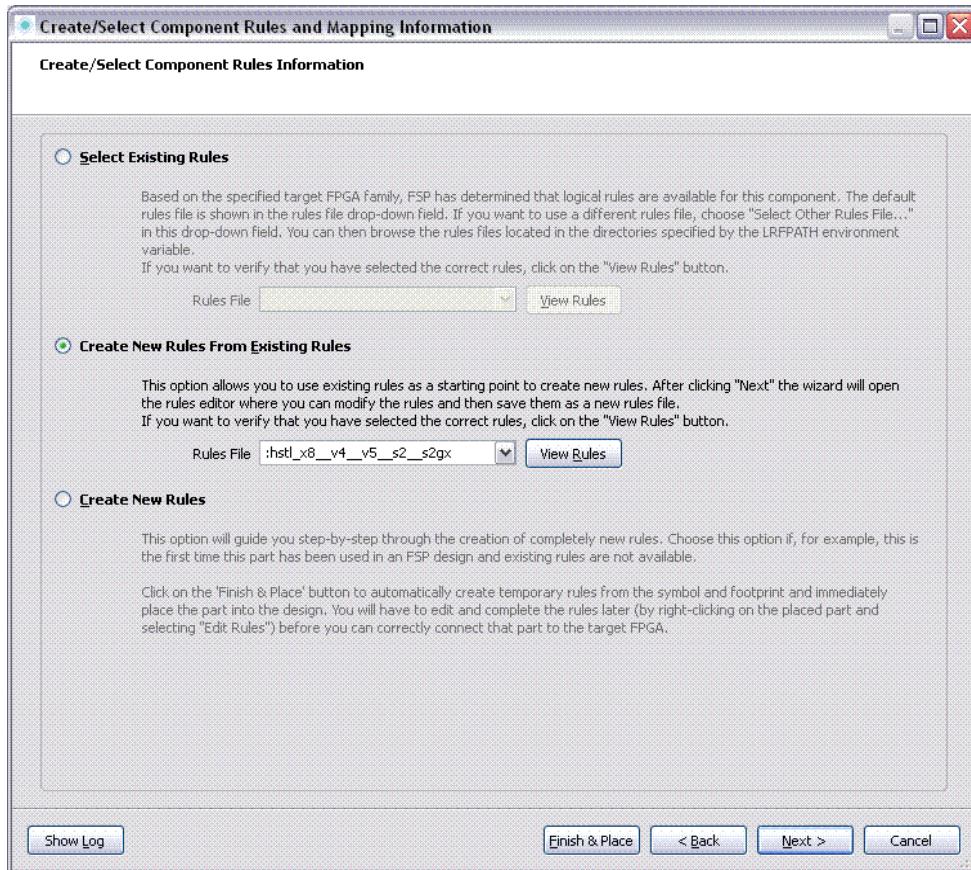


7. Select a FPGA family name to which you want to target the interface.
8. Click *Next* to advance to Create/Select Component Rules Information page.

The Create/Select Component Rules Information page is displayed. By default the Create New Rules option is selected. You are recommended to proceed with this option since you do not have mapping file or rules file.

# Allegro FPGA System Planner User Guide

## Working with Components

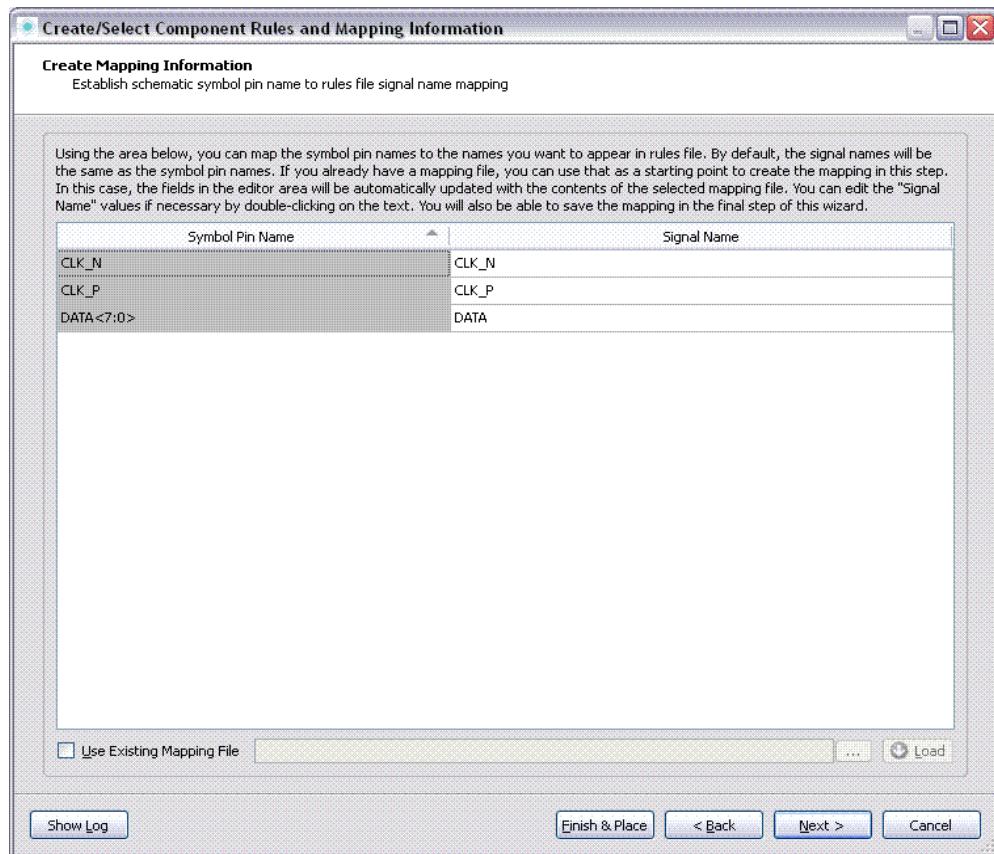


### 9. Click *Next* to advance to the Creating Mapping Information page.

The Create Mapping Information page is displayed. By default the signal names are displayed in Signal Name column.

## Allegro FPGA System Planner User Guide

### Working with Components



**10.** Do any of the following:

- Modify the names manually as required and click *Next*.

Or

- a. Click *Use Existing Mapping File* option.
- b. Click *browse* to select a mapping file.
- c. Click *Load* to import the mapping information.

**11.** Click *Next* to advance to the Specify Voltage page.

The Specify Voltage page is displayed.

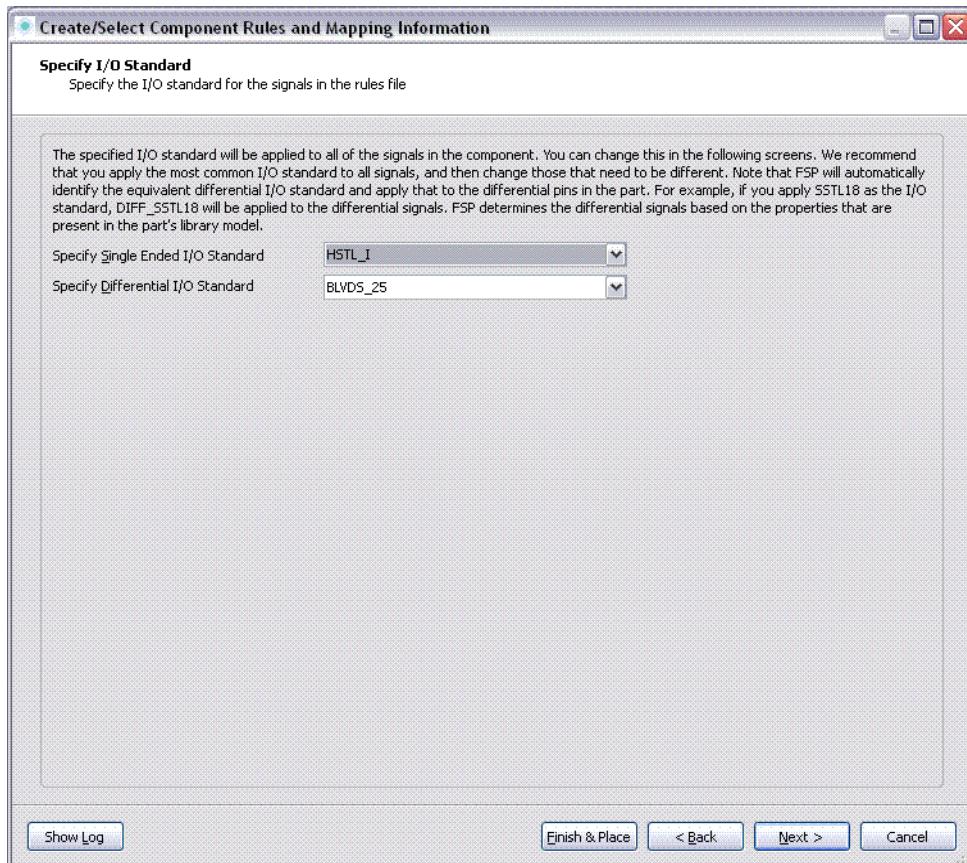
**12.** Specify the voltages levels manually now or later also. This is an optional step.

**13.** Click *Next* to specify the IO standards for the logical model.

The Specify IO Standards is displayed.

## Allegro FPGA System Planner User Guide

### Working with Components



**14.** Do the following:

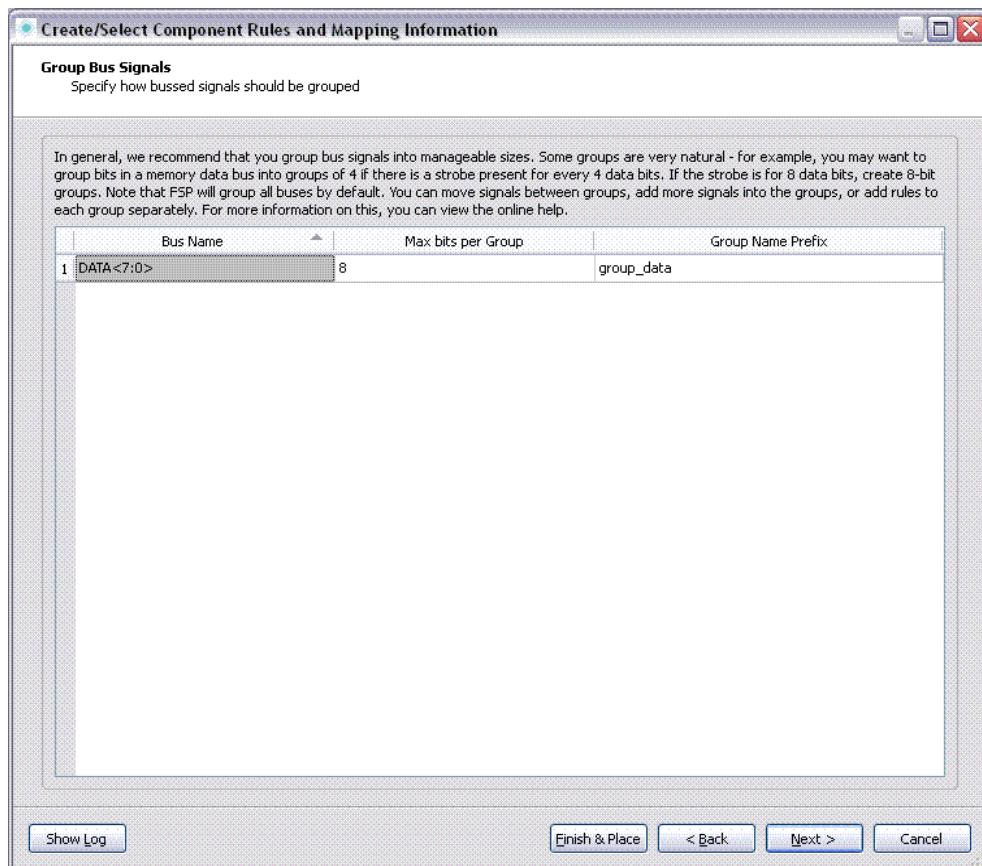
- Click the Specify Single IO standards drop down list and select the IO standards as required.
- Click the Specify Differential IO standards drop down list and select the IO standards as required.

**15.** Click *Next*.

The Group Bus Signals page is displayed.

# Allegro FPGA System Planner User Guide

## Working with Components

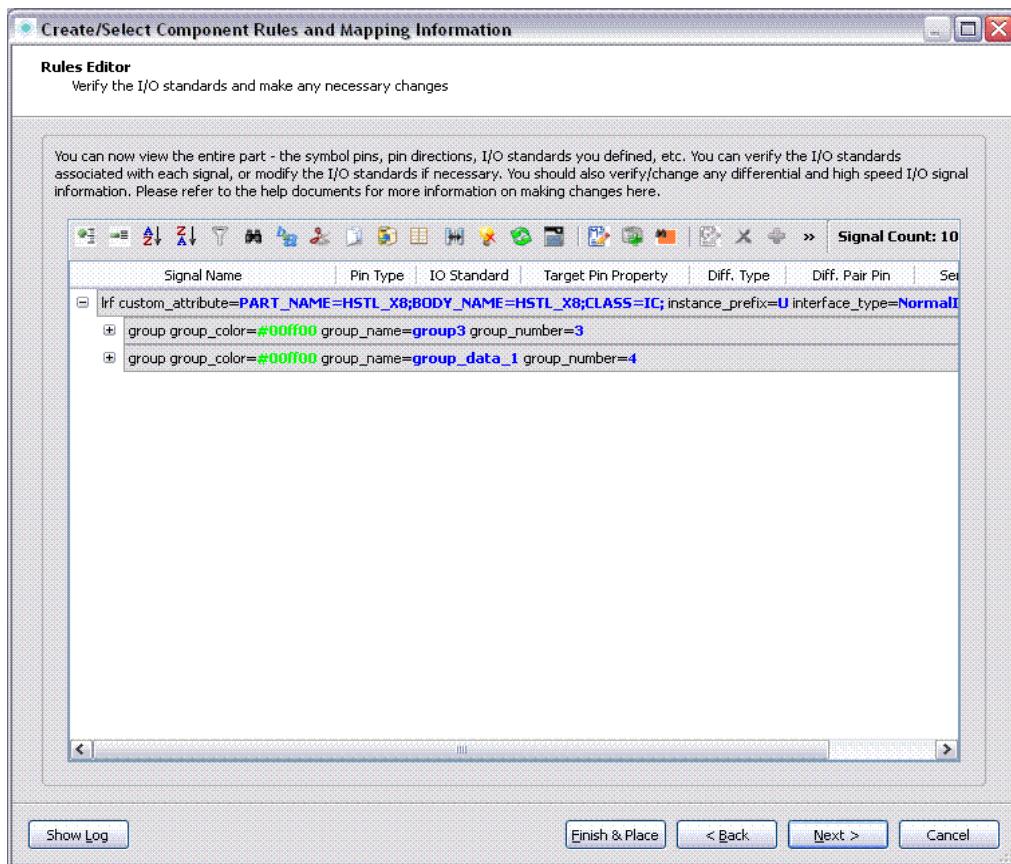


16. Modify the group name prefix as required and click *Next*.

The Rules Editor page is displayed.

# Allegro FPGA System Planner User Guide

## Working with Components

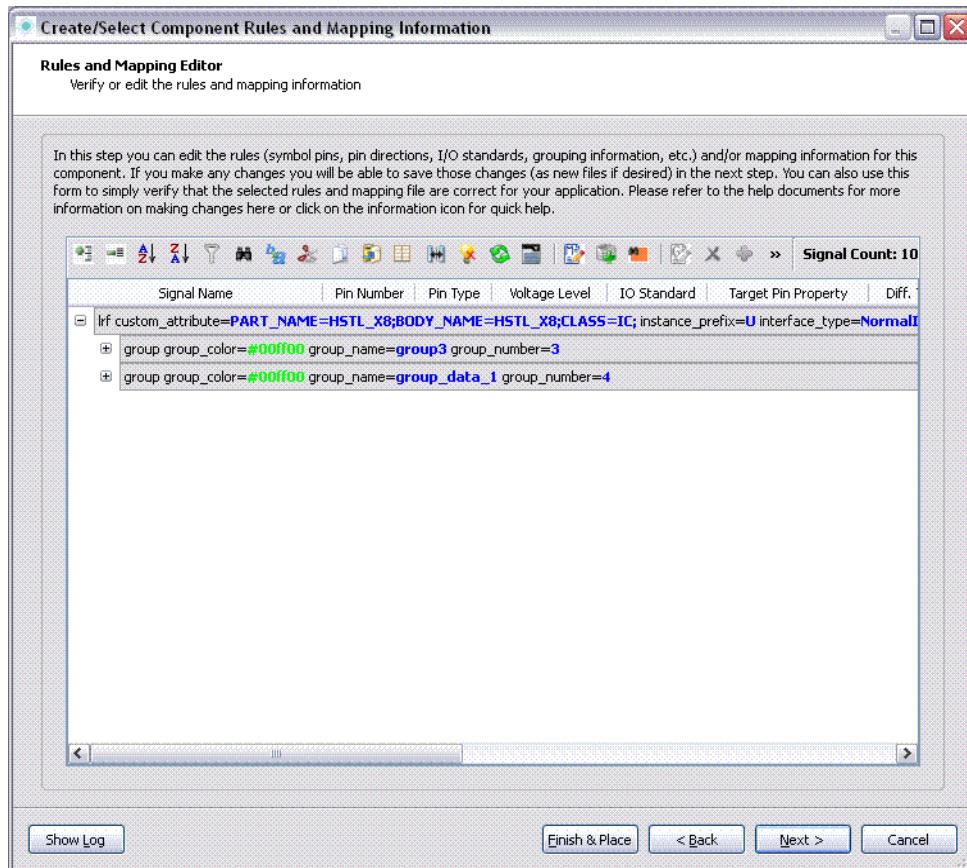


17. Modify the constraints as required and click *Next*.

The Rules and Mapping Editor page is displayed.

# Allegro FPGA System Planner User Guide

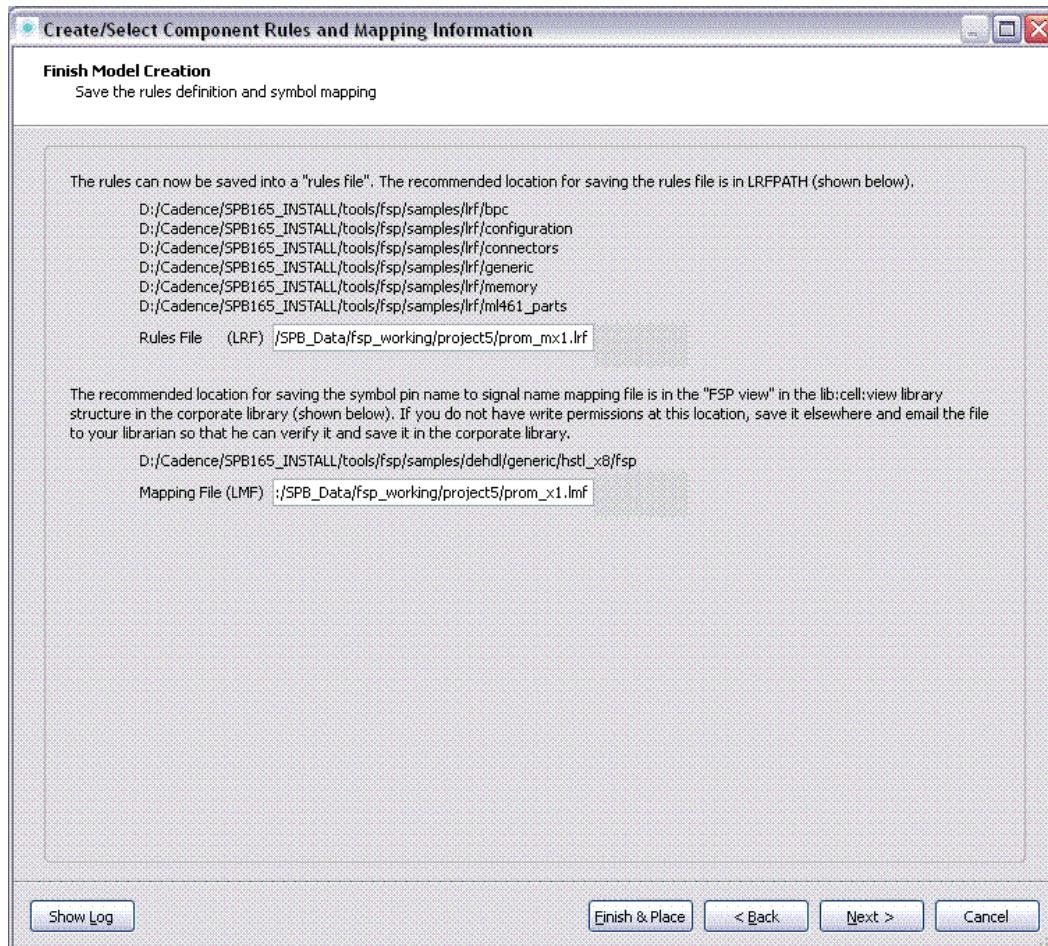
## Working with Components



18. Modify the field values as required and click *Next* to advance to the Finish Model Creation page is displayed.

# Allegro FPGA System Planner User Guide

## Working with Components



19. Enter the name for the rules file in the *Rules File (LRF)* field.
20. Enter the name for the mapping file in the *Mapping File (LMF)* field.
21. Click *Next* to advance to the Place Part page.

**Note:** When you click *Next*, by default the mapping file is saved in the working directory. It is recommended not to use absolute and relative paths while saving the mapping file. Saving the mapping file using relative or absolute paths may lead to error prone state when you next time open the project.

The Place Part page is displayed.

22. Click *Finish & Place* to place the component on canvas.

After clicking *Finish & Place*, a graphical view of component is displayed. Left click to drop the component on canvas and right click to disable the graphical view.

## **Example of Adding Interface Component Using Wizard Flow**

The following example demonstrate the steps to add an interface component on canvas using the wizard flow.

**Note:** The tasks covered in this example are applicable in the DE-HDL schematic environment.

### ***Guidelines***

To ensure that the correct logical rules file, footprint, and symbol file is picked up during component placement, it is recommended that you follow the following guidelines.

- The logical library names with paths are set in the *Rules File Path Editor* dialog box.
- The footprints and padstacks are set in the psmpath and padpath variables, and JEDEC\_TYPE property is set in the ptf file.
- The logical mapping file is placed in a folder fsp in the part library as depicted below:<>>

In this example, you will place a Micron's mt46v32m16 part on canvas. It is assumed that you have set the necessary search path settings as per the guidelines.

The tasks covered in this example are:

- Setting up the FSP\_LOGICAL\_MODEL property in the PTF file for Micron's part mt46v32m16.
- Use Component Browser to browse and select the part mt46v32m16.
- Use the Create>Select Component Rules and Mapping Information wizard for the following tasks:
  - Select targeting family for part mt46v32m16.
  - Browse and select other model of part mt46v32m16 based on the selected device family.
  - Edit rules and mapping file information.
  - Place the part.

### ***Setting up the FSP\_LOGICAL\_MODEL Property***

This section is optional. You can opt this section, when you have a set of `mt46v32m16` files for different FPGAs.

To add the `FSP_LOGICAL_MODEL` property in the PTF file:

1. Create a new folder named `part_table` in the `part mt46v32m16` directory.
2. Open the `part_table` folder and create a new text file named `part.ptf`.
3. Copy the following text in the `part.ptf` file:

```
FILE_TYPE = MULTI_PHYS_TABLE;
PART 'mt46v32m16'
=====
:PART_NAME = FSP_LOGICAL_MODEL ;
=====
'mt46v32m16'(!) =
'mt46v32m16_cyiv,mt46v32m16_cyiii,mt46v32m16_sp6,mt46v32m16_a2gx,mt46v32m
16_sp3_sp3a,mt46v32m16_s3,mt46v32m16_s2_s2gx_agx,mt46v32m16_v4_v5'
END_PART
END.
```

4. Save and close the `part.ptf` file.

### ***Use the Component Browser to Select the Part***

In this section, you will browse and select the part `mt46v32m16` using Component Browser.



To work with the wizard flow, you must select the *Show wizard while placing part on canvas* option in the *Preference* window.

To select the part `mt46v32m16`, using Component Browser:

1. Click the *Component Browser* icon in the .  
The Component Browser is displayed.
2. Select the *Micron* library in the Library pane.
3. Select `mt46v32m16` under the cells pane.
4. Select the physical part table row that appear in the *Search Results* tab.

**Note:** The *Search Results* tab displays the list of model names that you have specified in the part.ptf file.

5. Click *Select* in the Part Details tab.

The *Create>Select Component Rules and Mapping Information* wizard with *Select Target FPGA Family* page is displayed by default.

**Note:** At this stage, you have the choice to place the part without going through the complete wizard flow. Select the device family and click *Finish & Place* to place the part *mt46v32m16* on canvas.

6. Select the *Cyclone IV GX* option and click *Next*.

The Create>Select Component Rules Information page is displayed.

7. In the *Select Existing Rules* text box, the rules file name *mt46v32m16\_cyiv* for the Cyclone IV GX device family is automatically specified.

8. Click *Next* to go to next page.

The *Rules and Mapping Editor* page is displayed.

9. Click *Next* to go to next page.

The *Place Part* page is displayed.

10. Click *Finish & Place* to place the part.

The image of the part *mt46v32m16\_cyiv* is displayed on the canvas.

11. Click anywhere on the canvas.

The part *mt46v32m16\_cyiv* is placed on the canvas.

## Using Add Part dialog box

The Add Part dialog box helps you to place the interface rules file with a little effort.

Following are the four scenarios possible while placing the interface rules file on canvas through Add Part dialog box:

### Scenario   Rules File   Mapping File   Procedure

1	Yes	Yes	Read <u>Steps Required for Scenario 1</u> section.
---	-----	-----	----------------------------------------------------

## Allegro FPGA System Planner User Guide

### Working with Components

---

<b>2</b>	Yes	No	Read <a href="#">Steps Required for Scenario 2</a> section.
<b>3</b>	No	Yes	Read <a href="#">Steps Required for Scenario 3</a> section.
<b>4</b>	No	No	Read <a href="#">Steps Required for Scenario 4</a> section.

**Note:** To invoke the Add Part dialog box unselect the Show wizard while placing part on canvas option in Preference dialog box.

### Steps Required for Scenario 1

To place a component, perform the following steps:

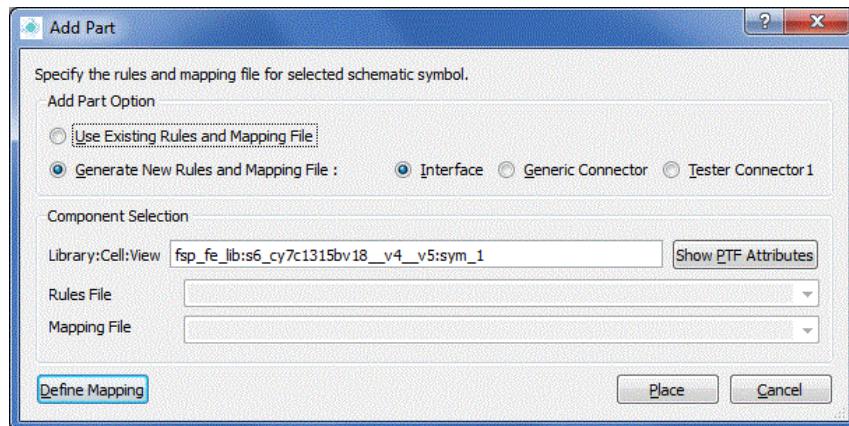
1. To invoke Component Browser click *Add Part* icon in .  
The Component Browser is displayed.
2. Select a library name in Library pane, whose component you want to bind with the FSP logical model.
3. Select a cell name in *Cells* pane or enter the name of cell in *Cells* text box.
4. In *Search Results* pane, click the row corresponding to the physical part you want to add.

**Note:** For more information on assigning `FSP_LOGICAL_MODEL` property in PTF file see [Setting Up PTF for Component Selection](#) section.

The symbol and footprint name for the component is displayed in *Part Name* tab.

5. Select a symbol view from *Symbol* drop down list.
6. Click *Select*.

The Add Part dialog box is displayed. Since you have rules file and mapping file by default the rules file name and mapping file name is displayed in their respective fields.



7. Click *Place* to place the interface component on canvas.

## Steps Required for Scenario 2

To place a component perform the following steps:

1. To invoke Component Browser click *Add Part* icon in .  
The Component Browser is displayed.
2. Select a library name in Library pane, whose component you want to bind with the FSP logical model.
3. Select a cell name in *Cells* pane or enter the name of cell in *Cells* text box.
4. In *Search Results* pane, click the row corresponding to the physical part you want to add.

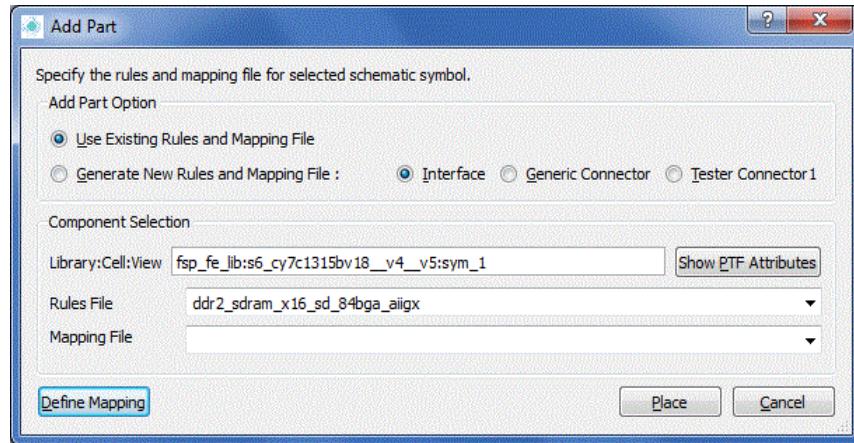
**Note:** For more information on assigning `FSP_LOGICAL_MODEL` property in PTF file see [Setting Up PTF for Component Selection](#) section.

- The symbol and footprint name for the component is displayed in *Part Name* tab.
5. Select a symbol view from *Symbol* drop down list.
  6. Click *Select*.

The Add Part dialog box is displayed. Since you have rules file by default the rules file name is displayed in Rules File field.

## Allegro FPGA System Planner User Guide

### Working with Components



#### 7. Do any of the following:

- Click the Mapping File drop down list and select a mapping file name.

Or

- Select *Select Other Mapping Rules File* option.

The Select Mapping File windows dialog box is displayed.

- Select the mapping file and click *Open*.
- Click *Place* to place the interface component on canvas.

#### 8. Click *Define Mapping* to create new mapping file.

For more information on creating mapping files see [Creating Mapping File](#) section.

#### 9. Click *Place* to place the interface component on canvas.

### Steps Required for Scenario 3

To place a component perform the following steps:

1. To invoke Component Browser click *Add Part* icon in .  
The Component Browser is displayed.
2. Select a library name in Library pane, whose component you want to bind with the FSP logical model.
3. Select a cell name in *Cells* pane or enter the name of cell in *Cells* text box.
4. In *Search Results* pane, click the row corresponding to the physical part you want to add.

## Allegro FPGA System Planner User Guide

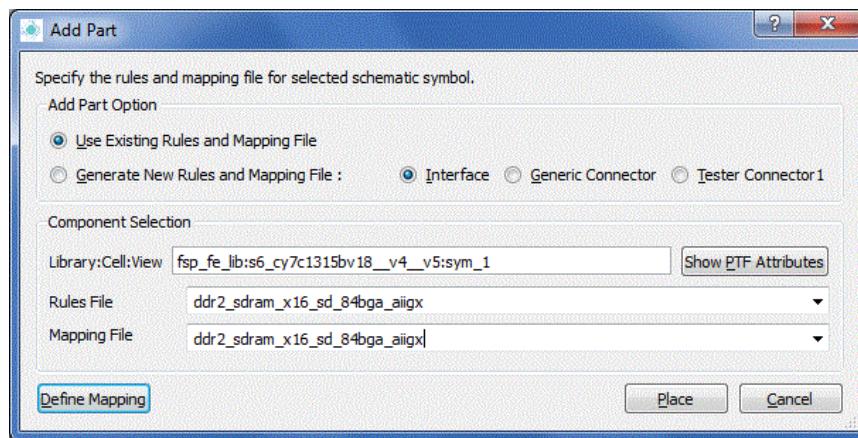
### Working with Components

**Note:** For more information on assigning FSP\_LOGICAL\_MODEL property in PTF file see [Setting Up PTF for Component Selection](#) section.

The symbol and footprint name for the component is displayed in *Part Name* tab.

5. Select a symbol view from *Symbol* drop down list.
6. Click *Select*.

The Add Part dialog box is displayed.



7. Click *Place* to place the interface component on canvas.

On clicking Place, the rules file creation and mapping of new rules file to selected symbol is automatically done by FSP.

### Steps Required for Scenario 4

To place a component perform the following steps:

1. To invoke Component Browser click *Add Part* icon in .  
The Component Browser is displayed.
2. Select a library name in Library pane, whose component you want to bind with the FSP logical model.
3. Select a cell name in Cells pane or enter the name of cell in Cells text box.
4. In *Search Results* pane, click the row corresponding to the physical part you want to add.

**Note:** For more information on assigning FSP\_LOGICAL\_MODEL property in PTF file see [Setting Up PTF for Component Selection](#) section.

## Allegro FPGA System Planner User Guide

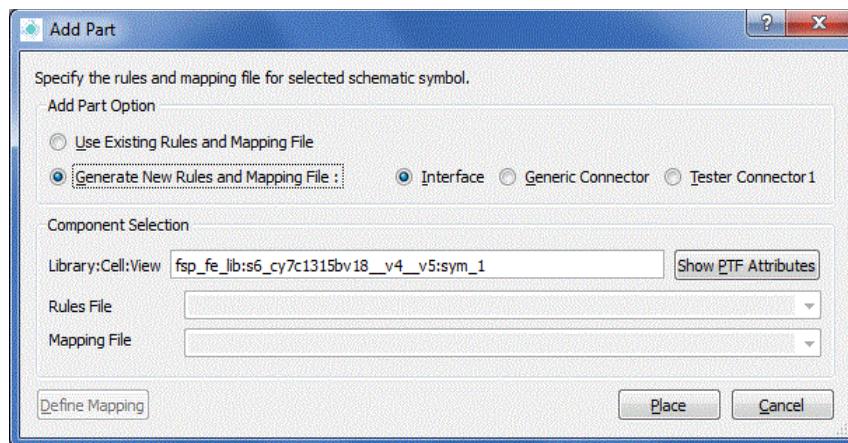
### Working with Components

The symbol and footprint name for the component is displayed in *Part Name* tab.

5. Select a symbol view from *Symbol* drop down list.

6. Click *Select*.

The Add Part dialog box is displayed. Since you do not have rules file and mapping file both the Rules File and Mapping File field appears blank.



7. Click *Place* to autogenerate rules file and mapping file in your project directory.

On clicking *OK*, a confirmation dialog box pop-ups prompting you about generation of temporary files in the project directory.

8. Click *Yes* to generate the rules file and mapping file or click *No* to generate the connector.

On clicking Yes, a successful message about creation of rules file and mapping file is displayed in log window. Also a graphical view of the component is displayed. Left-click to drop the component on canvas and right-click to disable the graphical view.

### Example of Placing an Interface Component Using Add Part

The following example demonstrates the steps to place an interface component on the canvas using the Add Part dialog box.

**Note:** The tasks covered in this example are applicable in the DE-HDL schematic environment.

## **Guidelines**

Before you place an instance on canvas, you must ensure that the rules and mapping files search paths settings are set according to the guidelines. For more information, refer to the guidelines covered in the [Guidelines](#) subsection of *Example of Placing Interface Component Using Wizard* section.

## **Setting up the FSP\_LOGICAL\_MODEL Property**

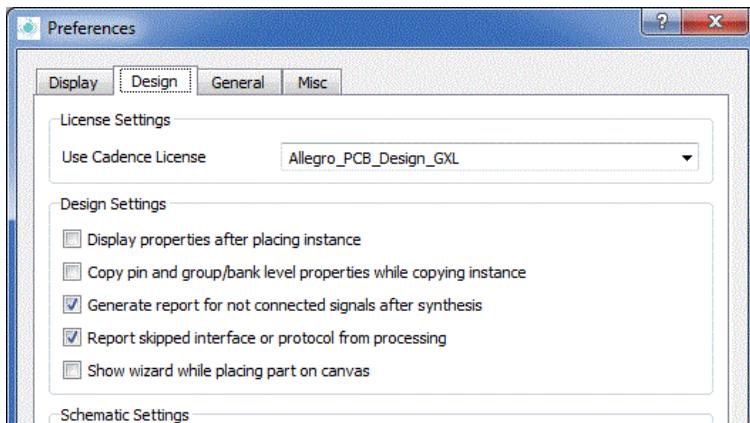
To set the FSP\_LOGICAL\_MODEL property, refer to the steps covered in the [Setting up the FSP\\_LOGICAL\\_MODEL Property](#) subsection of *Example of Placing Interface Component Using Wizard* section.

In this example, you will place the Micron's part mt47h128m8\_\_cyiii on canvas. It is assumed that you do not have a mapping file for the part mt47h128m8\_\_cyiii and all the necessary search path settings have been set according to the guidelines.

To place the part mt47h128m8\_\_cyiii:

1. Choose *File – Preferences*.

The Preference dialog box opens.



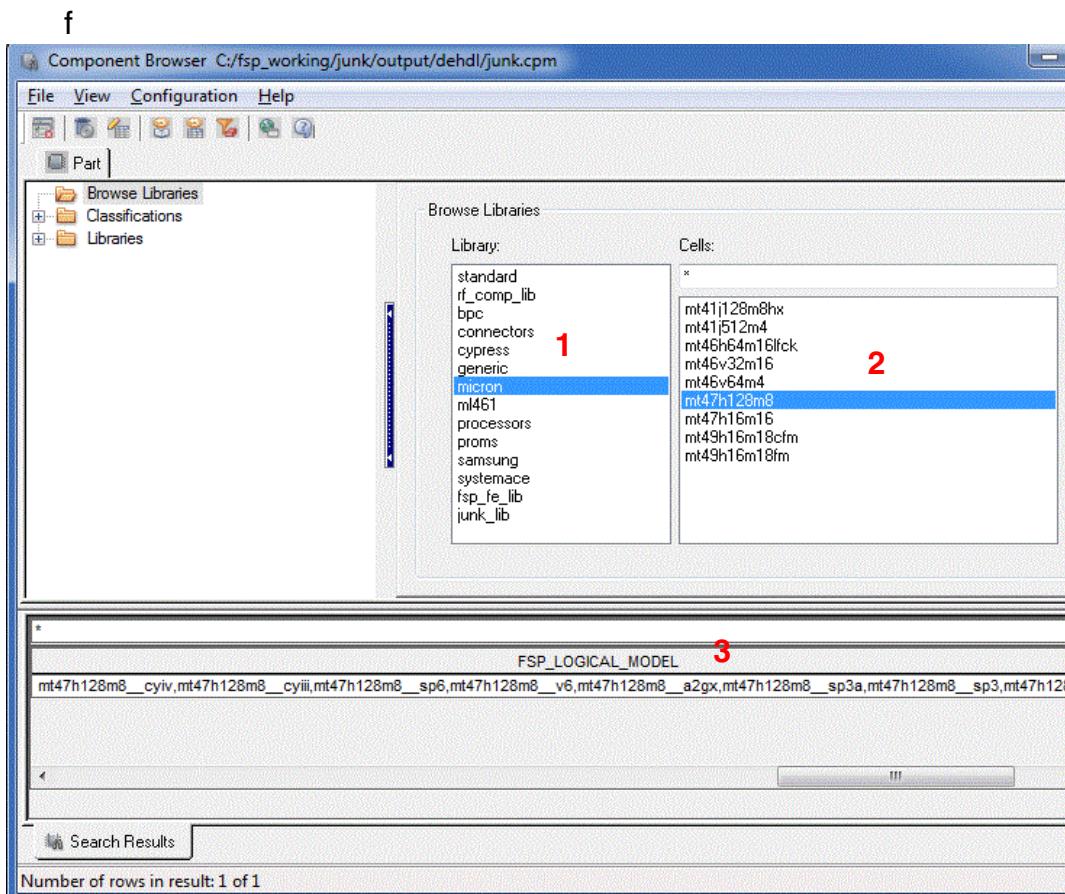
2. Uncheck the *Show wizard while placing part on canvas* option and click *OK*.
3. Click the *Component Browser* icon in the .



# Allegro FPGA System Planner User Guide

## Working with Components

The Component Browser opens.



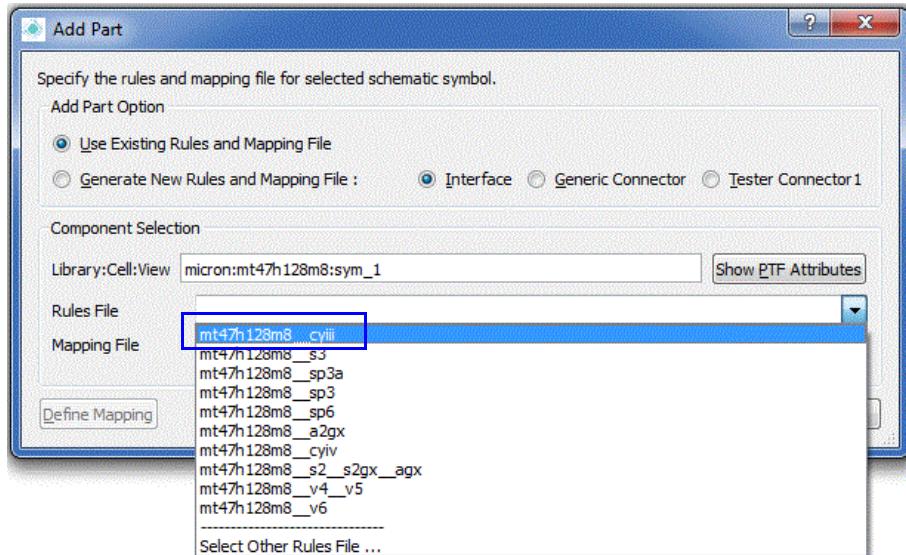
4. Click *Select* in the Part Details tab.

The Add Part dialog box appears with the *Use Existing Rules and Mapping Files* option selected.

5. Select *mt47h128m8\_cyiii* from the drop-down list.

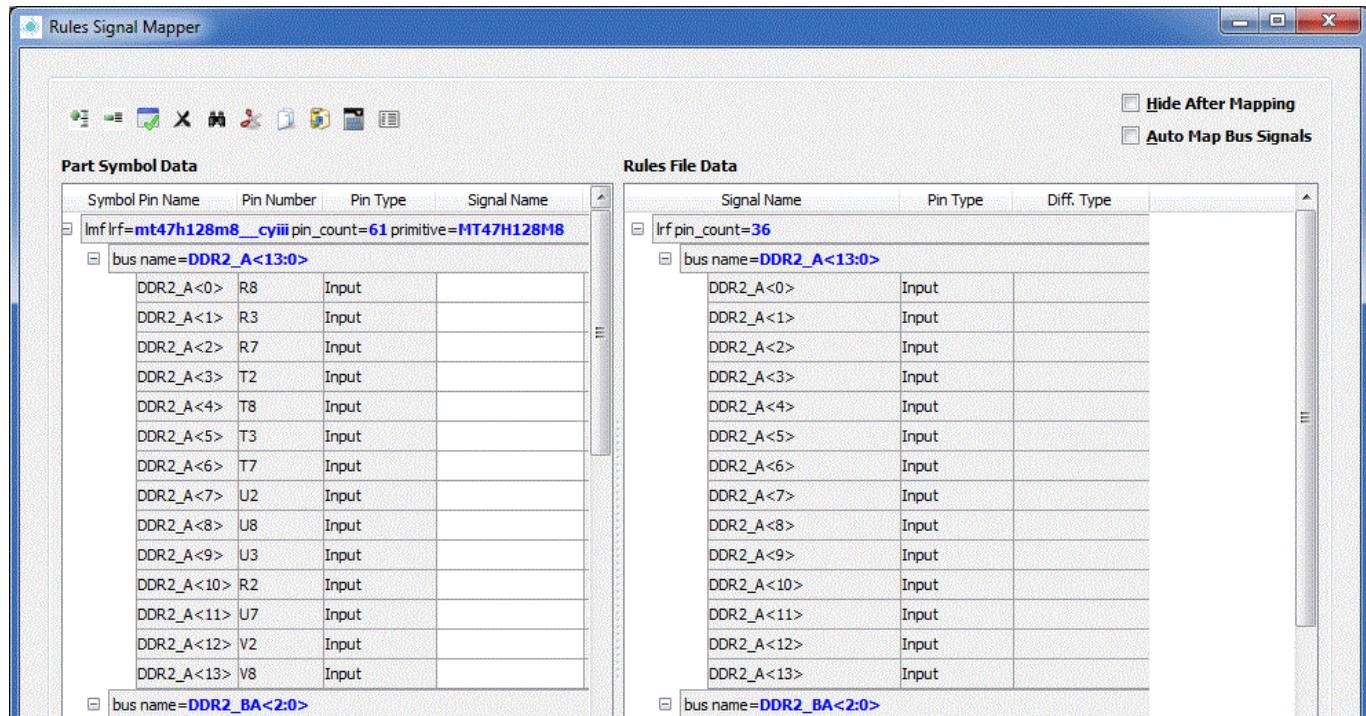
# Allegro FPGA System Planner User Guide

## Working with Components



### 6. Click *Define Mapping*.

The *Rules Signal Mapper* form appears.

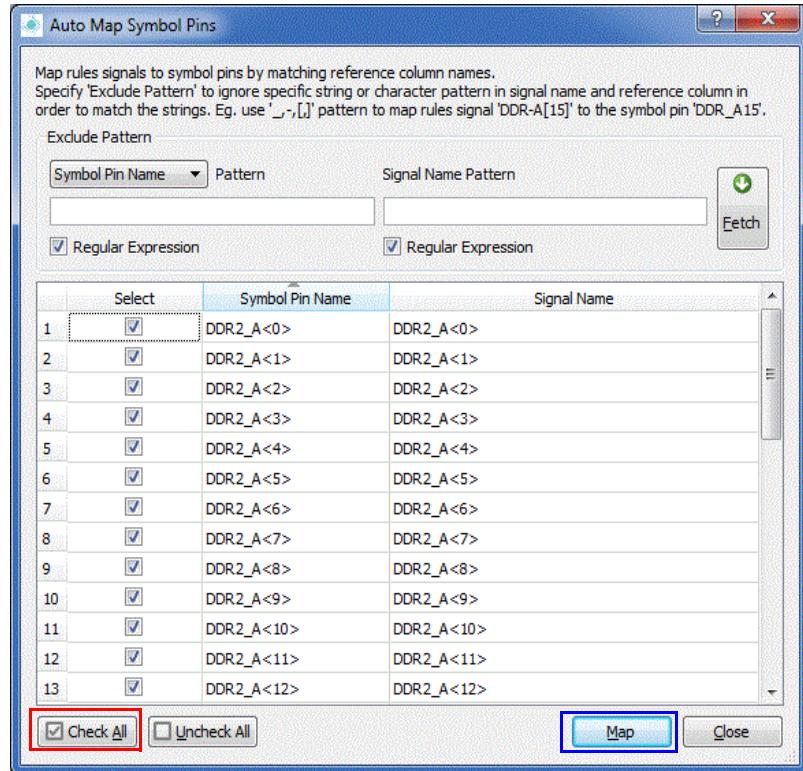


### 7. Click the *Auto Map Signals* icon.

## Allegro FPGA System Planner User Guide

### Working with Components

The Auto Map Symbol Pins dialog box opens.



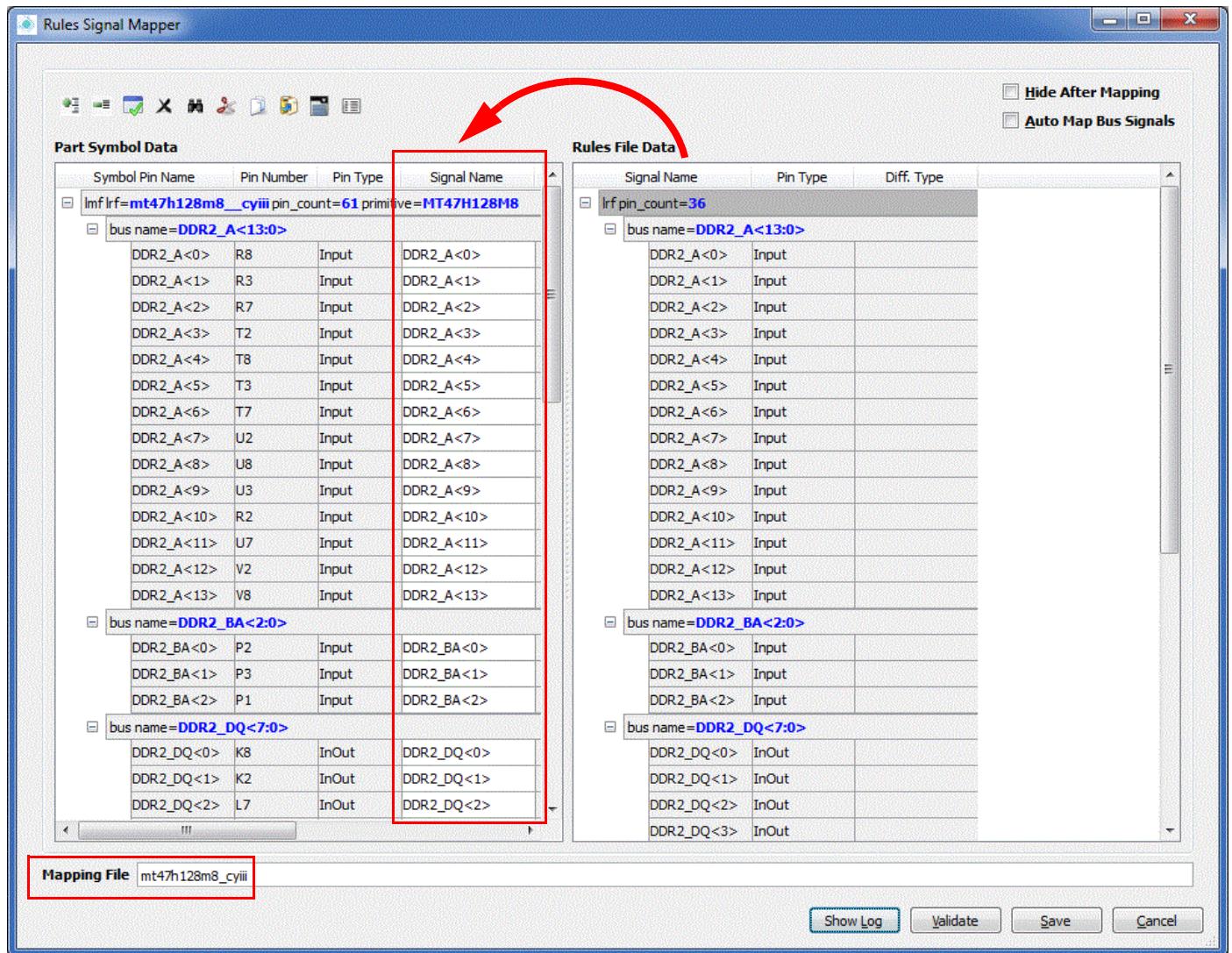
8. Click *Check All* to select all the names for mapping.

9. Click *Map*.

The signal names are mapped to the symbol pin names.

# Allegro FPGA System Planner User Guide

## Working with Components



10. Type *mt47h128m8\_cyiii* in the *Mapping File* field.

11. Click *Save*.

12. Click *Place* in the Add Part dialog box.

The image of the part appears on the canvas.

13. Click anywhere on the canvas to place the part.

The part *mt47h128m8\_cyiii* is placed on the canvas.

## Adding Interface Component (OrCAD)

In OrCAD schematic environment, you will be walked through a wizard called Create>Select Component Rules and Mapping Information. This wizard will guide through a series of steps and tasks that are required to link the selected symbol to interface rules file and place it on canvas. There is another option to place the interface rules file on canvas in order to avoid the wizard steps called Add Part dialog box. To use the Add Part dialog box first you must hide the wizard by selecting *Show wizard while placing part on canvas* option in Preferences window. After selecting this option, the Add Part dialog box is invoked in place of wizard. The Add Part dialog box allows you to quickly select the rules file and mapping file and place it on canvas.



The process of interface rules file selection and placement in both the schematic environments are similar except the symbol selection task. In DE-HDL environment the Component Browser is used as an interface for symbol selection where as in OrCAD environment the Select the OrCAD Symbol page of the Create>Select Component Rules and Mapping Information wizard is used for symbol selection.

FSP lets you bind the OrCAD symbols to interface rules file at the time of instantiation. You can choose any of the following methods to bind the symbols with logical models and place the interface rules file:

- [Using Create>Select Component Rules and Mapping Information Wizard](#)
- [Using Add Part dialog box](#)

### Using Create>Select Component Rules and Mapping Information Wizard

Following are the four scenarios which are possible while placing the interface rules file on canvas:

Scenario	Rules File	Mapping File	Procedure
1	Yes	Yes	Read <a href="#"><u>Procedure for Scenario1</u></a> section.
2	Yes	No	Read <a href="#"><u>Procedure for Scenario2</u></a> section.
3	No	Yes	Read <a href="#"><u>Procedure for Scenario3</u></a> section.

## Allegro FPGA System Planner User Guide

### Working with Components

---

4	No	No	Read <a href="#">Procedure for Scenario4</a> section.
---	----	----	-------------------------------------------------------

#### Note:

- To enable the wizard invoke select *Show Wizard while placing part on canvas* option in Preference Window.
- In every page of the wizard Finish & Place option is common. This option helps you to place the component on canvas at any time. If you have both mapping file and rules file, you can use the Finish & Place option at any time. Once you click the option the interface rules file and its associated mapping file is automatically detected.



The procedure for placing an interface rules file and the above four possible scenarios are very similar to DE-HDL environment steps. However there is only one dissimilarity in between both the procedure is selection of symbols. In DE-HDL environment the Component Browser is used as an interface for symbol selection where as in OrCAD environment the Select the OrCAD Symbol page of the Create/Select Component Rules and Mapping Information wizard is used for symbol selection. The below section describes only the steps required for OrCAD symbol selection. For remaining procedure it is suggested that you go through the [Adding Interface Component](#) section.

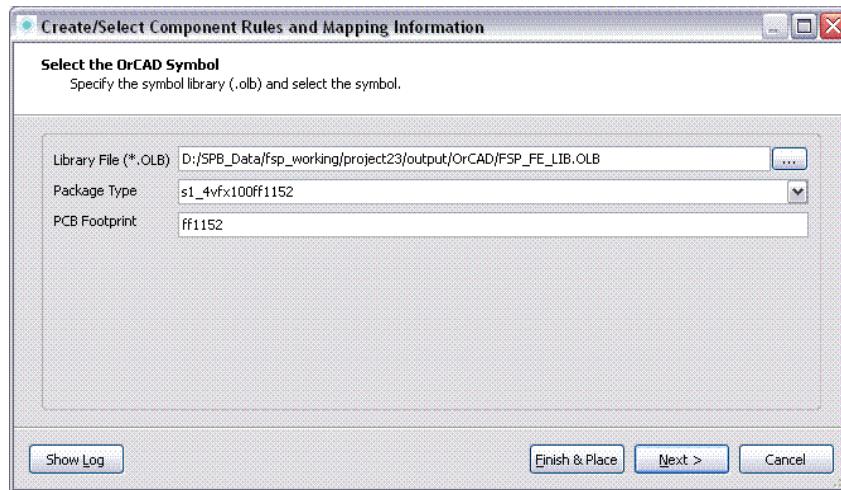
#### Procedure for Scenario1

1. To invoke the Create>Select Component Rules and Mapping Information wizard click *Add Part* icon in .

The Select OrCAD Symbol is displayed.

## Allegro FPGA System Planner User Guide

### Working with Components



2. Do one of the following:
  - In Library File (\*.olb) field specify the olb file name along with the path.
  - Click *browse (...)* to browse to the location where the symbol file exists.
3. Click the Package Type drop down list and select the package name as required.  
The footprint name is automatically displayed in PCB Footprint field.
4. Click *Next* to advance to Select Target FPGA Family page.

For remaining steps see [Steps Required for Scenario 1](#) of DE-HDL section.

### Procedure for Scenario2

1. To invoke the Create>Select Component Rules and Mapping Information wizard click *Add Part* icon in .  
The Select OrCAD Symbol is displayed.
2. Do one of the following:
  - In Library File (\*.olb) field specify the olb file name along with the path.
  - Click *browse (...)* to browse to the location where the symbol file exists.
3. Click the Package Type drop down list and select the package name as required.  
The footprint name is automatically displayed in PCB Footprint field.
4. Click *Next* to advance to Select Target FPGA Family page.

## **Allegro FPGA System Planner User Guide**

### Working with Components

---

For remaining steps see [Steps Required for Scenario 2](#) of DE-HDL section.

#### **Procedure for Scenario3**

1. To invoke the Create>Select Component Rules and Mapping Information wizard click *Add Part* icon in .
2. The Select OrCAD Symbol is displayed.
3. Do one of the following:
  - In Library File (\*.olb) field specify the olb file name along with the path.
  - Click *browse (...)* to browse to the location where the symbol file exists.
4. Click the Package Type drop down list and select the package name as required.  
The footprint name is automatically displayed in PCB Footprint field.
5. Click *Next* to advance to Select Target FPGA Family page.

For remaining steps see [Steps Required for Scenario 3](#) of DE-HDL section.

#### **Procedure for Scenario4**

1. To invoke the Create>Select Component Rules and Mapping Information wizard click *Add Part* icon in .
2. The Select OrCAD Symbol is displayed.
3. Do one of the following:
  - In Library File (\*.olb) field specify the olb file name along with the path.
  - Click *browse (...)* to browse to the location where the symbol file exists.
4. Click the Package Type drop-down list and select the package name as required.  
The footprint name is automatically displayed in PCB Footprint field.
5. Click *Next* to advance to Select Target FPGA Family page.

For remaining steps see [Steps Required for Scenario 4](#) section.

## Example of Placing an Interface Component on the Canvas

The following example demonstrates the steps to add an interface component on the canvas using wizard.

**Note:** The tasks covered in this example are applicable in the OrCAD schematic environment.

### **Guidelines**

To ensure that the correct rules file, footprint, and symbol file is picked up during component placement, it is recommended that you follow the following guidelines:

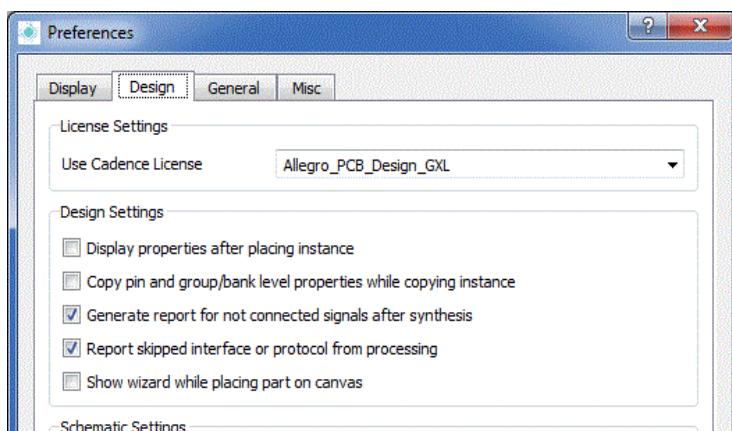
- The logical library name and path are set in the *Rules File Path Editor* dialog box.
- The footprint and padstacks are set in the psmpath and padpath variable of Allegro User Preference Env file located at \$HOME/pcbenv/env .
- The logical mapping file is placed in the same directory where the part symbol (.olb) exists.

In this example you will place a udimm\_184pin\_72b\_ddr\_x8\_v4 part on canvas. It is assumed that you have set the necessary settings as per the guidelines.

To place the udimm\_184pin\_72b\_ddr\_x8\_v4 part on canvas:

1. Choose *File – Preferences*.

The Preference dialog box is displayed.



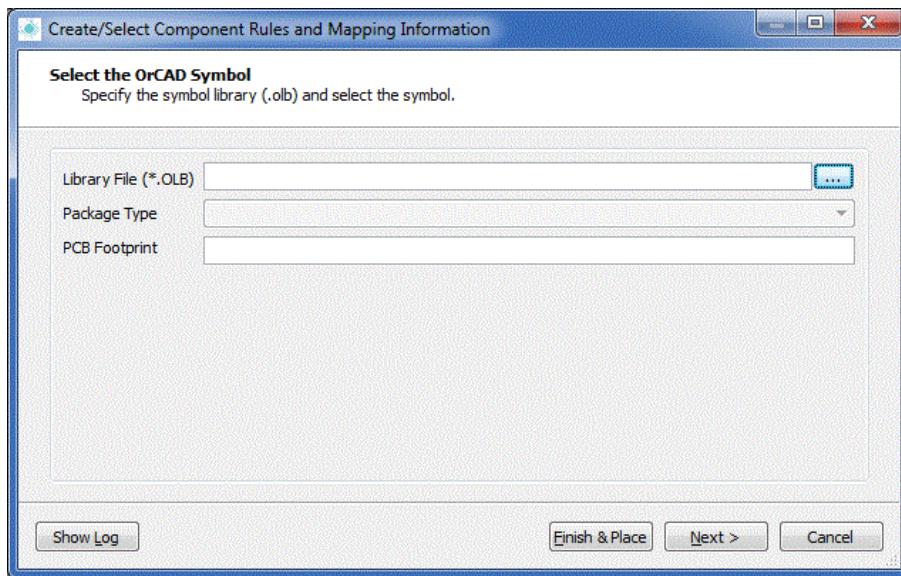
2. Select the *Show wizard while placing part on canvas* option and click *OK*.
3. Click the *Component Browser* icon.

## Allegro FPGA System Planner User Guide

### Working with Components

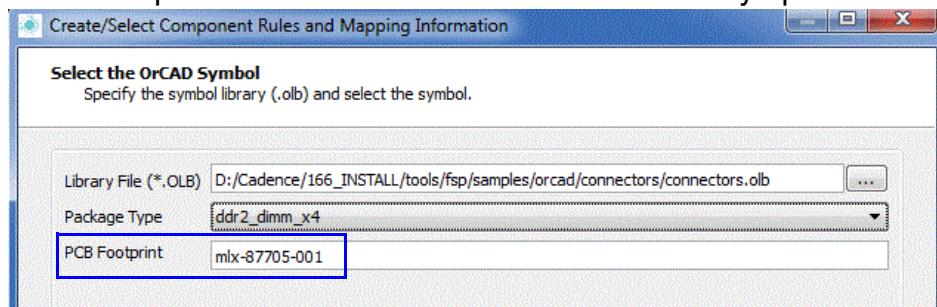


The *Create/ Select Component Rules and Mapping Information* wizard is displayed.



4. Click *browse (...)*.
5. Browse to the %cdsroot%\tools\fsp\samples\orcad\connectors and select the connector.olb file.
6. Select the ddr2\_dimm\_x4 option from the *Package Type* drop-down list.

The footprint name mlx-87705-001 is automatically updated.



**Note:** At this stage, you have the choice to place the part ddr2\_dimm\_x4 on the canvas without going through the complete wizard flow. Click *Finish & Place* to place the part on the canvas.

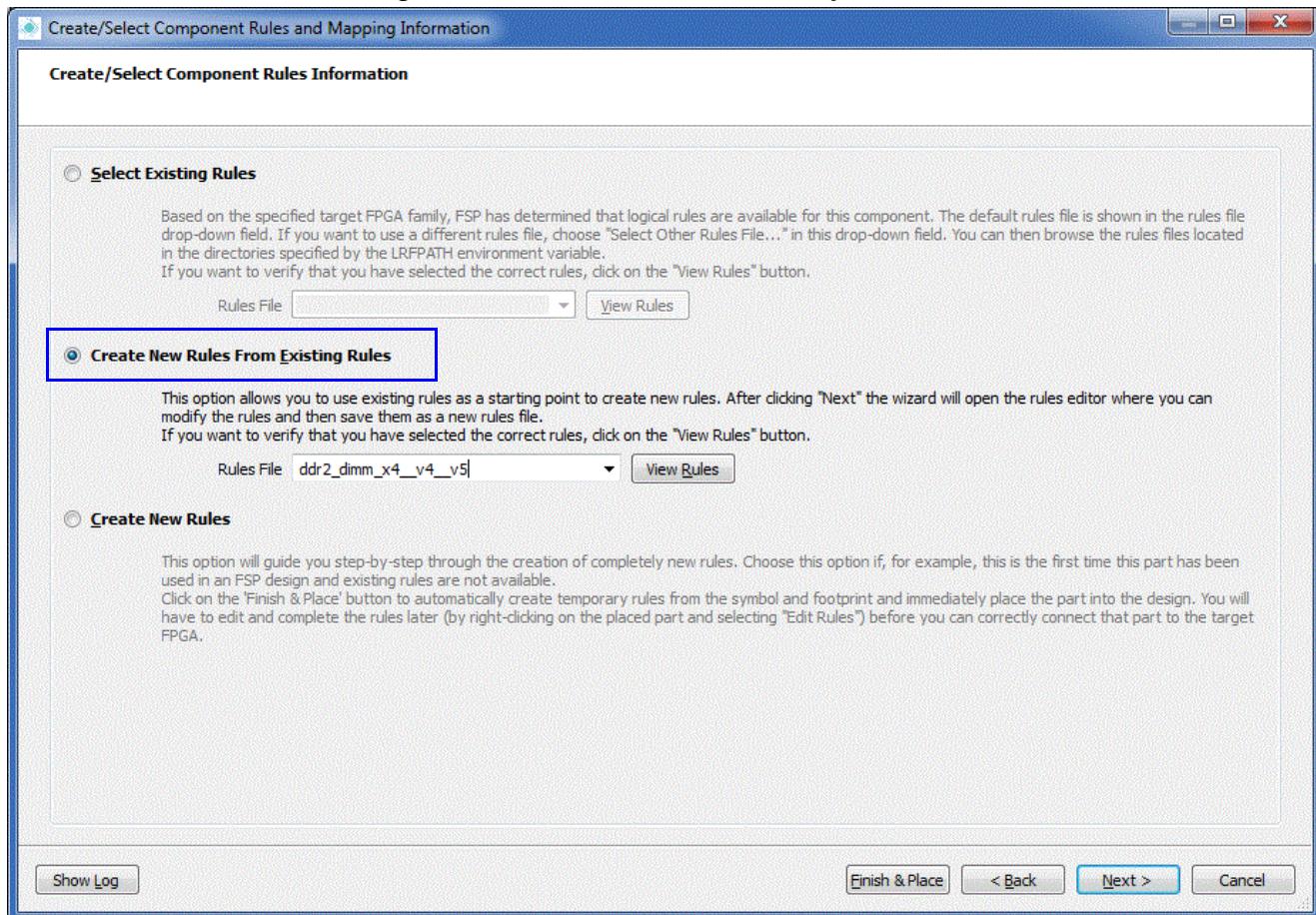
## Allegro FPGA System Planner User Guide

### Working with Components

7. Click *Next* to go to the next page.

8. In the *Select Target Family* page, select the *Virtex-7* option and click *Next*.

The *Create New Rules From Existing Rules* option is selected by default, since you do not have a *ddr\_dimm* logical rules file for Virtex-7 family device.



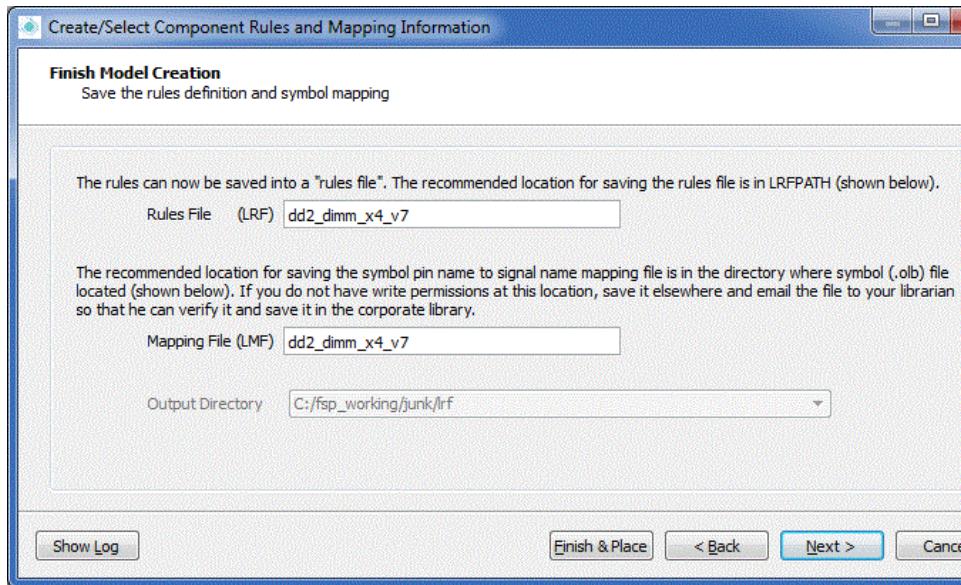
9. Click *Next* to go to the next page.

The Rules and Mapping Editor is displayed.

10. Click *Next*.

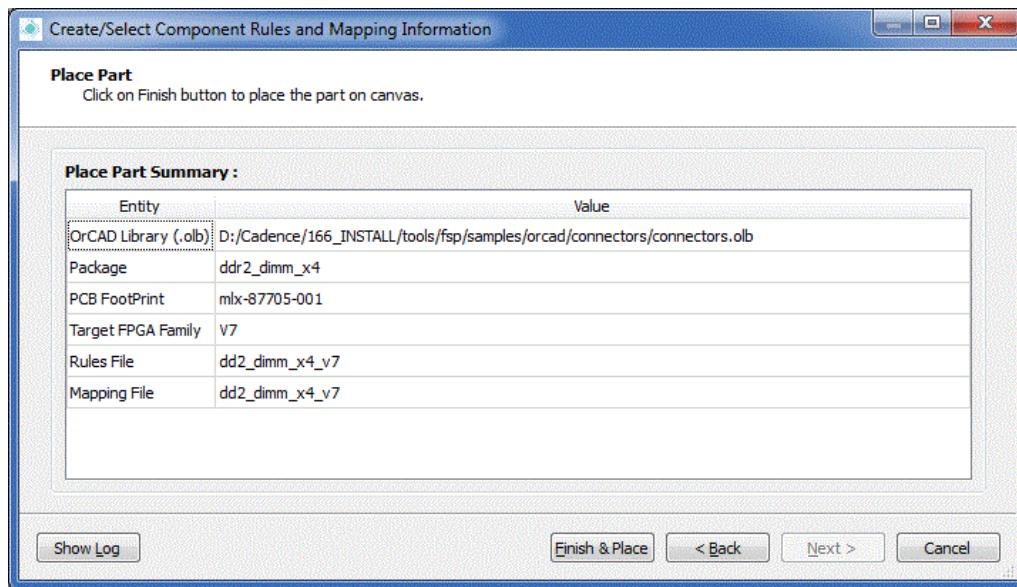
# Allegro FPGA System Planner User Guide

## Working with Components



11. Type ddr2\_dimm\_x4\_v7 in *Rules File* and *Mapping File* text boxes.
12. Click *Next*.

The Place Part page appears with a summary of part details.



13. Click *Finish & Place* to place the part.  
The image of the part is displayed on the canvas.
14. Click anywhere on the canvas to place the part.

The part ddr2\_dimm\_x4\_v7 is placed on the canvas.

## Using Add Part dialog box

To place a component you also use the Add Part dialog box. The Add Part dialog box helps you to place a interface rules file with little effort.

Following are the four scenarios possible while placing the interface rules file on canvas through Add Part dialog box:

### Scenario   Rules File   Mapping File   Procedure

1	Yes	Yes	Read <a href="#">Steps Required for Scenario 1</a> section.
2	Yes	No	Read <a href="#">Steps Required for Scenario 2</a> section.
3	No	Yes	Read <a href="#">Steps Required for Scenario 3</a> section.
4	No	No	Read <a href="#">Steps Required for Scenario 4</a> section.

**Note:** To invoke the Add Part dialog box unselect the Show wizard while placing part on canvas option in Preference dialog box.

### *Important*

The procedure for placing an interface rules file through Add Part dialog box and the four possible scenarios above are very similar to steps explained in DE-HDL section. However there is only one dissimilarity in between both the procedure is selection of symbols. In DE-HDL environment the Component Browser is used as an interface for symbol selection where as in OrCAD environment the Add Part dialog box itself has options for symbol selection. For OrCAD symbol selection it is suggested to go through any of the procedure explained in [Using Create>Select Component Rules and Mapping Information Wizard](#) section. For selecting rules file and mapping files in Add Part dialog box see [Using Add Part dialog box](#) section.

## Adding Interface Component (Libraries)

If you do not wish to bind the symbol and footprint to interface rules file at the time of instantiation, you may choose Libraries to place the interface rules file. Which means you can

opt for late binding once you are done with your design. Libraries lists all the available interface rules file names. You can quickly select the logical model from Libraries and place on the canvas using drag and drop method. The interface rules file which are placed on canvas from Libraries are not real components. An interface rules file is said to be real component when the interface rules file is linked with the symbol and footprint data. There may be a situation like you are neither working in DE-HDL flow nor OrCAD flow and you wish to bind the symbol and footprint data to the interface rules file after completing the design. This is called as post binding. Once you complete your design you can map the interface rules file with associated front-end symbol and footprint. You can do this task any time during the design. For more information see [Converting Components to Real Components](#) section.

If you do not find the desired part in Libraries, you can either create your own interface rules file or Virtual Interface. The functionality of a interface rules file and virtual interface (without symbol and footprint data) is alike. After creating the virtual interface you can convert it to real component at any time.

**Note:** You also have the option to access your own interface rules file from your local directory. To access your own local libraries you need to set the rules file (.lrf) paths in config.ini file. For detailed information see [Working with Libraries](#) chapter for more information.

To place a interface rules file from Libraries you perform the following steps:

1. Click *Interface Rules* tab in the Libraries.
2. Click + in the *Libraries* to expand the tree view structure.
3. Click a part name and drag the selected part name to the canvas and drop it or right-click and select the *Place Component* option from the pop-up menu.

The interface rules file is placed on the canvas. After completing the design or at any time during the design you can convert the interface rules file to real component.

## Replacing Logical Reference Model of the Component

After placing the interface rules file through Component Browser on canvas, you might like to replace the logical reference model of the interface rules file with another interface rules file. Consider an example, at first you selected and placed a ninety two BGA package component (with frontend symbol and footprint). Now you wish to have same package component but with different set of supply and no connect pins. Since you already have an identical component on canvas, FSP lets you to dynamically swap the logical model of instantiated interface rules file. The new interface rules file can have new name or different properties but it should have a compatible footprint (JEDEC\_TYPE property) which will help you to retain the customizing on that object. Replacing reference model of the interface rules file can be

done at any time during the design. But it is recommended that you replace the component before targeting the interface rules file and making connections. Otherwise you may loose connections.

The procedure for replacing the logical reference model of an interface rules file is classified into following flows:

- DE-HDL
- OrCAD

## **DEHDL**

To replace the logical reference model of a interface rules file in DE-HDL schematic environment you follow any of the following methods:

- [Replacing Logical Reference Model with other Logical Model Using PTF Property](#)
- [Replacing Logical Reference Model with other Logical Model from FSP Library](#)

### **Replacing Logical Reference Model with other Logical Model Using PTF Property**

FSP provides you a cascaded pop-up menu on interface rules file called Change Rules File. The Change Rules File menu displays the interface rules file names which are defined under FSP\_LOGICAL\_MODEL property in the PTF file. These interface rules file names associated with the selected PTF row are stored at the time of placing interface rules file on canvas and displayed as options in pop-up menu.

For more information on assigning FSP\_LOGICAL\_MODEL property in PTF file see [Setting Up PTF for Component Selection](#) section.

For example, displayed below is the PTF file of the DDR2\_DIMM\_X4 component.

```
PART 'DDR2_DIMM_X4'
{=====}
=====}
:PACK_TYPE | FSP_LOGICAL_MODEL= PART_NUMBER | DESCRIPTION |
JEDEC_TYPE ;
{=====}
=====}
'XXXX' | 'ddr2_dimm_x4_V4_V5'= 'DDR2_DIMM_X4' | 'ddr2_dimm_x4' | 'mlx-
87705-001'
'DDR2_DIMM_X4' | 'ddr2_dimm_x4_A2GX'= 'DDR2_DIMM_X4' | 'ddr2_dimm_x4' | 'mlx-
87705-001'
```

## Allegro FPGA System Planner User Guide

### Working with Components

---

```
'DDR2_DIMM_X4'
'ddr2_dimm_x4_S2_S2GX_AGX:ddr2_dimm_x4_S3_S4GX_S4E:ddr2_dimm_x4_SP3:ddr2_dimm_x4_
SP3A:ddr2_dimm_x4_V4_V5' = 'DDR2_DIMM_X4' | 'ddr2_dimm_x4' | 'mlx-87705-001'
```

When you select the third PTF row (from example above) in Component Browser and right-click on the interface rules file on canvas the following interface rules file names are displayed as sub menu:

- ddr2\_dimm\_x4\_S2\_S2GX\_AGX
- ddr2\_dimm\_x4\_S3\_S4GX\_S4E
- ddr2\_dimm\_x4\_SP3
- ddr2\_dimm\_x4\_SP3A
- ddr2\_dimm\_x4\_V4\_V5

Select any one interface rules file name from list above to replace the logical reference model of the selected interface rules file. After selecting cell name the complete logical and electrical constraints associated with old interface rules file are replaced with new one.

**Note:** The above options are available for only those interface rules file which are placed through Component Browser.

To replace the logical reference model of a interface rules file perform the following steps:

1. Place an interface rules file on canvas through Component Browser.
2. Right-click on interface rules file and choose *Rules – Change – <interface\_rules\_file\_names>*.

After selecting the interface rules file name FSP performs a consistency check to map the new interface rules file detailed with associated front-end symbol and mapping file. If any inconsistency found between the interface rules file and mapping file, the Select Rules File dialog box is automatically displayed. You can also use the Select Rules File dialog box to resolve the error by creating a new mapping file or using an existing one.

### **Example of Replacing Logical Reference Model of a Real Part**

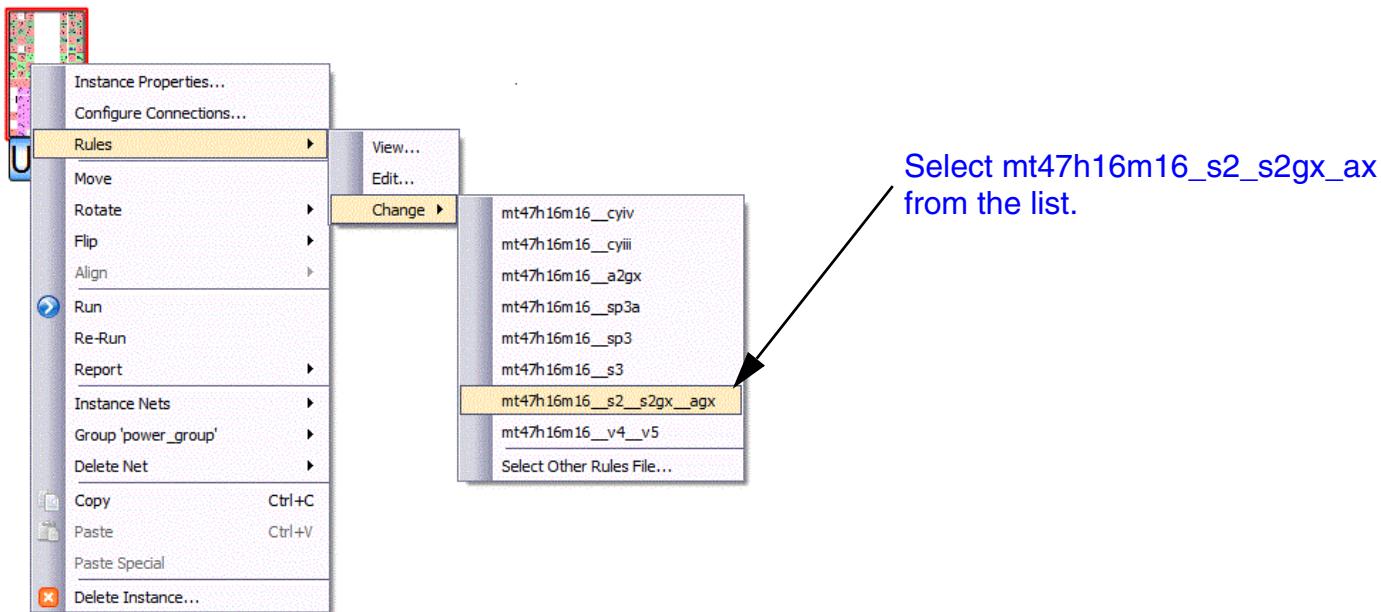
The following example demonstrates the steps to replace the logical reference model of the component which is placed on canvas through wizard flow.

In this example, you will use the mt47h16m16\_\_v6 part from memory\_protocols library to replace with the mt47h16m16\_\_s2\_s2gx\_ax part.

It is assumed that you have placed the mt47h16m16\_\_v6 part on canvas using wizard flow.

To replace the logical reference model of the mt47h16m16\_v6 part:

1. Right-click on the instance mt47h16m16\_v6 and choose *Rules – Change*.



The mt47h16m16\_v6 part is replaced with the mt47h16m16\_s2\_s2gx\_ax part.

### Replacing Logical Reference Model with other Logical Model from FSP Library

You can also replace the logical reference model of a placed interface rules file with any of the existing interface rules files from FSP library.

To replace the logical reference model of the interface rules file perform the following steps:

1. Place an interface rules file on canvas through Component Browser.
2. Right-click on interface rules file and choose *Rules – Change – Select Other Rules File*.  
The Select Rules File dialog box is displayed.
3. Expand the tree view structure by clicking + and select a interface rules file name.
4. Click *OK*.

On clicking Change, FSP checks the consistency between the interface rules file and mapping file and place the interface rules file on canvas. An error message is displayed in log window if any inconsistency is found while replacing.

### Example of Replacing Logical Reference Model of Non-Real Parts

The following example demonstrates the steps to replace the logical reference model of part placed on canvas through Libraries.

**Note:** The steps covered in this example are applicable in both DE-HDL and OrCAD schematic environments.

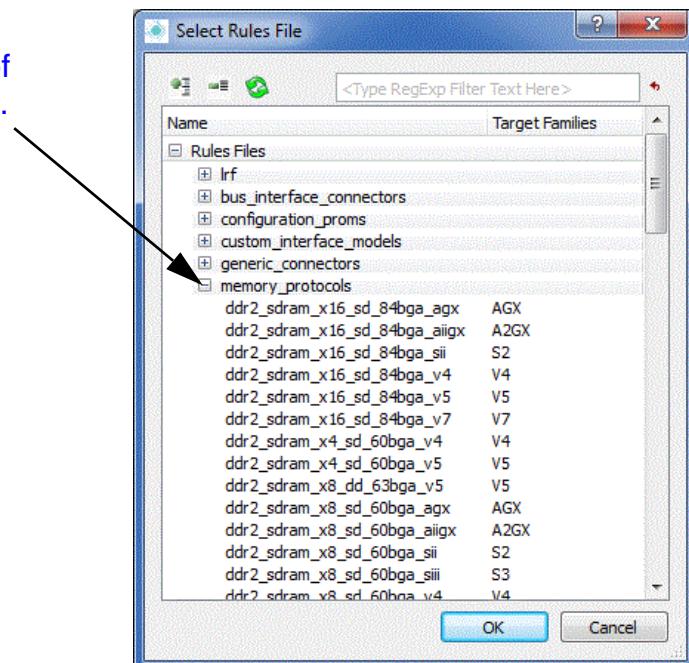
In this example, you will replace the udimm\_184pin\_72b\_ddr\_x8\_v4 part with the sodimm\_200pin\_72b\_ddr\_x8\_v4 part. It is assumed that you have placed the udimm\_184pin\_72b\_ddr\_x8\_v4 part on canvas from Libraries.

To replace the udimm\_184pin\_72b\_ddr\_x8\_v4 part:

1. Right-click on the udimm\_184pin\_72b\_ddr\_x8\_v4 instance and choose *Rules – Change*.

The Select Rules File dialog box is displayed.

Click to see the list of logical model names.



2. Select the sodimm\_200pin\_72b\_ddr\_x8\_v4.
3. Click *OK*.

The udimm\_184pin\_72b\_ddr\_x8\_v4 part is replaced with the sodimm\_200pin\_72b\_ddr\_x8\_v4 part.

## OrCAD

In OrCAD schematic environment, to replace the logical reference model of the placed interface rules file you perform the following steps:

1. Right-click on interface rules file and point to *Change Rules* option.  
A sub menu with list of interface rules file names is displayed.
2. Select any one interface rules file name to replace with the reference model of the current interface rules file.
3. Choose *Rules – Change – Select Other Rules File*.  
The Select Rules File dialog box is displayed.
4. Expand the tree view structure by clicking + and select a interface rules file name.
5. Click *OK*.

After clicking *Change*, FSP checks the consistency between the interface rules file and mapping file and place the interface rules file on canvas. An error message is displayed in Log Window if any inconsistency is found while replacing.

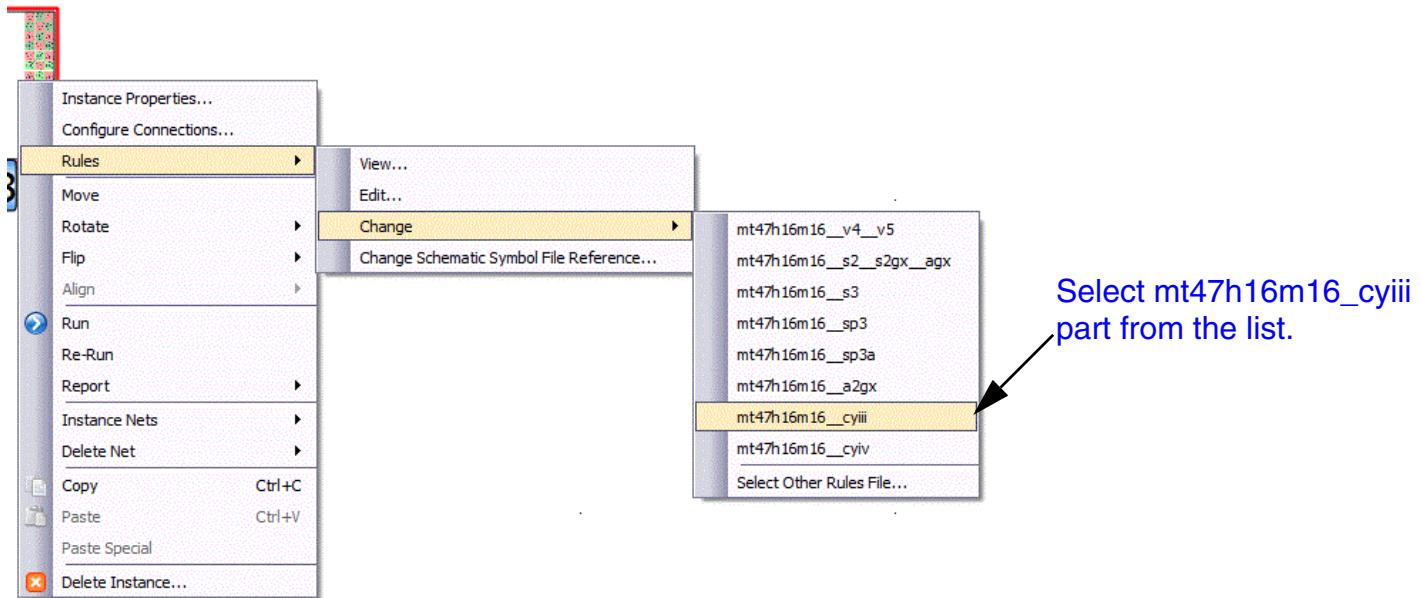
### Example of Replacing Logical Reference Model of Real Part

The following example demonstrates the steps to replace the logical reference model of a real part.

In this example, you will replace mt47h16m16\_\_v6 part with the mt47h16m16\_\_cyiii part. It is assumed that you have placed mt47h16m16\_\_v6 part on canvas through wizard flow.

To replace the mt47h16m16\_\_v6 part:

1. Right-click on the mt47h16m16\_\_v6 instance and choose *Rules – Change*.



The mt47h16m16\_v6 part is replaced with the mt47h16m16\_cyiii part.

## Re-referencing Logical and Mapping of the Instantiated Component

All the design interface instances are referenced to particular interface rules file on disk. The interface rules file can be located at your local project directory area or search path area or library:cell:view directories. After placing the interface instance on canvas through Component Browser (real component), you may decide to modify the interface rules file definition. You can do this through Rules Editor dialog box. Modifying the logical constraints of a real component leads to mapping out of sync. To avoid this error re-reference of logical and mapping of the component need to be done. Once you complete the interface rules file modification following choices are available:

- You can overwrite the new definition on existing definitions of the rules file on the disk. This may cause updation of other real components on canvas reference to the same modified interface rules file.
- You can save the new rules file definition in local directory. This option helps you to manually reference the modified interface rules file to all the similar real components on canvas.

After placing the interface rules file through Create>Select Component Rules and Mapping Information wizard (in OrCAD), the Change Schematic Symbol File Reference dialog box allows you to re-reference the schematic symbol file.

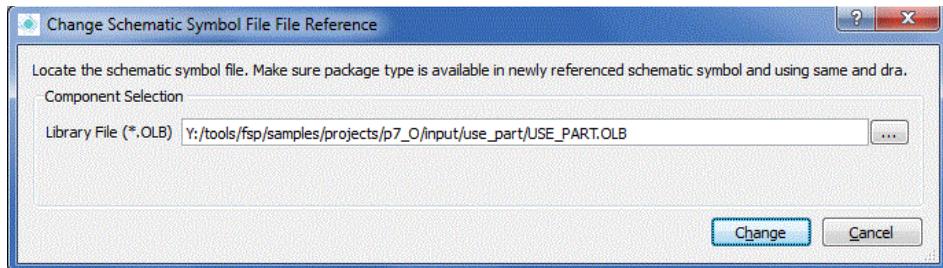
The re-referencing schematic symbol file for the instantiated component is supported only in the OrCAD environment.

## OrCAD

To re-reference the rules and mapping file for an instantiated component perform the following steps:

1. Right-click on the interface rules file and choose *Rules – Change Schematic Symbol File Reference*.

The Change Schematic Symbol File Reference dialog box is displayed.



2. Do one of the following:

- In Library File (\*.olb) field specify the olb file name and path.
- Click *browse(...)* to browse to the location where the olb file exists.

3. Click *Change*.

On clicking Change, FSP automatically validates the new referenced mapping file with interface rules file and report errors if any.

## Converting Components to Real Components

The components which are placed using Component Browser, the front-end symbol selection for them is already done. This means you can directly generate the schematics for the components. To include the interface rules file (without mapping) and virtual interfaces into schematics you have to first convert them into real components. Otherwise the signals

connected to the interface rules file and virtual interfaces are treated as external signals in schematics.

Converting interface rules files to real components process is nothing but mapping the pin names of rules file to symbol pin names. This process can be done at any time during the design. The process of converting interface rules file into real component is similar in both DE-HDL and OrCAD schematic environments. The process of conversion begins with selecting the front-end symbol and footprint followed by selection of logical interface model and mapping pin names. While converting if the pin numbers in both front-end symbol and logical model are unmatched, an error message is displayed and process is stopped.

**Note:** The process of converting interface rules file into real components is common for both interface rules file and virtual interface.

The procedure for conversion is classified into following flows:

- DE-HDL
- OrCAD

## DE-HDL

To convert the interface rules file into real components you use the Component Browser for front-end symbol selection followed by Convert Rules File Instance to Real Part dialog box to select rules and mapping file.

To convert the interface rules file into real component perform the following steps:

1. Place a interface rules file on canvas using *Libraries*.
2. Right-click on the interface rules file and select *Convert to Real Interface* option from pop-up menu.

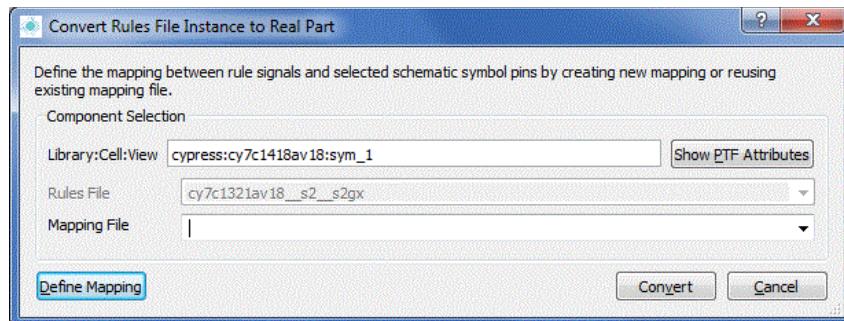
The Component Browser is displayed.

3. Select a library name from the *Library* pane.
4. Select a cell name from the *Cell* pane.
5. Select and click the PTF row from the *Part Name* column.

**Note:** For more information on assigning FSP\_LOGICAL\_MODEL property in PTF file see [Setting Up PTF for Component Selection](#) section.

6. Select the symbol view from the *Symbol* drop down list.
7. Click *Select*.

The Convert Rules File Instance to Real Part dialog box is displayed. By default Library:Cell:View and Rules File name is displayed also the Rules File field is disabled.



**Note:** In this process you are mapping the current interface rules file to the front-end symbol so there is no option to reselect the rules file.

8. Click the *Mapping File* drop down list and select existing mapping file name.
9. Click *Define Mapping*, to define a new mapping file.
10. Click *Convert*.

The selected front-end symbol and footprint is mapped with the placed interface rules file.

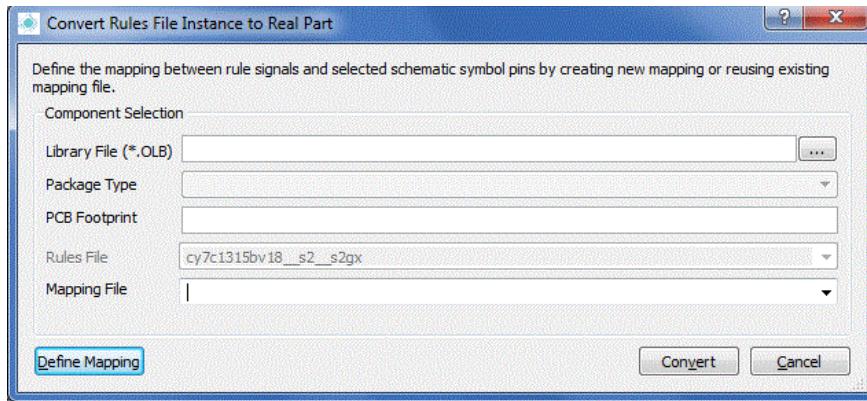
## OrCAD

To convert the interface rules file into real components you use the Convert Rules File Instance to Real Part dialog box.

To convert the interface rules file into real component perform the following steps:

1. Place a interface rules file on canvas using *Libraries*.
2. Right-click on the interface rules file and select *Convert to Real Interface* option from pop-up menu.

The Convert Rules File Instance to Real Part dialog box is displayed.



**3.** Do one of the following:

- In Library File (\*.olb) field specify the olb file name and path.
- Click *browse(...)* to browse to the location where the olb file exists.

**4.** Click the *Package Type* drop down list and select a package name.

After selecting the package name the PCB Footprint field displays associated footprint name.

**5.** Click the Mapping File drop down list and select a mapping file name.

**6.** Click *Define Mapping* to define a new mapping file.

**7.** Click *Convert*.

The selected front-end symbol and footprint is mapped with the placed interface rules file.

## Converting Virtual Interface to Real Components

The functionality of a interface rules file which is placed on canvas and virtual interface is similar. Since both are not considered as real components until and unless they are not mapped to front-end symbol and footprint. The main dissimilarity between the two is the signals of virtual interface are treated as external signals in schematics and also it is not included in the schematics. If you wish to include the virtual interface in schematics you have to convert the virtual interface into real components. The process of converting a virtual interface to real component is similar to the converting component to real component. For more information see [Converting Components to Real Components](#) section.

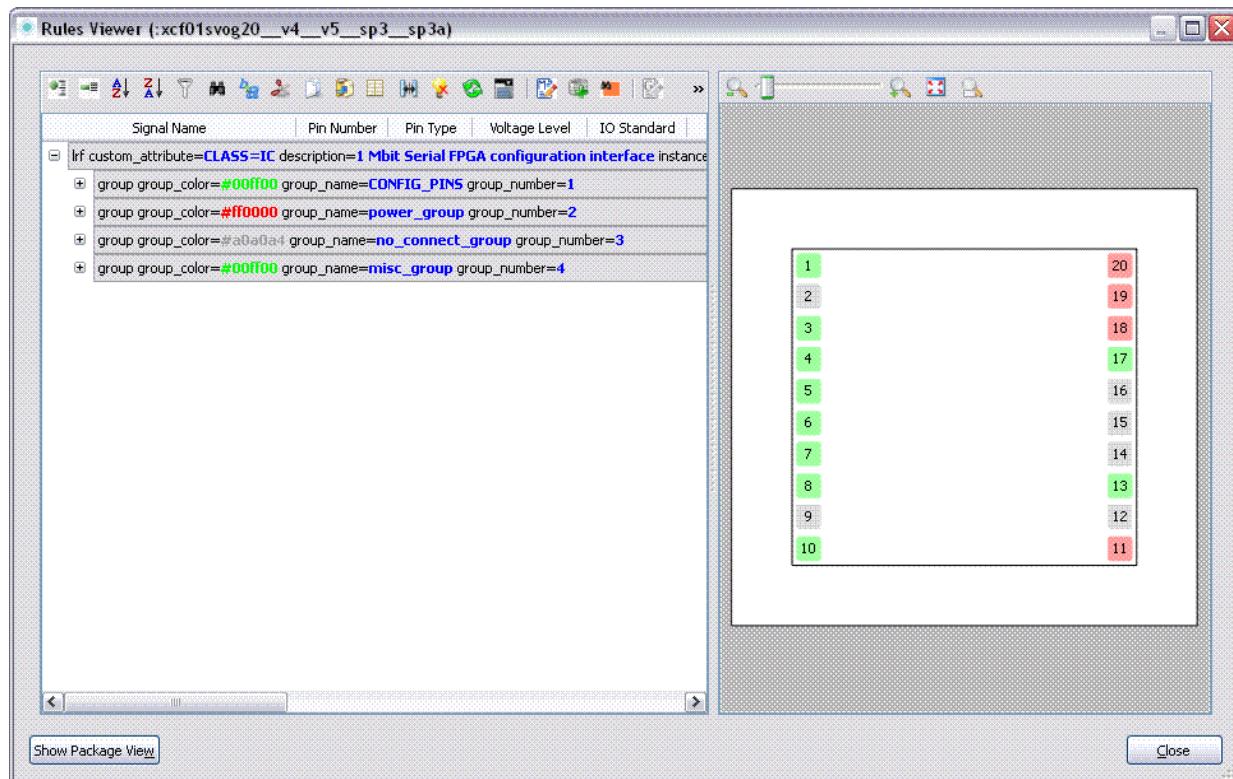
## Viewing Components

FSP provides a spreadsheet view of the logical and electrical constraints of the interface rules file. The Rules Viewer is a read only spreadsheet view of the interface rules file details as it appears in Rules Editor form. This form can only be used to verify whether you have assigned the correct pin and group information details in Rules File editor. You can also export the group and pin informations through this Rules Viewer dialog box.

To view the details of an interface rules file perform the following steps:

1. Place a interface rules file on canvas.
2. Right-click on interface rules file and select *Rules – View*.

The Rules Viewer displays all the group and pin constraints details in disabled mode.



3. Click *Show Package View* to view the symbol view of the component.
4. Click *Export CSV* to export all the pin details in text file.
5. Click *Close* to exit.

## Modifying Components

The interface rules file are modified for various reasons, such as change in the logic design requirements during course of the project. This could be because of updates in the logic itself or for time driven optimization needs. Due to these changes the interface rules file logic and layout need to be changed multiple times during the course of a project. These modification includes setting the differential pair pin type, opposite pair pin type, target pin type (CC, GCC etc.), pin function (data strobe), and group-level constraints (such as use of the same bank to connect a group of pins) so on. You can modify both interface rules file which is placed through Libraries or Component Browser. The procedure for editing is same for both the real component and interface rules file except few dissimilarities. The following table displays the dissimilarities while editing real component and interface rules file.

For detailed information on modifying components see [Editing Parts](#) section.

<b>Real Component</b>	<b>Interface rules file</b>
The following options are not present when you edit the real component through Rules Instance Editor:	All options are present.
<ul style="list-style-type: none"><li>■ Remove Group</li><li>■ Add Signal</li><li>■ Add Bus</li></ul>	
Saving of mapping file and Rules file option is present.	Saving of mapping file and rules file option is not present.
When you edit a real component, it may happen that the associated symbol or dra files are removed from the disk or corrupted, or FSP is not able to read the files, you can still edit and save the component.	You can edit the component at any time.

**Note:** Editing parts from the canvas is not recommended since this deletes the connectivity definitions and leads you to error prone state.

## Adding Device Component

Device rules file are treated differently compare to interface rules file. To select the device component from the central library using Component Browser, FSP\_FPGA property need to be defined in the PTF file. For more information on FSP\_FPGA property see [Setting Up PTF for Component Selection](#) section. FSP provides different ways to place the device rules file on canvas. The procedure to place the device rules file on canvas is classified into following flows:

- [DE-HDL](#)
- [OrCAD](#)
- [Libraries](#)



In both the schematic environments, you are facilitated with two methods for device placement. One is through wizard and another is using Add Part dialog box. The basic steps required for placing the device on canvas using both methods are very similar. The process contains selecting symbols and device rules file. This section describes the procedure to place the device rules file on canvas using wizard.

### DE-HDL

In DE-HDL schematic environment, you use Component Browser to search, select and place the device rules file on canvas. Before you begin the placement, you need to specify the device rules file name in `cell/library/PTF` row as `FSP_FPGA` property value. Though not mandatory step. The `FSP_FPGA` property helps Component Browser to identify the component as device component otherwise the component will be treated as interface which leads you to error prone state. If you do not specify the `FSP_FPGA` property, FSP automatically detects the component and gives you the option to consider it as device rules file and map it to the existing FPGA library component. For more information on specifying properties see [Setting Up PTF for Component Selection](#) section.

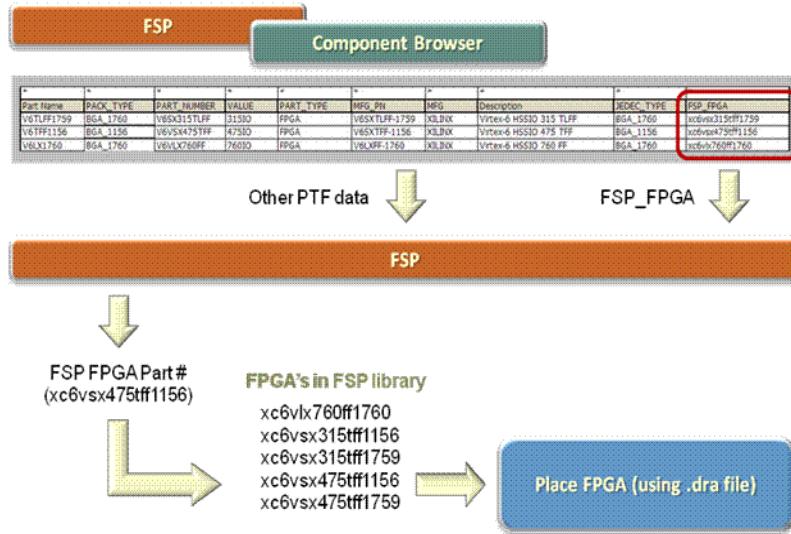
The following diagram illustrates the device placement flow with and without `FSP_FPGA` property:

# Allegro FPGA System Planner User Guide

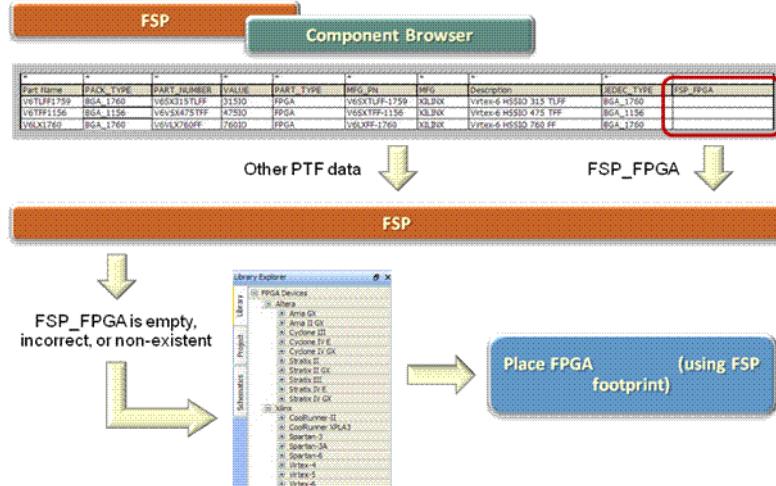
## Working with Components

---

### **With FSP\_FPGA Property**



### **Without FSP\_FPGA Property**



To place the device rules file on canvas using Component Browser perform the following steps:

1. Click Add Part icon in .

The Component Browser dialog box is displayed.

2. Browse for the library whose component you want to add in *Library Pane*.
3. Select a cell name in *Cells* pane or enter the name of cell in *Cells* text box.
4. In *Search Results* pane, click the row corresponding to the physical part you want to add.  
The symbol and footprint name for the component is displayed in *Part Name* tab.
5. Select a symbol view from *Symbol* drop down list.
6. Click *Select*.

After clicking *Select*, a graphical view of device is displayed. Left click to drop the device rules file on canvas and right click to disable the graphical view.

**Note:** After clicking *Select* in Component Browser if you unexpectedly see the Add Part dialog box, then you must specify the `FSP_FPGA` attribute and re-invoke the Component Browser to place the device rules file on canvas.

To place the device rules file (without `FSP_FPGA` property) using Component Browser perform the following steps:

1. Click *Add Part* icon in .  
The Component Browser dialog box is displayed.
2. Browse for the library whose component you want to add in *Library Pane*.
3. Select a cell name in *Cells* pane or enter the name of cell in *Cells* text box.
4. In *Search Results* pane, click the row corresponding to the physical part you want to add.  
The symbol and footprint name for the component is displayed in *Part Name* tab.
5. Select a symbol view from *Symbol* drop down list.
6. Click *Select*.  
The *Create>Select Component Rules and Mapping* Information wizard appears.
7. Select the *FPGA* option and click *Next*.  
The *Place Part* page appears.
8. Click *Finish and Place* to place the component.

## OrCAD

The process of placing the device is similar to placing the interface component. The only difference in both the methods is you do not required a mapping file while placing the device on canvas. FSP lets you to select the OrCAD symbol and link it to associated FSP device component. Basic device placement procedure involves selecting the OrCAD symbol, Package type, foot print and associated device file.

To place the device component on canvas using Create>Select Component Rules and Mapping Information wizard perform the following steps:

1. Click *Add Part* icon in .

The Create>Select Component Rules and Mapping Information wizard is displayed.

2. Do one of the following:

- In Library File (\*.olb) field specify the olb file name along with the path.
- Click *browse (...)* to browse to the location where the symbol file exists.

3. Click the Package Type drop down list and select the package name as required.

The footprint name is automatically displayed in PCB Footprint field.

4. Click *Finish*.

After clicking *Finish*, a graphical view of device is displayed. Left-click to drop the device component on canvas and right-click to disable the graphical view.

## Libraries

Libraries lists all the available device rules file names. You can quickly search and select the device rules file from Libraries, and place on the canvas using drag and drop.

**Note:** When you place a device component on the Canvas, you may see the following message.

*This part has been partially qualified, using <third part software name and version>. It should be considered a “beta” part. For questions about using this part in your design, contact Cadence Customer Support.*

This message implies that the selected device component is partially verified using the specified tool. You may notice that the some of the IO DRC rules are not obeyed while working with the selected device. This kind of device is known as beta device. If you have questions about how to use the part in your design, contact Cadence Customer Support.

To place the device rules file using Libraries perform the following steps:

1. Click + in *Libraries* to expand the tree view structure.
2. Do one of the following:
  - a. Select a part name.
  - b. Drag the selected part name and drop on canvas or right-click and select *Place Component* option from pop-up menu.

Or

- a. Specify an logical file name starting with semi colon (:) or click ... browse to location where the file exists.
- b. Click *Add*.
- c. Move the mouse pointer to canvas.

A part image is displayed.

- Left-click to drop the component on canvas.

The device rules file is placed on canvas.

## Linking Device Component to Front-End Symbol

After placing the device rules file through Libraries you may wish to generate the symbols from FSP. But if you have your own symbol and footprint and wish to use these information for generating schematics you can do this by linking the device rules file to your symbol. While linking the device rules file to front-end symbol the pin numbers of the available dra must match with the device rules file shipped by FSP. The dra specified for the selected front-end symbol is used by the FSP by overriding the existing FSP pin layout information. The process of linking the device rules file to front-end symbol can be done at any time during the design. The process in both the schematic environments are same and begins with selecting the symbol followed by selecting footprint.

The procedure for linking the device rules file to front-end symbol is classified into following flows:

- DE-HDL
- OrCAD

## DE-HDL

To link the device rules file to front-end symbol you use the Component Browser.

To link the device rules file to front-end symbol perform the following steps:

1. Right-click on the device rules file and select *Link to FE Symbol* option from pop-up menu.  
The Component Browser is displayed.
2. Select a library name from *the Library* pane.
3. Select a cell name from the *Cell* pane.
4. Select and click the PTF row from the *Part Name* column.

**Note:** For more information on assigning `FSP_FPGA` property in PTF file see [Setting Up PTF for Component Selection](#) section.

5. Select the symbol view from the *Symbol* drop down list.
6. Click *Select*.

The selected front-end symbol and footprint is mapped with the placed device rules file.

## OrCAD

To link the device rules file to front-end symbol you use the Convert Rules File Instance to Real Part dialog box to select rules and mapping file.

To link the device rules file to front-end perform the following steps:

1. Right-click on the device rules file and select *Link to FE Symbol* option from pop-up menu.  
The Convert Rules File Instance to Real Part dialog box is displayed.
2. Do one of the following:
  - In Library File (\*.olb) field specify the olb file name and path.
  - Click *browse(...)* to browse to the location where the olb file exists.
3. Click the *Package Type* drop down list and select a package name.

After selecting the package name the PCB Footprint field displays associated footprint name.

4. Click the *Rules File* drop down list and select a rule file.
5. Click *Convert*.

The selected front-end symbol and footprint is mapped with the placed device rules file.

## Replacing Device

FSP lets you to dynamically change the device rules file with a new device rules file. The device rules file you place on the canvas through Libraries can be replaced at any time during the design without disturbing the interface settings except the Use Bank. The Use Bank settings is wiped out when you replace the device rules file. You are allowed to replace the current device rules file from the device rules files of same family

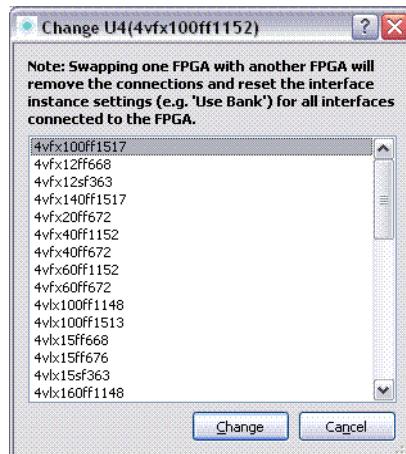
The device you place on the canvas through Component Browser can also be replaced. You will be prompted with a confirmation window before replacing since it is linked with front-end symbol and footprint there might be chances in breakage of links at the time of replacing. So you must link the device and symbol again.

To replace the device perform the following steps:

1. Place a device on canvas through *Libraries*.
2. Right-click on the device and select *Change FPGA* from pop-up menu.

**Note:** A confirmation window is displayed for the device which is placed through Component Browser. Click OK to proceed further.

The Change FPGA dialog box is displayed with all the device rules file names of same family.



## **Allegro FPGA System Planner User Guide**

### Working with Components

---

3. Select one device name and click *Change*.

The device rules file on canvas is replaced with the new device rules file.

---

## **Preparing Components for Connection**

---

This chapter describes the following sections:

- [Overview](#)
- [Creating Interface Protocol \(Interface to Device Connection\)](#)
- [Creating Device Protocol \(Device to Device Connection\)](#)
- [Creating Interface Protocol \(Single Interface to Multiple Devices/Connectors\)](#)
- [Creating Deep and Wide Connections](#)
- [Creating System Ace Device Chain](#)
- [Creating Daisy Chain Connections](#)

## Overview

When a component is dragged and dropped on FSP canvas it called as instances. Device component are called as device instances and interface component as interface instances.



Throughout this chapter we would refer the component placed on canvas as *Instance*.

After placing component on canvas you closely work with them to define the various properties and meet the design requirement of your design.

The following are some of the features you can perform on instances:

- Target the interface instance to device instance
- Specify the contiguous pins in an interface group
- Confine group signals to set of banks
- Specify the number of pins of a bank to use as output
- Configuring device instance pins
- Changing FPGA
- Create Deep and Wide groups to connect same type of interfaces
- Create protocols to connect devices and set common pins

The features above automates the tedious task of preparing the components for connections. In this chapter you will learn about some of the major features listed above such as targeting device instances, creating device protocols, deep and wide connections and System Ace device chain.

**Note:** This chapter does not discuss about creating connections between the instances. For detailed information on how to create connections between the instances see [Running Synthesis](#) section.

## Modifying Instance Name Prefix

FPGA System Planner assigns an instance name to every component you place on canvas. The instance name is defined by the prefix value and a number that is appended by FSP. For

## Allegro FPGA System Planner User Guide

### Preparing Components for Connection

---

example, if `M` specified as instance name prefix value by FSP, the instance names for components will start from `M1`, `M2`, `M3`, and so on.

By default, FSP uses `U` as the instance name prefix. But not all the interface components or devices listed in Libraries are assigned with instance name prefix `U`. Following are the list of component names and the respective prefix used when placing on canvas:

- Devices (Altera, Xilinx and Actel), Configuration, Memories are assigned as `U` by default.
- Connectors (BPC) as assigned as `J` by default.
- Connectors (`DDR2_DIMM_X4`, `DDR2_DIMM_X8` and so on) are assigned as `XP` by default.

At any time during the design you can change the instance name prefix and the number in *Properties of Interface Instance* dialog box for interfaces or in *Properties of Device Instance* dialog box for devices. Modifying instance name prefix will not effect the instance name prefix numbering process. FSP continues to append the prefix number in incremental order.

For example, three components with instance names as `U1`, `U2`, and `U3` are on canvas. If you change the instance name prefix `U3` to `M1`, then the instance name prefix of the next component you place on the canvas is assigned as `U4`.

**Note:** If two components (one from Connector other from Memories) are placed on canvas whose prefix are `U` and `J` respectively, then FSP assigns instance name prefix as `U1` and `J1`.

