

Analog Resource Constraint

1.50

Features



- Limits analog routing of a signal to a specific routing resource
- All terminals on the signal must connect directly to the routing resource

Note Routing is strict. All of the devices connected to the net with the resource constraint must have a direct hardware connection to the resource. Refer to the Analog Routing Diagram in the applicable Technical Reference Manual (TRM), which is available from the Cypress website, www.cypress.com. If the resources do not have a hardware connection to the specified constraint, an error will occur.

General Description

The Analog Resource Constraint component allows you to define the route of the analog signal to which it is connected. This is an advanced feature that is not needed for most designs, and should be used with caution.

Analog Resource Constraint Deprecated

The Analog Resource Constraint component is not shown in the Component Catalog by default, and it should no longer be used. When constraining a net, use the Net Constraint component. When, constraining a switchable connection of an Analog Mux, use the Mux Constraint component.

Input/Output Connections

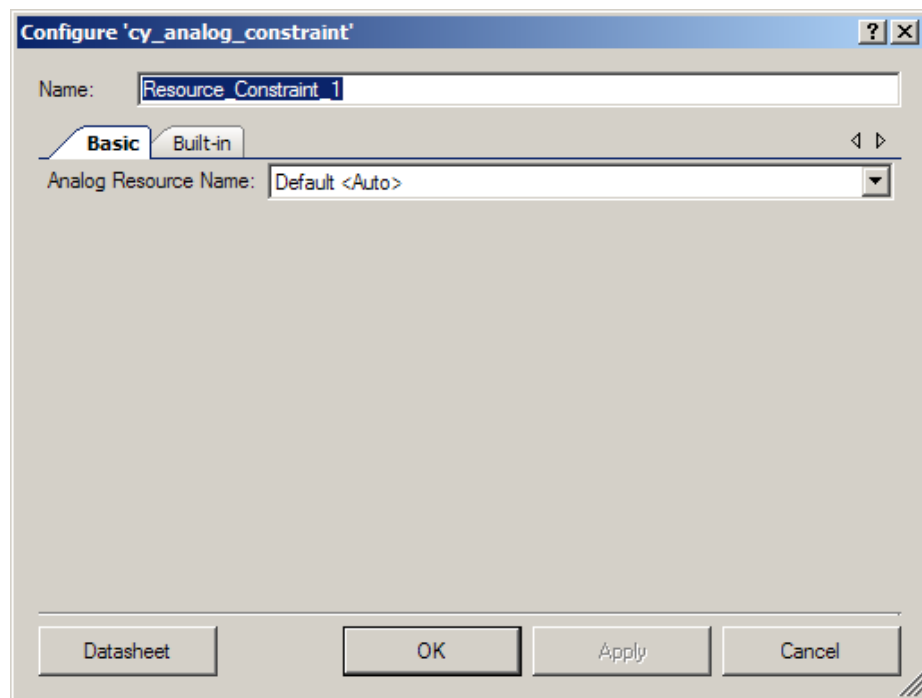
This section describes the various input and output connections for the Analog Resource Constraint.

connect – Input /Output

Provides the connection to the analog signal to which the Analog Resource Constraint applies.

Component Parameters

Drag an Analog Resource Constraint onto your design and double-click it to open the **Configure** dialog.



The Analog Resource Constraint provides the following parameters.

Analog Resource Name

The analog resource to assign to the connected signal. The default value, **Auto**, has no effect. The routing resource list depends on the selected family.

Placement

The Analog Resource Constraint consumes hardware resources, because it specifies which hardware resource must be used by the router. It has no other placement specification.

Resources

The Analog Resource Constraint component causes the connected analog signal to consume the selected analog routing resource.

Functional Description

The following analog routing resource names are available for PSoC 3 and PSoC 5. Not every analog routing resource connects to every component terminal. For detailed information about analog connectivity, refer to the applicable device datasheet and TRM. These documents are available on the Cypress website, www.cypress.com.

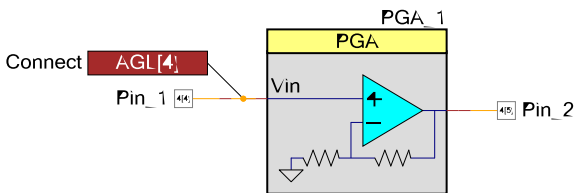
- Analog globals: AGL[0] – AGL[7], AGR[0] – AGR[7]
- Analog local bus: abusl0 – abusl3, abusr0 – abusr3
- Analog mux bus: AMUXBUSL, AMUXBUSR
- Combined left/right resources: AG[0] – AG[7], abus0 – abus3, AMUXBUS

When an Analog Resource Constraint is present on a signal, the signal will be routed using **only** the specified resource. All of the component terminals connected to the signal must have a direct connection to the routing resource. The analog placer might not be able to automatically ensure that components are placed in way that satisfies Analog Resource Constraints. Components connected to constrained signals should be placed manually.