## Creating Interface Protocol (Interface to Device Connection)

When you have both the interface and device instances in the Canvas, you are allowed to make the connections between them. Before making connections you must ensure that the interface instance belongs to the same family device that you are targeting too.

For example, the `mt46v64m4__v4__v5` instance is eligible to connect to the Virtex-4, 5 devices. In `mt46v64m4__v4__v5` part name, V4 is referred to Virtex-4 and V5 to Virtex-5.

**Note:** To target the `mt46v64m4__v4__v5` instance to other family device see [Adding Properties](#) section for more information.

To create connections you first set the target and few other interface instance parameters which is required at the time of making connections. FSP provides you both automatic and interactive options to let you target the interface to device, edit, and remove the interface instance settings.

The Property Editor, lets you interactively assign and re-assign the target. You can also assign the target on individual interface groups by choosing a device instance name for the respective interface groups. You can do this simultaneously on all the interface groups.

The following example illustrates how to target a single or two groups of interface.

In FSP, a x16 memory part is pre-organized into two separate x8 groups. To create x8 connection you need to select one x8 group and target to the device and to create x16 connection you need to select both x8 groups and target it to the device.

After assigning the target, you can also assign various constraints, such as use bank, contiguous pins, and clock buffer type. Use bank feature enables you to target the interface groups to specific bank of the device.

**Note:** Contiguous pins and clock buffer type are device specific that means these constraints are enabled only for supported devices.

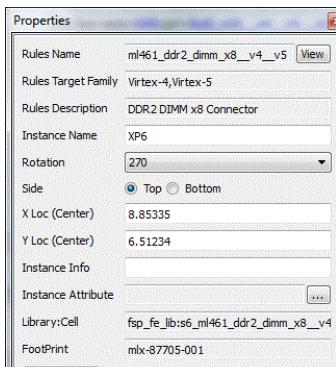
To create a interface protocol connection place an interface and device instance in the Canvas, and perform the following steps:

1. Right-click on the interface instance and do any of the following:
  - ❑ Choose *Target To Device – <Instance Name>*.  
After you choose this option, all the groups of the interface are targeted to device at one go.
  - ❑ Click *Instance Properties*.

# Allegro FPGA System Planner User Guide

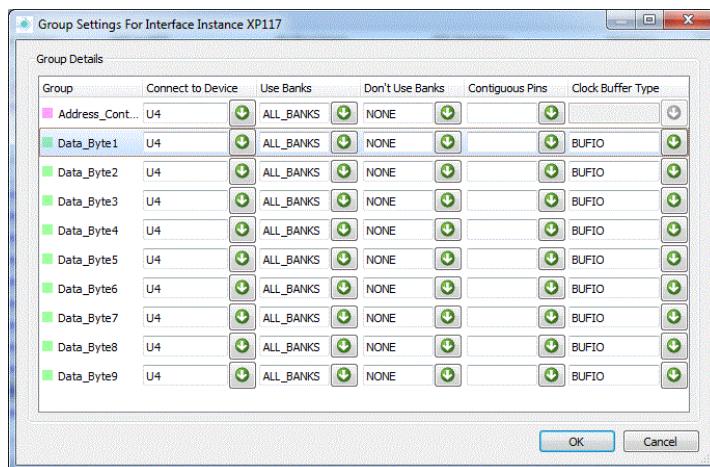
## Preparing Components for Connection

The following page appears in the Properties.



2. Click the *Group Settings* button.

The *Group Settings for Interface Instance<inst\_name>* is displayed.



3. To target all the groups, click the *Connect to Device* column name, click on any one of the drop-down button, and select the device instance name from the drop-down list.
4. To target a single group, click the drop-down button in the first group under *Connect to Device* column and select the instance name.

**Note:** You can perform the same step for other groups.

5. To target the interface groups to a specific bank of the targeted device, do the following:
  - a. To target all the interface groups to one bank, click on the *Use Bank* column name and click on any one of the drop-down button.

## Allegro FPGA System Planner User Guide

### Preparing Components for Connection

---

A pop-up menu is displayed with package view of the targeted device and the list of banks available in the device.

- b.** Select a bank number from the list.
- c.** Click *OK*.

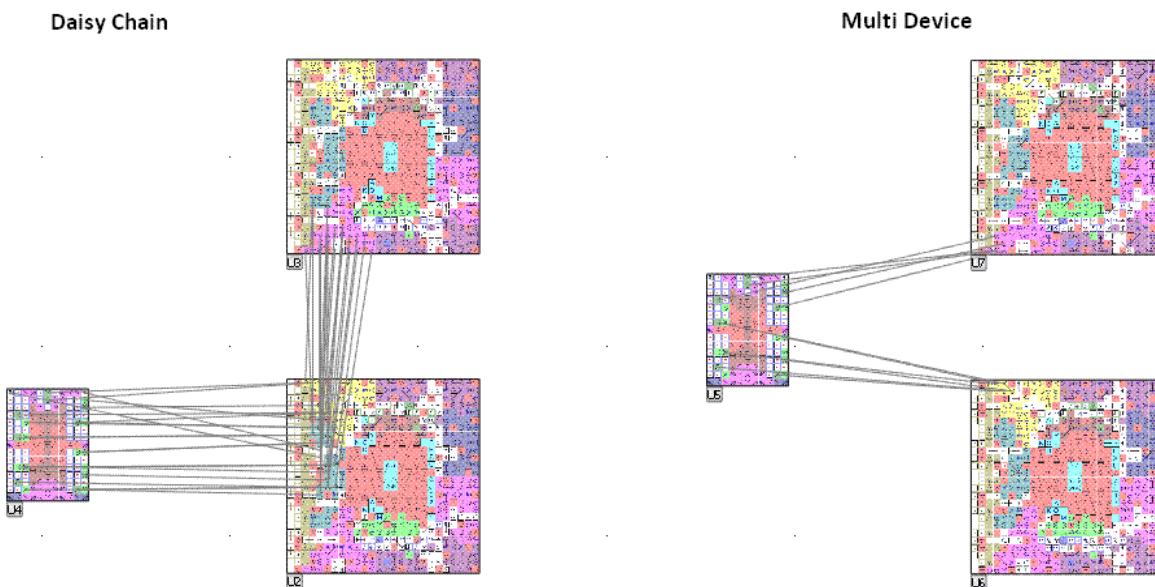
**Note:** To select a bank from the package view, select a pin of a bank to which you want to target the interface group. After you select the pin, the remaining pins of the bank are automatically selected and highlighted.

- 6.** Click *OK* in the Group Settings for Interface Instance <inst\_name> dialog box.

The interface instance group (s) is targeted to the device instance.

## Creating Interface Protocol (Single Interface to Multiple Devices/Connectors)

Sometimes you may require to connect a single interface to multiple devices based on your design requirements. FSP lets you to connect a single interface to multiple devices or connectors. However the interface instance should belong to the same device family that your targeting too. Connections between the instances may be a daisy-chain type or a multi-device type connections. The following figure is intended to provide you the basic difference between daisy-chain and multi-device type connections.



For daisy-chain type connections, you must have atleast more than two device instances on canvas. To target the interface instance to multiple device instances, you use the Properties. After targeting you choose the *Is Daisy Chain* option which appears under the *Connect to Device* column sub-pane. Daisy-chain connection settings can be modified or removed at any time during the design.

### *Important*

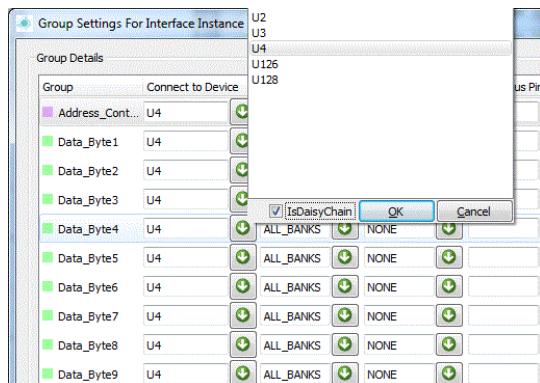
Targeting single interface to multiple devices is supported only in daisy-chain type connections. The *Is Daisy Chain* option must be selected after targeting an interface group to multiple devices.

To create a daisy-chain type connections, perform the following steps:

# Allegro FPGA System Planner User Guide

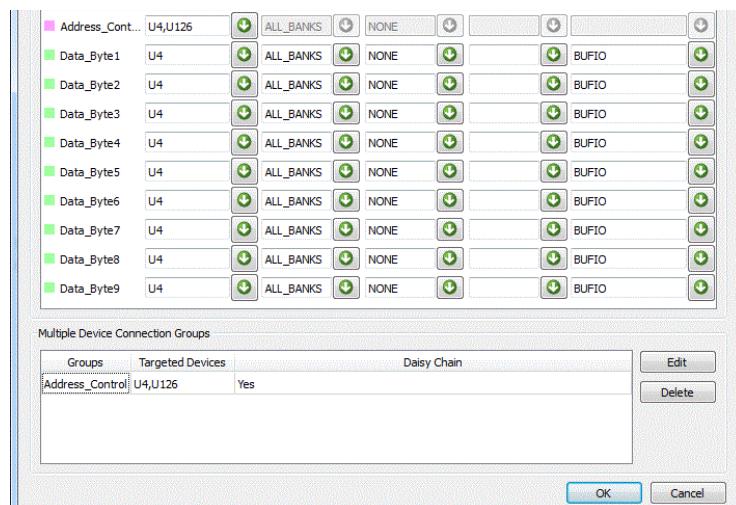
## Preparing Components for Connection

1. Right-click on the interface instance and choose *Instance Properties*.  
The Properties is displayed.
2. In the Connect to Device column, click on the drop-down icon of any one of the group.  
A pop-up menu is displayed.
3. Select instance names from the drop-down list.  
**Note:** Perform the same step for remaining groups also.
4. Select the *Is Daisy Chain* option to create daisy chain connections.



**Note:** After selecting multiple device instance names, you must select the *Is Daisy Chain* option. If you do not select and click OK, FSP displays an error window and halts the process.

A small pane is displayed at bottom of the Properties.

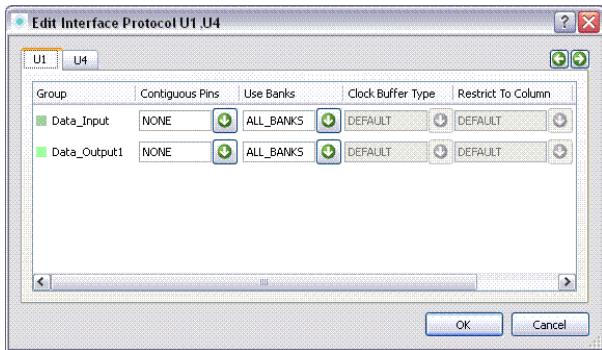


## Allegro FPGA System Planner User Guide

### Preparing Components for Connection

5. Select the row and click *Edit*.

The Edit Interface Protocol dialog box is displayed with multiple tabs. These tabs are displayed based on the number of devices you selected for connection. The group settings provided in the tabs are independent and applicable to the respective devices only.



6. Click *Use Banks* and select the bank number from the drop-down list and click *OK*.
7. Click *OK* in the Edit Interface Protocol dialog box.

To create a single interface to multiple devices, perform the following steps:

1. Right-click on the interface instance and choose *Instance Properties* option.  
The Properties is displayed.
2. In *Connect to Device* column do the following:
  - a. Click the drop-down icon of the first group and select a device instance name.
  - b. Click the drop-down icon of the second group and select another device instance name.
3. Click *OK*.

## Creating Device Protocol (Device to Device Connection)

When a design has multiple devices, communication between them becomes complex. Communication between two devices happens only through buses or protocols. With FSP, you connect multiple devices to meet the requirements for creating protocols for complex designs. For example, an application specific integrated circuit (ASIC) design has multiple devices connected to each other and serving a kind of data signaling. In such cases devices need to connect to each other. You do this by creating protocols between them. Using FSP you can also create point-to-point, multi-point protocols for communication between the devices.

In FSP you use Edit Protocol form to create device protocol. The Edit Protocol form gives a spreadsheet editor view through which you assign the pin details required to create device protocol for two or more devices. A device protocol creation starts with adding set of signals and defining the following information:

- Signal Name
- Pin Type
- I/O standards
- OnChip Termination
- Target Pin Property
- Diff. Type
- Diff Pair Signal
- Serial IO TX/RX Signal
- FPGA Ext Termination
- Net Name
- FPGA Port
- External Port
- Assigned to Pin

**Note:** Signal Name, Pin Type and IO Standards column values are mandatory for creating a basic device protocol. Rest of the column values can be defined later based on the requirements of your design.

Using device protocols information, FSP rule engine optimizes pin-out connectivity between multiple devices in design that meets DRC constraints. If multiple devices are available, you can create more than one protocol to meet the requirements of your design.

## The Edit Protocol Editor



**Note:** By default, the Edit Protocol dialog box is displayed blank. When you add pins in the editor then rows are displayed.

The Edit Protocol editor provides an intuitive and easy way of creating device protocol. It provides a spreadsheet editor view through which all the logical information can be specified. Pin details can be specified manually or can be imported from external data files or rules file. The Edit Protocol editor contains tabs for the devices to which you are going to create the protocol. Each tab represents the ports of the respective device instance. The first tab on the left hand side represents the source device instance of the protocol and then the order continues. In the figure above, U1 is the source device instance of the protocol, which is followed by U3 in the continuous order.

After adding pins to the editor you can start assigning the value for the columns signal name, pin type, IO standards, pin termination and so on. You can also create groups, edit groups, pins, clock groups and specify logical constraints based on the device you have selected.

The process of defining and assigning the column values are similar to the part creation process but you need to remember few points while creating device protocol:

- Signal Name and I/O Standard column value is applied to all the device instance tabs. That means if you edit or add a new signal name in Signal Name column value this will be updated in all the Signal Name column of device instance tabs.
- For two device protocol, when you assign the Pin Type value for first device then Pin Type column value for second device is automatically updated as opposite value. For example, if you select Input as Pin Type for first device then automatically for second device Output is assigned. For more than two device protocol, only InOut is supported as Pin Type.
- Other column values such as On Chip Termination, Pin Property, FPGA Ext. Termination and so on are applicable to the respective devices. Which means changes will not be applied to all the tabs.
- You can shuffle the device protocol chain order using *Left* and *Right* arrow buttons in Edit Protocol editor.

Once you finished with device protocol creation you can validate the inputs you have assigned in Edit Protocol editor using *Validate* option. This option will check the protocol information and generate a report in the log window according to it.

For more information on fields and buttons of Edit Protocol form see [Edit Protocol](#) section and for creating groups and defining values in the Edit Protocol editor you can refer to the [Creating Parts](#) section.

The Edit Protocol editor gives you the following quick options using which you can create device protocol at one shot.

- [Using Existing Rules File](#)
- [Using Existing Protocol](#)
- [Using Constraints/Pinouts File](#)
- [Using CSV File](#)

## Using Existing Rules File

FSP lets you to import the pin details from existing rules file and creating device protocol from it. The process begins with selecting existing rules file followed by editing the imported pin details if necessary and ends with validating it. While importing the rules file, few column values are ignored based on the device you selected for protocol creation. For example the device you selected does not support HSTL\_I\_15 standard and the rules file you are importing contains the HSTL\_I\_15 as IO standard then the IO Standard values are ignored

by FSP while importing. Values for any missing headers can be specified later after importing the rules file editor.

**Note:** Invalid characters or invalid bus notations are unsupported by FSP. For more information see [Guidelines for Creating Parts](#) section of Creating Parts chapter.

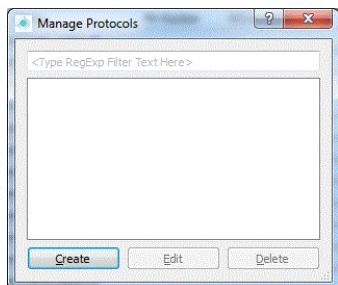
To create device protocol from existing rules file, perform the following steps:

1. Right-click on Device Instance and choose *Instance Properties*.

The Properties is displayed.

2. Click *Manage Protocols*.

The Manage Protocol dialog box is displayed.



3. Click *Create*.

The *Create New Protocol for <inst\_name>* dialog box is displayed.

4. Select the instance names in the right side pane and click  $\rightarrow$  button to move them to the left side pane.

You can also shuffle the order of the device instances for connection by clicking the *Up* and *Down* arrow button.

5. Click *OK*.

The Edit Protocol editor is displayed.

6. Click *Import Signals From* in Edit Protocol dialog box.

7. Select *Existing Rules* from pop-up menu.

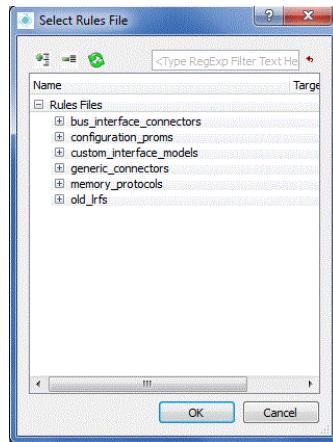
A confirmation pop-ups prompting you about overriding the existing definitions in editor.

8. Click *Yes* to continue.

The Select Rules File dialog box is displayed.

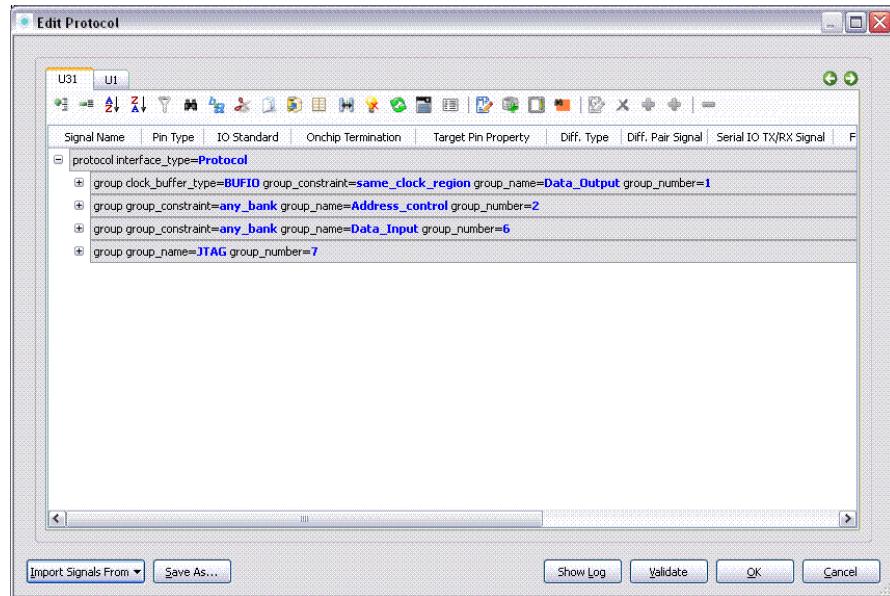
# Allegro FPGA System Planner User Guide

## Preparing Components for Connection



9. Select a rules file from the list and click *OK*.

The pin details of the selected rules file is imported and displayed in Edit Protocol editor. Before you save the protocol it is recommended that you must check whether if any missing column values are present in the editor or not. Otherwise you will not be able to save the protocol.



10. Click *Validate* to check the pin details.
11. Click *Save As* to save the protocol in other directory.
12. Click *OK* to save the protocol in current project directory.
13. Click *OK* of Properties of Device Instance <device instance name> dialog box.

The protocol is created using existing rules file at specified location.

## Using Existing Protocol

This is another option to create device protocol using existing protocol file. In FSP, the protocol information are stored in protocol rules file (.prf). When you import the file in Edit Protocol editor the protocol details are read by FSP. Any incorrect information present in the protocol file is ignored by FSP.

To create protocol from existing protocol rules file perform the following steps:

1. Click *Import Signals From* in Edit Protocol dialog box.

2. Select *Existing Protocol* from pop-up menu.

A confirmation pop-ups prompting you about overriding the existing definitions in editor.

3. Click *Yes* to continue.

The Load Protocol File window is displayed.

4. Browse to the folder where the file exists and select protocol rules file and click *Open*.

The pin details of the selected rules file is imported and displayed in Edit Protocol editor. Before you save the protocol it is recommended that you must check whether if any missing column values are present in the editor or not. Otherwise you will not be able to save the protocol.

5. Click *Validate* to check the pin details.

6. Click *Save As* to save the protocol in other directory.

7. Click *OK* to save the protocol in current project directory.

8. Click *OK* of Properties of Device Instance <device instance name> dialog box.

The protocol is created using existing protocol file at specified location.

## Using Constraints/Pinouts File

This is one more option to create device protocol using constraints/pinouts file. FSP lets you to use the verilog files and constraint/pinout files generated using tools from FPGA vendor Xilinx, Altera and Actel to create device protocol. It is not mandatory to have constraint files while creating device protocol. You can define the pin constraints (IO Standard) later after importing the verilog file in Edit Protocol editor. You are also facilitated with an option to select the signals from the files you want to use in device protocol.

## Conversion Details

- The following values are read from HDL file:
  - Signal Name
  - Pin Type
- The following values are read from constraint files:
  - Signal Name
  - IO Standard
  - On Chip Termination
  - Pin Number
  - Diff Type
  - Diff Pair Pin
  - Serial IO TX/RX Pin
- When you import both HDL file and constraint file, the port names are matched in the both the file.
  - If port names are unequal, the signal names are read from constraint file and associated pin types are read from HDL type.

Example, if a, b are the signal names present in HDL file and d, e are the signal names in constraint file. Since both are unequal, while importing d, e are read from constraint file as signal names and pin types of a, b signal names are read from HDL file.

- If port names are equal, the matched port names will be read as signal names and pin types will be read from HDL file.
  - The IO standards will be ignored while importing, if the IO standard in constraint files is not supported by the device. You must select a different standard from IO Standard drop down list.

For example, HSTL\_II IO standard is unsupported by Spartan-3 device. While importing the constraint files, the HSTL\_II is ignored. You need to select a different standard supported by Spartan-3 device.

To create device protocol from constraints/pinouts file perform the following steps:

1. Click *Import Signals From* in Edit Protocol dialog box.
2. Select *Constraints/Pinouts File* from pop-up menu.

**Note:** In case pin details exist in the editor, a confirmation pop-ups prompting you about overriding the existing definitions in editor.

3. Click Yes to continue.

The Import Verilog/VHDL Signals for Device Instance <instance\_name> is displayed.

4. Select the type of HDL file you are importing in HDL Type field.

For example, Select *Verilog* option if your are importing verilog file.

5. Specify the HDL file name and path in *HDL File* text box or click ... to browse to the location where the file exists.

6. Specify the constraints/pinouts file name and path in *Constraints/Pinouts File* field or click ... to browse to the location where the file exists.

7. Select the bus notation available in constraint file from *Module/Entity* drop down list.

8. Click *Check All* to select all the signals or you can manually select the check boxes under *Import* column.

9. Click *Import* to import all the signals.

The pin informations are displayed in Edit Protocol dialog box. You can proceed with further modifying the pin details if required.

10. Click *Validate* to verify the pin details.

11. Click *Save As* to save in other directory.

12. Click *OK* to save in current project directory.

The device protocol is created using constraint/pinouts file at specified location.

## Using CSV File

Another option to create device protocol from external data files of different format such as comma separated file (.csv), tab separated file or space separated file. The process of importing and conversion details of the pins are explained in detail in Creating Parts section. For more information see [Creating Parts from External Files](#) section.

To create device protocol from comma separated file perform the following steps:

1. Click *Import Signals From* in the Edit Protocol dialog box.
2. Select *CSV* from the pop-up menu.

**Note:** In case pin details exist in the editor, a confirmation pop-ups prompting you about

# Allegro FPGA System Planner User Guide

## Preparing Components for Connection

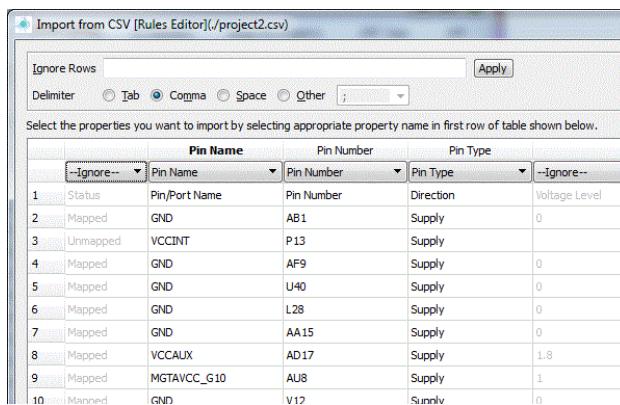
overriding the existing definitions in editor.

3. Click Yes to continue.

The *Select CSV File* dialog box is displayed.

4. Browse to the file, select the file, and click *Open*.

The Import from CSV dialog box is displayed.



5. Specify the file name and path to the directory or click *Browse* to browse to the location where the CSV file exists.

6. Specify the row numbers of the CSV file to be ignored while importing.

7. Select the delimiter option.

8. Click *Load /Refresh* to display the file content.

By default, the column names are automatically detected and displayed. You can change the column header based on your requirement.

9. Click the drop down list of the first cell of every row to select Column Names.

**Note:** Selecting the column names is very crucial. Based on the selected column names, the pin information gets imported in Rules Editor dialog box.

10. Click OK to import pin details in the *Rules Editor*.

The extracted information is displayed in Edit Protocol editor.

**Note:** Error or warning messages are displayed in log window for any incorrect information.

You can either save the protocol information or proceed further organise the pin details into groups based on your requirements. For detailed information on editing options in Edit Protocol editor see [Creating Parts Manually](#) section.

## **Allegro FPGA System Planner User Guide**

### Preparing Components for Connection

---

11. Click *Validate* to verify the logical information.
12. Click *Save As* to save in other directory.
13. Click *OK* to save in current project directory.

The device protocol is created using external data file at specified location.

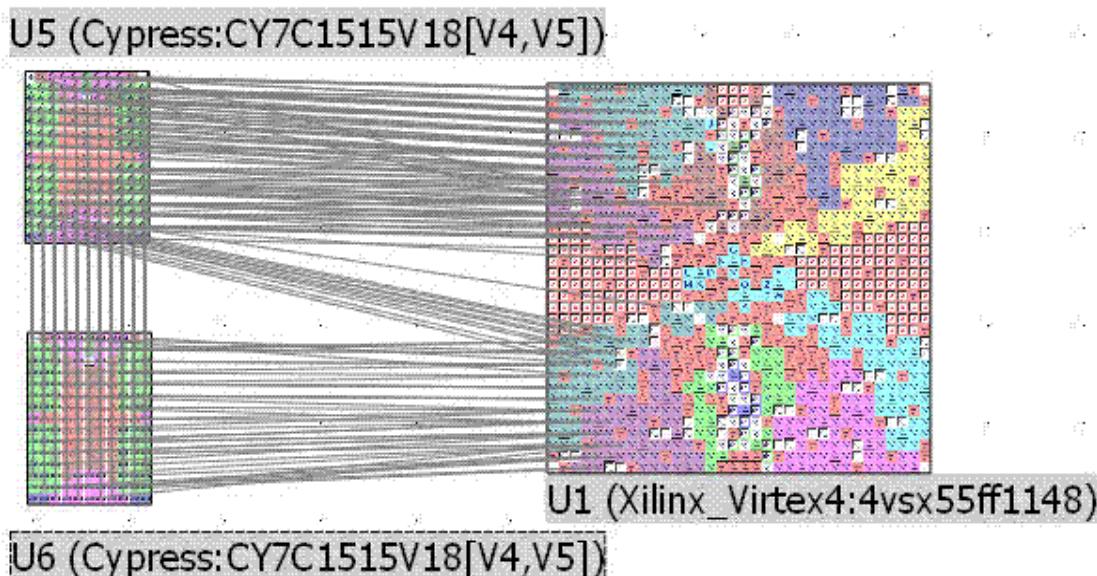
## Creating Deep and Wide Connections

Some times smaller memory components may have to be combined to form a larger memory component that has a wider data bus or a deeper address bus. The Deep and Wide functionality lets you define the connections between two similar interfaces and bind them into a larger component. For example, you create an x32 bit DDR2 interface using four x8 bit components by defining the required Deep and Wide groups.

Deep and Wide groups are classified into two types:

- **Common Groups:** A common group is an ordered set of interfaces and a set of pins for each of these interfaces that share a common connection to the FPGA. The first interface of the list of selected interfaces is called primary interface, and all the remaining interfaces are called secondary interfaces.
- **Wide Buses:** A wide bus is an ordered list of interfaces (which can be repeated) and a corresponding set of pins that must be wide buses into a single bus. You can provide the name of the wide buses. If an interface is repeated in the list of interfaces, the sets of pins selected for each of the entries should be mutually exclusive.

The following figure is intended to provide you an overview of deep and wide connections on FSP canvas.



You define the Deep and Wide groups (either common groups or wide buses), using interfaces even before targeting them to any device.

Deep and Wide groups can be defined either on a set of homogenous interfaces (interfaces of the same type of component) or a set of heterogeneous interfaces (interfaces of different types of components). If heterogeneous interfaces are used, a one - to - one mapping for the pins must be provided as well.

## Creating Deep and Wide Common Group for Homogenous interfaces

Consider an example of four X4 DDR SDRAM interfaces placed on canvas. Assume that the address pins of the interfaces (U4 U5 U6 U7) , data pins of the interfaces (U5 U6) and the clock pins of interfaces (U6 U7) should share a common connection to the FPGA.

In this case, there are three common groups need to be created:

- CommonGroup0 (U4 U5 U6 U7) -> DDR\_A[0:12]
- CommonGroup1 (U5 U6) -> DDR\_D[0:8]
- CommonGroup2 (U6 U7) -> DDR\_CLK\_N and DDR\_CLK\_P

To create the common group perform the following steps:

1. Choose *Tools – Deep and Wide groups*.

The Define Deep N Wide Groups dialog box is displayed.



2. Click *Create Common Group*.

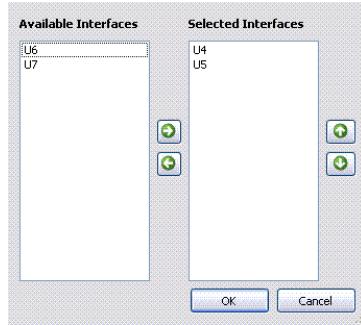
By default a group is created as CommonGroup0 in Deep and Wide Groups text box. If needed you can redefine the group name by double-clicking on the CommonGroup0 text.

3. Click *Select Interface* to select the interface.

A new sub-pane pop ups as below shown.

## Allegro FPGA System Planner User Guide

### Preparing Components for Connection



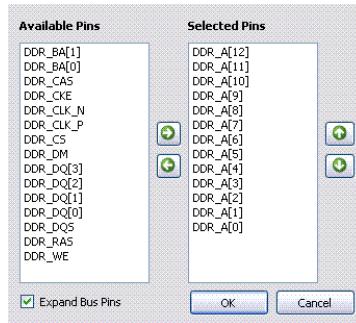
All the interfaces which are placed on canvas are listed in Available Interfaces pane. The interfaces that are intended for common group need to be moved in Selected Interfaces pane.

4. Select *U4 U5 U6 U7* in Available Interfaces pane and click > icon to move them to Selected Interfaces pane and click *OK*.

You can even change the order of the selected interfaces by clicking *Up* and *Down* arrow button. New text boxes as *U4 U5 U6 U7* is displayed in CommonGroup0 column.

5. Click *browse ...* to select the pins for wide connection.

A sub pane pop-ups as below shown.



6. Select *DDR\_A[12:0]* in Available Pins column and click > icon to move them to Selected Pins column and click *OK*.

**Note:** Select *Expand Bus Pins* to enable the pin-level selection instead of bus selection.

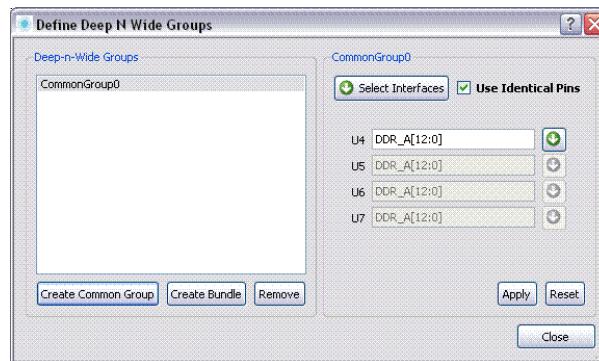
**Note:** In any case, the count of the selected pins should be same for the selected interfaces, otherwise tool will pop a warning window.

7. Select *Use Identical Pins* to continue with the same set of selected pins *DDR\_A[12:0]* of U4 for other interfaces also.

## Allegro FPGA System Planner User Guide

### Preparing Components for Connection

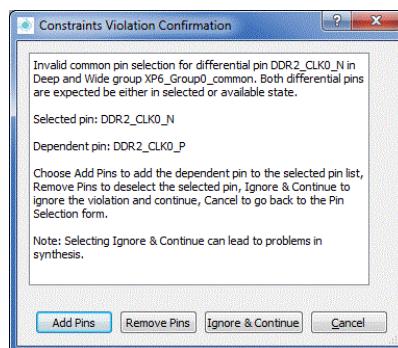
After selecting Use Identical Pins option the U5, U6, U7 text boxes are disabled and DDR\_A[0:12] pin names are updated in there respective text boxes. Going further if you make any changes U4 selected pins, the tool automatically updates the selected pins for the other interfaces also.



8. Click *Create Common Group* to create second group for U5, U6.  
Select DDR\_D[0:8] as common pins by following Steps 1 to 7 as above described.
9. Click *Create Common Group* to create third group for U6, U7.  
Select DDR\_CLK\_P/N as common pins by following Steps 1 to 7 as above described.

#### Important

It is recommended to include the differential pair pins, such as DDR\_CLK\_N and DDR\_CLK\_P pins in the same common group. If you fail to do so, you receive a *Constraint Violation Confirmation* window, as depicted below:



You can use the options available in the confirmation window to resolve this violation.

10. Click *Apply* to apply the settings.
11. Click *Close* to exit.

**Note:** You can edit the common groups at any time during the design.

After you create the Deep and Wide groups, you can target the interfaces (if you have not already done so).

## **Creating Deep and Wide Common Group for Heterogeneous interfaces**

To create Deep and Wide common group for heterogeneous interfaces follow the steps from 1 to 7 of Creating Deep and Wide Common Group for Homogenous interfaces section.

Few points you need to remember:

- Use Identical Pins option is disabled for this type of connection.
- The pin count selection should be always same for the selected interfaces otherwise the tool will pop up a Warning window.
- Pins of any interface that are not used in a Deep and Wide group are treated as normal pins. They can also be used to define other Deep and Wide groups.
- Pins will not be visible in the pin selection pane if they are already used in other Deep and Wide groups.
- If the IO standard of pins of the primary and secondary interfaces are different, the tool makes FPGA pin assignments based on the standards of the primary interface pins. User is responsible for verifying the compatibility of the standards of the primary and secondary interface pins.

## **Creating Deep and Wide Buses for Homogenous interfaces**

When your design has multiple interfaces with nets of different names, you can group these nets into a bus by creating Deep and Wide buses. A group of nets is called as Bus.

For example, to create the three wide buses with data pins, DM and DQS of three x4 DDR SRAM components, as shown below.

- Wide buse0 (U3 U4 U5) -> DDR\_DQ[3:0]
- Wide buse1 (U3 U4 U5) -> DM
- Wide buse2 (U3 U4 U5) -> DQS

To create the Deep and Wide Buses perform the following steps:

1. Choose *Tools – Deep N Wide Groups*.

# Allegro FPGA System Planner User Guide

## Preparing Components for Connection

The Define Deep N Wide Groups dialog box is displayed.

**2. Click *Create Bundle*.**

By default a group as First Bundle is created in Deep N Wide Groups pane. You can redefine the bundle name double clicking on the text.

**3. Click *Select Interface*.**

A new sub-pane pops up.

**4. Select *U3 U4 U5* in Available Interfaces pane and click *>* icon to move them to Selected Interfaces pane and click *OK*.**

You can also change the order of the selected interfaces by clicking *Up* and *Down* arrow icon. New text boxes is displayed as U3, U4 and U5.

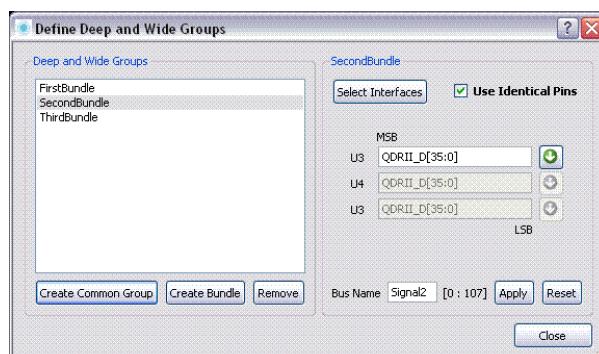
**5. Click *Down* arrow icon of U3.**

A sub pane pop-ups.

**6. Select *DDR\_DQ[3:0]* in Available Pins pane and click *>* icon to move them to Selected Pins pane and click *OK*.**

**7. Select *Use Identical Pins* to use the same set of pins for U4 U5 also.**

**8. Enter *Signal1* in the Bus Name text box as bus name for the selected interface pins.**



**9. Create wide buses for DM, DQS pins by following steps from 2 to 8.**

**10. Click *Apply* to apply the settings.**

**11. Click *Close* to exit.**

**Note:** You can view the bus names in Change Signal Form.

## Creating System Ace Device Chain

System Ace is considered as a normal interface in FSP which ships along with supplied libraries. The System Ace component is pre-organized into eight groups such as clock controls, power, ground, CONFIG\_JTAG, No Connect, Test\_JTAG, MPU and Compact Flash. In FSP only CONFIG\_JTAG group pins is used to interface with FPGA's.

You can view the group and pin details of the System Ace component by clicking View Model in Interface Instance Properties of System Ace dialog box. A System Ace component can connect to any number of devices which is called as device chain. FSP provides you an interactive way through which you connect a System Ace component to any number of devices on canvas. Using the method you can even edit the settings or remove any devices from the chain during the design.

Drop an System ACE and three devices on canvas and perform the following steps to create System ACE device chain:

1. Right-click on the ACE instance and select *Instance Properties* from pop-up menu.

The Properties for Interface Instance <Instance Name> is displayed.

2. Click *Edit* of the System Ace Device Chain text box.

The SystemAce Dialog dialog box is displayed. All the device instance names are displayed in Available Device Instance(s) pane.



3. Select the device instance with which you want to create ACE device chain and click > to move to Connected Device Instance(s).

Move the instance names up and down by using *up* and *down* arrow button.

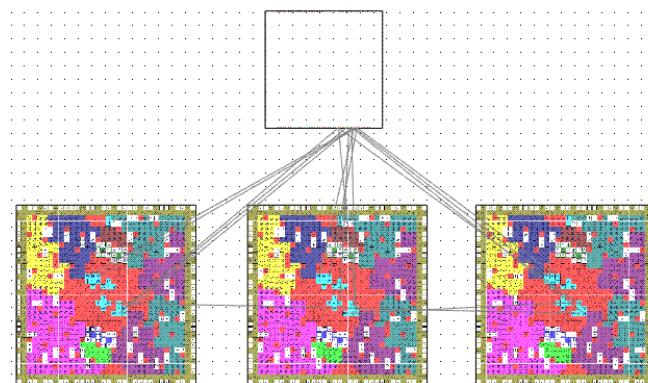
4. Click *OK*.

5. Click *OK* of Properties of Interface Instance <Instance Name> dialog box.

After running the design you will observe the following connections on canvas:

- The TMS, TCK and CFG\_PROG pins are connected to the configuration pins of three FPGA's.
- The INT pin of Ace is connected to common INT pins of the opposite devices.
- CFG\_TDO of Ace is connected to the TDI pin of the first device.
- Then the TDO pin of the first device pin is connected to the TDI pin of the second device, and the TDO pin of the last device is connected to the CFG\_TDI pin of ACE.

For a clear view of the System Ace device chain in FSP see the below diagram.



## Creating Daisy Chain Connections

A tester board contains one or more Design Under Test (DUT) components and other components such as FPGA and connectors to which the DUTs have to be connected. To optimize the connectivity between the DUTs and connectors for minimal crossovers, FSP provides you the ability to create a daisy chain connectivity. In daisy chain connectivity, FSP selects pins of one connector at a time from a target set.

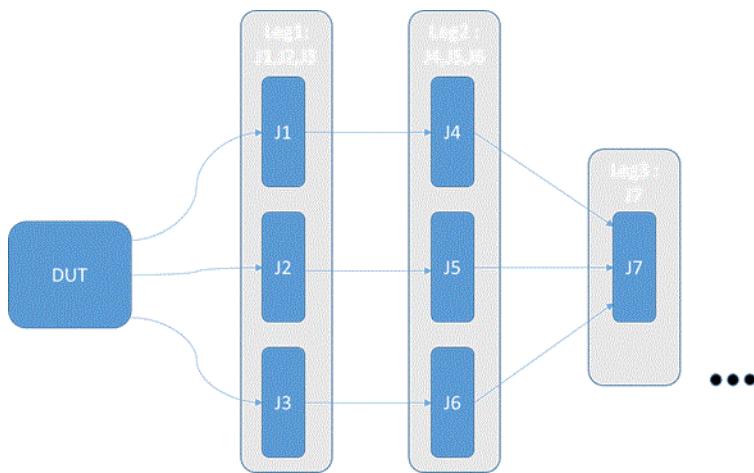
### Defining a Target Set for Connectors

You can associate multiple connectors to a target set by assigning a target set name to the connectors.

The following diagram illustrates different target sets in a design.

# Allegro FPGA System Planner User Guide

## Preparing Components for Connection



In the diagram above, J1, J2, and J3 are associated to a target set, Leg1. Similarly, J4, J5, and J6 belong to another target set, Leg2. Finally, J7 belongs to a target set, Leg 3. You can create as many target sets as required based on your design need. During synthesis, one connector from every leg will be used to connect with the Design under Test (DUT) component. This type of connectivity is known as daisy chain connectivity. For example, from the diagram the following are considered daisy chain connections:

- DUT > J1 > J4 >J7
- DUT > J2 > J5 > J7
- DUT > J3> J6 > J7

To create a target set, do the following:

1. Choose Design – Create Target Sets.  
The Create Target Sets dialog box appears.
  2. Click Add to create a new group.  
A name, TS1 appears by default.
  3. To rename the group, double-click the field label, and then type the new name.
  4. Select one or more connector names in the Available Connectors pane.
  5. Click > to move the selected names to the Selected Connectors pane and click Save.
- Note:** You can perform steps three to six to create more target sets.
6. Click Close once you finish creating target sets.

## Defining Connectivity for Target Sets

After defining the target sets, you target the DUT groups to target sets.

You should ensure the following points before setting the target:

- In DUT component, pins are appropriately distributed and defined in mode levels. This allows better control on connectivity and optimization.
- In DUT component, the group pins are surrounded nearer to each other. This helps to achieve better route connectivity.

To specify a target for a target set, do the following:

1. Right-click on the DUT and choose Group Settings.

The Group Settings for Interface Instance <instance\_name> dialog box appears.

2. In the Connect to Device column, click the drop-down button, press Ctrl and select one or more target sets or a connector.

**Note:** To establish daisy chain connectivity, you must select at least two or more target sets or connectors.

3. Select the IsDaisyChain option in the drop-down pane and click OK.

**Note:** If you have selected a target set, it is recommended that you select the IsDaisyChain option. If you click OK without selecting the option, a warning window appears prompting you about selecting the option.

4. Follow steps one to five to apply targets for other groups of DUT.

When you select the same set of target sets or connectors for DUT groups as a target, the groups will be considered as a multiple device connection group and appear in the bottom pane. You can either delete or edit the group settings.

## Establishing connectivity between target Sets and DUT

After setting the target, create connections between the DUT component and target sets.

To establish connectivity, do the following:

1. Choose Design – Run Design.

The Process Options Editor dialog box appears. By default, all the instances are selected.

2. Click Run to run the design.

After run process completes, the connectivity is established between target sets and DUT.

## Re-Optimizing Connectivity

Once connectivity is established, there may be a chance that you need to update the connectivity of a specific portion in the design. You can re-optimize the connectivity of a specific design section without disturbing the connectivity of the complete design.

You can re-optimize the connectivity using one of the following methods:

### ***Method 1***

1. Right-click on the DUT and choose Re-Run.
2. The following options appear:
  - All Interface Pins: Select to re-run the complete DUT.
  - Group Wise: Choose <group\_name> - <Group set name> or <connector> to re-run the selected target set or connector.
  - Segment Wise: Choose <Segment Name> - <target set name> and select an option.

### ***Method 2***

1. Right-click on a net or select a set of nets on canvas and choose Net – Minimize Crossovers on Selected Nets.  
The Select Device Instances to optimize form appears.
2. Click and select an instance for which connectivity needs to be re-optimized.

### ***Method 3***

You can also re-optimize the connectivity in Allegro PCB Editor. The FSP engine running in the background in Allegro PCB Editor guides you through the valid pin swaps that conform to FPGA I/O DRCs. Pin swaps (re-optimization) minimize the length of rats and crossovers.

**Note:** Pin swap or pin optimization across component is not supported in Allegro-FSP integration.

## **Allegro FPGA System Planner User Guide**

### Preparing Components for Connection

---

For more information, see the FSP – Allegro Integration Flow in the Allegro Design Entry HDL – FPGA System Planner flow guide.

# **Allegro FPGA System Planner User Guide**

## Preparing Components for Connection

---

---

## **Working with Nets and Ports**

---

This chapter contains the following sections:

- [About the Nets and Connectivity](#)
- [Net Naming Conventions](#)
- [Assigning Net Name Template](#)
- [Specifying Net Names Manually](#)
- [Click Yes to remove and No to cancel the operation.](#)
- [Modifying Net Names](#)
- [Re-routing Nets](#)
- [Deleting Nets](#)
- [Locking and Unlocking Nets](#)
- [Showing and Hiding Nets](#)
- [Using NetGroups](#)
- [Using RTL Port Names](#)

## About the Nets and Connectivity

A net represents an electrical connection between from one pin of an instance to the another pin(s) of different instance.

It is important to identify each inputs and outputs of the instances and a net between them with a name. FSP lets you assign a name for net(s). Net with a name helps you to indentify a net in the design. Each net must have a unique name.

A net name also let you enter following information:

- Bits

Nets can have a single bit or multiple bits. The bit portion of the net is always enclosed in the angle brackets, such as <5..0>.

A net with a bracket can be a scalar or a vectored net. A net without a bracket is caller scalar.

- Properties

To convey the physical and instance information.

The names you assign to the nets in the design are written into the design net list and are generated in DE-HDL and OrCAD schematics.

## About Bus Name

A bus is a group of scalar nets. You can group nets with similar funtionality to form a bus. Any constraints applied to a bus are inherited by all the nets of the bus.

## Net Naming Conventions

Net names must follow the following naming conventions:

- Net names can use letters, numbers, and the following valid characters:

- underscore (\_)
  - dot (.)
  - hash (#)
  - dollar (\$)

- percentile (%)
  - plus (+)
  - apostrophe (`)
  - dash (-)
  - plus (+)
- Net names can have greater than 255 characters.
  - FSP is case-sensitive. FSP treats the two net names that differ only in uppercase and lowercase as different name.
  - Vector net names, can use the following brackets: <> {} () [].

**Note:** To treat the following brackets as vector you need to specify the brackets as bus notation in config.ini file located in %CDSROOT%/tools/fsp/bin. For example, to use [] brackets as vector, open the config.ini file in a text editor and specify as bus\_notation = "[]".

- The following characters must not be used in net names:
  - White space
  - Backslash (\)
  - Forward slash (/)
  - Exclamation mark (!)
  - At sign (@)
  - Left single quote (')
  - Tilde (~)
  - comma (,)

## Assigning Net Name Template

FSP lets you plan the names for the nets at early stage of the design cycle. You can define a template for the nets before synthesizing the design. A template contains set of attributes, instance information, and aplha-numeric characters that are applied to each net during synthesis. You can apply the net names using the *Net Name Template* feature in the *Settings* dialog box. The *Net Name Template* feature lets you quickly define and edit the names for hundreds of interface and device protocol nets at one go. The *Net Name Template* feature provides few pre-defined attributes such as instance names, pin names, pin number and

more to include in the template. You also have the option to include alpha-numeric characters along with the pre-defined attributes.

After defining the template for the net names, you need to run the design in order to apply the template on complete design nets.

**Note:** In case you modify the template after running the design, then it is recommended that you rerun the design to apply the modified template on the nets.

## Assigning Net Name Template on Interface Protocol

To define a template for the interface protocol, perform the following steps:

1. Choose *File – Settings*.

The Settings dialog box is displayed.

2. Click *Net Name Template*.

3. Specify the fields value as required.

**Note:** For detailed information on the buttons and fields see [Change Net Names](#) section.

4. Click *Default* to apply default template.

5. Click *OK*.

After running the design, the defined net name templates is applied to interface protocol nets.

## Define Net Name Template for Interface Protocol Examples

The following examples shows how to define a net name template for interface protocols:

- To start a interface net name with *device pin name* and end with *device pin number*. Example, `IO_DQS2T_G10`. `IO_DQS2T` is the device pin name and `G10` is device pin number.
  - Click and select *Start Pin Name* from first drop-down list.
  - Select *Start Pin Number* from second drop-down list.
- To start a interface net name with an alphabet followed by interface pin name and end with special character (`_`) with interface name such as `MDDR2_CLK1_P_U10`. `M` is the alphabet, `DDR2_CLK1` is interface pin name and `U10` is the interface instance name.
  - Enter an alphabet in first text box such as `M, N, A` so on.

- Click and select *End Pin Name* from first drop-down list.
- Enter an special character in second text box as “\_”.
- Click and select *End Instance Name* from second drop-down list.

## Assigning Net Name Template on Device Protocol

To define a net name template for device protocol, perform the following steps:

1. Choose *File – Settings*.  
The Settings dialog box is displayed.
2. Click *Net Name Template*.
3. Click *Protocol*.
4. Specify the field values as required.
5. Click *Default* to apply default net name templates.

**Note:** The following template is applied on the device protocol by default:

<Protocol\_Name>\_<Pin\_Name>.

6. Click *OK*.

After running the design, the defined net name templates is applied to device protocol nets.

## Define Net Name Template for Device Protocol Examples

The following examples shows how to define a net name template for device protocol:

- To define a template for a device protocol between four FPGAs (U1, U2, U3, and U4), perform the following:
  - Select the *Connected Instances* option from first drop-down list.
  - Select the *Protocol Signal Name* option from second drop-down list.

After running the design, you see the following device protocol names:

U1\_U2\_U3\_U4\_Data<0>

U1\_U2\_U3\_U4\_Data<1>

U1\_U2\_U3\_U4\_Data<2>

... more.

- To include the FPGA port name followed by the last device instance name U4, perform the following:
  - Select the *FPGA Port Name* option from first drop-down list.
  - Select the *End Instance Name* option from second drop-down list.

## Specifying Net Names Manually

Sometimes it is a common practice that you may opt to specify net names of your own choice instead of specifying through the net name template feature. FSP provides you the liberty to specify net names of your own choice. You can either specify the net names manually before running the design or else you can modify over the net names that are generated after running the design. The net names that are manually specified by you have precedence over the net names generated by FSP.

The below scenario explains the usage of manually specifying the net names.

After running the design, the net names for the signals are auto-generated and displayed in the *Net Name* column of the Design Connectivity. These net names are displayed in basic text as depicted below.

### Before deleting nets

Status	Pin/Port Name	n Type	Net Name	Pin Termination	FPGA Ext
Allocated	DDR2_DQ0		Data_Byte1 target_device=U2		
Allocated	DDR2_DQS0_N		U2_XP78_DDR2_DQS0_N		
Allocated	DDR2_DQS0_P		U2_XP78_DDR2_DQS0_P		
Allocated	DDR2_DQ<0>		U2_XP78_DDR2_DQ<0>		
Allocated	DDR2_DQ<1>		U2_XP78_DDR2_DQ<1>		
Allocated	DDR2_DQ<2>		U2_XP78_DDR2_DQ<2>		
Allocated	DDR2_DQ<3>		U2_XP78_DDR2_DQ<3>		
Allocated	DDR2_DQ<4>		U2_XP78_DDR2_DQ<4>		
Allocated	DDR2_DQ<5>		U2_XP78_DDR2_DQ<5>		
Allocated	DDR2_DQ<6>		U2_XP78_DDR2_DQ<6>		
Allocated	DDR2_DQ<7>		U2_XP78_DDR2_DQ<7>		
Allocated	DDR2_A<0>		U2_XP78_DDR2_A<0>		
Allocated	DDR2_A<1>		U2_XP78_DDR2_A<1>		
Allocated	DDR2_A<2>		U2_XP78_DDR2_A<2>		
Allocated	DDR2_A<3>		U2_XP78_DDR2_A<3>		
Allocated	DDR2_A<4>		U2_XP78_DDR2_A<4>		
Allocated	DDR2_A<5>		U2_XP78_DDR2_A<5>		
Allocated	DDR2_A<6>		U2_XP78_DDR2_A<6>		
Allocated	DDR2_A<7>		U2_XP78_DDR2_A<7>		
Allocated	DDR2_A<8>		U2_XP78_DDR2_A<8>		
Allocated	DDR2_A<9>		U2_XP78_DDR2_A<9>		

When you delete the design nets, these net names are displayed in italic text.

# Allegro FPGA System Planner User Guide

## Working with Nets and Ports

### After deleting nets

Status	Pin/Port Name	n Type	Net Name
group group_constraint= <b>same_clock_region</b> group_name= <b>Data_Byte1</b> target_device=U2			
Unallocated	DDR2_DM0		<i>U2_XP78_DDR2_DM0</i>
Unallocated	DDR2_DQS0_N		<i>U2_XP78_DDR2_DQS0_N</i>
Unallocated	DDR2_DQS0_P		<i>U2_XP78_DDR2_DQS0_P</i>
Unallocated	DDR2_DQ<0>		<i>U2_XP78_DDR2_DQ&lt;0&gt;</i>
Unallocated	DDR2_DQ<1>		<i>U2_XP78_DDR2_DQ&lt;1&gt;</i>
Unallocated	DDR2_DQ<2>		<i>U2_XP78_DDR2_DQ&lt;2&gt;</i>
Unallocated	DDR2_DQ<3>		<i>U2_XP78_DDR2_DQ&lt;3&gt;</i>
Unallocated	DDR2_DQ<4>		<i>U2_XP78_DDR2_DQ&lt;4&gt;</i>
Unallocated	DDR2_DQ<5>		<i>U2_XP78_DDR2_DQ&lt;5&gt;</i>
Unallocated	DDR2_DQ<6>		<i>U2_XP78_DDR2_DQ&lt;6&gt;</i>
Unallocated	DDR2_DQ<7>		<i>U2_XP78_DDR2_DQ&lt;7&gt;</i>
group group_name= <b>Address_Control</b> target_device=U2			
Unallocated	DDR2_A<0>		<i>U2_XP78_DDR2_A&lt;0&gt;</i>
Unallocated	DDR2_A<1>		<i>U2_XP78_DDR2_A&lt;1&gt;</i>
Unallocated	DDR2_A<2>		<i>U2_XP78_DDR2_A&lt;2&gt;</i>
Unallocated	DDR2_A<3>		<i>U2_XP78_DDR2_A&lt;3&gt;</i>
Unallocated	DDR2_A<4>		<i>U2_XP78_DDR2_A&lt;4&gt;</i>
Unallocated	DDR2_A<5>		<i>U2_XP78_DDR2_A&lt;5&gt;</i>
Unallocated	DDR2_A<6>		<i>U2_XP78_DDR2_A&lt;6&gt;</i>
Unallocated	DDR2_A<7>		<i>U2_XP78_DDR2_A&lt;7&gt;</i>
Unallocated	DDR2_A<8>		<i>U2_XP78_DDR2_A&lt;8&gt;</i>
Unallocated	DDR2_A<9>		<i>U2_XP78_DDR2_A&lt;9&gt;</i>
Unallocated	DDR2_A<10>		<i>U2_XP78_DDR2_A&lt;10&gt;</i>
Unallocated	DDR2_A<11>		<i>U2_XP78_DDR2_A&lt;11&gt;</i>
Unallocated	DDR2_A<12>		<i>U2_XP78_DDR2_A&lt;12&gt;</i>
Unallocated	DDR2_A<13>		<i>U2_XP78_DDR2_A&lt;13&gt;</i>
Unallocated	DDR2_A<14>		<i>U2_XP78_DDR2_A&lt;14&gt;</i>

The italic texts indicates that the net names are preserved and will be used in the next design run. However, in-between if you change the template of the net names, the new names will override the existing net names. The new net names is displayed in basic text.

In other scenario, when you manually modify the existing net names or enter the net names in the *Net Name* column, the net names are displayed in italic text. From thereafter, FSP will prefer the manually modified net names over the net names generated by the net name template feature. This means, if you delete the design nets, modify the net name template, and rerun the design, FSP will continue to use the modified net names.

FSP will continue to use the modified net names as design net names until you remove them. You can remove the manually modified net names with a single-click option.

To specify net names manually, perform the following steps:

**Note:** You can perform the following steps before or after running the design.

1. Invoke the *Design Connectivity* by choosing *Windows – Design Connectivity* or press Ctrl + 4.

The Design Connectivity is displayed. The net names are displayed both under the *Net Name* column.

2. Under the *Net Name* column, click on a cell and modify the net name.

## Resetting Net Names

FSP provides you a single-click option to remove the complete design net names, that is, net names that are auto-generated by FSP and the net names that are manually specified by you. However, to remove the net names, first you should delete the nets from the design.

To reset the net names, perform the following steps:

1. Choose *Design – Delete All Nets*.

The *Remove Net Confirmation* dialog box is displayed.

2. Click *OK* to remove the nets.

Or

1. Choose *Design – Reset Cached Net Names*.

A confirmation window is displayed prompting you about the removal of net names.

2. Click *Yes* to remove and *No* to cancel the operation.

## Modifying Net Names

You can modify the net names manually, re-specify the net name template, or by replacing the net name pattern.

The following table lists the form names and the actions you can perform to modify the net names.

Dialog box name	Process
Settings	Using this form you can modify the net names for complete design.

## Allegro FPGA System Planner User Guide

### Working with Nets and Ports

---

#### Change Signal Form

1. Choose *Design – Change Net Names* or click Change Net Names icon in tool bar.

The Change Net Names dialog box is displayed.

2. In Change Net Name column, double click on any cell, and enter a new name.

**Note:** The vector signals `<net_name><0..10>` will be changed to `<new_net_name><0...10>`.

3. To redefine the net name template, click *Define Net Name*.

The Net Name Template pane is displayed at bottom of the window.

4. Specify new template for net names as required.

5. Click *Find and Replace* icon or press *Ctrl + H*, to replace the pattern of all the net names at one step.

The Find and Replace dialog box is displayed.

6. Specify the pattern to be changed in Find What field.

7. Specify the new pattern to be applied in Replace Pattern field.

8. Click *OK* of Find and Replace dialog box.

9. Click *OK*.

**Note:** If the net names are changed manually and design is re-run, the modified net names are changed to previous ones. The net names defined by the Net Name Template feature are unchanged. For more information on fields and buttons of the Change Net Names dialog box see [Change Net Names](#) section.

The net names are changed based on the specified in Change Net Name template.

#### Design Connectivity

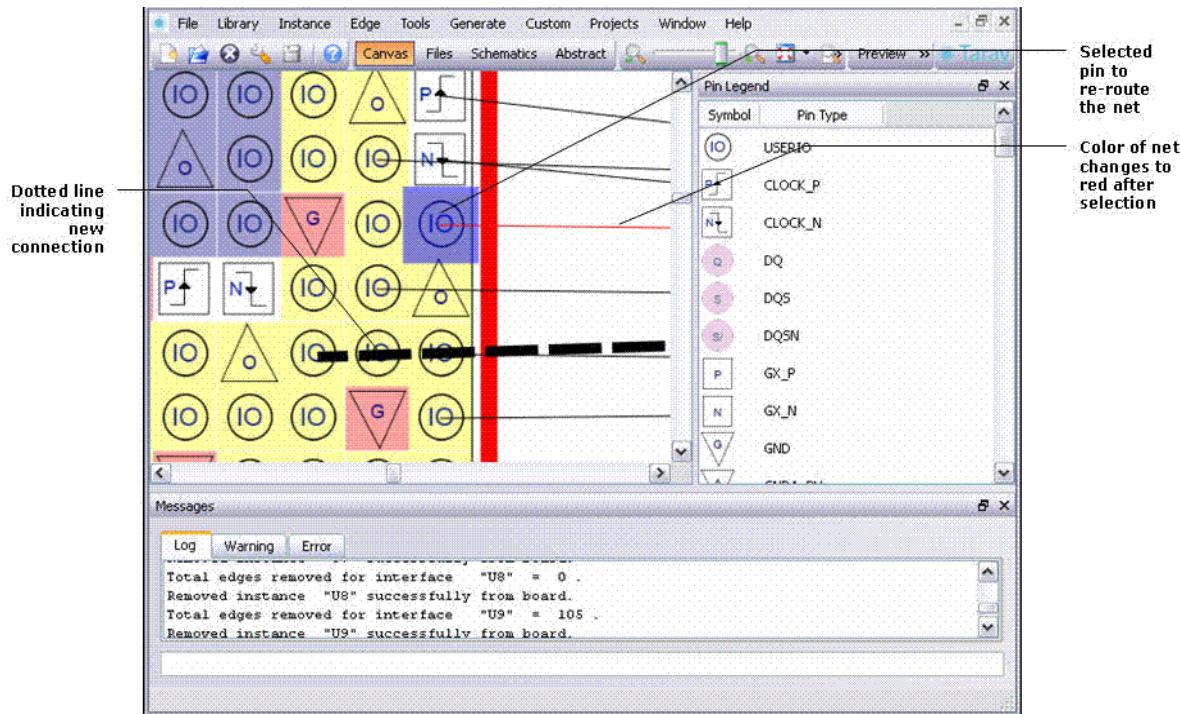
In the *Net Name* column, click on a cell and enter/modify the net name.

## Re-routing Nets

The Re-routing Nets feature allows you to manually change the path of nets. This feature is available when you select a pin of a device instance that has a net (connection). While rerouting the nets, first the design rules is checked by FSP and then re-routing is allowed to the suitable pins.

Re-routing is done based on the Group and Pin constraints defined to the interface pin. The listed are some of the cases you need to remember while re-routing:

- Interface pin (net) with Same Clock Region defined as group constraint can be re-routed within the Same Clock Region only.
- Interface pin (net) with Same Bank as group constraint can be re-routed within the same bank.
- Interface pin (net) with CLK, DCLK, CDPCLK defined as pin constraints can be rerouted only to those FPGA pins to whom they are targeted.
- Positive pin of a differential pair can be only be routed with a positive pin of another differential pair.



1. Select a device pin on canvas, whose net you want to re-route.
2. Right-click on the device instance and select *Re-Assign Net on pin <device\_pin\_name>* option from pop-up menu.  
The net is displayed in dotted line indicating that now you are ready to re-route the net with another net.
3. Double click on another pin of the same device.

If the new pin is not suitable for re-routing, an error message is displayed in Log window and if the pin is suitable a confirmation message window is displayed prompting you about the swapping of nets.

## Deleting Nets

To delete nets in the design, perform the following steps:

1. Choose *Design – Delete All Nets* or click *Clear All Edges* icon in .

The Remove Net Confirmation dialog box is displayed with two options selected by default.

2. Click *OK* to remove the nets.

After you click *OK*, all unlocked nets are removed in the design.

**Note:** Locked nets cannot be deleted using this option. You need to first unlock them and then use this option to delete the nets.

## Locking and Unlocking Nets

FSP enables you to lock and unlock nets. You can lock and unlock all the nets or selected nets of the design. Locked nets cannot be removed, re-routed, swapped with other locked and unlocked nets. In order to perform any operations on the locked nets you need to first unlock them. By default, a locked net is displayed in yellow color. You can customize the color of the locked nets through preference window.

Locking nets details can be stored in a text file. The Lock Nets dialog box enables you to export the lock nets details to text file. This text file can be utilized further to lock the unlock nets by importing it into Lock Nets dialog box. However before importing you must ensure that the net names in Lock Nets dialog box and text file should match.

Locking/Unlocking nets operation depends on the following items:

## ■ Group Constraints

Consider a 8-bit data (U1\_Data<0>....<7>) group known as *Data\_Byte1* with group constraint as *Same Clock Region*. When you select *U1\_Data<0>* net name then the remaining U1\_data<1>...<7> net names are automatically locked.

The above rule is applied to all the group constraints except *Any Bank*.

## ■ Differential Pair Type

When you select a P type pin of a differential pair for lock then automatically the N type pin gets locked.

## ■ High Speed Serial IOs

Nets can be locked and unlocked using the following methods:

- Using the Pop-up menu options in the canvas
- Using the Pop-up menu option in the Design Connectivity
- Using the Lock Nets dialog box

## Using the Pop-up menu options in the canvas

To lock nets of an instance,

- Choose *Right-click on Instance – Instance Nets – Lock All Nets*.

All the nets of the selected instance are Locked.

**Note:** Locked nets are displayed in yellow color.

To unlock nets of an instance,

- Choose *Right-click on Instance – Instance Nets – Unlock All Nets*.

All the nets of the selected instance are unlocked.

## Using the Pop-up menu option in the Design Connectivity

To lock the entire design nets, perform any one of the following:

- Right-click on the design-level row and choose *Nets – Lock*.

Or

- a. Click on the design-level row.
- b. Click on the *Design* button and choose *Nets – Lock*.

**Note:** Choose *Nets – Unlock* to unlock the nets.

To lock the nets of a specific device, perform any one of the following:

- Right-click on the device-level row and choose *Nets – Lock*.

Or

- a. Click on the device-level row.
- b. Click on the *Device:<inst\_name>* button and choose *Nets – Lock*.

**Note:** Choose *Nets – Unlock* to unlock the nets.

To lock the nets of a specific bank in a device, perform any one of the following:

- Right-click on the bank-level row and choose *Nets – Lock*.

Or

- a. Click on the bank-level row.
- b. Click on the *Bank:<inst\_name:bank\_name>* button and choose *Nets – Lock*.

**Note:** Choose *Nets – Unlock* to unlock the nets.

To lock the nets of a interface instance, perform any one of the following:

- Right-click on the interface-level row and choose *Nets – Lock*.

Or

- a. Click on the interface-level row.
- b. Click on the *interface:<inst\_name:bank\_name>* button and choose *Nets – Lock*.

**Note:** Choose *Nets – Unlock* to unlock the nets.

To lock the nets of a specific group in an interface, perform any one of the following:

- Right-click on the group-level row and choose *Nets – Lock*.

Or

- a. Click on the group-level row.
- b. Click on the *Group:<inst\_name:group\_name>* button and choose *Nets – Lock*.

**Note:** Choose *Nets – Unlock* to unlock the nets.

## Using the Lock Nets dialog box

The Lock Nets dialog box provides you much more option than expected. Using Lock Nets dialog box you can:

- Lock All Nets
- Unlock All Nets
- Lock Clocks
- Unlock Clocks
- Lock Selected
- Unlock Selected
- Lock Constraints Pins
- Unlock Constraint Pins
- Import Nets
- Export Nets

**Note:** For detailed information on above Lock Nets dialog box options see [Lock Nets](#) section.

To perform the various locking operations on nets complete the following steps:

1. Choose *Design – Lock Nets*.

The Lock Nets dialog box is displayed.

2. Do the following as per required:

a. Click *Lock Nets* icon to lock all the nets.

The locked nets are displayed in yellow color.

b. Click *Unlock Nets* icon to unlock all the nets.

The yellow colored nets are displayed in default color.

c. Select the clock names from the displayed list and click *Lock Clocks* icon to lock clock nets.

The locked clock nets are displayed in dark yellow color.

- d. Select the clock names from the displayed list and click *Unlock Clocks* icon to unlock clock nets.

The locked clock nets are displayed in default color.

- e. Select the net names from the displayed list and click *Lock Selected* icon to lock selected nets.

If the selected net has group constraint or is a differential pair or high speed serial IOs, the rest of the nets associated with the selected net will automatically get locked. For example,

If the selected net has group constraint *Same Bank*, then all the remaining nets from the same group will automatically get locked.

The Lock Nets Confirmation dialog box confirmation dialog box is displayed when you click *Lock Selected* icon. Click *Yes* to lock all the nets displayed in confirmation window or click *No* to cancel the operation.

- f. Select the net names from the displayed list and click *Unlock Selected* icon to lock selected nets.

The *Unlock Selected* option is very much similar to *Lock Selected* option. If the selected net has group constraint or is a differential pair or a high speed serial IO, the rest of the nets associated with the selected net will automatically get unlocked.

For example, if the selected net has group constraint *Same Bank*, then all the remaining nets from the same group will automatically get unlocked.

The Lock Nets Confirmation dialog box confirmation dialog box is displayed when you click *Unlock Selected* icon. Click *Yes* to lock all the nets displayed in confirmation window or click *No* to cancel the operation.

- g. Select the constraint nets from the displayed list and click *Lock Constraint Pins* to lock the constraint pins.

The locked constraint pins are displayed in dark cyan color.

- h. Select the constraint nets displayed in dark cyan color and click *Unlock Constraint Pins* to unlock the constraint pins.

The dark cyan colored locked constraint pins are displayed in normal color.

3. Click *Import Nets* icon to import net details from text file.

The Import Locked Nets dialog box is displayed. Browse and select the text file and click *Open*. The nets in the list are locked and displayed in yellow color.

4. Click *Export Nets* icon to export the locked nets to text file.

**Note:** Before clicking this option you must lock the nets and then click *Export Nets*.

The Export Locked Nets dialog box is displayed. Browse for the directory where you want to save the text file and enter the text file name and click *Save*.

5. Click *Close* to exit the dialog box.

## Showing and Hiding Nets

To hide and display the nets in the canvas, perform the following steps:

To hide nets of an instance,

- Choose *Right-click on Instance – Instance Nets – Hide All Nets*.

All the nets of the selected instance are hide.

To show nets of an instance,

- Choose *Right-click on Instance – Instance Nets – Show All Nets*.

All the nets of the selected instance are displayed.

**Note:** You can also hide the nets through the Design Connectivity. For more information, see the [Context-Sensitive Menus](#) section.

## Using NetGroups

A NetGroup is a group of nets that you treat as a single entity in FSP. The nets in a NetGroup can be a scalar, vector, or a combination of both. However, a net can belong to only one group at a time.

By definition, a NetGroup is a completely heterogeneous collection of nets. Unlike a bus, which is homogeneous collection of nets (scalar or vector), a NetGroup provides a greater flexibility in grouping nets together.



NetGroup does not overrides the functionality and value of a bus.

The main functionality of NetGroup is to group together a heterogeneous group of signals. These NetGroups are propagated via schematics to Allegro PCB Editor. In Allegro PCB Editor, rats are combined into bundles based on the NetGroup definitions. Bundles allow you to associate multiple connections and manipulate them as a single entity within the design.

These bundles enhance your visual understanding of the routing strategy for complex designs in Allegro PCB Editor. In Allegro PCB Editor, you can perform the *Auto Pin Swap* optimization on the bundles to optimize the crossovers across the edges of the components.

**Note:** The steps for creating bundles is not covered in this section.

## Naming NetGroups

In FPGA System Planner, nets are allocated together based on the interface logical groups. FSP lets you create NetGroups using these interface logical groups. NetGroups can be defined automatically or manually. However, automatic method is recommended over the manual method, as manual method is a tedious task. It requires a lot of efforts to define NetGroups for hundreds of nets. You can also combine automatic and manual methods to define NetGroups. For instance, you first choose for auto-create NetGroup option and then edit some of the NetGroups manually.

To set up your design for NetGroups, perform the following steps:

1. Choose File – Settings.

The Settings dialog box is displayed.

2. Click the NetGroups tab.

3. Select first option to auto create NetGroups on device protocol signals.

When selected, the NetGroups are automatically created on device protocols, when no NetGroups are existing.

4. Select second option to auto create NetGroups for interface signals.

When selected, the NetGroups are automatically created on interfaces based on their logical group names.

5. Specify the maximum number of nets allowable in a group.

For example, for a 64-bit size of interface group U2.Data<0>....<63>, select the first option and specify the NetGroup size as 16. The NetGroups for the interface group nets are automatically defined as.

- U2.Data1<0>...<15>
- U2.Data2<16>...<30>
- U2.Data3<31>...<45>
- U2.Data4<46>...<63>

**6.** Click OK to save the settings.

When you place the interface components on canvas or create device protocol, NetGroups are automatically created based on the specified parameters.

By default, FSP creates the NetGroups using the template <interface instance\_name>\_<interface group\_name>. However, FSP provides you few options using which you can change the naming conventions of NetGroups. You can either use the *Design – Create NetGroups* menu or use the Context sensitive menu options available in the *Design Connectivity*.

## Using RTL Port Names

Port names are considered as ports for each device pins in the design. These port names are generated as ports in constraint files and top design level (.verilog). The port names in top design file and constraint files are used to synthesis the design in other FPGA tools. The port names are useful while importing and exporting constraints and also port mapping.

A port name can be of any name. It can be equal to device pin names or net names. If you do not specify the port names for the device, the net names are automatically used as port names while generating constraint files. In FSP, you can define the port names by manually, mapping port names with net names or by importing constraint files.

### Specifying Port Names for FSP Nets

This section describes the steps to add RTL port names in FSP.

#### Specifying RTL Port Names for Interface Protocol

To define RTL port names for the interface signals:

- In the *RTL Port Name* column in Design Connectivity, click on any cell and specify port names.

When you generate constraint files the specified RTL port names are generated as port in the files.

#### Specifying RTL Port Names for Device Protocol

To define RTL port names for device signals, perform the following steps:

1. Right-click on the device instance and choose *Protocol – Manage Protocol*.  
The Manage Protocol dialog box is displayed.
2. Select an existing protocol name from the list and click *Edit*.
3. In the *RTL Port Name* column, click on any cell and specify port names.
4. Click *OK*.

The defined RTL port names are saved as ports when you generate the constraints file.

### **Specifying RTL Port Names for Virtual Interface Nets**

To define RTL port names for virtual interface nets perform the following steps:

1. Right-click on the device instance and choose *Virtual Interface – Create New Virtual Interface*.  
The Define Virtual Interface for Device Instance dialog is displayed.
2. In the *RTL Port Name* column, click on any cell and specify the port names.
3. Click *OK*.

The defined RTL port names are used as ports in constraint files.

## **Allegro FPGA System Planner User Guide**

### Working with Nets and Ports

---

---

## Running Synthesis

---

After specifying the targets and defining the net name template, you can make connections between the interface and device instances. FSP provides a synthesis environment that allows multiple synthesis attempts using different types of run command options and constraints.

You can create and save synthesis details for complete design or for each instances. Later, these synthesis details can be applied (imported) while running synthesis.

You can also monitor progress, view log reports, and quickly identify and modify the synthesis details and rerun the synthesis.

During synthesis, FSP considers the following constraints to reduce net crossovers and the number of board layers:

- **FPGA I/O DRCs (Design Rules Check)**

This refers to the FPGA's I/O pin usage rules. For example, how each pin must be used and how that usage influences other pins in the FPGA. Moreover, FPGA's banking structures also play a vital role. For example, connecting two I/O standards with same voltage levels within a bank or exceeding current drains by switching too many signals at one time. These constraints are specified in the FPGA rules files (.frf).

- **Logical Constraints**

These rules are defined by you during the process of specifying the signal characteristics in the Rules Editor form. You also define few constraints at instance level, that is, while targeting the interfaces. These rules adhere the memory logic rules during synthesis. For example, in a Xilinx Virtex 5 device architecture, you will require that the clock pin of the memory interface connects to a clock pin (CC) in the FPGA. Also, the corresponding data pins of the interface group should connect within the same clock region.

- **Physical Constraints**

Physical constraints are derived automatically based on the position of the FPGAs and interfaces on the Canvas. The number of wire crossings (also termed as rats nest) on the Canvas is affected by these constraints. Also, pin selection without consideration to the physical constraints can increase the number of layers required to route the board.

This chapter covers the following topics:

Running Synthesis for Complete Design

Running Synthesis for Instances (Incremental)

Configuring and Monitoring Synthesis Options

Saving and Importing Synthesis Details

## **Running Synthesis for Complete Design**

You can run synthesis for complete design using *Process Options Editor*.

To run synthesis for complete design, perform the following steps:

1. Choose *Design – Run Design* or press **Ctrl + R**, or click the *Run Design* icon in the .  
The Process Options Editor dialog box appears. By default, all the instances and device protocols are selected.
2. Click *Run* to start synthesis.

Once you click *Run*, the status and results of the synthesis is continuously displayed in the *Message* window. If any error occurs during synthesis, errors and warning messages are reported in the *Message* window.

## **Incremental Synthesis**

If the design is expected to change often, consider an incremental approach to synthesis. In an incremental approach, individual instance(s) can be synthesized separately without affecting the rest of the design. The advantages of *incremental synthesis* are that they reduce tool runtime, eliminate re-verification of unchanged instances, and help achieve better ratsnets.

When you launch the *Process Options Editor* window, you will notice that all the instances are selected by default. This indicates that synthesis will be run on all the instances. However, you can exclude the instances from Synthesis process that you do not want to synthesis. Besides excluding the instances, you can also excludes the groups of interfaces and device protocol groups of multiple devices.

To run synthesis on the individual instance(s)(incremental), do the following steps:

1. Do any one of the following:

- Choose *Design – Run Design*.
- Click *Run Design* icon in the .
- Press Ctrl + R.

The *Process Options Editor* dialog box appears. You can browse through this window to find out how many instances are present on the Canvas and which device or interface you want to synthesis.

**Note:** It is assumed that the instances are selected by default.

2. Expand the <device\_inst\_name>[part name] branch by clicking the > button to the left of the <device\_inst\_name>[part name].
3. Uncheck the check box to the left of the <device\_inst\_name>[part name] to exclude the device and it's corresponding interface(s) or device protocol from Synthesis process.

Notice that the check boxes of it's corresponding interface(s) or device protocol are also unselected.

**Note:** If the <device\_inst\_name> is connected to interface(s), you will notice <interface\_inst\_name>[part name] branch or <device\_protocol\_name>[PROTOCOL] if the device is connected to multiprotocol chain.

4. Expand the <interface\_inst\_name>[part name] branch by clicking the > button to the left of the <interface\_inst\_name> [part name].
5. Uncheck the check box to the left of the <interface\_inst\_name>[part name] to exclude the interface and it's corresponding groups from Synthesis process.

The groups of the interface appear. Notice that the check boxes of it's correspoding groups are also unselected. You can also uncheck the groups to exclude the groups in synthesis process.

6. Select the check box to the left of the <group\_name> to exclude the groups.
7. Click *Run* to run the design.

## Configuring and Monitoring Synthesis

This section covers the following topics:

FSP provides you various synthesis options in *Process Options Editor* to meet your design/timing targets. These options are considered by FSP to optimize the pin assignments, reduce crossovers, and adhere FPGA I/O DRCs rules. These options can also be configured for

individual instances. FSP recommends that you apply these options before running synthesis. However, if you apply it after running, then you must run the design again.

## Applying Synthesis Options

The following synthesis options are available in *Process Options Editor*:

- Prioritizing the Processing Order

You can prioritize the order of processing instances. By default, the check boxes in the *Prioritize* column are unselected in *Process Options Editor*. This indicates that FSP will process the instances in an order they appear in *Process Options Editor*.

<<<You can also specify the instances that you want to process first during synthesis. only those corresponding to the device names, this indicates that FSP will process the instances in the order they are listed.>>>>

In the following example FSP will run the U8 instance first and then U10 instance.

Instance/Protocol Name	Prioritize
U6 [xc7v585tffg1157]	<input type="checkbox"/>
U7 [ddr2_sdram_x8_sd_60bqa_v7]	<input type="checkbox"/>
<input checked="" type="checkbox"/> U8 [5sgxeahf35]	<input checked="" type="checkbox"/>
U9 [cy7c1515v18_s5gx]	<input type="checkbox"/>
<input checked="" type="checkbox"/> Address_Control	<input type="checkbox"/>
<input checked="" type="checkbox"/> Data	<input type="checkbox"/>
<input checked="" type="checkbox"/> Data_write	<input type="checkbox"/>
<input checked="" type="checkbox"/> U10 [cy7c1515v18_s5gx]	<input type="checkbox"/>
<input checked="" type="checkbox"/> Address_Control	<input type="checkbox"/>
<input checked="" type="checkbox"/> Data	<input type="checkbox"/>
<input checked="" type="checkbox"/> Data_write	<input type="checkbox"/>

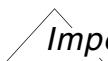
After selecting the *Prioritize* check box corresponding to the device names, you can also select the *Prioritize* check box corresponding to the interface names. In this case, FSP will first run the interfaces whose *Prioritize* check box are selected and then will run the remaining interfaces whose *Prioritize* check box are not selected.

You can also customize the priority order by moving the interfaces up and down. You can move the interfaces up and down by using options *Up* and *Down*. By default, these options are disabled. To enable, click on the interface name and then click *Up* or *Down*.

- Making Contiguous Connections

This feature adheres one of the FPGA I/O DRCs rules.

According to one of the Altera's IO DRC rules, the interface pins should connect to the pins whose tiles are adjacent to each other. In the following example, during synthesis FSP will use all the pins of tile 1 first and then move to tile 2, and vice versa.

 **Important**

This feature is restricted to single column, that is, signals of a bus should be connected to pins that belong to a single column. You can also allocate the signals to other columns as well; however, signals should have the valid group constraints. For example, to allocate address group, `addr[0:49]` with `same_clock_region` group constraint on die, FSP first allocate the forty signals in a single column of `same_clock_region`. After allocating forty signals, FSP will continue to allocate the remaining signals contiguously in the next column of `same_clock_region`.

You can select the *Contiguous* check box corresponding to the interface names or groups for which you want to enable contiguous connection.

In FSP, you use the *Die View* window to visualize the contiguous connections. In the *Die View* window, the I/O pins of Xilinx FPGAs are displayed in column wise and for Altera FPGAs the I/O pins are displayed into smaller rectangles, *tiles*. These pins and tiles are displayed based on the I/O pads information provided by the respective FPGA vendors.



## Applying Advanced Synthesis Options

After applying basic synthesis options, you can also apply various advanced synthesis options. When applied, all the synthesis options are considered together to optimize the FPGA pin assignments and reduce the crossovers in the design.

In this section:

### Prioritizing Interface Groups Connections

FSP helps you in prioritizing the groups connections based on the positions of device and interface instances on the Canvas. You can first use the banks for connection that are closer to the groups of the interface followed by the groups that are far from the device banks, or vice versa. The purpose of prioritizing the groups connections is to provide you a clean ratsnets.

The following example demonstrates the usefulness of prioritizing the interface group connections.

With Prioritize

<>>

Without Prioritize

<>>

### **Allocating Differential Pair Pins Together**

Sometimes You can properly utilize the differential pair pins for connection in a bank. <>>

### **Prioritizing Power and I/O Pins Connections**

Applying Constrained Optimization

## **Monitoring Synthesis Status**

After you click the *Run* option in *Process Options Editor*, the synthesis process starts. The status of the synthesis is continuously displayed in the *Message* window. Continuous display of commands and texts in the *Message* window indicates that the synthesis is running. Once synthesis completes, a detailed report of the design is displayed at the end. If any error occurs during synthesis, errors and warnings messages are reported in the *Synthesis Failure Report* tab. Errors are displayed in *red* and warnings in *blue*.

The synthesis and report messages are gradually saved in the `<project_name>.log` file located in the current project directory. You can also save the reports to any location on your system by right-click in the *Log* tab and choose *Save Log Messages*.

**Note:** For detailed information about the pop-up menu options in the *Message* window, see the [Context Menu Options](#) section.

## **Saving and Importing Synthesis Details**

This section covers the following topics:

Saving Synthesis Details

Importing Synthesis Details

## Saving Synthesis Details

Before running the synthesis, you can save the synthesis details. However, these synthesis details are saved internally and appear as a menu in *Load Process Options*.

To save the synthesis details, perform the following steps:

1. Click *Save As*.  
The *Save Process Options* dialog box appears.
2. Enter a name and click *Save*.
3. Click *Load Process Options* to see the name.

## Importing Synthesis Details

FSP enables you to import the saved synthesis details.



The import synthesis process will override the existing synthesis options in *Process Options Editor* with the values of the saved synthesis. It is recommended that you first save the existing synthesis options and then import the previous saved synthesis details.

To import the synthesis details, perform the following steps:

1. Click *Load Process Options*.
2. Select a name from the list.

A confirmation window appears prompting you about ignoring the changes made in the *Process Options Editor*.

3. Click *Yes* to import the synthesis details, or *No* to cancel the operation.

you can define advanced options of device and interface instances. FSP will then make the pin-out placement and connections according to the options you set.

In advanced options, you can try a second route for the connections to reduce the number of crossovers in your design. While making the connections, you can also define isolating high-speed serial I/O pins of Virtex devices. You can specify to connect the pins to have the least displacement between instances. Alternatively, you can specify to connect to the smallest fit bank available in an FPGA, or to connect to sorted groups of an interface. The sorted group can be nearest to FPGA or farthest from it.

FSP provides the flexibility to try other possible pin-out selection options and to choose the one best suitable so that you can complete a design within just a few hours.

Click the Advanced button next to an instance to define the advanced properties for that instance.



### Advanced Run Option for Devices



### Advanced Run Option for Interfaces

Once you click OK, the advanced properties are defined for the selected instance.

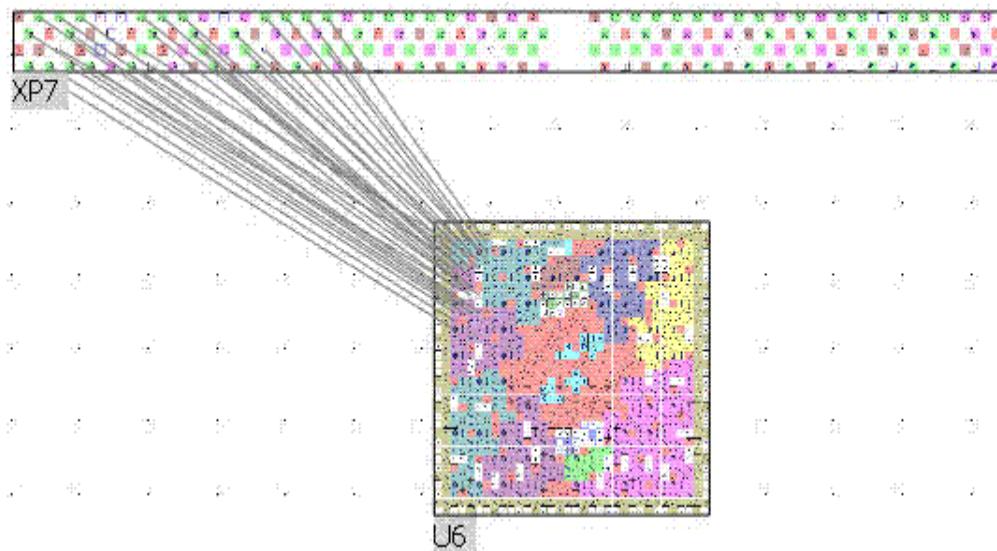
## Match Length for Group Nets

In the Match Length for the Group Nets process option, FSP connects all the groups defined with same bank or same clock region constraints to the banks that will have the least difference in net length...

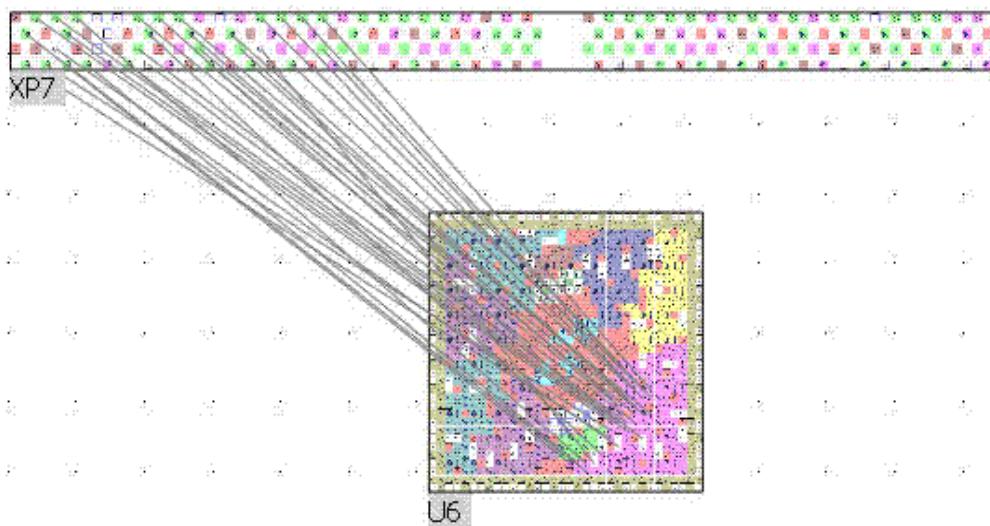
The following illustrations give you an idea of the Match Length for the Group Nets process option.

## Allegro FPGA System Planner User Guide

### Running Synthesis



Before selecting Match Length for Group Nets



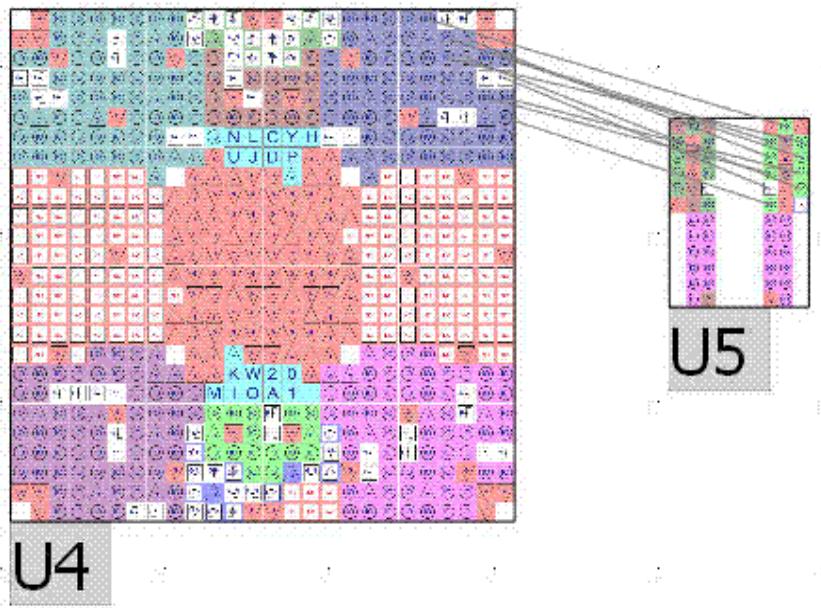
After selecting Match Length for Group Nets:

You can easily see the difference in net length in the newly selected bank. In the second image, it is reduced even though all the other constraints in the design are the same.

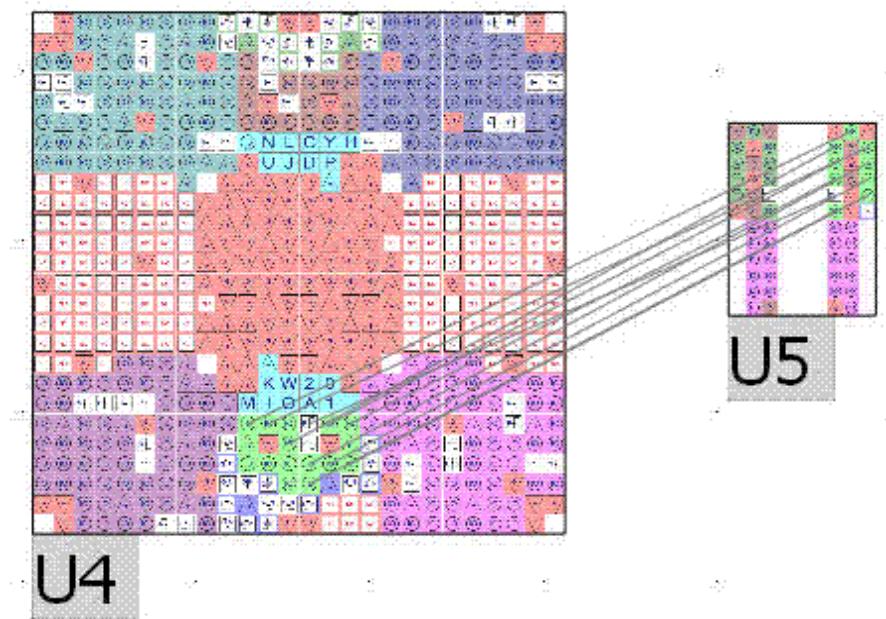
## Smallest Fit Bank

In the Smallest-Fit Bank process option, FSP connects all groups defined with same bank or same clock region constraint, to the smallest and nearest compatible bank in the selected device instance. This ensures freeing a larger number of pins at the device instance side.

The following illustrations give you an idea of the option in the Smallest-Fit Bank process.



Before selecting the Smallest Fit Bank



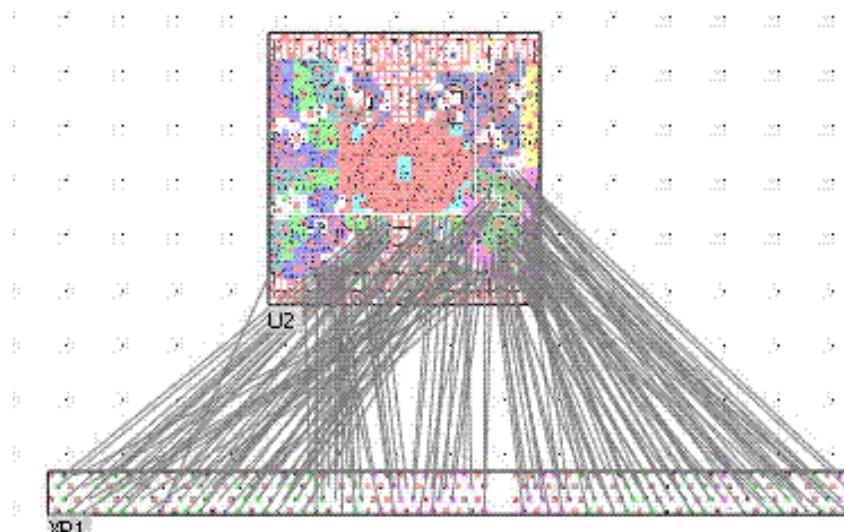
After selecting the Smallest Fit Bank

In the second figure, you can easily see that the pin connections are changed. They are based on the smallest and nearest compatible bank in the selected device instance.

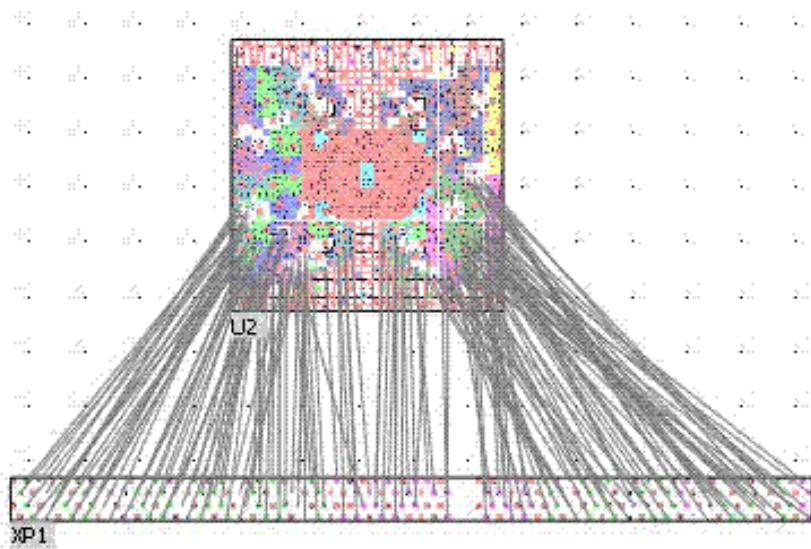
### Sorted Groups (Farthest First)

In the option process Sorted Groups (Farthest First), FSP sorts the groups farthest from the selected device instance but have the same bank or same clock region constraint. This process option then connects the sorted groups first before connecting other groups.

The following illustrations give you an idea of the Sorted Groups (Farthest First) process option.



Before selecting the Sorted Groups (Farthest First)



After selecting the Sorted Groups (Farthest First)

In the second figure, you can easily see that the pin connections are changed. They are based on the groups which are Farthest First in the selected interface instance.

### **Sorted Groups (Nearest First)**

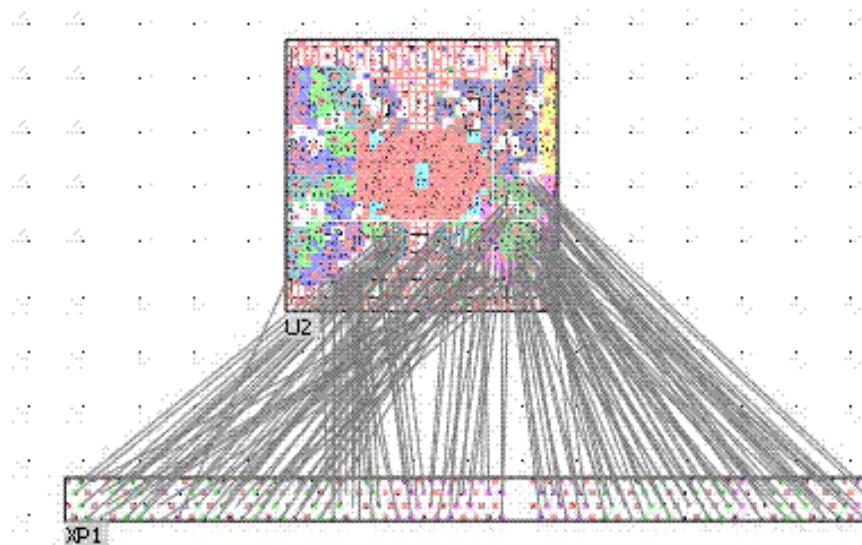
In the Sorted Groups (Nearest First) process option, FSP sorts the groups of the interfaces, having same bank or same clock region constraints, which are nearest from the selected

## Allegro FPGA System Planner User Guide

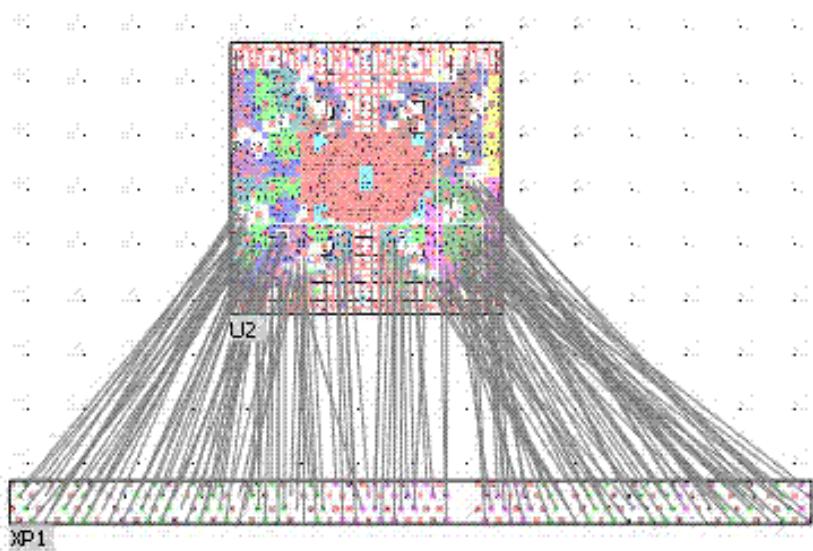
### Running Synthesis

device instance with the least number of crossovers. This process option then connects the sorted groups first, before connecting other groups.

The following illustrations give you an idea of the Sorted Groups (Nearest First) process option.



Before selecting the Sorted Groups (Nearest First)



After selecting the Sorted Groups (Nearest First)

In the second figure, you can easily see that the pin connections are changed. They are based on the groups which are Nearest First in the selected interface instance.

# **Allegro FPGA System Planner User Guide**

## Running Synthesis

---

---

## Working with Power Regulators

---

This chapter describes about working with power regulators and mapping them to device power pins in FSP. The following topics are covered in this chapter.

- [Overview to the Power connections](#)
- [Adding and Deleting Power Regulators](#)
- [Defining Power Mapping](#)
- [Mapping Power Regulators to Power Pins](#)
- [Adding Power Regulators Automatically](#)
- [Removing Power Regulators](#)
- [Resetting and Mapping Power Regulators](#)
- [Importing Power Mapping Information](#)
- [Exporting Power Mapping Information](#)

### Overview to the Power connections

Every FPGA contain set of power pins in each bank. However, these power pins require voltage of 3.3, 2.5, 1.8, 1.5, and 1.2V for the core, depending on the specific family of the FPGA's. Core Voltage/Power, I/O Voltage/Power, and Vaux Voltage/Power are the major voltage power requirements with which an FPGA is operated. When the design is complete, these pins must be connected to the associated voltages based on the recommendations provided by the FPGA family vendors. FSP provides you a feature with which you can decide and define voltage regulators and connect to the power pins of the FPGA.



Performing power connections before design synthesis process is not recommended since this may cause error while generating symbols and schematics. It is recommended that you perform power connections after synthesizing the design.

The following example demonstrates the need and usefulness of this feature.

Suppose in one of the FPGA bank, a HSTL\_I\_15 standard is connected. As per the design rules no power pins (Core voltage and Vaux voltage) in the FPGA bank should be left unconnected. You create two power regulators, one as 15 for VCCO power pins and other for VCCAUX pins based on the recommendations. After defining regulators, you connect or map them to FPGA power pins. As FPGA includes large number of power pins, connecting to each power pin is a cumbersome process. To avoid the situation, FSP provides you an interface by which you can connect the regulators to thousands of power pins at one step. Power mapping must be done only when you are assure that there will be not be any further changes in the bank assignments or the pin assignments. This is just to avoid regeneration of the schematic with the changes.

FSP includes a *Power Connections* window that allows you to define, add, delete, or reset the power connections in the current project.

The first column lists the regulator names and second column lists the regulator values. These two columns are freezed columns. This indicates that you cannot move the columns around the window. The cells under these two columns are editable. You can manually edit the regulator names and values using mouse clicks. The remaining columns depends on the number of instances present in the canvas. These columns contains check boxes. You may see these check boxes in enabled or disabled mode. The enable check box appears if the instance contains a IO standard of the selected voltage connected and disable check box if the instance does not contain any IO standard power by the selected regulator value. Consider the below case. In this case, the *U96* instance has LVCMS 12 IO standard connected. In this case, the *U3* instance does not contain any IO standard powered by the 1.2 voltage.

# Allegro FPGA System Planner User Guide

## Working with Power Regulators

Regulator Name	Voltage	U2 [4vlx25ff668]	U3 [4vlx25ff668]	U4 [4vlx25ff668]	U96 [ep2s15f484]	U97 [ep2s90f780]
GND	0	0	0	0	0	0
V_1_2	1.2	1.2	1.2	1.2	1.2	1.2
V_0_9	0.9	0.9	0.9	0.9		
V_0_75	0.75	0.75	0.75	0.75	0.75	0.75
V_1_8	1.8	1.8	1.8	1.8		
V_2_5	2.5	2.5	2.5	2.5		
V1_1_5	1.5		1.5	1.5	1.5	1.5
V2_1_5	1.5	1.5				

The values against each instance and regulator indicates the auto power regulator mapping preference based on either pin voltages or pin names.

## Points to Remember Before Adding Power Regulators

This section describes few important points you should remember while working with power regulators:

- The power regulator name must be unique, which means the regulator name should not match with existing net names of the design. Otherwise there may be chances of net name conflicts. Following regulator names are recommended V\_1\_2, V\_1\_5, V\_1\_8 and so on.
- To use special characters in regulator name refer to the [Net Naming Conventions](#) section.

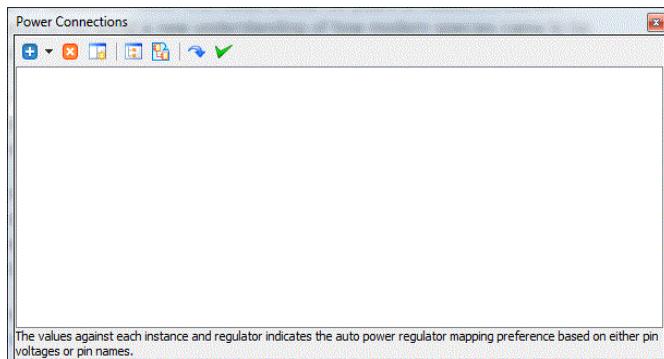
## Adding and Deleting Power Regulators

You can add power regulators and corresponding voltage values of your choice.

To add a regulator, perform the following steps:

1. Choose *Window – Power Connections*.

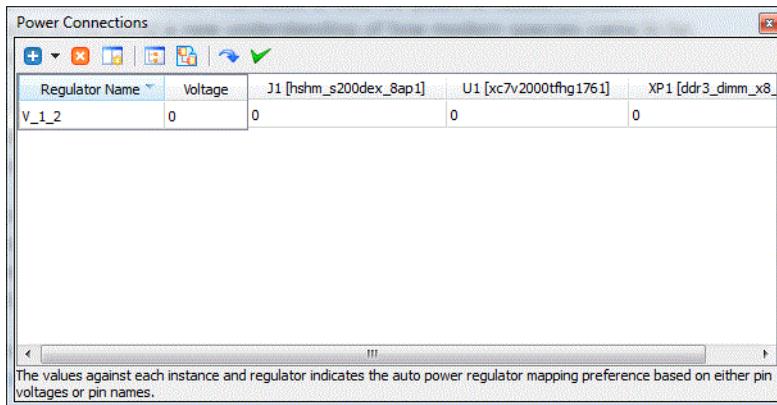
The *Power Connections* pane is displayed.



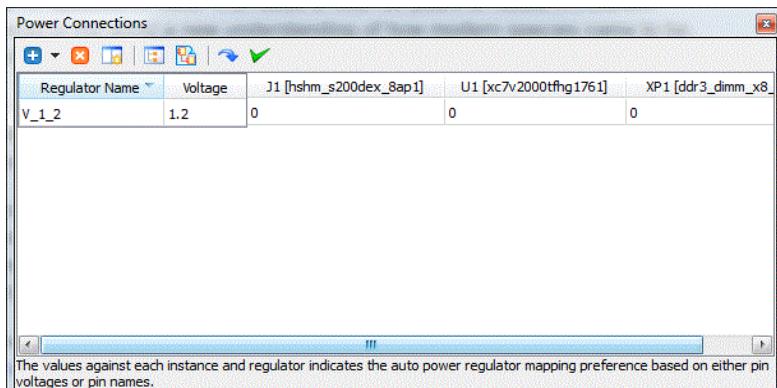
2. Click the + icon to add a new row.
3. Type a name under the *Regulator Name* column.

# Allegro FPGA System Planner User Guide

## Working with Power Regulators



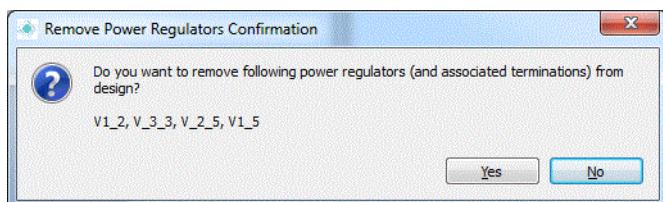
4. Enter a numerical value under the *Voltage* column.



To delete a regulator(s), perform the following steps:

1. Press and hold the Ctrl key and select the cells using the left mouse button.
2. Click the **x** icon to delete the selected cell(s).

A confirmation window is displayed about removing of regulators.



3. Click **Yes** to remove the regulator.

The power regulators are removed.

## Adding Power Regulators from Schematic Symbols

You can also add a schematic symbol of a power pin as a power regulator.

**Note:** When you add a schematic symbol of a power pin as a power regulator, note that the same regulator name is also updated in the *Power* section of the *Advanced Schematic Settings* form. The *Power* section provides you the option to output the power symbol in schematics.

To add a power regulator from a schematic symbol, do the following:

1. Click the down arrow button beside the + button, and choose *Add Regulator from Schematic Symbol*.

The *Add Regulator from Schematic Symbol* dialog box appears.

2. Click *Browse(...)*.

Component Browser appears.

3. Select a library name in *Library* pane that you want to add as a power regulator.
4. Select a cell name in *Cells* or enter the name of cell in the *Cells* text box.
5. In *Search Results*, click the row corresponding to the physical part you want to add.
6. Select a symbol view in the *Symbol* drop-down list.
7. Click *Select*.

The library:cell:view structure name appears in the *library:cell:view* text box of the *Add Regulator from Schematic Symbol* dialog box.

8. Select a name in the *Regulator Name* drop-down list box.
9. Select a value in the *Regulator Value* drop-down list box.
10. Click *OK*.

The power regulator name and voltage value are added in the *Power Connections* window.

## Defining Power Mapping

After adding the power regulators and corresponding voltage values, you define power mapping. You use this feature to define a mapping between a power regulator and a power/

ground pin name or its voltage value. For example, if you want to connect a power regulator to a power/ground pin, you first need to define the mapping between them. After defining the mapping, these mapping inputs are considered by FSP when you Automap Power Regulators.

**Note:** Automap Power Regulators is an automatic process that connects power regulators to power/ground pins based on the power mapping you define.

You can define power mapping using the following methods:

- [Mapping Voltage Value](#)
- [Mapping Power Pins](#)

## Mapping Voltage Value

In this method, you map a power pin's voltage value to a power regulator. For example, you add a power regulator  $V_{1\_2}$  of voltage value 1.2 v. To connect  $V_{1\_2}$  to all the power pins of an instance that are powered by 1.2 v, map the 1.2 v to  $V_{1\_2}$ .

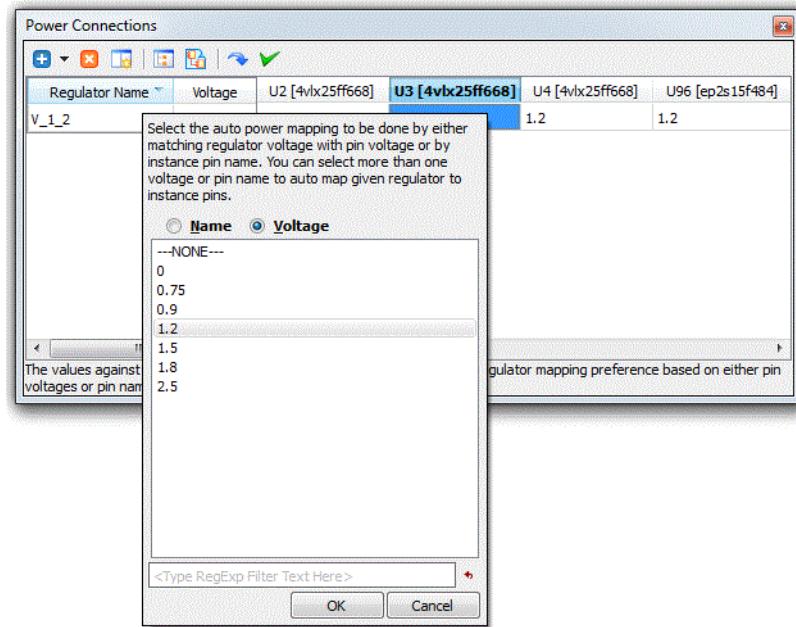


When you define power mapping, be careful to map voltage values to power regulators properly because these mapping inputs are considered during Automap Power Regulators process. Any incorrect inputs may lead to incorrect mapping. Incorrect mapping may cause errors while generating symbols and schematics.

To map a power pin's voltage value to a power regulator, do the following:

1. Under the instance name column, click on the cell for which you want to define power mapping.

A dialog box appears.



Note that the *Voltage* option is selected by default, and a list of the voltage values of the current instance appears.

2. Select a voltage value.
3. Click *OK*.

## Mapping Power Pins

In this method, you map a power pin name to a power regulator. For example, you define a power regulator, *V\_1\_5*, of voltage value 1.5 v. To connect *V\_1\_5* to all the power pins of an instance that are powered by 1.5 v, map the power pin name to *V\_1\_5*.

You can also add two power regulators of the same voltage value with different names, and map them to different power pins that are powered by the same voltage. For example, you add two power regulators: *V\_1\_2\_N* and *V\_1\_2\_P*. Now, you can map a power pin name to *V\_1\_2\_N* and another different power pin name to *V\_1\_2\_P*.

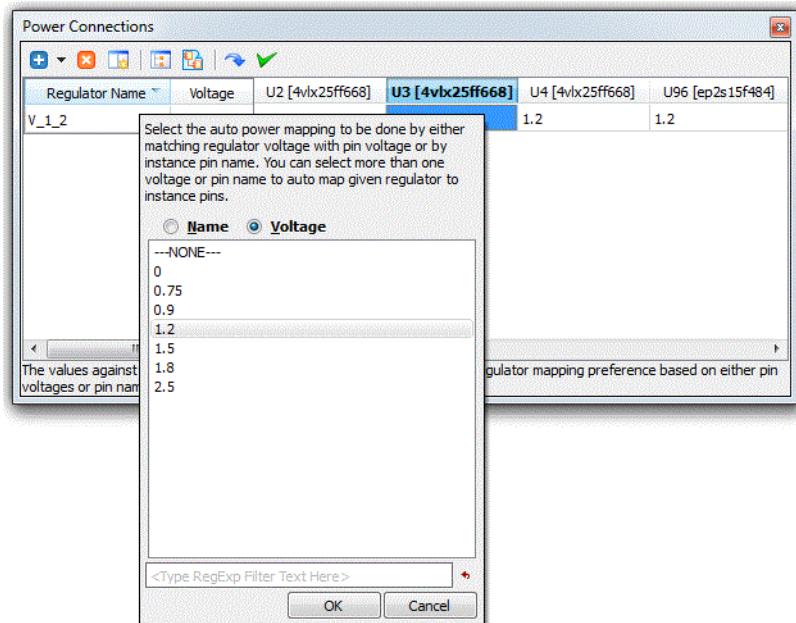
To map a power pin name to a power regulator, do the following:

1. Under the instance name column, click on the cell for which you want to define power mapping.

A dialog box appears.

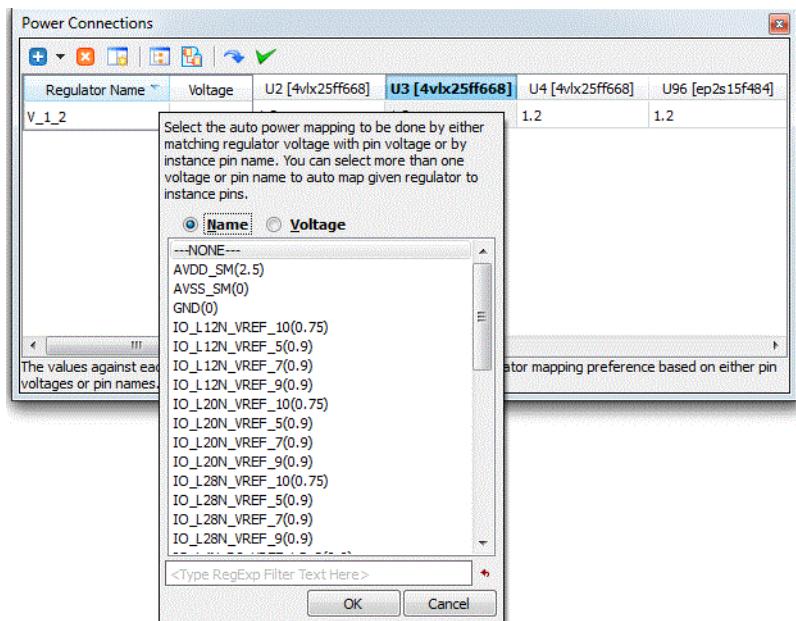
# Allegro FPGA System Planner User Guide

## Working with Power Regulators



### 2. Select Name.

You will see a list of power pin names of the current instance.



### 3. Select a name.

**Note:** You can also select multiple power pins names.

4. Click *OK*.

## Adding Power Regulators Automatically

Adding power regulators and defining power mapping are time consuming steps; however, you can perform these steps automatically by using a single-click option in *Power Connections*. When you choose this option, a set of regulator names and values, and power mapping is defined automatically.

**Note:** By default, power mapping is defined based on the voltage value method.

These regulator names and values are displayed based on the IO standards voltages present in the design. You can also manually modify the regulator names and values as per required.

The *Auto Add Regulator* feature is also useful to define a regulator for the voltages whose regulator is not defined. For instance, after power mapping you introduce a new IO standard of different voltage in the design. In this case, you do not need to manually define regulator name and value. You can use the auto add regulator option to define the regulator name and value for the new voltage.

**Note:** In this case, the auto add regulator option does not remove the existing regulators.

To define regulators automatically, perform the following steps:

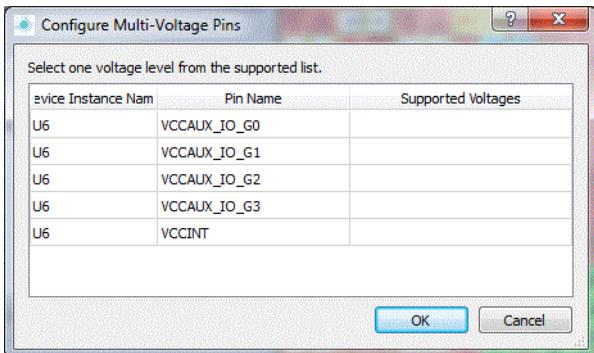
1. Click the *Auto Add Regulator* option.

A confirmation window is displayed about adding new regulators in the design.

2. Click *Yes* to proceed further.

The new regulator names and values are listed in the power connections window.

**Note:** If your design contains Virtex 7 devices, you will see the following window when you click the *Auto Add Regulator* option.



The power pins of the Virtex 7 devices such as VCCAUX and VCCINT can be powered with different voltages. For example, VCCAUX pins can be powered with 1.8 and 2 volts. This window lets you specify multiple voltages to similar type of power pins. In the *Supported Voltages* column, click and select a voltage from the drop-down list. When you click OK, a confirmation window is displayed about adding new regulators in the design.

## Mapping Power Regulators to Power Pins

You can manually map the power regulators in *Design Connectivity*. In *Design Connectivity*, you switch to power mapping view and select the regulator names under the *Regulator Name* column to map. The regulator values are automatically updated under the *Voltage Level* column based on the selected regulator name. Generally, mapping power regulators to hundreds of power pins manually is a time consuming process and not recommended. This may increase the chances of incorrect mapping. Incorrect mapping may cause error while generating symbols and schematics. FSP provides you a feature *Check Design Consistency* to catch errors related to power mapping. The *Check Design Consistency* option checks the inconsistency in the power mapping and reports error messages in the *Messages* window.

To avoid the manual mapping process, FSP provides you an option to automatically map the power regulators for the complete design. This option automatically detects the suitable regulators from the list of defined regulators in the power connections window, and map to the appropriate power pins. In manual mapping process, there might be chances that you may miss to map the regulators for few pins. In this case, this option provides you the flexibility of incremental power mapping. Incremental mapping indicates that the regulators will be automatically added for the missing power pins.

To map power regulators automatically, perform the following steps:

1. Select the check boxes under the instance name columns for which you want to map regulators.

2. Click  in the *Power Connections* window.

A confirmation window is displayed about mapping power regulators.

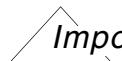
3. Click Yes to reset.

The power regulators are mapped to the power pins in the *Regulator Name* column of *Design Connectivity*.

## Removing Power Regulators

It is a common practice to modify or remove the existing power connections after mapping the power regulators to the power pins. FSP provides you an option to remove the power connections from the design. You can either manually remove the power connections, or remove the complete power connections by using a single-click option.

**Note:** *Removing power connections* implies that the power regulators that are mapped to the power pins need to be unmapped.



The step *removing power connections*, does not delete the defined regulators from the *Power Connections* window.

After the power connections are removed, the results can be seen in *Design Connectivity*. In *Design Connectivity*, the power regulator names are removed from the cells and displayed as blank in the *Regulator Name* column. In addition, the status of the power pins appears *Unmapped* in the *Status* column.

**Note:** While working with the power connections, it is recommended that you switch to *Power Mapping View* in Design Connectivity.

Another alternative way to delete the power connections of specific power regulator is to delete the defined power regulator from the *Power Connections* window. For instance, if you want to delete the V\_1\_2 voltages connections from the design, invoke the *Power Connections* window, select the V\_1\_2 regulator row, and click the *Delete* icon.

To remove the power connections from the design, perform the following steps:

1. Choose *Design – Delete All Nets* or click the *Delete All Nets* icon in the .

The Remove All Confirmation dialog box is displayed with the *Remove All Power Nets* option checked by default.

**Note:** The *Remove All Power Nets* option is displayed unchecked or disabled if the

power regulators are not mapped.

2. Click *OK* to remove all the power connections.

The power connections are removed from the design.

## Resetting and Mapping Power Regulators

Sometimes it may happen that you:

- perform an incorrect power mapping
- after power mapping you change the definition of the defined regulators
- define a new power regulator after power mapping.

FSP gives you the opportunity to revert back to the appropriate regulator names, update the modified power regulator name in Design Connectivity, and automatically perform the power mapping for missing regulators. The *Reset and Mapping Regulator* option help you to achieve this. This option is also useful when you have not specified any regulator names in the cell(s) and want to automatically map the regulator names in the empty cell(s).

To reset and map power regulators, perform the following steps:

1. Select the check boxes under instance name columns for which you want to reset the regulators.
2. Click  in the *Power Connections* window.  
A confirmation window is displayed about resetting power regulators.
3. Click *Yes* to reset.

The original regulator names are restored and mapped in the *Regulator Name* column of *Design Connectivity*.

## Exporting Power Mapping Information

FSP allows you to export the power mapping information in an external file of format comma supported value (csv). This feature is useful when you want to do ECO changes in your design. After exporting the power mapping information to csv file, you can continue to work on the other areas of the design. This feature provides a means to edit the power mapping information in a spreadsheet or in a text editor of your choice. You first export the power pin

properties to a external file, edit the properties in the application of your choice, and then import the edited properties back into the Design Connectivity.

The *Export CSV* dialog box lets you output the power pin properties for the design. Exporting power pin properties task is considered to be a part of the *Export CSV* process. For example, when you export the properties of the parts and pins through the *Export CSV* dialog box, the power pin properties are also exported by default. However, you can customize the process of exporting properties in a such way that only power pin properties are exported. You can achieve this by unselecting all the column names in the *Export CSV* dialog box except *Regulator Name* and *Voltage Value* columns. For more information on the *Export CSV* process, see the [Exporting Part and Pin Properties](#) section.

FSP provides an efficient way of exporting power pin properties as an alternative to unselecting columns method. You can switch to *Power Pins* view. The *Power Pin* view displays all the power pins of the instances that are present in the Canvas. In addition, FSP automatically customize the columns display, and display only those columns that are relevant while working with the power pin properties. You can then use the *Export CSV* dialog box to export the properties without unselecting the columns.

To export the power pin properties, perform the following steps:

1. In the *Design Connectivity*, click the drop-down button at the upper right corner and select *Power Pins*.  
The *Power Pin* view is displayed.
2. Click the *Export CSV* icon.  
The Export CSV dialog box is displayed.
3. Click *Export*.

After you click *Export*, FSP displays successful message in the *Log* window with the path of directory where the csv file is generated.

## Importing Power Mapping Information

After you export the power pin properties using the *Export CSV* dialog box, you can use a spreadsheet or a text editor application your choice to edit the properties and it values. In the applicaton, you can add a new regulator, modify and delete the existing regulator under the *Regulator Name* column.

After you complete editing properties, you use the *Update from CSV* dialog box to import the changes back into Design Connectivity.



***It is mandatory to import the changed properties in the Power Pins view, only if you have exported the power pin properties from the Power Pins view.***

Importing power pin properties task is considered to be a part of the *Export CSV* process. For example, when you import the properties of the parts and pins through the *Export CSV* dialog box, the changed power pin properties are also imported by default. For more information, see the [Importing Part and Pin Properties](#) section.

After you import the power pin properties, the regulator names in the *Regulator Name* in Design Connectivity are replaced or overridden by the changed regulator names in the spreadsheet.

The following changes you may see when you import the power regulators:

- If any new regulator name is imported, the new regulator name is immediately reflected in the *Power Connections* window.
- If any existing regulator name is removed from the spreadsheet, the regulator name is removed from the *Power Connections* window after importing.

To import power regulators, perform the following steps:

1. In the *Design Connectivity*, click the drop-down button at the upper right corner and select *Power Pins*.  
The *Power Pin* view is displayed.
2. Click the *Import from CSV* icon.  
The *Update from CSV* dialog box is displayed.
3. In the *CSV File* field, specify the name and path where the file exists.  
Or
4. Click *browse(...)* to browse to the directory where the file exists.
5. In the *Ignore Rows* field, specify the row numbers that you want to ignore during import process.
6. In the *Delimiter* field, select a delimiter that should be used to parse the data.
7. In the *Reference Column* field, select the appropriate option.
8. Click *Load/Refresh*.

The properties are displayed in the spreadsheet view.

9. In the first row, of the spreadsheet view specify the column names from the drop-down list whose values need to be imported.
10. Click *OK* to import the properties.

After you click *OK*, the import session begins and changed properties are highlighted in the Design Connectivity. In addition, the four options *Show*, *Reset Invalid Values*, *Save Import Values*, and *Cancel* options are displayed at the top right corner of the Design Connectivity.

11. Click *Save Import Values* to save or commit the changed properties.

The session exits and all the highlighted properties are displayed in default color.

---

## **Working with Associated Components**

---

This chapter describes the following sections:

- [Terminations](#)
- [Power Filters](#)
- [Defining Decoupling Capacitors](#)
- [Defining External Connections](#)

## Terminations

Termination is a set of components such as resistors, capacitors or diodes which are added to pins, signals or buses to prevent the overflow of current occurring at pins and buses.

FSP termination feature enables you to add termination circuitry on several nets (typically thousands of nets) quickly and consistently thereby saving time and effort. Various termination types are offered by FSP to be used in your design. The two most common termination techniques provided by FSP are:

- Series Termination
- Split Termination

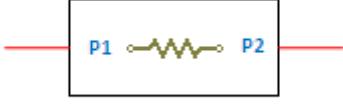
These termination techniques are further classified into various termination types based on the type of pin you are going to apply. Several GUI features enable you to quickly define these termination types on all the pins in the design which require termination. Detail description about each termination types are described in next section.

**Note:** Applying termination on individual nets is not done in Define Termination dialog box. The Define Termination dialog box enables you to define only the termination types that you want to use in your design.

## Termination Types

FSP supports seven different types of termination. These termination techniques are classified according to the status of the pin such as single and differential pin. The following table explains the different termination types and their functionality.

---

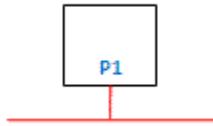
Termination type name	Figure	Functionality
Series Single Ended		This termination type may include a two terminal discrete component such as series resistor, clamp diode inductor, capacitor or combination of parallel and series resistors.  This termination type may also include a hierarchical symbol of complex circuitry.

# Allegro FPGA System Planner User Guide

## Working with Associated Components

### Split/Pull Up/Pull Down

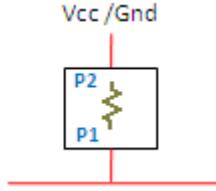
Single Ended



This method may include any termination circuitry which is captured as hierarchical symbol with single pin node.

### Pull Up/Down

Single Ended



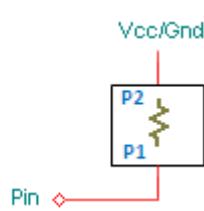
This type of termination are used for nets.

This method may include a two terminal discrete component. One end is connected to net and the other end is connected to Vcc/Gnd.

This method may also include any complex termination circuitry which is captured as hierarchical symbol with two nodes.

### Pull Up/Down

Power Filter

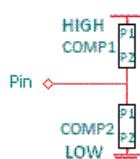


This type of termination is used for IO pins or power pins. This termination can be used to directly connect the filter circuitry to the component pins. For detailed information see [Power Filters](#) section.

This method may include a two terminational discrete component or combination of these with others.

### Thevenin

Power Filter



This type of termination may include two resistors. One connected to a low voltage and another connected to a high voltage.

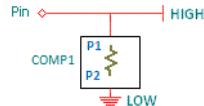
This termination is similar to the thevenin, but it is connected to a power pin.

# Allegro FPGA System Planner User Guide

## Working with Associated Components

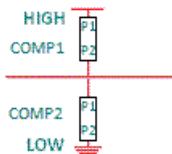
### Hookup

#### Power Filter



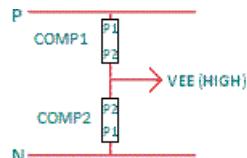
This type of termination may include one resistor. One connected to pin and another connected to a high or low voltage.

#### Thevenin



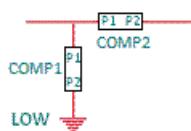
This type of termination may include two resistors. One connected to a low voltage and another connected to a high voltage.

#### Parallel



This type of termination may include two resistors. Both are connected parallel and in series with VEE.

#### Isolvpcl



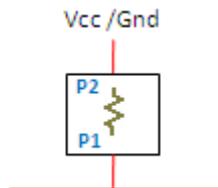
This type of termination may include one resistor and one capacitor connected in series.

### Terminations on Differential Pairs

FSP supports adding terminations to differential pair pins. You are also allowed to use some of the single ended terminations on differential pair pins. The table below lists the terminations (including single ended) that can be added to differential pair pins.

#### Pull Up/Down

#### Differential



In this method the differential signals will be terminated by putting a Pull up/down or both resistors near the nets. The other end can be connected to Vcc/Gnd.

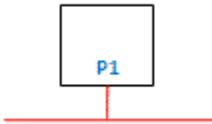
This method may include a two terminal discrete component. This method may also include any complex termination circuitry which is captured as hierarchical symbol with two nodes.

# Allegro FPGA System Planner User Guide

## Working with Associated Components

### Split

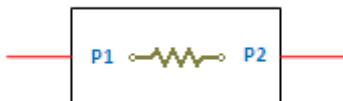
Differential



This method may include any termination circuitry which is captured as hierarchical symbol with in single pin node in between the two differential signals.

### Series

Differential

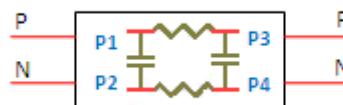


In this method the differential signals may include a two terminal discrete component (such as series resistor, clamp diode inductor, capacitor or combination of parallel and series resistors) on both the nets.

This method may also include any complex termination circuitry which is captured as hierarchical symbol with two nodes.

### Differential

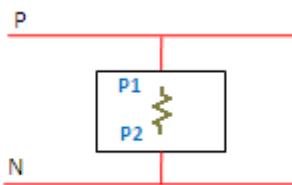
Differential-Series-Parallel



This method must include at least a series resistor as well as a parallel resistor connected across differential nets.

This method may also include any complex termination circuitry which is captured as hierarchical symbol with two nodes.

### Differential



This method includes a two terminal discrete component (such as series resistor, clamp diode inductor, capacitor or combination of these) in between the two differential signals.

This method may also include any complex termination circuitry which is captured as hierarchical symbol with two nodes.

## Mapping Terminations Ports

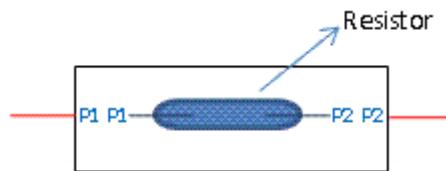
Mapping termination ports represents capturing connectivity between the termination circuitry and the components. Once the termination type is selected for your design, you can map the termination ports with any hierarchical symbol block of a complex termination circuitry or discrete components.

In FSP mapping termination ports is also a part of defining termination process which is not a compulsory task. That means you can keep the termination type unmapped. You use the Define Termination dialog box for termination mapping. Out of all the termination types available in Define Termination dialog box only few may include hierarchical symbol of complex termination circuitry and rest of them are mapped to discrete components. For more information see [Termination Types](#) section. While mapping you select a symbol of any discrete component or termination circuitry and map the termination ports with discrete component terminals or circuitry nodes. Mapping ports is possible only when the count of selected termination type and schematic symbol (nodes of termination circuitry) are equal.

The following section is the description of the various mapping scenarios possible while termination mapping.

### ■ Placing Discrete Component

In this scenario, you place a primitive/non primitive component in the termination block. Placing primitive/non primitive component in termination block means mapping component ports with selected termination block ports. FSP allows you to use any discrete component such as diodes, resistors or capacitors to place in termination block. However the number of ports for both termination block and discrete component should be equal. The figure below displays an overview of mapping resistor ports with series termination block.



#### ***Valid Mapping***

You can place a resistor component in series termination block since both of them have two ports.

#### ***Invalid Mapping***

You cannot place a four port component in series termination block.

## ■ Using Existing Hierarchical Schematic Symbol

In this scenario, you can use an existing hierarchical schematic symbol of any circuitry or create your own hierarchical schematic symbol and map with the selected termination ports. However the nodes of the hierarchical schematic symbol should be equal with termination ports.

## ■ Manually Mapping Termination Circuitry after FSP Schematic Generation

In this scenario, you can keep the selected termination type unmapped. After schematic generation you can complete your termination circuitry and manually map with FSP generated hierarchical schematic symbol in DE-HDL.



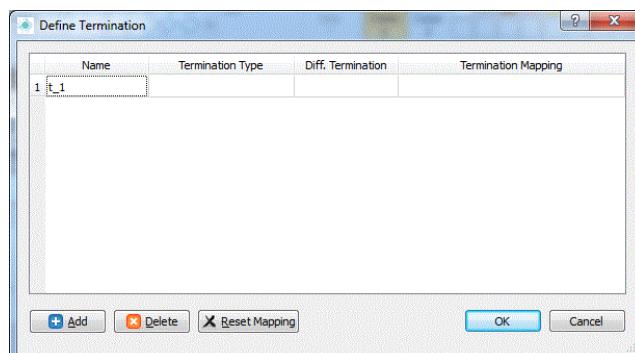
Defining termination for the instances is an important task. Updating termination automatically to the device side and differential pair pins depends on the settings you define in Define Termination dialog box. So before walking through define termination steps it is recommended that you first understand the functionality of the fields and buttons of Define Termination dialog box. For more information see [Define Termination](#) section for more information.

## Defining Series Termination For Single Ended Signals

To define a series termination for single ended signals perform the following steps:

1. Choose *Tools – Terminations and Power Filters*.

The Define Termination dialog box is displayed.



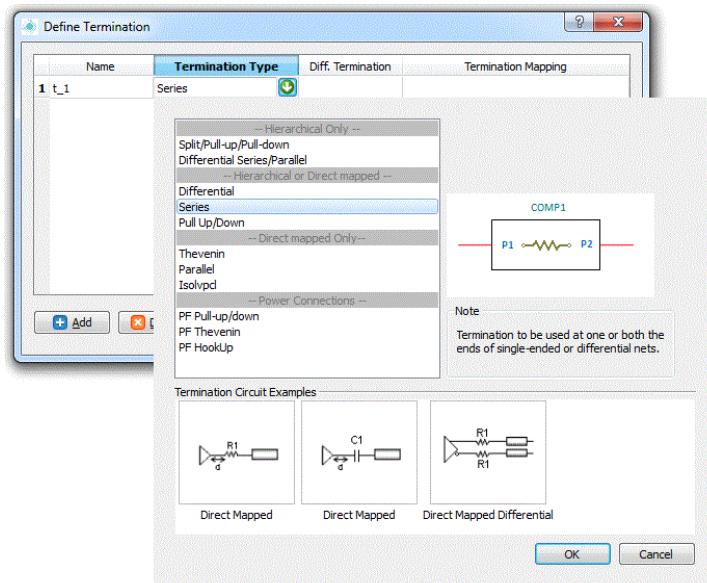
2. In the Name column, enter a termination name or use the same default name *t\_1*.

## Allegro FPGA System Planner User Guide

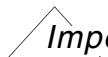
### Working with Associated Components

- In the Termination Type column, click the cell.

The Termination Type drop down pane displays the available termination types.



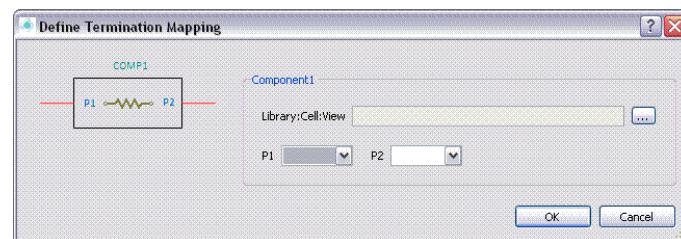
- Select the *Series* and click *OK*.

 **Important**

Do not specify anything in the Diff.Termination column since you are defining the termination for single ended signal.

- Click *browse (...)*, in the Termination Mapping column.

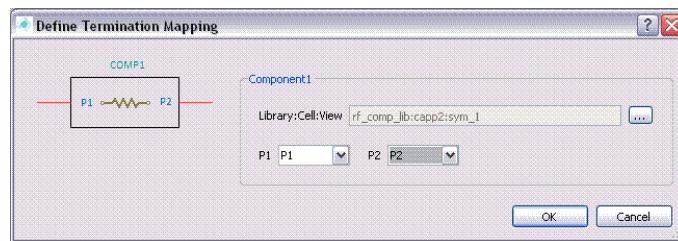
The Define Termination Mapping dialog box is displayed.



- Click *browse (...)* to invoke the Component Browser.

- Select a library name in Library pane.
- Select a cell name in Cells pane or enter the name of cell in Cells text box.

- c. In Search Results pane, click the row corresponding to the physical part you want to add.
  - d. The symbol and footprint name for the component is displayed in Part Name tab.
  - e. Select a symbol view from Symbol drop down list.
  - f. Click Select.
7. Click and select the component port name from P1 drop down list.
8. Click and select the other side of component port name from P2 drop down list.



9. Click *OK* to apply the termination mapping.
10. Click *Add*, to add a new blank row to define another termination in the Define Termination dialog box.
11. Click *OK* of the Define Termination dialog box.

## Defining Series Termination for Differential Ended Signals

The process of adding series termination for differential ended signals is described below:

1. Choose *Tools – Terminations and Power Filters*.  
The Define Termination dialog box is displayed.
2. In the Name column, enter a termination name *P1*.
3. In the Termination Type column, click cell.  
The Termination Type column drop down pane displays the available termination types.
4. Select the *Series* and click *OK*.
5. In the Diff.Termination column, select a defined series termination name *P1*. See below.

Name	Termination Type	Diff.Termination
------	------------------	------------------

P1              Series              P1

**Note:** If you apply P1 termination to any of the differential type pair the other will be automatically selected. For example, if you select *P1* for P type pin then automatically *P1* is applied to N type pin.

**6.** Click *browse (...)* in Termination Mapping column.

The Define Termination Mapping dialog box is displayed. Since you have selected the Series Termination it may include one resistor or one capacitor. You need to select a primitive component symbol.

**7.** Click *browse (...)* to invoke Component Browser.

- a.** Select a library name in Library pane.
- b.** Select a cell name in Cells pane or enter the name of cell in Cells text box.
- c.** In Search Results pane, click the row corresponding to the physical part you want to add.
- d.** The symbol and footprint name for the component is displayed in Part Name tab.
- e.** Select a symbol view from Symbol drop down list.
- f.** Click Select.

**8.** Click and select the component port name from *P1* drop down list.

**9.** Click and select the other side of component port name from *P2* drop down list.

**10.** Click *OK* to apply the termination mapping.

**11.** Click *Add* to add a new blank row to define another termination in the Define Termination dialog box.

**12.** Continue to perform the steps from two to eleven.

**13.** Click *OK* of Define Termination dialog box.

## **Adding Differential Termination to Differential Ended Signals**

To add differential termination to differential ended signals, perform the following steps:

**1.** Choose *Tools – Terminations and Power Filters*.

The Define Termination dialog box is displayed.

## Allegro FPGA System Planner User Guide

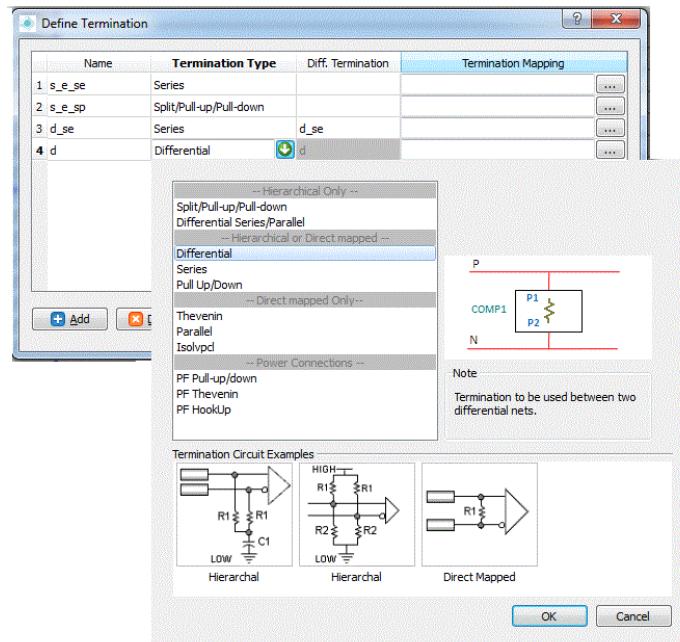
### Working with Associated Components

2. In the Name column, enter a termination name.

3. In the Termination Type column, click the cell.

The Termination Type column drop down pane displays the available termination types.

4. Select the *Differential* and click *OK*.



The selected termination name is automatically displayed in the Diff. Termination column.

5. Click *browse (...)* in the Termination Mapping column.

The Define Termination Mapping dialog box is displayed.

6. Click *browse (...)* to invoke Component Browser.

a. Select a library name in Library pane.

b. Select a cell name in Cells pane or enter the name of cell in Cells text box.

c. In Search Results pane, click the row corresponding to the physical part you want to add.

d. The symbol and footprint name for the component is displayed in Part Name tab.

e. Select a symbol view from Symbol drop down list.

- f. Click Select.
7. Click *OK* to apply the termination mapping.
8. Click *Add* to add a blank row to define another termination in Define Termination dialog box.
9. Continue to perform the steps from two to nine.
10. Click *OK* of Define Termination dialog box.

## **Adding Pull Up Termination to Single and Differential Ended Signals**

To add Pull Up/Down termination for both single and differential ended signals you can follow the explained above process. There is one extra step that you need to perform is to specify the regulator voltage for the Pull Up/Down termination.

While specifying a regulator for pull up/down termination you must remember a point that for each type of regulator you need to define different sets of pull up/down termination. Such as for V\_1\_2 voltage regulator you need to define one pull up/down termination, for V\_1\_5 voltage regulator define another pull up/down termination and so on.

When you take the FSP design to DE-HDL schematics, the power regulator you assigned in Pull Up/Down termination block will be visible.

## **Modifying a Termination**

You can modify the termination at any time during the design process. But you need to ensure that termination that you are going to modify is already applied to the nets/pins or not. If it is already applied you need to reapply the termination in all the nets/pins once again.

To modify the termination, perform the following:

1. Choose *Tools – Termination and Power Filters*.

The Define Termination dialog box is displayed with all the defined termination details.

2. Make the required changes and click *OK*.

## **Deleting a Termination**

Terminations can also be deleted by performing the following steps:

**1. Choose *Tools – Termination and Power Filters*.**

The Define Termination dialog box is displayed with all the pre-defined termination details.

**2. Select a row and click *Delete*.**

## Applying Termination to Instance Pins

After defining the terminations, you can start applying terminations to the appropriate pins. Terminations can be applied to both the device and interface instance pins. You apply the terminations to the instance pins using the Design Connectivity. Design Connectivity gives you a spreadsheet view that helps you to apply the termination on each pin of the instances. Several quick and right mouse button options are also available in the editor to quickly apply the termination on all the pins in the design. The Properties can also be used as an alternative for applying terminations to the pins. Invoke and arrange the Design Connectivity and Properties side-by-side. Click on a pin in the Design Connectivity, the properties of the selected pin is displayed in the Properties. In the Properties, use the *Pin Termination* and *FPGA Ext Termination* cells to apply the termination.

**Note:** The described below are the steps to apply series termination to both single ended and differential pair pins. You can follow the same steps for other terminations also.

### Applying Series Termination to Single Ended Pins (Interface/Device)

To apply the series termination to single ended pins, perform the following steps:

**1. Invoke the *Design Connectivity*.**

The Design Connectivity is displayed.

**Note:** In the Design Connectivity, the Pin Termination and FPGA Ext. Termination columns are used to apply terminations for instance pins.

**2. In the Pin Termination column, double-click on a cell and select a termination name from the drop-down list.**

**Note:** When you double click on a cell, a list of termination names are displayed. These termination names are defined for the single ended pins in the Define Termination dialog box are displayed.

**3. In the FPGA Ext Termination column, the termination name is automatically applied as defined in the Define Termination dialog box.**

**4. Apply the termination to other pins if required, by performing the steps 1 to 4.**

5. Click *OK* to save the settings.

## Applying Series Termination to Differential Pair Signals (Interface/Device)

To apply the series termination to differential pair pins, perform the following steps:

1. Invoke the *Design Connectivity*.

The Design Connectivity is displayed.

2. In the Pin Termination column, for a P-type pin double-click on a cell and select a termination name from the drop-down list.

The Pin Termination column value for N type pin is automatically updated.

**Note:** When you double click on a cell, a list of termination names are displayed. These termination names are defined for differential pair pins (Diff.Termination column) in the Define Termination dialog box are displayed.

3. In FPGA Ext. Termination column, termination name is automatically applied as defined in the Define Termination dialog box.

4. Apply the termination to other pins if required by performing steps 1 to 4.

5. Click *OK* to save the settings.

## Quick Options for Applying Terminations

FSP provides you various quick pop-up menu options in Pin Properties of Interface Instance editor to ease the task of applying terminations to pins which are typically large in number. The following are pop-up menu options that you can perform on terminations:

- Apply <Termination Name> to all <Standard Name> Standard - Select a termination name in Pin Termination column and click *Apply <Termination Name> to all <Standard Name> Standard* in context sensitive menu.

The termination name is applied to all the pins having same IO Standard compare to selected pin.

- Apply <Termination Name> to all <Pin Type> <Standard Name> Standard - Select a termination name in Pin Termination column and click *Apply <Termination Name> to all <Pin Type> <Standard Name> Standard* in context sensitive menu.

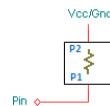
The termination name is applied to all the pins having same pin type and IO Standard compare to selected pin.

- Apply Termination to Bus - Select a termination name in Pin Termination column and click *Apply Termination to Bus* in context sensitive menu.

The termination name is applied to all the bus pins.

## Power Filters

In general, board designing pins are not directly connected to power supply. They are connected via some circuitry like inductors, resistors, or combination of these with other discrete components to reduce the power supply noise. In FSP, you use the power filter termination to filter the power supply noise. Power filter termination is a two port termination or circuitry (hierarchical termination) applied on power pins or IO pins with other end connected to power regulator or ground. See below for a pictorial view of power filter termination in FSP.



Generally power filters are applied on high speed pins (such as MGT) based on the power recommendations provided by FPGA vendors. But FSP allows you to apply the power filter termination on any of the device/interface instances pins. So you are recommended to apply power filter termination on the pins as per recommendations.

FSP treats power filter termination as Pull Up/Down termination and outputs the same in schematics also. The dissimilarities between pull up/down and power filter termination in FSP is Pull Up/Down termination is applied on pin (connected to a signal net) and other end of the termination is connected to supply or ground and power filter is applied only on power pins or IO pins. For detailed information on Pull Up/Down termination see [Termination Types](#) section for more information.

The following section describes how you can apply power filter termination to power pins and IO pins:

- [Defining Power Filters](#)
- [Applying Power Filters to Power Pins](#)
- [Applying Power Filters to Instance Pins](#)

## Defining Power Filters

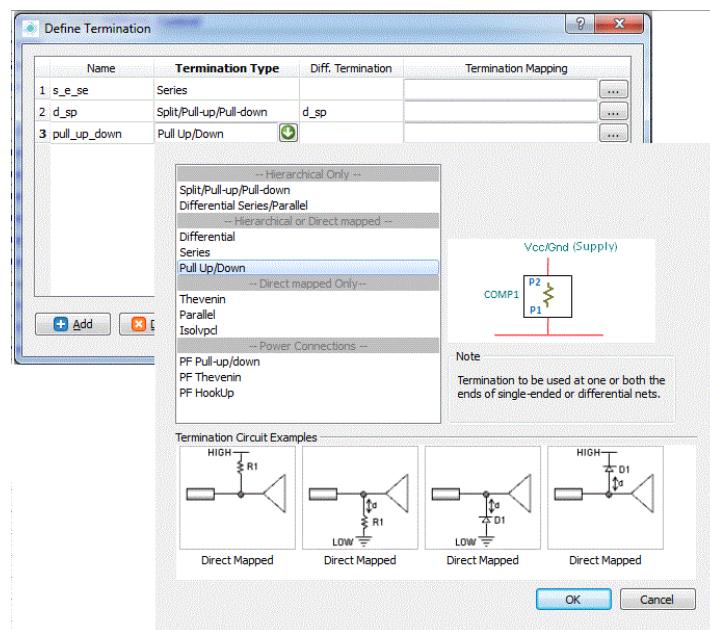
Before adding the power filters to pins first you need to define the power filters. To define power filters perform the following steps:

1. Choose *Tools – Termination and Power Filters*.

The Define Termination dialog box is displayed.

2. In the Name column, enter a termination name.

3. In the Termination Type column, select the *PF Pull-Up/Down* from the drop-down list and click *OK*.



The Diff.Termination and Other End columns are disabled for power filter termination.

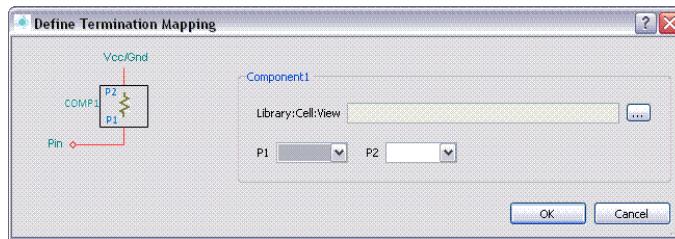
4. Click *browse (...)* in the Termination Mapping column.

The Define Termination Mapping dialog box is displayed.

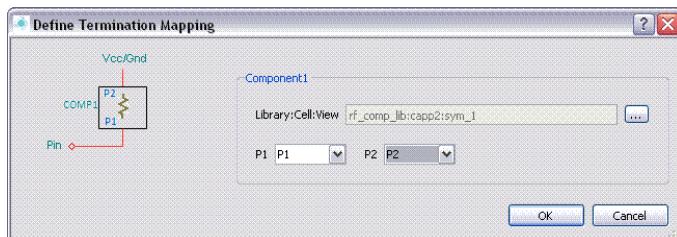
**Note:** Since you have selected the PF Pull Up/Down it may include one resistor or one capacitor or any two port discrete component. You need to map the termination ports with discrete components pins.

# Allegro FPGA System Planner User Guide

## Working with Associated Components



5. Click *browse (...)* to invoke Component Browser.
  - a. Select a library name in Library pane.
  - b. Select a cell name in Cells pane or enter the name of cell in Cells text box.
  - c. In Search Results pane, click the row corresponding to the physical part you want to add.
  - d. The symbol and footprint name for the component is displayed in Part Name tab.
  - e. Select a symbol view from Symbol drop down list.
  - f. Click Select.
6. Select the component port name from *P1* drop down list.
7. Select the other side of component port name from *P2* drop down list.



8. Click *OK* to apply the termination mapping.
9. Click *Add* to add a new blank row to define another power filter in the Define Termination dialog box.
10. Click *OK* of Define Termination dialog box.

The power filter termination is defined.

## Applying Power Filters to Power Pins

After defining the power filters, you can apply them onto the power pins. To apply power filters, you can either use the *Design Connectivity* or the *Properties* based on your convenience. In the *Design Connectivity*, you use the *Power Regulator* and *Power Filter* columns, whereas in the *Properties* you can use the *Power Regulator* and *Power Filter* rows to apply. When you apply power filter, it is mandatory to specify regulator name. This ensures that the selected power pin is connected to specified voltage regulator via power filter circuitry.

**Note:** You need to select appropriate power regulators based on your design requirements. For example, for a VCCAUX pin, you must apply a power filter termination and select V\_1\_2 as regulator name.

To apply power filters on power pins perform the following steps:

1. Choose *Tools – Power Connections* or click *Map Power Connections* icon in .  
The Power Mapping editor is displayed with instance names and part names in tree view structure.
  2. Click + to expand the tree view structure to apply the power filter for a particular instance.  
The instance bank names and bank numbers are displayed in tree view structure.
- Note:** Click + icon in tool bar to apply power filter for all the instances at one shot.
3. Click + to expand the tree view structure further.  
A list of pin numbers, signal names and voltage level of current bank is displayed.
  4. In Regulator Name column, click and select the voltage regulator name from drop down list.  
After selecting the voltage regulator name the voltage value is displayed in Regulator Voltage column.
  5. In the Power Filter column, click and select the power filter name from drop down list.  
**Note:** Power filters are not supported for external ports. That means if you define the power pins as external ports (by clicking External Ports check boxes) and apply power filters to it an error message is displayed in log window.
  6. Apply the power filters for other pins also, if required by performing steps 1 to 6.
  7. Click *OK* to save the settings.

This ensures that the power filters is added to power pins. When you generate the schematics FSP outputs power filter as series termination connected to power.

## Applying Power Filters to Instance Pins

**Note:** The following steps are applicable for the device instance and interface instance pins.

To apply power filters to Interface Instance IO pins, perform the following steps:

1. Invoke the Design Connectivity.

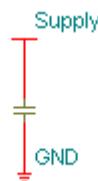
The Design Connectivity is displayed.

2. Click + icon next to the node rows to expand the tree view structure.
3. In the Power Filter column, double-click on a cell and select a power filter name from the drop-down list.
4. In the Regulator Name column, double-click on a cell and select a regulator name from the drop-down list.

**Note:** If you apply a power filter on the normal I/O pins, that is, any component pin which is connected to the regulator via circuitry or discrete component will not be used for other connections. This indicates that the pins will be internally preserved.

## Defining Decoupling Capacitors

Generally high speed switching environments generate noise on power lines due to charging and discharging of internal and external capacitors. The instantaneous current generated with rising and falling edges of the outputs causes the power line to generate noise. This behavior can violate the VCC recommended operating conditions provided by FPGA vendors creating serious problems. The key solution to minimize such problem is to connect decoupling capacitor to power pins. Placing a capacitor between the power regulator and ground connection maintain the power supply voltage at the device. Decoupling capacitor is a special kind of pullup/down termination provided by FSP is placed between power regulator and ground connection. The general overview of a decoupling capacitor topology in FSP is shown below.



FSP provides you an intuitive environment for adding decoupling capacitors to power pins of any component which is on canvas.

Once the design is finished you can add decoupling capacitors before or after power mapping. But to avoid incorrect power mapping related issues it is recommended that you add decoupling capacitors after power mapping.

After generating schematics, FSP outputs the decoupling capacitor topology as Bypass termination. When you open the schematics in DE-HDL all the decoupling capacitors are displayed in separate page.

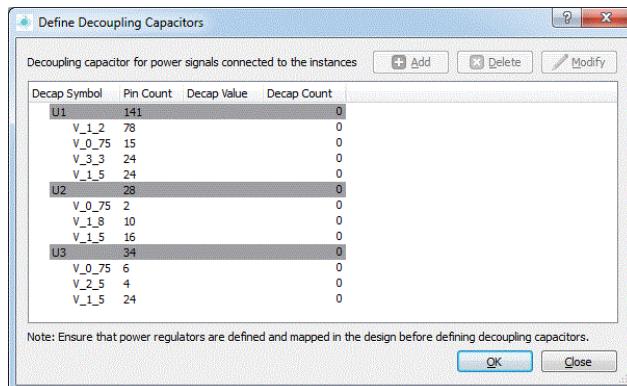
## Adding Decoupling Capacitor

Before adding decoupling capacitor ensure that power mapping is already done in your design.

To add a decoupling capacitor in your design, perform the following steps:

1. Choose *Tools – Decoupling Capacitor*.

The Define Decaps dialog box displays a list of instance names with associated power regulators and voltage values in tree view structure.



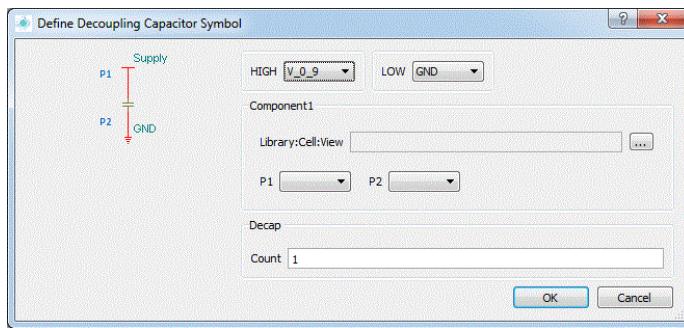
**Note:** After power mapping is done the regulator names and values are displayed in this dialog box.

2. Click a regulator name from list to which you want to connect capacitor.
3. Click *Add*.

The Define Decap Symbol dialog box is displayed.

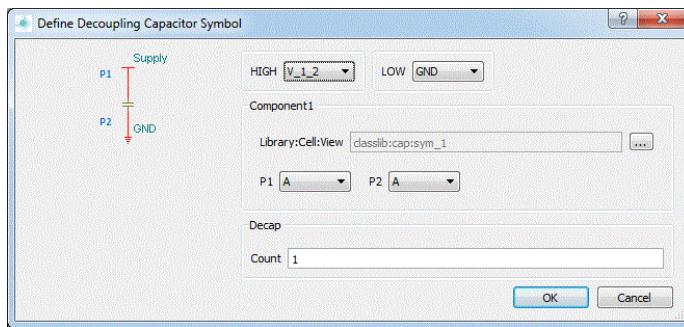
# Allegro FPGA System Planner User Guide

## Working with Associated Components



4. Click *browse (...)* to invoke Component Browser.
  - a. Select a library name in Library pane.
  - b. Select a cell name in Cells pane or enter the name of cell in Cells text box.
  - c. In Search Results pane, click the row corresponding to the physical part you want to add.
  - d. The symbol and footprint name for the component is displayed in Part Name tab.
  - e. Select a symbol view from Symbol drop down list.
  - f. Click Select.

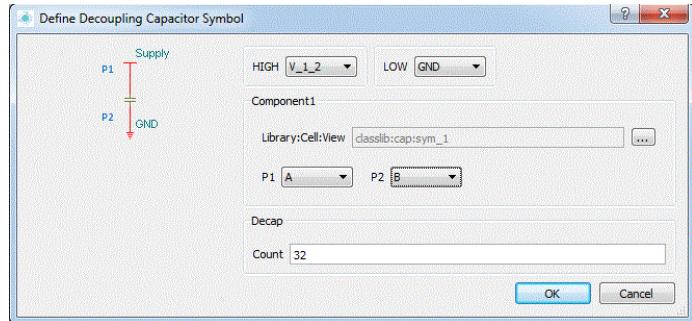
After selecting the symbol from Component Browser the `lib:cell:view` structure name is displayed in `Library:Cell:View` text box.



5. Click and select the component port name from *P1* drop down list.
6. Click and select the other side of component port name from *P2* drop down list.
7. In High field, click and select a port name from drop down list that you want to connect to regulator.
8. In Low field, click and select the other port name to that you want to connect to ground.

## Allegro FPGA System Planner User Guide

### Working with Associated Components



9. Enter the number of capacitors you want to connect to regulators in Count field.

For example, if you have fifteen V\_1\_5 power regulators in component you choose fifteen capacitors.

10. Click *OK* to save settings.

Once you click *OK*, the selected capacitor value is displayed in Decap Value column and number of capacitors is displayed in Decap Count.

Decoupling capacitor for power signals connected to the instances			
Decap Symbol	Pin Count	Decap Value	Decap Count
U126	0		0
U128	0		0
U139	0		0
U2	113		32
V_1_2	32	0.32uf	32
classlib:cap:sym_1		.01uf	32
V_0_9	4		0
V_0_75	8		0
V_1_8	18		0
V_2_5	15		0
V1_1_5	1		0
V2_1_5	35		0
U3	108		0
V_1_2	32		0

11. You can add capacitors for other regulators also, if required by performing steps 1 to 16.  
12. Click *OK* to add capacitors in your design.

Clicking *OK* indicates that capacitors are added in your design. When you generate the schematics FSP output these capacitors as Bypass termination in DE-HDL and will be displayed in separate schematic page.

## Modifying Decoupling Capacitors

You can also modify the connectivity, capacitance value or number of capacitors to be added in your design at any time. However you must remember a point, after adding capacitors at any time if you do any changes in the power mapping connections you need to remodify the capacitor value settings based on the changes.

1. Choose *Tools – Decoupling Capacitors*.

The Define Decoupling Capacitors dialog box is displayed with all predefined capacitors counts and their values.

2. Select the regulator name from the list and click *Modify*.

The Define Decoupling Capacitor Symbol dialog box appears with details of the defined capacitor.

3. Do the required changes by performing steps 1 to 18 of Adding Decoupling Capacitors chapter and click *OK*.

## Defining External Connections

In today's design, it may require that the nets or pins of the components in FSP canvas need to communicate with the components which are not captured in FSP. These nets are said to be external to the FSP canvas and declared as *External Port*. FSP provides you a convenient way to define ports. You can quickly define the export ports for routed and unrouted nets in the Design Connectivity window. The Design Connectivity window also provides various quick options to define external port to multiple pins, thereby saving time and effort. When you generate the schematics, FSP automatically creates a high-level port for these nets so that you can easily connect the FSP-generated design with the user-created design while keeping the optimized portion separate.

In DE-HDL schematic, the top-level design is encapsulated in a hierarchical block. Whereas external ports become interfaces of the hierarchical block and function as pins of the block symbol.

FPGA System Planner classify the definition of external ports into five following categories:

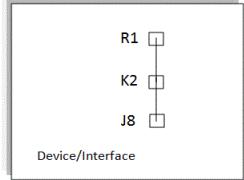
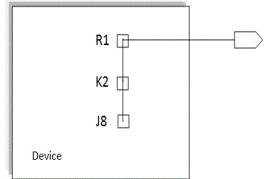
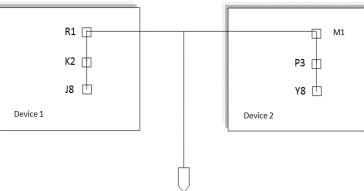
- Fixed Internal Port
- Fixed External Port
- Extend as External Port
- Do Not Connect

# Allegro FPGA System Planner User Guide

## Working with Associated Components

### ■ Preserve Pin

The following table explains the different external port types and their usage.

Connection Type	...looks like	Description	is connected to the following..
Fixed Internal Connection		This type of connection is applicable for un-routed nets(or pins). In this method of connection type, the pins are shorted together to form a single net with the same name.	All interface pins All connector pins All FPGA IO, power, and configuration pins
Fixed External Port		This method is similar to the Fixed Internal Port type. In this method, the unrouted nets (or pins) are shorted together to form a single net with the same name.  For vector pins, it is mandatory to define the <i>Fixed External Port</i> on all the bits.	All interface pins All connector pins All FPGA IO, power, and configuration pins
Extend as External Port		In this method, a routed net can be extended to communicate with the other components that are not captured in the FSP design.  For vector signals, it is mandatory to define the <i>Extend as External Port</i> on each bits of the signal.	Applicable only on routed nets.

## Allegro FPGA System Planner User Guide

### Working with Associated Components

---

Do not Connect	-	In this method, a pin is marked as tagged as do not connect. Going further, this pin will not be available for making connections.	All unrouted nets.  <b>Note:</b> In this case, you cannot apply a termination on the pin.
Preserve Pin	-	In this method, a pin is preserved from making connections for other purpose.	All unrouted nets.  <b>Note:</b> In this case, you cannot apply a termination on the pin.

## Applying an External Port

The connection types, are available as a drop-down list options in the cell(s) of the *Connection Type* column in Design Connectivity. However, these options are filtered and displayed based on the pin connections. For example, if a pin is connected to a net(or the pin has *Allocated* value in the *Status* column) then the *Extend as External Port* option is available in the *Connection Type* column.

To define a port connection type, perform the following steps:

1. Invoke the *Design Connectivity* window.
2. For a routed net:
  - a. In the *Connection Type* column, click on the cell and select the *Extend as External Port* option from the drop-down list.
  - b. In the *Net Name* column, type a name in the cell next to the cell (*Connection Type* column) and press *Enter*.
3. For an unrouted net:
  - a. In the *Connection Type* column, click in the cell and select an option from the drop-down list that displays the available connection types, then press *Enter*.
  - b. In the *Net Name* column, type a name in the cell next to the cell (*Connection Type* column) and press *Enter*.

**Note:** To define the same connection type to multiple pins or signals, press and hold *Ctrl* and select the cells, and click on the last selected cell and select an option from the drop-down list and press *Enter*.

**Note:** To define the same connection type for vectored pins or signals, define a connection type for any one of the bit, and right-click on the cell and choose the *Apply to Bus <Net Name>* option.

**Note:** You can also define the connection type in the *Properties*.

## Defining External Connections for Virtual Interface

FSP provides you support for Virtual Interface, which are created on the FSP canvas as place holders for real component interfaces. A virtual interface becomes an interface to the FSP hierarchical block. Therefore, the ports on the FSP hierarchical block symbol can be used to connect to a real component in the schematic.

After running the design, the virtual interface nets are automatically set as *Extend as External Port* in the *Connection Type* column in Design Connectivity. These nets are displayed in the disabled mode in Design Connectivity.

---

## Exporting and Updating Design

---

FPGA System Planner provides the ability to export the spreadsheet-based design information, that is, parts and pins properties in Design Connectivity in Comma Separated Value(CSV) file format. In addition, you are also allowed to update the spreadsheet-based design by importing the design information stored in the CSV file.

This chapter describes the following sections:

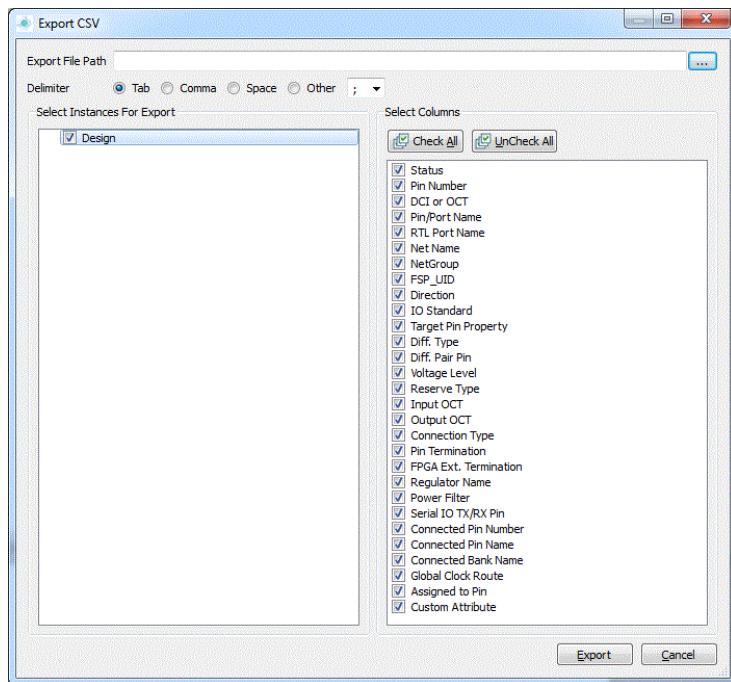
- [Exporting Part and Pin Properties](#)
- [Importing Part and Pin Properties](#)

**Note:** In this chapter, *properties* term is referred to the properties of instances, banks, groups, and pins, and *external file* is referred to the file with format csv, tab, and more.

## Exporting Part and Pin Properties

The Design Connectivity offers you an intuitive way of capturing the FSP design using spreadsheet. The spreadsheet-based design is very effective for capturing and editing the part and pin details of the design. However, sometimes for complex designs, you may experience that the Design Connectivity do not have much room to modify the large count of pin details.

### The Export CSV Window



The Design Connectivity lets you export the complete parts, groups, banks, and pins details in the .csv file format. The *Export CSV* dialog box in the Design Connectivity, provide a mean to edit the properties of parts and their pins in a spreadsheet or in a text editor that preserves tab or comma characters. You first export the properties to a external file, edit the external file in the application of your choice, and then re-import the edited properties in the Design Connectivity.

#### *Important*

After you export properties, it is recommended that you do not edit the design neither in Canvas nor in Design Connectivity, or library of which the properties were exported until after you import the changed properties. If you do, updating design

from csv process may lead to error prone state, and you will have to export and edit the properties again. It is a good practice to update the design or part before you export properties.

## Specifying Delimiters

The *Export CSV* dialog box provides you a choice of exporting the properties in different delimiters such as tab, comma, and space. Beside these delimiters, few other characters such as semi-colon(;), dollar(\$) and more, are supported as a part of a field's text. So before modifying the properties, be sure your spreadsheet application can save in any one of these formats.

## Exporting Pin Properties

The *Export CSV* dialog box outputs the complete information you see in the Design Connectivity. It outputs the properties in name and value format. The *Name* are the column names in the Design Connectivity and the *Values* are the values of the columns. See the following figure.



# Allegro FPGA System Planner User Guide

## Exporting and Updating Design

### Exporting Instance Banks/Groups Properties

The information of groups, banks, and instance names in the rows as depicted in the following figure are also exported as name and value format.

protocol name=U2_U1_U3
group target_device=U1 group_name=Data_Byte1 group_constraint=same_bank
group target_device=U1 group_name=Data_Byte2 group_constraint=same_bank

The text displayed in black color are exported as names and blue color text are exported as values. When you open this file in a spreadsheet or a text editor, the blue color text appears for all the pins to which group or bank they belongs to. See the following figure.

	A	B	C	D	E	F	G
1	Status	Pin/Port Name	Pin Number	Group Name	Target Device	Group Constraint	Instance/Protocol Name
140	Allocated	CLK0_P	4001	Data_Byte1	U1	same_bank	U2_U1_U3
141	Allocated	Data<0>	4002	Data_Byte1	U1	same_bank	U2_U1_U3
142	Allocated	Data<1>	4003	Data_Byte1	U1	same_bank	U2_U1_U3
143	Allocated	Data<2>	4004	Data_Byte1	U1	same_bank	U2_U1_U3
144	Allocated	Data<3>	4005	Data_Byte1	U1	same_bank	U2_U1_U3
145	Allocated	Data<4>	4006	Data_Byte1	U1	same_bank	U2_U1_U3
146	Allocated	Data<5>	4007	Data_Byte1	U1	same_bank	U2_U1_U3
147	Allocated	Data<6>	4008	Data_Byte1	U1	same_bank	U2_U1_U3
148	Allocated	Data<7>	4009	Data_Byte1	U1	same_bank	U2_U1_U3
149	Allocated	CLK1_N	4010	Data_Byte2	U1	same_bank	U2_U1_U3
150	Allocated	CLK1_P	4011	Data_Byte2	U1	same_bank	U2_U1_U3
151	Allocated	Data<8>	4012	Data_Byte2	U1	same_bank	U2_U1_U3
152	Allocated	Data<9>	4013	Data_Byte2	U1	same_bank	U2_U1_U3
153	Allocated	Data<10>	4014	Data_Byte2	U1	same_bank	U2_U1_U3
154	Allocated	Data<11>	4015	Data_Byte2	U1	same_bank	U2_U1_U3
155	Allocated	Data<12>	4016	Data_Byte2	U1	same_bank	U2_U1_U3
156	Allocated	Data<13>	4017	Data_Byte2	U1	same_bank	U2_U1_U3
157	Allocated	Data<14>	4018	Data_Byte2	U1	same_bank	U2_U1_U3
158	Allocated	Data<15>	4019	Data_Byte2	U1	same_bank	U2_U1_U3
159	Allocated	CLK0_N	4000	Data_Byte1	U2	same_bank	U2_U1_U3
160	Allocated	CLK0_P	4001	Data_Byte1	U2	same_bank	U2_U1_U3
161	Allocated	Data<0>	4002	Data_Byte1	U2	same_bank	U2_U1_U3
162	Allocated	Data<1>	4003	Data_Byte1	U2	same_bank	U2_U1_U3
163	Allocated	Data<2>	4004	Data_Byte1	U2	same_bank	U2_U1_U3
164	Allocated	Data<3>	4005	Data_Byte1	U2	same_bank	U2_U1_U3
165	Allocated	Data<4>	4006	Data_Byte1	U2	same_bank	U2_U1_U3
166	Allocated	Data<5>	4007	Data_Byte1	U2	same_bank	U2_U1_U3
167	Allocated	Data<6>	4008	Data_Byte1	U2	same_bank	U2_U1_U3
168	Allocated	Data<7>	4009	Data_Byte1	U2	same_bank	U2_U1_U3
169	Allocated	CLK1_N	4010	Data_Byte2	U2	same_bank	U2_U1_U3
170	Allocated	CLK1_P	4011	Data_Byte2	U2	same_bank	U2_U1_U3
171	Allocated	Data<8>	4012	Data_Byte2	U2	same_bank	U2_U1_U3
172	Allocated	Data<9>	4013	Data_Byte2	U2	same_bank	U2_U1_U3

### Selecting and Deselecting Instances and Columns to Export

For a design, you can export the properties of all parts or selective parts. The *Export CSV dialog box* provides you the option to select the instance for which you want to export the properties. Apart from selecting instances, it also facilitates you with the option to select or deselect the columns during export. This implies that you can skip those columns whose values you do not want to export. This feature benefit you by spending the time only on the properties of columns that you are interested in.

**Note:** When you open the external file in a spreadsheet, you may see the columns that were unselected and exported from the *Export CSV* dialog box. However, values are not exported.



When you export properties, it is recommended to export properties of the columns FSP\_UID, Pin Number, Pin/Port Name, and Instance/Protocol Name. As these column values are considered as reference by FSP during import process. For more information, see the [Choosing Reference column](#) section.

## Exporting Views

The Design Connectivity provides different functional views of the spreadsheet-based design. Each view represents the different aspects of the design information. For example, *Net View* displays the pins that are connected, the *Power Pins* displays the power pins of all the parts present in the canvas and more. The *Pin View* is the default view, that displays the complete properties of the parts and pins in the design.

Before exporting properties, it is recommended that you first understand about the views and how to switch between these different views. For detailed information on views, see the [Switching Views in Design Connectivity](#) section.

The *Export CSV* process depend on the view that you are working on. When you export properties, you must first decide in which view you want to export properties. As properties of the parts and pins are exported based on the view you are working on. The *Export CSV* process outputs the properties of only those pins that are visible in the current view. For example, if you are in *Power Pin* view, which displays the properties of power pins of all the parts. When you export properties, the *Export CSV* process exports the properties of power pins.

After you export properties, you are allowed to switch to different view and continue to work in other areas of design. After you complete modifying properties in a spreadsheet editor, you must import the changed properties in the view from which the properties were exported. So if you switch to a different view after exporting properties, you must switch back to the view from which the properties were exported and then import the changes. If you import changed properties in different view, the *Import CSV* process may fail or may go to error-prone state. For example, after you export properties from the *Power Pin* view, you switch to *Net View*. Now, when you import the changed properties, the *Import CSV* process will fail due to mismatched pins.



It is a good idea to import the properties in the view from which the properties were exported.

## Editing the External File

When you export properties, FSP creates a tab-delimited or comma separated values list of keywords and properties. You can edit this file in a spreadsheet or text editor of your own choice. Depending on which tool you use, you may see the names and values as rows and columns of cells or as lines of text. The external file starts with a row, containing names of column. These names represent the column names in the Design Connectivity. And the remaining subsequent lines are the values of the columns that are distributed in cells.

When you open the file in a spreadsheet, you can start modifying the values. However, it does not imply that you can change any column values. Any column that is editable in the Design Connectivity can be modified in the external file such as *Net Name*, *RTL Port Name* and more. In addition, any column that is non-editable or read-only in the Design Connectivity should not be modified in the external file. However, if you edit the value and import the changes back into the Design Connectivity, the changes will be ignored.



***Do not make changes in the properties of the FSP\_UID, Pin Number, Pin/Port Name, Instance/Protocol Name columns. As properties of these columns are considered as reference by FSP during importing comma separated values(csv) process.***

The following table describes the various changes you can make in the external file and the results you see when you import changed properties.

**You can make these changes..**

**You see these results..**

Add a column and  
subsequent cells (rows)

<>>>

Change a property to null This deletes the existing property.

# Allegro FPGA System Planner User Guide

## Exporting and Updating Design

### Delete a column

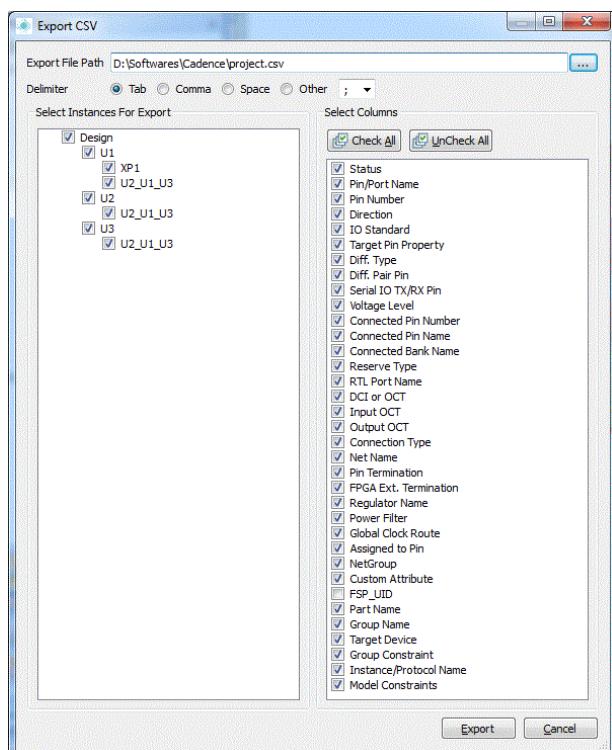
This has no effect on any part or pin. It is a good practice to delete columns you don't want to change. This make the external file easier to edit. If you delete a column name without also deleting the corresponding properties (rows), FSP does not report any error when you import file. However, FSP displays the column name as blank and the properties displayed as it is.

Change the column name This has no effect on any pin or part.

To export part and pin properties, perform the following steps:

1. Click the *Export CSV* icon in the of the Design Connectivity.

The *Export CSV* dialog box is displayed.



2. In the *Export File Path* field, specify the path and name of the file where you want to save the file.

Or

Click *browse(...)* to browse to directory where you want to save the file, enter the name of the file, and click *Save*.

## **Allegro FPGA System Planner User Guide**

### Exporting and Updating Design

---

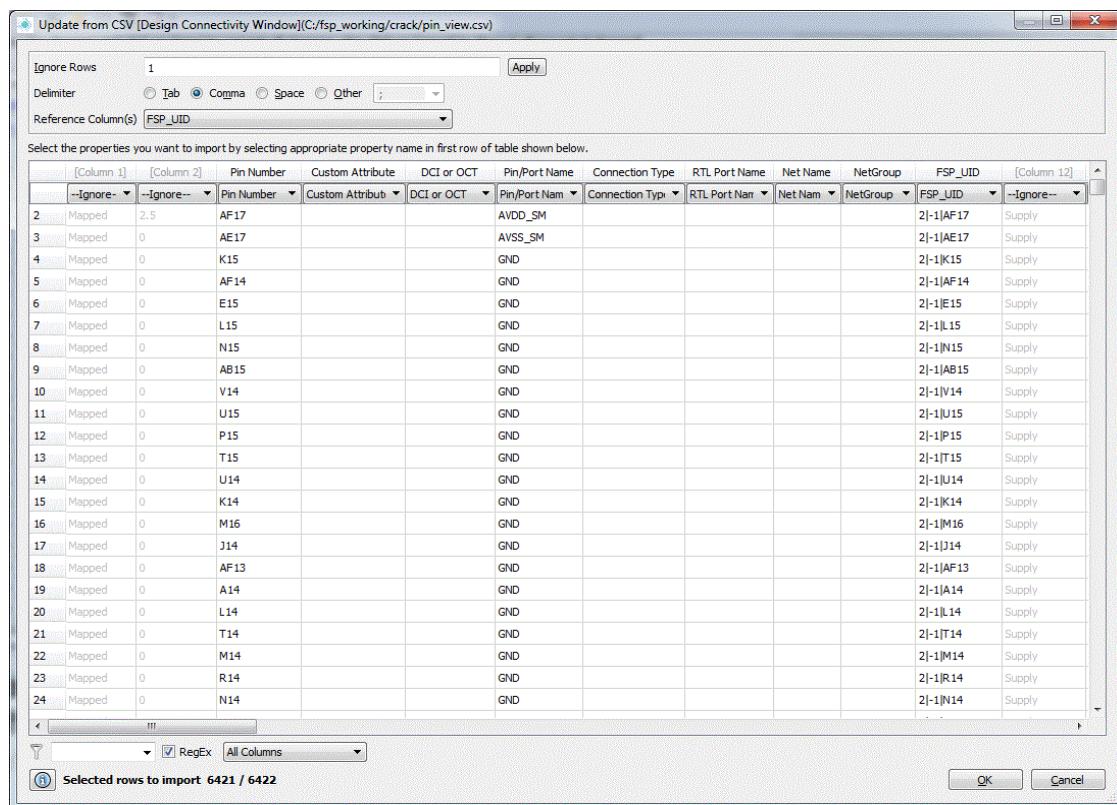
3. In the *Delimiter* field, select an option that should be used to export the data.
4. In the *Select Instances For Export* pane, do the following:
  - Select *Design* or all instances check boxes to include the properties for all the parts and pins in the external file.

Or
  - Select the check boxes of the instances, that you want to export the properties for the selected instances.
5. In the *Select Columns* pane, select the columns that you want to export.
6. Click *Export* to export properties.

## Importing Part and Pin Properties

After modifying properties in a spreadsheet or a text editor of your choice, you can import the modified properties back into the Design Connectivity using the *Update from CSV* dialog box.

### The *Update From CSV* window



The *Update from CSV* dialog box lets you import the modified properties from the external files of different format such as comma separated value (.csv), tab, or space format. The entries in the external file must be in the *Name and Value* format.

When you import properties, it is mandatory to import properties in the view from which the properties were exported. For example, after you export properties from the *Power Pin* view, you switch to *Net View* and import the changed properties. The *Import CSV* process may fail or lead to error-prone state due to mismatched pins.

The *Import CSV* process, does not import all parts and pins properties. For instance, if a property is exported using the *Export CSV* dialog box, it does not imply that you can change its values and then import the change back into the Design Connectivity. The rule is that the

## Allegro FPGA System Planner User Guide

### Exporting and Updating Design

---

any property that is editable in the Design Connectivity and can be imported. In addition, the columns (or property) that are non-editable or read-only cannot be imported. For example, the pin numbers in the *Pin Number* column is exported by the *Export CSV* dialog box. If you edit the pin numbers in the external file and import the file back into Design Connectivity, the pin number does not change.

The following table provides few examples what happens when the field values are different in editable and non-editable columns.

**Note:** The *Italicized* names are read-only (or non editable) column names and the names in **bold** format are editable column names.

Column Name	Columns in Design Connectivity	Columns in external file	results after importing (in Design Connectivity)
<i>Status</i>	Field is null or not null	Field is null or modified	The field value is not effected.
<i>Pin/Port Name</i>	Field is not null	Field is null or modified	The field value is not effected.
<i>Pin Number</i>	Field is not null	Field is null or modified	The field value is not effected.
<i>Pin Type</i>	Field is not null	Field is null or modified	The field value is not effected.
<i>IO Standard</i>	Field is not null	Field is null or modified	The field value is not effected.
<b>Net Name</b>	Field is null or not null	Field is null or modified	The field value changes to value specified in the external file.
<b>RTL Port Name</b>	Field is null or not null	Field is null or modified	The field value changes to value specified in the external file.
<b>Pin Termination</b>			
<b>NetGroup</b>	Field is null or not null	Field is null or modified	The field value changes to value specified in the external file.

## Allegro FPGA System Planner User Guide

### Exporting and Updating Design

---

<i>FSP_UID</i>	Field is null or not null	Field is null or modified	The field value is not effected.
----------------	---------------------------	---------------------------	----------------------------------

---

### Choosing Reference column

When you import properties, the Design Connectivity expects to find the following columns values unchanged:

- Pin Number
- Pin/Port Name
- Target Device
- Instance/Protocol Name
- *FSP\_UID*

The above listed columns are considered as reference columns. The values in these columns are considered as reference or address of the values in the other columns. FSP internally maps the values of the reference columns with the values of the remaining columns. For example, if you select the *Pin Number* column as reference column, then each pin number is mapped with the values of the remaining columns.

Pin Number	NetGroup	Custom A	bank_vref	Part Name	Group Name	Target Dev	Group Constraint	Instance/Protocol Name	Model Constraints
82	XP1_DATA_BYT5			ddr3_dimm_x8_v7	Data_Byt5	U1	memory_data	XP1	use_same_vcaux_io

From the above figure, pin number 82 is internally mapped to the ddr3\_dimm\_x8\_v7, data\_byt5, and memory\_data. When you import, FSP first looks for the pin number 82 and check for any changes in the mapped values, that is, ddr3\_dimm\_x8\_v7, data\_byt5, and memory\_data. If it finds, then replaces the values in Design Connectivity with the values specified in the external file.

As explained in the *Exporting Parts and Pin Properties* section, it is not recommended to modify the values of the reference columns. If you do, the changes will not be considered during *Import CSV* process.

**Note:** When you export properties, the *Target Device* and *Instance/Protocol Name* column values are derived from parts properties that are displayed on the instance-level rows.

The following example demonstrates what values are fetched from the Design Connectivity for *Target Device* and *Instance/Protocol Name* columns.

Consider the part properties displayed in the following figure.

# Allegro FPGA System Planner User Guide

## Exporting and Updating Design

---

Status	Pin/Port Name	Pin Number	Allocated Pin N
<b>Design [Pin View]</b>			
device name= <b>U1</b> part=xc7v2000tfhg1761			
device name= <b>U2</b> part=xc7z101clg225			
device name= <b>U3</b> part=xc7z100ffg1156			
interface model_constraint= <b>use_same_vcaux_io</b> name=XP1 part=ddr3_dimm_x8_v7			
group target_device= <b>U1</b> group_name=Address_Control group_constraint=memory_address			
group group_name=NoConnect			
group group_name=Power			
group target_device= <b>U1</b> group_name=Data_Byte1 group_constraint=memory_data			
group target_device= <b>U1</b> group_name=Data_Byte2 group_constraint=memory_data			
group target_device= <b>U1</b> group_name=Data_Byte3 group_constraint=memory_data			
group target_device= <b>U1</b> group_name=Data_Byte4 group_constraint=memory_data			
group target_device= <b>U1</b> group_name=Data_Byte5 group_constraint=memory_data			
group target_device= <b>U1</b> group_name=Data_Byte6 group_constraint=memory_data			
group target_device= <b>U1</b> group_name=Data_Byte7 group_constraint=memory_data			
group target_device= <b>U1</b> group_name=Data_Byte8 group_constraint=memory_data			
group target_device= <b>U1</b> group_name=Data_Byte9 group_constraint=memory_data			
group group_name=Ground			
group group_name=TempSensor_SPDEEPROM			
protocol name= <b>U2_U1_U3</b>			
group target_device= <b>U1</b> group_name=Data_Byte1 group_constraint=same_bank			
group target_device= <b>U1</b> group_name=Data_Byte2 group_constraint=same_bank			
protocol name= <b>U2_U1_U3</b>			
protocol name= <b>U2_U1_U3</b>			

When you export properties, the *U1* is exported as values in the *Target Device* column and *U1, U2, U3, XP3*, and *U2\_U1\_U3* are exported as values in the *Instance/Protocol* column, as shown below.

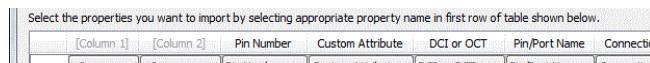
A	B	C	AH	AI
Status	Pin/Port Name	Pin Number	Target Device	Instance/Protocol Name
Unallocated	PS_DDR_DQS_N2_E32		U3	
Unallocated	PS_DDR_DQ8_502_A33		U3	
Unallocated	PS_DDR_DQ0_502_A28		U3	
Allocated	DDR3_RESET	168	U1	XP1
Allocated	DDR3_S0	193	U1	XP1
Allocated	DDR3_S1	76	U1	XP1
Preserve	DDR3_S2	79	U1	XP1
Preserve	DDR3_S3	198	U1	XP1
Allocated	DDR3_WE	73	U1	XP1
Unallocated	NC	126		XP1
Unallocated	NC	135		XP1
Unmapped	VDD	78		XP1
Unmapped	VDDSPD	236		XP1
Unmapped	VREFCA	67		XP1
Unmapped	VREFDQ	1		XP1
Unmapped	VTT	240		XP1
Unmapped	VTT	120		XP1
Allocated	DDR3_DQ<22>	146	U1	XP1
Allocated	DDR3_CB<5>	159	U1	XP1
Allocated	DDR3_CB<6>	164	U1	XP1
Allocated	DDR3_CB<7>	165	U1	XP1
Allocated	DDR3_DM<8>	161	U1	XP1
Allocated	DDR3_DQS8_N	42	U1	XP1
Allocated	DDR3_DQS8_P	43	U1	XP1
Unmapped	VSS	98		XP1
Preserve	DDR3_SA0	117		XP1
Preserve	DDR3_SA1	237		XP1
Preserve	DDR3_SA2	119		XP1
Preserve	DDR3_SCL	118		XP1
Preserve	DDR3_SDA	238		XP1
Allocated	CLK0_N	4000	U1	U2_U1_U3
Allocated	CLK0_P	4001	U1	U2_U1_U3
Allocated	Data<0>	4002	U1	U2_U1_U3
Allocated	Data<1>	4003	U1	U2_U1_U3
Allocated	Data<2>	4004	U1	U2_U1_U3
Allocated	Data<3>	4005	U1	U2_U1_U3
Allocated	Data<4>	4006	U1	U2_U1_U3
Allocated	Data<5>	4007	I11	I13

## Allegro FPGA System Planner User Guide

### Exporting and Updating Design

---

The *Update from CSV* dialog box, provides you an option to select reference columns. Specifying reference column is not same in all the cases.



The following section describes the two cases in which the selection of reference column differs.

- To import instance properties

To import instance properties, you can either select *Pin Number + Instance/Protocol Name* or *Pin/Port Name or Instance/Protocol Name* or *FSP\_UID* options. The *Instance/Protocol Name* column is by default selected for all the designs.

**Note:** In this case, it is not recommended to select the *Target Device* option.

**Example,**

Select the *FSP\_UID* option as reference column. When you import properties, FSP looks for the instance names under the *Instance/Protocol Name* column for which the changes need to be imported. Once FSP has instance name it checks for the ID's in the *FSP\_UID* column.

- To import instance protocol properties

To import instance/device protocol properties, you must select the *Pin Number + Instance/Protocol Name + Target Device* or *Pin/Port Name or Instance/Protocol Name + Target Device* options. You can also select the *FSP\_UID* option as an alternative.

**Example,**

Select the *Pin Number + Instance/Protocol Name + Target Device* options. The *Instance/Protocol Name* column is by default selected for all the designs. When you import properties, for instance protocols FSP looks for the pin number and instance names for which the changes need to be imported and for device protocols it looks for the protocol name.

### Specifying Columns to Import

After you specify reference column options, you must load the properties in the *Update from CSV* dialog box. After you load, the properties are displayed in a read-only spreadsheet view at the bottom of the dialog box. You are not allowed to edit the values in this view.

The columns and their values are displayed in the following way:

## Allegro FPGA System Planner User Guide

### Exporting and Updating Design

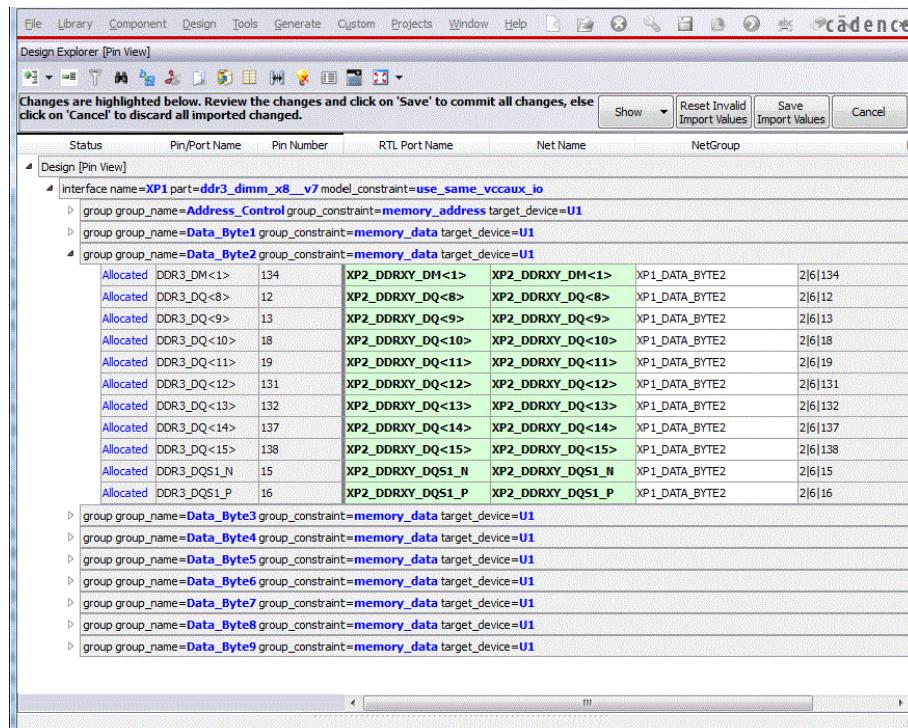
- The reference columns values are displayed in enabled mode.
- All the read-only columns of the Design Connectivity are displayed in disabled mode such as *Status*, *Pin Type*, *IO Standard* and more.
- All the editable columns of the Design Connectivity are displayed in enabled mode such as *Net Name*, *RTL Port Name*, *Regulator Name* and more.

The top row of the spreadsheet provides a drop-down list of the column names. You can select a column name from the drop-down list to import their values.

**Note:** Specifying column names is not applicable for reference columns and read-only columns of the Design Connectivity. For example, if you select *Net Name* from the drop-down list for *Pin/Port Name* column, the values of the *Pin/Port Name* column are unchanged and continue to display the existing values.

## Importing Changed Properties

After you specify columns, you can import the changed properties. When you import properties, an import session starts in the Design Connectivity. In this session, the properties of parts and pins are temporarily highlighted in the Design Connectivity. The buttons in the top right corner of the Design Explorer indicates the begin of the session.



## Allegro FPGA System Planner User Guide

### Exporting and Updating Design

---

During this session, all the main menu bars and **S** are disabled. In addition, you cannot switch to other views either. However, the **S** in the Design Connectivity are available to perform frequent tasks. This session remains active until you commit or save the highlighted properties.

**Note:** When you import properties, if no changes are found in the external file a confirmation window is displayed.

The following table describes the usage of each options that are available during the import session:

Name..	Description..
Show	The following options are available when you click this option: <ul style="list-style-type: none"><li>■ Valid Changes Only Click to see the valid changes.</li><li>■ Invalid Changes Only Click to see the invalid changes</li><li>■ All Changes Click to see valid and invalid changes.</li><li>■ Everything Click to see valid and invalid changes, and the properties that are not changed.</li></ul>
Reset Invalid Import Values	Click to remove the invalid values that are temporarily imported. After you click this option, it resets to previous value.
Save Import Values	Click to save or commit the valid changes. After you click this option, the session exits.
Cancel	Click to exit the session.

During this session, you are allowed to manually modify the highlighted and non-highlighted properties. After you modify the non-highlighted properties, the changes are immediately reflected in different color.

To import properties, perform the following steps:

## Allegro FPGA System Planner User Guide

### Exporting and Updating Design

---

1. Click the *Import CSV* icon in the *Design Connectivity*.
2. The *Select CSV File* dialog box is displayed.
3. Locate to the .csv file and click *Open*.  
The *Update from CSV* dialog box is displayed.
4. In the *Ignore Rows* field, specify the row numbers that you want to ignore during import process.
5. In the *Delimiter* field, select a delimiter that should be used to parse the data.
6. In the *Reference Column* field, do the following:
  - a. To import instance properties:
    - Select the *Pin Number + Instance/Protocol Name* or *Pin/Port Name + Instance /Protocol Name* options.  
Or
    - Select the *FSP\_UID* option.
  - b. To import instance protocol properties:
    - Select the *Pin Number + Instance/Protocol Name + Target Device* or *Pin/Port Name + Instance /Protocol Name + Target Device* options.  
Or
    - Select the *FSP\_UID* option.

The properties are displayed in the spreadsheet view.

7. In the first row, of the spreadsheet view specify the column names from the drop-down list whose values need to be imported.
8. Click *OK* to import the properties.

After you click *OK*, the import session begins and changed properties are highlighted in the *Design Connectivity*. In addition, the four options *Show*, *Reset Invalid Values*, *Save Import Values*, and *Cancel* options are displayed at the top right corner of the *Design Connectivity*.

9. Click *Save Import Values* to save or commit the changed properties.

The session exits and all the highlighted properties are displayed in default color.

---

## **Schematic in Design Entry HDL**

---

After completing the design and verifying the logic, the next step in design process is to create a schematic of the logic design in Design Entry HDL (DE-HDL).

FSP offers full integration with Cadence DE-HDL tool suite, allowing you to use all of FSP's schematic generation capabilities to export the schematic information to DE-HDL for editing schematic.

In FSP, there are a few steps that need to be performed to prepare the schematic for DE-HDL. However, actual tasks related to schematic changes are performed in DE-HDL.

This chapter lists the design tasks and the best practices that must be followed before generating symbols and schematic. This ensures that process of exporting data to DE-HDL is completed smoothly.

The tasks covered in this chapter are:

- [Preparing the Design for Schematic](#)
- [Enabling Name Space Checking](#)
- [Customizing Schematic Symbols](#)
- [Setting Up Symbols](#)
- [Generating DE HDL Symbols](#)
- [Enabling and Disabling DE-HDL Symbol Generation Options](#)
- [Generating DE HDL Schematics](#)

## Preparing the Design for Schematic

Before you export a logical design to DE-HDL, you should validate your design to ensure that the names used in the logical design follow the naming convention required in DE-HDL or PCB Editor. This section lists some of the recommendations or best practices to be followed in FSP to ensure that the logical design is successfully exported to DE-HDL.

### Best practices for FSP - DE-HDL flow

- Use brackets “(),<>,{},[]” in signal names and pin names.
- Naming nets, pins, or parts:
  - Keep the maximum length of a net, pin, or part names or alias up to 31 characters.
  - Do not use special characters in a net names, part names, reference designators or pin names.
- Do not use duplicate names for pins other than power pins and No Connect pins.
- Run the *Check Design Consistency* or *Check Power Connections* command before generating symbols and schematics.
- Set the CPM file path for DE-HDL before generating schematics.
- Set the symbol path and FPGA hierarchical blocks path for DE-HDL before generating symbols.

### Unsupported field values for FSP - DE-HDL Flow

You should avoid the use of following special characters when defining pin names, signal names, or part names in the FSP - DE-HDL flow. Using any one of the following characters in names may fail or halt the symbol or schematic generation process in FSP.

- leading or trailing white spaces
- ! (exclamation mark)
- ' (single-quote)
- . (period)
- \ (backslash)

## Enabling Name Space Checking

Besides the manual validate method, FSP offers a check, *Name Space Check* that automatically validates the part names, signal names, or pin names, before exporting the design to DE-HDL.

The Name Space Check (NMP) mechanism validates all the strings such as signal names, instance names, custom properties, and model name that are to be propagated for various entities used in the FSP design. This ensures that the process of propagating and backannotating the design data within different cadence tools is done smoothly.

For example, when you specify the cell name while creating an FSP model, specify the signals to generate the schematics and verilog files, the NMP Check is used to get the right name space conversion. When you specify the bus signal names in your design, the NMP validates only the signal names not the complete bus signal names. For example if the bus signals are U2\_Address\_Control<1>...U2\_Address\_Control<13>, NMP validates the U2\_Address\_Control.

### Name Space Checking at Various Stages

The NMP Check is used at various stages in the FSP to Design Entry HDL flow to validate and output the signal names and DE-HDL symbol names. The NMP Check is used at following stages:

- Selecting DE-HDL Symbol in the Setup Symbol Data dialog box
- Creating FSP Model Using DE-HDL Symbol
- Generating and Parsing Verilog and VHDL Files
- Generating the Allegro DE-HDL Schematics

#### Selecting DE-HDL Symbol in the Setup Symbol Data dialog box

When you perform the steps below to select the DE-HDL symbol in the Setup Symbol dialog box, the NMP Check is used to validate the cell name of the DE-HDL symbol. One additional libaccess API is used when you browse for the DE-HDL symbol. When you select the DE-HDL symbol FSP creates a temporary cds.lib file in the C:\Temp\ directory. The temporary cds.lib file contains the entire FSP libraries. The libaccess API is used by the FSP to get all the library names and DE-HDL symbol lists part from the generated temporary cds.lib file. The generated cds.lib file is automatically deleted from the C:\Temp\ directory when you close the project.

To select the DE-HDL symbol perform the following steps:

1. Choose *Generate – Setup Symbol Data*.

The Setup DE-HDL Symbols Data dialog box is displayed.

2. Click *browse (...)* in Symbol Path column.

The *Specify DE-HDL Symbol To Generate* dialog box is displayed. After you click *browse (...)* in the Symbol path column a temporary cds.lib file is created in the *C:\Temp* directory.

3. Click the *Library name* option.

After you click the *Library name* option, the FSP library names with DE-HDL symbol names of temporary cds.lib file are displayed.

4. Select a library name and click *OK*.

5. Click *OK*.

### **Creating FSP Model Using DE-HDL Symbol**

The NMP check is also used while creating the FSP model using existing DE-HDL symbol. When you create the FSP model, NMP Check is used to validate the DE-HDL symbol directory name. The NMP Check runs the Library to ASCII command on the DE-HDL symbol directory to get the right name space conversion DE-HDL symbol name. After running the command NMP Check obtains two values, ASCII name space value and Library name space value. If the ASCII name space value and library name space value is same FSP allows you to save the FSP model in the DE-HDL symbol directory. If the ASCII name space value and library name space value is different, the Save As dialog box is displayed to save the FSP model in different directory.

You can create an FSP model from existing DE-HDL symbol by using any one of the following methods:

- Through the Create Part from Allegro DE-HDL Symbol dialog box – This dialog box can be access through Library Menu.
- Through the Right click option in Libraries.

The NMP Check is used by the FSP on both the methods above. For more information on how to create FSP model using existing DE-HDL symbol, see the [Creating Parts from DE-HDL Symbol](#) section.

***Example1:***

The following example demonstrates how NMP check allows you to save the FSP model in the different directory while creating an FSP model using DE-HDL symbol with directory name tmp #20257. In the following example the right-click option in Libraries is used to create the FSP model.

**1. Launch *FSP*.**

The tmp#20257 symbol name is displayed in Libraries.

**2. Right-click *tmp#20257*.**

A pop-up menu is displayed.

**3. Click *Edit Component*.**

The *Rules Editor* window is displayed.

**4. Modify the model property, groups properties, and pin properties as neccessity.**

**5. Click *OK*.**

When you click OK, the NMP Check runs the Library to ASCII command on tmp#20257 directory. The NMP Check receives the value as tmp 20257. Since both library name space value tmp and ASCII value 20257 are different the *Save As* dialog box is displayed.

**6. Select the directory and enter a name for the model.**

**7. Click *OK*.**

**Note:** The NMP Check follows the same checking process if you create a model through the *Create Part – Interface* command.

***Example2:***

The following example demonstrates how NMP Check allows you to save the FSP model in the DE-HDL symbol directory while creating an FSP model using DE-HDL symbol with directory name test123. In the following example the right-click option in Libraries is used to create the FSP model.

**1. Launch *FSP*.**

The test123 symbol name is displayed in Libraries.

**2. Right-click *test123*.**

A pop-up menu is displayed.

**3. Click *Edit Component*.**

The *Edit Rules* window is displayed.

**4. Modify the model property, groups properties, and pin properties as neccessity.**

**5. Click *OK*.**

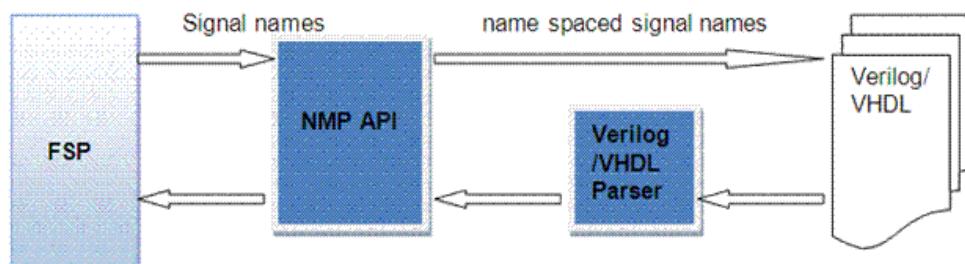
When you click OK, the NMP Check runs the Library to ASCII command on test123 directory. The NMP Check receives the both ASCII value and library name space value as test123. Since both library name space value and ASCII value are same FSP saves the model with test123 model name in the test123 symbol directory.

## **Generating and Parsing Verilog and VHDL Files**

The NMP Check is also used at following stages:

- Generating the Verilog and VHDL files.
- Parsing Verilog and VHDL files while importing.

The following figure describes how NMP Check validates the signal names and port names while generating and parsing the verilog and vhdl files.



## **Generating Verilog and VHDL files**

When you generate the verilog and VHDL files through Generate – Verilog and VHDL Files, the NMP Check validates the signal names, port names of the design that are needed to be written to the verilog and VHDL files. While validating if any signal names are incorrect, the signal names are converted to verilog and vhdl name space and then written to the verilog file.

## Parsing Verilog and VHDL Files

When you import the verilog and vhdl files in FPGA Port and Use Pin Map dialog box, the NMP Check parse and validate the verilog and vhdl files. If any signal names are incorrect, the NMP Check converts the signal names to name space properly and allows the signal names to import.

## Generating the Allegro DE-HDL Schematics

When you create a FSP design, various conventions such as project name, signal names, library names, and entities need to be checked before the design can go to the Design Entry HDL. These various entities are validated by NMP Check while generating the schematics. During NMP Check, the custom attribute property name and value are skipped by NMP Check. NMP Check does not validate the attribute property name and value while generating schematics. After NMP Check, the property name and property values are validated by `fsp_char_support.txt` file.

### ***fsp\_char\_support.txt***

The `fsp_char_support.txt` file is used by FSP during schematic generation. The file contains the set of supported and unsupported characters rules for different schematics entities and used for validation while generating Capture/DE-HDL symbol and schematics. The entities validation through `fsp_char_support.txt` file succeeds over the NMP Check. After NMP Check, the `fsp_char_support.txt` file validates all the strings and entities that are validated and skipped by the NMP checking.

The `fsp_char_support.txt` file is a tab separated ASCII text file located at `Cadence\SPB_XXX\tools\fsp\templates` directory. First column contains the schematic entity names. Second column contains the maximum number of character supported for entity string. Third column contains set of legal characters. The rules are derived from the DE-HDL and OrCAD Naming Rules and Conventions. For more information naming rules and conventions see Naming Rules and Conventions section in Allegro Design Entry HDL Reference Guide.

**Note:** It is recommended that you do not modify the `fsp_char_support.txt` file. The changes in the file may cause inconsistency to the generated schematics.

FSP provides you the option to disable or enable the `fsp_char_support.txt` file validation. To enable/disable the `fsp_char_support.txt` validations perform the following steps:

1. Choose *File – Preference*.

The Preference dialog box is displayed.

2. Click *Design* tab.

3. Do the following

- a. Select *Skip namespace validation while generating symbols and schematics* option if you want to enable the validation.
- b. Unselect *Skip namespace validation while generating symbols and schematics* option if you want to disable the validation.

4. Click *OK*.

## Setting Up Symbols

After adding properties, the next step is to generate the symbols for the instances. However, in order to correctly map a logical component to the schematic environment, you need to specify the symbol information for all the logical components in the design that are to be included in the DE-HDL.

### Specifying Directory Path to Generate Symbols

There are two types of logical components on FSP canvas.

- Components that are placed using *Component Browser*. This means these components are mapped to the associated symbol and footprint files.
- Components that are placed using *Library Explorer*. This means these components are not mapped to any symbol and footprint files.

For more information about how to place a component using *Component Browser*, see the [Adding Interface Component \(DEHDL\)](#) section.

FSP lets you generate a fresh symbol for the design components that are placed using *Library Explorer*. Using the *Setup DE-HDL Symbols Data* window you can use the existing project library or a separate directory where you want to generate fresh symbols. When you invoke the *Setup DE-HDL Symbols Data* window, you will notice that the `fsp_fe_lib:<cell_name>` directory is set for all the components which are not mapped to any symbols.

`fsp_fe_lib` is the default project library name and `<cell_name>` is the default cell name assigned by FSP.

In the *Setup DE-HDL Symbols Data* window, you will notice a text box in DISABLED MODE in the *Symbol Path* column. The DISABLED MODE text box indicates that the instance/part is placed using *Component Browser*, that is, mapped with the symbol and footprint files.

To setup symbol paths, perform the following steps:

1. Choose *Generate – Setup DE-HDL Symbol Data*.

The *Setup DE-HDL Symbol Data* window is displayed.

2. In the *Symbol Path* column, click the *browse (...)* button.

The *Specify DE-HDL Symbol To Generate* dialog box is displayed.

3. Click and select a new library name from the *Library Name* drop-down list.

**Note:** In case you do not want to use the project library, you can create and add your own library, and select from this drop-down list.

4. Enter a new cell name in the *Cell Name* text box.

5. Click *OK*.

After you click *OK*, the new library and cell names are updated in the *text box*.

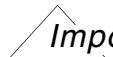
## Customizing Schematic Symbols

FSP provides you a feature to customize the look and feel of a schematic symbol of an instance such as device and interface. Using this feature, you can manipulate the symbol settings and define your own die based on your requirement.

You can do the following operations in *Schematic Symbol Editor*:

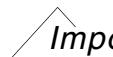
- Create new splits
- Remove existing splits
- Move pins from one split to another split
- Split pins by group (interface) and by bank (device)
- Merge splits
- Change the direction of a pin
- Change the location value of a pin
- Modify custom attributes

When you generate a DE-HDL symbol after customizing, FSP ensures that the symbol is generated based on the defined settings.



***Important***

You cannot customize a symbol of an instance that is placed using Component Browser.



***Important***

If there are multiple instances in the *Setup Symbol Data* form, you must customize them individually.

FSP also provides you various quick menu options in *Schematic Symbol Editor*. These options help you to quickly customize the symbol; however, you can also customize the symbol manually. For example, for interface instance, you can split the pins by group, and for device instance, you can split the pins by bank.

For more information on the menu, see the [Instance Symbol Editor](#) section.

## Cross-Probing between Tree View and Graphics View

In general, for a large schematic symbol, it is difficult to quickly identify interface groups/device banks and their associated pins for debugging purposes. To handle such parts, *Schematic Symbol Editor* provides you two views: Tree view and Graphics view.

A Tree view provides you a spreadsheet-based view. Using this Tree view you quickly modify the symbol pin settings. There are several other right-click options in the tree view that can be used to quickly modify the symbol pin settings in a simple manner.

The Graphics view provides you a graphical view of a symbol group/bank. You can also use this graphics view to view the changes that you make in the Tree view. Besides visualizing, you can change the pin direction of a pin by using a drag-and-drop operation.

The *Schematic Symbol Editor* form maintains the association between these two views. When you make any changes in the Tree view, such as modifying the direction of a pin, merging splits, or moving one pin from one split to another split, the changes are updated in the Graphics view as well. Similarly, when you make any changes in the Graphics view, such as moving a pin from one side to another side of a bank/group, the changes are also updated in the Tree view.

Besides updating the changes between the two views, you can also navigate a pin, group, or bank between the views. When you select a group or a bank in the Tree view, the graphical representation of the selected group/bank is displayed in the Graphics view. Similarly, when

you select a pin in the Tree view, the respective pin is displayed and highlighted in the Graphics view. In the same way, when you select a pin in the Graphics view, the respective pin is displayed in the Tree view.

To customize a symbol, do the following:

1. In the *Setup Symbol DE-HDL Data* window, select a row.

The *Customize Symbol* option at the bottom-left of the window is enabled.

2. Click *Customize Symbol*.

The *Schematic Symbol Editor* window appears.

3. Right-click on the first-level node and select *Add Split Symbol*.

The *Add Split Symbol* dialog box is displayed.

4. Enter a name and click *OK*.

5. Click + button to expand the group/bank of which pins you want to move to the new split.

6. To move one or more pins to the new split, do the following:

- a. Press the Ctrl button and click on the pin names.

- b. Release the Ctrl button, right-click, and choose *Move Selected Pins To Split – <split name>*.

The selected pins are moved to the new split.

7. To move all the pins of a bank/group to the new split, do the following:

- a. Right-click on a pin name and select *Select Pins in Current Selection*.

The pin names in the current bank/group are automatically selected.

- b. Right-click on a pin name and choose *Move Selected Pins To Split – <split name>*.

All the selected pins are moved to the new split.

**Note:** To create multiple splits with different pins, you can repeat steps six to sixteen.

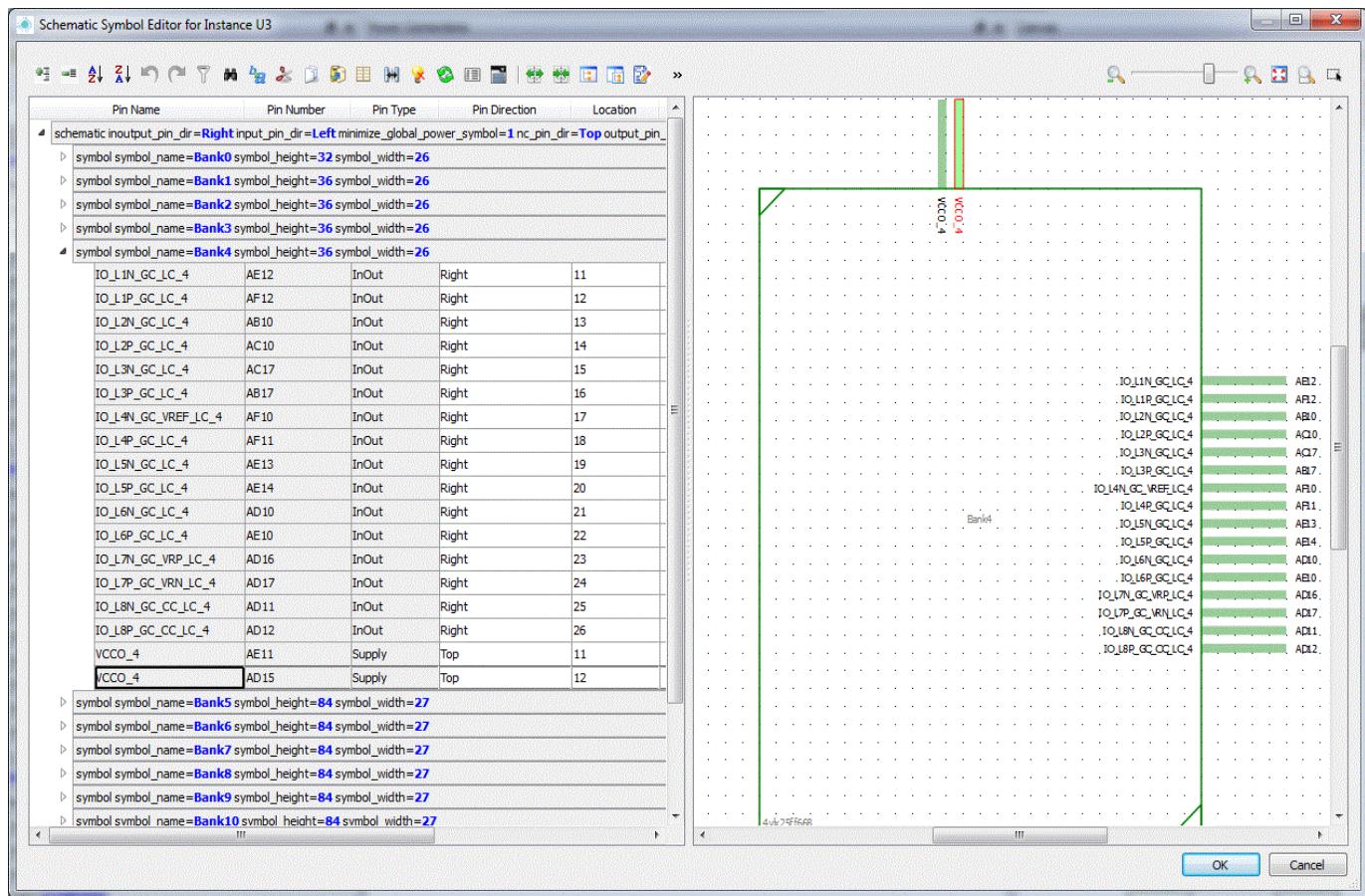
8. Click *OK*.

To change the pin direction of a pin in the Graphics view, do the following:

1. Select a pin whose direction you want to change.

# Allegro FPGA System Planner User Guide

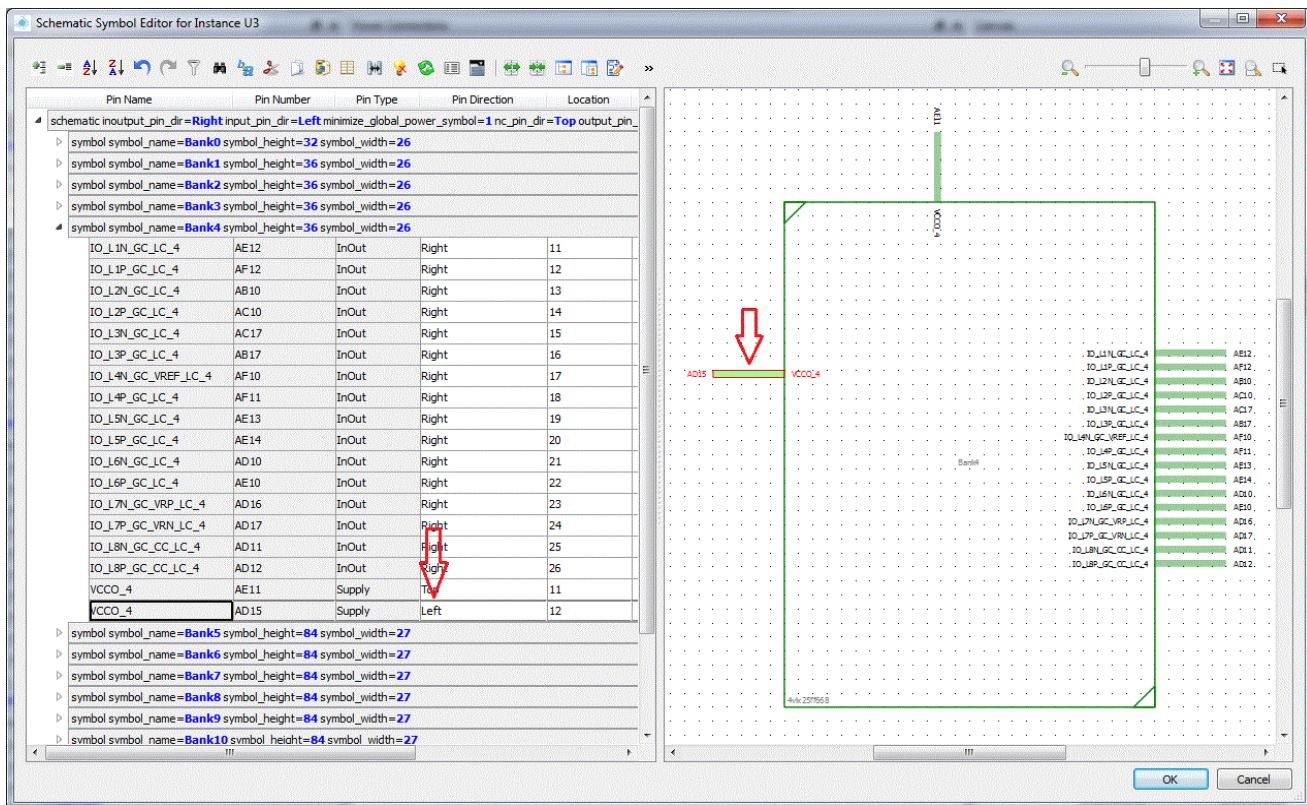
## Schematic in Design Entry HDL



- Drag the pin from one side to another side of the group/bank and drop it.

# Allegro FPGA System Planner User Guide

## Schematic in Design Entry HDL



The new direction of the pin is automatically updated in the Tree view.

## Generating DE HDL Symbols

After setting up the symbol paths and customizing the symbols, you can generate the symbols that represents the instances in your design. FSP reads the logical information of the design instances to create symbols for the schematic. When you generate a new symbol with the *Generate Allegro DE-HDL Symbols* dialog box, FSP creates a chips.prt file. The new chips.prt appears in the fsp\_fe\_lib/<cell\_name>/chips directory. Along with the chips folder, FSP also creates multiple folders such as sym\_1, sym\_2...sym\_n based on the number of banks/groups defined in the *Instance Symbol Editor* dialog box.

The following section displays a hierarchy structure of a device instance's output directory.



Using the *Generate Allegro DE-HDL Symbols* dialog box you can generate a new symbol and update the pin definitions of the existing symbol which allows for engineering change orders (ECOs).

When you invoke the *Generate Allegro DE-HDL Symbols* dialog box, you will notice two following sections:

- Parts with symbols
- Parts without symbols

**Note:** The two sections are displayed based on the availability of symbols for the instances. For example, the *Parts with Symbols* section appears if all the design instances are mapped to the symbols, the *Part without Symbols* section appears if all the design instances does not have symbols in the disk.

### **Parts with Symbols**

This section indicates that the instances are mapped to the symbols. This means that you do not need to generate symbols for the instances. However, FSP lets you regenerate the symbols for the instances. Regenerating symbols will update the changes to the existing symbol, that is, chips.prt file.

### **Parts without Symbols**

This section indicates that the you need to generate the symbols for the instances.

## **Mapping Attributes Names to Properties**

The symbol attributes such as component and component pins that you defined in the *Rules Editor* needs to be transferred to DE-HDL and PCB Editor. For a attribute to flow from FSP to DE-HDL or PCB Editor it needs to be mapped with properties.

The *Generate DE-HDL Symbols* dialog box provides you a list of valid properties that need to be mapped with attributes. These properties are available in the form of a drop-down list in the *Map Custom Attributes To Properties* section. In the *Map Custom Attributes To Properties* section you will notice a column, *Entity*. FSP provides you a list of attribute names for every entity available in the *Entity* column. The *Entity* column, lets you apply the attribute globally on similar types of entities. For example, if you select *Symbol Pin* in *Entity*, all the symbol pins in the design will get the attribute that is specified in the *Attribute Name* column. However, you can also enter your own attribute name or modify the existing one.



In the *Attribute Name* column, you must enter the same attribute name that you have defined in *Rules Editor*.

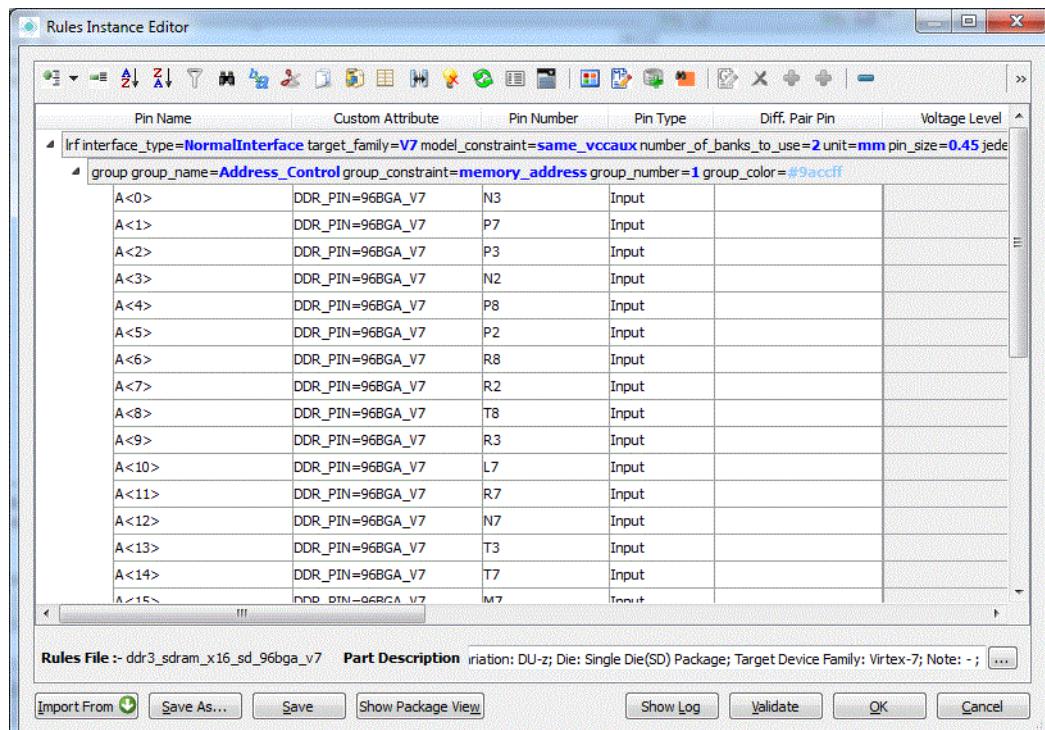
## Allegro FPGA System Planner User Guide

### Schematic in Design Entry HDL

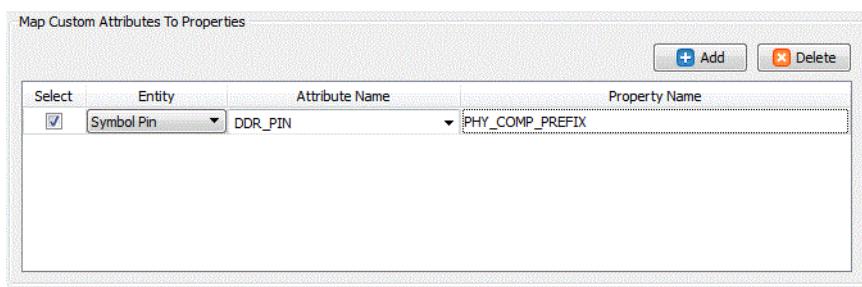
After specifying the attribute name in the *Attribute Name* column, you can specify the property name in the *Property Name* column that is relevant in DE-HDL or PCB Editor.

The following example demonstrates how to map a symbol attribute to a property.

Following figure shows the attribute name and value defined on the component pin.



To map the attribute DDR\_PIN with a property, you must enter the details as shown below.



After propagating the design to DE-HDL or PCB Editor, you see the following details.

---

PHY\_COMP\_PREFIX      96BGA\_V7

---

## Assigning “No Connect” Power Pins

When you take your design to the DE-HDL or PCB Editor, by default all power pins will be shorted together. For example, all the VCC pins are shorted and all GND pins are shorted. This means that on your board you will need to route all the power pins. In the case of large count pin devices, this can be tedious and time consuming task.

To overcome this, you can select an instance whose power pins you want to route and then set all the other power pins as No Connect pins. The *Global Power Prop* feature in the *Generate DE-HDL Symbols* dialog box lets you set No Connect on all the power pins. After you generate the symbols, the No Connect property is included in the chips.prt.

## Steps to Generate DE-HDI Symbols

To generate DE-HDL symbols, perform the following steps:

1. Choose *Generate – Symbols*.
2. The *Generate DE-HDL Symbols* window is displayed.

If you notice *Parts without Symbols* section then perform the following steps:

3. Select the check box for the instances in the *Generate* column for whose you want to generate symbols.
4. Select the check box for the instances in the *Global Power Prop* column for whose power pins you want to add No Connect property.

If you notice *Part with symbols* section then perform the following steps:

5. In the *Re-generate* column, select the check boxes for the instances for whose you want to re-generate the symbols.
6. In the *Global Power Prop* column, select the check boxes for the instances for whose power pins you want to add *No Connect* property.
7. Select *Backup Schematic Symbol* option if you need to back up the previous schematic symbol data.
8. In the *Map Custom Attributes To Properties* section, click *Add* to a new row.
9. Select *check box* to enable the row fields.

10. Select a entity name in the *Entity* column.
11. In the *Attribute Name* column, enter a attribute name.

 **Important**

You must enter the same attribute name that you have defined in other editors such as *Rules Editor*, *Instance Editor*.

12. In the *Property Name* column, enter a property name with which you want to map the attribute.
13. Click *Delete* to delete an existing row.
14. Click *OK*.

After you click *OK*, a series of messages is displayed in the *Message* window. Along with the messages FSP also displays the path of the location where the symbols are created. For the instances for whose you have re-generated the symbols, a backup folder is created at the path <library>/<.backup>/<time stamp>/<cell>/<chips sym>. FSP backups the previous symbols into <.backup> folder, which helps you to reuse any of the backup cell further.

**Note:** FSP backups the previous symbol data only when you select the *Backup Schematic Symbol* data, before re-generating the DE-HDL symbols.

## Enabling and Disabling DE-HDL Symbol Generation Options

In FSP, you can enable or disable the DE-HDL symbol generation process. This is useful, when you want to restrict to use schematic symbols from corporate library, i.e browse and place using Component Browser and do not want to walk through the DE-HDL symbol generation process.

You can disable all the symbol generation options by adding the “is\_allow\_symbol\_generation” variable in the config.ini file.

Set the value to `no` to disable the following options:

- *Setup Symbol Data* in the *Generate* menu
- *Symbols..* in the *Generate* menu

Set the value to `yes` to see the symbol customization and generation options.

**Note:** After adding/removing the variable or modifying the value, restart FSP.

## Creating Split Symbols for Design Block

FSP provides a solution to manage large design block symbols by splitting them into multiple symbols. Instead of generating one large symbol, you can distribute the ports of a design block across multiple symbols. In addition, you can arrange the port locations in a given symbol to any directions as required. This allows user to reduce the size of a design block symbol. In addition, ports can be logically distributed and placed on different symbols to create symbols that can be placed across schematic sheets.