Analog Resource Constraint Example

In this example, the Analog Resource Constraint is used to force a signal (Pin_1) on P4[4] to be routed on Analog Global Left 4.

Figure 1. Analog Resource Constraint Example



Design-Wide Resources Settings

The following figures show settings in the Design-Wide Resources editors.

Figure 2. Directives Editor

Component (Signal) Name	Directive Type	Directive Value
\PGA_1:SC\	ForceComponentFixed	F(SC,0)



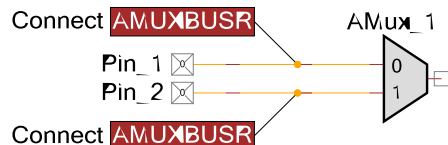
Figure 3. Pin Editor

Alias	Name	Pin	Lock
	Pin_1	P4[4]	<input checked="" type="checkbox"/>
	Pin_2	P4[5]	<input checked="" type="checkbox"/>

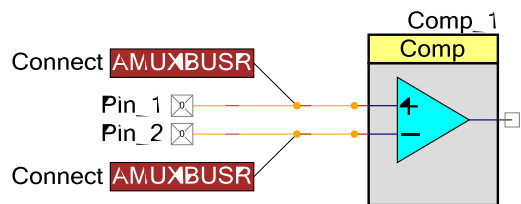
Analog Resource Constraint Example with AMux

When the Analog Resource Constraint is connected to an input of an analog mux, it controls the routing of the mux rather than the routing of the input signal.

Figure 4 shows that the constraint applies to the routing of the AMux. Pin_1 is connected to Pin_2 when the firmware connects both inputs of the AMux.

Figure 4. Analog Resource Constrained with AMux**Short Circuit Example**

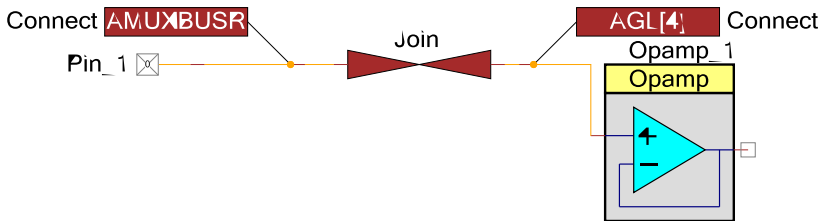
If the same analog routing resource is specified for multiple signals that are not connected to an AMux, the signals will be connected to each other, which is not always the intended behavior.

Figure 5. Short Circuit Example

Analog Net Join Example

To apply multiple constraints to a signal or constrain a subset of the connections in a signal, use a Net Join component. Refer to the Net Join component datasheet for details (available from the PSoC Creator Component Catalog).

Figure 6. Net Join Example



Component Changes

This section lists the major changes in the component from the previous version.

Version	Description of Changes
1.50.c	Updated to direct users to either Net Constraint or Mux Constraint
1.50.b	Cosmetic change removing wire guide from terminal
1.50.a	Minor datasheet edits and updates

© Cypress Semiconductor Corporation, 2010-2012. The information contained herein is subject to change without notice. Cypress Semiconductor Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in a Cypress product. Nor does it convey or imply any license under patent or other rights. Cypress products are not warranted nor intended to be used for medical, life support, life saving, critical control or safety applications, unless pursuant to an express written agreement with Cypress. Furthermore, Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress products in life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

PSoC® is a registered trademark, and PSoC Creator™ and Programmable System-on-Chip™ are trademarks of Cypress Semiconductor Corp. All other trademarks or registered trademarks referenced herein are property of the respective corporations.

Any Source Code (software and/or firmware) is owned by Cypress Semiconductor Corporation (Cypress) and is protected by and subject to worldwide patent protection (United States and foreign), United States copyright laws and international treaty provisions. Cypress hereby grants to licensee a personal, non-exclusive, non-transferable license to copy, use, modify, create derivative works of, and compile the Cypress Source Code and derivative works for the sole purpose of creating custom software and or firmware in support of licensee product to be used only in conjunction with a Cypress integrated circuit as specified in the applicable agreement. Any reproduction, modification, translation, compilation, or representation of this Source Code except as specified above is prohibited without the express written permission of Cypress.

Disclaimer: CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Cypress reserves the right to make changes without further notice to the materials described herein. Cypress does not assume any liability arising out of the application or use of any product or circuit described herein. Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress' product in a life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

Use may be limited by and subject to the applicable Cypress software license agreement.