### Points to Remember

You should remember the following points before creating split symbols:

- You must generate a schematic to access the Design Block Symbol Editor forms for the first time. If you open the form without generating the schematic, a warning message appears prompting you to generate the schematic.
- After generating a schematic for the first time, information about the ports in the Full symbol appears in the Design Block Symbol Editor form.

**Note:** The port information of a full symbol always appears in the Design Block Symbol Editor form, unless you split the design block and generate it once.

- After generating a schematic, if you add or delete any external ports in FSP, it is recommended that you regenerate the schematic.
- When you add or delete any external ports in FSP, the ports are adjusted as follows:

<b>FSP Design</b>	<b>Design Block Symbol Editor</b>	<b>Design Block Full Symbol Editor</b>
Delete External Port	External port will be removed.	External port will be removed.
Add External Port	External port will be added to a new split symbol, <i>sym_new_ports</i> .	External port will be added.

### Preserving the Split Symbol Changes

During split symbol creation, if you plan to edit all the ports of a design block, the recommended step is to uncheck the “Preserve Graphics” option and edit the ports. Once you uncheck this option, all the ports are available for editing.

After editing, do not check the “Preserve Graphics” option. If you check it, all the changes made by you will be lost. Once you are done with the editing, you can generate the split symbols information by clicking the Generate option.

Whenever you open the Design Block Split Symbol Editor form, the “Preserve Graphics” option is checked by default. This indicates that the split symbols and ports that already exist on the graphics for the symbols should be retained when regenerating the symbols and schematics. This option also prevents you from making changes to the existing split symbols. This means that you cannot edit the ports, both in the Tree and Graphics views; however, you can edit ports that are added after the last schematic generation. In addition, you can perform all the editing operations that are available in the Tree view on the newly added ports. After editing, it is recommended that you not uncheck the “Preserve Graphics” option. If you uncheck it, all changes will be lost.

## **Cross-probing between Tree view and Graphics view of the Design Block Symbol Editor**

Split symbols are created using the Design Block Symbols Editor. The Design Block Symbol Editor provides two views: Tree view and Graphics view.

The Tree view provides you a spreadsheet-based view of symbol and its ports. You can use this view to quickly create split symbols, change port directions, and merge split symbols.

The Graphics view provides you a graphical view of an active split symbol. You can use this view to see the changes that you make in the Tree view. You can also change the port direction of a pin by using the drag-and-drop method.

## **Creating Split Symbols**

To create a split symbol, do the following:

1. Choose Generate – Edit Design Symbol Block – Splits.

The Design Block Split Symbol Editor form appears and displays a list of the current ports of the design block.

**Note:** If you have created the split symbols earlier, the details of those splits symbols appears in this form.

2. You can split the design block and its ports into various split symbols by choosing one of the following methods:
  - a. Split the symbol manually

- b.** Split the symbol by Virtual Interface
- c.** Split the symbol automatically

**Note:** To access these features, right-click on the schematic row.

**3.** To split the design block manually, do the following:

- a.** Uncheck the “Preserve Graphics” option.
- b.** Right-click on the schematic row and choose Add Split Symbol.

The Add Split Symbol dialog box appears with a default name such as sym\_new\_1.

- c.** Click Ok.

A new row with the new name appears at the bottom of the schematic row.

- d.** Select a set of rows of the existing split and right-click.
- e.** Choose Move Selected Pins to Split - <Split\_Name>.

**Note:** To move all the pins of the existing split to the new split symbol, right-click and choose Move All Pins to Split - <Split\_Name>.

**4.** To split the design block by virtual interface, do the following:

- a.** Right-click on the schematic row and choose Split by VI.

New split symbols are created based on the number of virtual interfaces present on the canvas. The instance name of the virtual interfaces is used as split symbol names, and ports are automatically moved based on the virtual interface pin definition.

Ports that are not part of any virtual interfaces are placed in the Misc symbol.

**5.** To split the design block automatically, do the following:

- a.** Right-click on the schematic row and choose Auto Split Symbol Pins.

The Specify Max Pins per Split dialog box appears. Use this feature when the split symbols have to be created based on the number of ports on each symbol. The number of ports on each symbol is determined by the port count you specify in the Max pins per split field.

**Note:** When you auto split, the existing split symbol will be retained; however, the port count of the existing split symbol will be lower than the specified count. For example, if you have a split symbol with ten ports and specify five in the Max pins per split field then auto split, the number of ports in the existing symbol will be reduced to five and the remaining ports will be used for the new split symbol.

- b.** Enter the port count.
- c.** Click OK.

New split symbols are created and ports are distributed across new split symbols based on the settings you specified in the Max Pins Per Split field.

## Merging Split Symbols

You can merge split symbols by choosing the following methods:

- Merge All Split Symbols
- Merge Split Symbol with Another Symbol

### Merge All Split Symbols

To merge all split symbols, do the following:

1. Right-click on the schematic row and choose *Merge All Splits*.  
A confirmation window appears prompting you to merge the splits.
2. Click Yes to merge all the split symbols.

### Merge Split Symbol with another Symbol

To merge one split symbol with another split symbol, do the following:

1. Right-click on a split symbol and choose *Merge with Split - <Split\_Symbol>*.  
A confirmation window appears prompting you to merge the splits.
2. Click Yes to merge the split symbols.

**Note:** *<Split Symbol>* indicates the split symbol name with which you want to merge the selected split symbol.

## Miscellaneous Operations

The Design Block Symbol Split Editor provides the following miscellaneous operation that you can perform during split symbol creation:

## **Editing Port Directions**

For better distribution of ports, you can change the location of ports to Left, Right, Top, and Bottom using the Edit Pin Direction Settings form. Changing the location ports results in cleaner looking split symbols.

To change the port directions, do the following:

1. Right-click on the schematic node and choose Edit Pin Direction Settings.  
The Edit Port Direction Setting form appears.
  2. To change the Input port direction, click and select a value from the drop-down list.
- Note:** To change the direction for Output and InOut ports follow step 3.
3. Click Ok to save the settings.

**Note:** Follow steps 1 to 3 to edit the port directions for a full symbol.

## **Working with Full Symbol**

FSP lets you manage the ports of full design block symbol using the Full symbol editor dialog box. After schematic generation, full symbol is always in sync with FSP and schematic design; however, for split symbols, you need to update it either manually or follow FSP recommended flow.

### **Points to Remember**

You should remember the following points:

- You must generate a schematic to access the Design Block Full Symbol Editor forms for the first time. If you open the form without generating the schematic, a warning message appears prompting you to generate the schematic.
- After generating a schematic for the first time, information about the ports appear in the Design Block Full Symbol Editor form.
- After generating a schematic, if you add or delete any external ports in FSP, it is recommended that you regenerate the schematic.
- When you add or delete any external ports in FSP, the ports are adjusted as follows:

## **Allegro FPGA System Planner User Guide**

### Schematic in Design Entry HDL

---

Delete External Port	External port will be removed.
Add External Port	External port will be added to a new split symbol, new_split<name>.

---

### **Editing Port Directions**

You can change the location of ports to Left, Right, Top, and Bottom using the Edit Pin Direction Settings form.

For more information, see the Miscellaneous Operations topic in the Creating Split Symbols for Design Block.

## Generating DE HDL Schematics

This section describes about the DE-HDL schematic generation process and the tasks you need to perform to specify the advanced schematic settings to generate the DE-HDL schematics.

The topics covered in this section are:

- [Overview to Generating DE-HDL Schematics](#)
- [Preparing the FSP Design to Generate the Schematics](#)
- [Using CPM Directives](#)
- [Steps for Generating DE-HDL Schematics](#)
- [Understanding DE-HDL Schematic Generation](#)

## Overview to Generating DE-HDL Schematics

FPGA System Planner helps you to generate the DE-HDL schematics for the FSP design. The generated schematics can be used by DE-HDL to capture the design in schematic form. If required you can modify the component placement, or connectivity and these changes can be updated while updating FPGA schematics. FSP provides you various options to manage and control the DE-HDL schematic generation see [Generate Allegro DE-HDL Schematics](#) section.

DE-HDL schematics are generated after creating the pin-outs and power connections in FSP design. The DE-HDL symbol existence is checked by FSP for each of the instances used in the FSP design before generating the DE-HDL schematics. If any of the DE-HDL symbols are missing in library database or if you have not generated the DE-HDL symbols, a warning window is displayed prompting you to generate the DE-HDL symbols for the instances.

Using FSP, you can either generate a hierarchical schematic or flat schematic for your FSP design. In the hierarchical schematic, the design intent is captured in the FSP design blocks and the rest of the design is encapsulated in different hierarchical schematic blocks.

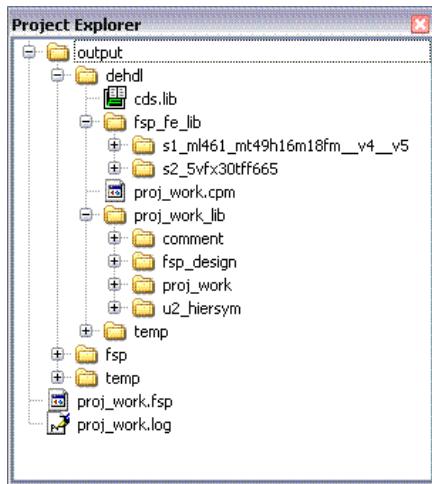
**Note:** Hybrid design is not generated by FSP.

After you have generated the schematic for the first time, for all other subsequent generations you can generate the documentation in the preserve mode. It is recommended that preserve mode must be used in cases where you have made schematic changes such as placement to the generated schematic and do not want these changes to be overridden. For more information, see the [Generating Schematic in Preserve Mode](#) section.

FSP allows you to specify the advanced schematic settings before generating the DE-HDL schematics. The DE-HDL schematic design refers to any schematic information derived from FSP. The schematic information includes design net lists, attribute properties and device files. After generating the DE-HDL schematics from FSP, a schematic view is created in the `./output/allegro/schematics` folder. Schematic view includes all the schematic design files under the associated cell name. After generating the schematics following directory structure appears.

## Allegro FPGA System Planner User Guide

### Schematic in Design Entry HDL



- **dehdl** - Contains the following:
  - DE-HDL project file (cpm file)
  - **fsp\_fe\_lib** directory – Contains hierarchical symbol blocks of the current design instances.
  - <project name\_lib> – During schematic generation by default design library is created with the name <project name>\_lib.

**Note:** You also have the option to modify design library name in Generate Allegro DE-HDL Schematics dialog box.

- **fsp** – Contains the FSP design net name list XML file, log file and .xcon file.

## Preparing the FSP Design to Generate the Schematics

Before generating the DE-HDL schematics for the FSP design, ensure that the following criterion are met:

- DE-HDL symbols are properly generated for each instances used in your FSP design.
- Specify the project information properly required for DE-HDL schematic generation. You must follow the following guidelines when specifying the project informations:
  - White space must not be used in Schematics Output Directory, Project Name, and Schematic Name.
  - Project Names and Schematic Names uses alphanumeric characters. Special characters are not supported, except underscore and hyphen.
  - Do not start the Project Name and Schematic Name with Numbers.

### ***Example***

Correct Names	Incorrect Names
Project1	fsp project1
fsp_project1	2fsp_project
fsp-project2	fsp#%&project

- Specify the colors to be used for drawing schematic objects. To know more about specifying color, see [Generate Allegro DE-HDL Schematics](#) section.
- Ensure that the terminations are properly mapped with associated components before generating the DE-HDL schematics. After termination mapping, if any discrete library is missing the DE-HDL schematic generation may fail.
- Do not modify the generated symbols before generating DE-HDL schematics. This may lead to error prone state.
- Libraries containing the page border and TAP symbols must be defined in the cds.lib file for your project, and must also be added to the list of FSP project libraries. You can add the libraries for the project through Settings dialog box.
  - If the specified page border symbol is not found in the library, the schematic generation process fails.
  - If the default TAP symbols are not found in the library, FSP generates the symbols in the design library where the cell for the root or top-level design exists.

## **Allegro FPGA System Planner User Guide**

### Schematic in Design Entry HDL

---

- The logic grid settings are used for placing components in the schematic pages and to route the design. It is recommended that you use a smaller logic grid, as using a larger logic grid may result in irregular placement and routing in the schematic. The recommended grid size is 50.
- Ensure that the attribute and property names are mapped properly in your design.
- Specify all the Advanced Schematic Settings options required for DE\_HDL schematic generation. To know more about these options, see [Generate Allegro DE-HDL Schematics](#) section.

## Using CPM Directives

This section describes the CPM directives available in the cpm file and discusses how they are processed by FSP during schematic generation.

The `cds.cpm` file located at the `%CDSROOT%\share\cdssetup\projmgr`, by default contains the following FSP and SCM directive sections.

```
START_FSP
page_border 'standard.d size page:sym_1'
wire_color 'Red'
comp_color 'Pink'
END_FSP
```

```
START_DSSCHGEN
wire_color 'Violet'
comp_color 'Yellow'
page_border 'standard.b size page:sym_1'
ctap 'standard.ctap:sym_1'
offpage 'standard.offpage:sym_1'
inport 'standard.inport:sym_1'
outport 'standard.outport:sym_1'
ioport 'standard.ioport:sym_1'
.....
.....
END_DSSCHGEN
```

The FSP directive section contains the following set of default directives which are different from the SCM directives:

```
page_border 'standard.b size page'
comp_to_comp_spacing_in_grids
schgen_grid
nosofgrids_for_bottom_side_separation
nosofgrids_for_left_side_separation
nosofgrids_for_right_side_separation
nosofgrids_for_top_side_separation
nosofgrids_for_maxleftstublength
nosofgrids_for_maxrightstublength
left_in_offpage 'standard.offpage'
left_io_offpage 'standard.offpage'
left_out_offpage 'standard.offpage'
right_in_offpage 'standard.offpage'
```

```
right_io_offpage 'standard.offpage'
right_out_offpage 'standard.offpage'
```

 **Important**

Any other directives you append to the list of default directives in the FSP directive section, are ignored by FSP during schematic generation.

During schematic generation, FSP directives win over SCM directives. By default, FSP directives set the corresponding UI options in the *Generate Allegro DE-HDL Schematic* dialog box. You can also configure the directives based on your requirements and retain the same settings for any project you open irrespective of the directives present in the install-level cpm file, cds.cpm file.

You can configure the FSP directives at the following levels:

- install level (cds.cpm)
- site level (site.cpm)
- project level (<project\_name>.cpm)

The FSP directives in the <project>.cpm file have higher precedence over the other cpm files (install- and site-level) with FSP directives. If no FSP directives are found in any of the cpm file located at different levels, FSP searches for the SCM schgen directives first in the <project>.cpm file.

When you invoke the *Generate Allegro DE-HDL Schematic* dialog box, FSP reads the FSP directives from the <project>.cpm file and displays the corresponding UI options in the dialog box. You can either opt to generate the schematic or modify the UI options and generate the schematic. After the schematic is generated, the FSP directives in the <project\_name>.cpm file is updated with their corresponding values in the dialog box. You can also manually add directives or modify the existing directives in the cpm file using any text editor. The modified directives set the corresponding UI options in the *Generate Allegro DE-HDL Schematic* dialog box.

## Reading Directives from CPM Files at Different Levels

The following section describes the precedence FSP follows to read the directives from the CPM files located at different levels.

### <project>.cpm

This is the first cpm file to be read.

## Allegro FPGA System Planner User Guide

### Schematic in Design Entry HDL

---

After completing the design, when you first time invoke the *Generate Allegro DE-HDL Schematic* dialog box, FSP directives are read from the `site.cpm` file and values are displayed in the corresponding UI options of the dialog box.

**Note:** If no FSP directives are present in the `site.cpm` file, FSP directives are read from the `cds.cpm` file.

When you generate a schematic without changing any UI options of the dialog box, the following SCM schegen directives are copied to the `<project>.cpm` file:

```
START_DSSCHGEN
page_border 'standard.a size page:sym_1'
wire_color 'Green'
schgen_grid '25'
...
...
END_DSSCHGEN
```

If you modify the UI options of the dialog box, the following FSP directives are copied:

```
START_FSP
page_border 'standard.a size page:sym_1';
wire_color 'green'
...
...
END_FSP
```

Now the `<project>.cpm` with FSP directives wins over the other cpm files. From now onwards FSP directives are read from the `<project>.cpm` file and then from the other cpm files in case no FSP directives are found. During the design, if you manually modify the FSP directives in the `<project>.cpm` file, the FSP directives are updated in both the FSP and SCM directive sections after the schematic is generated.

### Example

---

CPM File	Case 1	Case 2
----------	--------	--------

## Allegro FPGA System Planner User Guide

### Schematic in Design Entry HDL

---

<b>site.cpm</b>	<pre> START_DSSCHGEN page_border 'standard.a size page:sym_1' .... .... END_DSSCHGEN </pre>	<pre> START_FSP page_border 'standard.e size page:sym_1'; wire_color 'violet' ... ... END_FSP START_DSSCHGEN .... .... END_DSSCHGEN </pre>
<b>cds.cpm</b>	<pre> START_FSP page_border 'standard.b size page:sym_1'; wire_color 'red' ... ... END_FSP START_DSSCHGEN page_border 'standard.a size page:sym_1' .... .... END_DSSCHGEN </pre>	<pre> START_FSP page_border 'standard.a size page:sym_1'; wire_color 'green' ... ... END_FSP START_DSSCHGEN ... ... END_DSSCHGEN </pre>
<project>.cpm	<p>In this case, the directives are read from FSP section of the cds.cpm file and displayed in the <i>Generate Allegro Schematic</i> form. When you generate the schematic for the first time, the following FSP directives are copied under the SCM directive section of the &lt;project&gt;.cpm file.</p> <pre> START_DSSCHGEN page_border 'standard.b size page:sym_1' wire_color 'red' ... END_DSSCHGEN </pre>	<p>In this case, the directives are read from FSP section of the site.cpm file and displayed in the <i>Generate Allegro Schematic</i> form. When you generate the schematic for the first time, the following FSP directives are copied under the SCM directive section of the &lt;project&gt;.cpm file.</p> <pre> START_DSSCHGEN page_border 'standard.e size page:sym_1'; wire_color 'violet' ... END_DSSCHGEN </pre>

### **\$CDS\_SITE (site.cpm)**

This is the next cpm file to be read in the sequence.

When a project is loaded, FSP and SCM directives in the `site.cpm` file are honored unless they are also specified in the `<project>.cpm` file.

### **\$CDSROOT (cds.cpm)**

After the `site.cpm` file, the `cds.cpm` file is read.

By default, the `cds.cpm` file located at the `$CDSROOT\share\cdssetup\projmgr` includes the FSP directive section. You can append directives to the FSP directive section in this file. Both FSP and Schegen directives in the `cds.cpm` file are honored unless they are also specified in the `<project>.cpm` and `site.cpm` files.

**Note:** It is not recommended to modify the `cds.cpm` file, as it is overwritten when you install an ISR.

### **Example**

The following example shows the sequence in which directives are read and honored from the cpm files at various levels.

---

CPM File	Directive Definition
<code>&lt;project&gt;.cpm</code>	<pre>START_FSP page_border 'standard.a size page:sym_1'; ... ... END_FSP START_DSSCHGEN page_border 'standard.b size page:sym_1' wire_color 'red' schgen_grid '25' .... ... END_DSSCHGEN</pre>

## Allegro FPGA System Planner User Guide

### Schematic in Design Entry HDL

---

```
site.cpm START_FSP
 page_border 'standard.b size page:sym_1';
 wire_color 'green'
 ...
 ...
 END_FSP
 START_DSSCHGEN
 wire_color 'Green'

 END_DSSCHGEN

cds.cpm START_FSP
 page_border 'standard.e size page:sym_1';
 wire_color 'blue'
 ...
 ...
 END_FSP
 START_DSSCHGEN
 page_border 'standard.a size page:sym_1'
 wire_color 'Green'
 schgen_grid '25'

 END_DSSCHGEN
```

---

The following directives are honored:

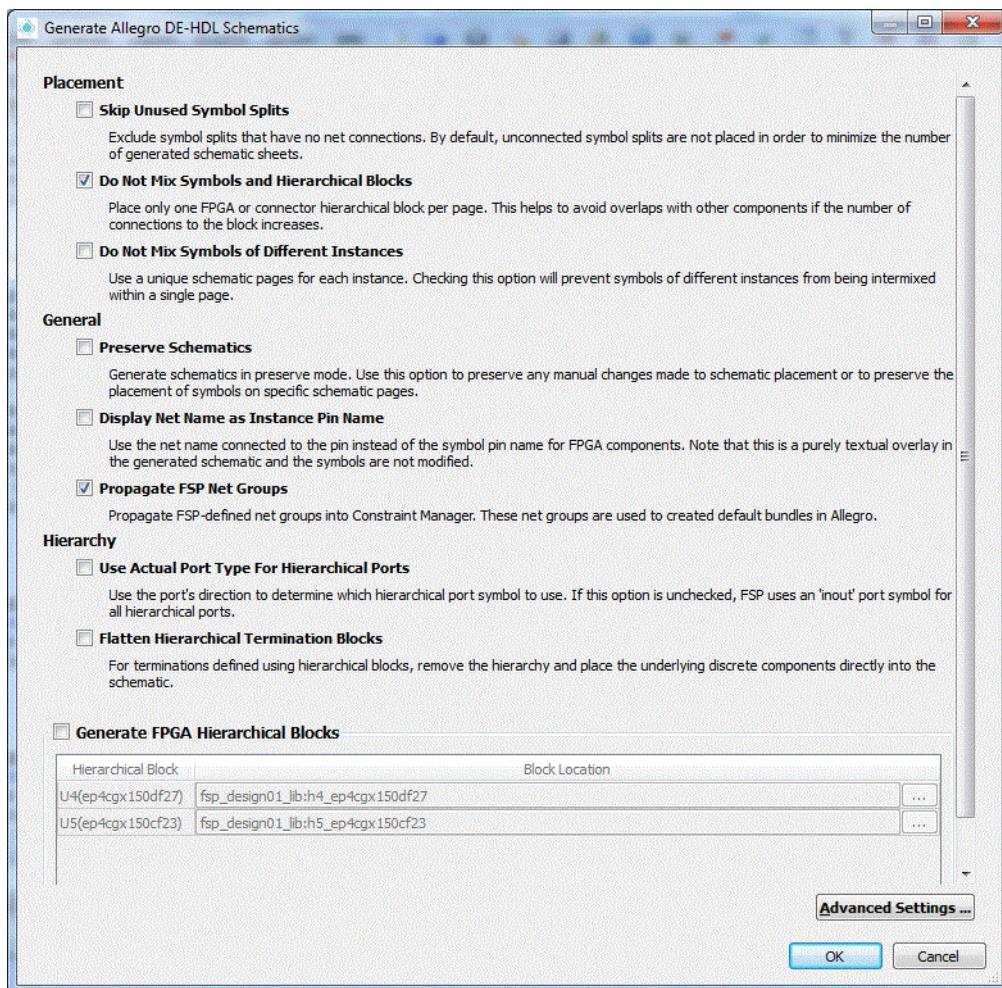
- The directive `page_border 'standard.a size page:sym_1'` in FSP directive section of the `<project>.cpm` file is honored.
- The directive `schgen_grid '25'` in the SCM schgen section of the `<project>.cpm` file is honored.
- The directive `wire_color 'green'` in the FSP section of the `site.cpm` file is honored.

## Steps for Generating DE-HDL Schematics

To generate the DE-HDL schematics complete the following steps:

1. Choose *Generate – Allegro DE-HDL - Schematics*.

The Allegro DE-HDL Schematics dialog box is displayed.



2. Specify the necessary options in the main page and in the *Advanced Settings* dialog box.

For detail information on each tabs, see the [Generate Allegro DE-HDL Schematics](#) on page 736.

3. Click *OK*, to generate the DE-HDL schematics.

## **Allegro FPGA System Planner User Guide**

### Schematic in Design Entry HDL

---

A successful message is displayed in the Messages window along with the path of the schematic directory. The schematic directories are created at the project level.

**Note:** If the schematic generation fails, following are some of the possible reasons for the schematic generation process:

- Ensure that packaging is done properly.

## Understanding DE-HDL Schematic Generation

Schematics are generated based on inputs specified in the Generate Allegro DE-HDL Schematic dialog box. A combination of these inputs determine the way the generated schematic looks. This section describes the basics and the features available in the Generate Allegro DE-HDL Schematic dialog box:

- [Hierarchical Design Vs. Flat Design](#)
- [Placements of Components](#)
- [Capturing Connectivity in the DE-HDL Schematic](#)
- [Representing Associated Components in the Schematic](#)
- [Generating Schematic in Preserve Mode](#)
- [General Features for Schematic Generation](#)
- [Mapping Schematic Attributes to Properties](#)

### Hierarchical Design Vs. Flat Design

Before you generate a schematic for your design, you need to decide the type of design to create. This section introduces you to the two basic schematic design methods, that you can generate using the *Generate Allegro DE-HDL Schematic* dialog box.

- [Hierarchical Design](#)
- [Flat Design](#)

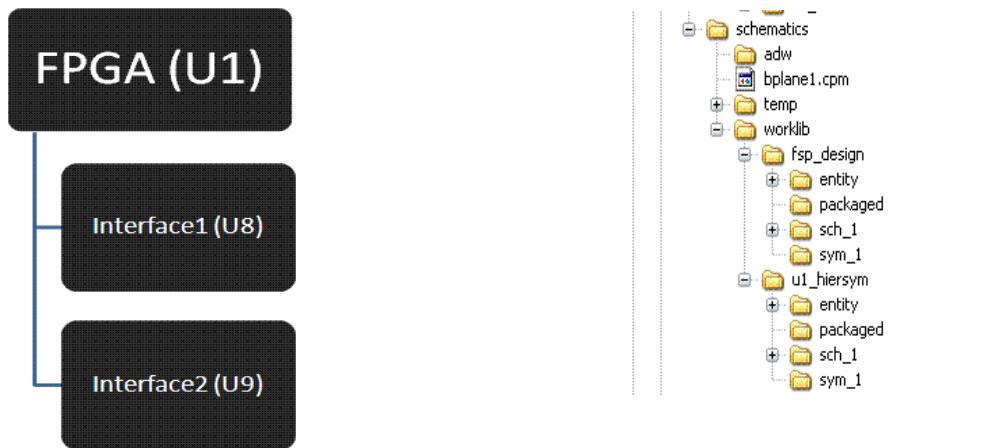
### Hierarchical Design

In Hierarchical design, a design is divided into blocks or sub designs, where each block represents a logical function. Each of the blocks can be further divided into sub blocks. The FSP design ports are defined in the hierarchical block that allow you to connect the design to external circuitry. You can integrate your schematic block in the FSP design project by importing your design on separate pages in the FSP design project. You can then make connections of the FSP hierarchical block with external circuitry. The hierarchical design method is best suited for large and complex designs and when the connectivity between the FPGA and the interface components is not frozen and is likely to undergo some iterations.

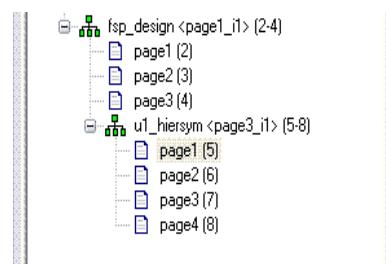
To generate a hierarchical design you select the *Generate a top-Level design block* option in the *Generate Allegro DE-HDL Schematics* dialog box. FSP automatically create a hierarchical based schematic design for your FSP design. The hierarchical design is

generated at the project level. In the hierarchical design, there are two levels of hierarchy, the FPGA block and the rest of the FSP design. FPGA block symbols and other interface components are placed at the same level of hierarchy and interconnected.

Consider a FSP design with one FPGA and two interfaces. If you generate the DE-HDL schematic for the following FSP design, a schematic view is created in FSP as shown in following figure.



The same design is displayed in the hierarchy viewer of the Allegro DE-HDL is shown below.



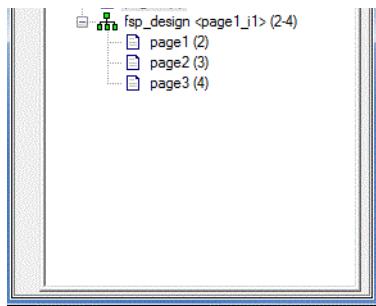
If you generate the DE-HDL schematics for the above example design, the order of the schematic pages in the generated schematic will be listed as below.

1. The first schematic page contains fsp\_design block.
2. Next pages contains four instances block U1, U8, U9.

## Flat Design

A flat design has no hierarchy except fsp\_design block. Regardless of the number of schematic pages in a flat design, all parts appear at the same level of hierarchy. All schematic

pages of a flat design are saved in `sch_1` view of the `<root_design>` cell. The following figure depicts the view of a flat design hierarchy structure.



To generate a flat design, you select the *Generate Flat Schematics* option in the *Generate Allegro DE-HDL Schematics* dialog box. If you uncheck this option, by default a hierarchical design is generated.

## Placements of Components

Placement of components can be controlled using several options available in Generate Allegro DE-HDL Schematics dialog box. The available options are:

- Ignoring Sections of Design in the Schematic
- Placing Unique Component on Same Sheet
- Placing Single FPGA Hierarchical Block Per Page

### Ignoring Sections of Design in the Schematic

When you generate schematic for a FSP design, by default schematic is generated for all the components in your design. However if required, you can specify the components, symbol split and blocks in your design, that you want to ignore during schematic generation process. To ignore the components or symbol splits select the *Skip Unused Symbol splits* option in the Generate Allegro DE-HDL Schematic dialog box.

### Placing Unique Component on Same Sheet

By default, when you generate schematics, set of symbols of different components are placed together in a single same schematic page. These symbols of different components are placed based on the connectivity and tried to be placed in minimum number of schematic pages. If a single schematic page is not enough to accommodate all the symbols then rest of symbols

are placed on the immediate next page. FSP provides you with the ability to specify unique component symbol in a single schematic page. You can use the *Do Not Mix Symbols of Different Instances* option in the Generate Allegro DE-HDL Schematic dialog box. Selecting this option ensures that symbols of different components are not going to be placed together on same schematic pages.

Sometimes, it may happen that your design contain external ports. FSP may generate large hierarchical block symbols of FPGA hierarchical block or FSP hierarchical block. Large hierarchical block symbols can become difficult to manage or place in a single schematic page because of the substantial number of pins coming out of a single symbol. At times, such blocks cannot be placed on the standard page boarder either. To resolve this problem, you can use the Hierarchical Split Symbols (HSS) feature in Allegro Design Entry HDL. This feature manage large hierarchical block symbols by splitting them into multiple split symbols and makes schematic readable. This support reduces the size of the block symbol.

When doing ECO, to use the HSS feature with FSP it is recommended that you follow the following flow:

1. Generate schematics from FSP.
  2. Open the generated schematic in DE-HDL and split the hierarchical block using HSS feature.
- Note:** Do not instantiate or connect the hierarchical block splits.
3. Close DE-HDL.
  4. Regenerate schematics with the *Preserve Schematic* option unchecked.

**Note:** Before regenerating schematics, in case you do any changes such as adding, deleting, or renaming nets in the FSP design, it is recommended that you perform steps one, two, and three.

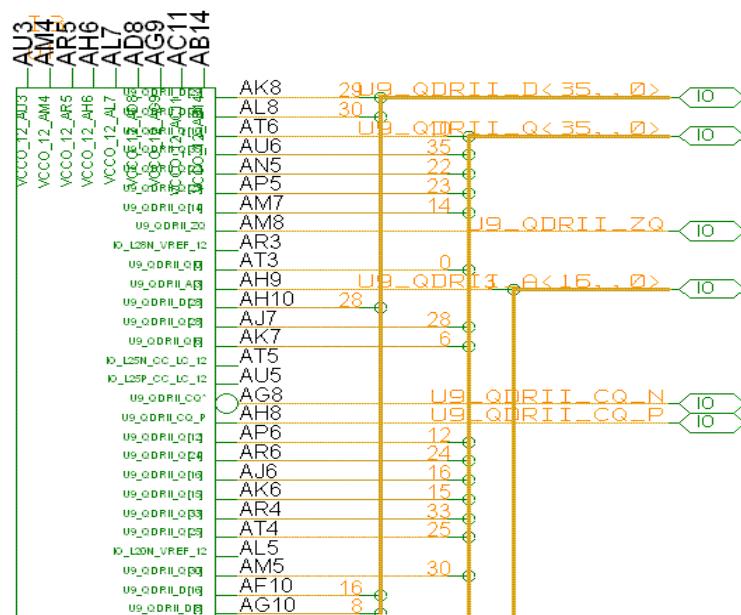
## Placing Single FPGA Hierarchical Block Per Page

By default symbols and hierarchical blocks are placed on same schematic page. You can place single FPGA hierarchical block per page using the Do Not Mix Symbols and Hierarchical Blocks option in the Generate Allegro DE-HDL Schematic dialog box. Selecting this option ensures that FPGA hierarchical blocks are not placed together with any other components on same schematic page.

**Note:** The feature above is also similar for connector.

# Capturing Connectivity in the DE-HDL Schematic

When you generate the DE-HDL schematics using advanced schematic settings, all the vectored signals are shorted to a single signal with same name. The single signal is displayed as thick line in the DE-HDL canvas. For example, in the following figure the signals XP2\_DDR\_DQ<1>, XP2\_DDR\_DQ<2>....XP2\_DDR\_DQ<5> are shorted to XP\_DDR\_DQ<1...5>. See the following figure.



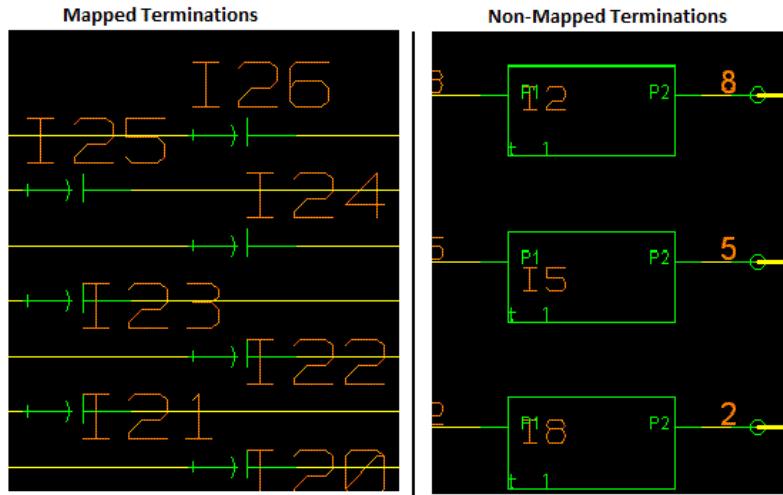
**Note:** The bus net names are displayed as scalar nets, if the bus nets has series or split type termination applied on it.

## Representing Associated Components in the Schematic

In the DE-HDL schematic, simple two port terminations such as series, split and decaps are placed next to the connected pins. This arrangement is also same when these two port terminations are not mapped to discrete components. The difference is mapped terminations are visible as discrete component and non-mapped terminations are visible as blocks in schematic page. See below.

## Allegro FPGA System Planner User Guide

### Schematic in Design Entry HDL



In both the cases, the two port terminations are placed on the same page where it was applied. The other hierarchical termination blocks or complex filters or circuitry may be placed on the same page or next page based on the availability of space on the page.

The following table contains the list of types terminations and their placements details in the DE-HDL schematic page.

Termination Type	Placement Details...
Series	These termination types can be used as hierarchical or direct mapped. If used..
Differential	
Pull Up/Down	<ul style="list-style-type: none"><li>■ As direct mapped (for example, if mapped to resistor or capacitor or any primitive component), then will be displayed in series where it was applied.</li><li>■ As hierarchical blocks, then may place on the same page or on next page according to the available space on the page.</li></ul>
	<b>Note:</b> Flattening option is available for this type of terminations.
Split/Pull Up/Pull Down	These termination types are used as hierarchical blocks and can be placed on the same page or on next page according to the available space on the page.
Differential Series/Parallel	<b>Note:</b> Flattening option is available for this type of terminations.

## Allegro FPGA System Planner User Guide

### Schematic in Design Entry HDL

---

Thevenin, Parallel, Isolvpci	These termination types are used as direct mapped and displayed on the same page.
Power Filter Pull Up/Down	These termination types are used for power connections, placed near to the connected component pin and displayed on the same page.
Power Filter Thevenin	

---

## Generating Schematic in Preserve Mode

In the FSP - Allegro flow, it is important that the changes you do in the downstream tools, that is, Design Entry HDL and Allegro PCB Editor, are preserved while iterating over design changes in FSP. In FSP, schematic generation is one of the key tasks to bring the logical, the schematic, and the board design in sync. However, due to multiple iterations of schematic regeneration the schematic symbol placement and the component reference designator may not be retained in the schematic. This could lead to components and etch rip up in the Allegro board. This happens due to changes are propagated from the schematic to Allegro.

In general, the following changes could be lost during successive regeneration of a schematic to propagate the changes made in the FSP design:

- In the DE-HDL Schematic:
  - Minor text update
- In the Allegro Board:
  - Layout placement of passive and non-passive FSP components
  - Component routing

You can overcome this problem by regenerating the schematic in the preserve mode in FSP. This functionality is provided through the *Preserve Schematics* option in the *Generate Allegro DE-HDL Schematics* dialog box. This functionality is useful if you are regenerating a schematic and want to preserve the modification done to the schematic generated initially. When the schematic is generated in the Preserve mode, the placement of components and associated components, both passive and active components is preserved in the previously generated schematic pages. This helps you avoid component rip up in Allegro, when the board file is updated with the regenerated schematic from FSP.

## Preserving Schematic in the Front and Back flow

In the FSP - Allegro flow, whether you make changes in FPGA System Planner, Allegro Design Entry, or in Allegro PCB Editor, it is important that the schematics are regenerated in

the Preserve mode in order to preserve the placement of components in the generated schematic. The following two entities are preserved in the generated schematic sheet:

- Component reference designator
- Component placement in schematic hierarchy

### **Recommendation for Preserving Schematic**

To preserve the component reference designator and placement in the schematics, the following sequence of tasks is recommended:

After generating the schematics from FSP, you may continue to make changes in DE-HDL such as rearranging, adding or deleting passive and active. After modifying the design, you can package the design and open the Allegro board. In Allegro PCB Editor, you can perform the reference designator and routing changes.

**Note:** You may also synchronize the placement of passive and active components from FSP to Allegro, using the `place fsp` command in Allegro.

If you want to propagate the changes made in FSP to downstream tools, you must first bring FSP and Allegro designs in sync. You can synchronize the designs by importing the Allegro board in FSP. After synchronizing the FSP design with the Allegro board, you can propagate the FSP design changes to the schematic by regenerating the schematic with the *Preserve Schematics* option selected in the *Generate Allegro DE-HDL Schematics* dialog box.

You need to perform the following tasks to preserve the modifications done to the schematic generated initially.

You can make changes in Design Entry HDL and preserve the original changes by performing the following steps:

- a. Update the board by running *Export Physical* command from Allegro Project Manager.  
For more information on running *Export Physical*, see the [Updating the Board with the Changes in Logical Design](#) on page 477.
- b. Update the logical design using board file. Choose *File – Update Design from Board* in FSP.  
For more information on updating the logical design, see the [Updating the Logical Design with the Changes in the Board](#) on page 480.
- c. Regenerate the schematic in the Preserve mode. Choose *Generate – Schematics* with *Preserve Schematics* option selected in FSP.

You can make the layout changes in Allegro PCB Editor and preserve the original changes by performing the following steps:

- a. Update the logical design using the board file. Choose *File – Update Design from Board* in FSP.

For more information on updating the logical design, see the [Updating the Logical Design with the Changes in the Board](#) on page 480.

- b. Regenerate the schematics in the Preserve mode. Choose *Generate – Schematics* with *Preserve Schematics* option selected in FSP.

You can make the changes in FSP and preserve the modification done to the generated schematic and layout.

- Regenerate the schematic in the preserve mode. Choose *Generate – Schematics* with *Preserve Schematics* option selected in FSP.

## General Features for Schematic Generation

The Generate Allegro DE-HDL Schematics form has the following additional features:

- [Flattening Hierarchical Termination Blocks](#)
- [Using Net Name as Instance Pin Name](#)
- [Using Actual Port Type for Hierarchical Ports](#)

### Flattening Hierarchical Termination Blocks

Representing hierarchical termination blocks as complex circuitry on schematic page is known as flattening of hierarchical termination blocks. This is true only when you map a selected termination block terminals with complex circuitry ports. Otherwise an unmapped hierarchical termination blocks will be shown as blocks in the schematic page. While generating DE-HDL schematics, you use *Flattening Hierarchical Blocks* option in Placement tab of Generate Allegro DE-HDL Schematics dialog box.

Not all the available terminations in the Define Termination dialog box are used as hierarchical termination blocks. For detailed information, see [Termination Types](#) section.

### Using Net Name as Instance Pin Name

By default, FPGA component pin names are used as pin names for symbol in schematic. FSP lets you use the net names connected to FPGA pins as symbol pin names in schematic. You

use Display Net Name as Instance Pin Name option in the Generate Allegro DE-HDL Schematic dialog box. This feature is purely for textual purpose in the generated schematic and does not modify the symbols.

## Using Actual Port Type for Hierarchical Ports

By default, InOut pin type is used as port symbol for all hierarchical ports and top-level block ports. FSP lets you use the FPGA symbol port directions for FPGA hierarchical ports and for top-level block, the ports are decided based on the connectivity between the interface and the FPGA components. You use the Use Actual Port Type for Hierarchical Ports option in the Generate Allegro DE-HDL Schematics dialog box.

## Mapping Schematic Attributes to Properties

The Custom Attributes pane allows you to map the schematic attributes. Schematic attributes are the properties that are attached to nets, instances, and instance pins in your design. FSP supports both pre-defined and User - defined attributes. After mapping the attributes, all the property names and values that are defined in your design are transferable when you generate the DE-HDL schematic. You can see the property names and values in DE-HDL that are defined in FSP. The attribute names and values are user purpose. You may change or use the property names and values as per your need.

The following section describes how you can map the attributes in Advanced Schematic Settings dialog box before generating the DE-HDL schematics:

- [Mapping Attributes in the Design](#)
- [Modifying the Attributes](#)
- [Mapping Predefined Attribute Names](#)

### Mapping Attributes in the Design

If the instances of a component have the attribute name with value defined in your design, and if you want to see property name and value in the DE-HDL you need to map the attribute name.

The following example describes you how mapping attributes works at various stages:

### **Example**

If the instances have the attribute name with property value, and if you want to see the property value in DE-HDL you must map the specified attribute name.

In Properties of Device Instance dialog box if you define as

Attribute Name	Value
INST_ATTR1	MYINST1

You must map the INST\_ATTR1 to get the value MYINST1. In Custom Attribute pane of Advanced Schematic Settings dialog box specify as following:

Attribute Name	Property Name
INST_ATTR1	FSP_INST1

When you generate the DE-HDL schematic, instance and open the design in DE-HDL, following is displayed.

FSP_INST1	MYINST1
-----------	---------

### **Note:**

- If the multiple instances have attribute name with the same property value, all the property values matching with attribute name is visible in DE-HDL.
- If the instance have different attribute names with same property value, you must map all the attribute names to see the property values in DE-HDL.

### **Steps to Map Attributes**

1. Click *Advanced Settings* in the Allegro DE-HDL schematics dialog box.

The Advanced Schematic Settings dialog box is displayed. By default the Custom Attributes tab is displayed.

2. Select *Select* check box to enable the Entity, Attribute Name, and Property Name fields.
3. Select *Entity* on which you have defined the attribute name and property value in your design.
4. Click and enter the attribute name that you have defined in the design and you want to map.

5. Click and enter the property name.
6. Click *OK*.

**Note:** You can also cut, copy and paste property values.

### **Modifying the Attributes**

To modify the attributes perform the following steps:

1. Choose *Generate – Schematics*.  
The Allegro DE-HDL Schematic dialog box is displayed.
2. Click *Advanced Settings*.  
The Edit CPM Settings dialog box is displayed.
3. Click *Custom Attributes*.
4. In the *Entity* field, click and select a new entity if you want to select a different entity.
5. In the *Attribute* field, enter a new attribute name of the attribute that you want to map.
6. In the *Property* field, enter a new property name.
7. Click *OK*.

### **Mapping Predefined Attribute Names**

FSP lets you to use the predefined attribute names for mapping. FSP provides you various standard predefined attribute names that you can use for map the property value. Click Attribute Name field in Advanced Schematic Settings dialog box to see the predefined attribute names. You must use the same predefined attribute names while specifying the property values at various places in your design.

To use the predefined attribute names perform the following steps:

1. Choose *Generate – Schematics*.  
The Allegro DE-HDL Schematic dialog box is displayed.
2. Click *Advanced Settings*.  
The Edit CPM Settings dialog box is displayed.
3. Click *Custom Attributes*.

4. In the *Entity* field, click and select a new entity.
5. In the *Attribute* field, click and select a attribute name.
6. In the *Property* field, enter a property name.
7. Click *OK*.



The following messages may generate while working with Allegro DE-HDL Schematic dialog box.

Message	Description
Not all the symbol data available for schematic creation, please generate/create the symbols.	This message appears when you are generating the Allegro DE HDL schematics for the design and the symbols for all the components in the design are not available. First generate or select the symbols and then generate the schematic.
Externals Nets have been modified since last schematic generation. Do you want to update the top level schematics for consistency?	This message appears when you are regenerating the regenerate the FSP schematic after connecting some of the external nets to internal nets only

## Files Created after Generating Schematics

After generating schematic, the Log window displays a successful schematic generation report along with the path of the directory (<output>/dehdl) where the files are generated. Click the link to go to the directory. You see the following files and folders:

---

File and Folder Name	Description
----------------------	-------------

## Allegro FPGA System Planner User Guide

### Schematic in Design Entry HDL

---

<project>.cpm

This is a DE-HDL project file. The project file is used by FSP to invoke Component Browser and for DE-HDL symbols and schematic generation. The default generated <project\_name>.cpm file contains the following minimum settings required for the project:

- The name of the current design and design library.
- The list of DE-HDL symbol libraries.
- The location of the temporary directory where tool generates the temporary files.

You can also point your own <project\_name>.cpm file of the master board schematics in FSP. While specifying the cpm file, the cds.lib file with all the libraries entries (which are listed in cpm file) should be available in the same directory as cpm file.

cds.lib

The cds.lib file contains the list of DE-HDL symbol libraries which you are going to browse through the Component Browser in FSP. The file contains the logical names of the DE-HDL symbol libraries and their physical locations. The cds.lib file also contains the local library name as fsp\_fe\_lib pointed to local library of the current project and design library name <design\_name>.lib used as design library.

fsp\_fe\_lib

The fsp\_fe\_lib directory is a local directory. This local library is used by FSP for symbol generation. You need to specify the local library name in Settings dialog box in order to generate the symbols.

## Allegro FPGA System Planner User Guide

### Schematic in Design Entry HDL

---

<project\_name>\_lib

The <project\_name>\_lib is also a local directory. You may also use this directory for symbol generation. To do that, you need to select <project\_name>\_lib option in the *Generate FPGA Hierarchical Blocks in* of Setting dialog box.

By default the schematic files are generated in this directory. If you want you can customize the schematic generation setting in *Block Location* field of the *Generate Allegro DE-HDL Schematic* dialog box. The <project\_name>\_lib directory is further organized into following sub directories:

- **fsp\_design**

The FSP block schematic is created in the sch\_1 view of this directory.

- **<instance\_name>\_<part\_name>**

FSP generates the hierarchical block schematics based on the number of FPGA instance used in the design. For example, if you use two FPGAs, then FSP generates hierarchical schematic pages separately for both the instances.

- **<project\_name>**

FSP generates the root block schematic in the sch\_1 view of this folder.

---

---

## **Schematic in OrCAD Capture**

---

After completing the design and verifying the logic, FSP provides you another option to create a schematic of the logic design in OrCAD Capture.

FSP offers full integration with Cadence DE-HDL tool suite, allowing you to use all of FSP's schematic generation capabilities to export the schematic information to OrCAD for editing schematic.

In FSP, there are a few steps that need to be performed to prepare the schematic for OrCAD. However, actual tasks related to schematic changes are performed in OrCAD.

This chapter lists the design tasks and the best practices that must be followed before generating symbols and schematic. This ensures that process of exporting data to OrCAD is completed smoothly.

The tasks covered in this chapter are:

## Preparing the Design for Schematic

Before you export a logical design to OrCAD, you should validate your design to ensure that the names used in the logical design follow the naming convention required in OrCAD or PCB Editor. This sections lists some of the recommendations or best practices to be followed in FSP to ensure that the logical design is successfully exported to OrCAD.

For more information, see the [Preparing the Design for Schematic](#) section.

## Enabling Name Space Checking

For more information, see the [Enabling Name Space Checking](#) section.

## Setting Up Symbols

After adding properties, the next step is to generate the symbols for the instances. However, in order to correctly map a logical component to the schematic environment, you need to specify the symbol information for all the logical design parts that are to be included in the OrCAD.

### Specifying Directory Path to Generate Symbols

There are two types of logical components on FSP canvas.

- Components that are placed using *Component Browser*. This means these components are mapped to the associated symbol and footprint files.
- Components that are placed using *Library Explorer*. This means these components are not mapped to any symbol and footprint files.

For more information about how to place a component using *Component Browser*, see the [Adding Interface Component \(DEHDL\)](#) section.

FSP lets you generate a fresh symbol for the design components that are placed using *Library Explorer*. Using the *Setup DE-HDL Symbols Data* window you can use the existing project library or a separate directory where you want to generate fresh symbols. When you invoke the *Setup DE-HDL Symbols Data* window, you will notice that the `fsp_fe_lib:<cell_name>` directory is set for all the components which are not mapped to any symbols.

fsp\_fe\_lib is the default project library name and <cell\_name> is the default cell name assigned by FSP.

In the *Setup DE-HDL Symbols Data* window, you will notice a text box in DISABLED MODE in the *Symbol Path* column. The DISABLED MODE text box indicates that the instance/part is placed using *Component Browser*, that is, mapped with the symbol and footprint files.

To setup symbol paths, perform the following steps:

1. Choose *Generate – Setup DE-HDL Symbol Data*.

The *Setup DE-HDL Symbol Data* window is displayed.

2. In the *Symbol Path* column, click the *browse (...)* button.

The *Specify DE-HDL Symbol To Generate* dialog box is displayed.

3. Click and select a new library name from the *Library Name* drop-down list.

**Note:** In case you do not want to use the project library, you can create and add your own library, and select from this drop-down list.

4. Enter a new cell name in the *Cell Name* text box.

5. Click *OK*.

After you click *OK*, the new library and cell names are updated in the *text box*.

## Customizing Symbols

The process of customizing the symbols is similar in both DE-HDL and OrCAD schematic environments. For more information about how to customize a symbol, see the [Customizing Schematic Symbols](#) section.

## Generating OrCAD Symbols

After setting up the symbol paths and customizing the symbols, you can generate the symbols that represents the instances in your design. FSP reads the logical information of the design instances to create symbols for the schematic. When you generate a new symbol using the *Generate OrCAD Symbols* dialog box, FSP creates a .OLB file in the directory that is specified in the *Symbol Path* column in *Setup OrCAD Symbols Data* window.

Using the *Generate OrCAD Symbols* dialog box you can generate a new symbol and update the pin definitions of the existing symbol which allows for engineering change orders (ECOs).

When you invoke the *Generate OrCAD Symbols* dialog box, you will notice two following sections:

- Parts with symbols
- Parts without symbols

**Note:** The two sections are displayed based on the availability of symbols for the instances. For example, the *Parts with Symbols* section appears if all the design instances are mapped to the symbols, the *Part without Symbols* section appears if all the design instances does not have symbols in the disk.

### Parts with Symbols

This section indicates that the instances are mapped to the symbols. This means that you do not need to generate symbols for the instances. However, FSP lets you regenerate the symbols for the instances. Regenerating symbols will update the changes to the existing symbol, that is, .OLB file.

### Parts without Symbols

This section indicates that the you need to generate the symbols for the instances.

## Mapping Attributes Names to Properties

The process of mapping attribute names to properties are similar in both DE-HDL and OrCAD schematic environements. For more information about how to map attribute names to properties, see the [Mapping Attributes Names to Properties](#) section.

## Steps to Generate OrCAD Symbols

To generate OrCAD symbols, perform the following steps:

1. Choose *Generate – Symbols*.

The *Generate OrCAD Symbols* window is displayed.

If you notice *Parts without Symbols* section then perform the following steps:

2. Select the check box for the instances in the *Generate* column for whose you want to generate symbols.
3. Select the check box for the instances in the *Global Power Prop* column for whose power pins you want to add No Connect property.

If you notice *Part with symbols* section then perform the following steps:

4. In the *Re-generate* column, select the check boxes for the instances for whose you want to re-generate the symbols.
5. In the *Global Power Prop* column, select the check boxes for the instances for whose power pins you want to add *No Connect* property.
6. Select the *Backup Schematic Symbol* option if you need to back up the previous schematic symbol data.
7. In the *Map Custom Attributes To Properties* section, click *Add* to a new row.
8. Select *check box* to enable the row fields.
9. Select a entity name in the *Entity* column.
10. In the *Attribute Name* column, enter a attribute name.



You must enter the same attribute name that you have defined in other editors such as *Rules Editor*, *Instance Editor*.

11. In the *Property Name* column, enter a property name with which you want to map the attribute.
12. Click *Delete* to delete an existing row.
13. Click *OK*.

After you click *OK*, a series of messages is displayed in the *Message* window. Along with the messages FSP also displays the path of the location where the symbols are created. For the

## **Allegro FPGA System Planner User Guide**

### Schematic in OrCAD Capture

---

instances for whose you have re-generated the symbols, a backup folder is created. FSP backups the previous symbols into <.backup> folder, which helps you to reuse any of the backup cell further.

**Note:** FSP backups the previous symbol data only when you select the *Backup Schematic Symbol* data, before re-generating the OrCAD symbols.

## Generating OrCAD Schematics

This section describes about the OrCAD schematic generation process and the tasks you need to perform to generate the OrCAD schematics.

FSP helps you to generate the OrCAD schematics for the logical design. The generated schematics can be used by OrCAD to capture the design in schematic form. If required you can modify the component placement, or connectivity and these changes can be updated while updating FPGA schematics.

OrCAD schematics are generated after creating the pin-outs and power connections in FSP design. The OrCAD symbol existence is checked by FSP for each of the instances used in the FSP design before generating the OrCAD schematics. If any of the OrCAD symbols are missing in library database or if you have not generated the OrCAD symbols, a warning window is displayed prompting you to generate the OrCAD symbols for the instances.

**Note:** FSP reads few directive values such as pin to pin spacing value, from `capture.ini` file located at `<CDSROOT>/tools/fsp/templates/orcad`. FSP also refers to `config.ini` file (site-level). If the values are found in the `config.ini` file, then FSP overrides the default values found in `capture.ini` file. This make sure that the value of directives are not left empty and `config.ini` values are set to the schematic page.

### Understanding OrCAD Schematic generation

Schematics are generate based on inputs specified in the *Generate OrCAD Schematic* window. A combination of these inputs determine the way the generated schematic looks. This section describes the basics and the features available in the *Generate OrCAD Schematic* window:

### Create Top Level Design

FSP will create an Hierarchical block that represents the Top Level of the FSP-related circuitry. If nets in the FSP-related logic connect to other, user-created portions of the design, pins will be added to the hierarchical block (named using the associated net name), and ports will be created on the underlying schematics. This makes it very easy to make connections between components in FSP and components that lie outside of FSP. To do so,

1. Select the Create Top Level Design check box.
2. Enter the Top Level Schematic name in the text box. (`root` is the default name)
3. Select the Create symbol graphics check box to create symbols graphically.

## Place Termination blocks in separate page

FSP lets you place the Termination blocks on a separate schematic page. To do so, select this option.

## Generate / Specify FPGA symbol Location

To generate the schematics, select the check boxes of the parts under the Generate column.

For a Device you can specify the path for the generation of schematics under the Specify FPGA symbol location column. To do this,

1. Click the ... button. A new window called Symbol Part appears on screen
2. Select the Browse button, and select the .olb file to generate the schematics in that particular Instance folder directory.
3. Enter the Part Name and click Finish.

For Interface there is no need to specify the locations.

To generate OrCAD Schematics,

1. On the Generate menu, point to OrCAD and click Schematics. The OrCAD Schematics dialog box appears on screen.
2. On the OrCAD Schematics dialog box, select the parts and click OK.

## Skip Unused Splits

Split (Banks/Groups) which doesn't have any connections can be skipped through this feature, to do so

1. Select the check box under the Skip Unused Splits column for the parts for which you do want to use the splits.

**Note:** These check boxes are enabled for the instance's (interface and FPGA) which have connections.

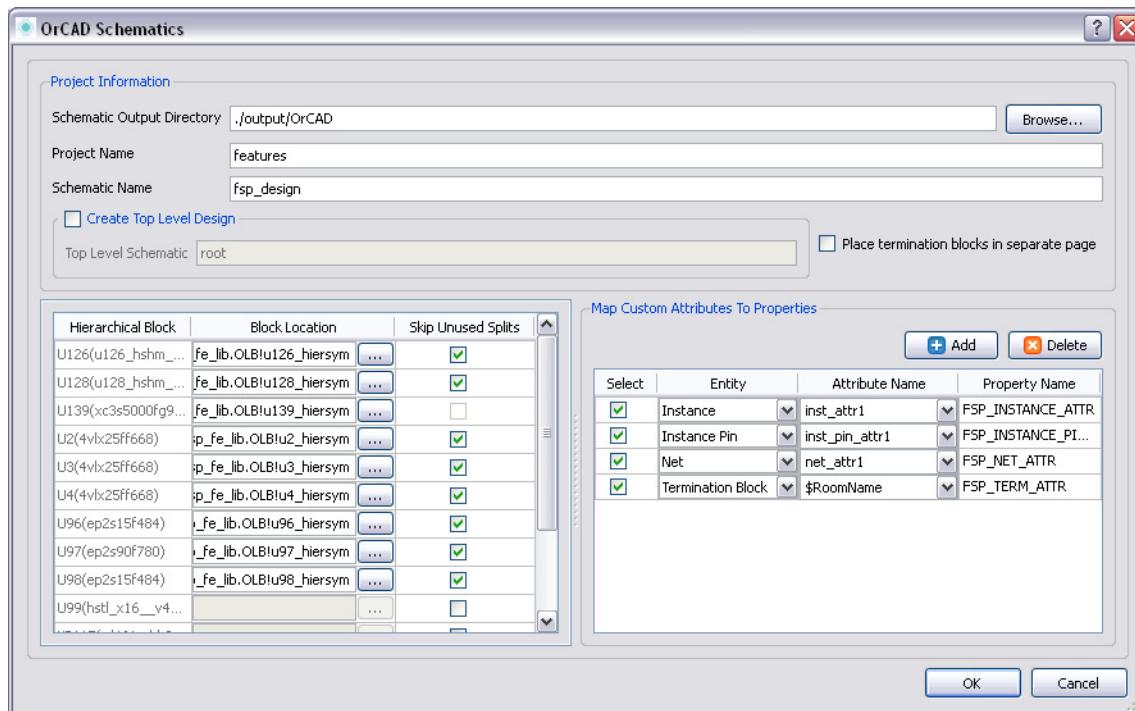
## Specify FPGA Symbol Location

This window appears when you click on ... button to specify the FPGA symbol location. Click on the Browse button and select the appropriate library folder where the FPGA symbol resides. After selecting the library directory name, the OLB file name displays in the Library

## Allegro FPGA System Planner User Guide

### Schematic in OrCAD Capture

File Name combo box. Click on the Package combo box and select the appropriate symbol to use and click Finish.



The schematics are generated at *project directory\output\OrCAD* locations. Once the schematic is generated, the Message Log section displays the successful creation message and path of schematic where they are created. You can click the link to directly open the location.

These generated files are imported to complete the rest of the board-design process.

**Allegro FPGA System Planner User Guide**  
Schematic in OrCAD Capture

---

---

## Exporting the Logical Design to a Board

---

This chapter covers the following topics:

- [Overview](#)
- [Creating a Board Outline for Logical Design](#)
- [Generating Board File and Placement Data](#)
- [Updating the Board with the Changes in Logical Design](#)
- [Updating the Logical Design with the Changes in the Board](#)
- [Importing an FSP/Non-FSP Initiated Board](#)



In this chapter we will refer to the design in FPGA System Planner as *logical design* and the physical design in Allegro PCB Editor as *board*.

## Overview

While designing a PCB, it is important that at every stage of the design process, the logical design is in sync with the board design. To ensure this, you need a mechanism to export and import changes made in the logical design as well as in the board design.

The FPGA based design is an iterative process that starts with a design creation in FSP and is followed by:

- Assigning FPGA pin locations
- Creating front-end symbols
- Capturing schematics
- Creating PCB footprint
- Routing the PCB

This process increases the number of iterations between FPGA System Planner and Allegro PCB Editor. Last minute changes in the logical design such as adding components, terminations, and pin swaps impacts the board design in PCB Editor. Similarly modifications to the board outline and component placement impacts the physical layout of the board also have an impact on the logical design. Irrespective of how the changes made in either of the tool, it is important that the logical design and board design are always synchronized.

FSP provides support for exporting modifications made in the logical design as well as for updating the logical design with modifications made to the physical layout of the board. Using FSP, you can either import all changes or can review the changes and selectively import the changes.



*Any changes* refer to any logical change in FSP design where as *specific change* refer to component placement, board outline and reference designator.

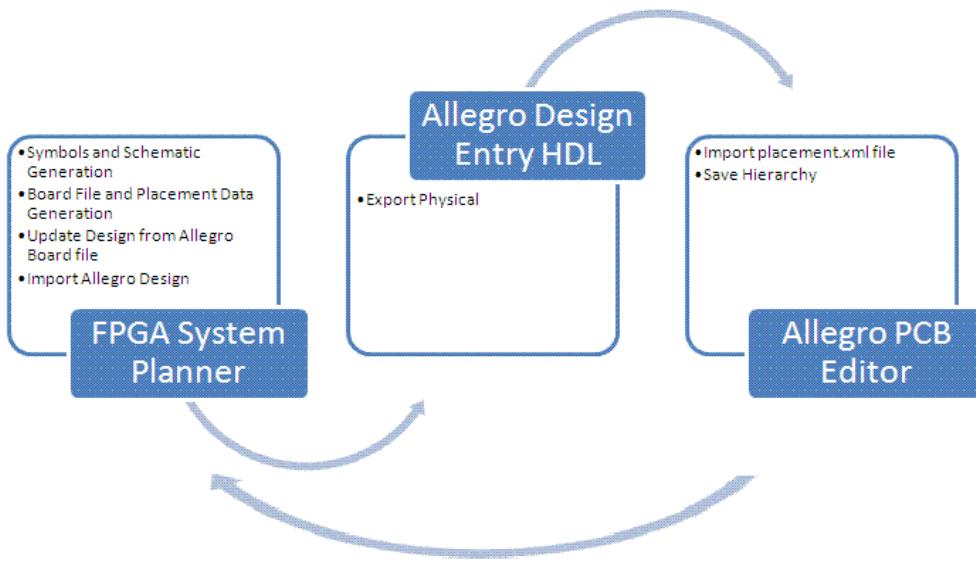
This chapter describes to the process of translating the logical design to physical design and then back annotating the physical design changes to logical design. This chapter also covers the relevant functionalities available in the Project Manager and Allegro PCB Editor.

## FPGA System Planner to Allegro PCB Editor Flow

The figure below provides an overview of design process using FPGA System Planner, Design Entry HDL and Allegro PCB Editor.

# Allegro FPGA System Planner User Guide

## Exporting the Logical Design to a Board



The top-level design tasks included in the flow are:

1. Prepare the logical design for layout in Allegro PCB Editor.

For detailed information, see the [Generating Board File and Placement Data](#) on page 476.

2. Export the logical design to layout in Allegro PCB Editor.

For detailed information, see the [Updating the Board with the Changes in Logical Design](#) on page 477.

3. Back annotate the changes in board to the logical design.

For detailed information, see the [Updating the Logical Design with the Changes in the Board](#) on page 480.

4. Import an FSP-initiated board or Allegro-initiated board in FSP.

For detailed information, see the [Importing an FSP/Non-FSP Initiated Board](#) on page 483.

**Note:** Back annotation and updating logical changes to board can be done multiple times to ensure that the logical design is in sync with the board in design.

## Creating a Board Outline for Logical Design

Before walking through the topics, updating the board outline from FSP generated template board and importing a board outline for non-FSP generated board into FSP you must know how to create a board outline for your logical design. Board outline is one of the entity involved in updating and import board file process. However it is not mandatory to create board outline always. You may create it based on your design requirements.

After creating a new project in FSP, you create a board outline for the new design. The board outline defines the area within which the components on your board are placed and routed. A board outline can be of any shape but FSP supports only square and rectangular shapes for creating board outline. However importing board outline of any shape using other boards is supported.

Specifying the details in *Define PCB Outline* section in *Settings* dialog box automatically creates a board outline on FSP canvas.

The board outline can be created at any time during the design capture stage or even after completing the logical design. But it is recommended that you create the board outline before placing the components on canvas. This helps you to ensure that your components are within the FSP board outline.

Using the *Settings* dialog box, you can remove or modify the existing board outline. To create a board outline for FSP design, perform the following steps:

1. Choose *File – Settings*.

Alternatively click *Settings* icon in tool bar or press the Ctrl + T keys.

The *Settings* dialog box is displayed.

2. Select the *Define PCB Outline* option.

3. In the *Bottom Left X* and *Bottom Left Y* text boxes, specify the X-Y coordinates of the bottom left corner of the board outline.

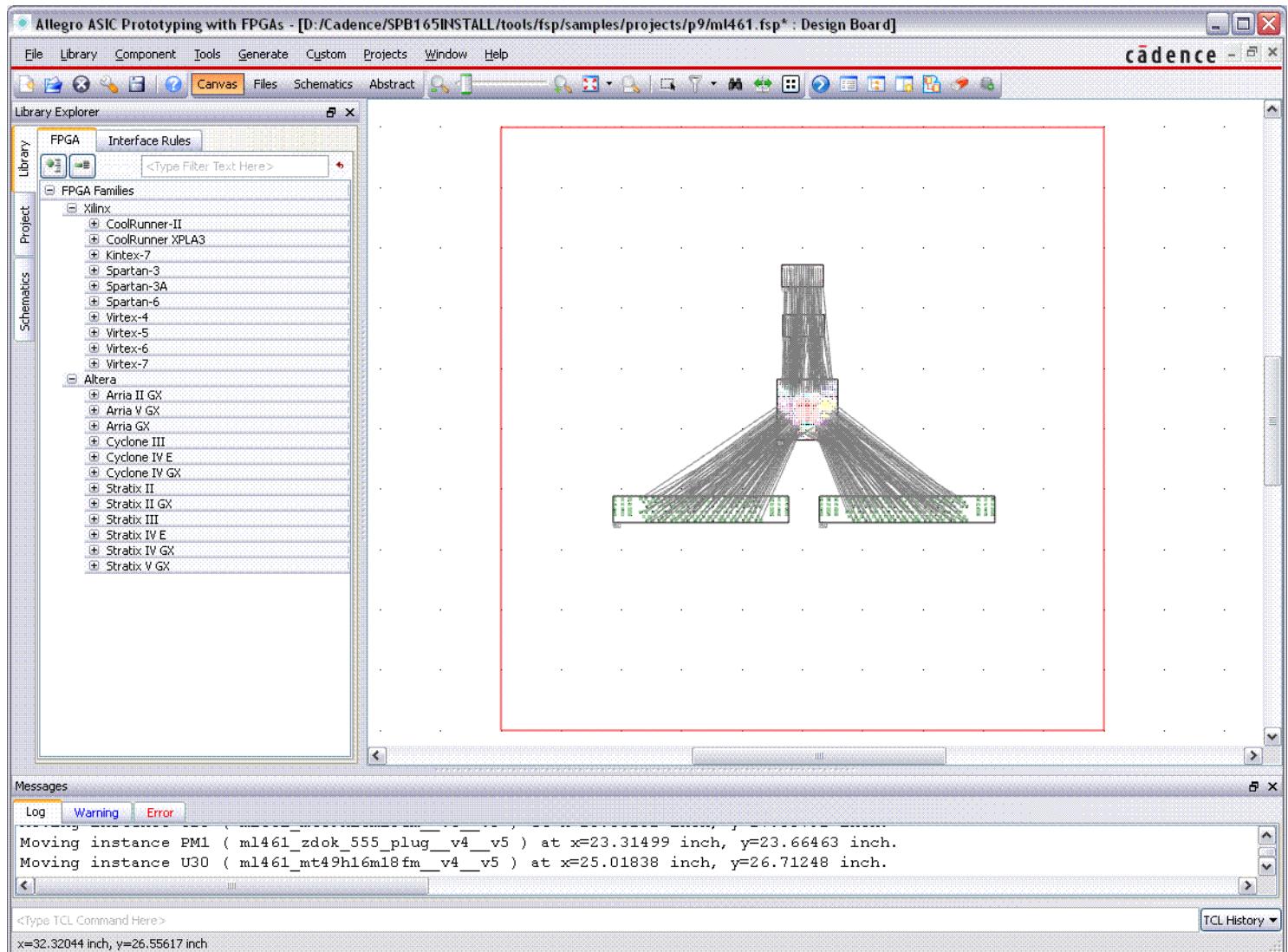
4. Use the *Width* and the *Height* text boxes to specify the length of the board outline on x-axis and y-axis respectively.

5. Click *OK*.

A new board outline is created on FSP canvas as below shown.

# Allegro FPGA System Planner User Guide

## Exporting the Logical Design to a Board



## Generating Board File and Placement Data

After completing symbols and schematics generation process the next step is to generate the board file for the logical design. The FSP generated board file is used as an input to Allegro PCB Editor for creating the physical layout of the logical design in PCB Editor. For more information on fields and button of *Generate/Update Layout Placement Data* dialog box.

To generate the board file perform the following steps:

- Enter *GenerateLayoutData* in the TCL command bar.

The status messages and the location of the output files is displayed in the Log window.

### The Output Files

The board file is created in a subdirectory known as *physical* at `\<project_name>\output\dehdl\<project_name_lib>\<project_name>\` path. The following files are also created in the same directory along with .brd and .xml file:

File Names..	Notes
t7c2layout.bat, update_placement.bat, make_board.scr, placement.scr	These files are generated for FSP internal operations and demo purpose.

### *The Placement.xml file*

**Note:** The placement.xml file is applicable in the OrCAD flow.

The placement.xml file contains the following information:

- Identification of FSP instances (Instance ID's) and their location on the canvas (X, Y)
- Mirror to identify if the instances is to be placed on top or bottom layer on the board
- Rotation angle details

The placement.xml contains the component placement details and is used to create the FSP floor plan exactly same in Allegro PCB Editor board also. After importing placement.xml file into Allegro PCB Editor, Allegro extracts the placement info from xml file and manipulate and place the components accordingly on Allegro canvas. This whole operation is done inside the Allegro and FSP just passes the placement information including the design units and extents from to Allegro. If required you can also modify the component placements on FSP canvas

and these changes can be preserved while updating the placement.xml file. Use *Placement Data Only* option in *Generate/Update Layout Placement Data* dialog box to update the placement changes next time.

Once the initial board is generated, the board designer can work to finalize the board design. You can also update the incremental changes in logical design to board file at anytime during the design. Generating the board file for second time will update the changes in board file. FSP prompts you with a confirmation window about overwriting the board file at the time of generating the board file second time. You can take the necessary actions as required and generate the board file.

In FPGA System Planner, you use *Generate/Update Layout Placement Data* window to generate the board file and placement data file. This dialog box lets you to generate or regenerate the board file and update the placement data.

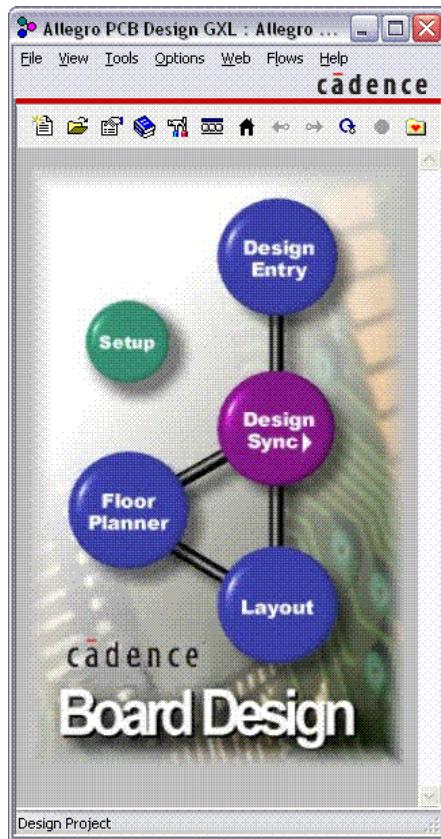
## **Updating the Board with the Changes in Logical Design**

The dehdl folder that is created on running the Allegro PCB Placement command, has multiple files besides the board file. One such file is the project file <project\_name>.cpm. This file is created in the <project\_directory>/output/dehdl directory.

Double-click this file to open the FSP generated Allegro project in the Allegro Project Manager window as shown in the following figure.

## Allegro FPGA System Planner User Guide

### Exporting the Logical Design to a Board



Click *Design Entry* in the Allegro Project Manager. This launches the Design Entry HDL tool displaying tool displaying the top level design of the FSP project. You can observe the FSP design in the form of a symbol with external nets represented as symbol pins. If required you can modify the top-level schematic in DE-HDL. To update the board design with these changes, run the *Export Physical*.

To run the Export Physical command from DE-HDL perform the following steps:

1. Choose *File – Export Physical*.

The Export Physical dialog box appears.

2. In the *Input Board File* field, enter the name and path to the existing board file that needs to be updated.

Ensure that the FSP generated Allegro Board file, <project\_name>.brd is specified as the input board file. By default, the name of the board file is displayed in the *Input Board File* field. If the physical path to the board file is not displayed, it indicates that the board file exists in the physical view of the root design.

## **Allegro FPGA System Planner User Guide**

### Exporting the Logical Design to a Board

---

3. To use another board file as the input board file, enter the name and path to the board file, or click *Browse* to select the file.

4. In the *Output Board File* field, enter the name and path to resulting updated board file.

By default, the name of the board file is displayed in the *Output Board File* field. If the physical path to the board file is not displayed, it indicates that the board file exists in the physical view of the root design.

5. Select the other options in various fields as per required.

6. Click *OK*.

The progress of the export process is displayed in the Session Log Window.

## Updating the Logical Design with the Changes in the Board

This section outlines how you can update the logical changes with layout changes made in PCB Editor.

**Note:** Before reading this section, it is important that you first understand the FSP – Allegro integration flow. For more information on the FSP – Allegro integration flow, see the FSP – DE-HDL flow guide.

Modifications to the board file, such as redrawing the board outline, changing components placement and renaming reference designators, need to be communicated back to the logical design. This process is known as back annotation.

In FSP, you can back annotate the changes through integration flow. The following points discuss the different cases of updating the logical design:

- If you are working with a single project database, you can directly save the layout changes using Ctrl + S or synchronize the changes in Allegro PCB Editor. Both logical and layout changes will be retained when you open the design in FSP.
- If you are working with a multiple project database, for example, if you take a back of your FSP design and associate the original FSP design in Allegro PCB Editor. Layout changes such as board outline, component placements, and reference designators can be back-annotated by importing board file using the *Updating Design From Allegro Board* dialog box. Logical changes such as creating renaming or removing nets, and swap pins can be back annotated using *Design Comparison* form.

**Note:** Layout changes such as component placements and reference designators can also be back annotated using *Design Comparison* form.

The board changes should be imported in the original logical design. Which means you must open the logical design before importing the board file.

The *Update Design from Allegro Board* process involves the following tasks:

- Open the original FSP design.
- Open *Updating Design From Allegro Board* dialog box by choosing *File – Update Design From Allegro Board*.
- Specify the board file name and path and select the entities.

## Points to Remember Before Updating the Logical Design

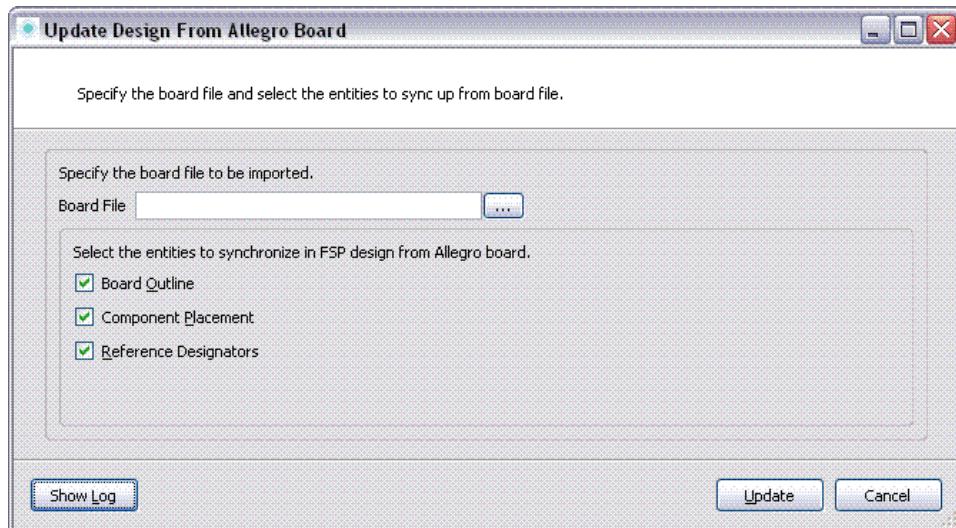
Before you update the logical design you must remember the following points:

- The *Updating Design From Allegro Board* option is visible only when the original design is opened in FSP canvas.
- FSP supports updating existing board outline from a non-FSP board outline as well.
- Any new component or connectivity changes made in Design Entry HDL/PCB Editor are ignored by FSP when you run *Updating Design From Allegro Board*.
- Importing board outline, component placement and reference designators changes does not impact other changes made to the logical design in FSP.

### To update the logical design with layout changes made in the PCB Editor

1. Choose *File – Update Design From Allegro Board*.

The Update Design From Allegro Board is displayed.



2. Specify the board file name and path to the board file in Board File field or click *Browse* to select the board file.
3. By default all check boxes are selected to ensure that changes to the board outline, component placement, and reference designators are imported in the FSP design. However you can modify the selection to import selective data.
4. Click *Update* to update the design.

## **Allegro FPGA System Planner User Guide**

### Exporting the Logical Design to a Board

---

The progress of updating design is displayed in Log window.

## Importing an FSP/Non-FSP Initiated Board

This section describes to import FSP initiated or non-FSP initiated board in FSP.



In this section *board* term refers to board initiated with FSP or non-FSP (third party tool).

Importing board process refers to extracting board outline, components from board, connectivity, and importing into a new FSP project. .

FSP supports reusing boards by importing components and board outlines from the board there by enhancing the FPGA board design process. The *Import Allegro wizard* that enables you to import a board in FSP. This wizard is available only when you create a new design or open an empty design.

The *Import Allegro Wizard* provides FPGA designers as easy method to quickly select the components from a board and import them on the FSP canvas. While importing components if required you can use the existing rules or symbol files to specifying the mapping information for the extracted component. In case you do not specify a rules or mapping file, the wizard creates a dummy rules file based on the component symbol and dra file.

Before invoking *Import Allegro Wizard* ensure the following:

- For a non-FSP initiated board, the dra files are extracted from the board and set in the PSMPATH.



Even if you do not set the PSMPATH, the *Import Allegro Wizard* does not fails. This step is in case after *Import Allegro Wizard* completes you save and reopen the design. FSP fails to open the design if the dra files are not found in PSMPATH.

- The LRFPATH and cpm/cds.lib files are set. This is useful for rules and symbol mapping.

When you invoke *Import Allegro Wizard*, it guides you through a series of steps which are required to import Allegro board into FSP. Before you invoke the wizard, you specify the board file that need to be imported in FSP. Choose *File – Import Instances from Board* to import the board file. The *Select a Board File* window appears. Use this window to browse and locate the board file. After specifying the board file, the *Import Allegro Board* wizard appears.

For detailed information about the columns, fields, and options of *Import Allegro Wizard*, click the help button at the left-bottom side of the page. When you click the help button, a pane appears at the right side of the page.

# **Allegro FPGA System Planner User Guide**

## Exporting the Logical Design to a Board

---

---

## **Generating Output**

---

This chapter lists the tasks that must be followed to generate the following:

- [Generating Constraint Files](#)
- [Generating Verilog Design Files](#)
- [Generating Plan Ahead scripts](#)
- [Generating CSV Files](#)
- [Viewing Outputs and Their Location in Explorer](#)

### **Generating Constraint Files**

FSP provides you the option to generate the constraints files for all the devices present on the Canvas. You can generate ucf and xdc files for Xilinx devices, and .tcl file for Altera devices. Besides generating ucf, xdc, tcl files, FSP also generates the verilog and vhdl files in the same directory where the constraint files are generated.

To generate the constraint files,

- Choose *Generate - Constraint Files*.

The constraints files are generated in the

`<project_directory>\output\constraint\instance_name` location. Once the files are generated successfully, the Message Log section displays the message of successful generation along with the location path of the output files. You can click the path to view the files.



The system may generate the following message while generating constraint files for Altera devices.

Message	Description
Do you want to check connectivity using Quartus II fitter for Device Instance Now?	This message appears when the Altera family device is connected to an interface and you are generating the constraint files. Click Yes if you wish to check the connectivity with Quartus II if it is installed. Otherwise, click No.

## Generating Verilog Design Files

FSP allows you to generate the verilog design files for all the devices present in the design.

To generate verilog design files,

- Choose *Generate – Verilog Design Files*.

The Verilog files are generated at project directory\output\Verilog\filename location. It generates two different files— i.e. .v and .vb. Once the files are generated successfully, the Message Log section displays the message of successful generation along with the location path of the design files. You can click the path to view the files.

## Generating Plan Ahead scripts

FSP lets you generate the PlanAhead scripts for all the devices that are present on the Canvas.

To generate scripts:

1. Choose *Generate – Plan Ahead scripts*.
2. Generates
  - a. batch\_unix.tcl (For Unix environment)
  - b. batch\_window.tcl (For Windows)

The user can run these scripts in two ways.

You can run these scripts by following ways:

# Allegro FPGA System Planner User Guide

## Generating Output

---

### **Batch mode**

Open a new DOS command window and type D:\Xilinx\10.1\PlanAhead\bin\planahead.bat -mode batch -source {path of the project} (With"/")

### **Plan Ahead Console**

1. In Plan Ahead, choose Window-->Run scripts.
2. After successfully generating the TCL commands a new folder will be created with the name of the device. This folder will have all of the output file i.e. .ppr , csv, net list related to the current project.

Below is the functionality of the scripts.

Serial No.	Command	Functionality
1	## Plan Ahead script for Device Instance U10 (Xilinx_Virtex5:5vfx70tff1136)	
2	hdi::project new -name U10 -dir {<path of the working dir>}\project3\output\planahead\U10}	Creates a new project.
3	hdi::project setArch -name U10 -arch virtex5	Selects the Device package ie V5.
4	hdi::floorplan new -name U10_FSP -part {xc5vfx70tff1136-1} -project U1	Create's a floor plan for the current device
5	hdi::port import -project U10 -floorplan U10_FSP -csv {<path of the working dir>}\project3\output\planahead\U10\U10_FSP.csv}	Imports the .csv file.
6	hdi::drc run -name {<path of the working dir>}\project3\output\planahead\U10\results_FSP} -project U10 -floorplan U10_FSP	Run the DRC check.
7	hdi::floorplan save -name U10_FSP -project U10	Save's the project.
8	hdi::project close -name U10	Close the project.

## Generating CSV Files

FSP allows you to generate the CSV files as output files.

To generate the CSV files,

1. Choose *Custom – Generate CSV Files*.

The CSV files are generated at *project directory\output\csv\<time>* location. Once the output is generated, the Message Log section displays the successful creation message and path of output files where they are created. The net bus width and pin net CSV files are created for every instance present in the design. You can click the link to directly open the location where the files are created.

The net bus CSV file contains the net name, pin type, and bus width whereas the pin net CSV file contains the pin number, pin name, and pin type.

## Viewing Outputs and Their Location in Explorer

FSP generates the Allegro DE HDL symbols and schematics, OrCAD symbols and schematics, and Constraint files. This section discusses the FSP output files, their extensions and paths in Explorer.

### Allegro DE HDL Symbols

Allegro DE HDL symbols are generated at the library part level for the specified symbol. FSP generates following directories:

---

chips	This directory has two files having extension .prt and .tag.
entity	If you have installed Allegro DE-HDL on your machine, FSP uses van and newgenasym tools to generate verilog, vhdl, and .sir files. These files are used for packaging the design.  If Allegro DE-HDL is not installed on your machine, FSP creates only the Verilog file in this folder. However, the design cannot be packaged unless a .sir file is available.
sym_n	The _n depends upon the number of symbols of the part. This directory has two files, with extensions .tag and .css.

## Allegro FPGA System Planner User Guide

### Generating Output

---

.backup	This folder is created when the Back symbol data checkbox is selected in the Regenerate Allegro Symbol form.
---------	--------------------------------------------------------------------------------------------------------------

If you have selected the Generate for FPGA , by default the symbols are generated at Project Directory\Local\Instance Name. However, you can change the location through the [Generating DE HDL Symbols](#) dialog box.

## Allegro DE HDL Schematics

The Allegro DE HDL schematics are generated in *project directory\output\allegro\schematic* directory. The file extensions are,

cds.lib	Cadence library file.
<project>.cpm	CPM file. Double click this file to open Allegro DE-HDL schematics. The first schematic page is the documentation page. It contains the instance/part name and the page number map.
hiersymlib	This folder stores the hierarchical schematic pages of devices used in the current project.

**Note:** On re-generation of the schematic, for root design, FSP will a create backup of the page files in the sch\_1 folder

## Allegro DE CIS Symbols

Allegro DE CIS symbols are generated at the library part level for the [Generating OrCAD Symbols](#) specified symbol. If you have selected the Generate Setup OrCAD Symbols for FPGA, by default the symbols are generated at Project Directory\Allegro\Library\Local. However, you can change the location through [Generating OrCAD Symbols](#) dialog box.

## Allegro DE CIS Schematics

The Allegro DE CIS schematics are generated in the *projectdirectory\output\orcad* directory.

## Constraint Files

The constraint files are generated in the *project directory\output\constraint\instance name* directory. All the constraint files are saved in different directories with name of their instances. The file extensions are

---

.ucf	This is a constraint file generated for Xilinx family FPGAs.
.tcl	This is a constraint file generated for Altera family FPGAs.
.v	This is verilog file describing the constraint information.
.vhd	This is a VHDL file describing the constraints information.

---

## Allegro PCB Placement

Allegro PCB Placement files are generated in *project directory\output\allegro\schematics\worklib\root\physical* directory. FSP generates the following files.

---

.brd	This is a Allegro board file.
placement.scr	This is a placement script file. It contains the placement information for all the instance in the FSP canvas.

---

---

# Synchronizing With FPGA Tools

---

## Overview

FPGA design and PCB design have become increasingly parallel. In FSP, FPGA designer generate the top level design and constraints. Then designer run it through synthesis, place and route and simulation in other FPGA tools. Once the logic design meets all the timing and performance criteria, it is passed to PCB designer for symbol generation schematic entry and layout.

Here FPGA design starts with the pin constraints generated by FSP. However the logic designer may want to change the FPGA pin constraints at later stage to meet the logic timing.

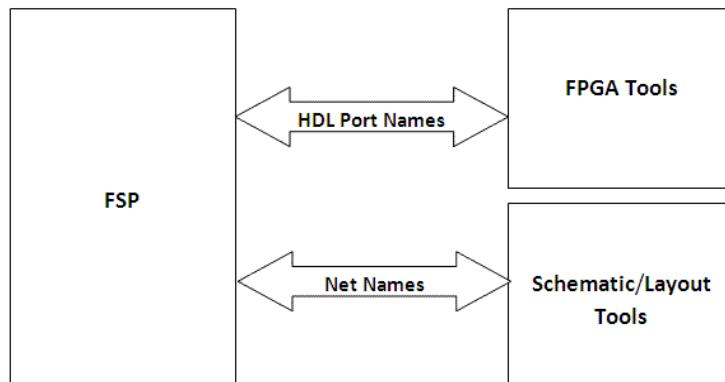
Synchronizing with FPGA tools is a flow that enables you to modify and update pin constraint and design files at any stage. An important advantage of the Synchronizing with FPGA tools flow is you can modify and update the pin constraint and design files until you meet your logic timings. You can do synchronization with various FSP features. The feature level details of synchronizing with FPGA tools flow are covered later in this chapter.

## Net Name Convention

Net names for all signals are different in FPGA design and PCB design flow. For example in FPGA tools span of the net is till FPGA and in schematic/layout tool span of the net is throughout the entire design. If you are using the FPGA tool, you can use the same logic multiple times for different FPGA's in a single board. In such cases, nets connecting two different FPGA's can have a common name. However, to avoid editing constraints manually and ensuring that nets that are logically correct have different names, it is necessary to generate net names in two levels. The two levels are:

- FPGA Port Names
- Net Names

The figure below displays different net name flows supported by FSP.



## FPGA Port Names

In FSP, FPGA Port names are applied to the FPGA ports level and are generated in constraint and verilog files. Further the generated constraint and verilog file are used to synthesis the design in other FPGA tools. Port names are useful while importing and exporting constraints and port mapping. When synchronizing with FPGA Tools following are the different features where you use FPGA Port names:

- Mapping FPGA Port names and Use pins.
- Import Constraints
- Export Constraints

Design flows you use FPGA port names while synchronizing with FPGA tools process are listed below:

- Design Flow starts with FSP
- Design Flow does not start with FSP

### Design Flow starts with FSP

- Initialize a design in FSP
- Generate full constraints or partial constraints. See [Exporting Constraints](#) section.
- Take the files to FPGA tools with RTL design. Run design and PAR for the design.
- Generate the constraints from FPGA tools.

- Import constraints in FSP and optimize it. See [Importing Constraints from an External File](#) and [Optimizing with Constrained Settings](#) section.
- Generate the constraints files.
- Map the ports with existing net names/port names and optimize it again. See [Mapping FPGA Port Names and Use Pins](#) section.

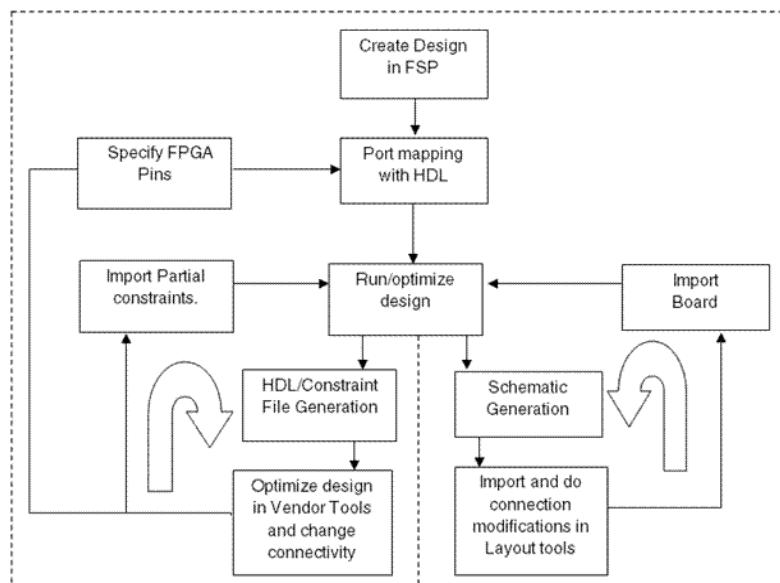
## **Design Flow does not start with FSP**

In some cases, you may not want to create a design with FSP library interfaces. In these cases, you can create a virtual interface as per your preference by using FPGA files. After creating virtual interface follow the above Design Flow starts with FSP topic steps from two.

## Net Names

Net names are applied to the design at the schematic level and are generated in schematics and design net list. These net names are then imported to Cadence Allegro FPGA (Concept), and Cadence Allegro CIS (OrCAD) tools.

The figure below describes how the FPGA Port names and Net names flow through the FSP, FPGA tools, and Schematic Tools.



## Defining FPGA Port Names for FSP Nets

FSP generates constraint, Verilog and VHDL files that you can use to optimize design. FPGA port names are used as ports for each component used in the design to generate the constraint files. For each component pins you can define the port names as ports. These FPGA port names are saved as ports in constraints files and Verilog entity declaration file.

You can define FPGA port names through following form:

- Interface Instance Configuration form
- Device Instance Configuration form
- Edit Protocol form
- Virtual Interface form

You can specify the FPGA port names by:

- Manually entering the names under the FPGA port name column
- Mapping port names in Port Map form
- Importing Constraints

## Defining FPGA Port Names for Interface Signals

To define port names for interface signals:

1. Right-click on the interface.  
A pop-up menu with various options is displayed.
2. Select *Configure Pins*.  
The Pin Property for Interface Instance dialog box is displayed.
3. Specify port names in the FPGA Port Name column.  
Net names will be considered as port names if you do not define the port names here.
4. Click *OK* after specifying the port names.

## Defining FPGA Port Names for Protocol Signals

To define port names for device signals:

**1.** Right-click on the FPGA.

A pop-up menu with various options is displayed.

**2.** Click *Create Protocol*.

The Create New Protocol dialog box is displayed.

**3.** Click **>** to move the device instance names to left side pane.

**4.** Click *OK*.

The Edit Protocol dialog box is displayed.

**5.** Specify port names in FPGA Port Name column.

Net names will be considered as port names if you do not define the port names in FPGA Port Name column.

**6.** Click *OK* after specifying the port names.

## Defining FPGA Port Names for Virtual Interface Signals

To define port names for virtual interface signals,

**1.** Right-click on the device instance.

A pop-up menu with various options is displayed.

**2.** Place the mouse cursor over Virtual Interface option.

A small pop-up menu is displayed.

**3.** Click *Create New Virtual Interface*.

The Define Virtual Interface for Device Instance dialog is displayed.

**Note:** The Define Virtual Interface for Device Instance form is same as like Edit Protocol dialog box.

**4.** Specify port names under FPGA Port Name column.

Net names will be considered as port names if you don't define the port names here.

**5.** Click *OK* after specifying the port names.

## Mapping Resources

Before mapping of FPGA port names and use pins and Importing Constraint process you must understand about the mapping resources feature. Resource mapping feature is useful while importing the IP generated sources and mapping the existing resources with the design resources. You can lock the constraints or relocate while re optimizing. Resources mapping is available at the time of ports and use pins mapping and while importing constraints.

There are two types of resources

- Dependent resources
- Independent resources

Dependent resources are available at interface levels and independent resources are available at FPGA levels.

This section covers the following:

- About Resource Mapping Window
- Mapping Resources

### About Resource Mapping Window

The Resource Mapping UI is divided into two panes:

- Resource Mapping
- Resources

### Resource Mapping

Resource mapping pane is divided into two columns:

- Group Name: Displays the name of the Device Instance name, Interface Instance name and interface groups in tree view. Click + in front of interface instance name to expand the hierarchy.
- Mapped Resources: By default the field is empty. You drop the resources from right pane to this column.

**Note:** You can add more resources in Mapped Resource column.

## Resources

Displays all the used resources in FPGA after you import constraint files in first page both in FPGA Ports and Use Pin mapping and Import Constraints wizard. Resource pane has the following options:

- Hide Identical Resources: Select this option to hide the identical resources.
- Hide Comments: Select this option to hide the comments

## Rules for Mapping Resources

Improper resource mapping is not checked by FSP in this wizard. FSP swaps the entire connections and resources of the complete group based on the following rules:

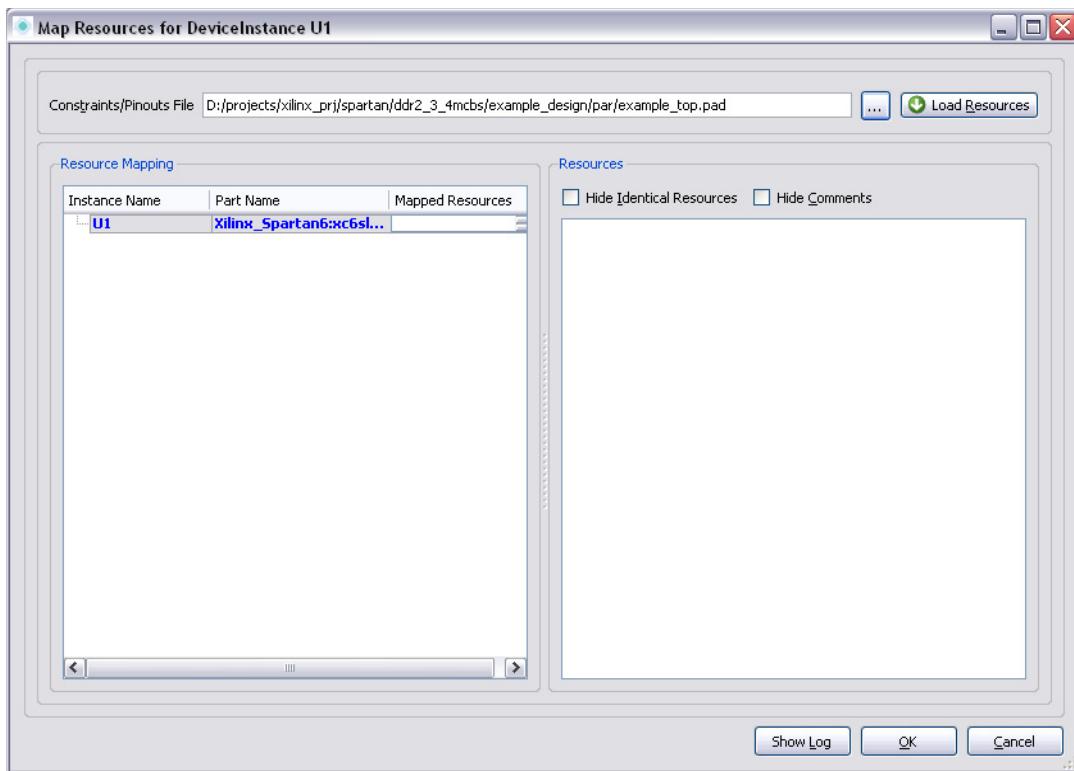
- Group Constraint should be identical
- Pin properties should be identical
- Number of pins should be identical
- Types of resources should be identical

## Steps to map the resources,

1. Right-click on the device instance and choose *Constraints – Map FPGA Resources*.  
The *Select Constraints File* dialog box is displayed.
2. Browse to the file, select the file, and click *Open*.  
The *Map Resources For DeviceInstance* dialog box is displayed.

## Allegro FPGA System Planner User Guide

### Synchronizing With FPGA Tools

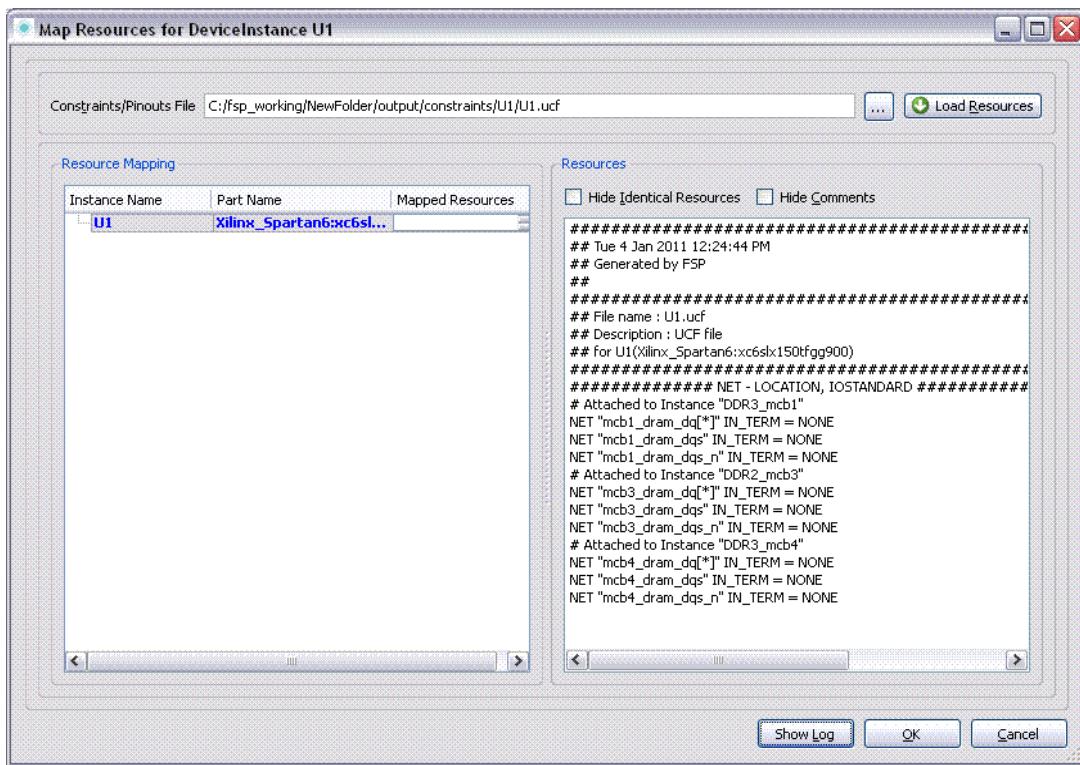


3. Specify the path to the constraints file in Constraint File field. Or click ... to browse to the constraints file location.
4. Click *Load Resources*.

The Resources is displayed in Resource pane.

# Allegro FPGA System Planner User Guide

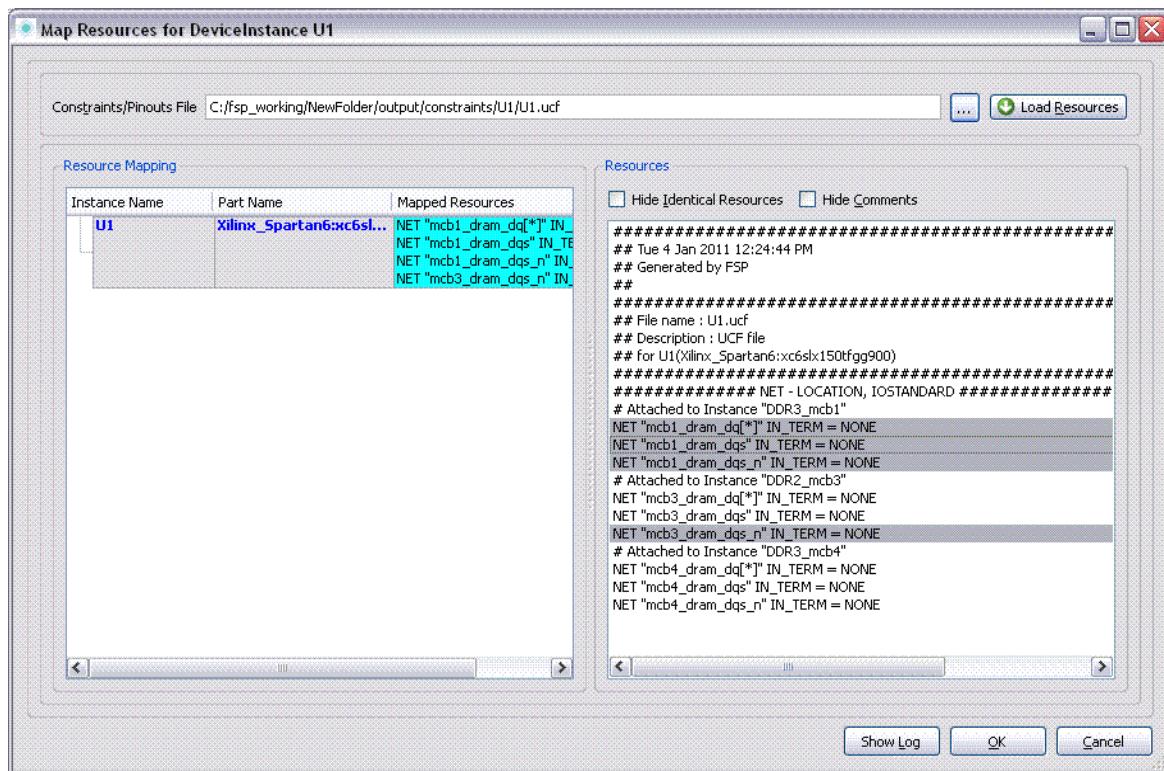
## Synchronizing With FPGA Tools



5. Select *Hide Identical Resources* option to hide the resources which are identical.
6. Select *Hide Comments* option to hide the comments in Resource pane.
7. Select resource in Resource pane with mouse pointer.
8. Drag the resource from Resource Mapping pane and drop it in Mapped Resource column of Resource pane.

# Allegro FPGA System Planner User Guide

## Synchronizing With FPGA Tools



9. Click *Show Log* to see the reports.

10. Click *OK*.

## Mapping FPGA Port Names and Use Pins

You can map FPGA port names from FPGA files for all signals connected to a device by using the Port Mapping window. This window also enables you to map the FPGA pin location constraints to the FPGA ports from constraint files. Port mapping window allows you graphically map FPGA port names and pin locations to the FPGA pins.

### Understanding different Mapping Port names and Use Pins Scenarios

Before you start importing the files you must understand following scenarios:

- Having FPGA file but no Constraint file

You can use this form to map port names to nets.

- Having Constraint file but no FPGA file

You can use this form to set use pin constraint for FSP nets.

- Having both FPGA and UCF files

You can use this form to map both FPGA ports and use pin locations.

### Step by Step Instruction for Port Mapping

Mapping of FPGA Ports and Use Pins are done on following basis:

- Existing Net Names
- Pin Names

## Mapping Port Names and Use Pins with Existing Net Names

The Existing Net Name scenario is used to demonstrate the steps in mapping the port names and use pins.

The FPGA Port Mapping Wizard wizard guides you through a series of steps that you need to perform for FPGA port names and use pin mapping.

Importing Constraints and FPGA files

# Allegro FPGA System Planner User Guide

## Synchronizing With FPGA Tools

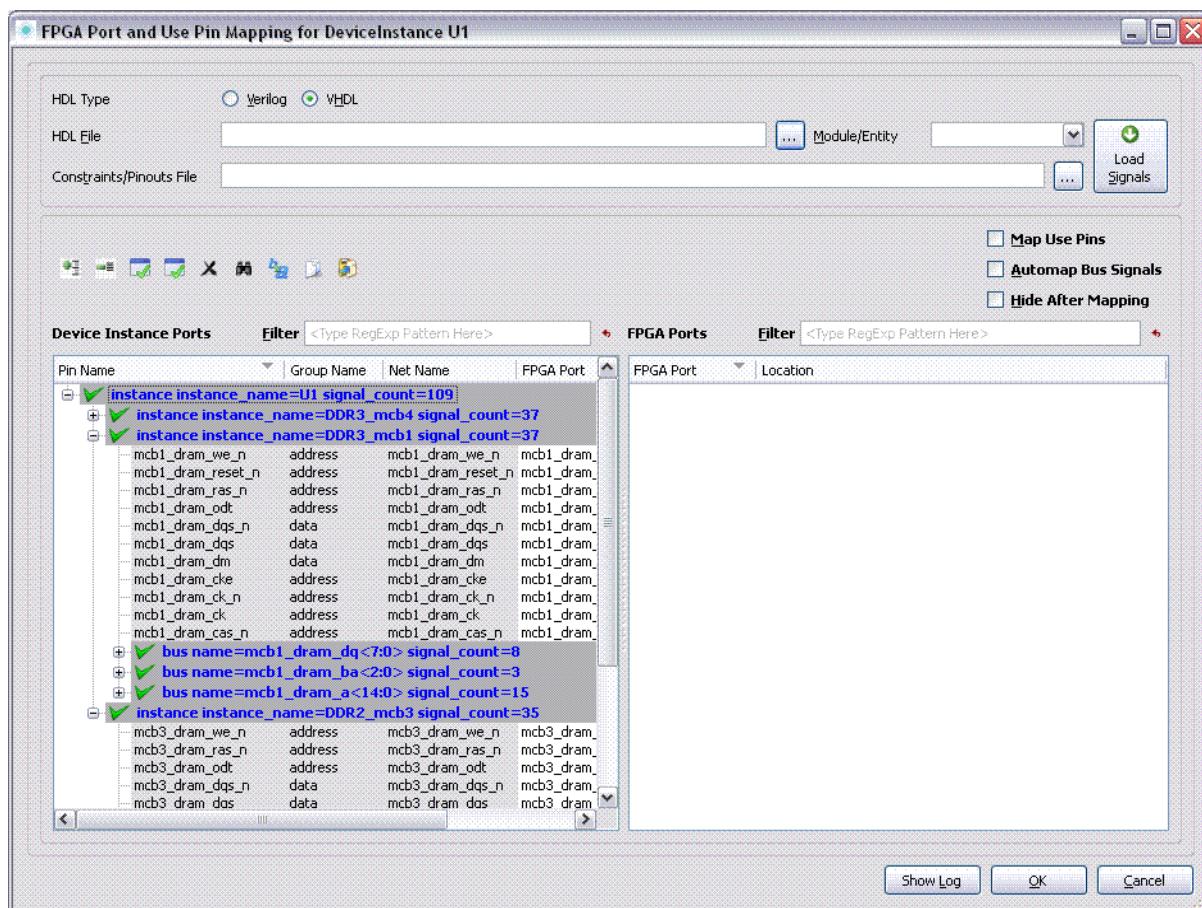
If connection exists in your design follow the below steps

1. Right-click device instance and choose *Constraints – Map FPGA Port and Pins*.

The *Select Constraints File* dialog box is displayed.

2. Browse to the file, select the file, and click *Open*.

The *FPGA Port and Use Pin Mapping* dialog box is displayed.



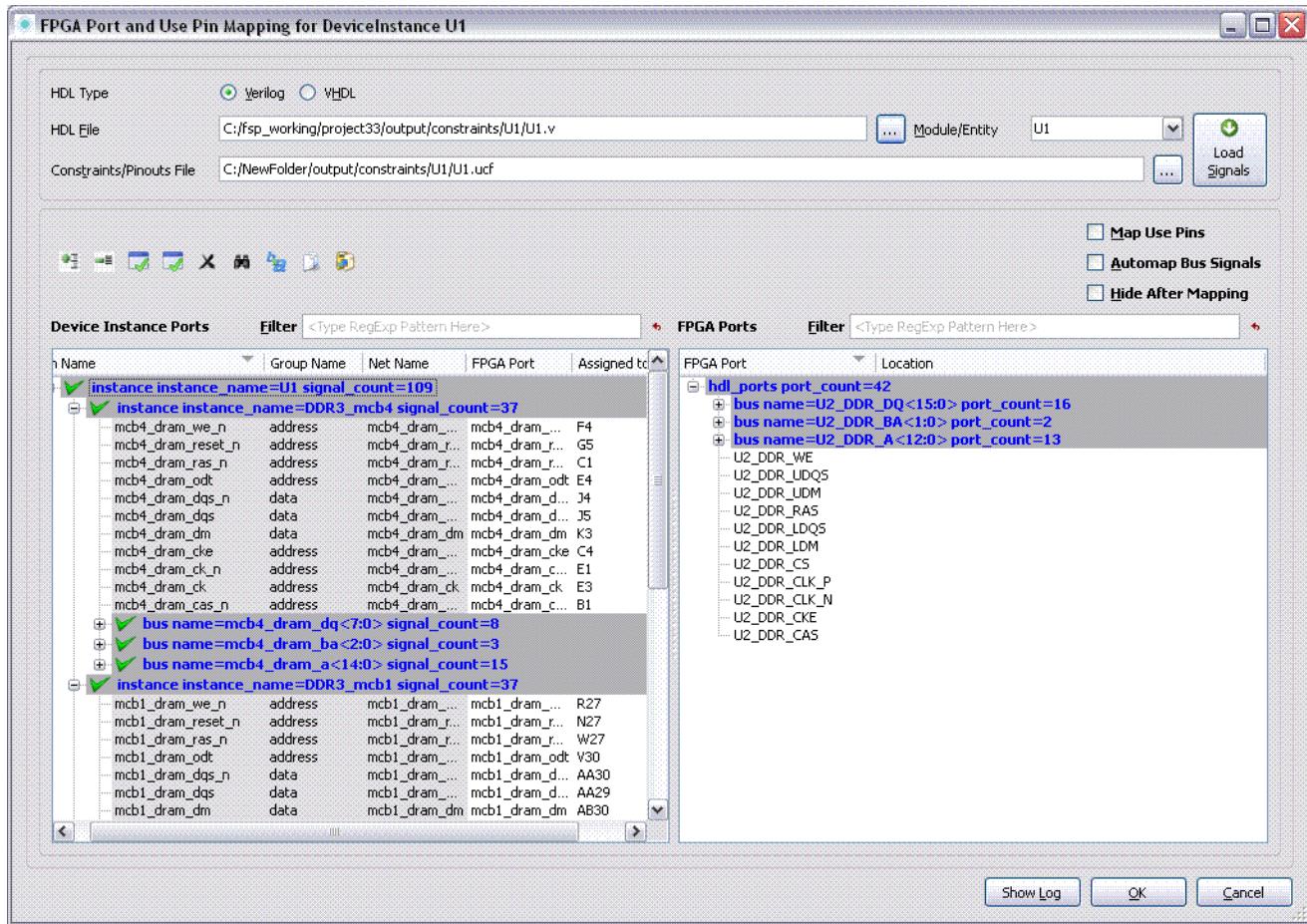
3. Select the type of file to import in HDL Type options field.
4. Select the appropriate Verilog or VHDL module in Module/Entity Field.
5. Specify the path to the verilog file in HDL File field. Or click ... to browse to the verilog file location.
6. Specify the path to the constraints file in Constraint File field. Or click ... to browse to the constraints file location.

## Allegro FPGA System Planner User Guide

### Synchronizing With FPGA Tools

7. Click *Load Signals* to display the FPGA Ports and Use Pin information in FPGA Ports pane.

The FPGA Ports and Use Pin information is displayed in FPGA Ports pane.



8. Select *Hide After Mapping* option to hide the FPGA Ports and Use Pin in FPGA Port pane after mapping.

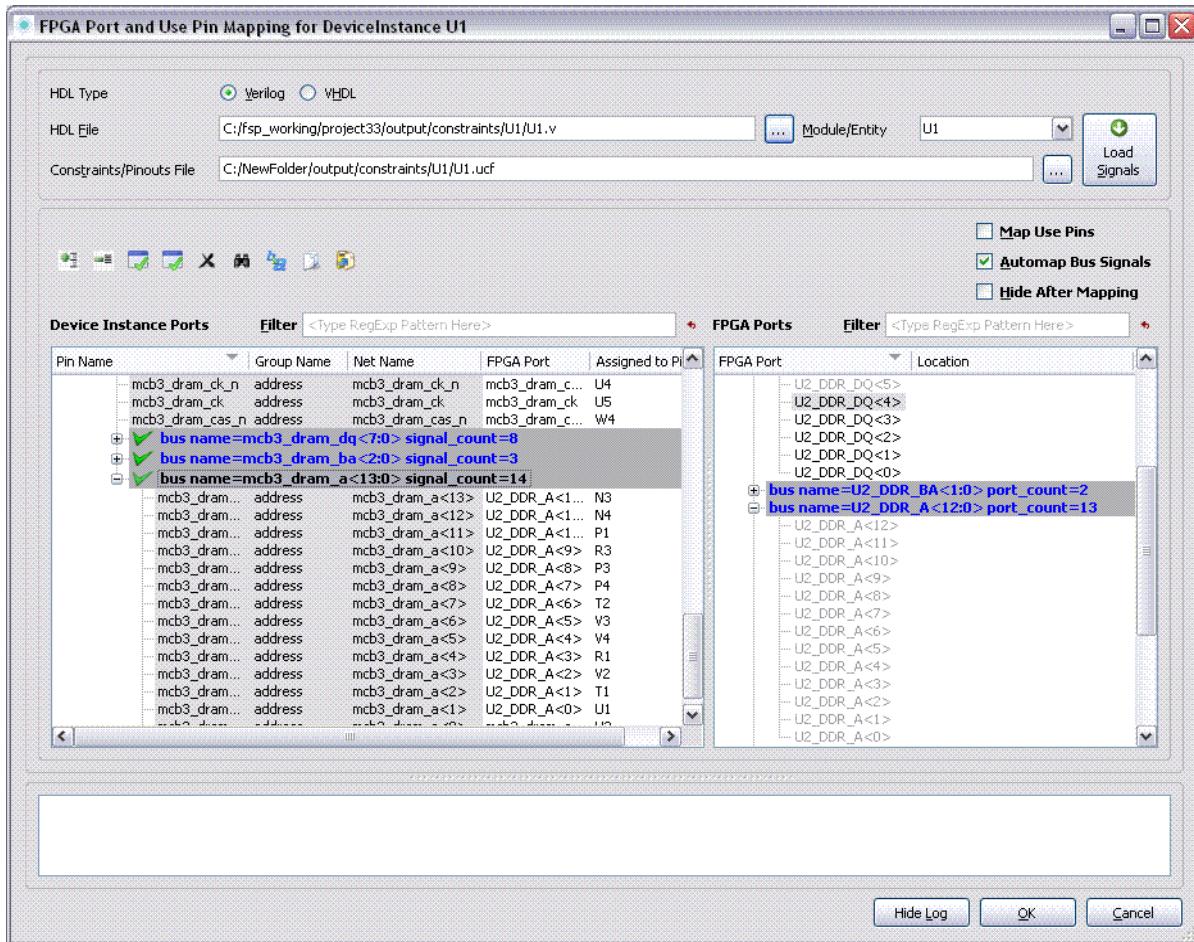
9. You can do the port mapping by using one of the following methods:

- Select one or two FPGA ports in FPGA Ports pane to map the ports on pin level basis . Drag and drop to the required net under FPGA Port column of Device Instance Ports pane.
- Select *Automap Bus Signals* option to map all the bits of a bus when mapping a single bit. Select a FPGA port in FPGA Ports pane. Drag and drop to the required net under FPGA Port column of Device Instance Ports pane.

## Allegro FPGA System Planner User Guide

### Synchronizing With FPGA Tools

- Select the bus header item in FPGA Ports pane. Drag and drop to the appropriate bus item in Device Instance Ports pane to map the complete bus.

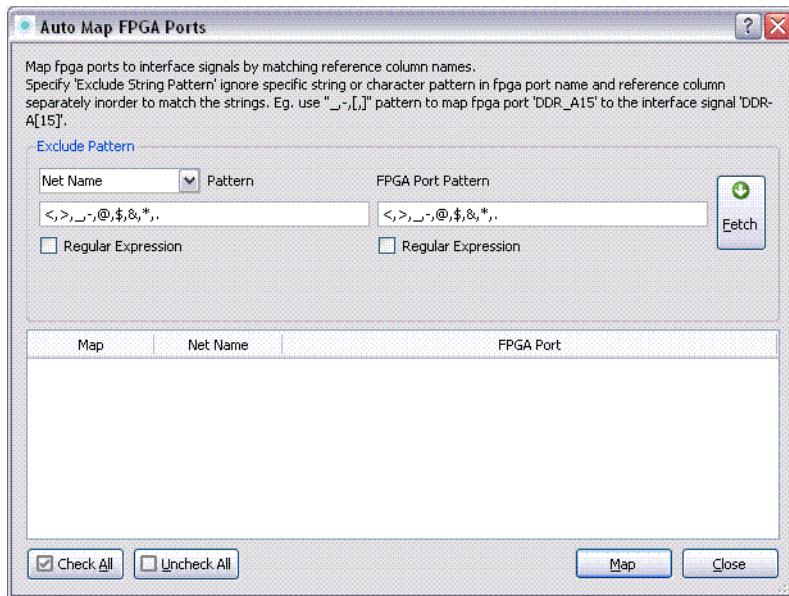


- Click Auto Map FPGA Ports icon to auto map all the ports with single click.

The Auto Map FPGA Ports dialog box is displayed.

## Allegro FPGA System Planner User Guide

### Synchronizing With FPGA Tools



11. Specify the necessary options and patterns to match the FPGA ports to interface signals.

**Note:** For detailed information on regular expressions and field and buttons of Auto Map FPGA Ports dialog box see [Auto Map FPGA Ports](#) section.

12. Click *Map* to automap the FPGA Ports.

13. You can do the use pin mapping by using one of the following methods:

- ❑ Select one or two Use Pin names in FPGA Ports column to map the use pins on pin level basis. Drag and drop to the required net under Use Pin column of Device Instance Ports pane.
- ❑ Select *Map Use Pins* option to map the Use Pin when mapping FPGA Ports.
- ❑ Click *Auto Map Use Pins* icon to auto map all the use pins at single step.

After clicking the icon, a confirmation dialog box is displayed, Do you want to auto map use pins for mapped port names? Click *Yes* to complete the operation.

14. Click *OK*.

**Note:** If you design has connections, you may be prompted to remove the connections. Click *Yes* to remove the connections.

**Note:** After clicking *Yes*, entire group is deleted if any of the used pin updated in the group.

## **Mapping Port Names and Use Pins with Pin Names**

If your design does not have connections follow the steps:

1. Follow the Mapping Port Names with Existing Net Names topic steps till step10.
2. Invoke Auto Map FPGA Ports Confirmation dialog box and select Pin Names option in Select Reference Column combo box and click *OK*.
3. Follow rest of the steps.

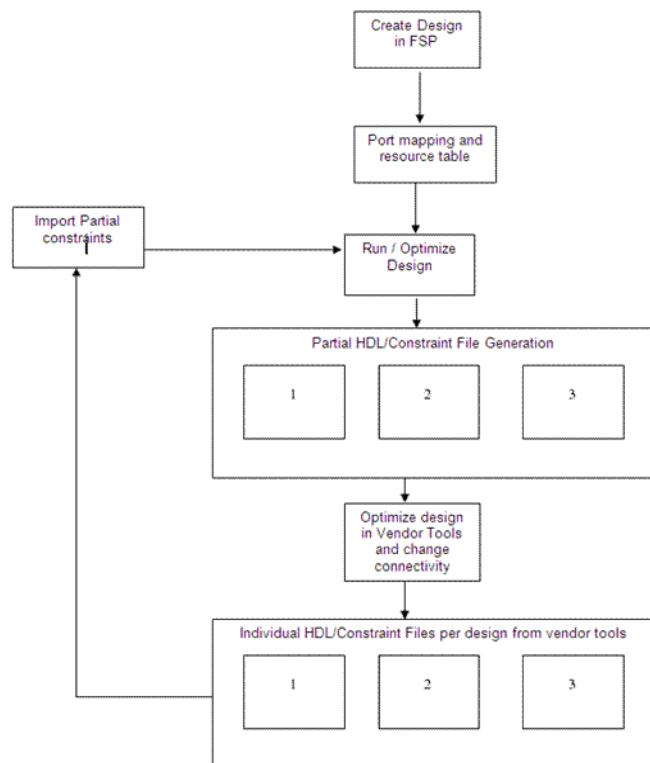
## Importing Constraints from an External File

FSP helps you to re optimize the design based on inputs (for example port names and locations) available in constraint files. These constraints and inputs are present in the constraint files. In FSP and other FPGA tool design flow, you can design and optimize each partition and later integrate with the top level design. FPGA designers may develop their FPGA logic and pin assignments periodically. For example from the whole design designer starts working on High Speed Interfaces first and validates those pin assignments before moving to other part of the design.

To import constraints partially from the whole FSP design, you import partial constraints files individually and merge the constraints with rest of the FSP design.

**Note:** For Altera FPGAs, you can import constraints from the <file name>.pin file.

Following figure shows the functional relationship between the FSP and other FPGA tools for logic design and optimization analysis.



The following list describes the above flowchart Import and export flow process from FSP to other FPGA tools:

1. Creating a design in FSP.
2. Do port mapping, if you have constraints/FPGA files.  
Or  
Run the design.
3. Generate partial constraint files.
4. Import constraint files in other FPGA tools, optimize it and change connectivity as per your need.
5. Generate constraint/FPGA file individually from FPGA tools.
6. Perform the port mapping again and re optimize the design.

**Note:** In Step 7, if FPGA ports are not mapped you should perform port mapping. Or if FPGA ports are mapped you can always do import constraints.

## About Constraint File

The Import Constraint feature allows you select the constraints individually. The constraint file that is read into FSP is a file that will typically have one or more of the following information:

- Port Name
- Pin Location
- IO standard
- Bank Number
- Resources

Import Constraints feature is available after or before running the design.

**Note:** For Altera FPGAs, you can import the constraints from an external file of format .pin; however, you cannot export the constraints to .pin file.

## Step by Step Instruction for Importing Constraints

This section guides you through a series of steps required that you need to perform while Importing Constraints.

## Allegro FPGA System Planner User Guide

### Synchronizing With FPGA Tools

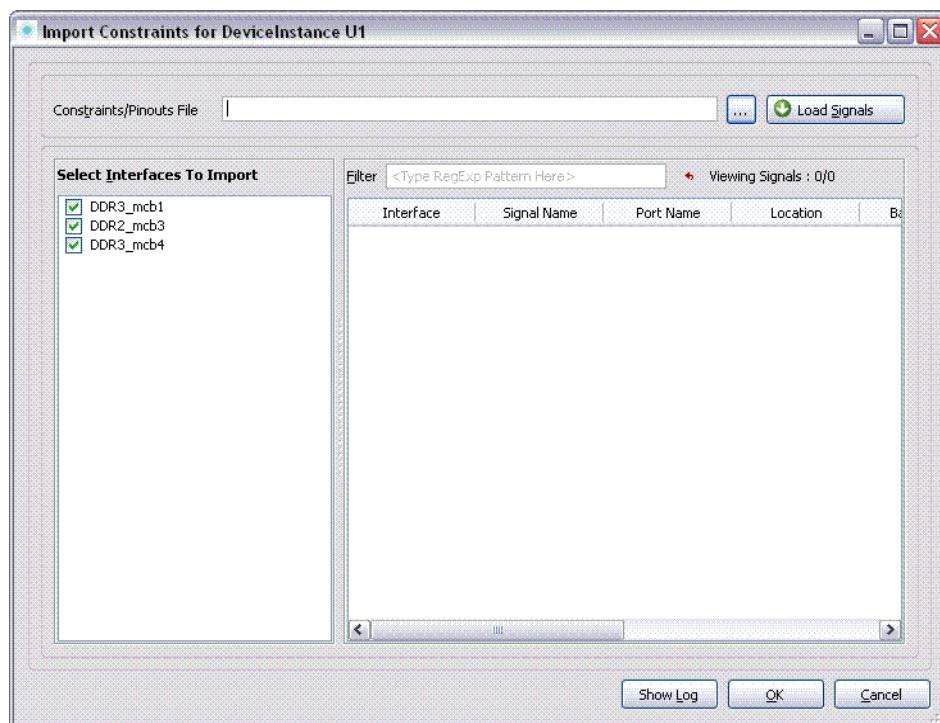
To Import Constraints follow the steps:

1. Right-click device instance and choose *Constraints – Import Constraints*.

The *Select Constraints File* dialog box is displayed.

2. Browse to the file, select the file, and click *Open*.

The Import Constraints dialog box is displayed.



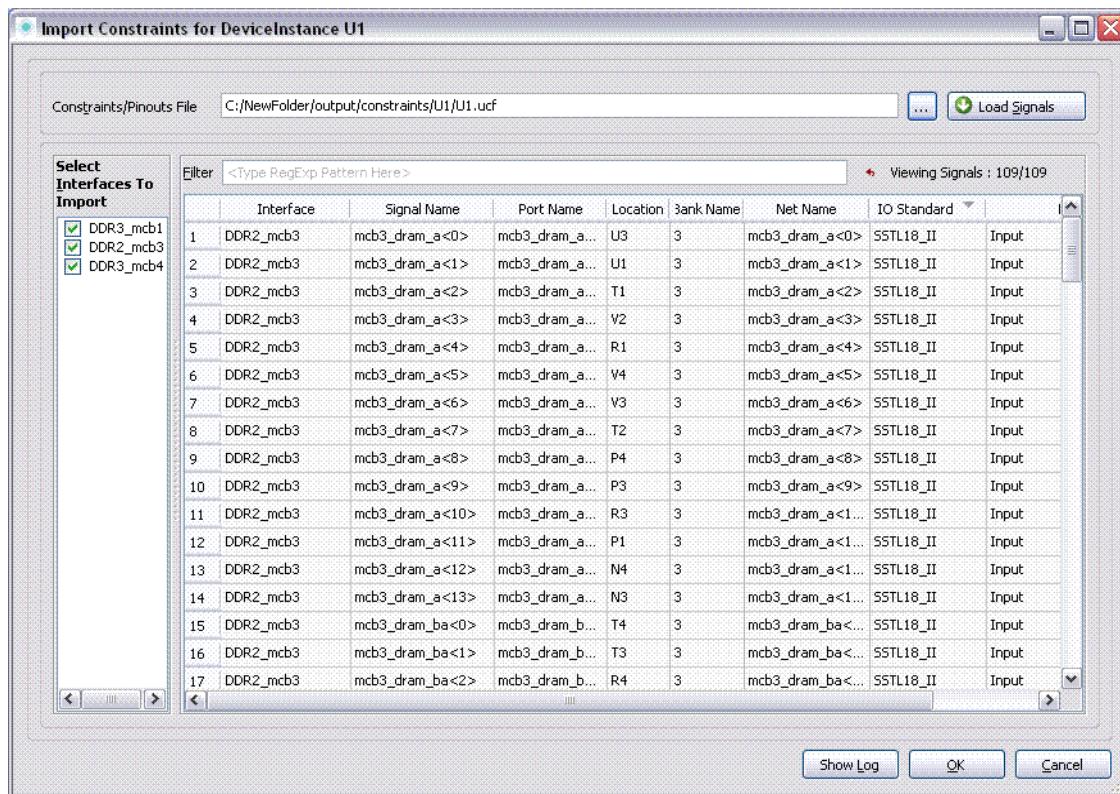
3. Specify the path to the constraints file in Constraint File field. Or click ... to browse to the constraints file location.

4. Click *Load Signals*.

The Constraint signals are displayed.

# Allegro FPGA System Planner User Guide

## Synchronizing With FPGA Tools



5. Select one interface and click *OK* to import the constraints of the selected interface.
6. Select all the interfaces and click *OK* to import the constraints for complete design.

## Importing Details

After clicking Import, if the connectivity already exists, FSP updates the connectivity instantly based by checking all the DRC and connectivity defined in the model definition. FSP prompts you with an error message if do not find any pin information for all the pins in the group.

## Points to Remember when Importing Constraint Files

The below figure describes the scenarios that you need to remember before or after importing constraints:

Scenario	Item
----------	------

## Allegro FPGA System Planner User Guide

### Synchronizing With FPGA Tools

---

Net connected in FSP and present in UCF/QSF file	Updates the connectivity as per constraint file. FSP doesn't change the schematic net name when you import the constraints. Use pin will be set for all imported constraint signals.
User constraint signals present in both design and UCF/QSF.	Updates the connectivity
Use pin and connections exist in design.	FSP check the standard DRC's rules.
Prohibit Constraints	You need to map in resource section.

---

### **Importing Incomplete Constraints Files**

Most of the times, other FPGA tools do not always generate complete constraints in files. For example:

- In Xilinx generated UCF files, IO standards are not generated for those pins which uses the default IO standard LVCMOS25.
- In Altera generated QSF files, pin locations are not generated for N side of the differential pairs, the assumption being made that the N side of the signal must connect to the N pin of N/P pairing of the FPGA.

When you import the constraints files, FSP reads the incomplete and defined constraints files and execute them properly by gathering all the missing and necessary information's.

## Optimizing with Constrained Settings

Optimizing design is a key part of the design. Reducing compilation, improving timing performance, reducing resource usage are some of the key part of optimizing design. FSP provides you method for design optimization.

With the use of this feature you import IP design and use the same pin outs or you optimize them with different constrained optimization options. After optimizing you can check the connectivity in other FPGA tools and re import it. You can follow this process until you get satisfied with your pin outs for design.

### Constrained Optimization

Constrained Optimization enables you optimizing pin connections in defined scope. This feature provides you with set of options to define the scope for optimization for set of pins. If the design does not fit into the device you want, you can try to optimize using the various options and settings. This feature is available only if the design has Use Pin information.

### Points to Remember when working with Use Pin

When working with Use Pin following are the scenarios:

- Connections do not exist.

FSP stores the use pin information in Pin Property of Interface Instance form. If your design does not have connections and you map the FPGA pins for component.

- Connections exist.

FSP prompts you with a confirmation window about removing connections in the design, if your design has connections.

### Start Optimizing Design

You can start optimizing design after following process:

- FPGA Ports and Use Pins Mapping
- Import Constraints

To optimize a design:

1. Choose *Design – Run Design*.

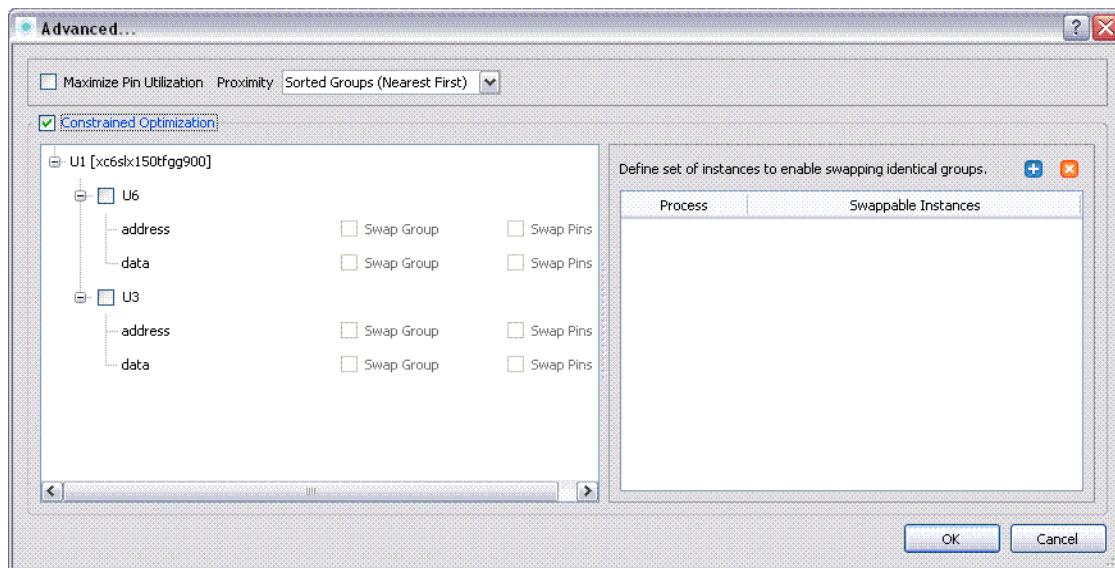
## Allegro FPGA System Planner User Guide

### Synchronizing With FPGA Tools

The Process Option Editor dialog box is displayed.

2. Click *Advance of Device Instance*.

The Constrained Optimization dialog box is displayed.



3. Click *Constrained Optimization*.

The Constrained Optimization pane gets enabled.

4. Select *U3* to swap the *U3* instance pins/groups within the instances.

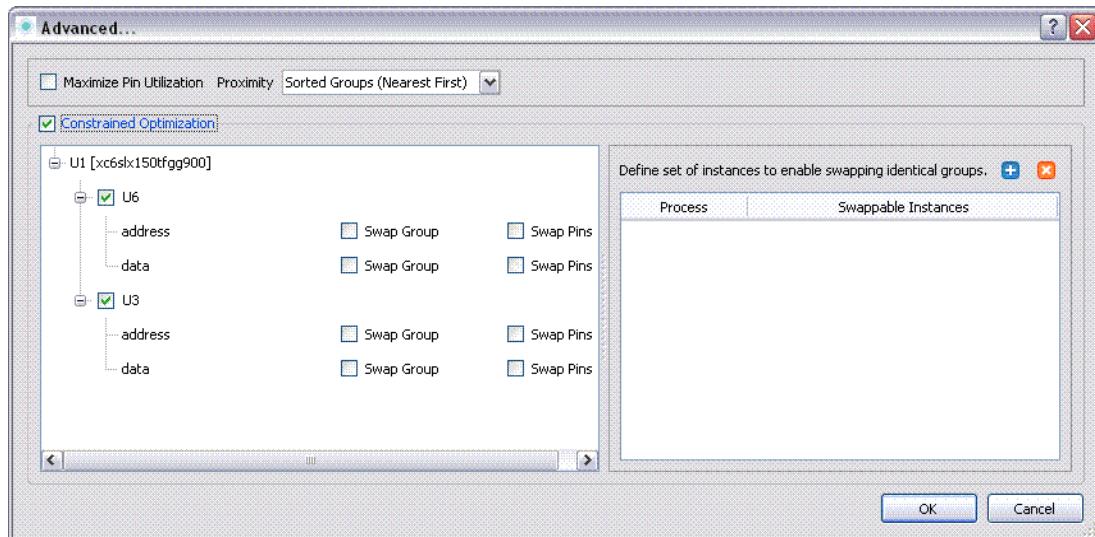
Or

5. Select *U3* and *U6* if you want to swap the groups/pins within the two instances.

The Swap Groups and Swap Pins options get enabled.

## Allegro FPGA System Planner User Guide

### Synchronizing With FPGA Tools



6. Click *Swap Groups* check boxes to select the groups that need to be swapped.

**Note:** You cannot swap the Data group with Address group. See below for more information.

After selecting the Swap Groups, Swap Pins gets automatically selected and is in disabled mode.

Or

7. Unselect the Swap Groups to swap the pins across single group.

8. Click + icon.

A new row gets added in Define the Swappable Instance pane.

9. Select *Process* check box.

10. Click drop down button.

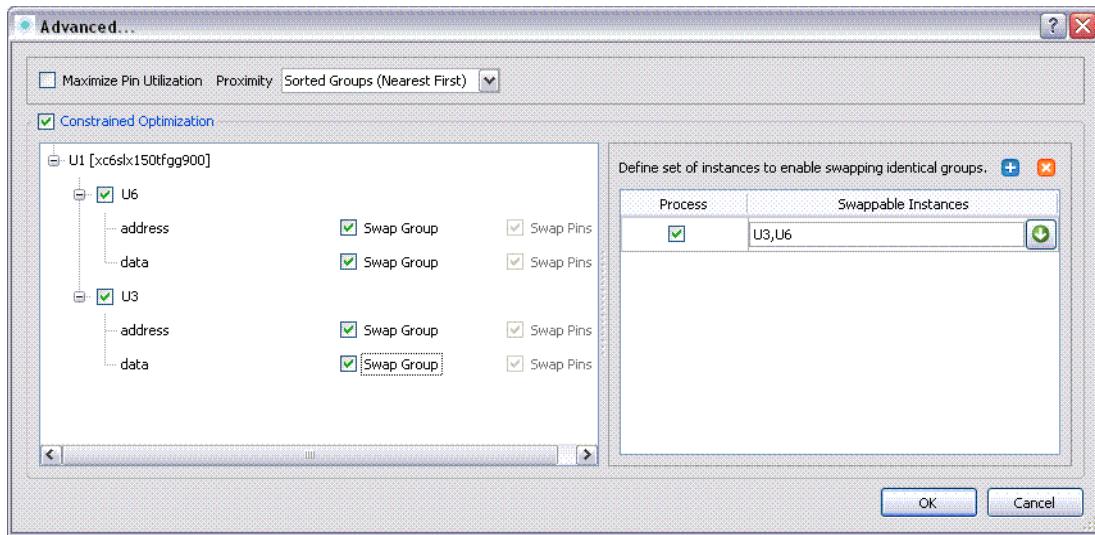
A pop-up menu with list of Instance names is displayed.

11. Select *U3* and *U6* to swap the pins between two interfaces.

Instance name gets added in the Swappable Instance text box.

# Allegro FPGA System Planner User Guide

## Synchronizing With FPGA Tools



Use X button to delete any row from the pane.

### 12. Click Run.

FSP checks following when optimizing at model level:

Action	Check
Swap pins in a group	If pin properties are same at model level
Swap all the pins of one group to other group	If number of pins in two groups and group properties are same
Swap all the pins of one interface to other interface	Both the interfaces should be identical logically.

## Exporting Constraints

Designer working in top level designs can export their optimized project as a design partition. FSP provides you the option to export the partial constraints on interface basis. After taking these partial constraints to other FPGA tools, finally FSP lead designer take all these constraints and integrate with his design.

FSP enables you to export the partial constraints to a specific location. These option is helpful when FSP designer and FPGA designer work closely on same design. It may happen sometimes FPGA design files will not be located in same directory as the FSP design files.

Export Constraints dialog box allows you to export the partial or full constraints file at specified location.

Before you start exporting the constraints you must understand following things:

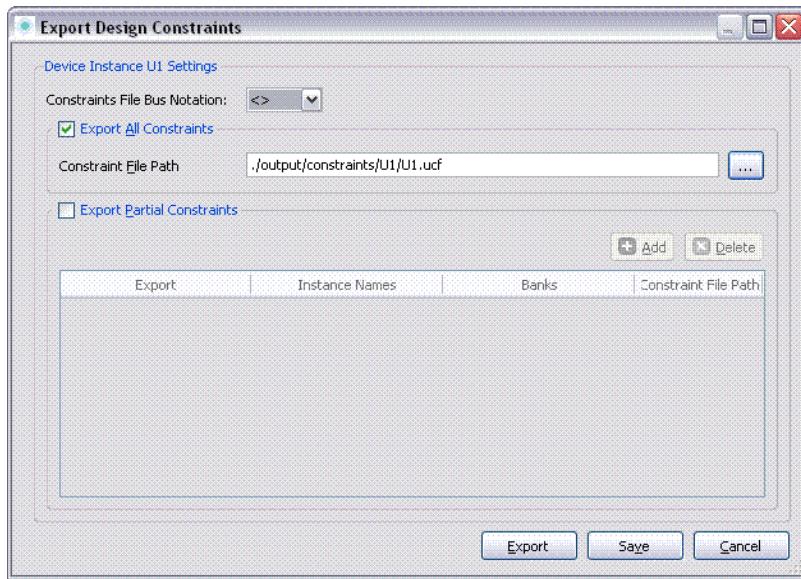
- Use Export All Constraints- To generate the resources for complete design.
- Use Export Partial Constraints- To generate the resources at instance level and specify the instance generate constraints.

### Exporting All Constraints

To export the constraints follow the below steps:

1. Right-click device instance and click choose *Constraints – Export Constraints*.

The Export Design Constraints dialog box appears.



2. Select the Bus Notation to output the signals with selected notation.
3. Click *Export All Constraints* if you want to export all the constraints.
4. Specify the directory in which to save the constraints file in *Constraint File Path* field. Or click ... to browse to the location of the constraint file.

**Note:** By default the project output folder directory is displayed.

5. Click *Export*.

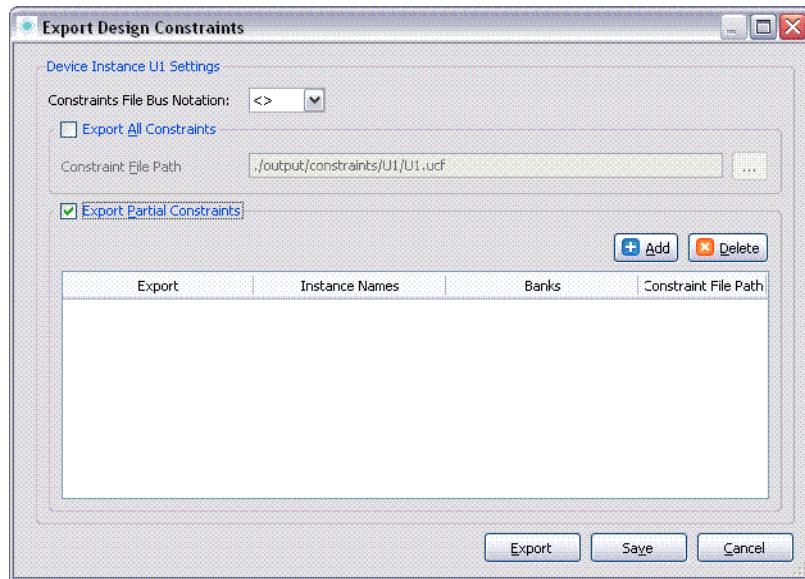
## Exporting Partial Constraints

To export the constraints follow the below steps:

1. Right-click device instance.  
A pop-up menu with various options is displayed.
2. Click *Export Constraints*.  
The Export Design Constraints dialog box is displayed.

## Allegro FPGA System Planner User Guide

### Synchronizing With FPGA Tools



3. Select the Bus Notation to output the signals with selected notation.
4. Click *Export Partial Constraints* option to export constraints partially.
5. Click *Add*.

A new row gets added to the Export Partial Constraints grid.

6. Select *Export* option.

7. Click .

A pop-up menu with list of interfaces names is displayed.

8. Select one or more interfaces.

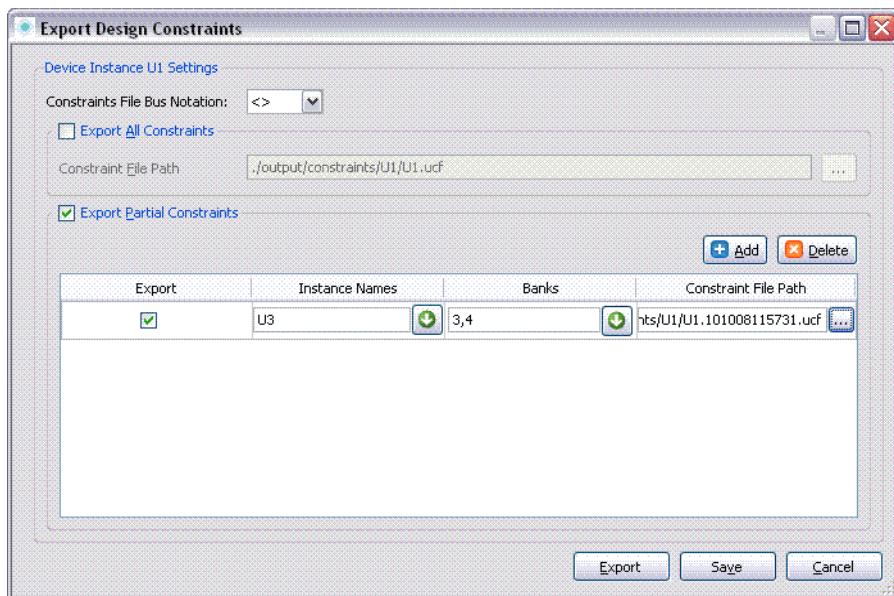
**Note:** With this option you can generate single or multiple constraints files with multiple interfaces, protocols.

9. Click and select one or more banks.

The device view of the FPGA and list of banks is displayed.

10. Select one or more bank numbers and click *OK*.

11. Specify the directory in which to save the constraints file. Or click and select the location under Constraint File Path.



**12. Click *Export*.**

After clicking Export the constraints file with selected settings is generated in specified path. It generates both verilog,VFPGA and ucf files with same name.

**13. Click *Save* to save the settings.**

## Swapping Groups

This topic describes the following:

- [Overview of Swapping Groups](#)
- [Swapping Groups Within an Interface](#)
- [Swapping Groups Between Two Interfaces](#)

### Overview of Swapping Groups

You can use the FSP to swap the groups. You can perform the individual swap operations on the groups automatically to uncross the rats nest. The FSP enables you to optimize the swapping group process to achieve better rats nests. By swapping groups you can minimize the average rats nest crossings. The Swap group feature in the Design Connectivity, gains more significance for identical interfaces where all the groups are logically same. When using IP generated pin outs, make sure that the constraint optimization options are set appropriately while swapping groups.

## Points to Remember Before Swap Groups

The following points you must remember before you start swapping groups:

- You are not allowed to manually swap interface groups when constrained optimization is not set.
- Irrespective of Constrained Optimization settings (On/Off) you are allowed to swap same logical groups (having Use Pin setting).

## Swapping Groups Within an Interface

The following illustrates the steps to swap groups using an example.

1. Invoke the Design Connectivity.

The Design Connectivity is displayed.

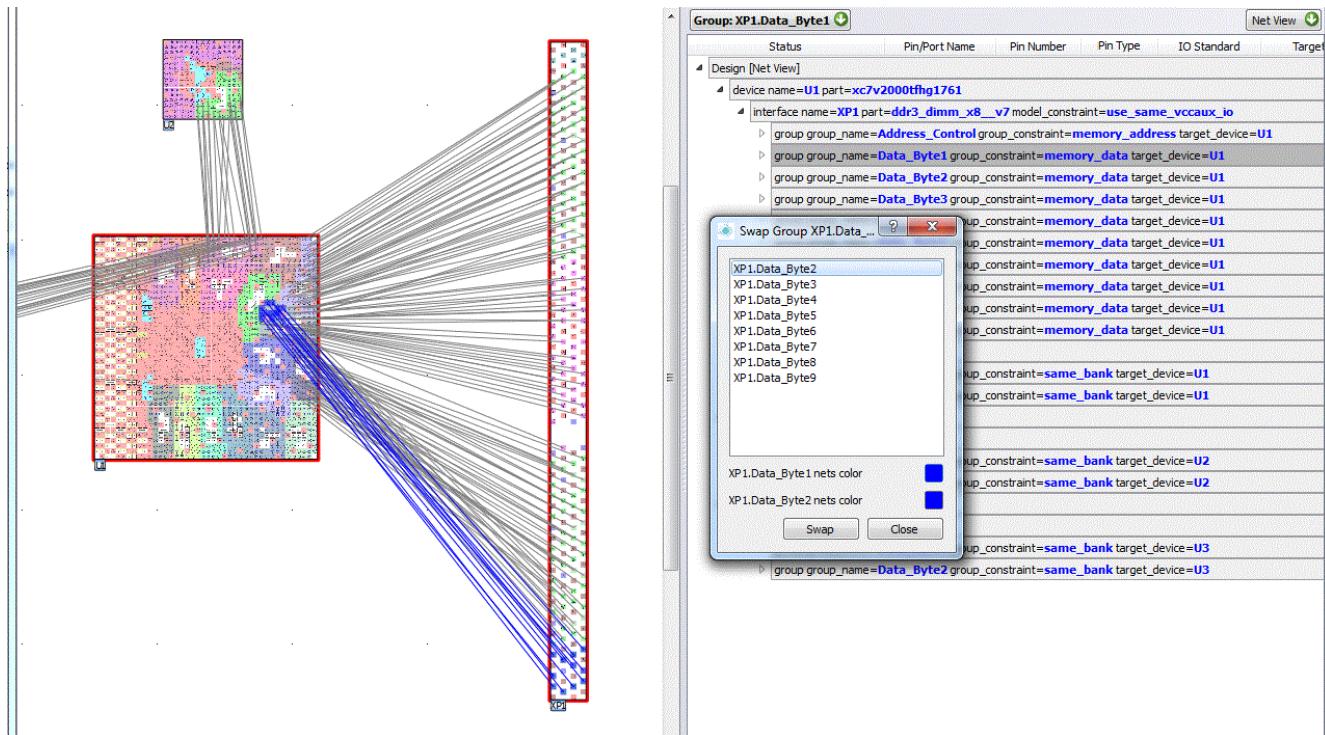


## Allegro FPGA System Planner User Guide

### Synchronizing With FPGA Tools

#### 2. Right-click on the XP1.Data\_Byte1 and choose *Swap Groups*.

The Swap Group <Inst\_name><Group\_name> dialog box is displayed. This dialog box lists all the names of the groups that are available in the selected interface.

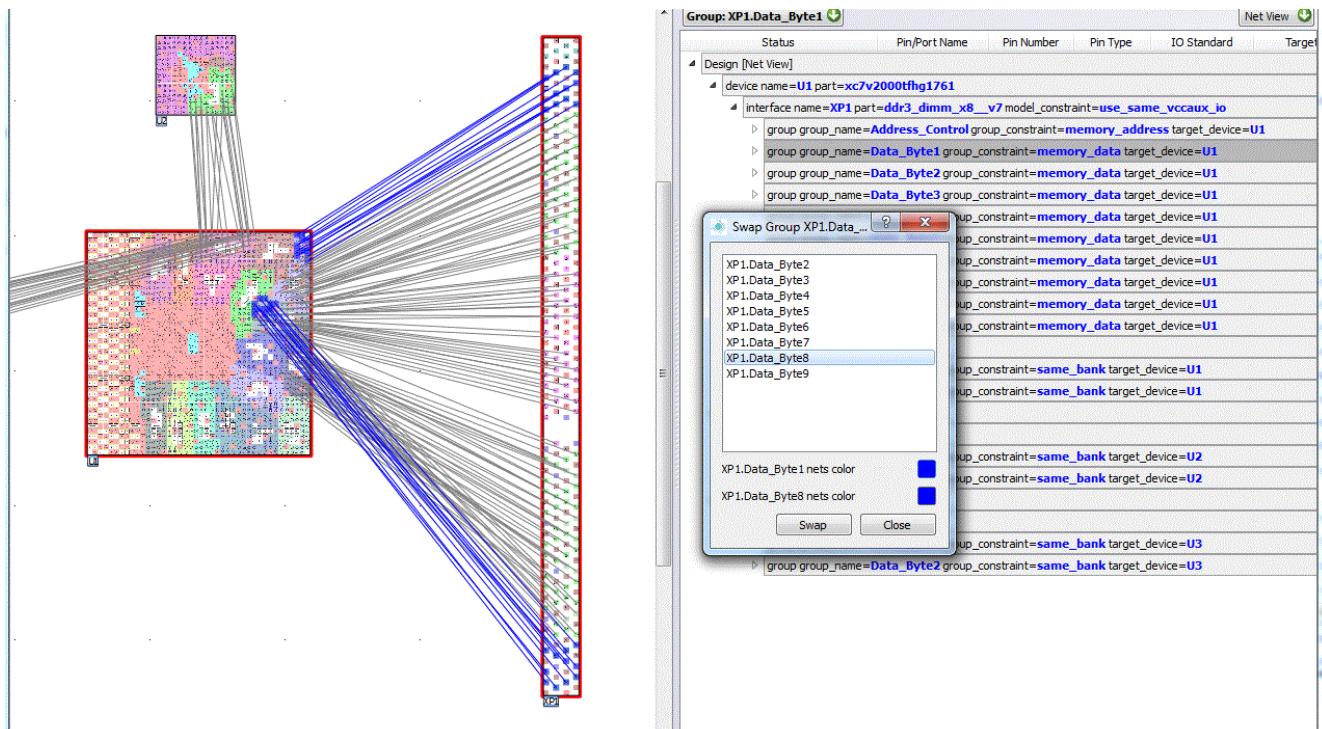


**Note:** The selected group name, XP1.Data\_Byte1 will not be listed in this list.

#### 3. Click on the XP1.Data\_Byte8 in the list.

# Allegro FPGA System Planner User Guide

## Synchronizing With FPGA Tools



FSP displays the names of the selected groups, XP1.Data\_Byte1 and XP1.Data\_Byte1, at the bottom of the window as a confirmation.

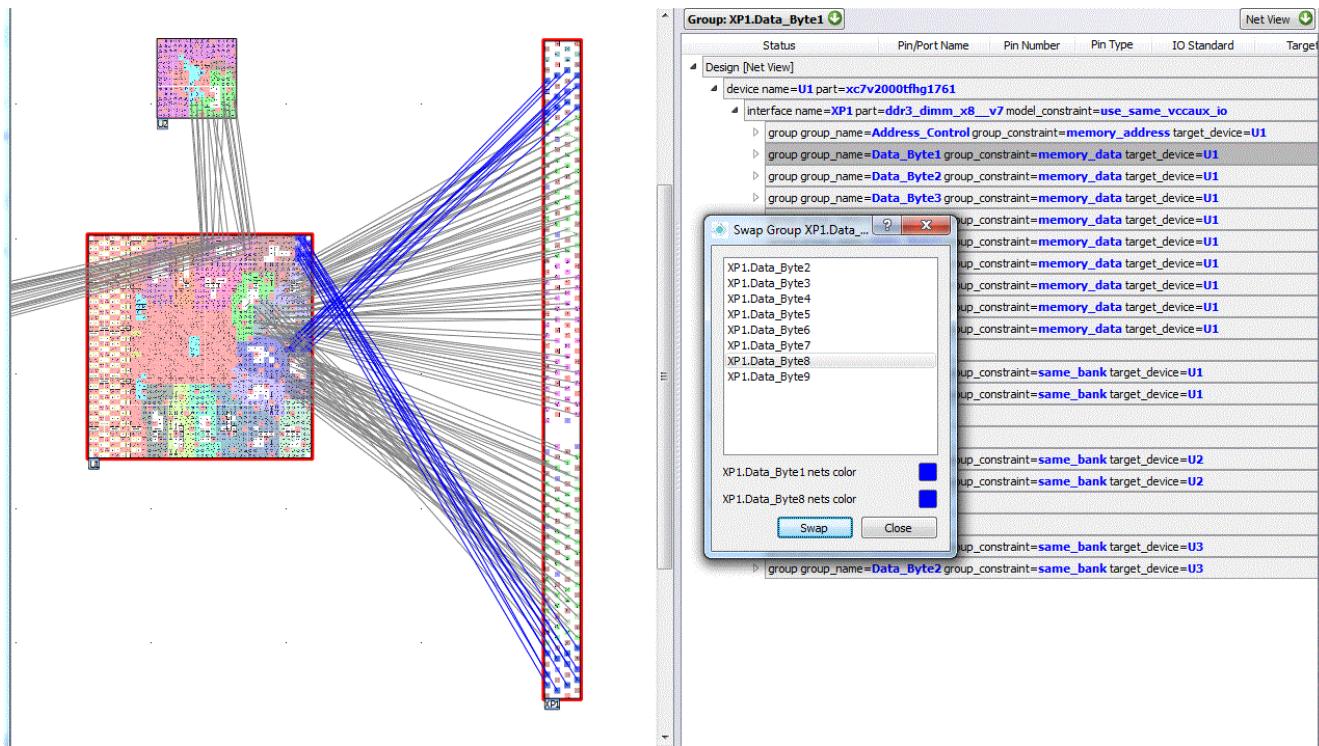
4. Click *Swap* to swap groups.

**Note:** After you click *Swap*, the *Swap Group* dialog box is continued to display. To close the dialog box click *Close*.

The two groups XP1.Data\_Byte1 and XP1.Data\_Byte8 are swapped.

# Allegro FPGA System Planner User Guide

## Synchronizing With FPGA Tools



## Swapping Groups Between Two Interfaces

The following illustrates the steps to swap groups using an example.

### *Important*

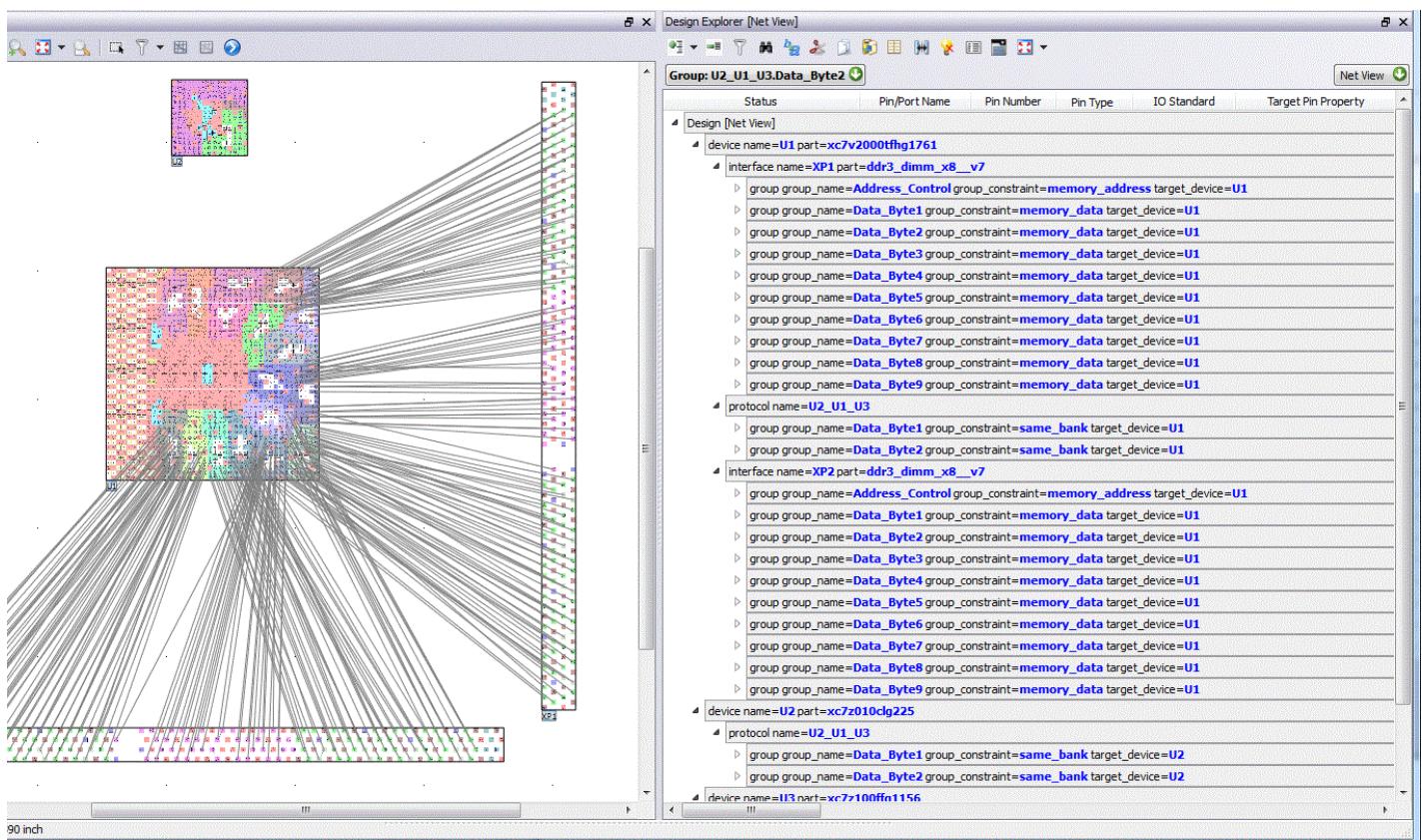
You can swap the groups of two interfaces only when the two groups does not have any model constraints.

#### 1. Invoke the *Design Connectivity*.

The Design Connectivity is displayed.

# Allegro FPGA System Planner User Guide

## Synchronizing With FPGA Tools

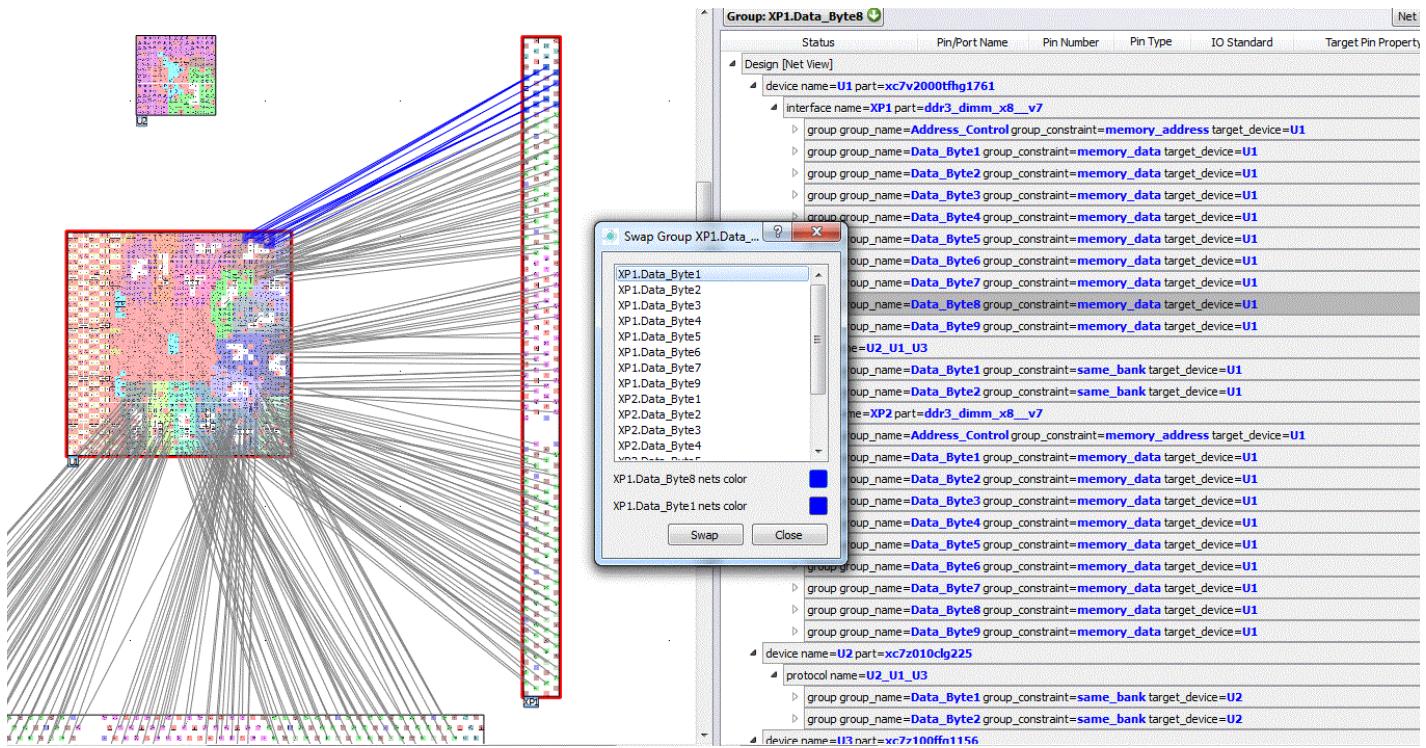


## 2. Right-click on the Data[Byte]8 and choose *Swap Groups*.

The Swap Groups dialog box is displayed. The dialog box lists all the names of the groups of the two interfaces.

# Allegro FPGA System Planner User Guide

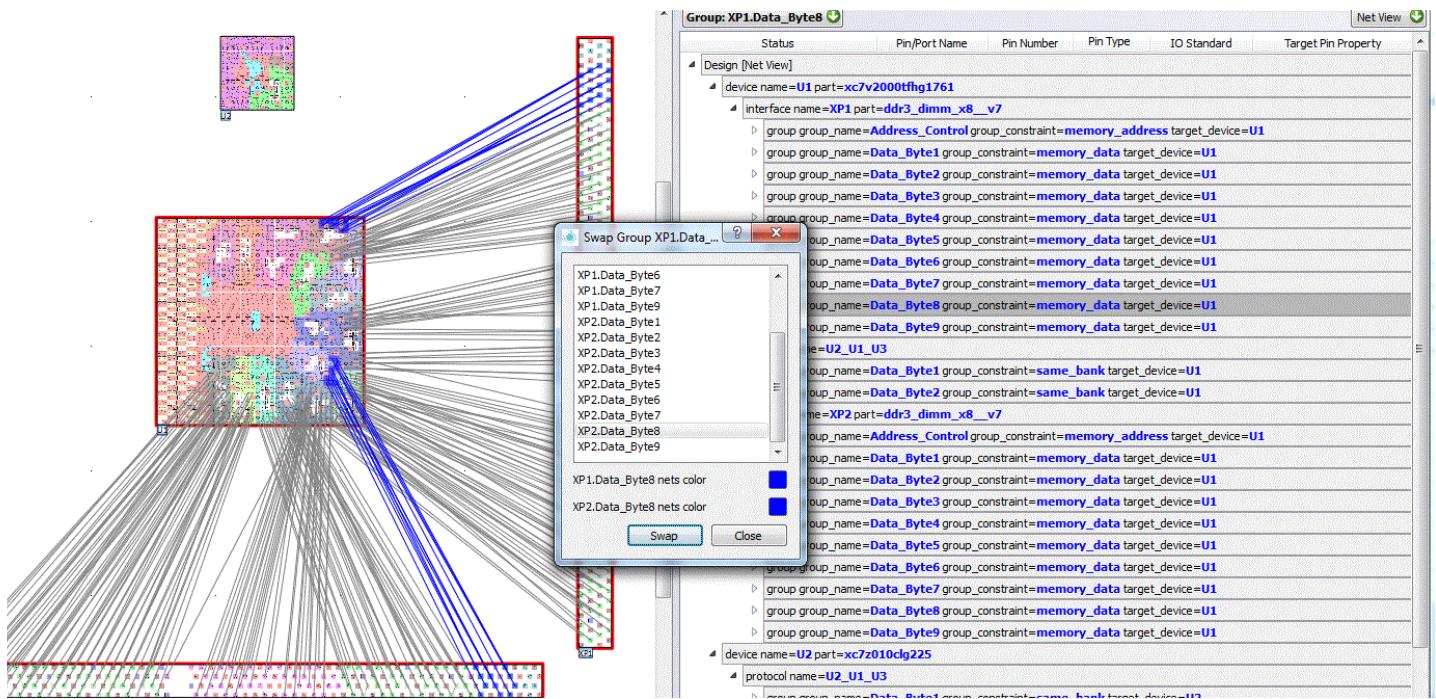
## Synchronizing With FPGA Tools



3. Click on the XP2.Data\_Byte8 in the list.

# Allegro FPGA System Planner User Guide

## Synchronizing With FPGA Tools



**Note:** Multi selection is not allowed in Swap Group pane.

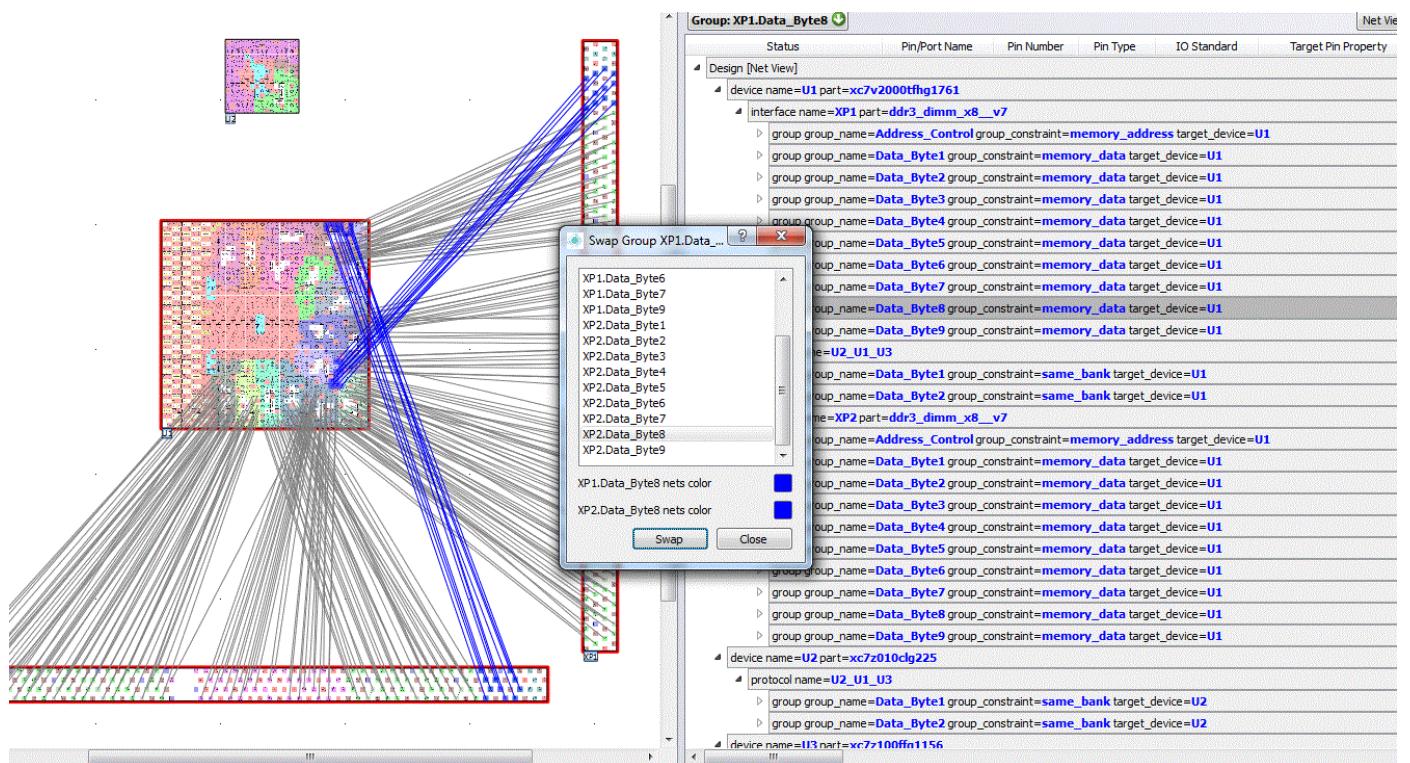
#### 4. Click Swap Group.

**Note:** After you click Swap, the Swap Group dialog box is continued to display. To close the dialog box click Close.

The two groups XP1.Data\_Byte8 and XP2.Data\_Byte8 are swapped.

# Allegro FPGA System Planner User Guide

## Synchronizing With FPGA Tools



---

# **Configuring Devices**

---

This section describes how to configure the devices with various PROMs and external devices. FSP supports the configuration of both Altera and Xilinx devices. This section also highlights the different configuration schemes that FSP supports. It is divided into two sections.

The first section is about the configuration of Xilinx devices, and includes a detailed explanation about PROM and JTAG chaining. The second section describes the configuration of Altera devices with a detailed explanation about PROM and JTAG chain strategy.

## **Xilinx Devices**

This section contains the following topics,

1. [Types of Configuration Schemes](#)
2. [Define PROM Chain](#)
3. [Build and Edit a Prom Chain](#)
4. [Define JTAG chain](#)

### **Types of Configuration Schemes**

FSP supports three types of configuration schemes:

1. Master/Slave serial mode with identical configuration (Same device)
2. Master/Serial serial mode with different configuration (different device)
3. Master/Slave using parallel PROMS in serial mode with identical configuration (same device)

**Master/Slave using parallel PROMS in serial mode with different configuration  
(different device)**

1. SelectMap mode (only identical configuration is applicable)

**Serial Mode with Identical/Different configuration**

In this mode, a Xilinx FPGA, two identical FPGAs, or two different FPGAs are configured by a Xilinx serial configuration PROM. All the Xilinx PROMs are connected in serial mode. While configuring the FPGA, user can use any of the configuration modes— i.e., Master or Slave modes.

For Identical

- D0 pins are shorted and connected to DIN pins of the FPGA
- CEO of the first PROM is connected to CE of the next PROM

For Different

- FPGAs are connected in daisy chain, DO net of PROMs in connected to DIN of first FPGA and DIN of next FPGA is connected to DOUT of previous FPGA
- CEO of the first PROM is connected to CE of the next PROM

**Parallel PROMs in Serial mode with identical/Different configuration**

In this mode, a Xilinx FPGA, two identical FPGAs, or two different FPGAs are configured by using Xilinx parallel PROMs. In this method all the Xilinx PROMs are connected in serial mode. While configuring the FPGA, the user can use any of the configuration modes method—i.e., Master or Slave modes.

(Same as Serial mode)

FPGAs are connected in daisy chain, DO net of PROMs in connected to DIN of first FPGA and DIN of next FPGA is connected to DOUT of previous FPGA

CEO of the first PROM is connected to CE of the next PROM

**Select Map mode**

The Select Map configuration interface provides an 8-bit bidirectional data-bus interface. You can configure the FPGA using a single device or a multi-device Select Map method. In this

method, all the devices should be identical. While configuring the FPGA, the user can use any of the configuration modes method— i.e., Master or Slave modes.

Parallel PROMs in parallel mode.

All the data signals D[7:0] of all devices and PROMs are shorted

Busy is also shorted

## Define PROM Chain

Before you define the PROM chain for your design, make sure you have all the respective FPGAs and PROMs on your canvas. To define the PROM chain, follow the steps below.

1. Choose *Tools – PROM Configurations*, the *Define PROM chain* dialog box appears.

This window helps you build new PROM chain connections, delete the PROM connections, and edit the existing connections. This dialog box also shows you the details for the present PROM chain connections.

2. Click the Add button to create a chain connection between the devices and PROMs, the Add PROM chain dialog box appears on screen.
3. Select a column from the Define PROM Chain dialog box, Click the Delete button to remove the connections.
4. Select a column. Click the Edit button to edit the existing chain connection, and the Edit PROM Chain dialog box appears on screen.



### Messages

The system may generate the following message while working with the Define PROM Chain dialog box.

---

Message	Description
No Device available to make PROM chain	This message appears when you have only PROM components with no devices on canvas and you click the <i>Add</i> button.
No PROM available to make PROM chain	This message appears when you have only devices with no PROM on canvas and you click the <i>Add</i> button.

---

## Build and Edit a Prom Chain

**Note:** The GUI strategy for the Add and Edit PROM chain is the same.

To build a new PROM chain, follow these steps:

When you click the Add button of the Define PROM Chain dialog box, the Add PROM Chain dialog box appears with all the lists of devices and PROMs in their respective sections.

### Serial Mode

1. Type *prom\_chain1* in the PROM chain name text box.
2. To move instances under the Selected PROMs field, select the instance under the Available PROMs field and click the --> button.
3. Click *Serial Mode* under Select Mode options.
4. To move devices under the Selected Devices field, select the device and click the --> button.
5. You can change the order of the device connection for this
6. Click the up and down arrow button.
7. Select the appropriate configuration under the *Select Configuration* field.
8. Click OK to save the settings.

### Select Map Mode

1. Type *prom\_chain1* in the PROM chain name text box.
2. To move instances under the Selected PROMs field, select the instance under the Available PROMs field and click the --> button.

**Note:** Select the two parallel PROMs to make the connections.

3. Click *SelectMap Mode* under the Select Mode options.

The Select Configuration section is disabled for this method. All the devices you have selected should be of the identical type.

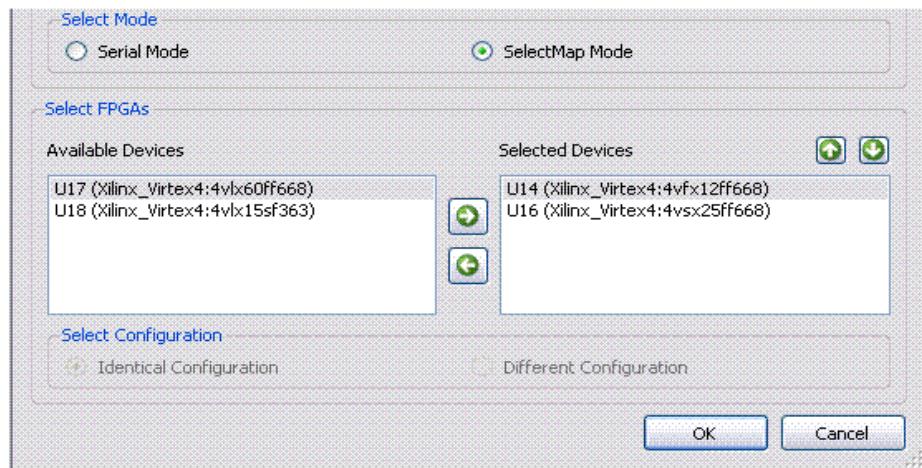
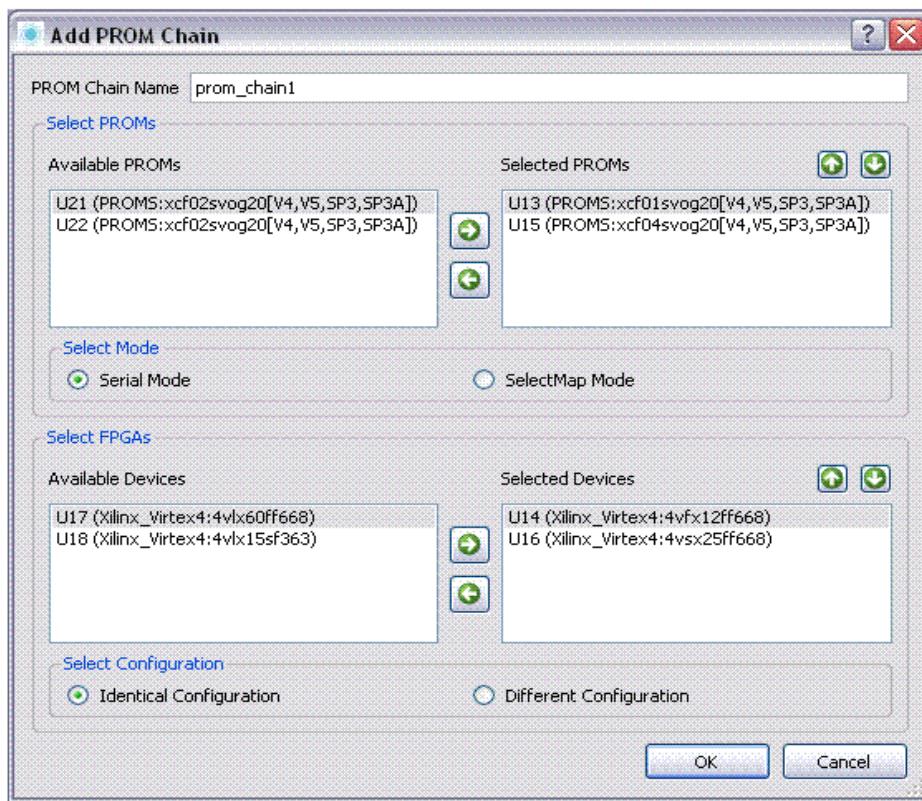
4. To move devices under the Selected Devices field, select the device and click the --> button.

You can change the order of the device connection for that

# Allegro FPGA System Planner User Guide

## Configuring Devices

5. Click the Up Arrow or Down Arrow button.
6. The *Select Configuration* field is disabled.
7. Click OK to save the settings.





The system may generate the following message while you work with the Add PROM Chain dialog box.

Message	Description
No Device selected for chaining	This message appears when you have PROMs under the Selected PROMs field, and you click the OK button without moving the devices to the Selected Devices field.
No PROM selected for chaining	This message appears when you have devices under the Selected Device field and you click OK button without moving the PROMs to the Selected PROMs field.
Do you want to update PROM chain configuration by removing connectivity (if already routed)	This message appears when you try to edit the connection or change the order of devices in the Edit PROM chain dialog box. Click Yes to update the new connection details.
Chain Name cannot be empty	This message appears when you click the OK button without defining the chain name.

## Define JTAG chain

To define a JTAG chain, follow the steps below,

1. Choose *Tools – JTAG Chain*, and the Define JTAG Chain dialog box appears.
2. Click the Add button or right- click under the JTAG Chain field to create the JTAG chain group.  
Now move the instance under the new JTAG chain group
3. Click the -> button to move the instance.

**Note:** You can create as many JTAG chains based on available devices.

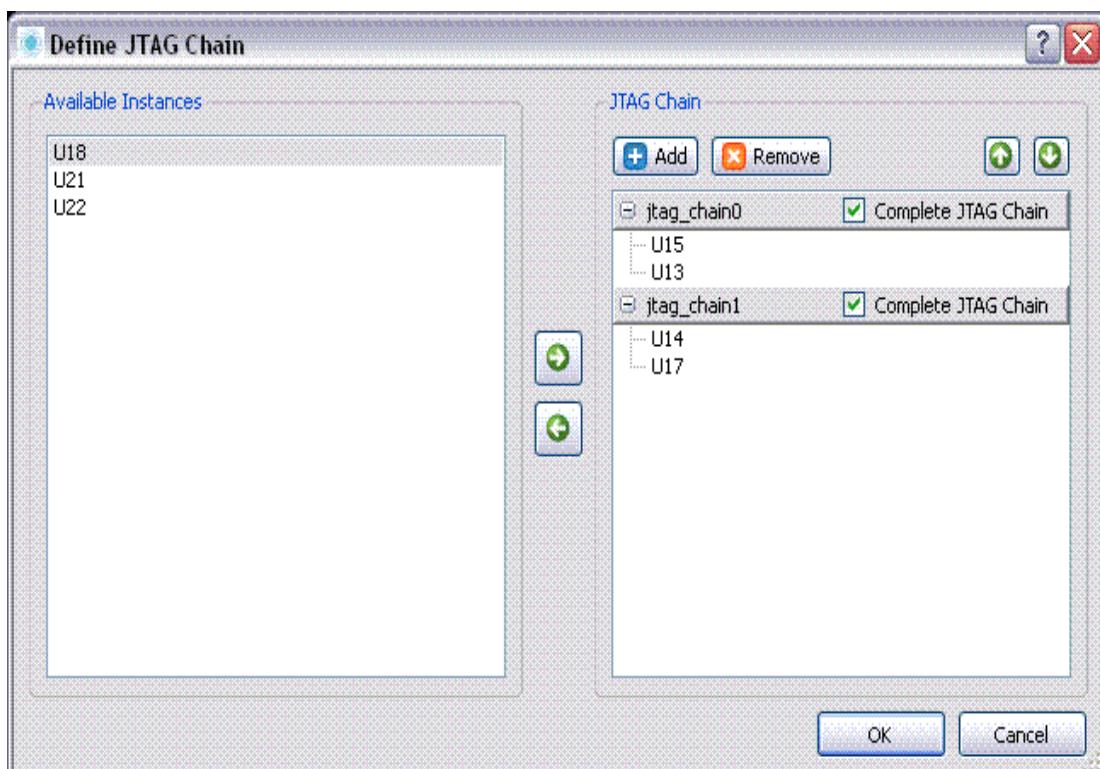
FSP supports JTAG connections between the JTAG connector to FPGA, PROM to FPGA, PROM to PROM, and FPGA to FPGA.

4. Click the Up Arrow or Down Arrow button to change the order of the connections.

## Allegro FPGA System Planner User Guide

### Configuring Devices

5. Select the *Complete JTAG connection* check box to enable the complete JTAG connection loop.  
(The TDO pin of the last FPGA is connected to the TDO pin of the JTAG connector.)
6. Click *OK* to save the settings.
7. Choose *Tools > Run Design* or Click the > icon.



#### *Important*

The system may generate the following message while working with the Define JTAG Chain dialog box.

Message	Description
JTAG chain 'chain name' should have at least two instances	This message appears when you click the OK button and you have only one instance in the JTAG chain group.

Some of the JTAG chain groups are modified (or removed). Do you want to update JTAG chain groups by removing connectivity (if routed)?

This message appears when you do some changes in the JTAG chain groups. An example is to change instance orders, remove an instance from the chain group, and click the OK button.

## Altera Devices

Data on this section is currently unavailable.

---

## Managing Model and Design Versions

---

### Overview

FPGA System Planner base releases and ISR's ships with new features and updated library database versions scheme. This new model and design version scheme ensures that design version and updated library versions both are in sync.

FSP provides you with the flexibility of editing existing designs created in a previous version of the tool in the latest version of the tool. When you create or save a design in FSP, the information about the tool version is stored in the design database. When you open the design, FSP compares the current version of the tool with the tool version stored in the design database. If there is any incompatibility between the design and the tool, a warning or error message is reported.

This chapter describes some of the scenarios, that how design and model versioning can impact your existing work flow behaviors, designs after installing Product Version 23.1.

### Comparing Different Versions

The following examples demonstrate how FSP handles design editing with multiple versions of the tool.

**Table 19-1**

---

<b>Examples</b>	<b>Design and Model Version</b>	<b>Tool Version</b>
1	Design v16.4, v16.4 ISR or v16.5 version	16.6
2	Model v16.4, v16.4 ISR or v16.5 version	16.6
3	Design v16.6	16.5 or lower version
4	Design v16.6	16.3 or lower version
5	Model v16.6	16.3 or lower version

---



At the time of opening the design there are much chances that you see various warning or confirmation dialog boxes. To control the warning or confirmation boxes invocation you can customize through the *Misc* tab of the *Preferences* dialog box.

## Managing v16.5 (or lower version) Design/Model in v16.6

The following scenario arises when you install the FSP v16.6 or 16.6 ISR and have design/model created in v16.5 or lower version.

### Opening v16.5 (or lower version) Design in v16.6

When you open a v16.5 or lower version design in v16.6, FSP displays the following warning message (in Blue color) in Log window.

*The design was last saved in older version of FSP(<@design\_edit\_tool\_version\_id>). Saving the design in current version of FSP can stop this design from opening in older version of FSP.*

**Note:** See the *Example1* in Table 19-1.

### Editing v16.5 (or lower version) model in v16.6

When you try to edit a v16.5 or lower version model in v16.6, FSP prompts you with following error window.

*The @part\_name part model cannot be opened because the part model was last saved using a newer version of the tool (@part\_edit\_tool\_version\_id). Open the design in newer version of FSP <@part\_edit\_tool\_version> and edit the part model.*

Click *OK* to exit the error window.

**Note:** See the *Example2* in Table 19-1.

## Managing v16.6 Design/Model in v16.5

The following scenario arises when you install v16.5, v16.5 ISR, or lower version and have design/model created in v16.6 or 16.6 ISR.

## Opening v16.6 (or higher version) design in v16.5

When you open v16.6 design in v16.5, FSP prompts you with the following warning window message.

*The design was last saved in higher version of FSP <@design\_edit\_tool\_version\_Id>. As some of the functionalities used in the design are not available in current version of the tool being used, opening the design in edit mode is not recommended. Saving the design, can corrupt the design database. Do you want to continue opening the design in edit mode?*

You can perform the following actions when you see the confirmation window:

1. Click *OK*, to open the design for editing.  
Or
2. Click *Read Only*, to open the design if you do not want to edit.  
Or
3. Click *Cancel*, if you do not want to open the design.

**Note:** See the *Example3* in Table 19-1.

## Opening v16.6 design in lower version

When you open v16.6 design in v16.3 or lower version, FSP prompts you with below error window.

*The design cannot be opened because the design was last saved in higher version of FSP <@design\_edit\_tool\_version\_id>. Open the design in higher version of FSP (@design\_edit\_tool\_version\_id).*

In this case the chances of failure is higher compare to the previous case. This case arises when you switch between higher version to very lower version.

Click *OK*, to close the error window.

**Note:** See the *Example4* in Table 19-1.

## Editing v16.6 Model in v16.5 or lower version

When you try to edit a v16.6 model in v16.5 or lower version, FSP prompts you with below warning window.

*The component was last saved using a newer version of FSP (@part\_edit\_tool\_version\_id). Editing the component, using the current version of the tool can cause loss of existing information. Do you want to continue and open the component for editing?*

Click **Yes**, to open and edit the model.

Click **No**, if you want to open the design.

**Note:** See the *Example5* in Table 19-1.

## Migrating a v16.5 Design to v16.6

The FPGA System Planner Version 16.6, ships with new design, library architecture, and project database scheme. If you currently have a design and libraries created in a previous version of FSP, you can automatically upgrade these using current version 16.6. This section provides a brief overview of the scenario you encounter when upgrading your design and libraries in version 16.6.

Migrating of design and libraries can be done by any of the following methods:

- Opening a v16.5 Design in v16.6
- Using TCL Command

### Opening a v16.5 Design in v16.6

The library architectural and project database changes in the current release provides you a dynamic model. In the pre 16.6 release, the FSP project database contains multiple database files. Now with 16.6 ISR, all the information present in the multiple database files is captured in a single project file. When you open any pre 16.6 release design in current version, the tool auto-detects the need for uprev. Tool automatically uprevs the previous project database to the latest version. For example, you open a FSP project created with Release 16.4 or 16.6 in current version 16.6 ISR.

The FSP Uprev process performs the following:

- Uprev the library parts (with new format supported in the current version) for both device and interface components used in the design.
- Creates a new project file called <project\_name>.fsp which contains all the setup information and component information used in the design.
- Creates a new dehdl folder in the output folder. dehdl folders contains dehdl project files.

## Allegro FPGA System Planner User Guide

### Managing Model and Design Versions

---

The FSP uprev process doesn't delete the existing project files or any folders of the previous design. The FSP uprev process generates reports after completion of the uprev process. The reports are stored in the <project\_name>.log file.

Once the design uprev process completes you can remove the following files and folders:

- output
- vinterface (If virtual interface is used in the design)
- <project\_name>.fsp

You can also remove the newly generated library files or move them to other directory for later use. Removing library files from project directory doesn't stop FSP from opening a project.

The following steps describes how FSP uprev the design and libraries when opening a project:

**1. Invoke FSP.**

The Cadence Project Choice dialog box is displayed.

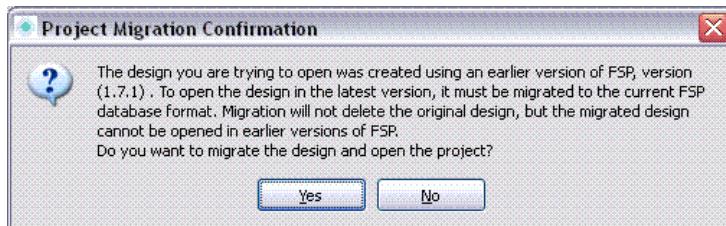
**2. Select a license name and click OK.**

The <License\_Name> - What do you want to do..? dialog box is displayed.

**3. Select *Open Existing..* option.**

**4. Browse to your project directory and select the project file.**

A project migration confirmation window is displayed prompting you about the migration process.



**5. Click Yes to start migration.**

During migration the reports and messages is continuously displayed in log window. The design is displayed in FSP canvas once the migration completes.

## Using TCL Command

The uprev process can also be achieved through TCL commands. FSP provides you separate TCL commands to uprev both design and library files individually.

### Uprev Design

Use UpRevDesignDatabase command to uprev your pre 16.6 release design. The UpRevDesignDatabase command takes the design database specified by you and converts to the latest version database.

#### *Syntax*

```
UpRevDesignDatabase <old_proj_path> <new_project_path>
```

#### *Example,*

```
UpRevDesignDatabase D:/fsp_working/CUSTOMER_DESIGN/qualcomm/core_demo-corrected_lin_integ/core.fsp D:/fsp_working/CUSTOMER_DESIGN/qualcomm/core_demo-corrected_lin_integ/core.fsp
```

### Uprev Pre Library Files

Use UpRevLibraryPartDatabase command to uprev the old library files. The UpRevLibraryPartDatabase command takes the library directory names from the cds.libs and converts all the FSP interface component library files into a single library file.

#### *Syntax*

```
UpRevLibraryPartDatabase <abs_path_to_libs_def_file>.
```

#### *Example,*

```
UpRevLibraryPartDatabase D:/fsp_working/project1_eco_deep_n_wide/cds.libs
```

## Limitations of the Uprev Process

The Uprev process has the following limitation:

- Termination mapping information might be lost. You need to redefine the termination mapping.
- In the pre-design, manually added libraries might be lost such as class libs. You need to manually add the libraries.
- In *Setup DE-HDL Symbol Data* dialog box, if any instances has *Generate Symbol* check box selected then you need to convert them to real part by creating mapping file.

## Migrating a Pre-16.6 Design to Current Version

In the pre-16.6 releases, the rules and mapping files can be accessed from anywhere on the disk. For example, the rules file path contains:

- ‘:’ if a part is placed from the *Libraries*
- ‘./’ if placed from the project directory
- absolute paths if placed from the other directories

The design with these parts is not portable across different systems.

In the current release, the rules and mapping files are fetched only from the directories specified in the *Rules Files Path Editor*. This ensures that the designs are portable across different systems and the search mechanism is aligned with the rest of the Cadence products. If you currently have a design created in a previous version of FSP, it is automatically upgraded to the current version.

For detailed information on the rules and mapping files search mechanism, see the [Setting up the Search Path to the Rules File](#) section.

When you open a 16.6 or pre-16.6 designs in the current release, FSP automatically detects whether a design needs to be migrated. A confirmation window is displayed prompting you to update the rules and mapping files path. After confirmation, FSP automatically migrates the design database to the current version by performing the following steps.

- Creates an `lrf` directory in the project directory.
- Adds `./lrf` path ahead of all the remaining search paths and sets as working directory in the *Rules File Path Editor*.
- Replaces all the absolute and relative reference paths of the design libraries with the search paths in database.
- The updated reference paths of the design libraries are added at top of the existing search paths in the *Rules File Path Editor*. This ensures that during the design migration

process FSP must not refer to other rules and mapping files directories. The order of the reference paths used in the design is retained when they are added to the top of the search paths list.

**Note:** You cannot specify absolute or relative paths using any forms in the FSP for browsing, placing, saving, or adding components after the migration process completes.

**Note:** The migration process does not delete the existing project files or any folders of the previous design.



The migrated design cannot be edited in the lower versions of FSP. After opening the migrated design in a lower version, all the referenced parts may not be detected. It is recommended that you take a backup of the previous design and then perform the migration process to avoid this situation.

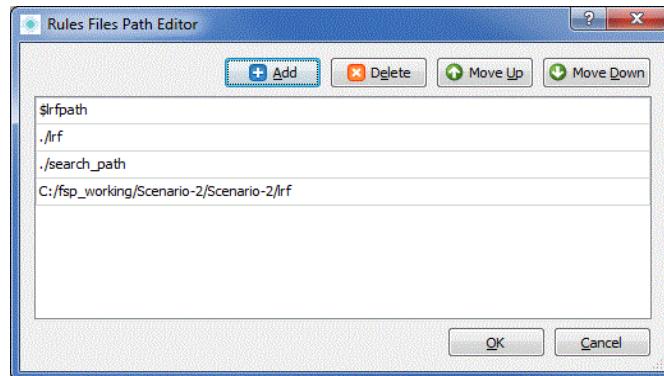


After migration, the design database is marked as *modified* and not directly saved to the disk. You still have a choice to roll-back the design migration until you save the design to the disk.

## Migration Process Example

The following example illustrates how the reference paths of the design libraries are replaced with the search paths when you open a pre-16.6 design in the current version.

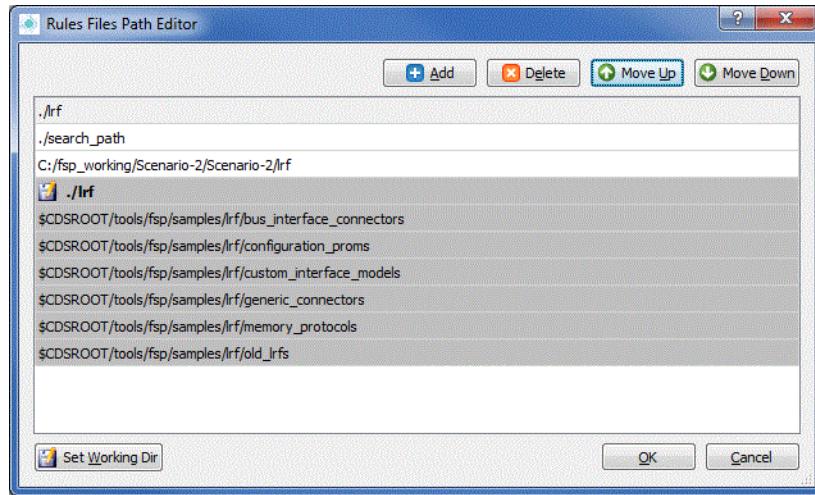
The following snapshot shows that the *Rules File Path Editor* contains a list of relative and/or absolute reference paths to the design libraries.



## Allegro FPGA System Planner User Guide

### Managing Model and Design Versions

When you open the design in the current version, the relative and/or absolute reference paths are automatically replaced with the search paths and added at the top of the list as shown below:



Design libraries can be upgraded by any one of the following:

- [Opening a Pre 16.6 Design](#)
- [Using TCL Command](#)

### Opening a Pre 16.6 Design

To open a pre 16.6 design in the current version, perform the following steps:

1. Click *Open* or press **Ctrl+O**.
2. Browse to your project directory and select the project file.
3. Click *Yes* to start migration.
4. Click the *Save* icon in the to save the design.

When you open a pre-16.6 design in 16.6, a project migration confirmation window is displayed prompting you about the migration process.

During migration the reports and messages are continuously displayed in the log window. The design is displayed on the canvas after migration completes.

## Using TCL Command

The migration process can also be achieved using the OpenProject TCL command.



The migration process using TCL command silently migrate the design without opening the design confirmation window.

## Open Design

Use OpenProject command to open pre 16.6 design. The OpenProject command opens the design specified by you and converts to the latest version database.

### Syntax

```
OpenProject <projectFilePathName>
```

### Example,

```
OpenProject D:/fsp_working/CUSTOMER_DESIGN/qualcomm/core_demo-
corrected_lin_integ/core.fsp.
```

---

## Creating Parts

---

FSP comes with a library of off-the-shelf part models from different vendors. The part models contain necessary logical grouping and constraints required for best performance of FSP. If you do not find the desired part in the Libraries, FSP allows you to create your own part through various part creation methods. Part models can be created manually from ground-up or by importing the pin information available in csv files, DE-HDL and OrCAD symbol files, from existing logical constraints of the other part models.

Besides the part models library, FSP installation also includes Cadence supplied FPGA models. These are non-editable, accessible from the Pin Properties of Device Instance dialog box. If required, you can use FSP to create connectors to be used as custom FPGAs. Connectors have modeling characteristics similar to FPGA's, and pin usage is programmable.

Methods used for creating part models are based on the source from which pin information is imported. This chapter describes various ways of creating part models and instantiating those in your design.

**Note:** The library parts supplied by the FSP and the parts created by you in FSP are logical parts. In order to get the real parts you need to bind them with DE-HDL or OrCAD symbols. For more information on converting component to real component and mapping files see Working with Components and Working with Libraries chapter.

The topics included in this chapter are:

- [Guidelines for Creating Parts](#)
- [Creating Parts from DE-HDL Symbol](#)
- [Creating Parts from OrCAD Symbol](#)
- [Creating Part using Custom Footprint](#)
- [Creating Parts from Existing Interface Rules File](#)
- [Creating Parts from External Files](#)
- [Creating Parts Manually](#)
- [Editing Parts](#)

## Guidelines for Creating Parts

For creating parts, you use Rules Editor, invoked by right-clicking on the instance on canvas. While specifying the pin names and pin numbers in Rules Editor you follow certain rules and guidelines to ensure consistency and to eliminate errors when you transfer the parts to other tools.

The following table describes the general rules for creating parts:

**Table 20-1 Guidelines for Creating Parts**

Names	Guidelines	Invalid Characters	Examples
Signal Name	<ul style="list-style-type: none"><li>■ Use only letters, numbers and the following valid characters:<ul style="list-style-type: none"><li><input type="checkbox"/> underscore (_)</li><li><input type="checkbox"/> dot (.)</li><li><input type="checkbox"/> hash (#)</li><li><input type="checkbox"/> dollar (\$)</li><li><input type="checkbox"/> percentile (%)</li><li><input type="checkbox"/> plus (+)</li><li><input type="checkbox"/> apostrophe (`)</li><li><input type="checkbox"/> dash (-)</li><li><input type="checkbox"/> plus (+)</li></ul></li><li>■ Maximum of 255 characters is allowed.</li><li>■ Use the following brackets for sizeable or vector pins &lt;&gt; {} [] ()</li></ul>	<ul style="list-style-type: none"><li><input type="checkbox"/> White space</li><li><input type="checkbox"/> Backslash (\)</li><li><input type="checkbox"/> Forward slash (/)</li><li><input type="checkbox"/> Exclamation mark (!)</li><li><input type="checkbox"/> At sign (@)</li><li><input type="checkbox"/> Left single quote (')</li><li><input type="checkbox"/> Tilde (~)</li><li><input type="checkbox"/> comma (,)</li></ul>	<p>Correct usage: data1_&lt;0&gt; data.1[0] data1#A21 data1 @&lt;0&gt;</p> <p>Incorrect Usage: X~signalN[0] and X,signal[2]</p>

## Allegro FPGA System Planner User Guide

### Creating Parts

---

Pin Number	■ Use the alphanumeric character including underscore ■ Use a maximum of 16 characters.	<input type="checkbox"/> Backslash (\)      Correct usage:  <input type="checkbox"/> Double quotation mark ("')      Incorrect usage:  <input type="checkbox"/> Exclamation mark (!)      w@1,3!3  <input type="checkbox"/> Ampersand (&)  <input type="checkbox"/> At sign (@)  <input type="checkbox"/> Tilde (~)  <input type="checkbox"/> Grave acent (`)  <input type="checkbox"/> ^  <input type="checkbox"/> Less-than (<)  <input type="checkbox"/> Greater-than (>)  <input type="checkbox"/> Period (.)  <input type="checkbox"/> Comma (,)  <input type="checkbox"/> Colon (:)  <input type="checkbox"/> Semicolon (;)  <input type="checkbox"/> Left curly brace ({})  <input type="checkbox"/> Right curly brace (})
------------	--------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## Using Rules Editor

To create parts in FSP you use Rules Editor dialog box. The Rules Editor provides an intuitive and easy way of creating a part. It provides a spreadsheet editor view through which all the logical information can be specified. Constraints can be specified manually or can be imported from existing interface rules file or external data files. You can also use the Rules Editor to modify the part constraints. After creating the parts you can include the new or modified part as part of your own library. For detailed information on fields and buttons of Rules Editor see [Rules Editor](#) section.

You can use the Rules Editor for following ways:

- To create part manually from scratch.
- To create part using existing library interface model information.
- To create part using external data files.
- To create part using symbols and footprints information.
- To edit the model definition for a part.

## Invoking Rules Editor

To invoke the model editor:

1. Do any of the following:
    - Choose *Library – Create Part – Interface*.
- Or
- a. In Libraries click *Interface Rules* tab.
  - b. Right-click on rules file name and select *Edit Component*.
- Or
- a. Right-click on the instance.
  - b. From the pop-up menu select *Edit Rules*.

## Creating Parts from DE-HDL Symbol

To create a part based on symbols used in DE-HDL, you use logical symbol data and footprint information part. The pin information details are extracted from chips.prt file and pin location values from dra file. It is not mandatory to have dra file, you can even create the part from chips.prt and later manually specify the X and Y location for the pins in Rules Editor. After creating the part you can add it to your design.

Pin information read from the chips.prt file:

- Pin Names
- Pin Numbers
- Pin Types
- IO Standards

You can create the part from DE-HDL symbols and split symbols (both the Asymmetrical and Symmetrical).

This section covers the following topics:

- [Understanding Symmetrical Symbol](#)
- [Steps to Create Part from Symmetrical Symbol](#)
- [Steps to Create Part from Asymmetrical Symbol](#)

### Understanding Symmetrical Symbol

This section describes the pin naming convention used while creating the parts from symmetrical symbol.

FSP follows a unique pin name generation policy at the time of model creation using Symmetrical split symbols. The following syntax are used to define the pin names.

#### Syntax

<section name>\_\_<pinname>

<section name> is the additional attribute attached to each pin node. <section name> attribute indicates each pin belongs to different symmetrical section. The <section name> is followed by a double underscore (\_\_). <pinname> is the pin name specified in the chips.prt file.

For example, Pin Name after translation = “S1\_\_<A>”.

S1 indicates the section name and A refers to symbol pin name.

The following three examples demonstrate how FSP handles different types of symmetrical split symbols.

### ***Example1***

Use a 74LS01 DE-HDL symbol to create a model. An example of 74LS01 chips.prt file is shown below.

```
primitive '74LS01';
pin
'A':
 INPUT_LOAD='(-0.010000,0.010000)';
 PIN_NUMBER='(2,5,8,11)';
end_pin;
```

In example above, pin number 2, 5, 8, 11 have the same pin name as A. While importing chips.prt file the pin names are converted as shown below:

---

<b>Pin Number</b>	<b>Pin Name</b>
2	S1_A
5	S2_A
8	S3_A
11	S4_A

---

### ***Example2***

Use a DE-HDL symbol with vector pins. An example of chips.prt file is shown below.

```
pin
'A'<0>:
 PIN_NUMBER='(36,47)';
 INPUT_LOAD='(-0.01,0.01)';
'A'<1>:
 PIN_NUMBER='(35,46)';
 INPUT_LOAD='(-0.01,0.01);
```

## Allegro FPGA System Planner User Guide

### Creating Parts

---

```
'A'<2>:
 PIN_NUMBER='(33,44)';
 INPUT_LOAD='(-0.01,0.01)';
'A'<3>:
 PIN_NUMBER='(32,43)';
 INPUT_LOAD='(-0.01,0.01)';
```

In chips.prt data above, the pin names with <> notation, are treated as bus pins. While importing chips.prt file the pin names are converted as shown below:

---

Pin Number	Pin Name
36	S1_A<0>
47	S2_A<0>
33	S3_A<1>
44	S3_A<1>
30	S4_A<2>
41	S4_A<2>
32	S5_A<3>
43	S5_A<3>

---

### ***Example3***

The chips.prt with sizable pins is shown below.

```
pin
'-Y'<1..0>:
 PIN_NUMBER='(11,8,6,3)';
 OUTPUT_LOAD='(8.0,-0.4)';
'A-'<0>:
 PIN_NUMBER='(13,10,5,2)';
 PIN_GROUP='1';
 INPUT_LOAD='(-0.4,0.02)';
'A+'<0>:
 PIN_NUMBER='(12,9,4,1)';
 PIN_GROUP='1';
 INPUT_LOAD='(-0.4,0.02)';
```

# Allegro FPGA System Planner User Guide

## Creating Parts

```
end_pin;
```

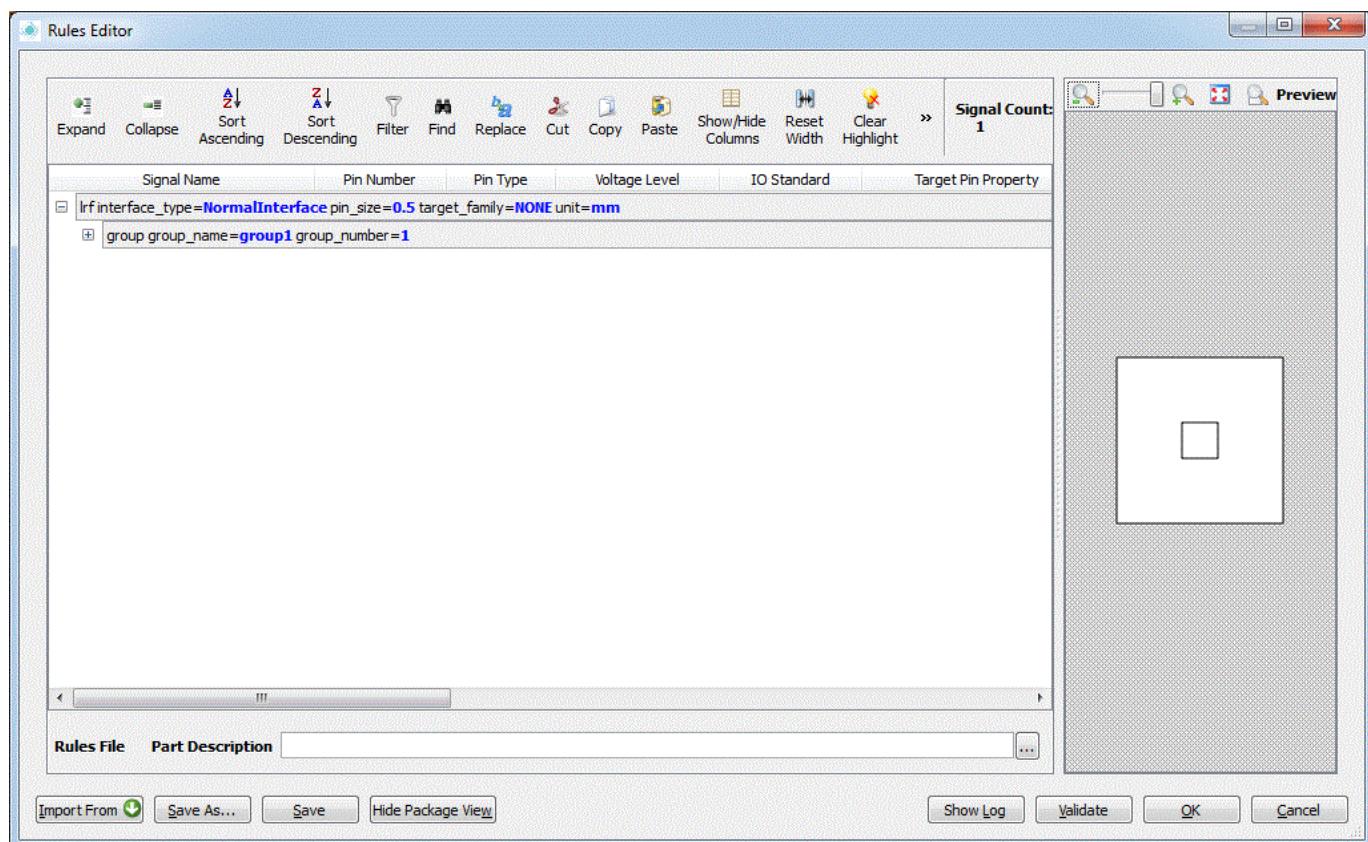
During pin name conversion, the <size-1..0> string is eliminated and the remaining string is considered as pin names. For example, in Y<SIZE-1..0>\* string the <SIZE-1..0> is removed and the two Y and \* character is considered.

## Steps to Create Part from Symmetrical Symbol

To create a part from symmetrical symbol perform the following steps:

1. Choose *Library – Create Part – Interface*.

The Rules Editor is displayed.



2. Click *Import From*.
3. From pop-up menu list select *DE-HDL Symbol*.

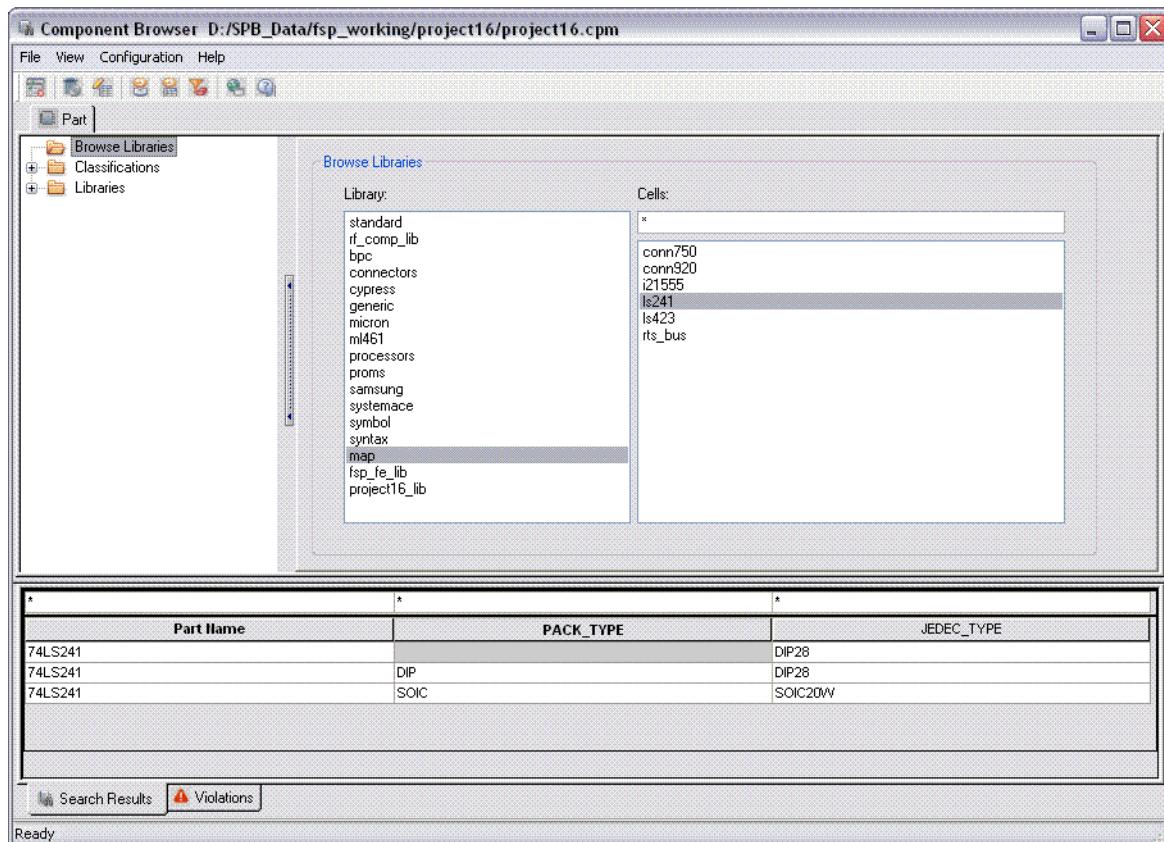
A confirmation window pop-ups prompting you about the overriding of existing pin definitions.

# Allegro FPGA System Planner User Guide

## Creating Parts

### 4. Click Yes.

The Component Browser is displayed.

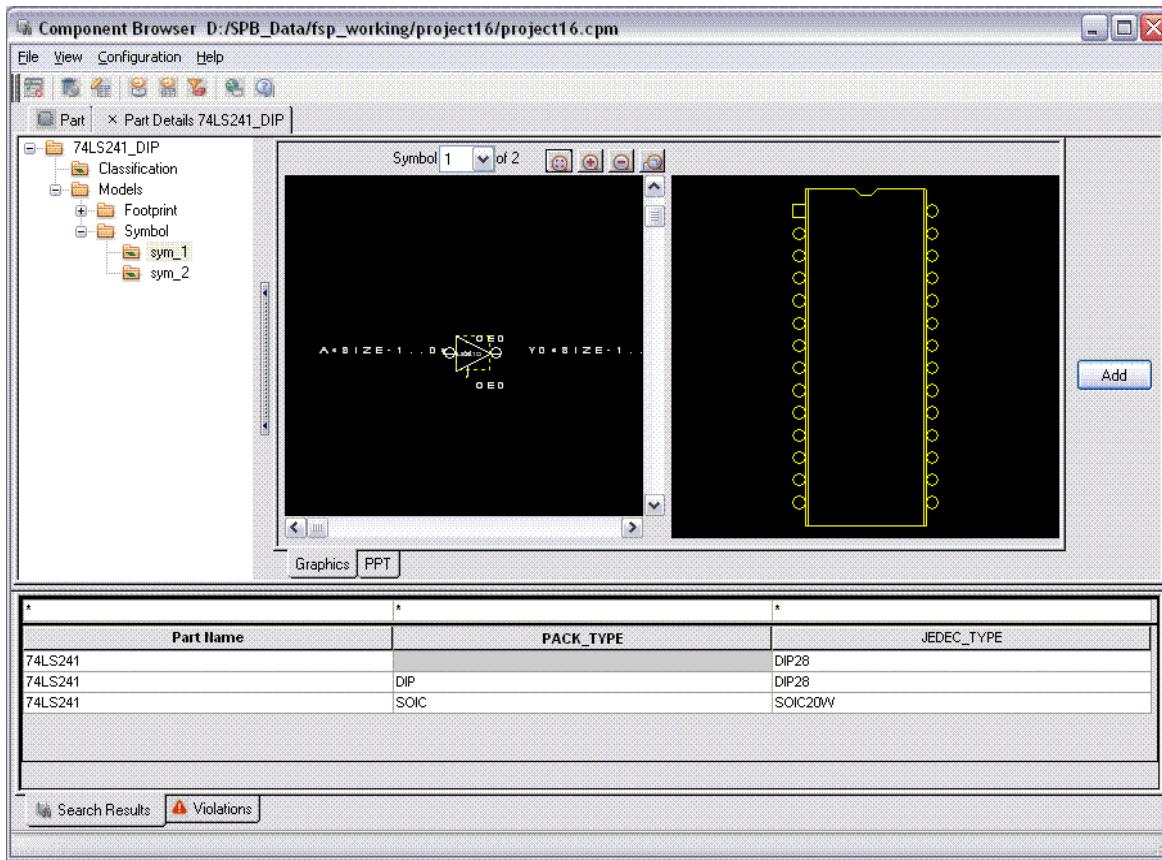


### 5. Select a cell name and click *PTF Row*.

### 6. Click *Select*.

# Allegro FPGA System Planner User Guide

## Creating Parts



### 7. Click Add.

The Rules Editor displays the logical pin constraints information. The duplicate pin names are converted into unique names.

### 8. Specify the group constraints and modify the pin constraints as required.

For detailed information on editing group and pin constraints see [Editing Parts](#) section.

### 9. Click Validate to verify the pin definitions.

### 10. Click Save As to save the pin informations.

**Note:** You may also use the *Save* option to save the part since you are creating the part from initial. Both *Save As* and *Save* options invokes the similar type of dialog box. Once you save the part, the *Save* option instantly save the part definitions in the file without opening the dialog box.

The Save Rules File dialog box is displayed.

### 11. Enter the name for the rules file in the *Rules File* field.

## Allegro FPGA System Planner User Guide

### Creating Parts

12. Click and select a directory path from the *Output Directory* drop-down list, where you want to save the rules file.
13. Click *OK* of the Save Rules File dialog box.
14. Click *OK* of the Rules Editor.

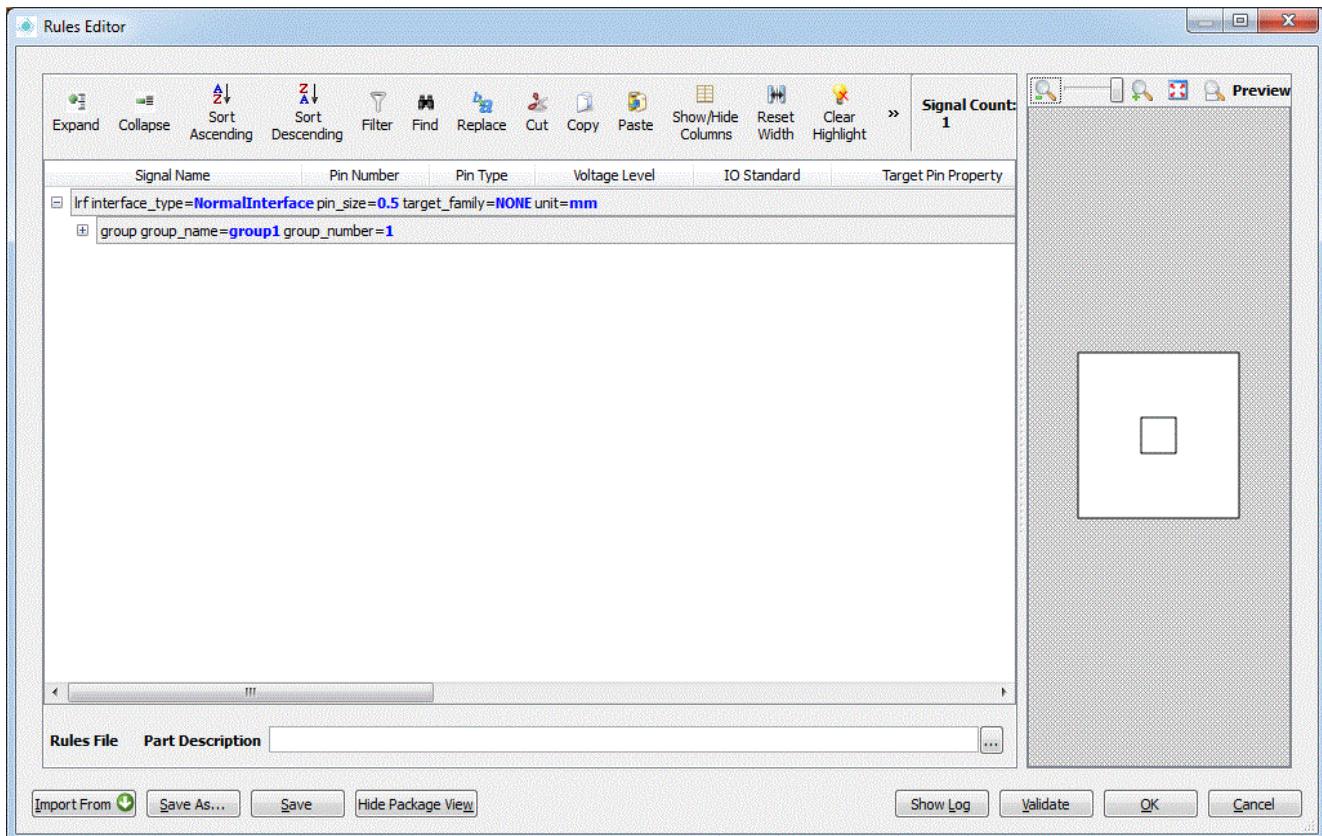
The part is created at the specified library.

### Steps to Create Part from Asymmetrical Symbol

To create a part from asymmetrical symbol, perform the following steps:

1. Choose *Library – Create Part – Interface*.

The Rules Editor is displayed.



2. Click *Import From*.
3. From pop-up menu list select *DE-HDL Symbol*.

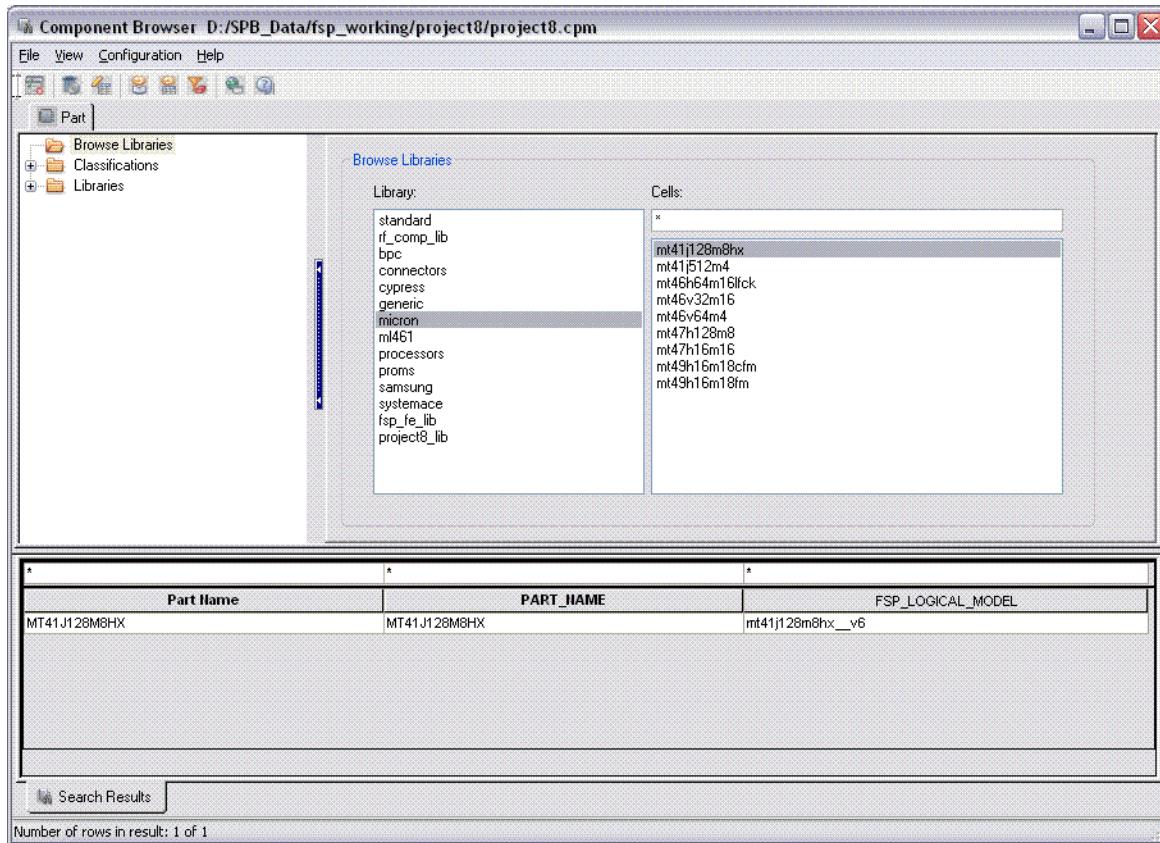
## Allegro FPGA System Planner User Guide

### Creating Parts

A confirmation window pop-ups prompting you about the overriding of existing pin definitions.

4. Click Yes.

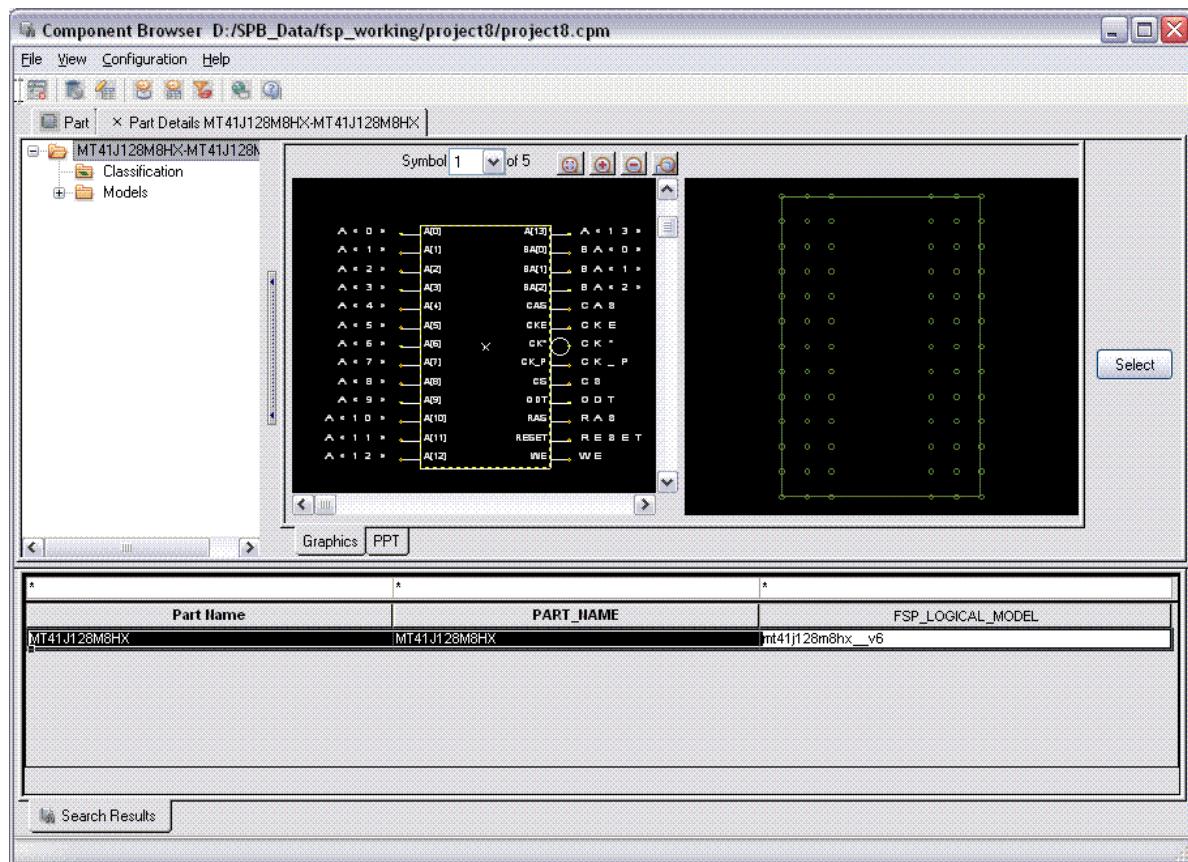
The Component Browser is displayed.



5. Select a cell name and click *PTF Row*.

# Allegro FPGA System Planner User Guide

## Creating Parts

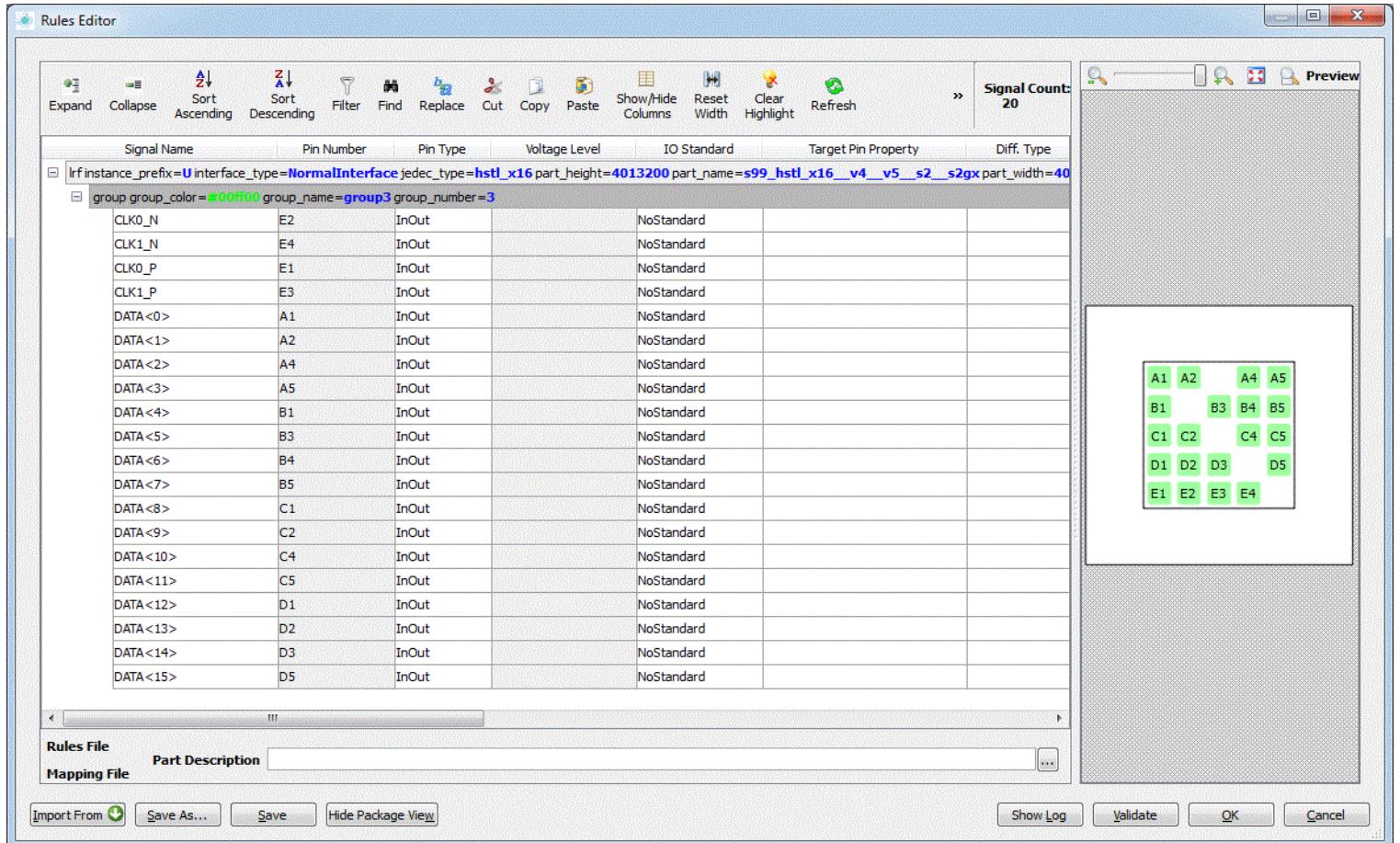


### 6. Click Select.

The Rules Editor displays the logical pin constraint information.

# Allegro FPGA System Planner User Guide

## Creating Parts



You can save the part or proceed further with modifying the logical constraints of the part.

7. Specify the group constraints and pin constraints if required.

For detailed information on editing group and pin constraints see [Editing Parts](#) section.

8. Click *Validate* to verify the pin definitions.

9. Click *Save As* to save the pin informations.

**Note:** You may also use the *Save* option to save the part since you are creating the part from initial. Both *Save As* and *Save* options invokes the similar type of dialog box. Once you save the part, the *Save* option instantly save the part definitions in the file without opening the dialog box.

The Save Rules File dialog box is displayed.

10. Enter the name for the rules file in the *Rules File* field.

## **Allegro FPGA System Planner User Guide**

### Creating Parts

---

- 11.** Click and select a directory path from the *Output Directory* drop-down list, where you want to save the rules file.
- 12.** Click *OK* of the Save Rules File dialog box.
- 13.** Click *OK* of the Rules Editor.

The part is created at the specified library.

## Creating Parts from OrCAD Symbol

You can also create the parts from available OrCAD symbols. The procedure is very much similar to creating part from DE-HDL symbol method. You select a symbol and footprint then modify the logical constraints if required and then add the part in your design or existing library.

**Note:** You can create the part from OrCAD symbol then only when you are working in OrCAD schematic environment.

To create a part from OrCAD symbol:

1. Choose *Library – Create Part From – Interface*.

The Rules Editor is displayed.

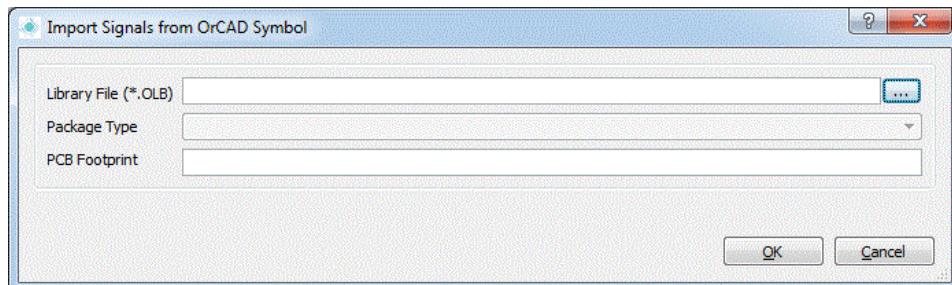
2. Click *Import From*.

3. From the pop-up menu select *OrCAD Symbol*.

A confirmation window pop-ups prompting you about the overriding of existing pin definitions.

4. Click *Yes*.

The *Import Signals from OrCAD Symbol* dialog box is displayed.



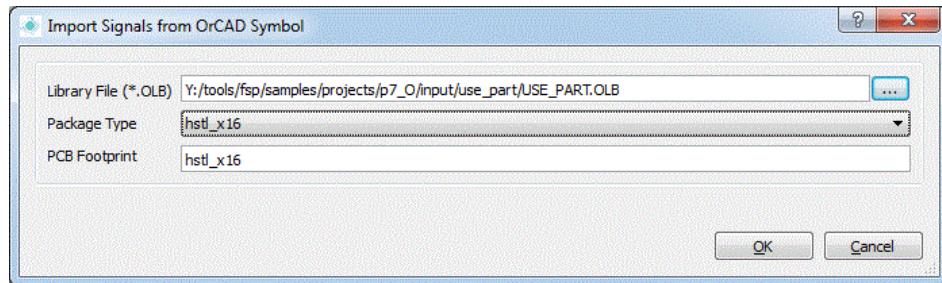
5. Specify the OrCAD symbol (.olb) file name and path or click *browse ...* to browse to the location where the file exist.

6. Select the type of package from *Package Type* drop down list.

On selecting the package type the footprint name is automatically displayed in *PCB Footprint* text box.

## Allegro FPGA System Planner User Guide

### Creating Parts



#### 7. Click *OK*.

The logical pin and group details are displayed in Rules Editor. You can save the part by clicking *OK* or proceed further with modifying the logical constraints of the part.

#### 8. Specify the group constraints and modify pin constraints if required.

For detailed information on editing pin and group constraints see [Editing Parts](#) section.

#### 9. Click *Validate* to verify the pin definitions.

#### 10. Click *Save As* to save the pin informations.

**Note:** You may also use the *Save* option to save the part since you are creating the part from initial. Both *Save As* and *Save* options invokes the similar type of dialog box. Once you save the part, the *Save* option instantly save the part definitions in the file without opening the dialog box.

The Save Rules File dialog box is displayed.

#### 11. Enter the name for the rules file in the *Rules File* field.

#### 12. Click and select a directory path from the *Output Directory* drop-down list, where you want to save the rules file.

#### 13. Click *OK* of the Save Rules File dialog box.

#### 14. Click *OK* of the Rules Editor.

The part is created at the specified library.

## **Creating Part using Custom Footprint**

You can create new part for your design using custom footprints information. The Custom Footprint feature enables you to decide the footprint for your part. When deciding your own custom footprint, the information that needs to be specified is pin configuration parameters such as pin size, row pitch and number of pins. The pin names and X, Y location values are derived from the information you provide in Custom Footprint dialog box. The Custom Footprint dialog box also provides you the preview of your symbol footprint.

Creating a new part using custom footprint process has following steps:

- Selecting a Footprint

FSP allows you to choose the pin configurations from following footprints:

- BGA/Connector/PGA
- DIP/SOIC/SSOP/TSOP
- QFP/TQFP

Select any one of the footprint to create the part.

- Specifying Pin Configuration Parameters

For detailed information see [Creating Parts Manually](#) section.

- Defining Logical Constraints

For more information on defining logical constraints see [Creating Parts Manually](#) section.

### **Points to Remember While Creating the Part Using Custom Footprint**

The following points you must know while using creating the part using custom footprint:

- Pin numbers and (X, Y) locations are automatically generated by FSP based on parameters specified by you.
- Pin numbers and (X, Y) locations values can also be changed while defining the logical constraints (Using the Update Location option in the Rules Editor).

#### ***Steps to create part from custom footprint***

To create a new part using Custom dialog box perform the following steps:

1. In Rules Editor form do the following:

# Allegro FPGA System Planner User Guide

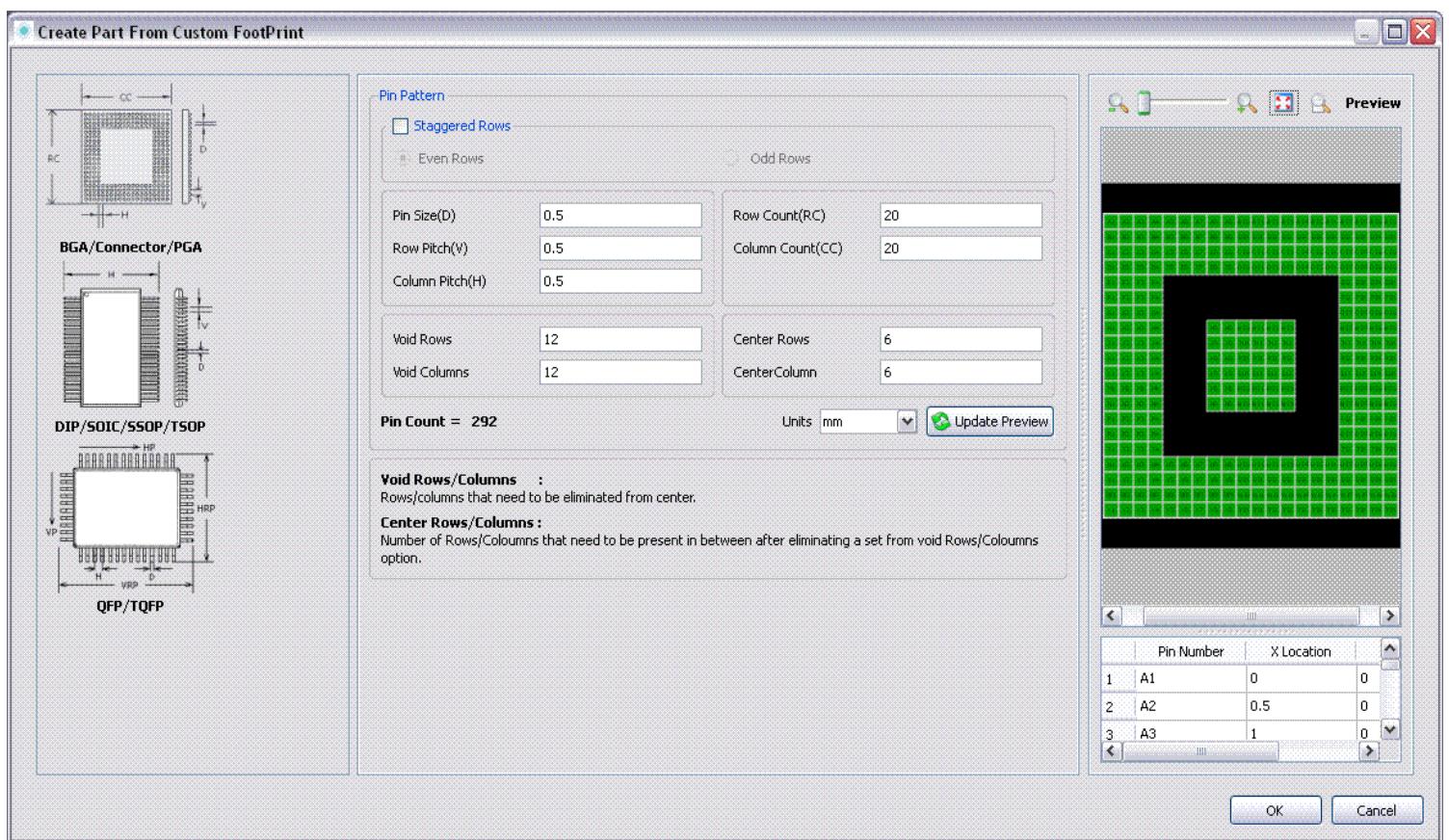
## Creating Parts

- Choose *Import Form – Custom Footprint*.

A confirming window is displayed prompting you about overriding of existing information in Rules Editor.

- Click Yes.

The Create Part From Custom Footprint dialog box is displayed.



- Select a footprint from three footprint images displayed in left side pane.

Based on your selection pin parameters are displayed. If required you can modify the parameters.

- Specify the pin configuration parameters.

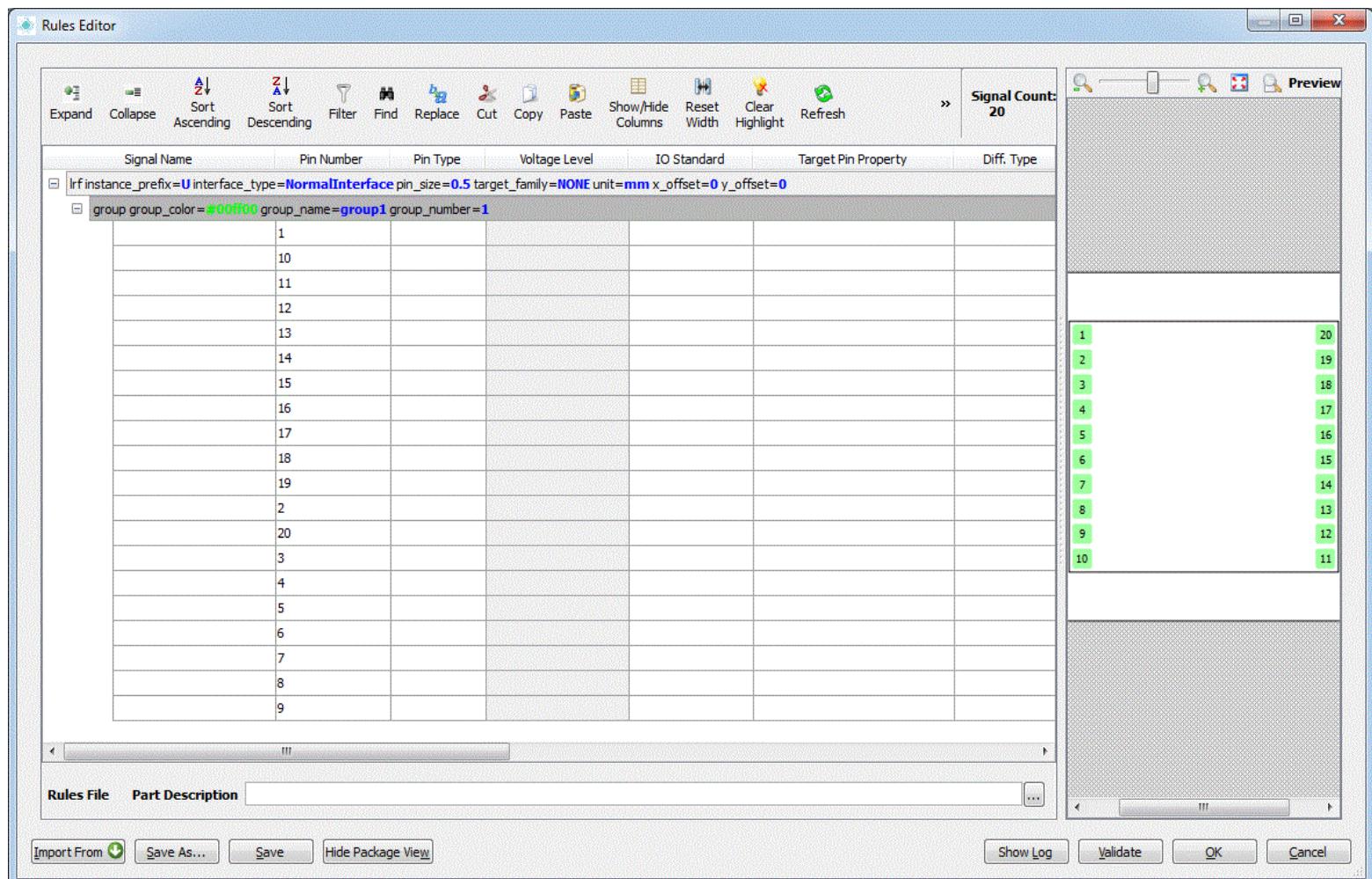
See Specifying Pin Configuration Parameters section in [Creating Part using Custom Footprint](#) section for more information.

- Click OK.

# Allegro FPGA System Planner User Guide

## Creating Parts

The pin numbers and X, Y location values is displayed in respective columns of Rules Editor.



6. Modify and specify the logical constraints of the part as required.

See [Creating Parts Manually](#) for detailed information.

7. Click *Validate* to verify the pin constraints.

8. Click *Save As* to save the pin informations.

**Note:** You may also use the *Save* option to save the part since you are creating the part from initial. Both *Save As* and *Save* options invokes the similar type of dialog box. Once you save the part, the *Save* option instantly save the part definitions in the file without opening the dialog box.

The Save Rules File dialog box is displayed.

## **Allegro FPGA System Planner User Guide**

### Creating Parts

---

9. Enter the name for the rules file in the *Rules File* field.
10. Click and select a directory path from the *Output Directory* drop-down list, where you want to save the rules file.
11. Click *OK* of the Save Rules File dialog box.
12. Click *OK* of the Rules Editor.

The part is created at specified path.

## **Creating Parts from Existing Interface Rules File**

To create part from existing interface rules file you use Rules Editor.

The part creation begins with selecting existing part (rule file) followed by editing logical constraints if necessary and ends with saving the part. You can then add the part in your design.

To create a part using existing interface rules file:

1. Do one of the following to invoke Rules Editor:

Choose *Library – Create Part – Interface*.

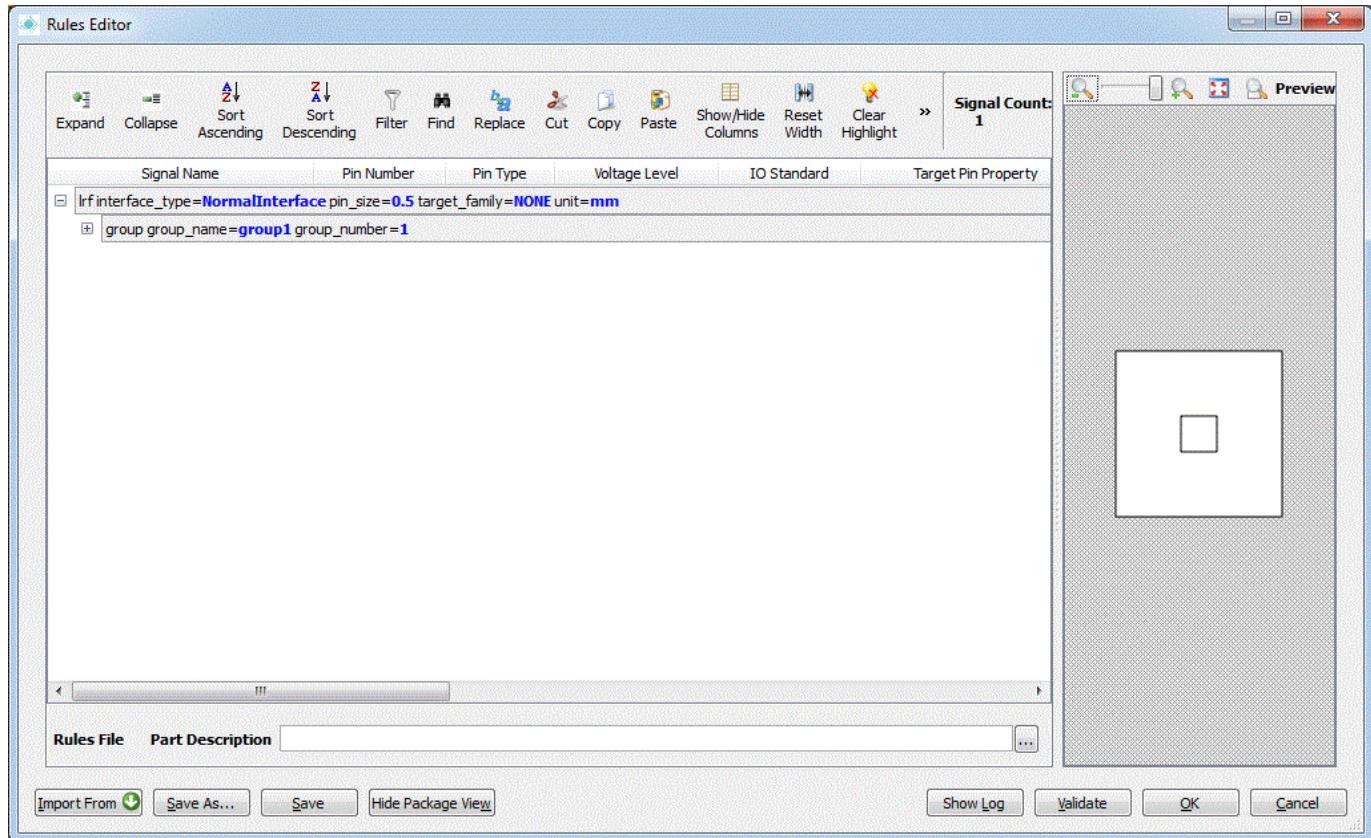
Or

- a. Click *Rules* tab in Libraries.
- b. Click + to expand the tree view structure.
- c. Right-click on a part name and click *Edit Component*.

The Rules Editor is displayed.

# Allegro FPGA System Planner User Guide

## Creating Parts



2. Click *Import From* at bottom side of the window.

A small pop-up menu is displayed.

3. Click *Edit Rules*.

A confirmation windows pops up prompting you about the overriding of existing definition.

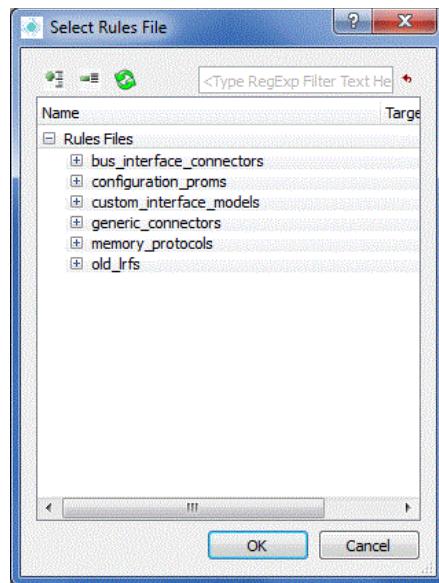
4. Click *Yes*.

The Select Rules File dialog box lists the interface rules file stored in FSP installation directory.

## Allegro FPGA System Planner User Guide

### Creating Parts

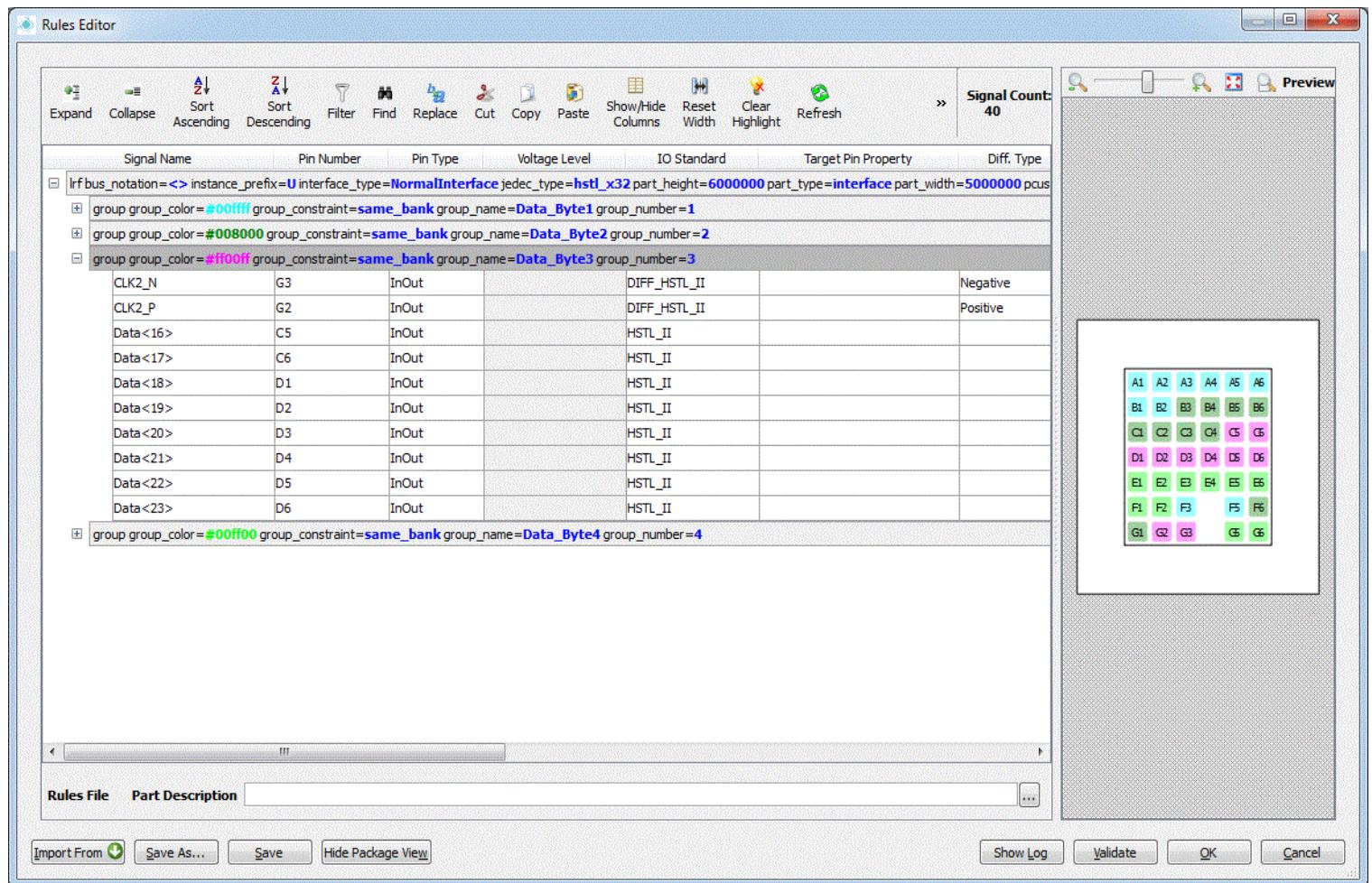
---



5. Do one of the following to select rule file from Select Rules File dialog box:
  - Click + to expand the tree view structure and select a part name from the list.
  - Search for the desired part name by entering name in *Filter* text box and select the part.

# Allegro FPGA System Planner User Guide

## Creating Parts



The part pin and group details are imported and displayed in Rules Editor.

You can either save the model by specifying the rule file name with path in Rules File field and click *OK* or you can proceed further to edit the model.

**Note:** Any incorrect information present in the rules file will halt the importing rules file process.

6. Modify the group and pin constraints as required.

For detailed information on editing group and pin constraints see [Editing Parts](#) section.

7. Click *Validate* to verify the logical information.

8. Click *Save As* to save the pin informations.

**Note:** You may also use the *Save* option to save the part since you are creating the part from initial. Both *Save As* and *Save* options invokes the similar type of dialog box. Once you save

the part, the *Save* option instantly save the part definitions in the file without opening the dialog box.

The Save Rules File dialog box is displayed.

9. Enter the name for the rules file in the *Rules File* field.
10. Click and select a directory path from the *Output Directory* drop-down list, where you want to save the rules file.
11. Click *OK* of the Save Rules File dialog box.
12. Click *OK* of the Rules Editor.

## Creating Parts from External Files



Importing pin and group properties through *Import from CSV* dialog box is available only in Rules Editor and Rules Instance Editor. In other editors the *Import from CSV* dialog box imports all the part information in a single group.

FSP allows you to import part details from external files of different format for example, comma separated value (.csv) file, tab separated file, or space separated file and create parts from it. The entries in the different files must be in the name and value format. By default all the header keywords that are available in Rules Editor are supported. Its not mandatory to have all the header with values in the files. Values for any missing headers can be specified later after importing the file in Rules Editor.

In FPGA System Planner, every part information is organized into groups and pins. These information are stored as headers and their values and can be visualize in Rules Editor and Rules Instance Editor. You are allowed to create the part exactly in the same structure containing groups and pins by importing external file. In order to create you must have Group Properties and their values specified as a comma separated list or tab separated list in a row of the external file.

### **Group Properties**

The group properties are determined in the following way:

- Entries under the Group Name column are read as group names.
- Entries under the Group Number column are read as group numbers.
- Entries under the Group Constraint column are read as group constraint.
- Entries under the Group Color column are read as group color.

### **Pin Properties**

The following header values are imported as pin properties:

- Signal Name
- Symbol Pin Name
- X Location
- y Location

## Allegro FPGA System Planner User Guide

### Creating Parts

---

- Diff. Type
- Pin Type
- IO Standard
- Pin Number
- Diff Pair Pin
- Part Unit
- Family to Connect
- Pin Size
- Part Description
- Part Custom Attribute
- Part Width
- Part Type
- Part Height
- JEDEC Type
- Interface Type
- Ref.Des Prefix
- Bus Notation

**Note:** Not all the headers above are required at initial part creation process.



While importing you can change the header keywords according to your specifications. Changing the header keywords is important step, because the values are derived from the files based on the header keyword you specify in Import CSV dialog box.

### Comma Separated File (CSV) Import Example

Consider the CSV file below which has the following entries. To create the part along with the group properties then you must prepare your CSV files in exactly same format shown below.

# Allegro FPGA System Planner User Guide

## Creating Parts

---

Group Col	Group Constraint	Group Name	Group Num	bus	notat	Ref.	Des.	Interface	JEDEC	TY	part	type	Pin Size	Family	To Diff. Pair	F	Pin Numb	IO Standar	Pin Type	Diff. Type	X Location	Y Location	Signal Name
#008000	same_bank	Data_Byte1	1	◇	U			NormalInthstl_x16	interface	1	V4,V5,S2,CLK0_P	E2		DIFF_HSTL	InOut		Negative		0	0	CLK0_N		
#008000	same_bank	Data_Byte1	1	◇	U			NormalInthstl_x16	interface	1	V4,V5,S2,CLK0_N	E1		DIFF_HSTL	InOut		Positive		0	0	CLK0_P		
#008000	same_bank	Data_Byte1	1	◇	U			NormalInthstl_x16	interface	1	V4,V5,S2,S2GX	A1		HSTL_II	InOut				0	4	Data<0>		
#008000	same_bank	Data_Byte1	1	◇	U			NormalInthstl_x16	interface	1	V4,V5,S2,S2GX	A2		HSTL_II	InOut				0	4	Data<1>		
#008000	same_bank	Data_Byte1	1	◇	U			NormalInthstl_x16	interface	1	V4,V5,S2,S2GX	A4		HSTL_II	InOut				0	4	Data<2>		
#008000	same_bank	Data_Byte1	1	◇	U			NormalInthstl_x16	interface	1	V4,V5,S2,S2GX	A5		HSTL_II	InOut				0	4	Data<3>		
#008000	same_bank	Data_Byte1	1	◇	U			NormalInthstl_x16	interface	1	V4,V5,S2,S2GX	B1		HSTL_II	InOut				0	3	Data<4>		
#008000	same_bank	Data_Byte1	1	◇	U			NormalInthstl_x16	interface	1	V4,V5,S2,S2GX	B3		HSTL_I	InOut				0	3	Data<5>		
#008000	same_bank	Data_Byte1	1	◇	U			NormalInthstl_x16	interface	1	V4,V5,S2,S2GX	B4		HSTL_II	InOut				0	3	Data<6>		
#008000	same_bank	Data_Byte1	1	◇	U			NormalInthstl_x16	interface	1	V4,V5,S2,S2GX	B5		HSTL_II	InOut				0	3	Data<7>		
#008080	same_bank	Data_Byte2	2	◇	U			NormalInthstl_x16	interface	1	V4,V5,S2,CLK1_P	E4		DIFF_HSTL	InOut		Negative		0	0	CLK1_N		
#008080	same_bank	Data_Byte2	2	◇	U			NormalInthstl_x16	interface	1	V4,V5,S2,CLK1_N	E3		DIFF_HSTL	InOut		Positive		0	0	CLK1_P		
#008080	same_bank	Data_Byte2	2	◇	U			NormalInthstl_x16	interface	1	V4,V5,S2,S2GX	C1		HSTL_II	InOut				0	2	Data<8>		
#008080	same_bank	Data_Byte2	2	◇	U			NormalInthstl_x16	interface	1	V4,V5,S2,S2GX	C2		HSTL_II	InOut				0	2	Data<9>		
#008080	same_bank	Data_Byte2	2	◇	U			NormalInthstl_x16	interface	1	V4,V5,S2,S2GX	C4		HSTL_II	InOut				0	2	Data<10>		
#008080	same_bank	Data_Byte2	2	◇	U			NormalInthstl_x16	interface	1	V4,V5,S2,S2GX	C5		HSTL_II	InOut				0	2	Data<11>		
#008080	same_bank	Data_Byte2	2	◇	U			NormalInthstl_x16	interface	1	V4,V5,S2,S2GX	D1		HSTL_II	InOut				0	1	Data<12>		
#008080	same_bank	Data_Byte2	2	◇	U			NormalInthstl_x16	interface	1	V4,V5,S2,S2GX	D2		HSTL_II	InOut				0	1	Data<13>		
#008080	same_bank	Data_Byte2	2	◇	U			NormalInthstl_x16	interface	1	V4,V5,S2,S2GX	D3		HSTL_II	InOut				0	1	Data<14>		
#008080	same_bank	Data_Byte2	2	◇	U			NormalInthstl_x16	interface	1	V4,V5,S2,S2GX	D5		HSTL_II	InOut				0	1	Data<15>		

When the example csv file is imported in FSP the following are determined:

- A group with group name as *Data\_Byte1* is created with group constraint as *same\_bank* and all the pin properties highlighted with red color in figure above are imported in this group.
- A group with group name as *Data\_Byte2* is created with group constraint as same bank and all the pin properties highlighted with blue color in figure above are imported in this group.



All the pin properties are imported in single group if you do not specify the group properties in external file.

## Points to Remember While Importing CSV File

The following points you must remember while importing CSV file:

- The importing group properties are available in only Rules Editor and Rules Instance Editor.

- Group Numbers properties are not imported from external files, they are automatically assigned by FSP while importing.
- The following minimum headers are required:
  - Pin Name
  - Pin Number
  - IO Standard
  - Pin Type
- The Pin Number, X and Y locations information can be determined later. When Rules Editor dialog box appears you can update the Pin Number, X and Y location values using the Update Location option.
- Pin types are automatically converted to supported pin types. Following are the supported pin types you need to define in files:
  - Input
  - Output
  - InOut
- Invalid characters or invalid bus notations in the pin names are unsupported by FSP. See [Guidelines for Creating Parts](#) section for more information. You should correct the text data according to NMP check rules by using the entries provided in the fsp\_char.txt file, which is located at  
`<install_directory>\share\tools\fsp_char_support.txt`.
- A warning message is displayed if any duplicate pins exist in the text file.

### **Steps to create part from csv file**

To create a part from a .csv file:

1. To invoke Rules Editor:
    - Choose *Library – Create Part – Interface*.
- Or
- a. Click *Rules* tab in Libraries.
  - b. Click + to expand the tree view structure.
  - c. Right-click on a part name and click *Edit Component*.

# Allegro FPGA System Planner User Guide

## Creating Parts

The Rules Editor is displayed.

2. Click *Import From* at bottom side of the window.

A small pop-up menu is displayed.

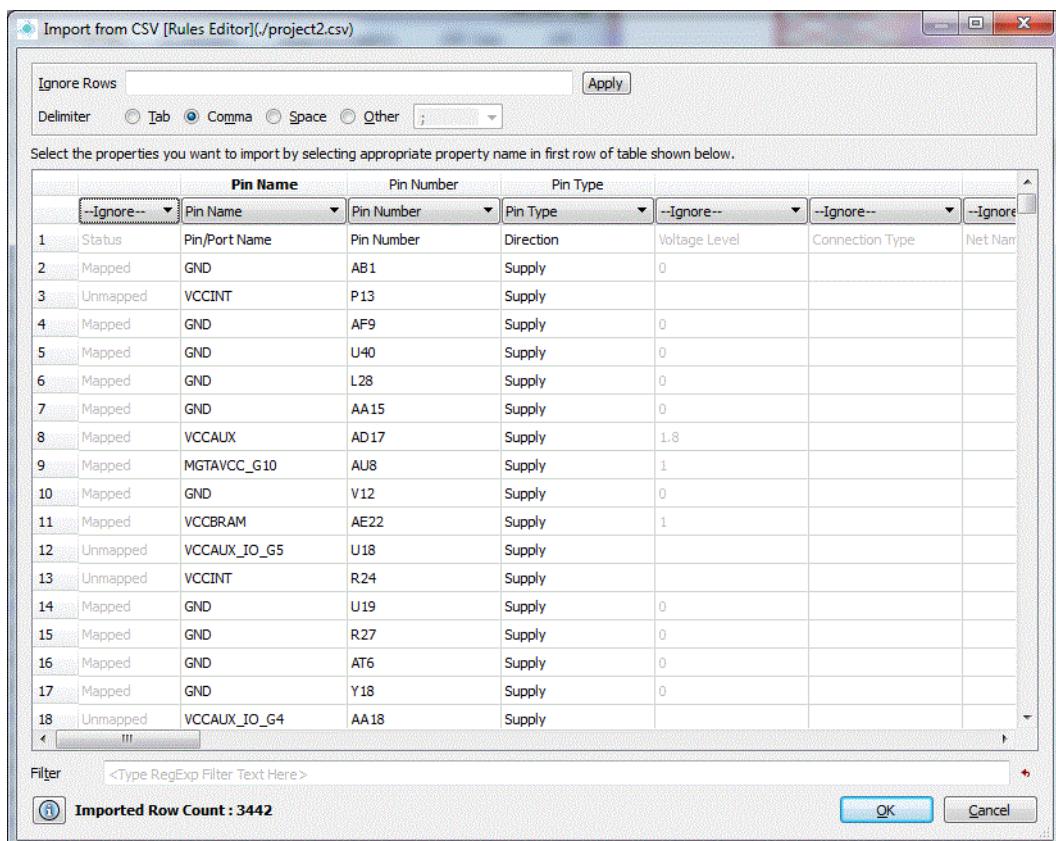
3. From the pop-up menu select *CSV*.

The *Select CSV File* dialog box is displayed.

4. Browse to the file, select the file, and click *Open*.

The Import CSV dialog box is displayed. The column names and values are automatically detected and displayed in this dialog box. You can change the column header based on your requirement.

**Note:** Selecting the column names is very crucial. Based on the selected column names, the pin information gets imported in Rules Editor.



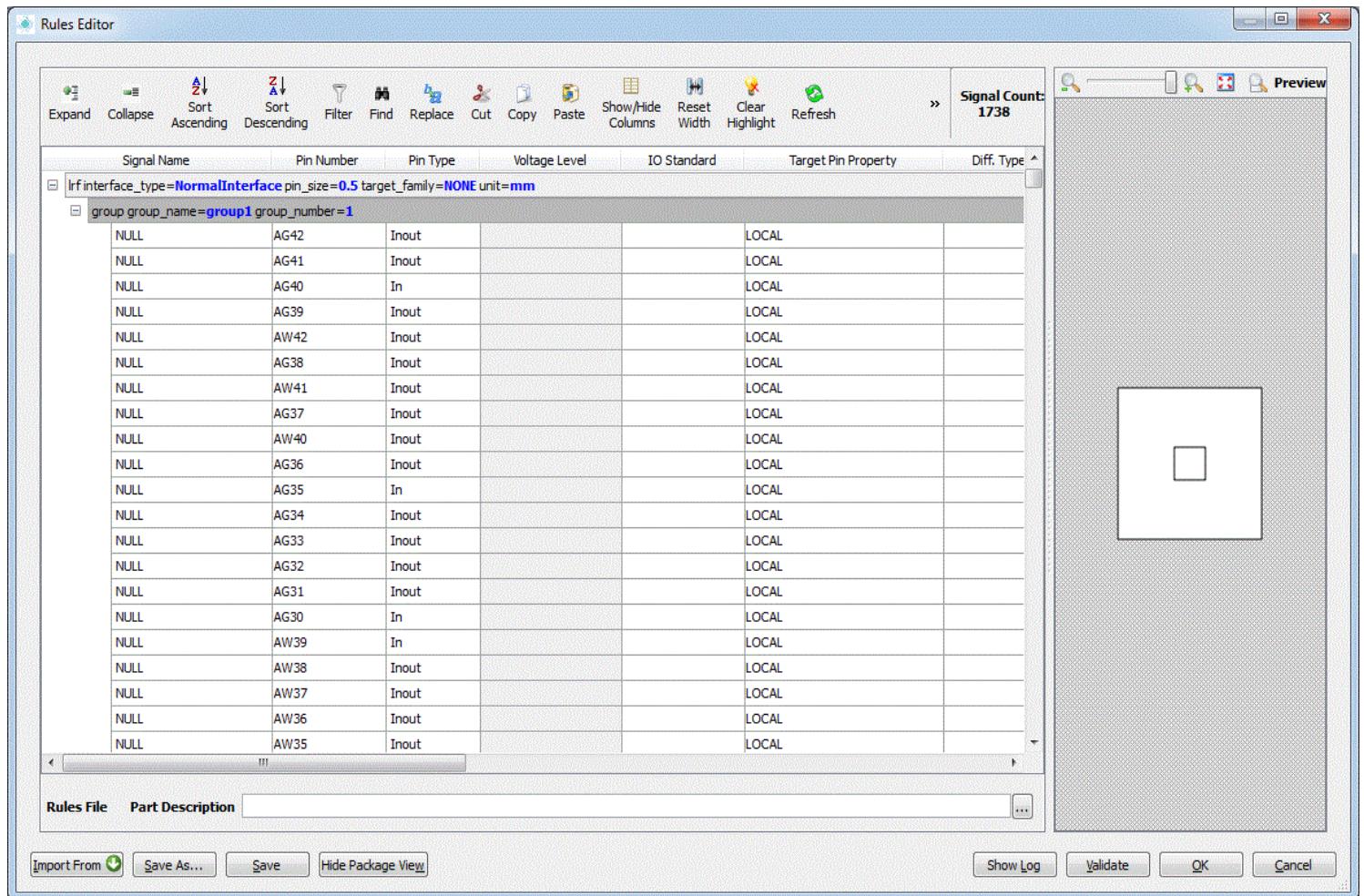
5. Specify the row numbers of the CSV file in the *Ignore Rows* field, whose values you do not want to import.

# Allegro FPGA System Planner User Guide

## Creating Parts

6. Select the delimiter option to import.
7. Click *OK* to import pin details in Rules Editor.

The extracted information is displayed in Rules Editor.



**Note:** Error or warning messages is displayed in log window for any incorrect information.

You can either save the model by specifying the rule file name with path in Rules File field and click *OK* or you can proceed further to edit the model.

8. Create groups, move pins into groups and modify the pin constraint as required.

For detailed information on editing options in rules editor, modifying group and pin constraints see [Creating Parts Manually](#) section.

9. Click *Validate* to verify the logical information.

10. Click *Save As* to save the pin informations.

**Note:** You may also use the *Save* option to save the part since you are creating the part from initial. Both *Save As* and *Save* options invokes the similar type of dialog box. Once you save the part, the *Save* option instantly save the part definitions in the file without opening the dialog box.

The Save Rules File dialog box is displayed.

11. Enter the name for the rules file in the *Rules File* field.
12. Click and select a directory path from the *Output Directory* drop-down list, where you want to save the rules file.
13. Click *OK* of the Save Rules File dialog box.
14. Click *OK* of the Rules Editor.

The part is saved at the specified location.

## **Creating Parts Manually**

Using Rules Editor, you can the create parts from scratch. The Rules Editor provides a spreadsheet view and various forms through which necessary logical information can be specified manually.

Creating parts process manually includes following major steps:

- Creating groups
- Adding pins
- Specifying necessary logical information required to complete the part such as pin type, IO standards, voltage level
- Verifying the specified logical constraints

After part creation is completed, the pins inside the group behaves according to the specified group constraints.

## **Invoking Rules Editor**

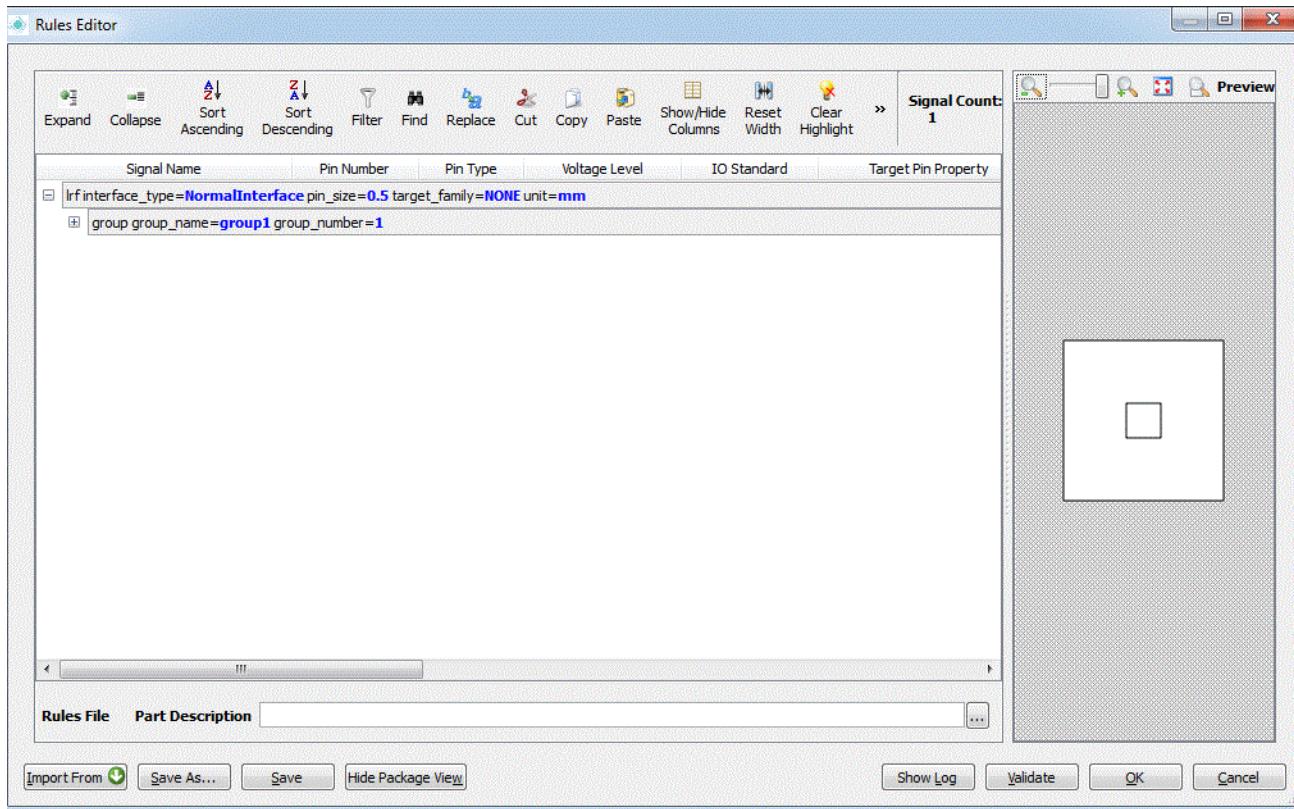
To invoke the Rules Editor:

1. Choose *Library – Create Part – Interface*.

The Rules Editor window is displayed. For more information on the fields and buttons of the Rules Editor dialog box see Rules Editor section.

# Allegro FPGA System Planner User Guide

## Creating Parts



Before you start adding pins in the editor, you need to specify the properties for the current part. However properties can also be specified at the end of the part creation.

### Adding Properties

Specifying properties is the most important step in part creation. This will decide the interface type for the model. For example, if you select the value for Interface type option as configuration in Edit Properties dialog box, then the model is used specifically for FPGA configuration. Properties are stored as part information and saved in the library file. These properties can be added or edited at any time during the part creation. Properties are added through Edit Properties dialog box.

To add or modify the properties perform the following steps:

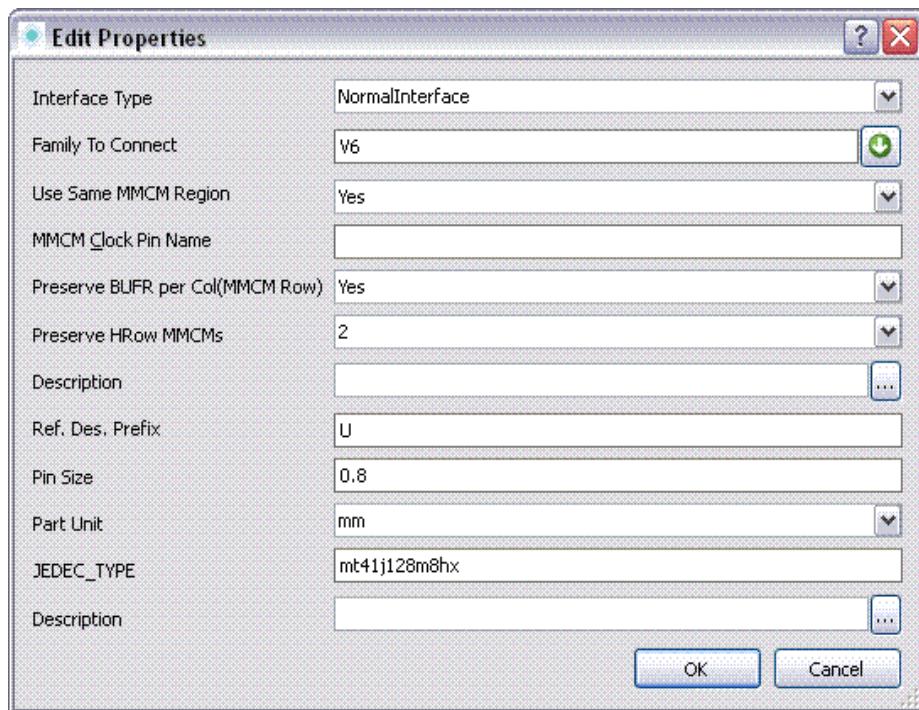
1. Do one of the following:

- Right-click on the first level node and select *Edit Properties* from pop-up menu.

Or

- ❑ Click *Edit Properties* icon.

The Edit Properties dialog box is displayed.



2. Specify the values in all fields as required.

For detailed information on Add Properties dialog box see Edit Properties section.

3. Click *OK*.

On clicking OK, the values are displayed in first-level node.

## Adding Signal

By default Rules Editor contains a dummy group. You can start adding signals to the default group and edit it group properties if required. Signals can be added in following ways:

- [Adding Scalar Signal](#)
- [Adding Bus Signals](#)

## Adding Scalar Signal

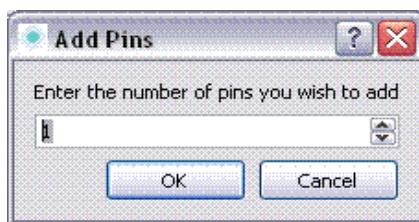
You can add signals to the group through Add Signal dialog box. These signals can be scalar signal or vector signal based on the name specified by you.

To add scalar signals to the group:

1. Do one of the following:

- ❑ Right-click on the second level node (group node) and select *Add Signal* option from pop-up menu.  
Or
- ❑ Click *Add Signal* icon.

The *Add Signal* dialog box is displayed.



2. Specify the number of pins you wish to add and click *OK*.

Now you can manually enter the signal names and pin numbers. For scalar signals you can specify as A, B, C, D and so on whereas for vector signals you can specify as following A<10....1>. To reduce the manual effort FSP provides you Add Bus for Group <group\_name> dialog box to add the bus signals at one step.

## Adding Bus Signals

Add Bus for Group <group\_name> dialog box allows you to add bus signals to the group. The dialog box also has some useful benefits over Add Pins dialog box:

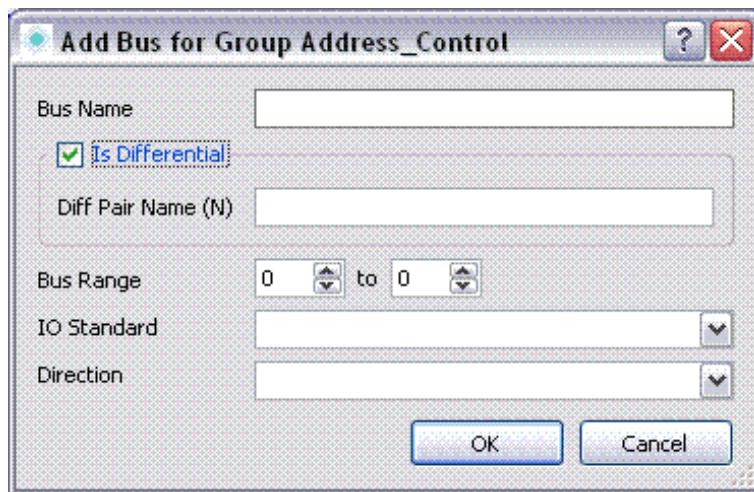
- Sizeable pins can be entered
- Vector pins can be entered at one step
- IO standards and pin type information can be entered for all the bus signals at one step
- Differential pair pins can also be entered

To add bus signals using Adding Bus Signals dialog box:

**1.** Do one of the following:

- Right-click on the second level node (group node) and select *Add Bus* option from pop-up menu.  
Or
- Click *Add Bus* icon.

The Add Bus for Group dialog box is displayed.



**2.** Specify the values in all the fields as required.

For detailed information on Add Bus for Group dialog box see [Add Bus for Groups](#) section.

**3.** Click *OK*.

The new signals are displayed with the names in Signal Name column. You can either now start specifying the pin details manually or create another group for your part.

## Adding Logical Group

Adding logical group is required to combine the signals into logical groups. Groups are created to logically arrange the pins. The group details you specify are useful at the time of I/O synthesis process.

To create a new logical group:

**1.** Do one of the following:

- Right-click on the first level node and select *Add Group* option from pop-up menu.

Or

- Click *Add Group* icon.

The *Edit Group* dialog box is displayed.



2. Specify the values in all the fields as required.

For detailed information on adding and editing group see [Edit Group](#) section.

3. Click *OK* of Edit Group dialog box.

Once the group is created you can now start adding pins to the group. For more information see [Adding Signal](#) section.

## Adding Pin Details

Once the group and logical pins are added to the spreadsheet editor, you can start specifying the pin details. Pin details need to be manually specified. There are few options available in the editor that are useful for organizing the pins and to make the editing process easier. For more information quick editing option see [Editing Options](#) and [Rearranging Pin Order](#) section.

To add pin details perform the following steps:

1. In Signal Name column, perform any one of the following:

- For scalar pins, specify the name as A, B, C, D.

- For vector pins, specify the name as A<10...1> or you also specify through Add Bus dialog box.
2. Specify the pin numbers in *Pin Number* column.
  3. Specify the pin type from the *Pin Type* drop-down list.
  4. Specify the voltage value for power pins in *Voltage Column*.
  5. Specify the IO standards from the *IO Standard* drop-down list.

**Note:** IO standard option is available in Add Bus dialog box if you have added signals through Add Bus dialog box.

6. Specify the target pin property values from the *Target Pin Property* drop-down list, if you wish to connect the group pins to specific FPGA pins.
7. Differential Type (Diff Type), Differential Pair Pin (Diff Pair Pin) and Serial IO TX/RX Pin column can be auto filled through *Auto Detect Pin Pair* dialog box. For more information see Auto Detecting Pin Pair section.
8. X and Y locations column field values can be autofilled by *Read Pin Locations* dialog box. For more information see *Updating Pin Locations* section.
9. Symbol Pin Name column field values are filled after mapping files.
10. Click on a single cell in *Description* column.  
A *browse (...)* button is displayed.
11. Click *browse (...)* and enter description for the pin and click *OK*.
12. Click on a single cell in *Custom Attribute* column and specify the attribute and its value and click *OK*.

## AutoMapping Diff Pair Pins

The Auto Detect Pin Pairs feature lets you automatically search and map the differential pair pins. This feature saves your time and removes the burden of manual searching of few differential pair pins among thousands of pins. Other than differential pair pins you can also search the serial IO TX/RX pins, Positive and Negative pins. Depending on the selected search criteria the pins are detected.

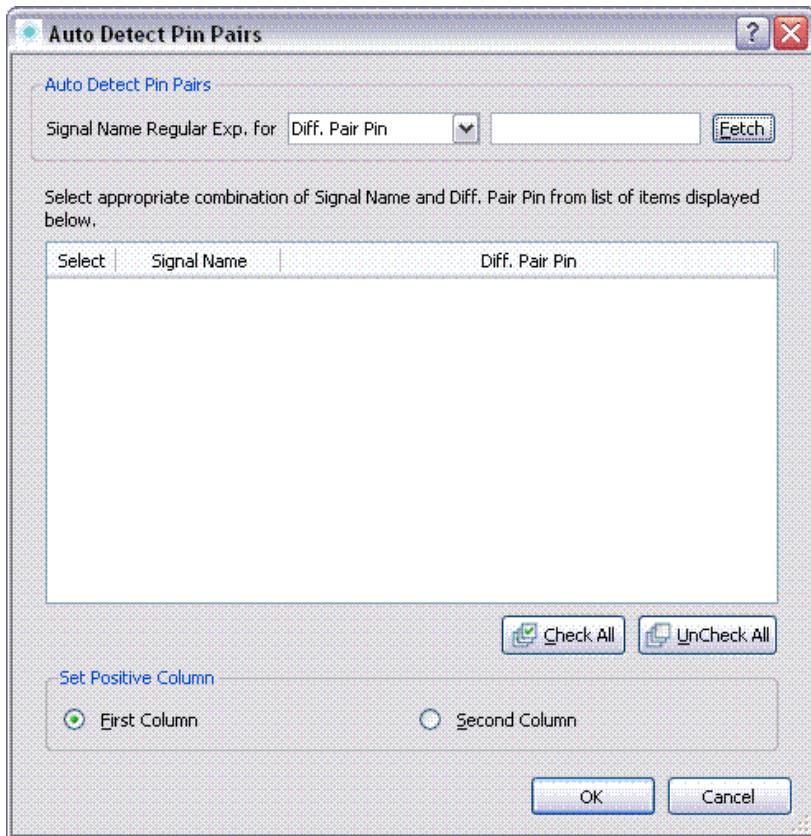
To auto detect pin pairs perform the following steps:

1. Right-click on the first level node and select *Auto Detect Pin Pair* option from pop-up menu or click *Auto Detect Pin Pair* icon.

## Allegro FPGA System Planner User Guide

### Creating Parts

The Auto Detect Pin Pairs dialog box is displayed.



2. In Signal Name Regular Exp. for field click and select the type of pair pins from drop down list for which you want to perform the search operation.
3. In Fetch text box specify the alphabet/word/character you are searching for in differential pair pins. Refer to the Auto Detecting Pin Pairs examples section.
4. Click *Fetch*.

The list of pin names and differential pair pins are displayed.

5. Click *Check All* to select all the pins.
6. In *Set Positive Column* field, select *First Column* to define all the P types pin as positive differential type or select *Second Column* to define all the N type pins as positive differential type.
7. Click *OK*.

After clicking OK, the selected pins are treated in accord with the option selected as Differential Pin Pairs, Positive, Negative, or Serial I/O TX/RX Pin.

### ***Auto Detect Pin Pairs Examples***

If you are searching the differential pin pair for the pins (click the drop-down and select Diff. Pair Pin from drop-down list)

- DDR2\_N[0] and DDR2\_P[0] the difference in the pin names is at only the middle point, i.e., N and P. Enter expression as PIN in the Fetch text box and click *Fetch*.
- PosData<0> and NegData<0>, the difference in the pin names is at only the first point, i.e., Pos and Neg. Enter expression as PosNeg in the Fetch text box and click *Fetch*.
- CLK and CLK\_N, the difference in the pin names is at only the end point, i.e., K and K\_N. Enter expression as KIK\_N in the Fetch text box and click *Fetch*.

If you are searching for the Serial I/O TXIRX pin, click the drop-down arrow and select Serial IO TX/RX Pin from the drop-down list.

- TXPRADA and RXPRADA, the difference in the pin names is at only the first point, i.e., TX and RX. Enter expression as TXIRX in the Fetch text box and click *Fetch*.
- Data\_T[0] and Data\_R[0], the difference in the pin names is at only the middle point, i.e., T and R. Enter expression as TIR in the Fetch text box and click *Fetch*.
- Data Transmit and Data Receiver, the difference in the pin names is at only the end point, i.e., Transmit and Receiver. Enter expression as Transmit and Receiver in the Fetch text box and click *Fetch*.

### **Adding Custom Attributes**

The custom attributes pane allows you to add the attributes to pins of the parts. After specifying the custom attributes save the part, the attributes get updated in the front-end symbol file of the instance.

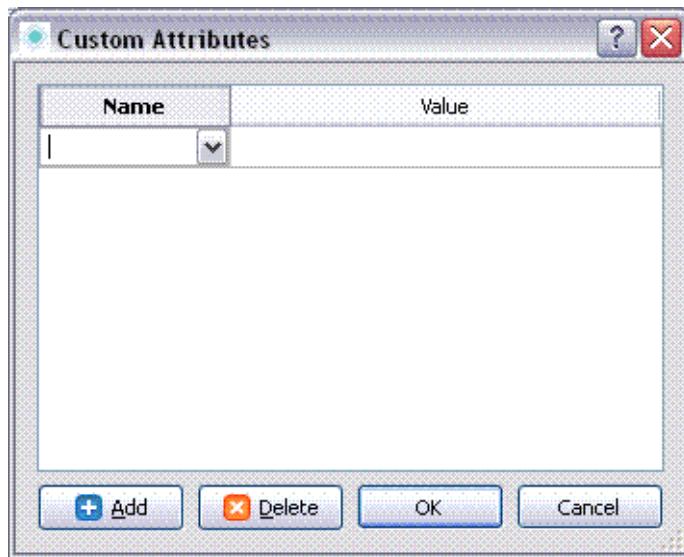
To add the custom attribute:

1. In Custom Attribute column do one of the following:
  - a. Double click on a cell.
  - b. Click *browse (...)*

Or

- Select the cell and press Ctrl + T to invoke Custom Attribute dialog box.

The Custom Attribute dialog box is displayed.



2. Click *Add*.
3. Specify a attribute name in *Name* column.
4. Specify a value for the attribute in *Value* column.
5. Click *OK*.

The attribute name and value is displayed in Custom Attribute column of the Rules Editor.

## Rearranging Pin Order

You can rearrange the pins in the Rules Editor according to your requirements and set the pin order. Pins can be moved across any groups and groups can be merged with any other group to change the schematic view of the component. You can logically arrange the pins by moving them across the groups. FSP allows you to move the individual pins as well as bus pins across group.

The following are the pop-up menu options available in Rules Editor, that will help you to quickly rearrange the pins or group.

## **Move Select Pins to Group**

To move the pins from one group to another group,

1. Select the pin(s) in a group that you wish to move.
2. Right-click on the selected pin(s) and select *Move Selected Pin(s) to Group* from pop-up menu.

The pins are moved to the selected group.

## **Move <Group\_Name> Bus to Group**

To move the bus pins from one group to another group:

1. Select the pin(s) in a group that you wish to move.
2. Right-click on the selected pin(s) and select *Move <Group\_Name>Bus to Group* from pop-up menu.

The bus pins are moved to the selected group.

## **Merging Splits**

You can merge the groups into another group to change the schematic view of the component.

To merge the splits:

1. Right-click on the group node and point to *Merge With Splits* option.  
The list of group names available in the current part is displayed.
2. Select any one of the group name.

The selected group merges with new selected group along with all pins and properties.

## **Editing Options**

This section describes some of the editing options that helpful for organizing the pins to make editing process easier.

## Select Pins In Current Section

To select all the pins in a group:

1. Select the pin (s) of any column.
2. Right-click on the selected pins and select *Select Pins In Current Section* from pop-up menu.

## Select Pin In Current Column

To select the pins in entire column:

1. Select the pin in group.
2. Right-click on the selected pins and select *Select Pins In Current Column* from pop-up menu.

## Remove Selected Pins

To remove the pins from group:

1. Select the pin (s) in *Signal Name* column.
2. Right-click on the selected pins and select *Remove Selected Pins* from pop-up menu.

**Note:** This option is visible only in Signal Name column.

## Rename Bus Pins

To rename the bus pins:

1. Select the pin(s) in *Signal Name* column.
2. Right-click on the selected pins and select *Renumber Bus Pins* from pop-up menu.

## Remove Bus Pins

To remove bus pins:

1. Select the pin(s) in *Signal Name* column.
2. Right-click on the selected pins and select *Remove Bus Pins* from pop-up menu.

## **Clearing Selected Cell Contents**

To delete the contents of a cell:

1. Select a pin.
2. Right-click on selected pin and select *Clear Selected Cell Contents* from pop-up menu.

**Note:** This option is not applicable for read only cells.

## **Apply To Bus <Bus\_Name>**

To apply a single change to entire bus:

1. Specify a IO standard from the *IO standard* drop down list.
2. Right-click on the pin and select *Apply To Bus '<bus\_name>'* from pop-up menu.

**Note:** This option is available for only few columns IO Standard, Target Pin Property, Differential Type (Diff.Type), Clock Group, Pin Type, Pin Direction and Custom Attribute.

## **Apply To All Rows**

To apply a single change to entire rows:

1. Specify custom attribute for a single pin.
2. Right-click on the pin and select *Apply To All Rows* from pop-up menu.

**Note:** This option is available for only few columns IO Standard, Target Pin Property, Differential Type (Diff.Type) ,Clock Group, Pin Type, Pin Direction and Custom Attribute.

## **Verifying the Part**

Once the necessary constraints are set, you need to validate the pin informations defined in the Rules Editor. Use *Validate* option to verify the part. A new pane is displayed at the bottom side of the window with messages. A successful report is displayed for correct pin informations or error report is displayed for incomplete or incorrect information.

## **Saving the Part**

Part can be saved only in the lrf working directory and in any lrf directory.

To save the part:

1. Click *Save As* to save the pin informations.

**Note:** You may also use the *Save* option to save the part since you are creating the part from initial. Both *Save As* and *Save* options invokes the similar type of dialog box. Once you save the part, the *Save* option instantly save the part definitions in the file without opening the dialog box.

The Save Rules File dialog box is displayed.

2. Enter the name for the rules file in the *Rules File* field.
3. Click and select a directory path from the *Output Directory* drop-down list, where you want to save the rules file.
4. Click *OK* of the Save Rules File dialog box.
5. Click *OK* of the Rules Editor.

After clicking OK the directory is saved at the specified path.

## Editing Parts

Using Rules Editor you can edit the constraints of a part. The components are modified for various reasons, such as change in the logic design requirements during course of the project. This could be because of update in the logic itself or for time driven optimization needs. Due to these changes the component logic and layout need to be changed multiple times during the course of a project. These modification includes setting the differential pair pin type, opposite pair pin type, target pin type (CC, GCC etc.), pin function (data strobe), and group-level constraints (such as use of the same bank to connect a group of pins) so on. Parts that have been instantiated on canvas can also be edited. Editing parts from the canvas is not recommended since this deletes the connectivity definitions and leads to error prone state.

You can do the following to edit the parts:

- [Editing the Logical Properties](#)
- [Editing the Group Constraints](#)
- [Editing the Pin Constraints](#)

### Editing the Logical Properties

To edit the properties of the part, use the Edit Properties dialog box. You can modify the values in any fields that are available in Edit Properties dialog box.

To edit the logical properties:

1. Right-click on the first level node and select *Edit Properties*.

The Edit Properties dialog box is displayed. For detailed information on Edit Properties dialog box see Edit Properties section.

2. Edit the parameters as required and click *OK*.

**Note:** In *Family To Connect* field, if you reselect a different device name then all the group constraints and pin constraints need to revised based on the targeted device family.

### Editing the Group Constraints

To modify the group constraints for parts use Edit Group dialog box. The type of modification that can be made to group constraint are as follows:

- [Adding Group](#)
- [Renaming Group](#)

- [Deleting Group](#)
- [Reselecting Group Constraint and Data Group Nodes](#)
- [Reselecting Group Color](#)
- [Editing Description](#)

## **Adding Group**

To add a group:

1. Do any of the following:
  - ❑ Right-click on group level node and select *Add Group*.
  - Or
  - ❑ Click *Add Group* icon.
- The *Add Group* dialog box is displayed.
2. Modify the fields value as required and click *OK*.

For more information on Adding Group see Adding Group section.

## **Renaming Group**

To rename a group:

1. Right-click on the group level node and select *Edit Group*.  
The Edit Group dialog box is displayed.
2. Click on the *Group Name* field and change the pin name.
3. Click *OK*.

## **Deleting Group**

To delete a group:

1. Right-click on the group level node and select *Remove Group* from pop-up menu.  
A small confirmation window is displayed.
2. Click *Yes* to remove the group.

## **Reselecting Group Constraint and Data Group Nodes**

To reselect group constraint and data group nodes:

1. Right-click on the group level node and select *Edit Group*.  
The Edit Group dialog box is displayed.
2. Select a different group constraint name from *Group Constraint* drop down list.
3. Select a different data group mode name from *Data Group Node* drop down list.
4. Click *OK*.

**Note:** Pin constraints also need to be modified based on the group constraint selection.

## **Reselecting Group Color**

To reselect group color:

1. Right-click on the group level node and select *Edit Group*.  
The Edit Group dialog box is displayed.
2. Click group color and select a different color from *Select Color* dialog box.
3. Click *OK* of Select Group dialog box.
4. Click *OK*.

## **Editing Description**

To edit the description:

1. Right-click on the group level node and select *Edit Group*.  
The Edit Group dialog box is displayed.
2. Click *browse (...)* of Description field.  
The Description dialog box is displayed.
3. Edit the content as per required and click *OK*.
4. Click *OK*.

## **Editing the Pin Constraints**

The type of modification can be made to the pin constraints are as follows:

- [Adding Signal](#)
- [Adding Bus](#)
- [Renaming Signal](#)
- [Renaming Bus Signals](#)
- [Renaming Pin Numbers](#)
- [Auto generating Pin Numbers](#)
- [Deleting Pins](#)
- [Moving Pins](#)
- [Other Pin Constraint Modifications](#)
- [Updating Pin Locations from DRA File](#)
- [Editing Custom Attributes](#)
- [Editing Description](#)

### **Adding Signal**

For more information on adding signal see Adding Signal section.

### **Adding Bus**

For more information on adding bus see Adding Bus section.

### **Renaming Signal**

To rename a single signal:

1. Select a signal in *Signal Name* column.
2. Change the signal name.

## **Renaming Bus Signals**

To rename bus signals:

1. Right-click on a signal in *Signal Name* column.
2. Select *Rename Bus <Signal\_Name>* option from pop-up menu.  
The Rename Bus dialog box is displayed.
3. Change the bus name.
4. Click *OK*.

## **Renaming Pin Numbers**

To rename pin numbers:

1. Select a cell in *Pin Number* column.
2. Change the pin number.

## **Auto generating Pin Numbers**

To autogenerate the pin numbers:

1. Right-click on a cell in *Pin Number* column.
2. Select *Auto Generate Pin Numbers* option from pop-up menu.  
A confirmation dialog box is displayed.
3. Click *Yes* to autogenerate pin numbers for one pin.
4. Click *Yes to All* to autogenerate pin numbers for all the pins.

## **Deleting Pins**

To delete a instance pin:

1. Right-click on a cell in *Signal Name* column.
2. Select *Remove Selected Pins* option from pop-up menu.
3. A small confirmation dialog box is displayed.
4. Click *Yes* to delete the pin(s).

**Note:** To select multiple pins for deletion, click and drag on the pins. Then, right-click and select *Remove Selected Pins*.

The pin is deleted from the editor.

### Moving Pins

You can move either a single pin or complete bus in the group to another group. To move a bus to another group you can either use the pop-up option or merge the group with another group.

For more information on moving pins and bus to another group see [Editing Options](#) section.

### Other Pin Constraint Modifications

The following pins can be modified at any stage of part creation:

- IO Standards
- Pin Types
- Target Pin Property
- Differential Type
- Differential Pair Pin
- Serial IO TX/RX Pin

To edit the constraint above:

1. Select the pin.
2. Edit the constraints in respective column as required.
3. Click *Validate* to check whether the entered values are appropriate.

**Note:** For more information on Differential Type, Differential Pair Pin or Serial IO TX/RX pin modification see Auto detecting Pin Pair section.

### Updating Pin Locations from DRA File

X and Y locations column values decides the layout view of the model. The values in this two columns are typically populated by reading the footprint values from dra file. If the footprint of the model changes, the layout view (X and Y location) of the model can be updated through

Update Pin Locations From PCB Footprint dialog box. After importing the dra file, the X and Y column values get updated. You can also see the symbol graphics view of the component at right side of the pane.

To update the X and Y locations perform the following steps:

1. Click *Update Location* option.

The Update Pin Locations From PCB Footprint log box is displayed.

2. Select a dra name from the list.
3. Click *OK*.

### **Editing Custom Attributes**

The custom attributes can either be renamed or deleted.

To edit the custom attribute:

1. Click on a cell in *Custom Attribute* column.

A small button is displayed.

2. Click *browse (...)*.

The Custom Attribute dialog box is displayed with predefined values.

3. Select a entry and click *Delete* to delete the entry.
4. Click and edit the values both in *Name* and *Value* column and click *OK*.
5. Click *OK*.

### **Editing Description**

The description of the pin can either be removed or modified.

To edit the description:

1. Click on a cell in *Description* column.

A small button is displayed.

2. Click *browse (...)*.

The Description dialog box is displayed with some text.

3. Edit the text as required.
4. Click *OK* of Description dialog box.
5. Click *OK*.

## Updating Pin and Group Constraints from External File



Updating group properties through *Update from CSV* dialog box is available only in Rules Editor and Rules Instance Editor. In other editors this feature is used to import only pin properties.

FSP enables you to modify the pin and group details in Rules Editor at any time during the part creation process. You use *Update from CSV* option to update the logical details present in the Rules Editor with the logical information stored in the CSV file. In this process you can update any column values except *Signal Name* and *Pin Number* columns. The *Signal Name* and *Pin Number* column values are considered as reference column. You have the choice to select any one of them as reference column. Selecting reference column means while importing CSV file into Rules Editor, the selected column values both in Rules Editor and CSV file should match while importing. For example, if you choose *Signal Name* column as reference column then signal names in Signal Name column both in Rules Editor and CSV file should match. Any mismatch may halt the process or may update the pin details incorrectly.

The Update from CSV dialog box helps you to update both pin and group details in Rules Editor. After importing CSV file the pin details are extracted and displayed at bottom of the Update from CSV dialog box in spreadsheet editor view. Based on the available column values in CSV file the columns in Update from CSV dialog box are filled. The remaining columns are displayed in disabled mode.

The process of updating the pin and group details from external file is very much similar to Import from CSV feature. For detailed information see [Creating Parts from External Files](#) section.

### Points to Remember while Updating from CSV

The following points you must remember while updating the pin and group details from CSV:

- Update from CSV feature is available only in Rules Editor and Rules Instance Editor.
- Updating Group Numbers property is not supported. You must specify the group numbers manually after importing.

- In Rules Editor, pins are shuffled across the groups as per defined in the CSV files.  
For example in Rules Editor *group1* and *group2* has ten pins each and CSV file contains *group1* eight pins and group twelve pins. When you import the CSV file two pins are moved from *group1* to *group2*.
- At certain case a new group is automatically created in Rules Editor. Group number for the new group is not taken from CSV file and will be randomly assigned by FSP.

**Case**

When CSV file contains extra group compare to groups in Rules Editor.

- While importing CSV file empty groups are removed from Rules Editor.

To update pin and group details from the CSV file perform the following steps:

1. In Rules Editor, choose *Create From – CSV*.

A confirmation window is displayed prompting you about the over-riding of existing pin details.

2. Click *Yes*.

The *Select CSV File* dialog box is displayed.

3. Browse to the file, select the file, and click *Open*.

The *Import from CSV* dialog box is displayed.

4. You can specify the rows number in the *Ignore Rows* field that you do not want to import.

For example 1-10, 24,25.

5. Specify the column names whose values you want to update.

6. Select the format type of the text file.

7. Click *OK* to import pin details.

After you click *OK*, all the columns in Rules Editor are updated with the text file information. You can further manually update the details if required.

---

## **Creating Virtual Interfaces**

---

This chapter contains the following sections that describe the procedures for creating single and multiple virtual interfaces in FPGA System Planner.

- [Overview to Virtual Interface](#)
- [Creating Single Virtual Interface](#)
- [Creating Multiple Virtual Interfaces \(Using Verilog/VHDL file\)](#)
- [Editing Options](#)

## Overview to Virtual Interface

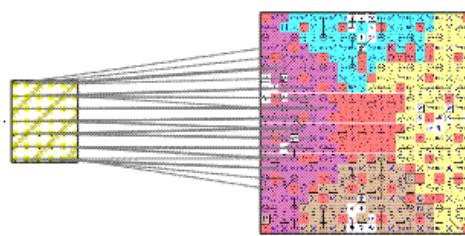
The FSP generated schematics are encapsulated inside a hierarchical block and all the interfaces which have been identified for this block are available on the hierarchical block symbol. These hierarchical block ports can be used to interface with components outside the FSP canvas.

FSP provides support for Virtual Interface, which are created on the FSP canvas as place holders for real component interfaces. A virtual interface becomes an interface to the FSP hierarchical block. Therefore, the ports on the FSP hierarchical block symbol can be used to connect to a real component in the schematic.

Note that interfacing the hierarchical block ports with external components outside the FSP canvas can also be achieved through Virtual Interface External Ports option. Virtual Interface External ports allows you to mark each net individually and make these nets as hierarchical block ports in schematic. But marking thousands of pins as external ports is cumbersome. To address this issue Virtual Interface has been introduced.

A dummy interface model (virtual interface) can be created using various methods. The process of creating the virtual interface is similar to creating the parts using different files of format. You can modify the virtual interface at any time during the course of project. Once you finish the part creation, a dummy interface is placed on FSP canvas. You can use the process option to generate nets between the virtual interface and FPGA.

The figure below displays an overview of connection between virtual interface and FPGA.



The following points you must remember while working with virtual interface:

- The virtual interface parts are not cached in project directory.
- The virtual interface model is not displayed in DE-HDL schematic. Only the nets are captured in hierarchical block.
- Virtual interface can neither be saved in FSP library nor can be used as normal interface. In order to use it as normal interface you must convert it to real component.

## **Allegro FPGA System Planner User Guide**

### Creating Virtual Interfaces

---

- The virtual interface is also supported for JTAG chain connections. For JTAG chain connections you need to select the available JTAG pin properties in Target Pin Property column in *Define Virtual Interface for Device Instance* editor.

## Creating Single Virtual Interface

FPGA System Planner lets you to create the virtual interface manually or using different methods. You can import the pin information stored in different files of format and create virtual interface from it. After you have instantiated the virtual interface in your design, if required you can modify the virtual interface at any time during the course of project.

The process of creating virtual interface is similar to creating parts. This section describes the process of creating a virtual interface using existing virtual interface file and constraints pinout file. For more information on creating virtual interface from existing lrf see the Creating Parts from Existing LRF section.

Virtual Interface can be created using following options:

- [Using Existing LRF](#)
- [Using Constraints/Pinout File](#)

### Using Existing LRF

You can create the virtual interface by importing the pin information from existing lrf file. For more information see Create Parts from Existing LRF section.

### Using Constraints/Pinout File

FSP lets you to use the verilog files and constraint/pinout files generated using tools from FPGA vendor Xilinx, Altera and Actel to create virtual interface. It is not necessary to have constraint files while creating virtual interface. You can define the pin constraints (IO standards) after importing the verilog file in Edit Virtual Interface editor. You are also facilitated with an option to select the signals from the files you want to use in creating virtual interface.

### Conversion Details

- The following constraint values are read when you import only HDL file:
  - Signal Name
  - Pin Type
- The following constraint values are read when you import only constraint file:
  - Signal Name
  - IO Standard

- On Chip Termination (If specified)
  - Pin Number
  - Diff Type (If specified)
  - Diff.Pair Pin (If specified)
  - Serial IO TX/RX Pin (If specified)
- When you import both HDL file and constraint file, the port names are matched in the both the file.
- If port names are unequal, the signal names are read from constraint file and associated pin types are read from HDL type.
- Example, if a, b are the signal names present in HDL file and d, e are the signal names in constraint file. Since both are unequal, while importing d, e are read from constraint file as signal names and pin types of a, b signal names are read from HDL file.
- If port names are equal, the matched port names will be read as signal names and pin types will be read from HDL file.
- The IO standards will be ignored while importing, if the IO standard in constraint files is not supported by the device for whose virtual interface you are creating. You must select a different standard from IO Standard drop down list in Define Virtual Interface for Device Instance <*Instance Name*> dialog box.

For example, HSTL\_II IO standard is unsupported by Spartan-3 device. While importing the constraint files, the HSTL\_II is ignored. You need to select a different standard supported by Spartan-3 device.

### ***Steps to create virtual interface using constraints/pinouts file***

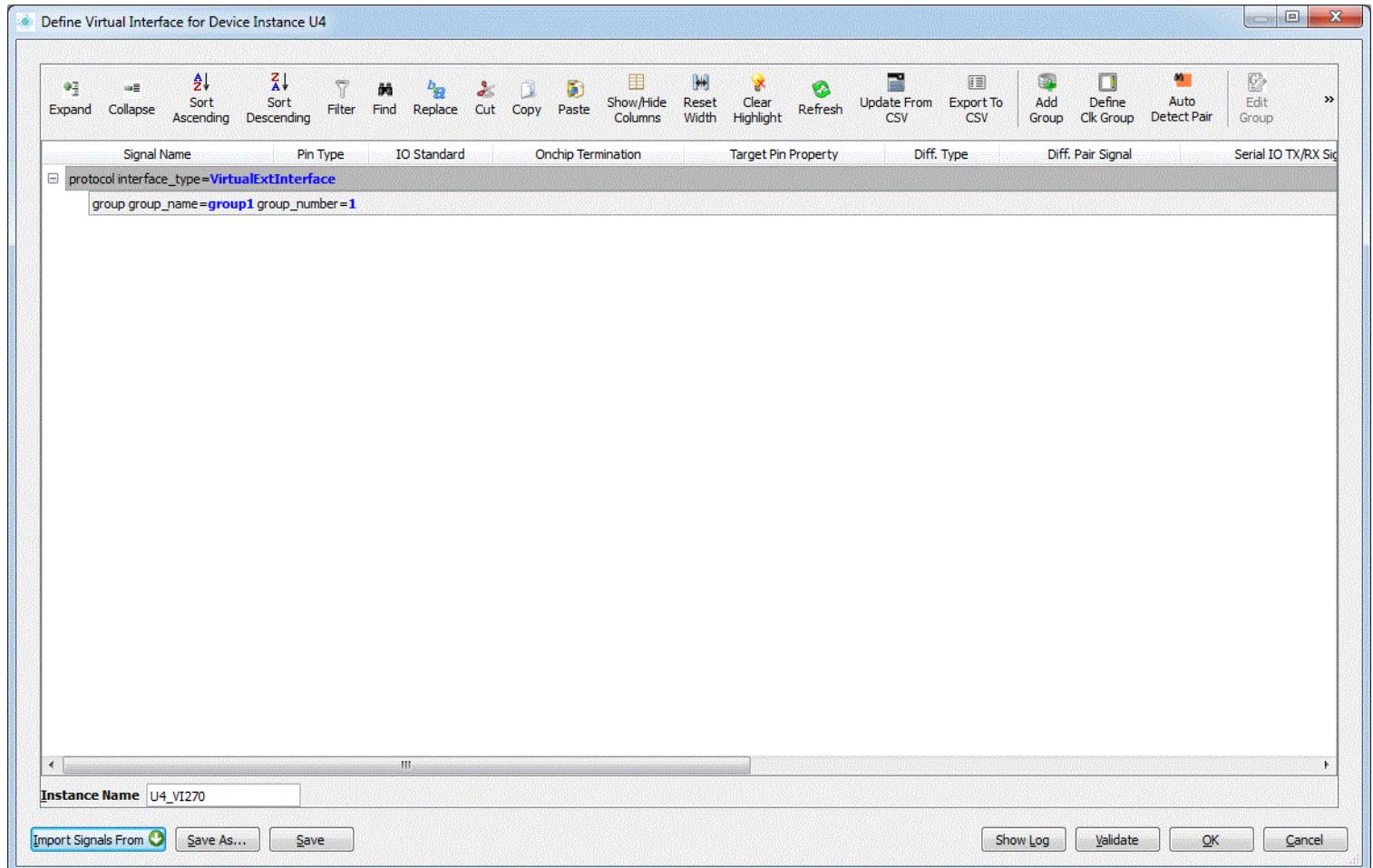
To create an virtual interface using constraints/pinouts file:

1. Place a FPGA on canvas.
2. Right-click on FPGA instance and choose *Virtual Interface - Create New Virtual Interface*.

The Define Virtual Interface for Device Instance <*Instance Name*> dialog box is displayed.

# Allegro FPGA System Planner User Guide

## Creating Virtual Interfaces



### 3. Click *Import Signals From*.

A small pop-up menu is displayed.

### 4. Click *Constraints/Pinout File*.

A confirmation windows pops up prompting you about the overriding of existing definition.

### 5. Click Yes to continue.

The *Import Constraint/Pinouts Signals for Device/Instance <Instance Name>* dialog box is displayed.

### 6. Specify the constraints/pinouts file name and path in *Constraints/Pinouts File* field or click ... to browse to the location where the file exists.

### 7. Click *Check All* to select all the signals or you can manually select the check boxes under *Import* column.

## Allegro FPGA System Planner User Guide

### Creating Virtual Interfaces

---

8. Click *Load Signals* to import all the signals.

The pin informations (Signal Name, Pin Type, IO standard, Diff Type, Diff.Pair Signal, Reserve Type, DCI or OCT, Net Name, FPGA Port and Assigned to Pin details) are displayed in *Define Virtual Interface For Device Instance <Instance Name>* dialog box. You can proceed with further modifying the pin details if required.

9. Click *Validate* to verify the pin details.
10. Click *Save As* to save the part in other directory.
11. Specify the name for the virtual interface in the *Instance Name* text box.
12. Click *OK* to save the part.

After clicking OK the virtual interface is displayed on canvas.

## **Creating Multiple Virtual Interfaces (Using Verilog/VHDL file)**

FSP lets you to create multiple virtual interfaces for device by using single HDL file. Creating multiple virtual interface depends on the number of ports available in HDL file. Using each set of signals from HDL files you create the virtual interfaces as per your requirement. The process of creating multiple virtual interfaces is similar to the creating single virtual interface. The key limitation behind the creating single virtual interface is you must remember the signal names that you have not used while creating the first virtual interface. For example, you create a first virtual interface using ten port names from HDL files. You open the same dialog box and import the same HDL file. Now you must be careful while selecting the port pins to create interface. You must be able to differentiate between the port names which has already been used and not used while creating first virtual interface and port names which is cumbersome. To avoid this situation FSP provides you a solution through which you can create multiple virtual interface using single HDL file in one go.

The following procedure describes the process of creating two virtual interfaces using single HDL file:

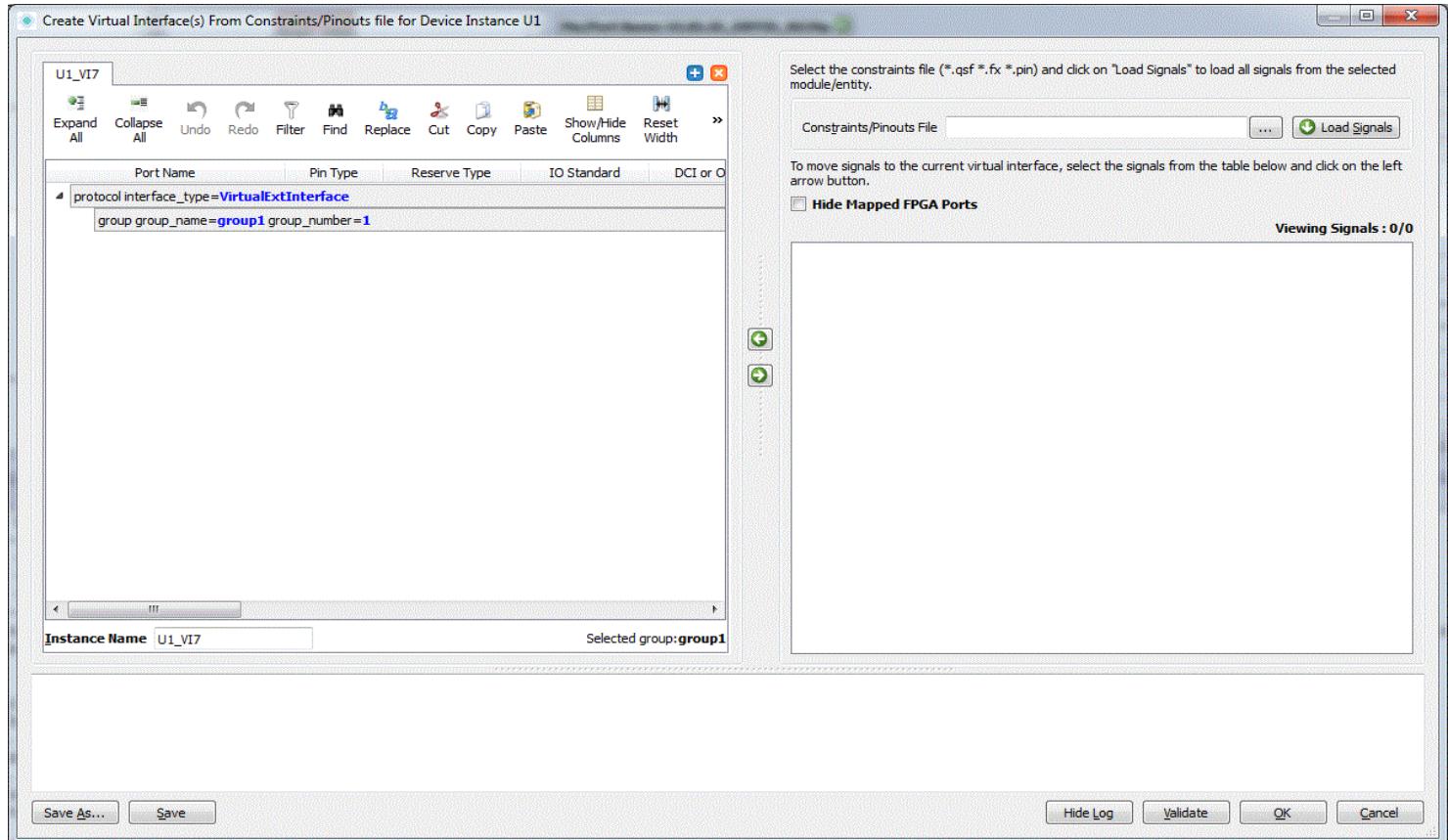
### **Creating First Virtual Interface**

1. Place a FPGA on canvas.
2. Right-click on FPGA instance and choose *Virtual Interface - Create New Virtual Interface From HDL*.

The Create Virtual Interface From Verilog/VHDL for Device Instance <Instance Name> dialog box is displayed.

# Allegro FPGA System Planner User Guide

## Creating Virtual Interfaces



At left side of the pane:

3. Specify the constraints/pinouts file name and path in *Constraints/Pinouts File* field or click *browse (...)* to browse to the location where the file exists.
4. Click *Load Signals* to display the signals at bottom side of the window.

The pin informations (Net Name, IO standard, Diff Type, Reserve Type, DCI or OCT, Diff.Pair Signal, Net Name, FPGA Port and Assigned to Pin details) are displayed in Define Virtual Interface For Device Instance <Instance Name> dialog box. You can proceed with further modifying the pin details if required.

5. Select and click < icon to move the signals from right side pane to left side pane and drop it.

You can further manually specify the group details and pin details if required. For more information see [Editing Parts](#) section.

6. Reenter the name in Instance Name field if you wish to modify the virtual interface name.

## Creating Second Virtual Interface

1. Click + to create a new tab in the dialog box.

A separate tab browser is created with name <instance\_name>\_<VI2>.

2. Click <instance\_name>\_<VI2>.

3. Select and click < icon to move the remaining signals from right side pane to left side pane and drop it.

You can further manually modify the pin details if required.

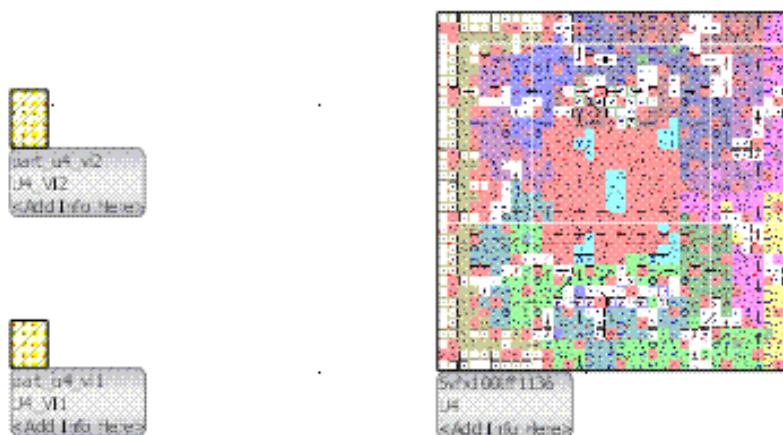
4. Reenter the name in Instance Name field if you wish to modify the virtual interface name.

**Note:** You can create multiple virtual interfaces with a single HDL file.

5. Click *Validate* to verify the pin details in the current browser.

6. Click *OK* to complete the process.

After clicking OK two virtual interfaces displays on canvas. See below.



## Editing Options

FSP provides you few miscellaneous editing options using which you can quickly perform tasks. To access this option right-click on the virtual interface in canvas. The following table describes the miscellaneous editing options which are available only for Virtual Interface on canvas. For more information on other editing options see [Editing Parts](#) section.

## **Allegro FPGA System Planner User Guide**

### Creating Virtual Interfaces

---

<b>Name</b>	<b>Functionality</b>
Edit	Lets you to open the Edit Virtual Interface < <i>Virtual Interface Name</i> > for Device Instance dialog box.
Convert Virtual Interface To Real Interface	Select this option to invoke the Component Browser. For more information on converting virtual interface to real interface see <a href="#"><u>Converting Virtual Interface to Real Components</u></a> section.
Delete	Select this option to delete the virtual interface from canvas.

# **Allegro FPGA System Planner User Guide**

## Creating Virtual Interfaces

---

---

## **Creating Connectors**

---

This chapter contains the following sections that describe the procedures for creating connectors in FPGA System Planner.

- [Types of Connectors](#)
- [Creating Connector](#)
- [Creating Testing Connector](#)

## Types of Connectors

Connectors, due to the programmable nature of their pin usage and having modeling characteristics similar to FPGA's FSP provides functionality to create models for connectors.

In FPGA System Planner, connectors are classified into types:

- Connector
- Testing Connector

Both the types are physical and electrical characteristics point of view are similar except logical characteristics.

---

### Connector

This is a definable connector and different from a DDR type connector, where the component that is mounted on the PCB is actually a connector but the pins are not user definable. This type of non connector has fixed pin/signal definitions, I/O standards, voltage pins, etc.

### Testing Connector

This type of connectors are primarily used for board to board connections.



The model definition of these connectors decides the optimization.

## Creating Connector

In general, connector type contains two types of banks i.e Normal IO and Power Banks. You may create the connector manually or import the pin information from external files of different files of format. The connector model is saved in .lrf file format and can be modified at any time during the design. However the connectors must not have connections when you are editing otherwise it may lead to error prone stage.

The process of creating connector model is very much similar to creating interface models process. The following are some of the dissimilarities you may notice while creating connector model compare to creating interface models:

- Groups are called as Banks
- *Target Pin Property* and *Family to Connect* options are not available. Since connectors are considered as FPGA's.

This section describes the process of creating connector model manually. For more information on how to create connector model using external file see [Creating Parts from External Files](#) section.

### Creating Connector Manually

Using Rules Editor, you can the create connector from scratch. The Rules Editor provides a spreadsheet view and various forms through which necessary logical information can be specified manually.

The following are the major steps involved in part creation:

- Creating banks
- Adding pins
- Specifying necessary logical information required to complete the part such as pin type, IO standards, voltage level
- Verifying the specified logical constraints

After model creation is completed, the pins inside the banks behaves according to the specified group constraints.

### Invoking Rules Editor

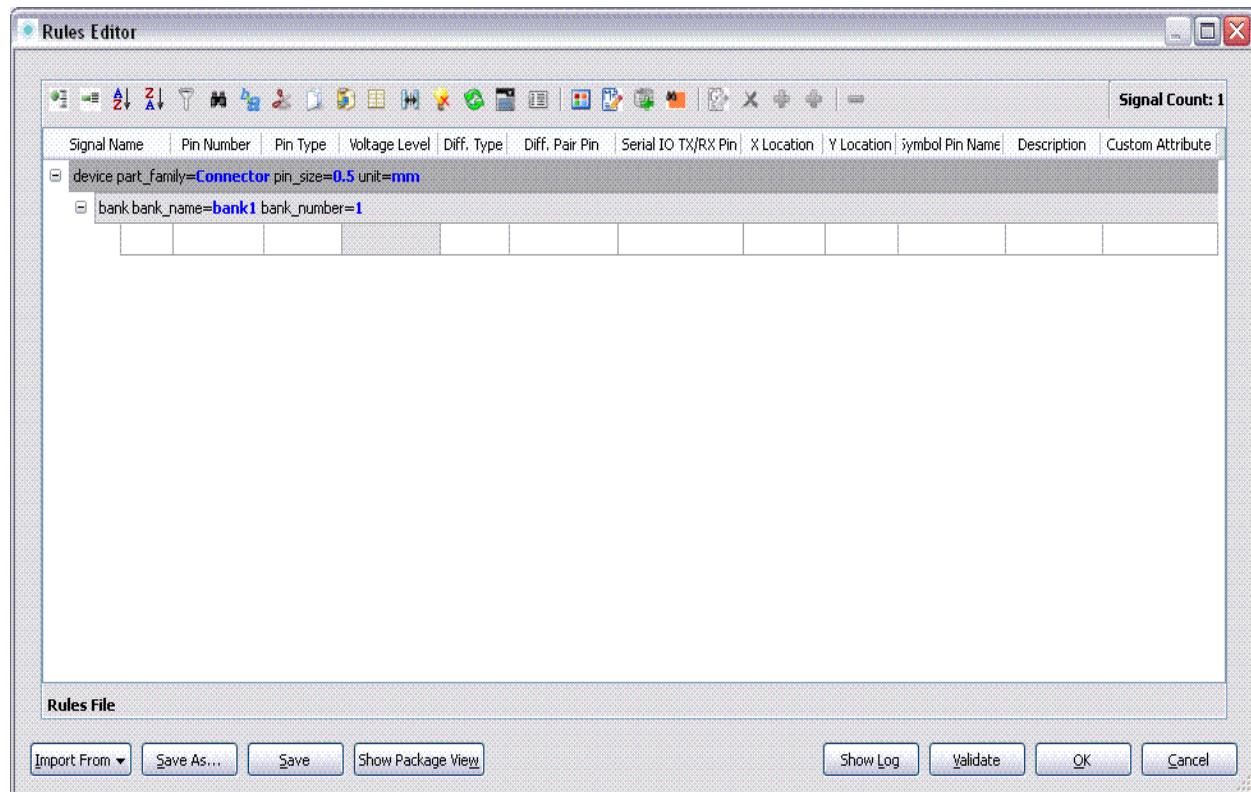
To invoke the Rules Editor:

# Allegro FPGA System Planner User Guide

## Creating Connectors

### 1. Choose Library – Create Part – Connector.

The Rules Editor window is displayed. For more information on the fields and buttons of the Rules Editor dialog box see Rules Editor section.



Before you start adding pins in the editor, you need to specify the properties for the current model. However properties can also be specified at the end of the part creation.

## Adding Properties

Specifying properties is the most important step in model creation. Properties are stored as model information and saved in the rules file. These properties can be added or edited at any time during the model creation. You use Edit Properties dialog box to specify properties.

To add or modify the properties perform the following steps:

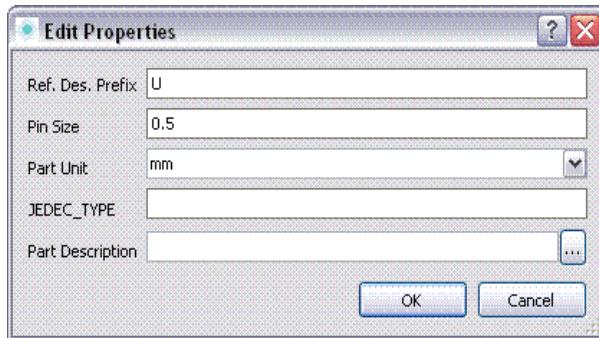
### 1. Do one of the following:

- Right-click on the first level node and select *Edit Properties* from pop-up menu.

Or

- Click *Edit Properties* icon.

The Edit Properties dialog box is displayed.



2. Specify the values in all fields as required.

3. Click *OK*.

On clicking *OK*, the values are displayed in first-level node.

## Adding Signal

By default Rules Editor contains a dummy bank. You can start adding signals to the default bank and edit it group properties if required. Signals can be added in following ways:

- [Adding Scalar Signal](#)
- [Adding Bus Signals](#)

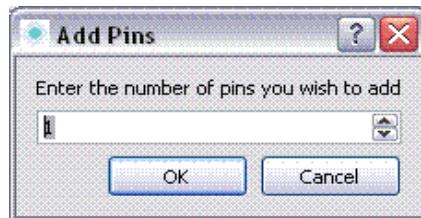
### Adding Scalar Signal

You use Add Signal dialog box to add signals to the bank. These signals can be scalar signal or vector signal based on the name specified by you.

To add scalar signals to the bank:

1. Do one of the following:
  - Right-click on the second level node (group node) and select *Add Signal* option from pop-up menu.  
Or
  - Click *Add Signal* icon.

The *Add Signal* dialog box is displayed.



2. Specify the number of pins you wish to add and click *OK*.

Now you can manually enter the signal names and pin numbers. For scalar signals you can specify as A, B, C, D and so on whereas for vector signals you can specify as following A<10....1>. To reduce the manual effort FSP provides you Add Bus for Group <group\_name> dialog box to add the bus signals at one step.

### **Adding Bus Signals**

Add Bus for Group <group\_name> dialog box allows you to add bus signals to the bank. The dialog box also has some useful benefits over Add Pins dialog box:

- Sizeable pins can be entered
- Vector pins can be entered at one step
- IO standards and pin type information can be entered for all the bus signals at one step
- Differential pair pins can also be entered

To add bus signals using Adding Bus Signals dialog box:

1. Do one of the following:

- Right-click on the second level node (group node) and select *Add Bus* option from pop-up menu.

Or

- Click *Add Bus* icon.

The Add Bus for Banks <bank\_name> dialog box is displayed.



2. Specify the values in all the fields as required.

For detailed information on Add Bus for Bank dialog box see [Add Bus for Groups](#) section.

3. Click *OK*.

The new signals are displayed with the names in Signal Name column. You can either now start specifying the pin details manually or create another bank for your model.

## Adding Bank

Adding bank is required to combine the signals into single bank. Banks are created to logically arrange the pins. The bank details you specify are useful at the time of I/O synthesis process.

To create a new bank:

1. Do one of the following:

- ❑ Right-click on the first level node and select *Add Bank* option from pop-up menu.  
Or
- ❑ Click *Add Bank* icon.

The *Edit Bank* dialog box is displayed.



2. Specify the bank name in *Bank Name* field.
3. Specify the bank color in *Bank Color* field.
4. Enter description about the bank in *Bank Description* field.
5. Click *OK* of Edit Bank dialog box.

Once the bank is created you can now start adding pins to the bank. For more information see [Adding Signal](#) section.

## Adding Pin Details

Once the bank and logical pins are added to the spreadsheet editor, you can start specifying the pin details. Pin details need to be manually specified. There are few options available in the editor that are useful for organizing the pins and to make the editing process easier.

To add pin details perform the following steps:

1. In Signal Name column, perform any one of the following:
  - For scalar pins, specify the name as A, B, C, D.
  - For vector pins, specify the name as A<10...1> or you also specify through Add Bus dialog box.
2. Specify the pin numbers in *Pin Number* column.
3. Specify the pin type from the *Pin Type* drop-down list.
4. Specify the voltage value for power pins in *Voltage Column*.
5. Differential Type (Diff Type), Differential Pair Pin (Diff Pair Pin) and Serial IO TX/RX Pin column can be auto filled through *Auto Detect Pin Pair* dialog box. For more information see Auto Detecting Pin Pair section.

6. X and Y locations column field values can be autofilled by *Read Pin Locations* dialog box. For more information see *Updating Pin Locations* section.
7. Symbol Pin Name column field values are filled after mapping files.
8. Click on a single cell in *Description* column.  
A *browse (...)* button is displayed.
9. Click *browse (...)* and enter description for the pin and click *OK*.
10. Click on a single cell in *Custom Attribute* column and specify the attribute and its value and click *OK*.

## AutoMapping Diff Pair Pins

The Auto Detect Pin Pairs feature lets you automatically search and map the differential pair pins. This feature saves your time and removes the burden of manual searching of few differential pair pins among thousands of pins. Other than differential pair pins you can also search the serial IO TX/RX pins, Positive and Negative pins. Depending on the selected search criteria the pins are detected.

To auto detect pin pairs perform the following steps:

1. Right-click on the first level node and select *Auto Detect Pin Pair* option from pop-up menu or click *Auto Detect Pin Pair* icon.  
The Auto Detect Pin Pairs dialog box is displayed.
2. In Signal Name Regular Exp. for field click and select the type of pair pins from drop down list for which you want to perform the search operation.
3. In Fetch text box specify the alphabet/word/character you are searching for in differential pair pins. Refer to the Auto Detecting Pin Pairs examples section.
4. Click *Fetch*.  
The list of pin names and differential pair pins are displayed. See below.
5. Click *Check All* to select all the pins.
6. In *Set Positive Column* field, select *First Column* to define all the P types pin as positive differential type or select *Second Column* to define all the N type pins as positive differential type.
7. Click *OK*.

After clicking OK, the selected pins are treated in accord with the option selected as Differential Pin Pairs, Positive, Negative, or Serial I/O TX/RX Pin.

### ***Auto Detect Pin Pairs Examples***

If you are searching the differential pin pair for the pins (click the drop-down arrow and select Diff. Pair Pin from drop-down list)

- ❑ DDR2\_N[0] and DDR2\_P[0] the difference in the pin names is at only the middle point, i.e., N and P. Enter expression as PIN in the Fetch text box and click *Fetch*.
- ❑ PosData<0> and NegData<0>, the difference in the pin names is at only the first point, i.e., Pos and Neg. Enter expression as PosNeg in the Fetch text box and click *Fetch*.
- ❑ CLK and CLK\_N, the difference in the pin names is at only the end point, i.e., K and K\_N. Enter expression as KIK\_N in the Fetch text box and click *Fetch*.

If you are searching for the Serial I/O TXIRX pin, click the drop-down arrow and select Serial IO TX/RX Pin from the drop-down list.

- ❑ TXPRADA and RXPRADA, the difference in the pin names is at only the first point, i.e., TX and RX. Enter expression as TXIRX in the Fetch text box and click *Fetch*.
- ❑ Data\_T[0] and Data\_R[0], the difference in the pin names is at only the middle point, i.e., T and R. Enter expression as TIR in the Fetch text box and click *Fetch*.
- ❑ Data Transmit and Data Receiver, the difference in the pin names is at only the end point, i.e., Transmit and Receiver. Enter expression as Transmit and Receiver in the Fetch text box and click *Fetch*.

### **Defining Patterns for Connector Pins**

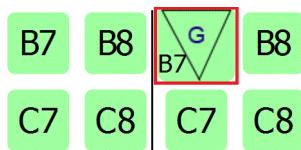
In Rules Editor, manually defining the pin types, differential pairs, and SerDes pairs is a tedious task, which requires a significant effort. To reduce the effort, you can opt to import the pin details from an external file to create a connector. Even after importing the pin details, you have to manually perform some tasks to create a connector precisely. To overcome this problem, FSP provides you a graphical solution to quickly define pin types (or patterns) for the connector pins or create a custom pattern with mouse clicks.



In this section, the term *Pattern* refers to the pin type on a single pin and the term *Custom Pattern* refers to the group of patterns (pin types).

The Rules Editor provides several patterns, such as single ended, differential, or serdes patterns. You can select one of the available patterns and click on the pin on which you want to apply the pattern. After applying the pattern, the pin symbol is changed to the selected pattern symbol. For example, if you select the *Ground* pattern and click on a pin, the pin symbol change to the *Ground* symbol as depicted in the following image:

**Before Applying  
Ground Pattern**    **After Applying  
Ground Pattern**



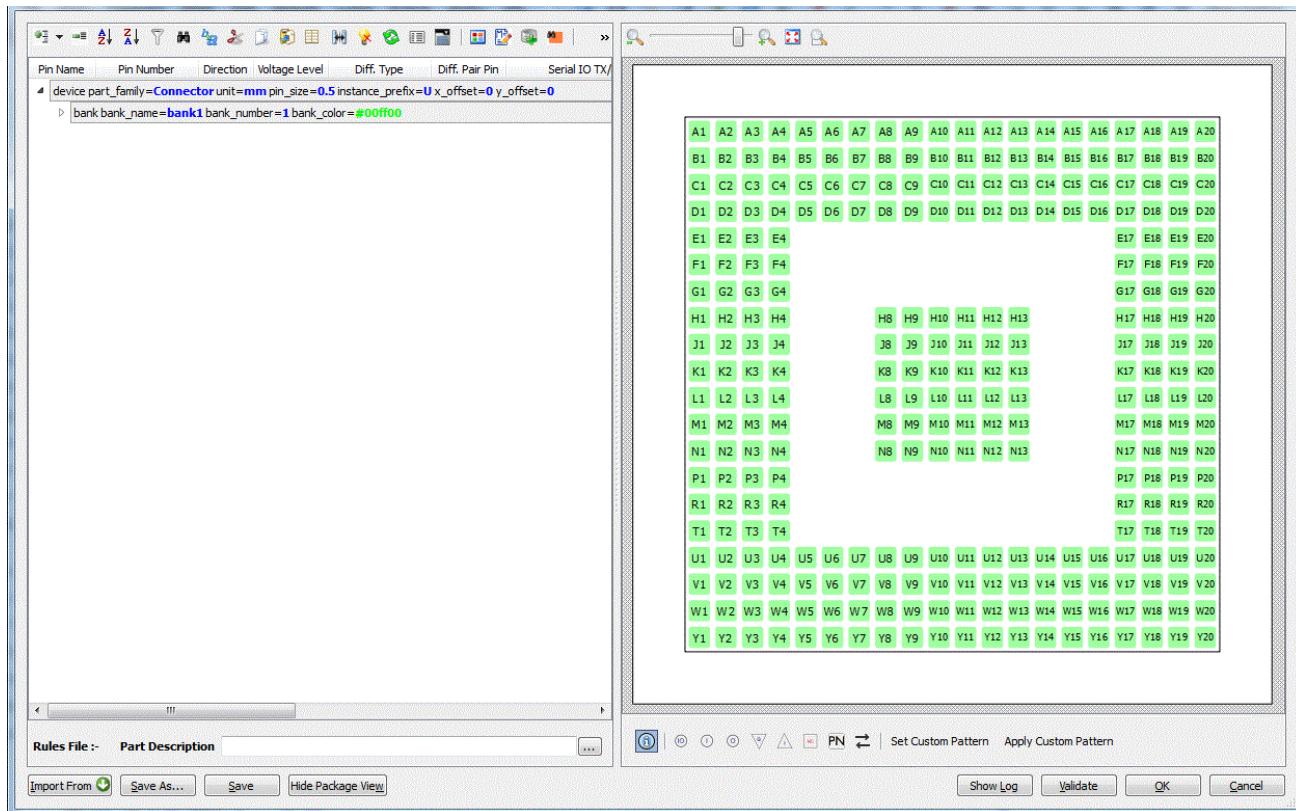
You can also override the existing pattern with a new pattern by selecting the new pattern and applying it on the existing pattern.

You can also create a custom pattern in addition to a single pattern. Creating a custom pattern involves grouping a set of single-ended patterns in a particular sequence. After creating a custom pattern, you can replicate it on the other pins. For more information, see the [Creating Custom Pattern](#) section.

The Rules Editor is divided into two views, *Spreadsheet View* and *Package View*.

# Allegro FPGA System Planner User Guide

## Creating Connectors



When you apply a new pattern or modify the existing pattern in the *Package View*, the changes are reflected immediately in the *Spreadsheet View* and vice versa. You can also make the changes directly in the *Spreadsheet View*.

### Defining a Single-Ended Pattern

The Rules Editor provides the following single-ended patterns:



- GND (D)
- Power
- No Connect
- Input and Output (IO)
- Input (I)
- Output (O)

To apply a pattern, you need to select the icon for the patterns and then click any pin on which you want to apply the pattern.

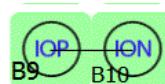
### Defining a Differential Pair Pattern

Rules Editor also provides a pattern option to define a differential pin pair.

To define a different pin pair pattern perform the following steps:

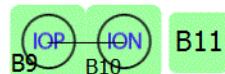
1. Click the PN icon.
2. Click on any pin in the package view.  
P-type pin is created.
3. Click another pin in the package view.

N-type pin is created. In addition a line is automatically drawn between the two pins (P and N), marking them as a differential pin pair as depicted in the following figure.

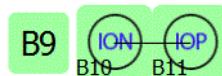


#### Note:

When you apply a new differential pin pair pattern on a pin that is already a part of another differential pair, the pin type of the other pin is reset. For example, if you have the following pattern.



Pin *B10* is part of a differential pair. When you click *B11* and *B10*, they *B11* and *B10* are paired as a new differential pair and *B9* is reset as a normal pin as depicted in the following figure:



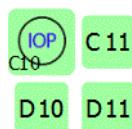
## Defining a SerDes Pattern

Rules Editor also provides a pattern option to support serdes pin pairs. Use this option to define a SerDes pattern on four pins. When you click the SerDes Pattern icon followed by four different pins, a SerDes pattern is created.

To define a Serdes pattern, perform the following steps:

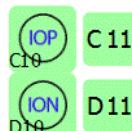
1. Click the *SerDes Pattern* icon.
2. Click on any pin in the *Package View*.

A positive input (IOP) pattern is created.



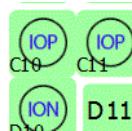
3. Click the another pin.

A negative input (ION) pattern is created.



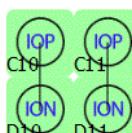
4. Click the another pin.

A positive output pattern is created.



5. Click the another pin.

A negative output pattern is created.



A line is automatically drawn between the Positive-Input and Negative-Input pins, and the Positive-Output and Negative-Output pins.

**Note:** When you apply a new SerDes pattern on a pin that is part of another SerDes pair, the pin type of the other three pins is automatically reset.

### **Creating Custom Pattern**

A custom pattern is a series of pins that contain a common pin pattern. A custom pattern can contain one or more pins of same type such as IO, ground, power pins, and differential pairs. You can also include a combination of different pin patterns in a custom pattern.

A custom pattern is useful, when a connector contains a group of pins of same type and you want to quickly define the pin types for the connector. FSP provides a mechanism to quickly create a pattern consisting of a group of pins graphically. After creating a pattern, you can apply the pattern on the connector pins.

The proper procedure to create a custom pattern are as follows:

- Start the Custom Pattern mode.

To create a custom pattern, start the Custom Pattern mode by clicking the *Set Custom Pattern* icon.

- Select the existing patterns.

After starting the Custom Pattern mode, you can start selecting the existing patterns to include in the custom pattern. You do this by clicking on the pin in the *Package View*. The first pin that you select is set as *Anchor Pin*. An anchor pin is the pin based on which the direction of the remaining pins are stored in the patterns.

**Note:** You cannot select a pin whose pattern is not defined.

- Stop the Custom Pattern mode.

Once you finish the pin patterns selection, you can stop the Custom Pattern mode by clicking the *Finish Selection* icon.

### **Guidelines for Creating a Custom Pattern**

- Pattern must be defined for the pins before creating a custom pattern, this means you cannot include a pin in a custom pattern on which pattern is not defined.
- After starting the custom pattern mode, if you try to perform other operations such as defining single-ended patterns or differential or serdes patterns, the current custom pattern mode stops and the patterns information is lost. Before applying any other pattern, it is recommended that you first finish the custom pattern pin selection and stop the mode.

# Allegro FPGA System Planner User Guide

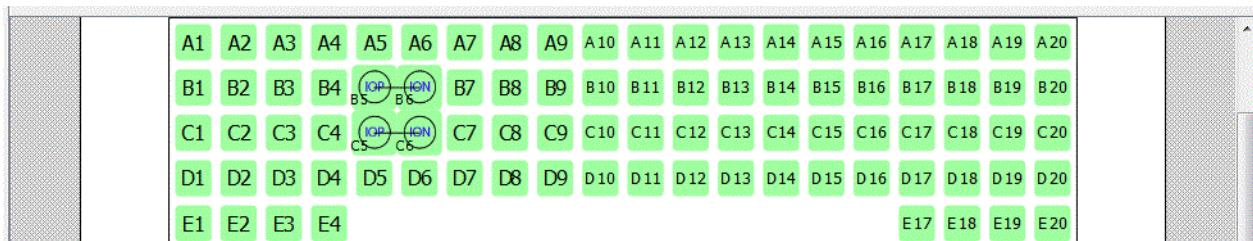
## Creating Connectors

- The custom pattern will override the existing pattern if you apply a custom pattern on an existing pattern or on a pin which is already a part of another custom pattern.
- If the pins are insufficient to apply a custom pattern at the corner of the BGA, the custom pattern is tried to fit on the available pins and rest of the patterns in the custom pattern are ignored.

The following example demonstrates how to create a custom pattern:

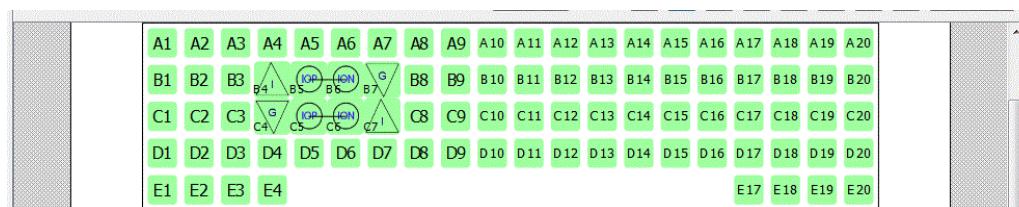
1. Click the *SerDes Pattern* icon.
2. Click the *B5*, *B6*, *C5*, and *C6* pins.

The following pattern is created.



3. Click the *Ground* icon.
4. Click the *B7* and *C4* pins.
5. Click the *Power* icon.
6. Click the *B4* and *C7* pins.

Patterns creation is completed. You can now create a custom pattern.

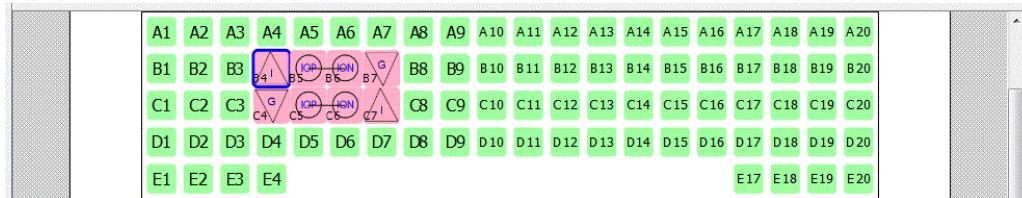


7. Click the *Set Custom Pattern* icon.
  8. Click on the *B4*, *B5*, *B6*, *B7*, *C4*, *C5*, *C6*, and *C7* pins.
- Note:** B4 is set as anchor pin since it is selected first.
9. Click the *Finish Selection* icon.

# Allegro FPGA System Planner User Guide

## Creating Connectors

The custom pattern pins are highlighted in light pink color.

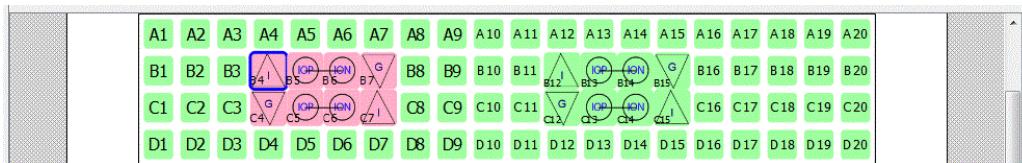


10. Click the *Apply Custom Pattern* icon.

You can now apply the custom pattern anywhere in the Package View.

11. Click the *B12* pin to apply the custom pattern.

The *B4* pin pattern is applied to the *B12* pin and the remaining pins are applied as per the pins are aligned relative to the custom pattern.



## Adding Custom Attributes

The custom attributes pane allows you to add the attributes to pins of the model. After specifying the custom attributes save the model, the attributes get updated in the front-end symbol file of the instance.

To add the custom attribute:

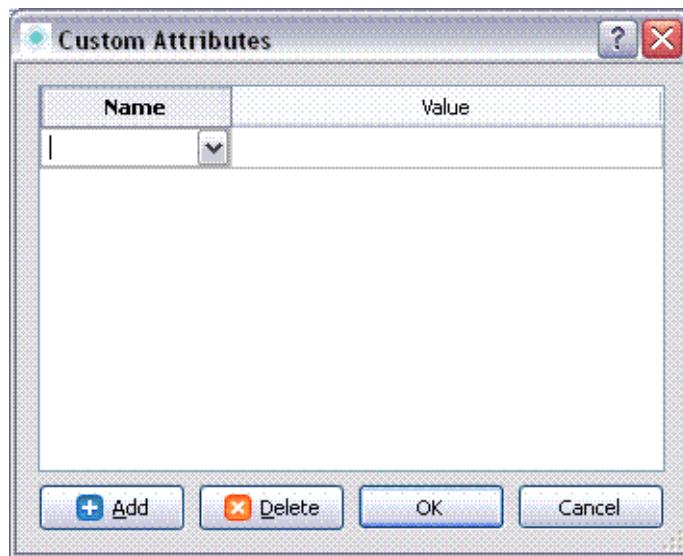
1. In Custom Attribute column do one of the following:

- a. Double click on a cell.
- b. Click *browse (...)*

Or

- Select the cell and press **Ctrl + T** to invoke Custom Attribute dialog box.

The Custom Attribute dialog box is displayed.



2. Click *Add*.
3. Specify a attribute name in *Name* column.
4. Specify a value for the attribute in *Value* column.
5. Click *OK*.

The attribute name and value is displayed in Custom Attribute column of the Rules Editor.

## Rearranging Pin Order

You can rearrange the pins in the Rules Editor according to your requirements and set the pin order. Pins can be moved across any banks and banks can be merged with any other bank to change the schematic view of the component. FSP allows you to move the individual pins as well as bus pins across group.

The following are the pop-up menu options available in Rules Editor, that will help you to quickly rearrange the pins or banks.

### Move Selected Pins to Bank

To move the pins from one bank to another bank,

1. Select the pin(s) in a bank that you wish to move.

2. Right-click on the selected pin(s) and select *Move Selected Pin(s) to Bank* from pop-up menu.

The pins are moved to the selected bank.

## Merging Banks

You can merge the banks into another bank to change the schematic view of the component.

To merge the banks:

1. Right-click on the group node and point to *Merge With Banks* option.  
The list of bank names available in the current model is displayed.
2. Select any one of the bank name.

The selected bank merges with new selected bank along with all pins and properties.

## Editing Connector

Using Rules Instance Editor, you can edit the connector model pin definitions at any time during the design. Modification can be of any type, adding new or modifying existing bank level or pin level details. Connectors which are placed on canvas can also be modified. But it recommended not to edit the model from the canvas since it may delete the connectivity definitions and leads to error prone state.

To edit the connector model do the following:

1. Right-click on the connector instance and click *Edit Rules*.  
The Rules Instance Editor dialog box is displayed.
2. Do the modifications as required.

**Note:** Since you are allowed to do any type modifications in the editor, you can follow the steps explained in Creating Part Manually section.

3. Click *Save* to save the changes.
4. Click *OK* of Rules Instance Editor.

## Saving the Connector Model

Connector model can be saved in any directory.

## **Allegro FPGA System Planner User Guide**

### Creating Connectors

---

1. Click *Save As* to save the model in other directory.
  2. In *Save Rules File* dialog box, specify the directory path with file name or click *browse* (...) to the location where you want to save the model.
  3. Click *OK*.
- Or
4. Click *Save* to save the model in current project directory.

This completes the creation of a connector model.

## Creating Testing Connector

Before you walk through the steps of creating testing connector, you must know the following terms and acronyms used in this section.

---

<b>Terms and Acronyms</b>	<b>Description...</b>
Time Domain	<p>In general a testing connector has thirty two pins and these thirty two pins are divided into two sets of groups called Time Domain groups. And each time domain groups contains sixteen pins each.</p> <p>For example,</p> <p>Time Domain (TD1) – x16 Pins</p> <p>Time Domain (TD2) – X16 Pins</p>
Scan Channels	<p>Time domain groups are further divided into four types of sub groups. These sub groups are called as Scan Channels and represented as XSC2, XSC4, XSC8 and XSC16.</p> <p>For example,</p> <p>A time domain group of eight pins will have the following scan channel pins:</p> <ul style="list-style-type: none"><li>■ one pin is considered as X8 pin</li><li>■ two pins are considered as X4 pins</li><li>■ four pins are considered as X2 pins</li></ul> <p>Any single connector pin can also be considered as more than one type of scan channel pins.</p> <p>For example,</p> <p>A testing connector pin can be considered as SC2 and SC4 i.e. SC2_SC4 or SC2, SC4, SC8 i.e. SC2_SC4_SC8 and so on.</p>

---

In FPGA System Planner, a testing connector is logical model which contains different group and pin properties based on which synthesis is performed by FSP. These types of connector are specifically designed to be connected to Device under Test (DUT) model.

**Note:** Device under Test is a logical model which contains logical grouping and target testing connector pin properties defined against each of the DUT pins.

**Note:** The details of device under test model creation and methodology of connecting a DUT model with testing connector is not captured in this User Guide. For more details see multi test feature application notes.

A typical testing connector model contains two types of banks i.e. Normal and Power banks. Normal Bank contains time domain groups and scan channel pins where as Power Bank contains LC and HC type banks. LC and HC type banks does not have time domain groups and scan channel pins.

To create a testing connector you use Rules Editor. There are much more options are provided in Rules Editor for testing connector. You can create the connector model manually or import the pin and group information from external files of different format. The connector model information is saved in rule file (.lrf). You can also edit the connector model at any time during the design.

## Creating Testing Connector Manually

This section describes the procedure for creating a testing connector manually using an connector model example. The testing connector model has the following definitions:

- Two time domain groups with sixteen pins each.
- Each time domain group should contain:
  - X16 – 1 pin
  - X8 – 2 pins
  - X4 – 4 pins
  - X2 – 8 pins
- Two power bank as LC and HC with thirty pins each

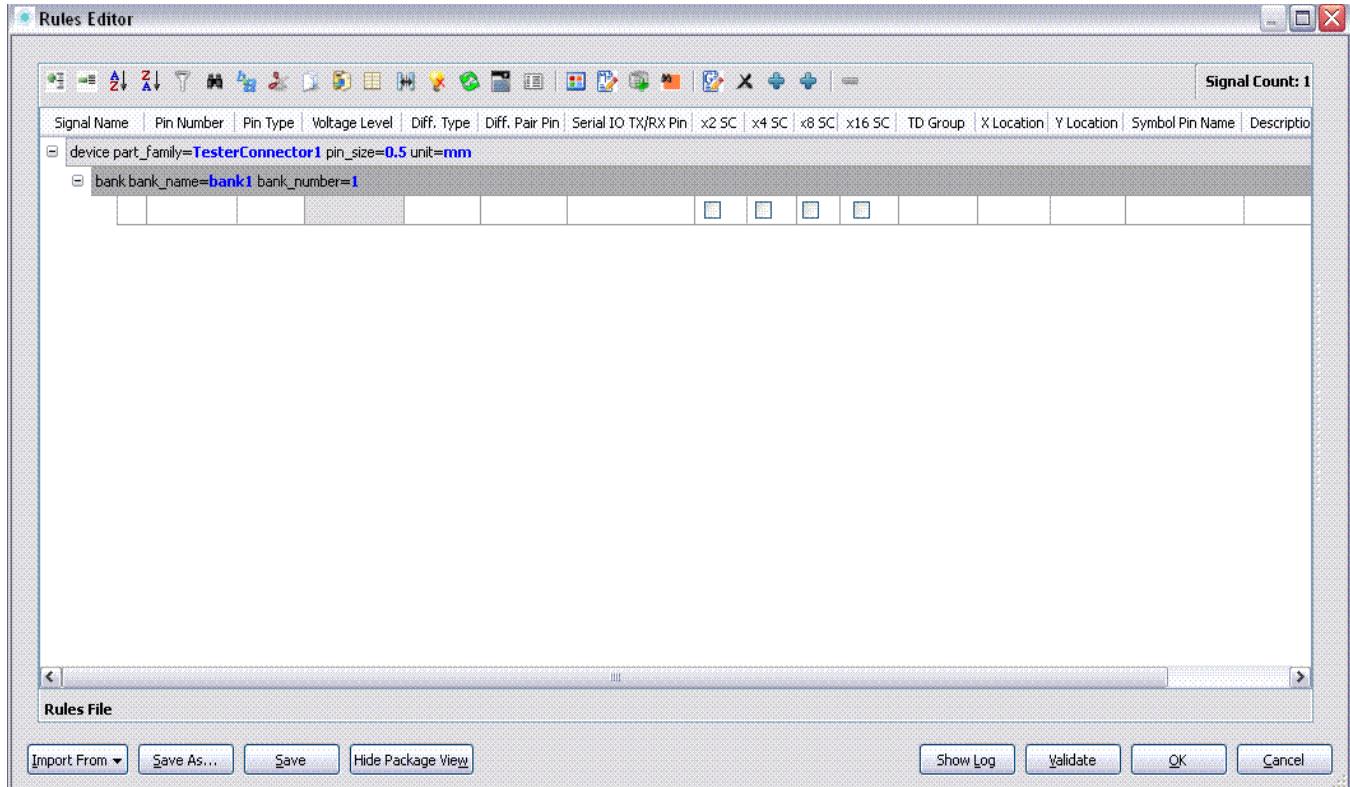
To create testing connector model perform the following steps:

1. Choose *Library – Create Part – Testing Connector1*.

The Rules Editor is displayed with default bank.

# Allegro FPGA System Planner User Guide

## Creating Connectors



2. Right-click on the first level node and select *Edit Properties*.

The Edit Properties dialog box is displayed.

3. Specify the field values as required and click *OK*.

4. Right-click on the second level node and select *Edit Bank*.

The Edit Bank dialog box is displayed.

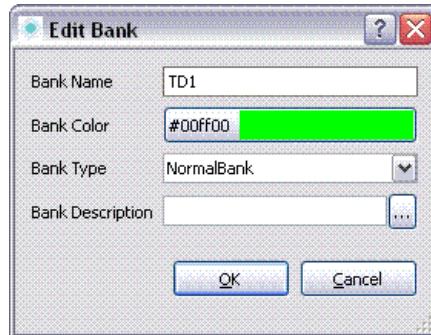
5. Do the following:

- a. Enter TD1 as bank name in *Bank Name* field.
- b. Select the bank color in *Bank Color* field.
- c. Click and select *Normal Bank* option from *Bank Type* drop down list.
- d. Click *OK*.

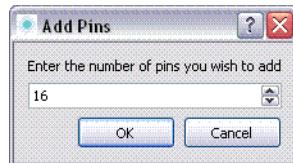
# Allegro FPGA System Planner User Guide

## Creating Connectors

---



6. Click *Add Signal* icon in tool bar.
7. Enter sixteen as number of pins in *Add Pins* dialog box.

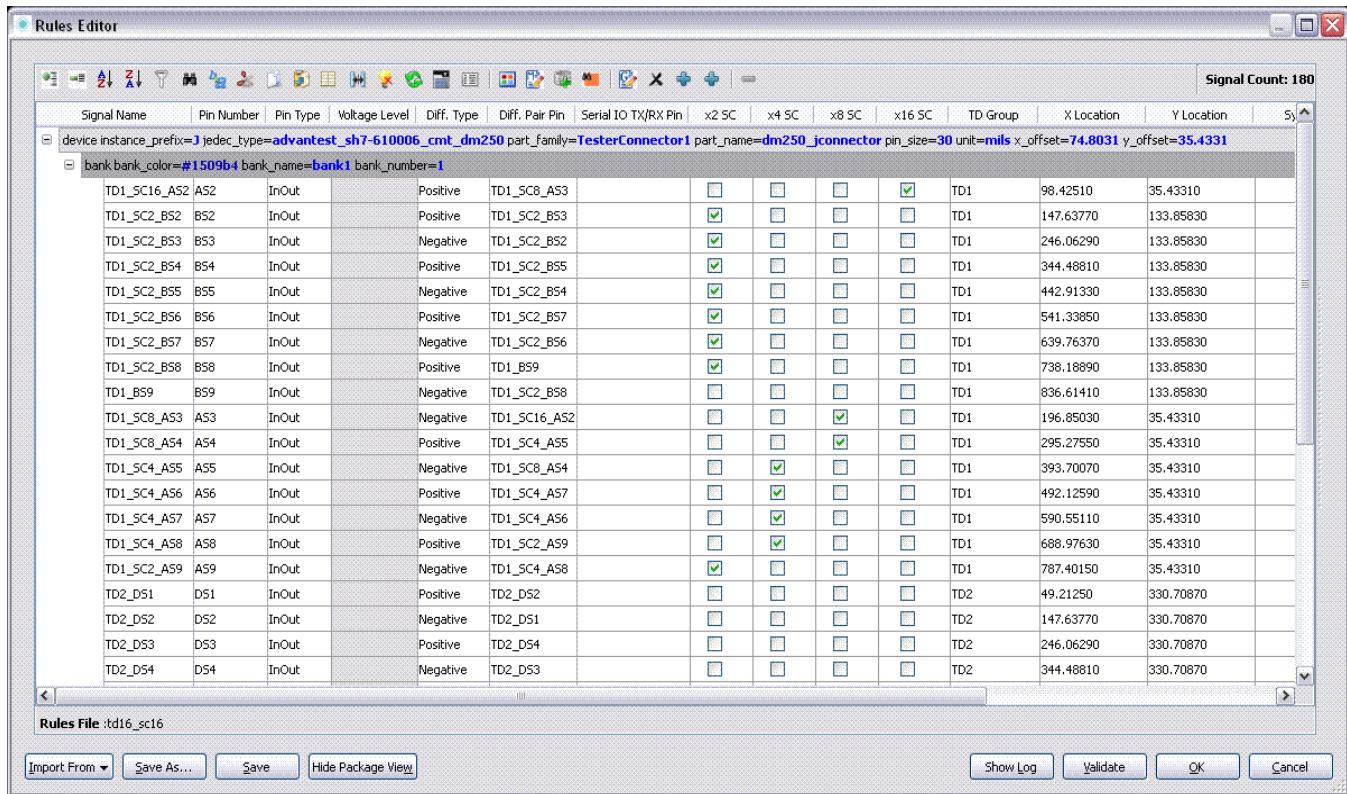


8. Specify *Signal Name*, *Pin Number*, *Pin Type* column values.
9. In *TD Group* column, specify *TD1* for all the pins.
10. Select scan channel options exactly as shown in figure below.

# Allegro FPGA System Planner User Guide

## Creating Connectors

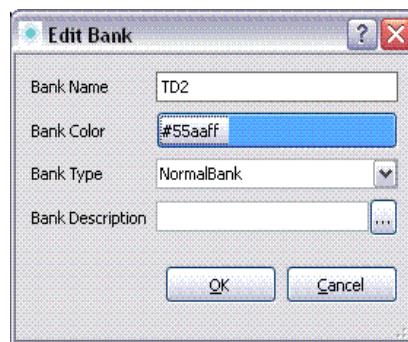
---



**11.** Click Add Bank icon in to create another bank as TD2.

**12.** Follow the steps 8 to 13.

**Note:** Specify the bank name as TD2.



These completes the creation of time domain groups and scan channel pins.

**13.** Click Add Bank icon in to create another bank.



14. Specify the bank name as GND\_Y in *Bank Name* field.
15. Select the bank color in *Bank Color* field.
16. Click and select *HC Bank* option in *Bank Type* field and click *OK*.
17. Add thirty pins using *Add Pins* dialog box.
18. Specify all the column values as required.
19. Create another bank with name LC by following steps 19 to 22.
20. Click *Validate* to validate the pin definitions entered.
21. Click *Save* to save the model in current project directory.
22. Specify as :<rules file name> in Rules File field of Save Rules File dialog box and click *OK*.
23. Click *OK* to complete the process.

This completes the testing connector model creation.

---

## **Automation in FSP**

---

This chapter covers the scripting and automation feature available support in FSP. The topics covered in this chapter:

- Working with FSP TCL



This chapter does not contains the description of each FSP TCL commands. For more information, see the *FSP TCL Command Reference* guide.

## Working with FSP TCL

FPGA System Planner includes a scripting functionality that lets you execute FSP TCL commands through a TCL Command bar. FSP also provides the support to store and later replay the command.

Every FSP command is logged in the form of a TCL script command. This command that is logged is registered with a TCL interpreter. When the command is played back, FSP communicates with the TCL interpreter to retrieve the command and execute it. This process makes logging and replaying of commands an intuitive and simple task.

The topics covered in this section are:

- [Using the TCL Command window](#)
- [Executing a TCL Script](#)

### Using the TCL Command window

The FSP's Messages window includes a TCL Command bar at the bottom. You use this window to execute a TCL command. Also, when you execute a TCL command in this window, the associated command is registered with the TCL interpreter and the command is logged in the window.

To open the TCL Command bar, perform the following step:

- Choose *Window – Messages*.

The TCL Command bar is located at the bottom of the *Messages window*.

### Executing a TCL Command

You use the TCL Command bar in Messages window to execute a TCL command. To execute a TCL command in the TCL Command bar, you simply type the command in the bar and press *Enter*. If any error occurs in executing the command, the error message displays in the *Messages window*.

**Note:** All TCL commands are case-sensitive.

Besides the FSP TCL commands, you can also execute the native TCL commands.

For example, the following TCL command returns the absolute path of the current working directory.

pwd

And to change the working directory, you can use:

cd

## Executing a TCL Script

You can create a FSP TCL script manually, in any text editor. After creating a TCL script, you can execute the script directly from the TCL Command bar by the source command.

source <Script path and name>

For example, to execute the script D:\working\sample\run.tcl

source D:/working/sample/run.tcl

**Note:** To specify a file location in a TCL command argument, you need to use the forward slash (/) path separator.

If you execute a TCL script in FSP that contains errors, the script exits after it encounters the first error and the error is displayed in the *Messages window*.

# **Allegro FPGA System Planner User Guide**

## Automation in FSP

---

---

## **Dialog Box Descriptions**

---

This appendix describes the dialog boxes in FPGA System Planner. The dialog boxes are listed in alphabetical.

## Add Bus for Groups

The Add Bus dialog box lets you to add new signal bus in protocol editor.

### ***How to Access the Add Bus dialog box***

- In Rules Editor, right-click on the second level node and select *Add Bus* or click the *Add Bus* toolbar icon.

Name	Description
Bus Name	Enter the name of bus to create
Is Differential	Select the check to define differential pair signals or buses.
Diff Pair Name	Enter the name of Diff Pair or buses
Bus Range	Enter bus range for bus creation
IO Standard	Click and select the appropriate IO standard.
Pin Type	Click and select the pin direction.
OK	Click to add the bus.
Close	Click to cancel the operation.

## Add Bus for Bank

The Add Bus for Bank dialog box lets you to add new bus signals in Rules Editor for connector models.

### **How to Access the Add Bus for Bank**

- Choose *Library – Create Part – Connector or Testing Connector1*.  
The Rules Editor dialog box is displayed.
- Right click on the second level node or click *Add Bus* toolbar icon.

Name	Description
Bus Name	Enter the name of bus to create.
Is Differential	Select the check to define differential pair signals or buses.
Diff Pair Name (N)	Enter the name of Diff Pair or buses
Bus Range	Enter bus range for bus creation
Pin Type	Click and select the pin direction.
OK	Click to add the bus.
Close	Click to cancel the operation.

## Add Group

The Add Group dialog box allows you to add new group in Rules Editor.

### ***How to Access the Add Group dialog box***

- In Rules Editor, right-click first level node and select *Add Group*.

Name	Description
Group Constraint	Lets you apply the constraints on the groups and pins of the interface.  Select the group constraint from the drop-down list to apply constraint on the pins.  For instance, select Same Bank as group constraint. During synthesis all the pins of the group will connect to a single bank in the targeted FPGA.  The group constraint list may vary based on the target FPGA.
Group Name	Enter the name of the group.
Group Color	Select the color for the group.  On the design board canvas view, pins in the group appear in the color that you set here. It helps to differentiate multiple groups.
Group Use Bank	Click and select the bank names from the list to which you want to connect.
Group Don't Use Banks	Click and select the bank names that you want to preserve or exclude from the connections.
Group Contiguous Pins	Click and select the contiguous pins.
Group Description	Enter description for the group.
OK	Click OK to save the settings.

## Add Bank

The Add Bank lets you to add a new bank in Rules Editor for connectors.

### **How to Access the Add Banks dialog box**

- Choose *Library – Create Part – Connector or Testing Connector1*.  
The Rules Editor dialog box is displayed.
- Right-click on the first level node or click the *Add Bank* toolbar icon.

Name	Description
Bank Name	Enter a name for the new bank.
Bank Color	Click and select the color for the bank from <i>Select Color</i> dialog box.
Bank Type	Click and select the type of bank you want to create from the drop down list.  For example,  Select the <i>Normal Bank</i> option to create a normal bank for connector model.
Bank Description	Enter description for the bank.

## Auto Detect Pin Pairs

The Auto Detect Pin Pairs dialog box lets you define and create the differential pairs, serial input and output pins, positive and negative pins. To create pairs, first you search the pairs and map them with the associated pairs. The Auto Detect Pin Pairs dialog box is available in various FSP editors such as Protocol Editor, Rules Editor, Virtual Interface Editor, and Rules Instance Editor.

### ***How to Access the Auto Detect Pin Pairs dialog box***

You can access the Auto Detect Pin Pairs dialog box by performing any one of the following:

- Click Auto Detect Pin Pairs tool bar icon in Edit Protocol/Rules Editor/Rules Instance Editor.
- Right-click the first level node and select *Auto Detect Pin Pairs*.

Name	Description
Signal Name Regular Exp. for	Displays the following type of pair pin names: <ul style="list-style-type: none"><li>■ Diff. Pair.Pin</li><li>■ Serial I/O TX/RX</li><li>■ Use Positive Pins</li><li>■ Use Negative Pins</li></ul> Select any one of the option for which you want to perform the search operation.
Find text box	Enter the search expression or keyword.
Fetch	Click this option to search for the specified expression. After clicking this option a list of signal names which contains the entered keyword is displayed.
Check All	Click to select all the detected pins.
UnCheck All	Click to un select all the detected pins.
First Column	Select this radio button to set the pins present in the first column as positive.
OK	Click to set the selected pins.
Cancel	Click to cancel the operation.

## Examples of Auto Detect Pin Pairs

This section shows few examples to perform search using regular expression in the *Auto Detect Pin Pairs* dialog box.

**Note:** Before you go through the following examples, it is important that you first understand and gain experience in the regular expressions. For more information, see the [Using Regular Expression](#) section.

### **Example1**

To search for differential pairs bus, such as DDR2\_CLK0\_N, DDR2\_CLK1\_N, DDR2\_CLK2\_N, DDR2\_CLK3\_N and DDR2\_CLK0\_P, DDR2\_CLK1\_P DDR2\_CLK2\_P, DDR2\_CLK3\_P, use the regex `\w_CLK(\d+)_\{PIN\}`.

'\w' matches DDR2 and (\d+) matches 0, 1, 2, 3.

### **Example2**

To search the PosData<0> and NegData<0> pairs, you use 'l' character. The 'l' is an alternation metacharacter. This metacharacter matches two different character strings, (either THIS) l (or THAT). To match *Pos* or *Neg*, you should use the regex PosINeg. The regex PosINeg matches the first alternative Pos and displays the signal names containing Pos string. Next it matches the alternative Neg.

After search completes, FSP returns a list of strings containing Pos or Neg and displays in the middle of the dialog box as depicted below.

```
PosData<0>
...
...
...
PosData<10>
NegData<0>
...
...
...
NegData<10>
```

### **Example3**

To search the C00CVREF0\_AE64 and C00CVREF1\_AM68 pairs, you use the regex [A-Z0-9]+[0|1]\_[A-Z0-9]+.

[ ] is a character class, which allows a set of possible characters to search, rather than just a single character. ‘-’ is a range operator with character class and ‘+’ is a metacharacter which searches the word or numeric character for one or more times.

[A-Z0-9]+ subpattern matches a word character containing numeric character, such as C00CVREF.

[0|1] subpattern matches C00CVREF0 or C00CVREF1.

‘\_’ character is considered as a normal character, that means it has no special meaning in this regex, C00CVREF0\_ and C00CVREF1\_.

[A-Z0-9]+ subpattern matches any word character containing numeric character at the end of string, such as C00CVREF0\_AE64 and C00CVREF0\_AE68.

#### ***Example4***

To search the MXAWCK\_A111\_8000 and CASENK\_A6\_8001 pairs, you can use the regex [A-Z]+\_A[0-9]+\_[0-9]+ or (\w+)\_A(\d+)\_(\d+).

[A-Z]+ subpattern matches the two strings, MXAWCK and CASENK. You can also use (\w+) inplace of [A-Z]+. \w matches a word character (alphanumeric or \_ ).

\_A’ is considered as normal characters.

[0-9]+ subpattern matches 111 and 6. You can also use (\d+) inplace of [0-9]+. \d matches a digit.

\_’ is considered as normal character.

[0-9]+ subpattern matches 8000 and 8001. You can also use (\d+) inplace of [0-9]+. \d matches a digit.

#### ***Example5***

To search the EDC<test23P>\_AH1 and EDC<test23N>\_AJ7 pairs, you can use the regex (\w+)(.\*)([a-z]+[PIN])(.\*)([A-Z0-9]+) or (.\*)[PIN](.\*).

(\w+) subpattern matches EDC.

(.\*) subpattern matches any character, <.

[a-z]+ subpattern matches word character in lower case, test.

## **Allegro FPGA System Planner User Guide**

### Dialog Box Descriptions

---

[PIN] subpattern matches P or N.

(.\*\_) subpattern matches any character followed by '\_' character, >\_.

[A-Z0-9]+ subpattern matches AH1 and AJ7.

You can also minimize the regex `(\w+)(.*)[a-z]+[PIN](.*_)` by using pattern `(.*)[PIN](.*)`.

(.\*) subpattern matches as much as string as possible, till P or N, that is, EDC<test23 and the last subpattern (.\* ) matches the remaining string after P and N, that is >\_AH1 and >\_AJ7.

## Add Part (DE-HDL)

The Add Part dialog box is the second option to place the interface rules file on canvas after Create>Select Component Rules and Mapping Information wizard flow. To use the Add Part dialog box first you must hide the wizard by selecting *Show wizard while placing part on canvas* option in Preferences window. After selecting the symbol using Component Browser the Add Part dialog box is invoked inplace of wizard. The Add Part dialog box allows you to quickly select the rules file and mapping file and place it on canvas.

### **How to Access**

- Choose *File – Preferences*.  
Click the *Design* tab in the Preferences dialog box.
- Select the *Show wizard while placing part on canvas* option.
- Do one of the following:
  - Click Add Part icon in the tool bar.
  - Choose *Component – Add Part*.The Component Browser is displayed.
- Select the cell name and click *Select*.  
The Add Part dialog box is displayed.

Name	Description
Use Existing Rules and Mapping File	This option lets you use the existing rules and mapping files from working directory and rules file directories.  The <i>Rules File</i> and <i>Mapping File</i> fields are enabled when you select this option.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

<b>Generate New Rules and Mapping File</b>	<p>This option lets you generate a new rules and mapping files. You can also create new connector or testing connector using this option.</p> <p>Perform any of the following steps:</p> <ul style="list-style-type: none"> <li>■ Select <i>Interface</i> to create a interface rules file.</li> <li>■ Select <i>Connector</i> to create a connector rules file.</li> <li>■ Select <i>Testing Connector1</i> to create a testing connector rules file.</li> </ul> <p>This option disables the <i>Rules File</i> and the <i>Mapping File</i> fields.</p> <p>The rules and mapping files are by default generated in the working directory.</p> <p><b>Note:</b> The working directory is set in the Rules File Path Editor dialog box.</p>
<b>Library:Cell:View</b>	Displays the selected library:cell:view names.
<b>Show PTF Attributes</b>	Click this option to select a PTF row.
<b>Rules File</b>	<p>This option lets you browse and select a interface rules file. This option is enabled only when you select the <i>Use Existing Rules and Mapping File</i> option.</p> <p>Click the drop-down button to select the rules file. The drop-down list displays a list of rules file fetched from the PTF file and the <i>Select Other Rules File</i> option. You may select a relevant rules file names or click the <i>Select Other Rules File</i> option to select the rules file from the Libraries.</p>
<b>Mapping File</b>	<p>This option lets you browse and select a mapping file. This option is enabled only when you select the <i>Use Existing Rules and Mapping File</i> option.</p> <p>The relevant mapping file is displayed after specifying the rules file name in the <i>Rules File</i> field. The mapping files are fetched from the lib:cell:view and all the LRF search path directories.</p>
<b>Define Mapping</b>	Use this option to create a new mapping file. The Rules Signal Mapper dialog box is displayed when you click this option.
<b>Place</b>	Lets you place the rules file on the canvas.

## Add Part (OrCAD)

The Add Part dialog box is the second option to place the interface rules file on canvas after Create>Select Component Rules and Mapping Information wizard flow. To use the Add Part dialog box first you must hide the wizard by selecting *Show wizard while placing part on canvas* option in Preferences window. After selecting the symbol the Add Part dialog box is invoked inplace of wizard. The Add Part dialog box allows you to quickly select the rules file and mapping file and place it on canvas.

### **How to Access**

- Choose *File – Preferences*.  
Click the *Design* in Preferences dialog box.
- Select the *Show wizard while placing part on canvas* option.
- Do one of the following:
  - Click Add Part icon in tool bar.
  - Choose *Component – Add Part*.

The Add Part dialog box is displayed.

Name	Description
Use Existing Rules and Mapping File	This option lets you use the existing rules and mapping files from working directory and rules file directories.  The <i>Rules File</i> and <i>Mapping File</i> fields are enabled when you select this option.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

<b>Generate New Rules and Mapping File</b>	<p>This option lets you generate a new rules and mapping files. You can also create new connector or testing connector using this option.</p> <p>Perform any of the following steps:</p> <ul style="list-style-type: none"> <li>■ Select <i>Interface</i> to create a interface rules file.</li> <li>■ Select <i>Connector</i> to create a connector rules file.</li> <li>■ Select <i>Testing Connector1</i> to create a testing connector rules file.</li> </ul> <p>This option disables the <i>Rules File</i> and the <i>Mapping File</i> fields.</p> <p>The rules and mapping files are by default generated in the working directory.</p> <p><b>Note:</b> The working directory is set in the Rules File Path Editor dialog box.</p>
<b>Library File (*.olb)</b>	Specify the .olb file name and path.
<b>Browse (...)</b>	Click this to browse to the location where you have the .olb file.
<b>Package Type</b>	Click this drop down list and select a package name.
<b>PCB Footprint</b>	After selecting the package name in Package Type field, the footprint name is automatically displayed in this field.
<b>Rules File</b>	<p>This option lets you browse and select a interface rules file. This option is enabled only when you select the <i>Use Existing Rules and Mapping File</i> option.</p> <p>Click the drop-down button to select the rules file. The drop-down list displays a list of rules file fetched from the PTF file and the <i>Select Other Rules File</i> option. You may select a relevant rules file names or click the <i>Select Other Rules File</i> option to select the rules file from the Libraries.</p>
<b>Mapping File</b>	<p>This option lets you browse and select a mapping file. This option is enabled only when you select the <i>Use Existing Rules and Mapping File</i> option.</p> <p>The relevant mapping file is displayed after specifying the rules file name in the <i>Rules File</i> field. The mapping files are fetched from the <i>lib:cell:view</i> and all the LRF search path directories.</p>
<b>Place</b>	Lets you place the interface rules file on the canvas.
<b>Cancel</b>	Click Cancel to abort the process and exit the dialog box.

## Auto Map FPGA Ports

The Auto Map FPGA Ports dialog box allows you to map the Device Instance port names to interface instance signal names or net names. You use regular expression options available in this dialog box to find the required signal or net names and port names for mapping. This process is especially useful when you are mapping large number of port names and signal names.

In this dialog box filtering the information is done in a unique way. The signal and port names which you don't want to map are excluded from the list using regular expression. And the displayed signal and port names are used for mapping. It is not mandatory to always use regular expression for filtering. You also have the choice to filter the information without using regular expression. The only difference between with and without regular expression is normal characters are treated as metacharacters with regular expression and as string without regular expression.

### Using Regular Expression

Regular expression are a concise and flexible notable for filtering patterns for text. The regular expressions are made up of normal characters and metacharacters. FSP provides you the option to use regular expressions and special characters to filter the information in Auto Map FPGA Ports dialog box.

**Note:** The regular expression option is also available in Auto Map Symbol Pin Name dialog box.

#### ***Normal Characters Without Regular Expressions***

Normal characters may include upper and lower case letters and digits A to Z, a to z and 0 to 9 and all characters.

**Note:** Without regular expression any normal character would be treated as string.

For example, if you enter Data\_<1> in the filter field for the *Signal Names* column and unselect the Regular Expression option, only the pins having the name Data\_<1> will be excluded from the list.

#### ***Metacharacters With Regular Expressions***

The metacharacters used with the regular expressions have special meanings and are described below.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

Metacharacter	Description
.	The period (.) matches any single character except the newline character ^. For example the regular expression r.t will match the strings rat, rut, r t, but not root.
[]	<p>Use the [ ] brackets to match any one of the characters between the brackets. For example, the regular expression r[aou]t matches rat, rot, and rut, but not ret.</p> <p>Ranges of characters can be specified by using a hyphen. For example, the regular expression [0-9] means match any digit. Multiple ranges can be specified as well. The regular expression [A-Za-z] means match any upper or lower case letter.</p> <p>To match any character except those in the range, use the caret as the first character after the opening bracket. For example, the expression [^269A-Z] will match any characters except 2, 6, 9, and upper case letters.</p>
\$	The dollar sign (\$) matches the end of a line. For example, the regular expression professional\$ will match the end of the line "He is a professional" but not the end of the line "They are professionals"
^	The caret (^) matches the beginning of a line. For example, the regular expression ^When in will match the beginning of the line "When in the course of human events" but will not match the beginning of the line "What and When in the"
*	The asterisk (*) matches zero or more occurrences of the character immediately preceding. For example, the regular expression .* matches any number of any characters. The regular expression [a-z]* matches zero or more lower-case characters.
\	A backslash (\) followed by any metacharacter matches the literal character itself. That is, the backslash "escapes" the metacharacter. For example, \\$ is used to match the dollar sign \$ rather than the end of a line. Similarly, the expression \. is used to match the period character rather than any single character.
	Matches two conditions. For example the regular expression him her matches the line "it belongs to him" and matches the line "it belongs to her" but does not match the line "it belongs to them"
?	A question mark (?) is an optional element. For example, xy?z matches either xyz or xz.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

+	Matches one or more occurrences of the character or regular expression immediately preceding. For example, the regular expression 9+ matches 9, 99, 999 and so on. The regular expression [a-z]+ matches one or more lower-case characters.
---	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

#### **How to Access**

- Click *Auto Map FPGA Ports* icon in *FPGA Port and Use Pin Mapping for DeviceInstance <Instance Name>* dialog box.

Name	Description
Patterns	Click and select any one of the option <i>Net Name</i> and <i>Signal Name</i> from drop down list. The selected option would be considered as reference column and would be used to map with FPGA port names.  <b>Note:</b> The second column header name in below is changed based on the option selected.
Exclude Patterns	Select this option to exclude the patterns.
FPGA Port Pattern	Specify the patterns as required to filter the information.
Fetch	Click to read and display all the signals.
Select	In Select column, select the check box of the signals for whom you want to map.
Signal Name/Net Name	<b>Signal Name</b>  If Signal Name is selected in <i>Pattern</i> drop down box, then this column name is visible. The signal names are displayed in this column based on the regular expression filtering.  <b>Net Name</b>  If Net Name is selected in <i>Pattern</i> drop down box, then this column name is visible. The net names are displayed in this column based on the regular expression filtering.
Check All	Click this option to select all the signals to map.
UnCheck All	Click this option to unselect all the signals.
Map	Use this option to map all the signal/pin names with FPGA ports.
Close	Click this option to exit the dialog box.

## Auto Map Symbol Pins

The Auto Map Symbol Pins dialog box allows you to map the symbol pin names or pin numbers to rules file signal names. You use regular expression options available in this dialog box to find the required symbol pin names or pin numbers and signal names for mapping. This process is especially useful when you are mapping large number of pin names and signal names.

In this dialog box filtering the information is done in a unique way. The pin and signal names which you don't want to map are excluded from the list using regular expression. And the displayed pin and signal names are used for mapping. It is not mandatory to always use regular expression for filtering. You also have the choice to filter the information without using regular expression. The only difference between with and without regular expression is normal characters are treated as metacharacters with regular expression and as string without regular expression.

For detailed information on regular expression see [Using Regular Expression](#) section.

### How to Access

- Click *Auto Map Signals* icon in Rules Signal Mapper dialog box.

Name	Description
Pattern	Click and select any one of the option <i>Symbol Pin Name</i> and <i>Pin Number</i> from drop down list. The selected option would be considered as reference column and would be used to map with rules file signal names.  <b>Note:</b> The second column header name in below is changed based on the option selected.
Signal Name Pattern	Specify the patterns as required in below text box.
Map	Select the check boxes for the signals and pins you want to map.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

Symbol Pin Name/ Pin Number	<p><b><i>Symbol Pin Name</i></b></p> <p>If Symbol Pin Name is selected in <i>Pattern</i> drop down box, then this column name is visible. The symbol pin names are displayed in this column based on the regular expression filtering.</p> <p><b><i>Pin Number</i></b></p> <p>If Pin Number is selected in <i>Pattern</i> drop down box, then this column name is visible. The pin numbers are displayed in this column based on the regular expression filtering.</p>
Check All	Click this option to select all the signals to map.
UnCheck All	Click this option to unselect all the signals.
Map	Use this option to map all the signal/pin names with FPGA ports.
Close	Click this option to exit the dialog box.

## Add Description

The Add Description dialog box lets you to add information to the component pin, group, and the component. The Add Description dialog box is available in the following dialog boxes:

- Rules Editor – In this dialog box, you can add information at pin level and component level.
- Rules Instance Editor – In this dialog box, you can add information at pin level and component level.
- Edit Group – In this dialog box, you can add information at group level.
- Add Group – In this dialog box, you can add information at group level.

### **How to Access**

- Click *browse(...)* in Part Description or Add Description dialog boxes.

Name	Description
	You can type words, special characters, numericals, or white space.
OK	Click to save the description.
Cancel	Click to quit the dialog box.

## Add Regulator from Schematic Symbol

The *Add Regulator from Schematic Symbol* dialog box lets you add a schematic symbol of a power pin as a power regulator.

### **How to Access the Add Regulator from Schematic Symbol dialog box**

- In the *Power Connections* dialog box, click the drop-down button beside the + button.
- The *Add Regulator from Schematic Symbol* option appears.
- Click the option.

Name	Description
Lib:cell:view	Click <i>browse(...)</i> to invoke Component Browser. Use Component Browser to browse and select the symbol of a power pin.
Regulator Name	Once you specify the symbol in <i>lib:cell:view</i> , the regulator name is automatically displayed in this cell; however, if the <i>symbol.css</i> file contains more than one power property name then any one regulator name is selected by default and other is displayed in the drop-down list.
Regulator Voltage	This is a read only field.  This field displays the voltage value of the regulator that is displayed in the <i>Regulator Name</i> field. If you select a power name from the drop-down list of <i>Regulator Name</i> , the corresponding voltage value is displayed in this field.
OK	Click to save the settings.
Cancel	Click to exit the dialog box without saving the settings.

## BUFR Nets for Device Instance

The BUFR Nets for Device Instance dialog box displays the number of and availability of BUFRS available in the device.

**Note:** This option appears only for Virtex 4,5 and 6 devices.

### ***How to Access the BUFR Nets for Device Instance dialog box***

- Right-click on the device and choose *Constraints – Show BUFR Nets*.

Name	Description
Available BUFRs	Shows you the current status of available BUFRs
BUFR Location	Displays the BUFR locations
Nearest CC pins	Displays the name of the CC pin nearest to the BUFR
Net Name	Displays the FSP Net name
OK	Click OK to finish the operation.

## Bank Settings for Device Instance

The *Bank Settings for Device Instance* dialog box provides you the bank details of the selected device instance. This dialog box also provides support to preserve the VREF pins and specify maximum outputs.

### **How to Access the Bank Settings for Device Instance dialog box?**

- In Design Connectivity, click on the device instance row.  
The properties of the device instance is displayed in the Properties.
- Click *Bank Settings* button.

Name	Description
Bank	Displays a list of bank numbers.
Total I/Os	Displays the total number of I/O pins of the device instance.
Available I/Os	Displays the available number of I/O pins of the device instance.  This is a read-only field. The number displayed here suggests the number of IO pins available for making the connections.
Max Outputs	Specify the maximum number of output pins that can be used on a specific bank.  The value you specify here will limit the number of pins in the bank that will be used as outputs. This option is available only for Xilinx family devices.
Preserve Pins	Select the pins that you want to preserve for the bank from the drop-down list.  FSP lets you lock dedicated pins such as VREF, VRP/VRN, and RUP/RDN.
Use Internal VREF	Select this option to use the internal VREFs.

## Column Chooser

The Column Chooser dialog box lets you hide or unhide the columns in the following editors of FSP:

- Rules Editor
- Rules Viewer
- Rules Instance Editor
- Design Connectivity

The Column Chooser dialog box lets you customize the display of columns in the above mentioned editors. The dialog box includes a list of column names that can be included in the Editor. To display any of these columns in the editor you need to select the check boxes and to hide unselect the check boxes.

The Column Chooser dialog box also provides support to arrange the column orders in the editors. The order of the columns in the editors are displayed according to the order of the column names in the Column Chooser dialog box. For example, if the Column Chooser dialog box has column names arranged as below shown:

- Pin Number
- Pin Type
- Diff.Pair Pin
- Voltage Level
- X Location
- Y Location

The *Rules Editor*, *Rules Instance Editor*, and *Design Connectivity* displays the columns in the same order as depicted below.

Pin Number	Pin Type	Diff.Pair Pin	Voltage Level	X Location	Y Location
------------	----------	---------------	---------------	------------	------------

You can also rearrange the columns order by moving the column names up and down in the Column Chooser dialog box. After you arrange the columns order by moving up and down, the columns in the Editor is re-arranged based on the order defined in the Column Chooser dialog box.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

**Note:** In the *Rules Editor* and *Rules Instance Editor* the *Pin Name* column and in *Design Connectivity* the *Status* column is considered as default column. This mean, there is no option to hide the *Pin Name* or *Status* columns and are always displayed as first column in the respective editors.

The Column Chooser dialog box in Design Connectivity provides you an additional feature, *Freezing Columns*. This option is useful, when you want to keep a column of the Design Connectivity window visible while you scroll to another column of the Design Connectivity window. You can lock specific columns that you do want to scroll left and right.

**Note:** The Freeze Column feature is not available in the Column Chooser dialog box in Rules Editor and Rules Instance Editor.

In the Column Chooser dialog box, the column names displayed above the *Freeze Column* bar are considered as freezed columns. To freeze and unfreeze you can move up and down the column names.

#### ***How to Access the Column Chooser dialog box***

- Click the *Show/Hide Column* tool bar icon.

Name	Description
Column Names	Select the check boxes to unhide and unselect to hide the columns in the editors.
Up and Down Arrow Buttons	Select the column name and click <i>Up</i> arrow button to move the column name upwards and <i>down</i> arrow button to move downwards in the list. The columns in the editors will be exactly displayed in the same order as specified in this dialog box.
Freeze Column	<b>Note:</b> This option is applicable for only Design Connectivity.  To freeze columns you can either move the column names up to the <i>Freeze Column</i> bar or drag and move the <i>Freeze Column</i> bar up and down.
Check All	Click to select all the column names.
Uncheck All	Click to unselect all the column names.
OK	Click to save the settings.

## Create Part from PCB Footprint

Use the Create Part from DRA dialog box to import the pin X and Y location details from dra file.

### **How to Access**

- In Rules Editor dialog box, click *Import From*, and select *DRA*.

Name	Description
Specify DRA File	Specify the dra file name and path to the directory where the file exists.
browse (...)	Use this option to browse to the directory where the file exists.
OK	Click OK to import the X and Y locations
Cancel	Click Cancel to exit the dialog box.

## Convert Rules File Instance to Real Part (DE-HDL)

Interface components which are placed on the canvas using Libraries are not real components. That means these components are neither binded with front-end symbol nor footprint. The Convert Rules File Instance to Real Part dialog box lets you to reselect the mapping file or create a new mapping file for the existing logical rule file. The process of conversion begins with selecting the front-end symbol and footprint from Component Browser followed by selection of logical interface model and mapping pin names in Convert Rules File Instance to Real Part dialog box.

For detailed information see [Converting Components to Real Components](#) section.

### How to Access

- Right-click on the interface instance and select *Convert to Real Interface*.  
The Component Browser is displayed.
- Select the `Library:Cell:View` and click `Select`.  
The Convert Rules File Instance to Real Part dialog box is displayed.

Name	Description
<code>Library:Cell:View</code>	Displays the selected library:cell:view names.
<code>Show PTF Attributes</code>	Click this option to select a PTF row.
<code>Rules File</code>	This is a read only field. Displays the current logical rules file name.
<code>Mapping File</code>	This option lets you browse and select a mapping file. This option is enabled only when you select the <i>Use Existing Rules and Mapping File</i> option.  The relevant mapping file is displayed after specifying the rules file name in the <i>Rules File</i> field. The mapping files are fetched from the <code>lib:cell:view</code> and all the LRF search path directories.
<code>Define Mapping</code>	This option let you define a new mapping file. The Rules Signal Mapper dialog box is displayed when you click this option.
<code>Convert</code>	Click <code>Convert</code> to complete the process.
<code>Cancel</code>	Click <code>Cancel</code> to abort the process and exit the dialog box.

## Convert Rules File Instance to Real Part (OrCAD)

When you are in OrCAD schematic environment, the Convert Rules File Instance to Real Part dialog box lets you to select .olb file and mapping file. If you do have the desired mapping file then you have the option to create a new mapping file through this dialog box.

For detailed information see [Converting Components to Real Components](#) section.

### **How to Access**

- Right-click on the interface instance and select *Convert to Real Interface*.  
The Component Browser is displayed.
- Select the `Library:Cell:View` and click `Select`.  
The Convert Rules File Instance to Real Part dialog box is displayed.

Name	Description
Library File (*.olb)	Specify the .olb file name and path.
Browse (...)	Click this to browse to the location where you have the .olb file.
Package Type	Click this drop down list and select a package name.
PCB Footprint	After selecting the package name in Package Type field, the footprint name is automatically displayed in this field.
Rules File	This is a read only field. Displays the current logical rules file name.
Mapping File	This option lets you browse and select a mapping file. This option is enabled only when you select the <i>Use Existing Rules and Mapping File</i> option.  The relevant mapping file is displayed after specifying the rules file name in the <i>Rules File</i> field. The mapping files are fetched from the <code>lib:cell:view</code> and all the LRF search path directories.
Define Mapping	This option let you define a new mapping file. The Rules Signal Mapper dialog box is displayed when you click this option.
Convert	Click this to bind the select symbol with the rules file.
Cancel	Click this to cancel the operation.

## Changing Schematic Symbol File Reference(OrCAD)

The Changing Schematic Symbol File Reference dialog box allows you to swap the referenced associated mapping file at one go after placing the component through Create/Select Component Rules and Mapping Information wizard (in OrCAD). Before specifying the schematic symbol file make sure that the schematic symbol that you are going to use has the dra associated with it.

### **How to Access**

- Right-click on the real component and choose *Rules – Change Schematic Symbol File File References*.

Name	Description
Library File (*.olb)	Specify the .olb file name and path or click <i>browse (...)</i> to browse to the directory where the olb file is located.
Change	Click Convert to complete the process.
Cancel	Click Cancel to abort the process and exit the dialog box.

## Creating Part From Custom Footprint

The Create Part from Custom Footprint dialog box enables you to decide the footprint for your part. When deciding your own custom footprint, the information needs to be specified is pin configuration parameters (for example, pin size, row pitch.. etc) and number of pins. The pin names and X, Y location values are derived from the information you provide in Create Part from Custom Footprint dialog box. The Create Part from Custom Footprint dialog box also provides you the preview of your symbol footprint.

### How to Access

- In Rules Editor, click *Import From*.
- Select *Custom Footprint* option from pop-up menu.

### Specifying Pin Configuration Parameters

If you select the BGA/Connector/PGA type of footprint you need to specify the following:

Name	Description	Notes
Staggered Rows	<ul style="list-style-type: none"><li>■ Specify if you wish rows to be staggered.</li><li>■ Select any one of the options even rows or odd rows.</li></ul>	Staggering the rows will place the pins of the staggered row in between the pins of the row above it. This helps FSP to connect the pins easily. Many available chips have staggered pin configuration.  For example Connector DDR2 DIMM.
Pin Size(D)	Specify the diameter size of the pin in terms of the unit specified (in mm or mils)	
Row Pitch(V)	Specify the distance between the center of two successive rows of pins (in mm or mils).	It should be same as, or more than the pin size.
Column Pitch(H)	Specify the distance between the center of two columns of pins. (in mm or mils)	It should be same as, or more than the pin size.
Row Count(RC)	Specify the total number of rows.	The rows are indicated alphabetically (A to Z).

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

Column Count(CC)	Specify the total number of columns.	The columns are indicated using numbers (1, 2, 3 and so on.)
Void Rows	Specify the number of rows where no pins should be inserted.	The entered number should always be an even number. Void rows are used for creating pin layouts where the pins are only around the border of the chips and not in the middle.
Void Columns	Specify the number of columns where no pins should be inserted.	The entered number should always be an even number. Void columns are used for creating pin layouts where the pins are only around the border of the chips and not in the middle.
Center Rows	Specify the number of rows in the center for which pins should be provided.	The entered number should always be an even number. Center rows are used for many chips where there are pins around the borders and in the center and there are void rows and columns in-between.
Center Columns	Specify the number of columns in the center for which pins should be provided.	The entered number should always be an even number. Center columns are used for many chips where there are pins around the borders and in the center and there are void rows and columns in between.
Pin Count	Based on the parameters specified by you in the respective fields the total number of pins is automatically calculated and displayed.	
Units	Select the units from the <i>Units</i> drop down list.	
Update Preview	Click <i>Update Preview</i> to view the footprint in the preview window.	You can also utilize the various zoom options for better preview.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

If you select the DIP/SOIC/SSOP/TSOP type of footprint you need to specify the following:

<b>Parameter</b>	<b>Description</b>	<b>Notes</b>
Units	Select the units from the <i>Units</i> drop down list.	The unit you specify here will be apply to Pin Size, Row Pitch, and Column Pitch measurements specified in the Pin Pattern group.
Pin Size (D)	Specify the size of the pin in terms of the unit specified (in mm or mil).	Pin size cannot be greater than row and pin pitch.
Pin Pitch (V)	Specify the distance between the center of two successive pins (in mm or mil).	It should be same as, or more than the pin size.
Row Pitch (H)	Specify the distance between the center of two successive rows of pins (in mm or mil).	
Pin Count	Specify the total number of pins.	The total number of pins for the TSOP type of parts should be even.
Update Preview	Click <i>Update Preview</i> to view the footprint in the preview window.	You can also utilize the various zoom options for better preview.

If you select the QFP/TQFP type of footprint you need to specify the following:

<b>Parameter</b>	<b>Description</b>	<b>Notes</b>
Units	Select the units from the <i>Units</i> drop down list.	The unit you specify here is apply to the measure of Pin Size, Row Pitch, and Column Pitch measurements specified in the Pin Pattern group.
Pin Count	Specify the details of pin count.	

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Horizontal Pins (HP)	Specify the total number of horizontal pins.	
Vertical Pins	Specify the total number of vertical pins.	
Pin Size (D)	Specify the size of the pin in terms of the unit specified (in mm or mil).	
Pin Pitch (H)	Specify the distance between the two successive pins (in mm or mil).	
Horizontal Row Pitch (HRP)	Specify the distance between the center of two successive horizontal rows of pins.	
Vertical Row Pitch (VRP)	Specify the distance between the center of two successive vertical columns of pins.	
Pin Count	Based on the parameters specified the total number of pins is automatically calculated and displayed.	
Update Preview	Click <i>Update Preview</i> to view the footprint in the preview window.	You can also utilize the various zoom options for better preview.

## Change Product Choices

The Change Product dialog box allows you to select the product whose license you want to use. Selecting product license allows you to access components and features that are not available in current product license. The Cadence product licenses available for use are displayed in this dialog box.

**Note:** To select the product, you need to close the project.

### ***How to Access the Change Product Choice dialog box***

The Change Product Choice dialog box is invoked when:

- You invoke the FSP for the first time and all subsequent invocations unless you specify the default choice.
- Close the project and choose *File - Change Product*.

### **Modifying the Default License**

To change the license, close the project and perform the following steps:

1. Choose *File – Preference*.

The Preference window is displayed.

2. Click the *Design* tab.

3. Click *Browse*.

The Cadence Product choices window is displayed.

4. Select a product from the list of choices available in the *Cadence Product Choice* dialog box.

5. Select the *Use As Default* option to invoke the selected product license every time you invoke FSP and click *OK*.

Once the license is changed, the message log window displays the successful report of changing license.

**Note:** You are not allowed to open the project with OrCAD FPGA System Planner license if the project is previously created with Allegro 4 FPGA System Planner Option (Previously Allegro FSP XL) license or higher. An error message is displayed in Messages window when you try to do this. Because number of FPGA's usage is restricted in these licenses compare to higher license.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Name	Description
Select the Product	<p>Displays the list of available Cadence Product license. Click and select the product to be used.</p> <p>You can choose any one of the following license strings:</p> <ul style="list-style-type: none"><li>■ Allegro ASIC Prototyping with FPGA's (Previously Allegro FSP GXL) Use this license to work with unlimited FPGA's.</li><li>■ Allegro 4 FPGA System Planner Option (Previously Allegro FSP XL) Use this license to work with four FPGA's or any number of FPGA's with pin limit of 4000 pins.</li><li>■ Allegro 2 FPGA System Planner Option Use this license to work with two FPGA's or any number of FPGA's with pin limit of 2000 pins.</li><li>■ Allegro FPGA System Planner - L Use this license to work with one FPGA or any number of FPGA's with pin limit of 1000 pins.</li><li>■ OrCAD FPGA System Planner Use this license to work with one FPGA or any number of FPGA's with pin limit of 1000 pins. DE-HDL schematic environment will be disabled after selecting this license.</li></ul>
Use As Default	<p>This option prevents the Cadence Product Choices dialog box from appearing every time you run FSP.</p> <p>After selecting a product, select this option.</p>

## Change Net Names

The Change Signal dialog box allows you to:

- Change the interface and device protocol names.
- Apply the net name templates.
- Specify the custom attributes to apply to the nets of the instances.

### ***How to Access the Change Signal dialog box***

- Choose *Design – Change Net Names*.

Name	Description
Start Instance	Displays the name of the instance from where the signal is originated.
Start Instance Pin	Displays the pin number of the instance from where the signal is originated.
End Instance	Displays the name of the instance where the signal is ended.
End InstancePin	Displays the pin number of the destination where the signal is ended.
Present Net Name	Displays the Present net name of the signal.
Changed Net Name	Displays the new name of the signal. The successfully changed net names appear in blue while conflicting net names appear in red.
Custom Attributes	Double click on cell and click ... button to invoke the Custom Attribute window. You can add Attribute properties in this window for more information see here.
AutoChange Bus Signal	Select this check box if you wish to automatically change all the signals present in a bus when any of them is changed manually.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Signal Name Template	<p>This feature allows you to change the net name of interfaces as well as protocols with a combination of templates for example, instance name, pin name, pin number, protocol signal name, and connected devices. You can specify them at the start, end, or middle of the net name with a separator of your choice. You can specify any keyboard character as a separator.</p> <p>The template patterns you see here appear from Project Settings. However, you can change the patterns using the Add or Delete buttons.</p> <p>An example is to change the interface signal names. You start with the instance name, end with the pin name with a separator as underscore. In the Interface tab, select Start Instance Name from the first drop-down list, Click End Pin Name from the second drop-down list, and then enter underscore (_) in the Separator text box. Click Apply to change all signal names.</p>
	Click to add one more drop-down to change the net name with combination of templates. The drop-down options are End Instance Name, End Pin Name, End Pin Number, Protocol Signal Name, Start Instance Name, Start Pin Name, Start Pin Number, and Connected Devices.
	Click to remove the selected drop-down.
Default	Click to apply the default net name template. When you invoke the change signal form first time tool display the two default net name string text boxes.
Apply	Click to update the signal names. The drop options for Apply include Apply to Both pattern, To interface signals only and To protocol signals only.
OK	Click to change the signal names.
Cancel	Click to cancel the operation.

## Custom Attribute

The Custom Attributes dialog box allows you to specify the attribute on following design entities:

Name	Description	How to Access
Device and Interface Instance Properties	specify the attribute on component pins	Click <i>Edit</i> .
Configure Pin Windows (Device/Interface)	specify the attribute on instance pins	Double-click in Custom Attribute column and click ...
Change signal	specify the attribute on nets	Double-click in Custom Attribute column and click ...
Edit Part	specify the attribute on symbol	Double-click in Custom Attribute column and click ...

Name	Description
Name	Select the attribute Name from the drop-down list.  You can create your own list of attributes by editing the <code>custom_attributes.xml</code> file.  This file is present in the <code>Template</code> directory where the project is installed.
Value	Double-click Value to enter or edit the value.
Add	Click Add the attribute to the selected pin.
Remove	Click Remove the attributes of the selected pin.
OK	Click OK to save the changes.
Cancel	Click Cancel to cancel the operation.

## Create New Protocol

The Create New Protocol dialog box allows you to create new device protocol.

### **How to Access the Create New Protocol dialog box**

- In Design Connectivity, right-click on the device instance row and choose *Protocol – Create*.

When you point to *Create*, a list of device instance names appear that are present on the Canvas. To create protocol between the selected device and any one of the listed device instances, select a instance name. Another option, *Multi Point* is available in the list. Use this option to create protocol between two or more devices.

Name	Description
Available Device List	Displays the device instance(s) present in your design.  Click to move the selected device instance from the Available Device Instance pane to the Protocol Device Instance pane.
	Click to remove the selected device instance from the Protocol Device Instance pane.
	Displays the devices which you have selected for creating the protocol.
Protocol Device List	Displays the device order of protocol.  Click to move the selected device up the order in the Protocol Device Instance pane.
	Use these arrow buttons to change the order of devices in the protocol.
	Click to move the selected device down the order in the Protocol Device Instance pane.
OK	Click to create the protocol.
Cancel	Click to cancel the operation.  Make sure the order of devices in the protocol is appropriate because you cannot change it at a latter stage.

## Create Virtual Interface(s) From Verilog/VHDL for Device Instance

The Create Virtual Interface(s) From Verilog/VHDL for Device Instance dialog box allows you to create virtual interface from Verilog/VHDL files. You can also use this dialog box to create multiple virtual interfaces with single Verilog/VHDL file. You do this by performing the following steps:

- Importing the signals from Verilog/VHDL files in this dialog box.
- Select bunch of signals to create first virtual interface.
- Create another group and select other bunch of signals to create another virtual interface.

The Create Virtual Interface(s) From Verilog/VHDL for Device dialog box is divided into two panes.

- Right Pane - Allows you to import the signals from Verilog/VHDL file.
- Left Pane - Allows you to add virtual interfaces and add, edit the pin details.

### ***How to Access the Create Virtual Interface(s) From Verilog/VHDL for Device Instance***

- Right-click on the *FPGA* and choose *Virtual Interface – Create New Virtual Interface from HDL*.

Name	Description
Left Pane	Displays the default virtual interface editor.
Right Pane	The Import Signals from Verilog/VHDL file pane is similar to Import Signals from Verilog/VHDL pane of Define Virtual Interface for Device Instance dialog box. See <a href="#">Define Virtual Interface for Device Instance</a> section.  After importing the signals, select bunch of signals and click left arrow button to move the signals to left pane.
Instance Name	Enter name for the virtual interface.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Save As	This option invokes Save Virtual External Interface File dialog box. Click <i>browse (...)</i> to browse the directory where you want to save the rules file, enter name and click <i>Save</i> . Or directly enter directory path with rules file name in <i>Rules File</i> field and click <i>OK</i> .  <b>Note:</b> A confirmation window is displayed for incomplete or incorrect details in this editor click <i>Yes</i> to move further.
Save	Use this option to save the rules file in \$projectdirectory/temp/ruleseditor/<lrf_name>.
Validate	Click this option to verify the virtual interface pin details.
Show Log/Hide Log	Click <i>Show Log</i> to see the log window or <i>Hide Log</i> to hide the log window.
OK	Click OK to save the virtual interface.

## Change FPGA

The Change FPGA dialog box allows you to dynamically change the device rules file with a new device rules file. The device rules file you place on the canvas through Libraries can be replaced at any time during the design without disturbing the interface settings except the Use Bank. The Use Bank settings is wiped out when you replace the device rules file. You are allowed to replace the current device rules file from the device rules files of same family.

**Note:** The device you place on the canvas through Component Browser can also be replaced. You will be prompted with a confirmation window before replacing since it is linked with front-end symbol and footprint there might be chances in breakage of links at the time of replacing. So you must link the device and symbol again.

### How to Access

- Right-click on the device instance and select *Change FPGA*.

The *Change <Instance\_name>* dialog box is displayed with a list of device names of the device family to which the current device belongs to.

Name	Description
Change	Select a device name and click Change to replace the current device on canvas with selected device.
Cancel	Click cancel to exit the operation.

## Create New Project

The Create New Project dialog box allows you to create a new FSP project.

### **How to Access the Create New Project dialog box**

- Choose *File - New*.

<b>Fields &amp; Buttons</b>	<b>Description</b>
Schematics Environment	Select an environment you want to work.  After clicking <i>OK</i> rest of the remaining environment feature's will be disabled. For example if you select Allegro option, <i>Generate symbols and schematics</i> for both OrCAD will be disabled.
	The following field appears when you select <i>DE-HDL</i> in the Schematics Environment field.
Design Name	Enter the name for the new project.
Select Project Directory	Type the complete path of the folder in which you want to create the new project.  Or  Click <i>browse(...)</i> , select a folder in <i>Choose Folder Path</i> dialog box, and then click <i>OK</i> .
FSP Database Location	This is a read-only field.  Displays the relative path of the folder followed by <design_name>.fsp. <design_name> is the name you enter in the <i>Design Name</i> field.  <b>Note:</b> When you type/modify a name in the <i>Design Name</i> field, the changes immediately appear in this field.
Generate Symbols in	This field lets you specify the library name in which you want to generate the symbols of FSP components.  By default, fsp_fe_lib library name is displayed. You can also choose an existing library names from the drop-down list or type a new one.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

Design Library	This field represents working directory name. All the design data, FSP project file, and view-related log files are stored in this directory.  By default, <design_name>.lib name appears in this field. You can also choose an existing names from the drop-down list or type a new one.
The following fields appears when you select OrCAD as schematic environment.	
Design Name	Enter the name for the new project.
Directory Path	Type the complete path of the folder in which you want to create the new project.  Or  Click <i>browse(...)</i> , select a folder in <i>Choose Folder Path</i> dialog box, and then click <i>OK</i> .
Configuration File	This field lets you specify the capture.ini file. This file will be used for schematic generation.  By default, <release_name>/tools/fsp/templates/orcad/capture.ini.olib appears. You can also specify your own capture.ini.
Cancel	Click to exit.

---

## Connector Net Name and Pin Assignment Mapping for Instance

This dialog box lets you perform mapping between net names and comma separated value(CSV) nets. You use this dialog box when you try to optimize the connector connections by importing the CSV file. However the net names of the CSV file do not match with the existing connector nets or power regulators names in the design.

After you complete the FSP and CSV file nets mapping, you can synthesize the design to make connections according to the new mapping.

### ***How to Access the Map Connector Pin Assignments dialog box***

- Right-click on the connector and select *Map Pin Assignments*.  
The *Select CSV File* dialog box is displayed.
- Browse to the file, select the file, and click *Open*.  
The *Import from CSV* dialog box is displayed.  
For more information on the GUI of *Import from CSV* dialog box, see the [Import from CSV](#) section.
- Click *OK* to import the pin details.  
The *Connector Net Name and Pin Assignment Mapping for Instance* dialog box is displayed.

The following table provides the description of the panes of the Connector Net Name and Pin Assignment Mapping for Instance dialog box.

Left Hand Side Pane	Right Hand Side Pane
---------------------	----------------------

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

The interfaces and device protocols that are connected to the connector are displayed in the tree view hierarchy.	After importing the CSV file, the content are displayed in the tree view hierarchy.
The first-level node is the device protocol, which displays the list of device protocol names connected to the connector. You can click + to expand and view the list in detail.	
The second-level node is the power regulator, which displays the connector pins that are connected to power regulators. Click + to expand and view the list in detail.	
The third-level node is the interface level, which displays the list of interface signal names that are connected to the connector. Click + to expand and view the list in detail.	
The fourth-level node is the extern, which displays the list of connector pins which are marked as external signals.	

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

<p>The left hand side pane is further organized into four columns:</p> <ul style="list-style-type: none"><li>■ Pin/Port Name: Displays the signal names of the device, interface, and power regulator that are connected to the current connector.</li><li>■ Group Name: Displays the group names of the device protocol and interface.</li><li>■ Net/Regulator: Displays the net names of the device protocol and interface, and regulator names.</li><li>■ Connected/Assigned to Pin: Displays the list of pin numbers of the connector, that are connected.</li></ul> <p>Initially the <i>Net/Regulator</i> and <i>Connected/Assigned to Pin</i> columns entries are displayed in grey color.</p> <p>These entries are displayed in black color, when you perform mapping by dragging entries from the RHS pane and drop to the LHS pane. The change in the color indicates successful mapping.</p> <p>These entries are displayed in yellow color, if any conflict occurs after mapping.</p>	<p>The right hand side pane is organized into two columns:</p> <ul style="list-style-type: none"><li>■ Net/Regulator: Displays the net and regulator names available in the CSV file.</li><li>■ Pin Numbers: Displays the list of pin numbers.</li></ul> <p>You use these column entries to map with the <i>Net/Regulator</i> and <i>Pin Number</i> column entries of the LHS pane. To perform mapping, you can choose for drag and drop method or choose any other automap options available in the dialog box.</p> <p>Initially entries are displayed in grey color. These entries are displayed in black color, after mapping.</p>
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## Filtering the Display of Entries

The Connector Net Name and Pin Assignment Mapping for Instance dialog box lets you provides a filter row that lets you use wildcard characters and regular expressions to filter the display of information in the two panes. Filtering the display of information lets you focus on working on a specific set of nets in the design. For example, you can filter the display of information in the Device Instance Nets Details pane to display a specific set of nets. This lets you focus on mapping the device nets with the connector nets. This is especially useful when you are mapping on a large signals-count.

For detailed information on the regular expression, see the [Using Regular Expression](#) section.

The following table describes the options available in the dialog box.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

Name	Description
Map	Select an option from the drop-down list on which you want to perform mapping. For example, select <i>Net/Regulator and Pin Number</i> to map regulator or net name.
Automap Bus Signals	Select this option to automatically map all the bits of a bus when you map a single bit.  Select this option and drag a single bit from the <i>Net/Regulator</i> column of the RHS pane and drop to the <i>Net/Regulator</i> column of the LHS pane.
Hide After Mapping	Lets you hide the entries in the RHS pane after mapping.  Select this option and perform mapping.
Show Log	Click to display the Log window in the bottom of the dialog box.
OK	Click to save the mapping.
Cancel	Click to exit the dialog box without saving the mapping.

## Create Target Sets

Use this dialog box to create group sets.

### **How to Access**

- Choose *Tools – Create Target Sets*

Name	Description
Target Sets	By default this pane is empty. The target sets names appears in this pane.
Add	Click to create a new target set.
Delete	Click to delete an existing target set.
Available Connectors	Displays a list of connector names that are available to include in the target set.
Selected Connectors	Displays a list of connector names that have been selected and used in a target set.
Save	Click to save the settings.
Close	Click to exit.

## DCI Master Slave Banks

The *DCI Master Slave Banks* dialog box provides you a post-synthesis list of master and slave banks, and the column numbers in which the master and slave banks reside. You can change the master bank selection using this dialog box. After selecting, the changes are reflected in the *Pin Properties for Device Instance* dialog box.

### **How to Access the DCI Master Slave Banks**

- In Design Connectivity, click on the device instance row. The device instance properties are displayed in the *Properties* dialog box.
- Select the *Use DCI Cascading* option. After you select the *Use DCI Cascading* option, the Master Slave Banks option is enabled.
- Click the *Master Slave Banks* option.

Name	Description
Master Banks	Displays the list of master banks available in the device. Click and select a master bank from the drop-down list.
Slave Banks	Displays the slave bank names.
Column Number	Displays the column numbers in which the master and slave bank reside.

## Define Termination

The Define Termination dialog box allows to define different termination to be used in your design. Once the termination is defined, you can add the termination circuitry on several nets quickly and consistently. The two common types of termination available in the Define Termination dialog box are:

- Series Termination
- Split Termination

For detailed information on types of termination, see [Working with Associated Components](#) chapter.

### ***How to Access the Define Termination dialog box***

- Choose *Tools – Termination and Power Filters*.

Name	Description
Add	Click to add a new empty row in the dialog box.
Name	Enter a name for the termination to be used in the design.
Termination Type	Displays the available termination types. Select a termination type and click <i>OK</i> .  FPGA System Planner displays a thumbnail view while browsing through different termination types.
Diff.Termination	Displays a list of predefined termination names.  Use this option to apply termination on differential pair pins. For example, if you apply a series termination on P type pin then the same termination type is automatically assigned to the N type pin.  Points to remember: <ul style="list-style-type: none"><li>■ This cell is disabled when a differential termination type is selected in <i>Termination Type</i> column.</li><li>■ To apply a single ended series termination such as <i>series</i> to differential pair pins you must select the pre-defined series termination name from this drop-down menu.</li></ul>

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Termination Mapping	Invokes the <i>Define Termination Mapping</i> dialog box where you can define and associate the discrete components such as resistor, capacitor and diodes using <i>Component Browser</i> that will be used in adding terminations.  Click <i>browse (...)</i> to invoke the <i>Define Termination Mapping</i> dialog box. For more information, see <a href="#">Define Termination Mapping</a> section.
Delete	Click to delete the termination type entry or row from the list.
Reset Mapping	Lets you reset the termination type port mapping.  Select a termination name in <i>Name</i> column and click <i>Reset Mapping</i> . The details displayed in <i>Termination Mapping</i> column and the values defined in <i>Define Termination Mapping</i> dialog box are erased.
OK	Click OK to save the settings.
Cancel	Click to Cancel the operation.

## Deep and Wide Groups

The Deep and Wide Groups dialog box lets you to define the connections between interfaces and bind them into a larger component.

### **Example**

You can create an x32 bit DDR2 interface using four x8 bit components by defining the required Deep and Wide groups.

### ***How to Access the Deep and Wide Groups dialog box***

- Choose *Tools – Deep and Wide Groups*.

Name	Description	Remark
Deep and Wide Groups		
Create Common Group	Click to create the common group.	
Create wide buses	Click to create a wide buses.	
Remove Group	Click to remove the selected common group or wide buses.	
Select Interface	Click to select interfaces for the selected Deep and Wide group.  For a wide bus, you can select any interface more than once.	
Use Identical Pins	Select this check box to use the same set of pins for other interfaces also.  This check box is disabled if the selected interfaces are not homogeneous or if there is a repetition in the list of selected interfaces (in case of wide buses).	
...	Click to pop a sub-pane for the selection of a common pin.	

## **Allegro FPGA System Planner User Guide**

### Dialog Box Descriptions

---

Select Pins for Interface <instance name>	This dialog box allows you to select the interface instance pins that you want to make it common.	
Available Pins	Displays the list of pins available in interface instance.	
Selected Pins	Displays the list of pins that you have selected to make them common pins.	
Expand Bus Pins	Select this option to view the complete bus pins.	
Group Name	Enter a name for the wide bus.  This option will appear only for wide buses.	
Save	Click to apply the changes to Deep and Wide groups.	
Reset	Click to Reset all the fields.	
Close	Click to quit the wizard.	

## Define Termination Mapping

The Define Termination Mapping dialog box lets you add and associate discrete components such as resistor, capacitor in defined termination block using Component Browser.

### **How to Access the Define Termination Mapping dialog box**

- In the Define Termination dialog box, click *browse (...)* in the Termination Mapping column.

#### **For DE-HDL**

Name	Description
Library:Cell:View	Specify the library, cell and view name or click <i>browse (...)</i> to invoke Component Browser for symbol selection.
P1,P2	Click and select the ports to map. For example, if you have selected a two port primitive component (port A and B), click <i>P1</i> and select <i>A</i> and click <i>P2</i> and select <i>B</i> .
OK	Click OK to save the settings.

#### **For OrCAD**

Name	Description
Library File (*.OLB)	Click <i>browse (...)</i> to specify an .olb file.
Package Type	Select a package type name from the drop-down menu.
PCB Footprint	Displays the footprint name in read-only mode.
P1,P2	Click and select the ports to map. For example, if you have selected a two port discrete component (port A and B), click <i>P1</i> and select <i>A</i> and click <i>P2</i> and select <i>B</i> .
Add	Click to add an empty row in bottom pane.
Property	Enter a property name.
Value	Enter a value for defined property.

## **Allegro FPGA System Planner User Guide**

### Dialog Box Descriptions

---

Delete	Select a row in bottom pane and click <i>Delete</i> to delete the selected row.
OK	Click OK to save the settings.

## Define PROM Chain

The Define PROM Chain dialog box allows you to define PROM chain.

**Note:** You must have atleast one PROM component on canvas to define PROM chain.

### ***How to Access the Define PROM Chain dialog box***

- Choose *Tools - PROM Configuration*.

Name	Description
Add	Use this option to add a new PROM chain. When you click <i>Add</i> , the Add PROM Chain dialog box is displayed.
Delete	Use this option to delete PROM chain entry.
Edit	Use this option to edit the existing PROM chain. When you click <i>Edit</i> , the Edit PROM Chain dialog box is displayed.
Chain Name	After creating a PROM chain, PROM chain name will be displayed in this column.
Chain Configuration	After creating a PROM chain, the PROM chain settings will be displayed in this column.
Add PROM Chain	Use the Add PROM Chain dialog box to create a new PROM chain.
PROM Chain Name	Enter the PROM chain name.
Available PROMS	Display the available PROMS on your canvas.
Selected PROMS	Display PROMs that will be used for the present connections.
Left Arrow Button	Click to move the instance under the Selected PROMs/Device section.
Right Arrow Button	Click to move the instance back to the Available List.
Up Arrow Button	Click to move the instance up. Master/Slave position of the device is decided based on this order.
Down Arrow Button	Click to move the instance down. Master/Slave position of the device is decided based on this order.

## **Allegro FPGA System Planner User Guide**

### Dialog Box Descriptions

---

Select Model	Select an appropriate mode for your design.
Select Configuration	Select an appropriate configuration scheme for the connection Not Applicable for the Select Map mode.

## Define JTAG Chain

The Define JTAG Chain dialog box allows you to define JTAG chain.

### ***How to Access the Define JTAG Chain***

- Choose *Tools – JTAG Chain*.

Name	Description
Available Instances	Display the list of devices present on your canvas
JTAG Chain	Devices that you will use for your connection are moved in this field.
Add	Click to Add a new JTAG chain group. You can also do that by a Right click.
Remove	Click to Remove the JTAG group.
Complete JTAG Chain	Select this check to enable the complete JTAG chain. The TDO pin of the last FPGA is connected to the TDO pin of the JTAG connector.
Up and Down arrow	Click to move the selected interface up/down the order within the group. This is used to set the order between interfaces.

## Define Virtual Interface for Device Instance

The Define Virtual Interface for Device Instance browser allows you to create or edit the virtual interface. To create the virtual interface you need to add the pins with the details of logical constraints. After completing this process, a dummy interface is displayed on FSP canvas. You can use the virtual interface to connect with the FPGA.

**Note:**

- The virtual interface is not saved in the FSP library.
- While taking the design to DE-HDL, the virtual interface gets disappear. Only FPGA along with the nets is displayed.
- The Name of the Edit Virtual Interface <Virtual\_Instance\_Name> for Device Instance <Instance\_Name> editor is similar to the Define Virtual Interface for Device Instance. Therefore, you can refer this section for more information on the Name of the Edit Virtual Interface for Device Instance editor.

***How to Access the Define Virtual Interface for Device Instance dialog box***

- Right-click on the *FPGA* and choose *Virtual Interface – New Virtual Interface From HDL*.

Name	Description
<b>Note:</b> The Define Virtual Interface for Device Instance dialog box is similar to Edit Protocol dialog box. For more information on columns see <a href="#">Edit Protocol</a> section.	

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Import Signals From..	Displays five menu options: <ul style="list-style-type: none"><li>■ Existing Virtual Interface Use this option to import the signals from existing virtual interface file.</li><li>■ Library Part Model Use this option to import the signals from existing FSP library part model.</li><li>■ Constraint/Pinouts File Use this option to import the signals from Verilog/VHDL and constraint files.</li><li>■ DE-HDL Symbol Use this option to import the signals from chips.prt and .dra file.</li><li>■ OrCAD Symbol Use this option to import the signals from .olb and .dra file.</li></ul>
Save	Click this option if you wish to save the virtual interface in current project directory.
Check	Use this option to verify the virtual interface pin details.
OK	Click OK to save the virtual interface.

### Searching and Filtering in the Rules Editor.

The Find feature lets you to find any keywords and strings in the any of the editors of FSP. You may also enter any search string, including wild cards '\*' and '^' as search criteria. Many filtering options are also provided in the Find feature to limit the search. You can also specify the forward and backward directions of the search.

**Note:** The *Find* feature is available in the *Rules Editor*, *Rules Instance Editor*, and *FPGA Port and Use Pin Mapping for Device Instance* dialog boxes.

### How to Access

- Click *Find* icon or press Ctrl + F.

Name	Description...
Search text box	Enter the string that you are searching for.
Look In Column	Click and select the column names from the drop down list in which you want to search keyword.
Regular Expression	Select this option to apply regular expression search operation.
Find	Click to start the search operation.

### Examples of Find

This section shows few examples to perform search using the Find feature using Regular Expression.

#### ***Example1***

To search the strings that starts with IO\_, such as IO\_L7N\_A4\_D20\_1, IO\_L7P\_A5\_D21\_1 and more, you can use the either regex IO\_\w+\_d\$ or IO\_.

'IO\_ ensures that the strings must start with IO\_.

\w+ subpattern matches all the alphabetic character and \_ character, such as L7N\_A4\_D20, L7P\_A5, D21\_.

\d\$ subpattern matches the numeric character (\d) and \$ matches the numeric character at the end of the string, such as 1.

#### ***Example2***

To search the strings that contains special characters such as PD11\_RXENB\_MII-TX-ERR, you can use the regex PD\d+\_.\*

'PD' ensures that the strings must start with PD\_.

'\d+' subpattern matches numeric character after PD, such as PD11, PD22 and more.

.\* characters matches any character expect newline character zero or more times, RXENB\_MII-TX-ERR.

#### ***Filtering the Display of Information***

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

The Rules Editor provides a filter bar that lets you to filter the display of information shown in the Rules Editor.

The *Filter* bar is regular expression enabled. You can use the alpha-numeric characters, underscore (\_), question marks, or asterisks to filter the display of information. You must select the *Reg Exp* option in the *Filter* bar to enable the regular expression.

The *Filter* bar provides three drop-down combo boxes that shows all the valid values and column names which you can use to filter out the information.

The following table describes the usage of each drop-down combo boxes:

- 
- 1 Enter the text that need to be matched and displayed in the editor.  
You can also use the alpa-numeric characters, underscore(\_), asterisks (\*) to perform regular expression based search operation. To enable the regular expression, select *Reg Ex* option.
  - 2 Select a column name from the drop-down list in which you want to search the desired text.  
For example, type DDR\_CS in the search text box and select *Pin Name* from this drop-down list, FSP displays only those rows whose *Pin Name* column cells contains DDR\_CS text.
  - 3 Select this option to enable the regular expression based search.
-

## Define Decoupling Capacitors

The Define Decoupling Capacitor Symbol dialog box provides support to define decoupling capacitors for the power signals in the design. The dialog box displays a list of instance names with connected power regulator name and count in tree view structure. To define a decoupling capacitor for a power signal, select a row from the list and click *Add*. Before defining the decoupling capacitors, make sure that power regulators are defined and mapped in the design.

### **How to Access**

- Choose *Tools – Decoupling Capacitor*.

Name	Description
Decap Symbol	Displays a list of FPGA instance names with connected regulator names.
Pin Count	Displays the connected power pins count.
Decap Value	Displays the value of the decoupling capacitor, after defining in the Define Decoupling Capacitor Symbol dialog box.
Decap Count	Displays the number of capacitors defined in the Define Decoupling Capacitor Symbol dialog box.
Add	Invokes the Define Decoupling Capacitor Symbol dialog box. This option is enabled when you select a row with power regulator name and count specified.
Delete	Removes the Decap Value and Decap Count column values from the row. Select a row and click <i>Delete</i> . This option is enabled when you select a row with Decap Value and Decap Count column values defined.
Modify	Invokes the Define Decoupling Capacitor Symbol dialog box. This option helps you to modify the defined decap value and count. This option is enabled when you select a row with power regulator name and count specified.
OK	Click OK to save the settings.

## Define Decoupling Capacitor Symbol

The Define Decoupling Capacitor Symbol dialog box lets you specify a capacitor symbol between Supply (High) and Ground (Low). You can also map the ports of the selected symbol with Supply and Ground terminals.

### ***How to Access the Define Decoupling Capacitor Symbol***

- In the Define Capacitor Symbol dialog box, select a row and click *Add*.

Name	Description
HIGH	Displays the selected power regulator name.
Low	Displays the GND name.
Library:Cell:View	Specify the library, cell and view name or click <i>browse (...)</i> to invoke Component Browser for symbol selection.
P1,P2	Click and select the ports to map.  For example, if you have selected a two port primitive component (port A and B), click <i>P1</i> and select <i>A</i> and click <i>P2</i> and select <i>B</i> .
Count	Enter the number of capacitors value that you want to apply to Supply and Ground pins.  For example, if you wish to map capacitor component to ten Supply and Ground pins then you should enter ten.
OK	Click OK to save the settings.

## Design Comparison

The Design Comparison dialog box provides you a sophisticated difference reporting and merging capabilities between Master FSP database and copy of FSP database. The Design Comparison dialog box is useful when you have chosen a two database approach during the FSP - Allegro Integration flow. During the flow, you create a copy of the FSP database, associate it with the Allegro database to make changes and use the *Design Comparison* dialog box to backannotate changes into the original FSP database. You import both FSP database files into the Design Comparison dialog box to differentiate and merge changes in the original FSP database.

**Note:** By default differences are displayed after importing the database files in the *Design Comparison* dialog box.

The Design Comparison provides support to merge the following changes into FSP:

- Netlist
- Placement
- NetGroups

The Design Comparison does not provide support to merge the following changes into FSP:

- Nets added or deleted
- Termination definition and mapping
- Power regulator definition and mapping
- New component added or removed

For complete information on the flow and tasks involved in FSP - Allegro Integration flow, see the [FSP – DE-HDL flow guide](#).

### How to Access

- Choose *File – Design Compare* or click the *Design Compare* icon in toolbar.

Name	Description
Design 1	Using the <i>browse (...)</i> , specify the path and project file (.fsp) where FSP design file1 is located.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

Design 2	Using the <i>browse (...)</i> , specify the path and project file (.fsp) where FSP design file2 is located.
Design 1 ( <i>Save</i> button)	Lets you save the changes in design 1, after merging the changes from design 2 to design 1 file.  This option is enabled when you merge the changes in the design 1.
Design 2( <i>Save</i> button)	Lets you save the changes in design 2, after merging the changes from design 1 to design 2 file.  This option is enabled when you merge the changes in the design 2.
Design Compare	Displays the content of the design files along with the differences in their respective panes.
Drop down menu	Do the following: <ul style="list-style-type: none"> <li>■ Select <i>Connectivity</i> to display only net lists in the pane.</li> <li>■ Select <i>Placement/Refdes</i> to display placement and reference designator information.</li> <li>■ Select <i>Net Groups</i> to display the net groups property.</li> </ul>
Show Log	Click to view the Log window.
Mergible Differences	Click the color button to browse and select a different color for the mergible differences.
Non-Mergible Differences.	Click the color button to browse and select a different color for the non-mergible differences.
Close	Click <i>Close</i> to exit.
The following toolbar options are useful while working compairing and merging database files.	
Show Only Diff	By default this option is selected when you click <i>Compare</i> . This signify only differences are being displayed in the dialog box.  Uncheck this option to see the non-difference signals.
Show Next Diff	Lets you locate the next difference in the rows. Select a row and click this option.
Show Previous Diff	Lets you located the previous difference in the rows. Select a row and click this option.
Move Left	Lets you move a signal or group of signals from <i>Left</i> pane to the <i>Right</i> Pane. Use this option when you want to merge a signal(s) from <i>Left</i> pane to the <i>Right</i> pane.

## **Allegro FPGA System Planner User Guide**

### Dialog Box Descriptions

---

Move Right	Lets you move a signal or group of signals from <i>Right</i> pane to the <i>Left</i> Pane. Use this option when you want to merge a signal(s) from <i>Right</i> pane to the <i>Left</i> pane.
Refresh	Lets you refresh the entries any time in both the panes to reflect the latest changes made by you, if any.
Undo	Use this option to restore the last change that existed before the last edit.
Redo	Use this option to reverse the effect of the most recent Undo.

## Design Block Full Symbol Editor

Use this dialog box to modify the port directions of the full symbol.

### **How to Access**

- Right-click on the schematic row and choose Edit Pin Direction Settings.

Name	Description
schematic	Right-click and choose Edit Pin Direction Settings. The Edit Pin Direction Settings dialog box appears. Use this dialog box to modify the port directions.
symbol symbol_name<>...	Displays the symbol information such as symbol name, height and width. Click > to expand the list. Displays the list of ports that belongs to the current symbol. You can also change the port direction manually. In the Pin Direction column, click and select an option from the drop-down list.
Generate	Click to output the settings to an external file.
Cancel	Click to exit.

## Design Block Split Symbol Editor

Use this dialog box to split a large design block symbol into multiple symbols. You can perform the following using this dialog box:

- Create split symbols
- Merge split symbols
- Edit port direction

### **How to Access**

- Choose *Generate – Edit Design Block Symbol – Splits*

Name	Description
schematic	Right-click and choose Edit Pin Direction Settings. The Edit Pin Direction Settings dialog box appears. Use this dialog box to modify the port directions.
symbol symbol_name<>...	Displays the symbol information such as symbol name, height and width. Click > to expand the list. Displays the list of ports that belongs to the current symbol. You can also change the port direction manually. In the Pin Direction column, click and select an option from the drop-down list.
Preserve Graphics	Use this option to retain the splits and their ports that already exists on the graphics for the symbols when regenerating the symbols and schematics.
Generate	Click to output the settings to an external file.
Cancel	Click to exit.

## Edit Rules (for OrCAD)

Use this dialog box to edit the existing real component. The process comprises of reselecting rules file, mapping file and redefining the mapping file steps if needed.

### **How to Access**

- Choose *Edit Part - Linked to OrCAD Symbol*.

Name	Description
Library File (*.OLB)	Specify the .OLB file name and path of the directory where the file exists.  ...
Package Type	Click the drop down list to select the package type.
PCB Footprint	Specify the footprint name.
Rules File	This option lets you browse and select a interface rules file. This option is enabled only when you select the <i>Use Existing Rules and Mapping File</i> option.  Click the drop-down button to select the rules file. The drop-down list displays a list of rules file fetched from the PTF file and the <i>Select Other Rules File</i> option. You may select a relevant rules file names or click the <i>Select Other Rules File</i> option to select the rules file from the Libraries.
Mapping File	This option lets you browse and select a mapping file. This option is enabled only when you select the <i>Use Existing Rules and Mapping File</i> option.  The relevant mapping file is displayed after specifying the rules file name in the <i>Rules File</i> field. The mapping files are fetched from the <i>lib:cell:view</i> and all the LRF search path directories.
Define Mapping	This option is enabled when you specify the rule file and mapping file in the respective text boxes.  Click this option to invoke Mapping Editor dialog box. Use this dialog box to set-up mapping between signal names and symbol pin names.
Edit	Click to invoke Rules Editor dialog box. Use this dialog box to edit the existing pin details.

## **Allegro FPGA System Planner User Guide**

### Dialog Box Descriptions

---

Cancel

Click Cancel to exit dialog box.

## Edit Rules (for DE-HDL)

Use this dialog box create a new mapping file or modify the existing mapping between the pin names and signal names.

### **How to Access**

- Choose *Edit Part - Linked to DEHDL Symbol*.

Name	Description
Library:Cell:View	Specify the library name:cell name:view name. For example, fsp_fe_lib:s2_cy7c1418av18__v4__v5:sym_1.
Show PTF Attributes	Click to select the PTF row.
Rules File	This option lets you browse and select a interface rules file. This option is enabled only when you select the <i>Use Existing Rules and Mapping File</i> option.  Click the drop-down button to select the rules file. The drop-down list displays a list of rules file fetched from the PTF file and the <i>Select Other Rules File</i> option. You may select a relevant rules file names or click the <i>Select Other Rules File</i> option to select the rules file from the Libraries.
Mapping File	This option lets you browse and select a mapping file. This option is enabled only when you select the <i>Use Existing Rules and Mapping File</i> option.  The relevant mapping file is displayed after specifying the rules file name in the <i>Rules File</i> field. The mapping files are fetched from the <i>lib:cell:view</i> and all the LRF search path directories.
Define Mapping	This option is enabled when you specify the rule file and mapping file in the respective text boxes.  Click this option to invoke Mapping Editor dialog box. Use this dialog box to set-up mapping between signal names and symbol pin names.
Edit	Click to invoke Rules Editor dialog box. Use this dialog box to edit the existing pin details.
Cancel	Click Cancel to exit dialog box.

## Edit Clock Group

The Edit Clock Group dialog box allows you to modify the clock group column in Edit Protocol and Rules Editor.

**Note:** The Edit Clock Group option is available for only spartan 3/3A devices.

### ***How to Access the Edit Clock Group dialog box***

- Right-click first level node and select *Define Clock Group*.

Name	Description
Clock Group	Enter a name for the Clock group.
Map To Design Clock	Select the Design Clock Group.  You will see the clock group names, if you have defined the Design Clock groups in Settings>Design Clock pane.
Add	Click Add to add a new row.
Delete	Click Delete to delete a row.
OK	Click to add the group.
Close	Click to cancel the operation.

## Edit Group

The Edit Group dialog box lets you modify the group in Rules Editor.

### ***How to Access the Edit Group dialog box***

- Right-click on a group and select *Edit Group*.

Name	Description
Group Constraint	Select the group constraint from the drop-down list.  For example, if you select Same Bank, all pins of the group will get connected to a single bank in the targeted FPGA.
Group Name	Enter the name of the group.
Group Color	Select the color for the group.  On the design board canvas view, pins in the group appear in the color that you set here. It helps to differentiate multiple groups.
Group Use Bank	Click and select the bank names from the list to which you want to connect.
Group Don't Use Banks	Click and select the bank names that you want to preserve or exclude from connections.
Group Contiguous Pins	Click and select the contiguous pins.
Group Description	Enter description for the group.
OK	Click OK to save the settings.

## Edit Bank

The Edit Bank lets you to edit the details of the bank in the Rules Editor for connector models.

### ***How to Access the Edit Bank dialog box***

- Choose *Library – Create Part – Connector*.  
The Rules Editor dialog box is displayed.
- Right-click on the first level node or click the *Edit Bank* toolbar icon.

Name	Description
Bank Name	Click and modify the existing bank name.
Bank Color	Click and reselect the color for the bank from <i>Select Color</i> dialog box.
Bank Type	Click and reselect the type of bank you want to create from the drop down list.  For example,  Select <i>Normal Bank</i> option to create a normal bank for connector model.
Bank Description	Modify the description for the bank.

## Edit Properties

The Edit Properties dialog box allows you to edit the FSP model properties in Rules Editor.

### **How to Access the Edit Properties dialog box**

- In Rules Editor, right-click first level node and select *Edit Properties*.

Name	Description
Interface Type	<p>Click and select the type of interface model you are creating from the drop down list.</p> <p>For example,</p> <p>Select <i>Configuration</i> option to create configuration model.</p> <p><b>Note:</b> This option is not available when you are creating a connector model in Rules Editor.</p>
Family to Connect	<p>Click and select the FPGA family name from the list that you are targeting the model.</p> <p>For example,</p> <p>Select <i>V6</i> to target or connect the model to Virtex 6 devices only.</p> <p><b>Note:</b> This option is not available when you are creating a connector model in Rules Editor.</p>
Ref. Des. Prefix	<p>Enter the instance prefix name for the model.</p> <p>For example,</p> <p>For interface model use U, connectors J and dedicated connectors use XP.</p>
Pin Size	Enter a size for the pin.
Part Unit	Select the unit of measurement for the part as <i>mm,mils,micron,inch</i> and <i>cm</i> .
JDEC_TYPE	Enter JDEC type name.
Part Description	Enter description for the part.
OK	Click to save the properties.
Cancel	Click to cancel the operation.

## Edit Interface Protocol

The Edit Interface protocol dialog box allows you to edit the multi device connection groups. For example, when you are connecting an interface to multiple device you can modify the interface protocol using this dialog box.

### **How to Access the Edit Interface Protocol dialog box**

- In Design Connectivity, click on the interface row.  
The interface properties are displayed in the *Properties*.
- Click *Bank Settings*.  
The *Group Settings for Interface Instance <inst\_name>* dialog box is displayed.
- Under Connect to Device column, click the drop-down button and select two device instance names and click *OK*.  
The Multiple Device Connection Groups pane is displayed at the bottom of the window.
- Select a row and click *Edit*.  
The *Edit Interface Protocol* dialog box is displayed.

Name	Description
Edit Interface Protocol	
Group	This column displays the groups and their colors.  The group appears in the color displayed on the design board Canvas view.
Contiguous Pins	Select the contiguous pins of the group.  If the targeted device is from the Xilinx family, only the contiguous pins may be selected.  Click the drop-down arrow to select the group's contiguous pins. They will then be connected in contiguous fashion to the targeted device instance.

## **Allegro FPGA System Planner User Guide**

### Dialog Box Descriptions

---

Use Banks	Select the banks for the targeted device instances to connect to the respective group  Click the drop-down arrow to select the banks of the device. FSP shows the preview of the selected banks. The group is then connected to the banks you have targeted here.
OK	Click to set the interface properties.
Cancel	Click to cancel the operation.

## Edit Protocol

The Edit Protocol dialog box lets you:

- Create a new device protocol between two devices or more.
- Modify the definition of the existing device protocol.

### ***How to Access the Edit Protocol dialog box***

Perform the following steps to invoke the Edit Protocol editor for creating new device protocol:

1. Right-click on the device instance and choose *Protocol – Create*.  
A list of device instance names is displayed which are present on the canvas.
2. Select *<Instance Name>*.

Or

1. Right-click device instance and click *<protocol name>* in Devices to Connect (Protocols) pane.
2. Click *Create*.

Perform the following steps to invoke the Edit Protocol editor for editing the existing protocol definitions:

1. Right-click on the device instance and choose *Protocol – Edit*.
2. A list of device protocol names of the current design is displayed.
3. Select *<Device Protocol Name>*.

Or

1. Right-click device instance and click *<protocol name>* in Devices to Connect (Protocols) pane.
2. Click *Edit*.

Name	Description
Signal Name	Displays the pin name.
Pin Type	Displays the pin types as InOut, Supply, Input, and NoConnect. The pin type indicates the signal direction for the respective device.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

On Chip Termination	Double-click to select the pin termination property.
Target Pin Property	<p>This field displays the pin termination supported by the subsequent device in the protocol.</p> <p>For example, if the subsequent device is an Altera family device, then this field displays pin termination supported by Altera device.</p> <p>Double-click to specify the pin property.</p> <p>This field displays the pin property supported by the subsequent device in the protocol.</p> <p>For example, if the subsequent device is an Altera family device, then this field displays pin property supported by Altera device.</p>
Diff. Type	Double-click to specify the pin use type as Negative or Positive.
Diff. Pair Signal	Enter the corresponding differential pin pair.
Serial I/O TX/RX Signal	Enter the corresponding serial IO transmitting / receiving pin.
FPGA Ext Termination	Double click to select and map the terminations.
Global Clock Route (Not shown in above figure)	<p>This column is applicable for only Spartan6 Global clock constraint.</p> <p>Before selecting this constraint, you need to set the target pin property as CC for clock pins in Target Pin Property column and should not assign any Group Constraint.</p>
Use Pin (Not shown in above figure)	<p>Click to select a pin.</p> <p>The Use Pin column is enabled for connector designs. <i>Use Pin</i> signifies setting a target to connect a FPGA pin to a connector pin. Note that while assigning a target for a diff pair pin both the selected pins in the combo box should be of a differential pair type. See here for more information on E dit connector part.</p>
Clock Group (Not shown in above figure)	Double click to select the clock group

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

Net Name	User needs to specify the net name in this column double click and type the name.
External Port	Select the check box to define the net as External port.
NetGroup	<p>Theses Net Name means "user defined net names or cached net names". After the design is run, these net names will be used as design net names</p> <p>Net Name Template settings are not allowed as net names.</p> <p>Displays the NetGroups.</p> <p>When you select the <i>first</i> option in the NetGroup tab of the <i>Settings</i> dialog box, and place the interface component on canvas, the NetGroups are automatically defined under this column. NetGroups are created based on the device protocol logical group properties.</p> <p>Under this column, you can modify, redefine, or remove the NetGroups.</p> <p>For more information, see the <a href="#">NetGroup</a> section.</p>
Import Signals From	Click to import signals from different sources.
Save As	When you import a protocol, all the contents of the protocol are loaded. You can edit them further to customize. Importing signal process are similar in Edit protocol and Define Virtual interface browser's.
	<p>This option invokes Save Protocol File dialog box. Click <i>browse (...)</i> to browse the directory where you want to save the protocol file, enter name and click <i>Save</i>. Or directly enter directory path with rules file name in <i>Rules File</i> field and click <i>OK</i>.</p> <p>A confirmation window is displayed for incomplete or incorrect details in this editor click <i>Yes</i> to move further.</p> <p>Use this option to save the protocol file at \$projectdirectory/temp/rules_editor/&lt;protocol_name&gt;.lrf</p> <p>Click to move the device instance tab.</p> <p>You can shuffle the order of the protocol chain order without deleting the protocol definition. Use the arrow buttons in the protocol editor to move the device instance tab.</p>

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

	Click to move the device instance tab.
	You can shuffle the order of the protocol chain order without deleting the protocol definition. Use the arrow buttons in the protocol editor to move the device instance tab.
Show/Hide Log	Click <i>Show Log</i> to display message log pane and <i>Hide Log</i> to hide the log pane.
Check Protocol	Click to check the protocol
OK	Click to save the protocol details.
Cancel	Click to cancel the operation.

### Searching and Filtering in the Rules Editor.

The Find feature lets you to find any keywords and strings in the any of the editors of FSP. You may also enter any search string, including wild cards '\*' and '^' as search criteria. Many filtering options are also provided in the Find feature to limit the search. You can also specify the forward and backward directions of the search.

**Note:** The *Find* feature is available in the *Rules Editor*, *Rules Instance Editor*, and *FPGA Port and Use Pin Mapping for Device Instance* dialog boxes.

#### How to Access

- Click *Find* icon or press Ctrl + F.

Name	Description...
Search text box	Enter the string that you are searching for.
Look In Column	Click and select the column names from the drop down list in which you want to search keyword.
Regular Expression	Select this option to apply regular expression search operation.
Find	Click to start the search operation.

#### Examples of Find

This section shows few examples to perform search using the Find feature using Regular Expression.

### **Example1**

To search the strings that starts with IO\_, such as IO\_L7N\_A4\_D20\_1, IO\_L7P\_A5\_D21\_1 and more, you can use the either regex IO\_\w+\_\\d\$ or IO\_.

‘IO\_’ ensures that the strings must start with IO\_.

\w+ subpattern matches all the alphabetic character and \_ character, such as L7N\_A4\_D20, L7P\_A5, D21\_.

\d\$ subpattern matches the numeric character (\d) and \$ matches the numeric character at the end of the string, such as 1.

### **Example2**

To search the strings that contains special characters such as PD11\_RXENB\_MII-TX-ERR, you can use the regex PD\\d+\_.<sup>\*</sup>.

‘PD’ ensures that the strings must start with PD\_.

\d+ subpattern matches numeric character after PD, such as PD11, PD22 and more.

.<sup>\*</sup> characters matches any character expect newline character zero or more times, RXENB\_MII-TX-ERR.

### ***Filtering the Display of Information***

The Rules Editor provides a filter bar that lets you to filter the display of information shown in the Rules Editor.

The *Filter* bar is regular expression enabled. You can use the alpha-numeric characters, underscore (\_), question marks, or asterisks to filter the display of information. You must select the *Reg Exp* option in the *Filter* bar to enable the regular expression.

The *Filter* bar provides three drop-down combo boxes that shows all the valid values and column names which you can use to filter out the information.

The following table describes the usage of each drop-down combo boxes:

## **Allegro FPGA System Planner User Guide**

### Dialog Box Descriptions

---

- 
- 1 Enter the text that need to be matched and displayed in the editor.

You can also use the alpa-numeric characters, underscore(\_), asterisks (\*) to perform regular expression based search operation. To enable the regular expression, select *Reg Ex* option.

- 2 Select a column name from the drop-down list in which you want to search the desired text.

For example, type DDR\_CS in the search text box and select *Pin Name* from this drop-down list, FSP displays only those rows whose *Pin Name* column cells contains DDR\_CS text.

- 4 Select this option to enable the regular expression based search.
-

## Export Design Constraints

The Export Design Constraints dialog box lets you export the design constraints to the UCF file. You can use this dialog box to export all constraints or export partial constraints.

### ***How to Access the Export Design Constraint dialog box?***

- Right-click on the device and choose *Constraints – Export Constraint*.

Name	Description
Export All Constraints	Select this option, if you want to export all the constraints.
Constraint File Bus Notation	Click and select to output the bus signals with bus notation.
Constraint File Path	Specify the path of the constraint files to export.
...	Click and select the constraint file.
Export Partial Constraints	Select this option, if you want to export the constraints partially.
Add	Click this button to add one more entry.
Delete	Click this button to delete the entry.
Export	Select this option if you want to export the constraint of this instance.
Instance Names	Click and select the instance names.
Banks	Click and select the instance banks.
Constraint File Path	Specify the path for the constraint files to export.

## Edit Symbol Settings

The Edit Symbol Settings dialog box allows you to specify the direction of the instance pins. After specifying the directions you can see the modified direction in Symbol Graphics View.

### ***How to Access the Edit Symbol Settings dialog box***

- In Instance Symbol Editor, right-click on the first level node and select *Edit Symbol Settings*.

Name	Description
Input Pin Direction	Select the Input pin direction.
Output Pin Direction	Select the Output pin direction
InOut Pin Direction	Select the InOut pin direction.
NC Pin Direction	Select the NC pin direction
Bank Power Pin Direction	Select the supply pin direction.
Global Power Pin Direction	Allows you to distribute the pins around the power symbol.
Default	Allows you to revert back to the original symbol settings.
OK	Click OK to change pin direction.
Cancel	Click Cancel to cancel the operation.

## Export CSV

The *Export CSV* dialog box lets you export the properties of the parts and their pins that are present in Canvas in an external file of comma separated values (csv) file.

### **How to Access**

- Click the *Export CSV* icon in the toolbar in Design Connectivity.

Name	Description
Export File Path	Specify the path and name of the file where you want to save the file.
Delimiter	Select the option (or character) to be used as the delimiter in the csv file. The values in the exported file will be separated by the specified delimiter.  For example, select <i>Comma</i> if you want the values to be separated by comma character in the csv file.
Select Instances for Export	Select the instances, of which the pin properties need to be exported in the csv file.  <b>Note:</b> Select <i>Design</i> if you want to export the properties of all the parts and their pins.
Check All	Click to select all the columns in the <i>Select Columns</i> pane.
UnCheck All	Click to unselect all the columns in the <i>Select Columns</i> pane.
Export	Click to export the properties.
Cancel	Click to exit the window.

## FPGA Port and Use Pin Mapping for Device Instance

The FPGA Port and Use Pin Mapping for Device Instance dialog box lets you to perform the following:

- Map FPGA Port names
- Map use pin

You can map the FPGA port from constraint files for all the signals connected to a device. You can also use this dialog box to map the FPGA pin location constraints to the FPGA ports from constraint files.

### **How to Access *FPGA Port and Use Pin Mapping for Device Instance* dialog box?**

- Right-click on the device and choose *Constraints – FPGA Port and Use Pin Mapping for Device Instance*.  
The *Select Constraints* dialog box is displayed.
- Browse to the directory where the csv file exists, select the file, and click *Open*.  
The FPGA Port and Use Pin Mapping for Device Instance dialog box is displayed.

Name	Description
Map	Select an option from the drop-down list on which you want to perform mapping. For example, select <i>RTL Port Name</i> to map RTL port name.
Automap bus Signals	Select this option, if you want to automatically map all the bus signals.
Hide After Mapping	Select this option if you want to hide the FPGA ports and location in FPGA ports pane after mapping.
Filter	<p>Use this option to filter constraints.</p> <p>For example you can specify a certain pin type in the Filter field to filter information for pins of that type only.</p> <p><b>Note:</b> This option is available in both the left and right panes.</p>

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

#### **Left Pane**

Displays the pin or port names, group names, and net names in tree view structure.

Click > to expand the list.

The first node displays the device instance name and signals count. Moreover, displays a list of interface instance names, device protocol names, and signals count.

Click > if you need to see the bus and pin details of any one of the instances.

Pin/Port Name	Displays the port or pin names of the interface instance or device protocol.
Group Name	Displays the group name of the pin/port names.
Net Name	Displays the net names.
RTL Port Name	When you first time invoke this dialog box, you will notice this column as an empty. You need to drop the RTL port name in this column of the right side pane.
Connected/ Assigned to Pin	Displays a list of pin numbers to which the pin/ports are connected.

#### **Right Pane**

RTL Port Name	Displays a list of RTL port names. These RTL port names are displayed after you import the CSV file.
Location	Displays a list of pin numbers.
Show Log	Click this button if you want to see the log message.
OK	Click to perform the mapping.
Cancel	Click to exit the window.

## Generate Allegro DE-HDL Symbols

The Generate/ReUse Allegro DE HDL Symbols dialog box allows you to generate the symbols for the parts used in design. This dialog box gives you the option of regenerating the symbols. You can also generate the global power properties for the part. The Generate Allegro DE-HDL symbols is divided into two panes:

- **Parts With Symbols**

Displays the instance names which has already DE-HDL symbols in the FSP directory. You can also regenerate the symbols.

- **Parts Without Symbols**

Displays the instance names which does not have DE-HDL symbols.

### ***How to Access the Generate Allegro DE-HDL Symbols***

- Choose *Generate - Symbols*.

Name	Description
Part(s) without Symbols	This section lists of parts in the design which do not have symbols.
Part Name	Displays the name of the part or the instance name.  If you have selected Generate Instance Specific Symbols for FPGAs, FSP displays the instance names.
Generate	Select the check box to generate symbols for the corresponding part.
Global Power Prop.	Select this option If you do not wish to see power-related pins on the generated symbols. (Make power pins invisible)  This check box is enabled only when generate/regenerate is checked.  Used to specify whether power pins are included as pins on the symbol or are added to the chips.prt file as implicit connections. If you do not wish to see power-related pins on the generated symbols, uncheck this box. After unselecting this check box these power pins will be added to the chips.prt file as No Connect pins
Back Up Symbol Data	Select this check box to back up the symbol data

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Part(s) with symbols	This section lists the part names, which already have symbols for them.  You can regenerate the symbols, if required.
Part Name	Displays the name of the part.
Re-Generate	Select the check box to re-generate the symbol for the corresponding part.
Global Power Prop.	Select the check box to generate the global power properties for the part.  This check box is available only when you select Re-generate check box.
Map custom attributes to properties	Select this check box to map the custom attributes.
Add	Click this button to add an entity
Delete	Click this button to delete an entity
Package pin	Display the package pin entity
Package	Display the package entity
Symbol pin	Display the symbol pin entity
Symbol	Display the symbol entity
OK	Click this button to generate the symbols.
Cancel	Click this button to cancel the operation.

## Generate Allegro DE-HDL Schematics

The Allegro DE-HDL Schematics dialog box lets you to set up the options for generating the DE-HDL schematic for your FSP design.

### **How to Access**

- Choose *Generate - Schematics*, to display the Generate Allegro DE-HDL Schematics dialog box.

### **Hierarchy**

Name	Description
Use Actual Port Type For Hierarchical Ports	Select this option if you wish to use the DE-HDL symbol port's direction for hierarchical ports. InOut port symbol is used by default for all hierarchical ports if this option is unchecked.
Flatten Hierarchical Termination Blocks	Select this option to remove the hierarchy for the termination hierarchical blocks and place underlying discretes into the schematic.

### **Placement**

Name	Description
Skip Unused Symbol Splits	Select this option to exclude symbol splits that have no net connections.
Do Not Mix Symbols and Hierarchical Blocks	Select this option to place one FPGA or connector hierarchical block per page. This may avoid overlapping with other components if the number of connections to the block increases.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

## General

Name	Description
Preserve Schematics	This option is useful if you are regenerating the schematic and want to preserve the modification done to the schematic generated initially. When the schematics are generated in preserve mode, placement of components and associated components (both passive and active components) are preserved in the previously generated schematic pages. This would help to avoid component rip up in Allegro, when the board file is updated with regenerated schematics from FSP.
Display Net Name as Instance Pin Name	Select this option to use the names of the nets connected to pins for FPGA components.  <b>Note:</b> This option overlays the text in the generated schematics and does not modify the symbols.
Propagate FSP Net Groups	Select this option to propagate the FSP defined net groups to Constraint Manager. These netgroups are used to create default bundles in Allegro PCB Editor.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Hierarchical Block	Block Location
Generate FPGA Hierarchical Blocks	<p>Select this option to generate schematic with FPGA's hierarchical blocks.</p> <p>By default, this option is unchecked when you invoke the <i>Generate Allegro DE-HDL Schematics</i> dialog box. This signifies that the flat schematic mode is enabled.</p>
<instance_name>(device name)	<p>Allows you to specify the location for hierarchical blocks generation.</p> <p>By default &lt;project name&gt;_lib:&lt;instance_name&gt;_hiersym is displayed.</p> <p>&lt;instance_name&gt;_hiersym is the default name assigned for the hierarchical blocks of the instance. You can change the hierarchical blocks name by clicking <i>browse (...)</i>.</p> <p>The Specify DE-HDL Symbol to Generate dialog box is displayed. Perform the following steps:</p> <ul style="list-style-type: none"><li>■ In the Library Name drop down list choose the library in which you want to generate the FSP block.</li><li>■ In the Cell Name field, re-enter a name for the hierarchical block and click <i>OK</i>.</li></ul> <p>The specified Library Name and Cell Name are updated accordingly in Block Location field.</p>

## Advanced Settings

Click to access the Edit CPM Settings dialog box. The Edit CPM Settings dialog box lets you set up and customize the settings for schematic pages.

## Symbols tab

Use this tab to specify the default symbols that should be used for generating the DE-HDL schematic. The library, cell and view of the selected symbols are displayed.

### Note:

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

- The default symbols are selected from the standard library. If you want to use symbols from the standard library, ensure that the standard library is defined in the cds.lib for your project and added to the list of project libraries. You can add libraries for the project using the Edit Libraries File of the Settings dialog box.
- If the specified page border symbol is not found in the library, the DE-HDL schematic will not be generated.

Page Border	<p>Click <i>browse (...)</i> to specify the page border symbol that will be used in the schematic.</p> <p>Component Browser invokes for page border selection when you click <i>browse</i>.</p>
Bus Tap	<p>Click <i>browse (...)</i> to specify the TAP symbol that will be used in the document schematic.</p> <p>Component Browser invokes for Ctap selection when you click <i>browse</i>.</p> <p>The TAP symbol must have:</p> <ul style="list-style-type: none"><li>■ Exactly two pins.</li><li>■ The pin names of both the pins must have the \NAC property and the pin name of the first pin must also have the \NWC property. For example, if the first pin is named B and the second pin is named S, enter the pin names as B \NAC \NWC and S \NWC.</li><li>■ The first pin must be at the origin of the symbol and must not have the BN property.</li><li>■ The second pin must be on the x-axis and must have the BN property.</li><li>■ The second pin must have a positive x-coordinate and be on the grid.</li><li>■ Select the TAP component from the Cadence standard library if the TAP component in your library does not meet these requirements.</li></ul>

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Ports	<p>Ports provides the connection between the levels of hierarchy on a schematic page. The FSP library contains a set of symbols for identifying the ports and offpage connections.</p> <p>Use this options to specify the Port settings. These options enable you to specify the symbols for the input, output and inout port symbols to be used in the schematic.</p>
Input	<p>Click <i>browse (...)</i> to specify the offpage symbol that will be used in the schematic as INPUT offpage.</p> <p>Component Browser invokes for Input port selection when you click <i>browse</i>.</p>
Output	<p>Click <i>browse (...)</i> to specify the offpage symbol that will be used in the schematic as OUTPUT offpage.</p> <p>Component Browser invokes for output port selection when you click <i>browse</i>.</p>
InOut	<p>Click <i>browse (...)</i> to specify the offpage symbol that will be used in the schematic as INOUT offpage.</p> <p>Component Browser invokes for inout port selection when you click <i>browse</i>.</p>
Add OffPage Symbols	<p>Off-page symbols provides the connection within the schematic pages. An off-page symbol is connected by name to other off-page symbols within the schematic page.</p> <p>Selecting this check box enables the options for specifying the input, output and inout offpage symbols to be used on the left and right side of the DE-HDL schematic.</p>
Input	Specifies the offpage symbol that will be used in the schematic as INPUT offpage.
Output	Specifies the offpage symbol that will be used in the schematic as OUTPUT offpage.
InOut	Specifies the offpage symbol that will be used in the schematic as INOUT offpage.

## Power Tab

Use this tab to specify the power symbol options for whether to generate the power symbols in schematics or not.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Name	Description..
Output Power Symbols in Schematics	Select to generate the power symbols in schematics. <b>Note:</b> All the below fields are enabled after selecting this option.
First Column	Displays the power regulator names of the current design.
Library	Click and select the library name from the drop down list.
Cell	Click and select the cell name from the drop down list.
View	Click and select the symbol view from the drop list.

## Placement Tab

Use this tab to specify the component placement options for placing components in generated schematic.

Name	Description
Grid Size	<p>The grid size settings is used for the schematics pages to place components and route the design.</p> <p>Specify the grid size to use in the generated schematic.</p> <p><b>Note:</b> This grid size is used internally by the Schematic Generator (schgen) and is independent of the grid size specified in Design Entry HDL.</p> <p> <i>Important</i></p> <p>If you are using DE-HDL grid size as 0.1, it is recommended that you should either set the grid size to 50 in this field or set the schgen directive in FSP section of the site cpm file to value 50.</p> <p>START_FSP</p> <p>schgen_grid '50'</p> <p>END_FSP</p>

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

Page Margins	You can use this option to specify the page margins for a schematic page. The symbols are placed within the margins specified.  Specify the page margins in grid multiples to determine the drawing area.
Top	Specify the top-most point on the desired drawing area as a multiple of the grid size.
Left	Specify the left-most point on the desired drawing area as a multiple of the grid size.
Bottom	Specify the bottom-most point on the desired drawing area as a multiple of the grid size.
Right	Specify the right-most point on the desired drawing area as a multiple of the grid size.
Component to Component spacing	Specify the grid spacing between any two components, as a multiple of the grid size.
Display Net Name as Symbol Instance Pin Name	Select this option if want to use the net names as pin names for the symbol pins of the component used in the design.  <b>Note:</b> If you use the same symbol multiple times in the same design, and select this option then net names of the respective components are used as pin names for the connected pins.
Page Border Information file	Use this text box to specify the customized exclude areas while generating schematic. The page border information is read from the crefer.dat file specified in the text box.

## Routing tab

Use this tab to specify the routing options for the generated schematic.

Name	Description
Stub Lengths for nets on the left	Use this text box to specify the maximum stub length for the nets that are connected to a pin on the left of the component. The value entered by you is the number of grids, therefore, the actual stub length is calculated as number of grids x grid size.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

Stub Lengths for nets on the right	Specify the maximum of grids that can be used for drawing stubs for the nets connected to the pins on the right of the component.
Add dots at points of net shorts	Select this check box to ensure that net to net connections are indicated with a dot.
Short NC nets	Select this check box to ensure that the generated schematic has all the unconnected pins short together.
Flatten all nets	Select this check box to flatten all nets in the generated schematic.
Handle Concatenated Signals	Select this check box if you want to add a prefix to signals in case of concatenation of signal names
Signal name prefix to be used for concatenation	Specify the value to prefix to the signal names in case of concatenation
Show NC for connection.	Select this check box if you want show NC for unconnected bits in the generated schematic.

## Colors tab

Use this tab to specify the colors for the drawing objects in generated DE-HDL schematic.

Name	Description
Color For Drawing Objects	Displays the color used for component symbols, wires, properties, comments, and the bounding box that will be drawn around associated components in the generated schematic.  ■ To change the color for an object, click on the color for the object to display the color palette. Click on the color you want to use in the color palette.
Symbol Color	Displays the default color for component instances in the generated schematic.
Wire Color	Displays the default color for wires in the generated schematic.
Property Color	Displays the default color for properties in the generated schematic.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Comment Color	Displays the default color for comments in the generated schematic.
Bounding Box Color	Displays the default color for the bounding box that will be drawn around associated components in the generated schematic.

## Properties tab

Use this tab to specify settings for default alignment, visibility and size of the properties for generating DE-HDL schematic.

Name	Description
Default Alignment	<p>Specifies the default option for aligning properties in the generated schematic.</p> <p><b>Note:</b></p> <ul style="list-style-type: none"><li>■ If FSP is unable to align notes or properties using the default option in the generated schematic, it will place them where it finds adequate space in the generated schematic.</li><li>■ The alignment setting specified for a property on the symbol for a component will always win irrespective of the settings you specify here. For example, if the alignment for a property on the symbol is set to Right, the property will be right aligned on the schematic even if you select Left as the default option in the Default Alignment field.</li></ul>
Default Visibility	Click and select the default option to display the properties name and value in the generated schematic.
Size	Specify the size of the text to be used for displaying properties in the schematic. The text size should be specified in 5000ths of an inch. The default size is 41.

## Custom Attribute Tab

Use this tab to map the attributes to properties.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

Name	Description	Remarks
Select	Select this check box to enable the Entity, Attribute Name, and Property Name fields.	
Entity	<p>Click and select the entity on which you want to add the property.</p> <p>Displays four types of entities:</p> <ul style="list-style-type: none"> <li>■ Instance Pin - Select this if you want to add the property on instance pin.</li> <li>■ Net - Select this if you want to add the property on net.</li> <li>■ Instance - Select this if you want to add the property on instance.</li> <li>■ Termination - Select this if you want to add the property on termination.</li> </ul>	
Attribute Name	<p>If the instances have the attribute name in your design, then specify the same attribute name in this field.</p> <p><b>Example</b></p> <p>If you have a instance with attribute name MYINST defined in Properties of Device Instance dialog box, then you must specify the same attribute name in this field.</p>	This option displays various standard attribute names.
Property Name	Allows you to specify the attribute value.	

## Generate OrCAD Symbols

The Generate OrCAD Symbols dialog box allows you to generate the symbols for the parts used in design. This dialog box gives you the option of regenerating the symbols. The Generate OrCAD symbols is divided into two panes:

■ **Parts With Symbols**

Displays the instance names which has already OrCAD symbols in the FSP directory. You can also regenerate the symbols.

■ **Parts Without Symbols**

Displays the instance names which does not have OrCAD symbols.

### ***How to Access the Generate Allegro DE-HDL Symbols***

■ Choose *Generate - Symbols*.

Name	Description	Remark
Part(s) without Symbols	This section lists of parts in the design which do not have symbols.	
Part Name	Displays the name of the part or instance name with the path location	If you have selected Generate Instance Specific Symbols for FPGAs, FSP displays the instance names.
Generate	Select the check box to generate the symbol for the corresponding part.	
Part(s) with symbols	This section lists the part names, which already have symbols for them.	You can regenerate the symbols, if required.
Part Name	Displays the name of the part or instance name with the path location	
Re-Generate	Select the check box to re-generate the symbol for the corresponding part.	

## **Allegro FPGA System Planner User Guide**

### Dialog Box Descriptions

---

Map custom attributes to properties	Select this check box to map the custom attributes.	
Add	Click this button to add an entity	
Delete	Click this button to delete an entity	
Package pin	Display the package pin entity	
Package	Display the package entity	
Symbol pin	Display the symbol pin entity	
Symbol	Display the symbol entity	
OK	Click this button to generate the symbols.	
Cancel	Click this button to cancel the operation.	

## Generate OrCAD Schematics

The OrCAD Schematics dialog box allows you to generate OrCAD schematic for your design.

### ***How to Access the Generate OrCAD Schematic***

- Choose *Generate - Schematics*.

Name	Description	Remark
Project Information		
Schematic Output Directory	Display the path of the Allegro schematic directory.	Click Browse to change the selected different location for schematic generation.
Project Name	Displays the project name. However, you can edit it to generate the schematic with another name.	The schematics are generated with the project name and design name specified here.
Schematic Name	Displays the design name. However, you can edit it to generate the design with another name.	
Create Top Level Design	Select this check box to create a hierarchical block that represents the top level of the FSP-related circuitry	This makes it very easy to make connections between components in FSP and components that lie outside of FSP'
Top Level Schematic	Display the name of the Top Level Schematic	
Place Termination blocks in separate Page	Select this feature to place the termination blocks on a separate page.	

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Bus Bit Order	<p>Select <i>MSB to LSB</i> if you want to name the vectored signals as A&lt;SIZE-1...0&gt;. For example, a 4-bit signal can be specified as A&lt;3..0&gt;.</p> <p>Select <i>LSB to MSB</i> if you want to name the vectored signals as A&lt;0...SIZE-1&gt;. For example, a 4-bit signal can be specified as A&lt;0..3&gt;.</p> <p> <i>Important</i></p> <p>When you generate OrCAD schematic using <i>Create Top Level Design</i>, the bus bit order is applied to both the top-level design (if the design contains extern nets) and hierarchical schematic blocks; however if you do not select the <i>Create Top Level Design</i> option, then the bus bit order will be applied to the hierarchical schematic blocks.</p>
<p><i>Generate Hierarchical Blocks for FPGAs</i></p>	
Hierarchical Block	A list of device instance names appear under this column.
Block Location	<p>Lets you specify the location in which you want to generate the hierarchical schematic blocks for device instances.</p> <p>To specify the block location, do the following:</p> <ol style="list-style-type: none"><li>1. Click <i>Browse (...)</i>. The Specify OrCAD Symbol to Generate dialog box appears.</li><li>2. Click <i>Browse (...)</i> in the <i>Library File (*.olb)</i> text box. The <i>Select OLB File</i> dialog box appears. Navigate to the .olb file and click <i>Open</i>.</li><li>3. Select a package name in the <i>Package Type</i> drop-down list.</li><li>4. Click <i>OK</i>.</li></ol>
Skip Unused Splits	Select this option to skip the unused splits in the generated schematics for any particular instance which is not used in the design.
<p><i>Map Custom attribute to properties</i></p>	
Add	Click this button to add a new entity

## **Allegro FPGA System Planner User Guide**

### Dialog Box Descriptions

---

Delete	Click this button to delete a new entity	
Select	Select the check box to select the attribute type	
Entity	Select the Entity from the drop down menu	Allows the user to apply the attribute globally on similar types of entities
Attribute Name	Select the Attribute Name	
Property Name	Specify the property name	
Net	Display the net entity	
Instance pin	Display the instance pin entity	
Instance	Display the instance entity	
OK	Click to generate the schematics.	
Cancel	Click to cancel the operation.	

## Group Settings For Interface Instance <inst\_name>

The *Group Settings for Interface Instance <inst\_name>* dialog box lets you configure the target settings. Using this dialog box, you can target the group of the selected instance to a device or specific bank of the device. You can also specify the bank that you do not want to connect the group.

### ***How to Access the Group Settings for Interface Instance <inst\_name>***

- Right-click on the interface instance on the *Canvas* and choose *Groups Settings*.

Name	Description
Group	Lists the groups name of the selected interface.
Connect to Device	Let's you target the specified group to a device instance. Click the drop-down button and select an instance name to target.
Use Banks	This option lets you to target the interface groups to device instance banks of your own choice. You can also exclude or preserve the banks from making connections using this feature.  To use the banks: <ul style="list-style-type: none"><li>■ Click the drop down button and select the bank numbers from the list and click OK.</li></ul> After clicking OK, the selected bank names is displayed in Use Bank text box. These selected banks will be used further during synthesis.  <b>Note:</b> When you hover mouse pointer over a bank, the bank region is highlighted.
Don't Use Banks	This column lets you to select the banks that you wish to preserve or exclude from connections.  To exclude or preserve the banks: <ul style="list-style-type: none"><li>■ Click the drop down button.</li></ul> Select the bank numbers and click OK.
Contiguous Pins	Let's you specify the contiguous pins of the group.

## **Allegro FPGA System Planner User Guide**

### Dialog Box Descriptions

---

Clock BUFR Type	This column is applicable and enabled for <i>Spartan 3/3A</i> devices.

## Generate/Update Layout Placement Data

The Generate Layout data dialog box allows you to generate the allegro board file for your design. You can also update the placement data.

### ***How to Access the Generate/Update Layout Placement Data***

- Choose *Generate - Allegro PCB Placement*.

Name	Description	Remark
Generate Board File and Placement Data	Select this check box to generate the board files.	FSP generate placement.scr and placement.xml files.
Update Placement Data	Select this radio box to update the placement data file.	
Board File Name	Enter the name of the board file.	
Layout Data Directory	Display the path of the board file/placement file.	Click Browse to change and select a different location for board files.
Check All	Click to select all the check boxes(of instances) to generate the board files	
UnCheck All	Click to Uncheck all.	
OK	Click OK to finish the operation	
Cancel	Click Cancel to cancel the operation	

## Generate SCM Design

Use *Generate SCM Design* dialog box to generate SCM design files.

### How to Access

- Choose *Generate – SCM Design*.

Name	Description
Generate FPGA Hierarchical Blocks	<p>Select this option to generate schematic with FPGA's hierarchical blocks.</p> <p>By default, this option is unchecked when you invoke the <i>Generate Allegro DE-HDL Schematics</i> dialog box. This signifies that the flat schematic mode is enabled.</p>
<instance_name>(device name)	<p>Allows you to specify the location for hierarchical blocks generation.</p> <p>By default &lt;project name&gt;_lib:&lt;instance_name&gt;_hiersym is displayed.</p> <p>&lt;instance_name&gt;_hiersym is the default name assigned for the hierarchical blocks of the instance. You can change the hierarchical blocks name by clicking <i>browse (...)</i>.</p> <p>The Specify DE-HDL Symbol to Generate dialog box is displayed. Perform the following steps:</p> <ul style="list-style-type: none"><li>■ In the Library Name drop down list choose the library in which you want to generate the FSP block.</li><li>■ In the Cell Name field, re-enter a name for the hierarchical block and click <i>OK</i>.</li></ul> <p>The specified Library Name and Cell Name are updated accordingly in Block Location field.</p>
Propagate FSP Net Groups	Select this option to propagate the FSP defined net groups to Constraint Manager. These netgroups are used to create default bundles in Allegro PCB Editor.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Select	Select this check box to enable the Entity, Attribute Name, and Property Name fields.
Entity	<p>Click and select the entity on which you want to add the property.</p> <p>Displays four types of entities:</p> <ul style="list-style-type: none"><li>■ Instance Pin - Select this if you want to add the property on instance pin.</li><li>■ Net - Select this if you want to add the property on net.</li><li>■ Instance - Select this if you want to add the property on instance.</li><li>■ Termination - Select this if you want to add the property on termination.</li></ul>
Attribute Name	If the instances have the attribute name in your design, then specify the same attribute name in this field.  <b><i>Example</i></b> If you have a instance with attribute name MYINST defined in Properties of Device Instance dialog box, then you must specify the same attribute name in this field. This option displays various standard attribute names.
Property Name	Allows you to specify the attribute value.

## Import Signals from OrCAD Symbol

The Import Signals from OrCAD Symbol allows you to create logical rule file using existing OrCAD symbol. You create the logical rule file by importing OLB file.

### **How to Access**

- In the Rules Editor dialog box, click *Import From* and select *OrCAD Symbol*.

Name	Description
Library File (*.OLB)	Specify the OLB file name and path to the directory where the file exists.
browse (...)	Click to browse to the directory where the OLB file exists.
Package Type	Click the drop down list to select the type of package.
PCB Footprint	Specify the footprint name.
OK	Click OK to import the pin details in Rules Editor dialog box.
Cancel	Click Cancel to exit the dialog box.

## Import Verilog/VHDL Signals for Device Instance

The Import Verilog/VHDL Signals for Device Instance allows you to create virtual interface using Verilog/VHDL files. You can create the virtual interface by importing the Verilog/VHDL files. You can also import the constraint files along with Verilog/VHDL files which is an optional. If you do not import the constraint files you need to manually specify information later.

### ***How to Access the Import Verilog/VHDL Signals for Device Instance***

- Choose *Import Signals From - Constraints and Pinouts File* in Define Virtual Interface for Device Instance dialog box.

Name	Description	Remark
HDL Type	Select the Verilog option to import the verilog file and VHDL option to import the VHDL file.	
HDL File	Specify the path of HDL file to load signals. Or Click ... to browse to the HDL file location.	
Constraints/Pinouts File	Specify the path of constraint and pinouts file to load the signal information. Or Click ... to browse to the constraint file location.	
Module Entity	Click and select the type of module.	
Load Signals	Click to display the signals in below pane.	
Filter	Use this option to filter constraints.	For example you can specify a certain pin type in the Filter field to filter information for pins of that type only.
Check	Click this option to select all the signals.	

## **Allegro FPGA System Planner User Guide**

### Dialog Box Descriptions

---

Uncheck	Click this option to unselect all the signals.	
Import	Click this option to import the signals.	

## Instance Symbol Editor

Use the Instance Symbol Editor dialog box to customize the symbol before generating the DE-HDL or OrCAD symbols and schematics.

The Instance Symbol Editor dialog box contains six columns. You can do the following:

- Change pin directions
- Add or edit the custom attributes
- Split the symbols into smaller number of splits

### ***How to Access the Instance Symbol Editor dialog box***

You can access the Instance Symbol Editor dialog box from Setup Symbol Data form of Allegro DE-HDL and OrCAD.

- Click *Customize Symbol*.

Name	Description	Remark
Pin Name	Displays the instance pin names.	
Pin Number	Displays the instance pin numbers.	
Pin Type	Displays the instance pin types.	
Pin Direction	Click and select this option to define the pin direction.	
Custom Attribute	Use this option to specify the custom attribute. <ul style="list-style-type: none"><li>■ Click cell and click ...</li><li>■ Specify the attribute name and value.</li></ul>	
Pin Name as Net Name	When you select the Generate Pin Name as Net Name option, the pin names are displayed.	
Generate Pin Name as Net Name	Select this option if you want to use the instance pin names as net names.	

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

The following table illustrates the pop-up menu options in first level node of Instance Symbol Editor form.

Name	Description
Add Split Symbol	Click to create a new split in symbol.  The Split Symbol Name dialog box is displayed. Enter a split name and click OK.
Merge All Splits	Click to merge all the splits of symbol in one split.
Split by Bank	Click to split the symbol by bank wise.
Split by Function	Click to split the symbol based on the pin connections.  The following split function is created: <ul style="list-style-type: none"> <li>■ &lt;bank_power_split&gt; - All the power and ground pins are moved into this split.</li> <li>■ &lt;global_power_pins&gt; - All the global power pins are moved into this split.</li> <li>■ &lt;unused_pins-split&gt; - All the unused pins of the instance are moved into this split.</li> <li>■ &lt;interface_instance_name1_interface_instance_name2..&gt; - The instance pins connected to &lt;inter_instan_name1&gt; are moved into this split.</li> </ul>
Auto Split Symbol Pins	Click to split the symbol into smaller number of splits.
Edit Pin Directions	This option will open Edit Symbol Settings dialog box.  Click and select the pin direction from the drop down list of any displayed options and click OK.  Click Default to apply the default settings.
Edit Custom Attributes	Click this option to add or edit the custom attributes.

The following table illustrates the pop-up menu options in second level node of Instance Symbol Editor form.

Name	Description

## **Allegro FPGA System Planner User Guide**

### Dialog Box Descriptions

---

Merge With Split	This option is visible only on splits. This option displays the list of splits of the current symbol.  Choose Merge With Split – split<1> or split<2>.. and select any one of the splits from the list with which you want to merge the selected symbols.
Change Split Name	Click this option to change the split name.  The Select Split Symbol Name dialog box is displayed. Reenter a new name and click OK.
Edit Custom Attribute	Click this option to add or edit custom attributes.

## Import Constraint For Device Instance

The Import Constraint For Device Instance dialog box lets you to import the constraint file in your design. FSP updates the connectivity during importing.

### **How to Access the Import Constraint For Device Instance dialog box?**

- Right-click *Device* and choose *Constraints – Import Constraint*.  
The *Select Constraint File* dialog box is displayed.
- Navigate to the file and click *Open* to load the file.  
The *Import Constraint for Device Instance* dialog box is displayed.

Field & Button	Description	Remark
Select Interface to Import	Select the interface instance name to import the constraint files.	Display interface instance names present on canvas.
Filter	Use this option to filter constraints.	For example you can specify a certain pin type in the Filter field to filter information for pins of that type only.
Show Log	Click to display the log pane.	The successful and error reports is displayed in log pane.

## Import from CSV

The Import from CSV dialog box lets you import details of the parts from the external files of different format such as comma separated value (.csv) file, tab separated file or space separated file in various forms of FSP. Both pin and group details can be updated using this dialog box. The Import from CSV dialog box is available in various forms and process of updating part information is similar in each of the forms. However, the *Import from CSV* dialog box in the Design Connectivity window provides an additional feature, *Selecting Reference Column*. When you select a column as reference column, the values in these columns are considered as reference or address of the values in the other columns. FSP internally maps the values of the reference columns with the values of the remaining columns. For example, if you select the Pin Number column as reference column, then each pin number is mapped with the values of the remaining columns.

### **How to Access**

- Choose *Library - Create Part - Interface*.  
The Edit LRF dialog box is displayed.
- Click *Import From* and select *CSV*.  
The *Select from CSV* dialog box is displayed.
- Browse to the file, select the file, and click *Open*.

Or

- In Design Connectivity, click the *Import from CSV* icon.

Name	Description
Ignore Rows	Specify the row numbers of the csv file that you do not want to ignore during importing and click <i>Apply</i> . For example. 1,3,4..so on.
Delimiter	Specify the type of file you are importing. For example, for tab separated file select Tab and for CSV file select Comma.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

First Row (1)	<p>Click the drop down list to select the column name.</p> <p>For example, for pin number column select the pin number name from drop down list and for IO standards select IO standard from drop down list.</p> <p><b>Note:</b> Based on the column names specified in the first row, the respective column details is imported in Rules Editor dialog box.</p>
Find and Filter text box	<p>This feature lets you search for specific text in the columns or search for a pin by name.</p> <p>In the <i>Find</i> toolbar, you enter a string and specify the column (one or two) in which you want to find the string. FSP searches all the strings in the specified column to find string that matches with the specified string.</p> <p>The <i>Find</i> search bar is regular expression enabled. You can use the alpha-numeric characters, underscore (_), question marks, or asterisks to perform search operation. You must select the <i>Reg Exp</i> option in the <i>Find</i> search bar to enable the regular expression search feature.</p>
OK	Click OK to import the pin details in Rules Editor.
Cancel	Click Cancel to exit the Import CSV dialog box.

## Lock Nets

The Lock Nets dialog box allows you to lock and unlock nets of your design. This feature specially helps you when you are modifying one part of the design and you do not want to disturb the other nets. There are various options provided to you for locking and unlocking different types of nets such as interface nets, device protocols, constraint nets, clock nets and many more. Before invoking Lock Nets dialog box you must remember few points. See [Locking and Unlocking Nets](#) section.

You can lock and unlock the nets by two ways:

- Through RMB options
- Through Lock Nets dialog box

### ***How to Access the Lock Nets dialog box***

- Choose *Design - Lock Nets*.

Name	Tool bar Icon	Description
Lock All		Lets you to lock all the nets of your design at one go. The locked nets are displayed in yellow color.
UnLock All		Lets you to unlock all the locked nets at one go. The unlocked nets are displayed back in normal color.
Lock Clocks		Lets you to lock the clock nets. Select the clock nets from the list and click <i>Lock Clocks</i> .  The locked clock nets are displayed in dark yellow color.
Unlock Clocks		Lets you to unlock the locked clock nets.  The unlocked clock nets are displayed back in normal color.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

Lock Selected		Lets you to lock the nets of your own choice. Select the nets from the list that you want to lock and click <i>Lock Selected</i> .  A confirmation dialog box known as Lock Nets Confirmation is displayed click Yes to lock the nets.  The selected locked nets are displayed in yellow color.
UnLock Selected		Lets you to unlock the locked nets.  A confirmation dialog box known as Unlock Nets Confirmation is displayed click Yes to unlock the nets.  The selected locked nets are displayed back in normal color.  <b>Note:</b> You can also use <i>Unlock All</i> option to unlock the selected nets.
Lock Constraint Pins		Lets you to lock the constraint nets.  Select the constraint nets from the list and click <i>Lock Constraint Nets</i> .
Unlock Constraint Pins		Lets you to unlock the locked constraint nets.  Select the constraint nets from the list and click <i>Unlock Constraint Nets</i> .  <b>Note:</b> You can also use <i>Unlock All</i> option to unlock the constraint nets.
Import Nets		Lets you to import the net details from text file to lock the nets.  Click this option. Browse for the file and click <i>Open</i> in Import Nets dialog box.
Export Nets		Lets you to export the locked nets details to text file.  Lock the nets and click this option. Browse for the directory where you want to save the file and enter the text file name and click <i>Save</i> in Export Nets dialog box.
Find		Click this icon to invoke Find dialog box.
Close		Click to close the window.

## Configure Multi-Voltage Pins

The *Configure Multi-Voltage Pins* dialog box lets you seamlessly assign voltage values to the multiple power pins. These power pins are left unassigned by FSP during synthesis. After synthesis, you can use this dialog box to assign voltage values to the left over power pins. This feature is applicable for Virtex 7 devices.

### ***How to access the Multiple Voltages Supported Pins in Design***

- Click the *Auto Add Regulator* icon in the *Power Connections* window.

Name	Description
Pin Name	Displays a list of power pins names.
Supported Voltages	Click and select a voltage value from the drop-down list.
OK	Click to save the changes.
Cancel	Click to discard the changes and exit the window.

## Multiple Device Connection Groups

The Multiple Device Connection groups pane lets you to edit and delete the interface protocol groups.

### ***How to Access the Multiple Device Connection Groups pane***

- In Design Connectivity, click on the interface row.  
The Instance properties are displayed in the Properties.
- Click *Group Settings*.  
The *Group Settings for Device Instance <inst\_name>* dialog box is displayed.
- Under the *Connect to Device* column, click the drop-down button and select two or more instance names.  
The *Multiple Device Connection Groups* pane is displayed at the bottom of the window.

Name	Description	Remark
Multiple Device Connection Groups		
Groups	Displays the name of the groups targeted groups	
Targeted Devices	Display the name of the targeted devices i.e. multiple devices	
Edit	Click this to edit the interface protocol.	After clicking Edit a new window called Edit interface protocol appears.
Delete	Click to delete the targeted groups	
OK	Click to set the interface properties.	
Cancel	Click to cancel the operation.	

## Map Clock Groups

The Map Clock Groups dialog box allows you map the clock groups.

**Note:** This option is available for only Spartan 3/3A devices.

### ***How to Access the Map Clock Groups***

If you have already defined the clock groups in Rules Editor then perform the following step:

- Right-click Interface instance and select *Interface Instance*.

Name	Description	Remark
Map Clock Groups		
Group Name	Displays the name of the clock group	
Design Clock Group	Displays the name of the global clock names	Design clock group names defined in the Setting > Design group appear here.
OK	Click to set the interface properties.	
Cancel	Click to cancel the operation.	

## Map FPGA Resources For Device Instance

The Map FPGA Resources For Device Instance dialog box lets you to map the resources available in constraint files. You can use this dialog box to map two types of resources Dependent and Independent resources.

### **How to Access Map FPGA Resources For Device Instance dialog box?**

- Right-click on the device and choose *Constraints – Map FPGA Resources For Device Instance*.  
The *Select Constraints File* dialog box is displayed.
- Browse to the file, select the file, and click *Open*.

Name	Description	Remark
Resource Mapping	Displays the device and instance name in tree view structure.  Drop the resources from Resource pane to this pane through mouse pointer.	
Resources	Display the resources when you click <i>Load Signal</i> .	
Hide Identical Resource	Select this option to hide the resources which are identical in Resources pane.	
Hide Comments	Select this option to hide the comments in Resources pane.	
Show Log	Click this button, if you want to see the log messages.	

## Manage Protocols

The *Manage Protocols* dialog box lets you create a new protocol, modify or delete the existing protocol.

### ***How to Access the Manage Protocol dialog box?***

- In Design Connectivity, click on the device instance row.  
The properties of the device instance is displayed in the *Properties*.  
■ Click *Manage Protocols*.

Name..	Description..
Create	Invokes the Create New Protocol dialog box.
Edit	Select the protocol name and click <i>Edit</i> . Invokes the Edit Protocol dialog box.
Delete	Deletes the existing protocol.

## Net Removal Confirmation

The Net Removal Confirmation dialog box is a confirmation window that displays the nets that you have decided to remove from the design. Before removing the nets you must remember the following points:

- Nets belonging to interface group with same\_bank(or same\_clock\_region) constraint in non-DeepNwide mode will disconnect all pins in same group of primary and secondary interface(s).
- Nets belonging to group with same\_bank constraint(or same\_clock\_region) will disconnect all pins in same group.
- Nets belonging to pin marked as replicate in DeepNWide mode, will disconnect same pins in all interfaces (primary and secondary) of DeepNWide group.

### ***How to Access the Net Removal Confirmation dialog box***

- Right-click any net on canvas and select *Delete Selected Nets*.

Name	Description	Remark
Ok	Select this option to remove the displayed nets.	
Cancel	Select this option if you do not want to remove the nets.	

## Power Connections

The Power Connections dialog box lets you add, modify, delete, and map power regulators.

For detailed information about power mapping, see the [Working with Power Regulators](#) chapter.

### **How to Access the Edit Power Regulators**

- Choose *Tools – Power Connections*.

Name	Description
Add	<p>This option lets you define a new power regulator.</p> <p>When you click this option, a new blank row appears. You can perform the following operations:</p> <ul style="list-style-type: none"><li>■ Under the <i>Regulator Name</i> column, &lt;New_Reg_0&gt; name appears by default. Modify the name based on your requirement.</li><li>■ Under the <i>Voltage</i> column, zero appears by default. Enter the voltage value.</li><li>■ Under the &lt;<i>Instance_Name</i>&gt;[<i>Part Name</i>] column, you can either specify power pin name or voltage value.</li></ul> <p>The <i>Add</i> option provides another option, <i>Add Regulator from Schematic Symbol</i>.</p>
Delete	Click on a row and click this option to delete the selected regulator.
Auto Add Reg	<p>Click to automatically define regulators and voltage values.</p> <p><b>Note:</b> This option also defines the regulators with voltages for the power pins for which the regulators are not defined earlier.</p>
Map Missing Regulator	<p>Use this option to automatically map the regulators to the power pins for which mapping is not done earlier.</p> <p><b>Note:</b> After you click this option, the regulators are automatically mapped to the power pins based on the required voltage. For voltage values, refer to the <i>Voltage Value</i> column in the <i>Design Connectivity</i> window. No mapping is performed for the power pins that do not require voltage.</p>

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Reset and Map Regulators	<p>This option is useful when you have not specified any regulator names in the cell(s) and want to automatically map the regulator names in the empty cell(s). This option is also useful to correct any incorrect mapping done by you.</p> <p><b>Note:</b> After you click this option, at first all the power regulators are reset and then automatically mapped to the power pins based on the required voltage for the specified power pin. For voltage values, refer to the <i>Voltage Value</i> column in the <i>Design Connectivity</i> window.</p>
Reset To Default Voltages	This options lets you revert back to the default voltage values of the power pins. If you have specified the power pin names then this option will reset to voltage based mapping based on the voltage requirement of the instances.
Regulator Name	Displays the regulators names. These cell(s) are editable. You can double-click on it and modify the name anytime during the project.
Voltage	Displays the voltage values. These cell(s) are editable. You can manually specify the voltage values for the defined regulators.
Instance Name <i>U1, U2, XP17..so on.</i>	<p>When you click on the cell, a small dialog box appears. This dialog box provides you the following features:</p> <ul style="list-style-type: none"><li>■ Name Based Mapping</li><li>■ Voltage Based Mapping</li></ul> <p>For detailed information, see the <a href="#">Working with Power Regulators</a> chapter.</p>

## Process Option Editor

The Process Option Editor dialog box lets you make the connections between the instances. You can also run the design using various advance process options to minimize the crossovers and reduce the number of board layers.

### ***How to Access the Process Option Editor dialog box?***

- Choose *Design – Run Design*.

Name	Description
------	-------------

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Instance/Protocol Name	<p>Displays a list of names of the devices and interfaces in a tree view structure that are present on the Canvas. You can browse through this column to find out how many interfaces are connected to each devices.</p> <p>You can expand the list by clicking the &gt; button to the left of the device instance names. You will notice a list of interface names. You may further expand the list corresponding to these interfaces and find information about groups of each interfaces.</p> <p>You can further expand the interface names by clicking the &gt; button to the left of the interface name. You will notice a list of group names corresponding to the interface.</p> <p><b>Note:</b> You can use the toolbar options, <i>Expand All</i> to expand the complete list and <i>Collapse All</i> to collapse the list.</p> <p>When you first time invoke the <i>Process Options Editor</i> dialog box, by default all the check boxes are selected. This indicates that the selected instances will be synthesised. However, you can unselect the check boxes of the device/interface/interface groups that you do not want to be synthesised.</p> <p>Points to remember when selecting and unselecting instances.</p> <ul style="list-style-type: none"><li>■ When you unselect device, the corresponding interfaces and their groups are unselected as well.</li><li>■ When you select device, the corresponding interfaces and their groups are selected automatically. If there is one interface listed under the device and you unselect the interface, the device and the interface groups are unselected automatically. However, if two interfaces listed under the device you can unselect any interfaces. You can also unselect and select the groups individually.</li></ul> <p><b>Note:</b> You can use the toolbar options, <i>Uncheck All</i> to unselect all the <i>Instance/Protocol Name</i> column checkbox corresponding to the instances and <i>Check All</i> to select all the check boxes.</p>
------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## Allegro FPGA System Planner User Guide

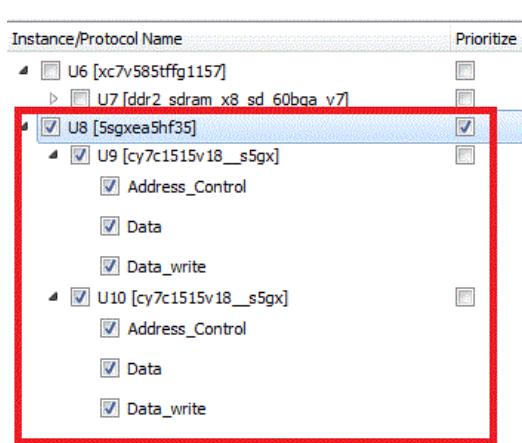
### Dialog Box Descriptions

---

Filter	<p>Use this option to search for specific instance in the Instance/Protocol column.</p> <p>Type an instance name that you want to find. FSP searches all the strings in the Instance/Protocol column to find instance name that matches with the specified name.</p> <p>The <i>Filter</i> search bar is regular expression enabled. You can use the alpha-numeric characters, underscore (_), question marks, or asterisks to perform search operation.</p>
--------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Prioritize	<p>Use this option to prioritize the order of processing instances.</p> <p>By default, the check boxes in the <i>Prioritize</i> column are unselected. This indicates that FSP will process the instances in random order.</p> <p>If you select the <i>Prioritize</i> check box corresponding to the device names, this indicates that FSP will process the instances in the order they are listed.</p> <p>For instance, in the following example FSP will run the U9 instance first and then U10 instance.</p>  <p>The screenshot shows a tree view of device and interface names. A red box highlights the 'Prioritize' column for the U8, U9, and U10 entries. Under U8, the 'U9' entry has its 'Prioritize' checkbox selected. Under U9, the 'Address_Control', 'Data', 'Data_write', and 'U10' entries have their 'Prioritize' checkboxes selected. Under U10, the 'Address_Control', 'Data', and 'Data_write' entries have their 'Prioritize' checkboxes selected.</p> <p>After selecting the <i>Prioritize</i> check box corresponding to the device names, you can also select the <i>Prioritize</i> check box corresponding to the interface names. In this case, FSP will first run all the interfaces whose <i>Prioritize</i> check box are selected and then will run the remaining interfaces whose <i>Prioritize</i> check box are not selected.</p> <p>You can also customize the priority order by moving the interfaces up and down. You can move the interfaces up and down by using toolbar options <i>Up</i> and <i>Down</i>. By default, these options are disabled. To enable, click on the interface name and then click <i>Up</i> or <i>Down</i>.</p>
------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

<b>Contiguous</b>	<p>Use this option to connect the interface pins to the device pins in a contiguous manner.</p> <p>In the <i>Die View</i> window, the I/O pins of Xilinx FPGAs are displayed in column wise and for Altera FPGAs the I/O pins are displayed into smaller rectangles, <i>tiles</i>. These pins and tiles are displayed based on the I/O pads information provided by the respective FPGA vendors.</p> <p>According to one of the Altera's IO DRC rules, the interface pins should connect to the pins whose tiles are adjacent to each other. In the following example, during synthesis FSP will use all the pins of tile 1 first and then move to tile 2, and vice versa.</p>  <p>You can select the <i>Contiguous</i> check box corresponding to the interface names or groups for which you want to enable contiguous connection.</p>
<b>Advanced</b>	<p>Use this option to set advance options for device instances.</p> <p>For more information, see the <a href="#">Advanced</a> section.</p>
<b>Save</b>	<p>Use this option to save the process settings in an external file.</p> <p>When you click this option, the <i>Save Process Options</i> window appears. Specify the name for the file and click <i>OK</i>.</p>

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Manage Process Options	<p>Lets you save and import the process option settings.</p> <p>When you click this option, the following options appears:</p> <ul style="list-style-type: none"><li>■ Save - Lets you save the process option settings. When you click this option, the <i>Save Process Option</i> dialog box appears. Enter a name and click <i>OK</i>.</li><li>■ Load - After you save the settings using the <i>Save</i> option, the name appears under this option. Select a name to load the process option settings.</li></ul> <p> <i>Important</i></p> <p>Be careful when you load an existing process option settings because the settings in <i>Process Options Editor</i> will be replaced by the settings of &lt;process_name&gt;. This means all the changes that you have made since the last time you saved the process options settings will be removed.</p> <ul style="list-style-type: none"><li>■ Delete - Lets you remove the saved process option settings. You can either remove the names individually, or remove all at once using <i>Clear All</i>. After you select a name, a confirmation dialog box appears. Click <i>Yes</i> to delete.</li><li>■ Rename - Lets you rename the saved process option settings. After you select a name, the <i>Rename Process Options</i> dialog box appears. Modify the name and click <i>OK</i>.</li></ul>
Run	Click to run synthesis. After you click <i>Run</i> , the <i>Log</i> window displays the progress report of the synthesis.
Cancel	Click to exit the window without running the synthesis.

## Advanced

The Advanced dialog box lets you specify various advance options such as preserve clock pins, Allocate Pair Pins Together, and constrained optimization to reduce the crossovers and number of layers.

### ***How to Access the Advanced dialog box?***

- Choose *Design – Run Design*.
- Click *Advanced* of device instance in Process Option Editor dialog box.

Name	Description
Prioritize	<p>Provides following options:</p> <ul style="list-style-type: none"><li>■ Nearest Groups Select to connect the groups of the interfaces having same bank or same clock region constraints, that are nearest from the targeted device instance with least number of crossovers. However, this option connects the groups with <i>Prioritize</i> check box first and then other groups.</li><li>■ Farthest Groups Select to connect the groups of the interfaces having same bank or same clock region constraints, that are farthest from the targeted device instance with least number of crossovers. However, this option connects the groups with <i>Prioritize</i> check box first and then other groups.</li></ul>
Preserve Clock Pins	<p>Use this option to preserve clock pins during synthesis.</p> <p><b>Note:</b> This option is disabled for Spartan devices.</p>
Do no defer allocation to special pins (VREF/VRP/VRN)	<p>Select this option if you want to give priority to the special pins such as VREF, VRP, VRN pins along with the IO pins for making connections at the beginning of the design run.</p> <p>If you unselect this option, the IO pins will have precedence over the special IO pins for making connections.</p>

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

Allocate Pair Pins Together	Select this option to connect the unconnected single-ended pins (P and N) of a device to the single-ended interface pins. Once the pins are allocated, the other differential pair pins (both P and N) of the device are used to connect to the remaining unconnected interface single-ended pins. This option controls the interface single-ended pins assignment by allowing to use least number of differential pair pins (both P and N) of the device, so that they can be used for true differential pair pins assignment later.
Allocate Single-Ended Signals to Single-Ended Pins	This option is visible when you have connectors on the Canvas. Select this option to connect single-ended signals of an interface to single-ended pins of the connector.
Constrained Optimization	Displays the connected device and interface instances in tree view. Select the instances to swap. Swap group and Swap Pin check boxes gets enabled when you select the instances. Select Swap Group option to swap the groups of instances. For more information on optimizing pin connections see section.  <b>Note:</b> This option is enabled when you assign the Assign Pin property for all the interface instance pins.
Define set of Instances to enable swapping identical groups	To swap the groups of an instance you need to define the instances in this pane.  Click + to add instance and click - to remove instance.
Check All	Use this option to select all the check boxes.
Uncheck All	Use this option to unselect all the check boxes.

## Properties

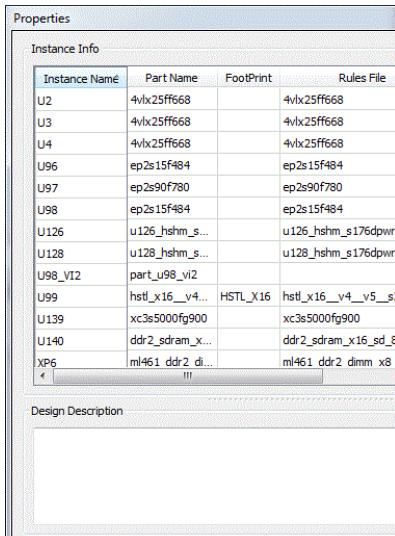
The Properties lets you view and modify the properties of the instances, groups, banks, and pins. The properties that appear in the Properties depends on the items you select in the Canvas or in Design Connectivity. For example, if you click on a pin, the properties of the selected pin appear in the Properties.

### **How to Access the Properties?**

- Choose *Window – Properties*.

The following section illustrates what appears in the Properties when you click or select in Design Connectivity and in Canvas.

### Design Information Page



This is the default view of the Properties. When you first time invoke the Properties or click the *Design [view name]* node, you see this page. This page displays the instance information and design description of the current design. The *Instance Info* displays a list of instance name and its associated information in a spreadsheet view. The *Instance Info* pane is a read-only pane. When you perform changes in the canvas or in any forms in FSP such as adding or deleting instance, editing instance names, rotating or flipping instances, or modifying the schematic symbol path, these changes are immediately reflected in the *Instance Info* pane.

The following table provides the description of each column in the *Instance Info* pane.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

Name	Description
Instance Name	Displays a list of instance names that are present on the canvas.
Part Name	Displays the part name of the instance.
FootPrint	Displays the footprint name.
Rules File	Displays the rules file name of the instance.
Mapping File	Displays the mapping file name of the instance.
Schematic Symbol	Displays the <i>lib:cell:view</i> structure of the instance.
Rotation	Displays the current rotation angle of the instance
Flip	Displays the flip status of the instance. Displays <i>No</i> if the instance is not flipped and <i>Yes</i> if flipped.

The *Design Description* pane lets you add information. The design information can include any text information such as parts information, board dimensions, and so on. This design information is saved in the project database so that you can see the information at any time when you invoke the *Properties*.

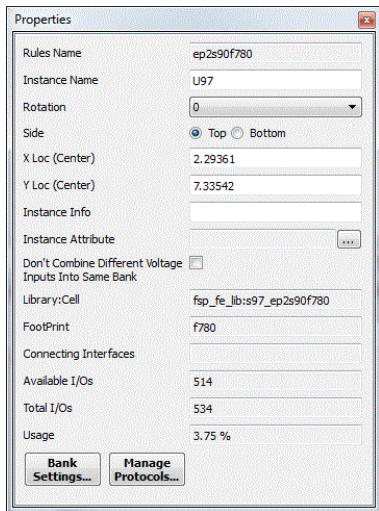
The Properties provides options to quickly access the *Project Settings* and *User Preferences* dialog boxes. These options are available at bottom left of the Properties.

**Note:** The two options, *Project Settings* and *User Preferences* options are not available when you are editing properties of the instances and pins in the Properties.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

#### Device Instance Properties



This page appears when you do any one of the following:

- Click the *Device Name = <inst\_name> part = <name>* row in the *Design Connectivity* window.
- On the Canvas, right-click on the device instance and choose *Instance Properties*.

You can use this page to modify the properties of the device instance. For example, you can change the instance name in the *Instance Name* field, add information in the *Instance Info* field, flip the device instance by selecting the options in the *Side* field, and more.

Name	Description
Rules Name	Displays the name of the part. <b>Note:</b> This is a read-only field.
Instance Name	Displays the name of the instance. You can modify the instance name.
Instance Info	Specify user-defined instance information. This is used to add any specific note to the instance. If configured in the Settings form, the specified information will be displayed on the canvas.
Side	Select to displays the top or bottom side of the instance on the PC board canvas.
Rotation	Specify the angle of rotation to rotate the instance.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

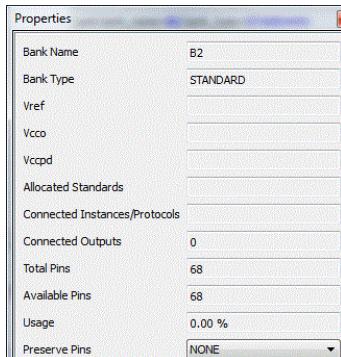
---

Custom Attributes	Click Edit button to invoke the Custom Attribute window.
Library:Cell	Displays the library and cell name. For OrCAD the complete path of the symbol directory is displayed.
Footprint	Displays the footprint name of the current instance.
Connecting Interfaces	Displays a list of interface instance names that are targeted to this device instance.
Total I/Os	Displays the total number of I/O pins of the device instance.
Available I/Os	Displays the number and percentage <Available I/O pins/Total I/O Pins> of the selected device instance.
Available HighSpeed Serial I/Os	Displays the number and percentage <Available HighSpeed Serial I/O pins/Total HighSpeed I/O Pins> of the selected device instance.  This option is visible if the selected device instance contains highspeed serial I/Os.
Use DCI Cascading	Select this option to enable the DCI cascading.
Center X	
Center Y	
Bank Settings	Click to invoke the <i>Bank Settings for Device Instance &lt;inst_name&gt;</i> dialog box.  The dialog box displays the bank details such as names of the banks, number of I/Os in each banks, and available I/Os of the selected device instance. In this dialog box, you can also set the maximum outputs and specify the power pins to preserve.
Manage Protocols	Click to invoke the <i>Manage Protocol</i> dialog box.  This dialog box lets you to create a protocol between the devices, modify or delete the existing protocol.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

#### Device Instance Bank Properties



This page appears by performing any one of the following:

- In the Design Connectivity window, expand the device level by clicking the plus sign (+) to the left of the row and click the *bank bank\_name = <name>* node.
- In the Canvas, press Shift and click on a pin from the bank that you want to select.

This page provides you the details of the selected bank. For example, reference voltage (VREF) is displayed in the VREF field, name of the I/O standard allocated to this bank appears in the *Allocated Standards* field and more.

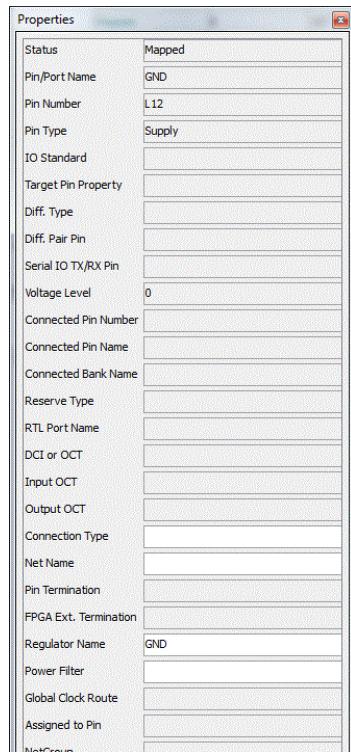
Name	Description
Bank Name	Displays the name of the selected bank.
Vref	Displays the vref voltage value of the selected bank.
Vcco	Displays the vcco voltage value of the selected bank.
Allocated Standards	Displays the name of the IO standard allocated in the selected bank.
Connected Instance/Protocols	Displays a list of interface instances names or protocol names connected to the selected device.
Available Pins	Displays the number and percentage <Available I/O pins/Total I/O Pins> of the selected bank of device instance.
Maximum Outputs	Specify the maximum number of output pins that can be used on a specific bank
Preserve Pins	Select the pins that you want to preserve for the bank from the drop-down list.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

#### Device Instance Pin Properties



This page appears by performing any one of the following:

- In the Design Connectivity window, expand the bank level by clicking the plus sign (+) to the left of the node and click any pin from the list.
- In the Canvas, double-click on a pin of a device instance.

This page provides you the liberty to modify the properties of the selected pin. However, not all the fields are editable. The columns that are editable in the Design Connectivity window, is also available as editable in this page. For example, RTL port name can be modified in *RTL Port Name* column in the Design Connectivity window and in *RTL Port Name* cell in this page.

Name	Description
------	-------------

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

Status	<p>Displays the connection status of the pin.</p> <p>For example,</p> <p><i>Allocated</i> is displayed, if the pin is connected to a signal after synthesis.</p> <p><i>Unallocated</i> is displayed, if the pin is not targeted or connected to any signal.</p> <p><i>Failed</i> is displayed, if the pin failed to connect to a signal after synthesis.</p> <p><i>Unmapped</i> is displayed, if the power pin is not mapped to any regulators.</p> <p><b>Note:</b> Click on the <i>Failed</i> text to find more information on the failure reasons.</p> <p><b>Note:</b> This is a read-only column.</p>
Pin/Port Name	<p>Displays the logical pin names of the interface instance or port names of the device instance.</p> <p><b>Note:</b> This is a read-only field.</p>
Pin Number	Displays the pin number.
Diff.Pair Pin	<p>Displays the differential pair pin of the pins of the instance.</p> <p><b>Note:</b> This is a read-only column.</p>
Serial IO TX/RX Pin	
Target Pin Property	Lets you target the selected pin to a device pin. For example, for Spartan 6 devices, if you want to target the selected pin to DCLK or CPDCLK, click and select DCLK from the drop-down list.
Diff Type	<p>Displays the differential type of the pins of the instance.</p> <p>For example, for positive clock pin name CLK_P it displays <i>Negative</i> and <i>Positive</i> for negative clock pin name CLK_N.</p> <p><b>Note:</b> This is a read-only column.</p>
Pin Type	Displays the pin type.
I/O Standard	Displays the name of the IO standards for the interface pins and for Device instance it displays for the pins to which the IO standards are connected.
Voltage Level	Displays the voltage value of the pin.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

<b>Reserve Type</b>	<p>The cell(s) under this column is enabled for Altera devices. When you click on a cell, the following values are displayed:</p> <ul style="list-style-type: none"> <li>■ AS BIDIRECTIONAL</li> <li>■ AS INPUT TRI-STATE</li> <li>■ AS OUTPUT DRIVING AN UNSPECIFIED SIGNAL</li> <li>■ AS OUTPUT GROUND</li> <li>■ AS DRIVING VCC</li> </ul> <p><b>Note:</b> These values are available for the pins of an interface that is targeted to an Altera device.</p> <p>These values allow you to specify the reserve state of all the unused pins on the Altera device. For example, select AS INPUT TRI-STATE to reserve the pins as tri-state input pins or AS OUTPUT DRIVING AN UNSPECIFIED SIGNAL to reserve the pins as output pins and drive any signal.</p> <p>FSP outputs these values to the quartus settings file (.qsf).</p>
<b>RTL Port Name</b>	<p>Displays the RTL port name for the pin. This name is automatically generated when you export the constraints.</p>
<b>DCI or OCT</b>	<p>Displays a list of DCI standards for the interface pins that are supported by the targeted device. Also, displays a list of OCT values supported by the targeted device.</p>
<b>Input OCT</b>	<p>This column is applicable for Altera devices.</p> <p>Displays a list of input OCT values supported by the Altera device. Select a value to apply on the pin.</p>
<b>Output OCT</b>	<p>This column is applicable for Altera device.</p> <p>Displays a list of output OCT values supported by the Altera device. Select a value to apply on the pin.</p>

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

<b>Connection Type</b>	<p>Use this column to define the external ports for the pins and nets. The cell of this column provides a list of pre-defined external ports. However, this list vary based on the instance, instance pin, and the connection.</p> <p>The following options are available:</p> <ul style="list-style-type: none"> <li>■ Extend as External Port This option is available for the interface pins (connected or not connected).</li> <li>■ Do Not Connect This option is available for the interface pins (not connected).</li> <li>■ Fixed External Port This option is availabe for interface pins (not connected) and device power pins (not mapped) .</li> <li>■ Fixed Internal Connection This option is availabe for interface pins (not connected) and device power pins (not mapped) .</li> <li>■ Preserve Pin This option is available for device pins.</li> </ul>
<b>Net Name</b>	<p>Displays the net name</p>
<b>Pin Termination</b>	<p>Lets you add termination to the interface pin.</p> <p>The termination names defined in the <i>Define Termination</i> dialog box are displayed in this column cells.</p> <p>Double click on cell and select termination name from the drop-down list.</p>
<b>FPGA Ext. Termination</b>	<p>Displays a list of termination names that are defined in the <i>Define Termination</i> dialog box.</p> <p>Double-click on a cell and select a termination name from the drop-down list.</p>

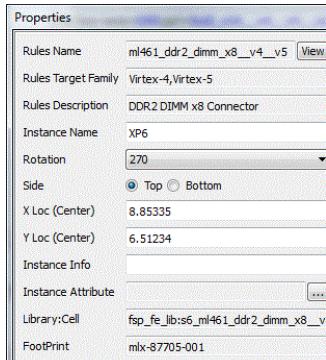
## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

Regulator Name	<p>Display the list of power regulators.</p> <p>Click and select a voltage regulator name from drop down list.</p> <p><b>Note:</b> If you are applying a power filter termination to a pin then you must select a power regulator.</p>
Power Filter	<p>Displays the list of power filter termination names defined in the <i>Define Termination</i> dialog box.</p> <p>Select a power filter name from the drop down list.</p>
Global Clock Route	This column is applicable for Spartan family devices.
Assigned to Pin	<p>This option lets you to connect the interface pin to FPGA pin. When you click the cell, displays the list of targeted FPGA pins available. Select a pin from the list.</p> <p><b>Note:</b> To see the list of FPGA pins, you should target the interface to FPGA first.</p>
NetGroup	<p>Displays the NetGroups.</p> <p>NetGroups are automatically defined under this column, when you select the <i>second</i> option in the NetGroup tab of the <i>Settings</i> dialog box and place the interface component on the canvas. NetGroups are created based on the interface logical group properties.</p> <p>Under this column, you can modify, redefine, or remove NetGroups.</p> <p>For more information, see the <a href="#">NetGroup</a> section.</p>
Custom Attributes	Double click on cell and click <i>browse (...)</i> button to invoke the Custom Attribute window.

### Interface Instance Properties



## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

This page appears when you perform any one of the following steps:

- In the Design Connectivity window, click the *interface name = <inst\_name> part = <name>* node.
- In the Canvas, right-click on the interface instance and choose *Instance Properties*.

This page provides you the information about the selected interface. For example, rules name in the *Rules Name* field, device name to which the selected interface is targeted in the *Rules Target Family* field and more. You can also use this page to modify the properties of the instance. For example, modify the instance name in the *Instance Name* field, flip the selected interface by selecting options in the *Side* field.

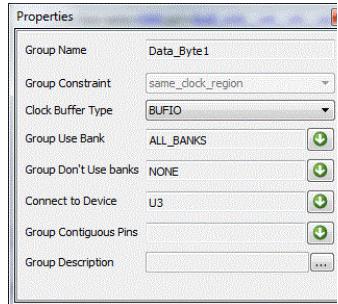
Name	Description
Rules Name	Displays the name of part.  <b>Note:</b> This is a read-only field.
View Rules	Click to view the model properties of the instance in read-only mode.  You can view the Logical constraints, schematic view, and layout (pin footprint) view of the instance.
Instance Name	Displays the name of the interface instance. You can modify the instance names.
Instance Info	Specify user-defined instance information.  This is used to add any specific note to the instance. The instance information is displayed on PC Board the canvas.
Side	Displays the top or bottom side of the instance on the PC board canvas
Rotate	Specify the angle of rotation to rotate the instance.
Custom Attributes	Click Edit button to invoke the Custom Attribute window.
Library:Cell	Displays the library and cell name.
Footprint	Displays the footprint name of the instance.
Center X	
Center Y	

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Group Settings	<p>Click to invoke the <i>Group Settings For Interface Instance &lt;inst_name&gt;</i> dialog box.</p> <p>This dialog box provides the following options:</p> <ul style="list-style-type: none"><li>■ Connect to Device<ul style="list-style-type: none"><li>Use this option to target the group to device instance.</li></ul></li><li>■ Use Banks<ul style="list-style-type: none"><li>Use this option to target the group to a specific bank of the targeted device.</li></ul></li><li>■ Don't Use Bank<ul style="list-style-type: none"><li>Use this option to specify the bank of the targeted device to which you do not want the group to be connected.</li></ul></li></ul>
----------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### Interface Instance Group Properties



This page appears when you perform any one of the following step:

- In the Design Connectivity, when you expand the interface level by clicking the plus sign (+) to the left of the node and click the *group group\_name = <name>* row.
- In the Canvas, press and hold Shift key, and click on a pin of interface instance.

This page lets you modify the properties of the selected group. For example, specify the bank name to which the group you want to target in the *Group Use Bank* field, add description about the group in the *Group Description* field and more.

Name	Description
------	-------------

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Group Name	Displays the name of the selected group.
Group Constraint	Select a constraint from the drop-down list to apply constraints on the pins of the selected group.
Clock BUFR Type	This field is applicable to Spartan 3/3A devices.
Group Use Bank	<p>This option lets you target the interface groups to device instance banks of your own choice. You can also exclude or preserve the banks from making connections using this feature.</p> <p>To use the banks:</p> <ul style="list-style-type: none"><li>■ Click the drop down button and select the bank numbers from the list and click <i>OK</i>.</li></ul> <p>After clicking OK, the selected bank names are displayed in <i>Use Bank</i> text box. These selected banks will be used further during synthesis.</p>
Group Don't Use Banks	This option lets you select the banks that you wish to preserve or exclude from connections.
Connect to Device	Click and select one or more instances names from the list to which you the selected group to be connected.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

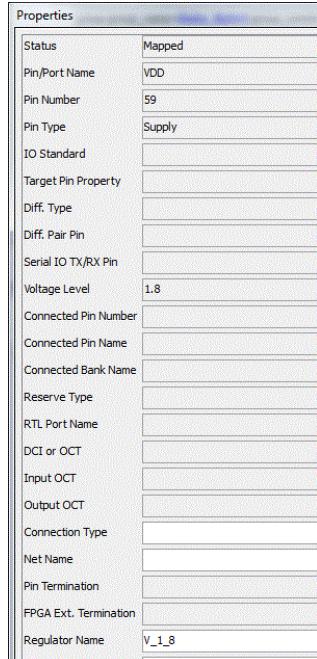
Contiguous Allocation	<p>Lets you setup or customize the settings for contiguous or interleaving allocation.</p> <p>When you click the drop-down button, a pane appears. You use this pane to setup or customize the contiguous or interleaving allocation.</p> <p><b>For Contiguous Allocation</b></p> <p>Select a bus from the left side of the pane and click -&gt; button to move the selected bus to the right side of the pane.</p> <p><b>Note:</b> For contiguous allocaton, you can select a single bus.</p> <p>After moving the bus to the right side of the pane, click <i>OK</i>. For more information on contiguous allocation, see the <i>Applying Synthesis Options</i> section in the <i>Running the Design</i> Chapter.</p> <p><b>For Interleaving Allocation</b></p> <p>Select two buses from the left side of the pane and click -&gt; to move the selected buses to the right side of the pane.</p> <p><b>Note:</b> For interleaving allocation, you should specify atleast two buses.</p> <p>After moving click <i>OK</i>.</p> <p>Consider you select the buses, read[0:9] and write[0:9] for interleaving allocation. When you run the design, FSP allocates the buses signals in the following:</p> <p>read[0] write[0] read[1] write[1] . . . . read[9] write[9]</p>
-----------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Group Description	Click <i>browse(..)</i> to specify the description for the group.
----------------------	-------------------------------------------------------------------

### Interface Instance Pin Properties



This page appears when you perform any one of the following step:

- In the Design Connectivity window, expand the interface group level by clicking the plus sign (+) to the left of the row and click any pin from the list.
- In the Canvas, double-click on a pin of the interface instance.

This page lets you modify the properties of the selected pin. However, not all the fields are editable. The columns that are editable in the Design Connectivity window, is also available as editable in this page. For example, RTL port name can be modified in *RTL Port Name* column in the Design Connectivity window and in *RTL Port Name* cell in this page.

## Preferences

The Preferences dialog box lets you specify various options in FSP.

### **How to Access the Setting User Preference dialog box?**

- Choose *File - Preferences*.

### Display Tab

Use the Display tab to specify the canvas and schematic settings, change the canvas color, and change the font size.

Name	Description
Canvas Settings	
Grid Dots (per inch)	Specify the value of grid dots per inch in the Canvas view. Maximum Grid dots per inch allowed are 20 and the minimum are 0.
Resolution	Select the appropriate radio button to set the screen resolution. You need to restart FSP for the resolution changes to take effect.
Graphics System (Linux OS Specific)	This option is enabled then only when you invoke FSP in linux.
Default	Select this option to reset to the original graphics settings.
Raster	This option improve the graphics quality for example font and colors.
Color Scheme	Select the appropriate radio button to set the background color.
Advanced	Click to set the custom colors for various things.  The Advanced dialog box appears where you can set the colors for Grid, Global net, Normal net, selected net, selected pin, instance border, instance info border, instance info text, instance info selected, symbol border, symbol pin, symbol net, symbol offpage connector, symbol pin name text, symbol pin number text, symbol net name text, symbol name text, and symbol net selected.
Instance Data Display	

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

Instance Name	Select this check box to display the instance name along with the instance on the canvas.  For example, when you select this check box, the application displays U10.
Part Name	Select this check box to display the name of the part along with the instance on the canvas.  For example, when you select this check box, the application displays Cypress:CY7C1315BV18[S2, S2_GX].
Instance Info	Select this check box to display instance information along with the instance on the canvas.  For example, when you select this check box the application displays the instance information you have specified.
Pin Number	Select this check box to display the pin number for each instance pin on the canvas.
Font Size	Select the appropriate radio button to set the font size for the part name, instance name, and instance information.
<b><i>Misc Settings</i></b>	
Show alignment hint while moving component on canvas	Lets you align the instance with other instances on canvas.
Show tooltip for instance, pin or net	Select this option to display the tooltip for instance pin and nets.
Highlight pins marked as Do Not Connect (or Preserve) on Canvas	Select this option to highlight the pins that are marked as Do Not Connect on the package view of the instance on Canvas.
Use Group or Bank color to draw nets on canvas	Select this option to apply the colors of the group or bank on the nets.
Show Fixed Intern/Extern nets on canvas	Select this option to see the fixed intern and extern on the canvas.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

## Design tab

Use the Design tab to change the license, enable and disable the design and schematics settings.

Name	Description
Design Settings	
Copy pin and group/bank level properties while copying instances	<p>Select this option if you want the instance-level settings such as group, pin, and bank-level properties of the original instance to be pasted on the new instance.</p> <p>After selecting this option, choose <i>right-click on the instance – Paste</i> to paste a new instance on the canvas with instance-level settings.</p> <p><b>Note:</b> The <i>Paste</i> command pastes a new instance on the canvas with default settings, if this option is unchecked.</p>
Generate Report for not connected signals after synthesis.	Select this option to generate a detailed report for not connected signals after synthesis.
Auto switch to “Synthesis Failure Pins” view after synthesis failure	Select this option to automatically switch to the <i>Synthesis Failure Pins</i> tab of the <i>Messages</i> window when any error occurs during synthesis.
Report skipped interface or protocol from processing	Select this option to skip the report of the interface or protocol prevented from proceeding
Show wizard while placing part on canvas.	Select this option to invoke the Create/Select Mapping and Rules file wizard.
Report warning for power pins without voltage specification	Select this option to report warning for the power pins that does not have voltage specified to it.
<b>Schematic Settings</b>	

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Skip namespace validation while generating symbols and schematics	Select this option to skip the namespace validation.
<b><i>Confirmation Dialog Settings</i></b>	
Display Confirmation dialog when opening a design created with a newer version of FSP.	Select this option to display the confirmation dialog box when you open the design created using older version with newer version of FSP.
Display Confirmation dialog when editing a design created with a newer version of FSP.	Select this option to display the confirmation dialog box when you try to edit the design with newer version of FSP.
Display confirmation dialog when editing a component created with newer version of FSP	Select this option to display the confirmation dialog box when you try to edit a component created with newer version of FSP.
Display Fully Validated Part Information	Select this option to display the fully validated part information.
Use the Misc tab to enable the confirmation window during opening and editing the design with newer version of the tool. For more information on model and design versioning see <a href="#">“Managing Model and Design Versions”</a> on page 537 chapter.	

## General Tab

Use the General tab to modify the application font settings, enable and disable the log window settings and misc settings.

Name	Description

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

Application Font Settings	Displays the current family and size of the font. Click the button to change the font size and font family.
XML Editor	Displays the current family and size of the font. Click the button to change the font size and font family.
Canvas	Displays the current family and size of the font. Click the button to change the font size and font family.
Log Window	Displays the current family and size of the font. Click the button to change the font size and font family.
Reset All Fonts	Click to apply the default (original) font settings.
<b><i>Log Window Settings</i></b>	
Display Time In Log window	Select this option to display the time in Log window
Display Smart colored report in log window	Select this option to display the colored report in the log window. You can also enable this feature by a directly right click on the message pane and selecting Show color or press Ctrl+K
<b><i>Misc Settings</i></b>	
Open the most recent project on invocation.	Select this option to open the most recent project when you reopen the FSP.  To invoke this action, the user needs to reopen the tool.
Show text below icon in XML Editor	This option will display the text below the icons in all XML editor window.
Highlight links in “Synthesis Failure Report”	Select this option to highlight the links that are displayed with report in the <i>Synthesis Failure Report</i> tab.
Auto resize spreadsheet editor columns to contents	This option lets you to resize and save the spreadsheet editors column width size.  For example, select this option and invoke any spreadsheet editor from FSP. Select a column and resize the column width size using mouse and click OK. Reinvoke the spreadsheet editor, the new column width size is saved.  Unselect this option if you do not want to save the column width size.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

Show mini toolbar	Select to access the mini toolbar icons when you click on the rows (or headers) in the <i>Design Connectivity</i> window.
Display Design Connectivity Window 'Net View' content in single row	Select this option to view the net information in a single row.
Create file back up while overriding files	Select to create a copy of the current files before overriding the files.
Design snapshot interval	<p>This feature helps you to save a copy of your design at each interval of time. After you enable this feature and specify minutes, the design is saved in a different file after each interval of the specified minutes. The design files are saved in the current project directory.</p> <p>You can access the saved design files through <i>Projects – Snapshots</i> menu.</p> <p>Select this option to enable the feature. Specify the minutes, after which the design need to be saved in a separate design file.</p>
Recent Project Entries	<p>Enter number of recent project entries you wish to see.</p> <p>Example, if you enter ten as recent entries then ten last recent project that you worked on are displayed in <i>File – Recent Projects</i>.</p>
Ok	Click to save the preferences.
Cancel	Click to cancel the operation.

## Remove Net Confirmation

The Remove Net Confirmation dialog box lets you remove all unlocked nets, power nets, fixed external ports, and fixed internal ports from the design. The options in this dialog box are enabled according to the nets existence in the design.

**Note:** Once the nets are deleted the operation cannot be undone.

### **How to Access**

- Choose *Design – Delete All Nets*. Or else click the *Delete All Nets* icon in toolbar.

Name	Description
Remove All Nets	Select this option to remove all unlocked nets in the design.
Remove All Power Nets	Select this option to remove all power nets in the design. <b>Note:</b> This option is disabled if no power nets are assigned in design.
Remove All Fixed External Ports	Select to remove all the fixed external ports.
Remove All Fixed Internal Pots	Select to remove all the fixed internal ports.

## Resolve Net Name Conflict

The Resolve Net Name Conflict dialog box appears while running the design if any conflicts are present in the design nets. Use this dialog box to resolve the design net names conflicts.

### ***How to Access the Resolve Net Name Conflicts***

- Choose *Design – Run Design*.

The Resolve Net Name Conflict dialog box is displayed if net name conflicts exists in the design nets.

Name	Description	Remark
split Name	Displays the design name whose net name is conflicting.	
Original Name	Displays the net name that is conflicting while the designs are merging	
Changed Name	Displays the updated net name.  The net names that are conflicting appear in red.	
Search and Replace	Use this feature to search and replace the net names by entering strings.	
Select Split	Select the designs to view net names that are conflicting.  When you select a design, the net names specific to the selected design appear.	
Search String	Enter the search string.	
Replace String	Enter the replace string to replace the net name.	
Apply	Click to change the net names with the replaced string.	
Show Only Conflicting Signals	Select this check box to show only the conflicting net names.  When this check box is not selected, all the net names for the selected design appear.	

## **Allegro FPGA System Planner User Guide**

### Dialog Box Descriptions

---

OK	Click to save the changed net names.	
Cancel	Click to cancel the operation.	

## Rules File Path Editor

The Rules File Path Editor dialog box displays the library names and their paths fetched from the config.ini file. These library names and paths are displayed in grey color, that is, read-only fields. The Rules File Path Editor dialog box also lets you specify the search paths used by FSP to look for rules and mapping files and add rules files in the Libraries. You can even add your own rules files to the Libraries using the Rules File Path Editor dialog box. You can use both relative and/or absolute paths to specify directory and rules file name.

You can also set the lrf working directory for your project in the Rules File Path Editor dialog box. A lrf working directory is the directory in which the rules and mapping files you create during the design or the default rules and mapping files generated by FSP will be automatically saved in the working directory.

### **How to Access**

Do one of the following:

- Choose *Library – Edit Rules File Path*.
- Choose *File – Settings*. Click *Edit* in Design Settings pane.

Name	Description
Add	Click to invoke windows Browse For Folder dialog box. Select the folder and click <i>OK</i> .
Delete	Select a row and click this option to delete the path specified in the dialog box.
Move Up	Click this option to move the rules file path upside and give first preference while fetching the rules file.
Move Down	Click this option to move the rules file path down and give lower preference.
Set Working Dir	Lets you set the working directory. Select an entry from the list and click this option.
OK	Click OK to save the settings.
Cancel	Click Cancel to cancel the operation.

## Rules Editor

The Rules Editor dialog box lets you to create your own logical rule file for your design. This dialog box also helps you to edit the existing logical rule file. You can also save the logical rule file and reuse it later. There are various options available in this dialog box which helps you to quickly create your own logical rule file.

### Cross-Probing between Tree View (Editor) and Graphics View

The *Rules Editor* form maintains the association between these two views. When you make any changes in the Tree view, such as modifying the direction of a pin, merging groups, or moving one pin from one group to another group, the changes are updated in the Graphics view as well.

Besides updating the changes between the two views, you can also navigate a pin, group, or bank between the views. When you select a group or a bank in the Tree view, the graphical representation of the selected group/bank is displayed in the Graphics view. Similarly, when you select a pin in the Tree view, the respective pin is displayed and highlighted in the Graphics view. In the same way, when you select a pin in the Graphics view, the respective pin is displayed in the Tree view.

For detailed information on how to create logical rule file using various options see [Creating Parts](#) chapter.

#### How to Access

Do one of the following:

- Choose *Library - Create Part - Interface*.

The Rules Editor dialog is displayed. You use this dialog box to create interface logical rule file.

- Choose *Library - Create Part - Connector*.

The Rules Editor dialog box appears. Use this dialog box to create logical rule file.

Name	Description
Signal Count	Displays the number of pins (Cells) available in this editor.
Signal Name	Enter a signal name.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

Pin Number	Enter a pin number.
Pin Type	<p>Click the drop down list to select the pin type. You can select any one of the following pin type:</p> <ul style="list-style-type: none"> <li>■ Input</li> <li>■ InOut</li> <li>■ Output</li> <li>■ NoConnect</li> <li>■ Supply</li> </ul>
Voltage Level	This is a read only field and displays the voltage value. This field is update based on the IO standard selection.
IO Standard	Click the drop down list to select the IO standard. The IO standards list are customized based on FPGA family selected to target in Edit Properties dialog box.
Target Pin Property	<p>Click the drop down list to select the target pin property. The target pin properties are customized based on the FPGA family selected.</p> <p>The target pin properties lets you to connect the selected interface pin to FPGA pin. For example: a pin that must connect to a clock capable pin needs to be tagged as such.</p>
Diff Type	Click the drop down list to select the differential type.
Diff Pair Pin	For more information on diff pair pin see Auto Detect Pin Pair section.
Serial IO TX/RX Pin	For more information on diff pair pin see Auto Detect Pin Pair section.
Clock Group	This field is enabled for Spartan3/3A devices. Click the drop down list and select the clock group entries. For example 0 and 90.
X and Y locations	The X and Y location values decides the exact location of the pin in the package. The field values are updated by importing dra file. Specifying the values manually is time consuming.
Symbol Pin Name	Specify symbol pin name.
Description	<p>You can assign some note or description for the pin. Perform the below steps to add description:</p> <ul style="list-style-type: none"> <li>■ Click on the cell. A <i>browse (...)</i> button is displayed.</li> <li>■ Click <i>browse (...)</i>. The Add Description dialog box is displayed.</li> <li>■ Add content in the dialog box and click OK.</li> </ul>

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Custom Attribute	Use this column to add custom autarkies to the symbol pin.
Rules File	Enter the name of the rules file in which you want to save the pin information.
...	Click to browse to the location where you want to save the file.
Import From	Displays five sub menu options. These menu's helps you to import the logical rule file pin details from various resources. For more information, see the <a href="#">Creating Parts</a> section.
Save As	<p><i>For component-instance which is not linked to symbol</i></p> <p>Use this option to save the logical rules file in the working directory or in rules file directories.</p> <p>The <i>Save Rules File</i> dialog box is displayed when you click this option. Enter the name for the rules file that you are creating in the <i>Rules File</i> field and select a directory from the <i>Output Directory</i> drop-down list.</p> <p><i>For component-instance which is linked to symbol</i></p> <p>Use this option to save both logical rule and mapping files in the working directory or in rules file directories.</p> <p>The Save Rules File dialog box is displayed when you click this option. Enter the name for the rules file that you are creating in the <i>Rules File</i> field and enter the name for the mapping file in the <i>Mapping File</i> field. Select a directory from the <i>Output Directory</i> drop-down list.</p> <p>Click <i>OK</i> to save rules and mapping files. When you click <i>OK</i> the rules and mapping files are saved in the specified output directory.</p> <p><b>Note:</b> Rules and Mapping files can be saved only in the working directory (set in the Rules File Path Editor dialog box) or any rules file directories.</p>

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

Save	<p><i>For component-instance which is not linked to symbol</i></p> <p>Use this option to save the logical rules file in the working directory or in rules file directories.</p> <p>The <i>Save Rules File</i> dialog box is displayed when you click this option. Enter the name for the rules file that you are creating in the <i>Rules File</i> field and select a directory from the <i>Output Directory</i> drop-down list.</p> <p><i>For component-instance which is linked to symbol</i></p> <p>Use this option to save both logical rule and mapping files in the working directory or in rules file directories.</p> <p>The Save Rules File dialog box is displayed when you click this option. Enter the name for the rules file that you are creating in the <i>Rules File</i> field and enter the name for the mapping file in the <i>Mapping File</i> field. Select a directory from the <i>Output Directory</i> drop-down list.</p> <p>Click <i>OK</i> to save rules and mapping files. When you click <i>OK</i> the rules and mapping files are saved in the specified output directory.</p> <p><b>Note:</b> Rules and Mapping files can be saved only in the working directory (set in the Rules File Path Editor dialog box) or any rules file directories.</p> <p><b>Note:</b> Both <i>Save As</i> and <i>Save</i> options invokes the similar type of dialog box. Once you save the part, the <i>Save</i> option instantly save the part definitions in the file without opening the dialog box.</p>
Show Package View	Click to view the package view of the component.
Show Log	Click to view/display the message log window.
Validate	Use this option to verify the logical details you have entered in this window.
OK	Click OK to save the logical rule file details.
Cancel	Click Cancel to exit the dialog box.
<b>Package View</b>	
Move Selected Pins to Group	<p>Lets you move the selected pin to another group. When you hover the mouse pointer on the pin, a list of group names appears.</p> <p>Right-click on a pin and choose <i>Move Selected Pins to Groups – &lt;group name&gt;</i>.</p>

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

First header node level  <pre>lrf interface_type= NormalInterface ...</pre>	<p>Displays the properties of the component. You can revise the properties details through Edit Properties dialog box. There are few more pop-up menu options are available which are useful while creating the parts. They are:</p> <ul style="list-style-type: none"> <li>■ Edit Properties</li> <li>■ Edit Custom Attributes</li> <li>■ Auto Detect Pin Pair</li> <li>■ Add Group</li> </ul>
Second header node level  <pre>group group_name= group1....</pre>	<p>Displays the group properties. You can add any number of groups or revise the existing group properties through Add group dialog box. There are few more pop-up menu options are available which are useful while working with groups.</p> <ul style="list-style-type: none"> <li>■ Edit Group</li> <li>■ Add Signal</li> <li>■ Add Bus</li> <li>■ Edit Custom Attribute</li> </ul>

### Searching and Filtering in the Rules Editor.

The Find feature lets you to find any keywords and strings in the any of the editors of FSP. You may also enter any search string, including wild cards '\*' and '^' as search criteria. Many filtering options are also provided in the Find feature to limit the search. You can also specify the forward and backward directions of the search.

**Note:** The *Find* feature is available in the *Rules Editor*, *Rules Instance Editor*, and *FPGA Port and Use Pin Mapping for Device Instance* dialog boxes.

### How to Access

- Click *Find* icon or press Ctrl + F.

Name	Description...
Search text box	Enter the string that you are searching for.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

Look In Column	Click and select the column names from the drop down list in which you want to search keyword.
Regular Expression	Select this option to apply regular expression search operation.
Find	Click to start the search operation.

### Examples of Find

This section shows few examples to perform search using the Find feature using Regular Expression.

#### ***Example1***

To search the strings that starts with IO\_, such as IO\_L7N\_A4\_D20\_1, IO\_L7P\_A5\_D21\_1 and more, you can use the either regex IO\_\w+\_d\$ or IO\_.

'IO\_ ensures that the strings must start with IO\_.

\w+ subpattern matches all the alphabetic character and \_ character, such as L7N\_A4\_D20, L7P\_A5, D21\_.

\d\$ subpattern matches the numeric character (\d) and \$ matches the numeric character at the end of the string, such as 1.

#### ***Example2***

To search the strings that contains special characters such as PD11\_RXENB\_MII-TX-ERR, you can use the regex PD\d+.\*.

'PD' ensures that the strings must start with PD\_.

'\d+' subpattern matches numeric character after PD, such as PD11, PD22 and more.

.\* characters matches any character expect newline character zero or more times, RXENB\_MII-TX-ERR.

#### ***Filtering the Display of Information***

The Rules Editor provides a filter bar that lets you to filter the display of information shown in the Rules Editor.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

The *Filter* bar is regular expression enabled. You can use the alpha-numeric characters, underscore (\_), question marks, or asterisks to filter the display of information. You must select the *Reg Exp* option in the *Filter* bar to enable the regular expression.

The *Filter* bar provides three drop-down combo boxes that shows all the valid values and column names which you can use to filter out the information.

The following table describes the usage of each drop-down combo boxes:

- 
- 1 Enter the text that need to be matched and displayed in the editor.  
You can also use the alpa-numeric characters, underscore(\_), asterisks (\*) to perform regular expression based search operation. To enable the regular expression, select *Reg Ex* option.
  - 2 Select a column name from the drop-down list in which you want to search the desired text.  
For example, type DDR\_CS in the search text box and select *Pin Name* from this drop-down list, FSP displays only those rows whose *Pin Name* column cells contains DDR\_CS text.
  - 4 Select this option to enable the regular expression based search.
-

## Rules Viewer

The Rules Viewer dialog box displays you a spreadsheet view of the logical and electrical constraints of the component. The Rules Viewer is a read only spreadsheet view of the component details as it appears in Rules Editor form. It displays all the group constraints and pin constraints required for a component. You cannot edit the fields in Rules Viewer dialog box. The Rules Viewer form can only be used to verify whether you have assigned the correct pin and group information details in Rules Editor dialog box.

### **How to Access**

- Right-click on the interface instance and choose *Rules – View*.

Name	Description
Show Package View	Use this option to view the package view of the component.
Close	Click Close to exit the dialog box.

## Rules Instance Editor

The Rules Instance Editor dialog box gives you a spreadsheet view through which you can modify the logical constraints of the logical rules file which is placed on the canvas. This dialog box is useful for the last type modification in the component. There is also a provision for saving the logical rule file. There are various options available in this dialog box which helps you to quickly modify the placed logical rules file.

For detailed information on how to create logical rule file using various options, see the [Creating Parts](#) chapter.

### **How to Access**

Do one of the following:

- Choose *Library - Create Part - Interface*.

The Rules Editor dialog is displayed. You use this dialog box to create interface logical rule file.

- Choose *Library - Create Part - Connector*.

The Rules Editor dialog box is displayed. You use this dialog box to create logical rule file.

**Note:** The Name of the Rules Instance Editor dialog box is very much similar to the Rules Editor. For more information see [Rules Editor](#) section. The below table describes only the options which are not present in the Rules Editor form.

Name	Description
Pin type	Display the pin type available in rules file. Click and select the drop down list to change the pin type.
Symbol Pin Type	This column appears only for real components.  Display the pin type available in symbol file. The pin type specified in Pin Type column wins over the pin type available in this column. For more information see <a href="#">Mapping Considerations</a> section.

## Rules Signal Mapper

The Rules Signal Mapper helps you to map the signal names of the logical model to symbol pin names. You can choose to manually map the signal names or use the Automap option to map all the signals at one go. After mapping you can save the information in file called Mapping File (.lmp).

For detailed information on mapping file see [Creating Mapping File](#) section.

### How to Access

Do any of the following:

- In the Add Part dialog box, click *Define Mapping*.
- In Convert Rules File Instance to Real Part, click Define Mapping.

Name	Description
Hide Signals After Mapping	Select this option if you do want to see the signal names in Rules File Data pane after mapping.
Auto Map Bus Signals	Select this option to auto map all the signals at one go.
Part Symbol Data	
Symbol Pin Name	Displays the pin names of the selected symbol.
Pin Type	Displays the pin types of the selected symbol.
Diff.Type	Displays the differential type pair.
Voltage Level	Displays the voltage value.
Target Pin Property	Click the drop down list to select the target pin property. The target pin properties are customized based on the FPGA family selected.  The target pin properties lets you to connect the selected interface pin to FPGA pin. For example: a pin that must connect to a clock capable pin needs to be tagged as such.
Signal Name	You need to drop the signals in this column, of the Signal Name column of the Rules File Data pane.
Rules File Data	

## **Allegro FPGA System Planner User Guide**

### Dialog Box Descriptions

---

Signal Name	Displays the signal names of the rules file.
Pin Type	Displays the pin type information in rules file.
Diff Type	Displays the differential pin pair information in the rules file.
Mapping File	Enter the name for the mapping file that you want to create.
Browse (...)	Click this to browse to the location where you want to save the mapping file.
Validate	Click this to validate the mapping information.
Show Log	Click this to display the log window at bottom side of the dialog box.
Save	Click this option to save the mapping file.
Cancel	Click this option to cancel the operation.

## Select Instance

The *Select Instance* dialog box lets you select a instance from the instance names list. The instance names in this dialog box are displayed based on the task you are working on. For instance, if you want to target groups through *Components – Target Groups*, the *Select Instance* dialog box lists all the interface instances names that are present on the Canvas.

### **How to Access**

The *Select Instance* can be invoked by performing any one of the following steps:

- Choose *Component – Target Groups*.
- Choose *Component – Bank Settings*.
- Choose *Component – Add Protocol*.
- Choose *Component – Add Virtual Interface*.
- Choose *Component – Map FPGA Port and Pin Assignment*.
- Choose *Component – Import Constraints/Pin Assignment*.
- Choose *Component – Export Constraints/Pin Assignment*.
- Choose *Component – Map FPGA Resources*.

Name	Description
Ok	After selecting an instance name from the list click this button.
Cancel	Click to exit the window.

## Select Rules File

The Select Rules File dialog box lets you select the rules file from the rules file directories.

### How to Access

The Select Rules File can be seen in various forms. Few of them are:

- Choose *Library - Edit Part - LRF* to display the Select Rule File dialog box.
- Choose *Library - Create Part - Interface*. The Rules Editor dialog box is displayed. Click *Import From* and select *Existing LRF*.
- Choose *Library - Create Part - Connector*. The Rules Editor dialog box is displayed. Click *Import From* and select *Existing LRF*.

Name	Description
Part	Displays the list of part names supplied by the Cadence.
Target Families	Displays the FPGA family name to which the selected component is targeted.  For example: <code>ddr2_dimm_x4_a2gx A2GX</code>  This means the ddr2_dimm_x4 component is pre-targeted to Ariall GX devices.
+	Click to display the part names.
-	Click to minimize the tree view structure.
<Type Filter Text Here>	Enter a specific reg exp to display only specific interface library part name.  For example: <code>ddr2*</code> to display all the components which contains the keyword dd2. <code>*cy*</code> to display all the cypress components.
Refresh	Click to refresh the complete library lists.
OK	Click OK to import/drop the selected logical rule file.
Cancel	Click Cancel to exit the window.

## Save Rules File

The Save Rules File dialog box lets you save the part definitions specified by you in the Rules Editor. You are allowed to save the rules file only in the working directory and in any rules file directories.

### ***How to Access the Save Rules File***

- In Rules Editor, do the following:
  - Click *Save* if you are saving the model first time.
  - Or
  - Click *Save As*.

### **Points to Remember..**

You must remember the following points before saving the part definition:

- For component-instance which is not linked to symbol  
Use this option to save the rules file in the working directory or in any rules file directories. After clicking this option the log window displays the directory path where the rules file is saved.
- For component-instance which is linked to symbol  
Use this option to save the rules and mapping files in the working directory or in any rules file directories. After clicking this option the log window displays the directory path where the rules and mapping files are saved.
- No constraint validation check is performed while clicking this option.

Name	Description
Rules File	Enter the name for the rules file that you are creating.
Output Directory	Select a directory from the drop-down list, where you want to generate the rules and mapping files.
OK	Click to save the rules file.
Cancel	Click to exit the dialog box.

## Swap Groups

The Swap Groups dialog box lets you swap the signals with in an interface or different interfaces. By using the Swap Groups dialog box you can minimize the average rats nests crossings.



You can swap only those groups that have same group constraints.

### ***How to Access the Swap Groups dialog box?***

- In Design Connectivity, right click on the interface group row and choose *Swap Groups*.

Name	Description
Displays a list of groups names that are swappable with the selected group.	Click on a group name that you want to swap with the selected group.
Color Text Boxes	Click to invoke the <i>Select Color</i> dialog box. Use this dialog box to apply color signals to the group.
Swap Groups	Click to swap groups.
Close	Click to exit the dialog box.

## Select Font

The Preferences dialog box contains the following fields and button

Name	Description
Font	Select the Font to change the Font of the tool.
Font Style	Select the Font style to change the appearance of the font.
Effects	Select the <i>Strikeout</i> and <i>Underline</i> check box to enable the effects.
Sample	Display the preview of the font changes.
Writing System	Select the <i>Language/Writing Style</i> to be used in the tool.
Ok	Click to save the changes.
Cancel	Click to cancel the operation.

## Settings

Use the Settings Dialog box to:

- Specify the Board dimensions, PCB Outline board values. You can also add, delete, or edit the libraries through File Paths.
- Specify the lrf paths and cpm file.
- Specify the Net Name Template for the connections in your design. You can specify the Net Name Template for both interface protocols and device protocols.
- Specify the Global Clocks for Spartan3/3A designs. After defining the Global Clocks you need to map with the Data and Address clock groups.
- Export the signals with NoStandard standards to UCF file.

### ***How to Access the Settings dialog box?***

- Choose *File – Settings*.

## Project tab

Use the Project tab to define board dimensions, pcb board outline values, and modify the libraries paths in design.

Name	Description
Project Directory	Displays the path where the project is saved. <b>Note:</b> This is a read-only field.
File Paths	
Output Files Path	Displays the path for output files. By default, all the output files are saved at the Output directory created at the location where the project is saved. You can change the path by clicking the Browse button.
Board Extents	
Width	Enter the width for the PC board canvas.
Height	Enter the height for the PC board canvas.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

Units	Click on the combo box and select the appropriate unit of measurement for the board dimensions such as Inches, Centimeters and so on.
Define PCB Outline	Select this option to specify the board outline width, height and top-left corner of the outline fo your own choice.  <b>Note:</b> While importing an allegro board file make sure that this option is unchecked otherwise the Allegro PCB Outline will be ignored and new outline will be drawn by FSP using the specified PCB Outline details.
Bottom Left X and Bottom Left Y	These two fields are optional. If you do not specify the values for these options, by default the outline is placed in the centre of the board extents.
Width	Specify the width of the board outline.
Height	Specify the height of the board outline.
Design Settings	
Rules Path	Allows you to add your own rules file in FSP Libraries.  Perform the following steps to add rules file path:  1. Click <i>Edit</i> . The Rules File Path Editor dialog box is displayed. 2. Click <i>Add</i> . 3. Browse for the rules file folder and click <i>OK</i> . 4. Click <i>OK</i> .
Perform consistency check at the time of opening project	Select this option to perform the consistency check between the rules file and mapping files.
Mark for ECO Mode	Select this option to set an ECO flag in the FSP design database. You use this option during the last phase of the design cycle. When you load the FSP design with ECO flag in Allegro PCB Editor, both Auto Pin Swap and Manual Pin Swap commands will make only those swaps that can be backannotated to the schematics from Allegro PCB Editor.
Project CPM File	Displays the current project cpm file.
FSP Database Location	Displays the path to the fsp project file (<project_name>.fsp).

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Generate Symbols in	This field lets you specify the library name in which you want to generate the symbols of FSP components.  By default, <code>fsp_fe_1.lib</code> library name is displayed. You can also choose an existing library names from the drop-down list or type a new one.
Advanced Settings	Invokes the Advanced Schematic Settings dialog box.

### Net Name Template tab

Use the Net Name Template tab to specify the net name template for the connections in your design.

Name	Description
Net Name Template	Select this check box to specify the pattern of net names.  If you have not selected the check box, FSP automatically generates the unique net names.  There is a possibility that after specifying the Net Name Template, the generated nets will conflict with each other. In such a case, you need to resolve them through the Change Signal Names dialog box.
Interface	Select this tab to specify the template pattern for the interface net names.
Protocol	Select this tab to specify the template pattern for the protocol net names.
	Click to add one more drop-down to change the net name with a combination of templates.  The drop-down options are End Instance Name, End Pin Name, End Pin Number, Protocol Signal.
	Click to remove the selected drop-down.  Name, Start Instance Name, Start Pin Name, Start Pin Number, and Connected Devices.
Default	Click to set the default net name template.
Replace Pattern	Select this check box to change the pattern of the net name string.  Unless you select this check box, you can not specify the replace string to change net names.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

	Click to add one more string pattern to change the net name template.
	Click to delete the selected string pattern.
Pattern	Enter the current string
Replace String	Enter the new string.

### Design Clocks tab

Use the Design Clock tab to define the global clocks for Spartan3/3A designs.

Name	Description
Add	Click to add a clock to the left side of the pane.
Delete	Click to delete the clock group.
OK	Click OK to complete the process.
Cancel	Click Cancel to leave the window.

### Xilinx tab

Use the Xilinx tab to export the NoStandard standards to UCF file.

Name	Description
Export UCF Settings	

### Altera

Use the Altera tab to enable or disable the Altera IO assignment analysis.

Name	Description
Run IO Assignment Analysis for the generated constraints	Select the option to run the IO assignment analysis for the Altera devices.

## NetGroup

Use the NetGroup tab to specify the netgroups related settings.

Name	Description
Automatically create netgroups for protocols	<p>Select this option to define NetGroups automatically in group wise for device protocol. While creating protocol manually using <i>Edit Protocol</i> editor, the NetGroups are assigned immediately after a signal is added to the group.</p> <p>Example,</p> <p>NetGroups for a device protocol with three groups data_group1, data_group2 and data_group3, are auto-assigned as following:</p> <p>&lt;instance_name1&gt;_&lt;instance_name2&gt;_data_group1 for first group &lt;instance_name1&gt;_&lt;instance_name2&gt;_data_group2 for second group &lt;instance_name1&gt;_&lt;instance_name2&gt;_data_group3 for third group.</p>
Automatically create netgroups for interfaces	<p>Select this option to automatically define NetGroups for interface components, only if no NetGroups exists in the design.</p> <p>NetGroups are automatically created at the time of placing the components on the canvas. These NetGroups are created based on the interface group wise.</p>

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Maximum size for auto-created net groups	<p>The <i>Maximum NetGroup Size</i> option, enables you divide and split the NetGroups into smaller NetGroups at the time of auto creation process. Typically, FSP cannot route a larger NetGroups and cannot route on single layer. To overcome this problem, FSP provides support to specify maximum NetGroup size. The maximum NetGroup size value is used to divide large NetGroups into smaller NetGroups.</p> <p>For example, assume that you have 64-bit size of net group U2.Data&lt;0&gt;.....&lt;63&gt; and you specify the netgroup size as 16 in this option. Now, when you use auto creating option to define net groups, the maximum size to split the netgroup into four sub smaller net group. See the list below:</p> <ul style="list-style-type: none"><li><input type="checkbox"/> U2.Data1&lt;0&gt;...&lt;15&gt;</li><li><input type="checkbox"/> U2.Data2&lt;16&gt;...&lt;30&gt;</li><li><input type="checkbox"/> U2.Data3&lt;31&gt;...&lt;45&gt;</li></ul> <p>U2.Data4&lt;46&gt;...&lt;63&gt;</p>
------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

### Read Me tab

Use the *Read Me* tab to add design information. The design information can include any text information such as parts information, board dimensions, and so on. This design information is saved in the project database so that you can see the information at any time when you invoke the *Settings* dialog box.

You can start entering the information in the white space and click *OK* to save the information.

## System ACE Dialog

The System ACE Dialog dialog box lets you connect the System ACE component with multiple FPGA's.

### **How to Access the System ACE Dialog dialog box**

- Right-click on the System ACE instance and click *Instance Properties*.  
The properties are displayed in the *Properties*.
- Click *browse(...)* of the System ACE Device Chain text box.

Name	Description
Available Device Instance(s)	Displays the list of FPGA instance names present on canvas. Select a device name with which you want to connect the System ACE component.
Connected Device Instance(s)	Displays the list of FPGA instance names, that you have decided to connect with System ACE component.
>>	Use this option if you wish to connect the FPGA with System ACE component. Click this option to move the device instance name to right pane.
<<	Use this option if you do not wish to connect the FPGA with System ACE component. Click this option to move the device instance name to left pane.
OK	Click this option to save the settings.
Cancel	Click this option to cancel the settings.

## Specify DE-HDL Symbols To Generate

The Specify DE-HDL Symbol To Generate dialog box allows you to select the library name and part name. You use this dialog box while working in Setup DE\_HDL Symbols to Use/Generate dialog box for following:

- Selecting the DE-HDL symbol as use part.
- Specifying the library location to generate the DE-HDL symbol.

### ***How to Access the Specify DE-HDL Symbol To Generate dialog box***

- Choose *Generate – Setup Symbol Data*.
- Click ..., in the Set up DE-HDL Symbols Data dialog box.

Name	Description
Library Name and Part Name	If the Generate Symbol option is checked in Setup DE-HDL Symbols to Use/Generate dialog box, then perform the following: <ul style="list-style-type: none"><li>■ Specify the library name, in which the DE-HDL symbol exists.</li><li>■ Specify the existing part name.</li></ul> If the Generate Symbol option is unchecked in Setup DE-HDL Symbols to Use/Generate dialog box, then perform the following: <ul style="list-style-type: none"><li>■ Specify the library name to generate the DE-HDL symbol in the selected library directory.</li><li>■ Specify the part name.</li></ul>
OK	Click OK to save the settings.

## Specify OrCAD Symbol To Generate

The Specify OrCAD Symbol To Generate dialog box allows you to select the library name and part name. You use this dialog box while working in Setup OrCAD Symbols Data dialog box for following:

- Selecting the OrCAD symbol as use part.
- Specifying the library location to generate the OrCAD symbol.

### ***How to Access the Specify OrCAD Symbol To Generate dialog box***

- Choose *Generate – Setup Symbol Data*.
- Click ..., in Setup OrCAD Symbols to Use/Generate dialog box.

Name	Description
Library Directory Location	If the Generate Symbol option is checked in Setup OrCAD Symbols to Use/Generate dialog box, then complete the following steps: <ul style="list-style-type: none"><li>■ Specify the library directory location where the OrCAD symbol exists Or click <i>Browse</i> to the OrCAD symbol location.</li><li>■ Specify the existing OrCAD symbol name.</li><li>■ Specify the package type.</li></ul>
Library File Name (*.olb)	If the Generate Symbol option is unchecked in Setup OrCAD Symbols to Use/Generate dialog box, then perform the following: <ul style="list-style-type: none"><li>■ Specify the library location, where you want to generate the OrCAD symbol Or to click <i>Browse</i> to the new location.</li><li>■ Specify the OrCAD symbol file name.</li><li>■ Specify the package.</li></ul>
OK	Click OK to save the settings.

## Setup OrCAD Symbols Data

The Setup OrCAD symbol to use/generate dialog box allows you to manage the symbol selection and PTF selection process. You can do the following:

- Setup OrCAD symbol data
- Select existing OrCAD symbol
- Use the existing footprint

### ***How to Access the Setup DE-HDL Symbol to Use/Generate dialog box***

- Choose *Generate – Setup Symbol Data*.

Name	Description
Instance Name	Displays the instance names present in the current design.
Part Name	Displays the part names present in the current design.
Symbol Path	Display the symbol name and symbol directory name.  The Symbol Path field is disabled for the instances which are mapped to the symbols while placing through Component Browser.  For FSP generated symbol, click <i>browse (...)</i> to change the symbol name and library location.
Customize Symbol	This option is enabled for generated symbols. Customize Symbol option lets you to do layout changes of the components. Such as changing pin directions or adding/editing custom attributes.  To customize a generated symbol, select a enabled row and click <i>Customize Symbol</i> .
OK	Click this button to save the settings.
Cancel	Click this button to cancel the operation.

## Setup DE-HDL Symbols Data

The Setup DE-HDL symbol to use/generate dialog box allows you to manage the symbol selection and PTF selection process. You can do the following:

- Setup DE-HDL symbol data
- Select existing DE-HDL symbol
- Select DE-HDL symbol with multiple views
- Use the existing footprint

### ***How to Access the Setup DE-HDL Symbol to Use/Generate dialog box***

- Choose *Generate – Setup Symbol Data*.

Name	Description
Instance Name	Displays the name of the components present in the current design.
Part Name	Displays the name of the parts present in the current design.
Symbol Path	Display the symbol name and symbol directory name.  The Symbol Path field is disabled for the instances which are mapped to the symbols while placing through Component Browser.  For FSP generated symbol, click <i>browse (...)</i> to change the symbol name and library location.
Customize Symbol	This option is enabled for generated symbols. Customize Symbol option lets you to do layout changes of the components. Such as changing pin directions or adding/editing custom attributes.  To customize a generated symbol, select a enabled row and click <i>Customize Symbol</i> .
OK	Click this button to save the settings.
Cancel	Click this button to cancel the operation.

## Update Design from Allegro Board

The Update Design from Allegro Board lets you to update the logical design (FSP) with layout changes made in PCB Editor. The following changes are allowed to synchronize logical design and layout design:

- Board Outline
- Component Placement
- Reference Designators

The board changes should be imported in the original logical design. Which means you must open the logical design before importing the board file.

### **How to Access**

- Choose *File – Update Design from Board*.

Name	Description
Board File	Specify the board file name and path to the directory where the board file exists or click <i>browse (...)</i> to browse to the file.
Board Outline	Select this entity to update the Allegro board outline in current FSP design.
Component Placement	Select this entity to update the Allegro board component placements in current FSP design.
Reference Designators	Select this entity to update the Allegro board components reference designator in current FSP design.
Hide Log	Click this to see the log window.
Update	Click to update the selected entities in current FSP design.

## Update Pin Locations from PCB Footprint

The Update Pin Locations From PCB Footprint dialog box allows you to import X and Y location values from footprint (.dra) file. After importing the file, the symbolic view of the component is updated as per the X and Y locations values.

**Note:** This option is available only for the components (real component) which are placed through wizard flow.

### ***How to Access the Update Pin Location dialog box***

- Right-click on the instance and select *Update Instance Footprint* option from pop-up menu.

Name	Description
Select DRA File	Enter a file name or select a name from the list.
OK	Click to import the file.
Cancel	Click to cancel the operation.

## Workflow

FSP provides a step-by-step and easy-to-use work flow that help you in performing tasks such as setting up a project, defining connections, and verifying and checking design. You can access these tasks from the menu bar or from the *Workflow* window. For example, to add power regulators in the design, you can choose *Tools – Power Connections* or click the *Add Power Regulators* option in the *Other Connections* section in the *Work Flow* window.

### Customizing the Work Flow Structure

You can customize the work flow by using any one of the following methods:

- Dragging and Droping
- Editing the default\_workflow.xml file

#### ***Dragging and Droping***

In the *Work Flow* window, you can seamlessly move an option from one section of to another section.

To move an option:

1. Click on an option to move.
2. Drag the option to the section where you want to place.
3. Drop the option.
4. After moving the option, you can save the changes by choosing *Manage – Save*.

This ensures that the work flow structure does not change, when you open the current project the next time.

#### ***Editing the default\_workflow.xml file***

You can also customize the *Workflow* by editing the default\_workflow.xml file.

At the time of installation, the default\_workflow.xml file is copied to the ..<release name>/share/cdssetup/fsp. It is not recommended to modify the file at this directory. Instead, you can copy the file to \$CDS\_SITE/cdssetup/fsp. Any file at the CDS\_SITE location ensures that when FSP invoked, FSP reads the files at the CDS\_SITE location.

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

You can also copy the file to a new location and create an environment variable \$HOME\cdssetup\fsp for individual designs. The \$HOME variable lets you locate the file outside the standard default location such as CDSROOT and CDS\_SITE.

FSP reads the default\_workflow.xml file in the following order when invoked:

- \$HOME\cdssetup\fsp
- \$CDS\_SITE\cdssetup\fsp
- \$CDSROOT\share\cdssetup\fsp



You should not modify the name of the default\_workflow.xml file at any locations.

To edit the file:

1. Open the xml file in any text editor.
2. Modify the content as required.
3. Save and close the text editor.
4. Choose *File – Open*.

The *Open Project* dialog box is displayed.

5. Browse to the file, select the file, and click *Open*.

After you click *Open*, the *Workflow* displays the modified options.

**Note:** Before you edit the xml file, you need to save and close the project.

#### ***How to Access the Work Flow window***

- Choose *Window – Work Flow*.

Name	Description

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

Manage	<p>The following options are displayed when you click this option.</p> <ul style="list-style-type: none"><li>■ Load Use this option to load an existing default_workflow.xml file. After you click this option, the <i>Open Workflow Config File</i> window is displayed. Browse to the file, select the file, and click <i>Open</i>.</li><li>■ Save When you make any changes in the work flow structure, use this option to save the changes.</li><li>■ Save As Use this option to save the changes in a different location. After you click this option, the <i>Save Workflow Config File</i> window is displayed. Browse to the directory where you want to save the file, enter a file name, and click <i>Save</i>.</li><li>■ Set Workflow Chart This option lets you set a flow chart for the work flow for your reference. You can set a workflow chart by using any one of the formats .jpg, .jpeg, .png, and .gif. After you click this option, the <i>Select a Workflow Image File</i> window is displayed. Browse to the file, select the file, and click <i>Open</i>. After you set the workflow chart, the <i>Workflow Chart</i> option gets enabled. Click to view the flow chart.</li><li>■ Reset Workflow Chart Use this option to reset the specified or existing workflow chart. After you click this option, the <i>Workflow Chart</i> option gets disabled</li></ul>
Workflow Chart	<p>Use this option to see the flow chart. This option is enabled when you set a flow chart and is disabled when you reset.</p>
	Click to expand the list.

## **Allegro FPGA System Planner User Guide**

### Dialog Box Descriptions

---

Click to collapse the list.



## What do you want to do....?

The *What do you want to do..?* dialog box allows you to do the following:

- Creating new project
- Opening existing project
- Opening recent project
- Selecting schematic environment
- Opening sample design
- Working with library tools

### ***How to Access the What do you want to do..? dialog box***

- The What do you want to do..? dialog box appears every time you invoke FSP.

Name	Description
<i>Open Recent</i>	Select this radio button to open the recently-opened project from the drop-down list. The drop-down list displays the recently-opened projects.
<i>Open Existing</i>	Select this radio button to open an existing project. After selecting this radio button, browse to the directory where the project is located.
<i>Create New</i>	Select to create a new project. After selecting, the following fields are enabled.
<i>Schematic Environment</i>	Select any one of the environments that you want to work. If you right-click on the device options, some of the Generate menu will be disabled. For more information on the schematic environment, See <a href="#">Selecting Schematic Environment</a> for more information.
The following fields appear when you select DE-HDL as schematic environment.	
<i>Design Name</i>	Enter the name for the new project.
<i>Select Project Directory</i>	Type the complete path of the folder in which you want to create the new project.  Or  Click <i>browse(...)</i> , select a folder in <i>Choose Folder Path</i> dialog box, and then click <i>OK</i> .

## Allegro FPGA System Planner User Guide

### Dialog Box Descriptions

---

<i>FSP Database Location</i>	<p>This is a read-only field.</p> <p>Displays the relative path of the folder followed by &lt;design_name&gt;.fsp. &lt;design_name&gt; is the name you enter in the <i>Design Name</i> field.</p> <p><b>Note:</b> When you type/modify a name in the <i>Design Name</i> field, the changes immediately appear in this field.</p>
<i>Generate Symbols in</i>	<p>This field lets you specify the library name in which you want to generate the symbols of FSP components.</p> <p>By default, <code>fsp_fe.lib</code> library name is displayed. You can also choose an existing library names from the drop-down list or type a new one.</p>
<i>Design Library</i>	<p>This field represents working directory name. All the design data, FSP project file, and view-related log files are stored in this directory.</p> <p>By default, &lt;design_name&gt;_lib name appears in this field. You can also choose an existing names from the drop-down list or type a new one.</p>
The following fields appears when you select OrCAD as schematic environment.	
<i>Design Name</i>	Enter the name for the new project.
<i>Directory Path</i>	<p>Type the complete path of the folder in which you want to create the new project.</p> <p>Or</p> <p>Click <i>browse(...)</i>, select a folder in <i>Choose Folder Path</i> dialog box, and then click <i>OK</i>.</p>
<i>Configuration File</i>	<p>This field lets you specify the capture.ini file. This file will be used for schematic generation.</p> <p>By default, &lt;release_name&gt;/tools/fsp/templates/orcad/capture.ini olb appears. You can also specify your own capture.ini.</p>
<i>Open Sample Design</i>	<p>Select this option to open the sample design from the example directory.</p> <p>Click on the drop-down button, and choose any one of the examples.</p>
<i>Work with Library Tools</i>	Select this radio button to create new parts using the Library menu.
<i>OK</i>	Click to proceed in FSP.

## **Allegro FPGA System Planner User Guide**

### Dialog Box Descriptions

---

Cancel

Click to cancel the operation.

## **Allegro FPGA System Planner User Guide**

### Dialog Box Descriptions

---