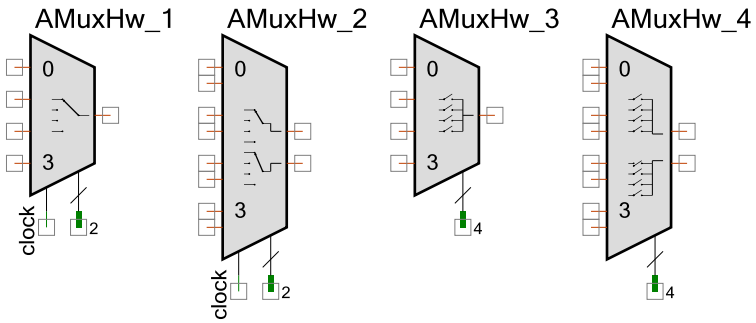
Analog Hardware Multiplexer (AMuxHw)

1.50

Features



- Single-ended or differential inputs
- Mux or switch mode
- From 1 to 256 inputs
- Hardware controlled
- Bidirectional (passive)

General Description

The Analog Hardware Multiplexer (AMuxHw) Component is used to provide hardware-switchable connections from GPIOs to analog resource blocks (ARBs).

When to Use an AMuxHw

Use an AMuxHw Component when you need a signal to be switched in hardware. Currently, only the GPIOs can be multiplexed with this multiplexer.

Because the AMuxHw Component is bidirectional, it can also be used as demultiplexer.

Input/Output Connections

This section describes the various input and output connections for the AMuxHw. An asterisk (*) in the list of I/Os indicates that the I/O may be hidden on the symbol under the conditions listed in the description of that I/O.

0 - 255 – Analog

The AMuxHw Component is can have between 1 and 256 analog “inputs.” Paired inputs are present when the **MuxType** parameter is set to **Differential**. These inputs must be GPIO pins.

enable – Input *

This input is optional. It is shown when the **Mode** parameter is set to **Mux** and the **ShowEnable** parameter is set to **true**. When this signal is low, all of the inputs are disconnected.

clock – Input *

This input is shown when the **Mode** parameter is set to **Mux**. If displayed, the clock input must be connected. It is used to provide synchronization during the decoding process of the selected signal. In **Mux** mode, you need two clock cycles to select the new input. The first clock disables the current selection (“break”), and the second clock enables the new connection (“make”).

If the **Mode** parameter is set to **Switch**, this input is hidden.

select – Input

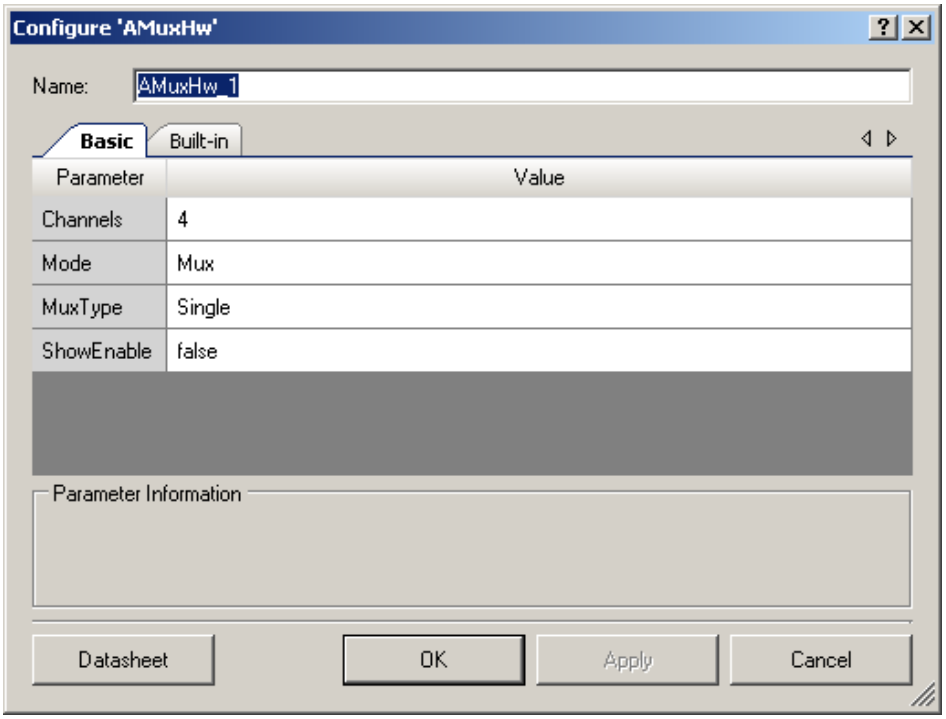
This input selects which analog input should be connected to the common connection. The width of the select input depends on the number of inputs and the **Mode** parameter. When the **Mode** parameter is set to **Mux**, this input is binary encoded. When the **Mode** parameter is set to **Switch**, there is one input connection per analog switch.

“common” – Analog

The “common” signal is the common connection; it is not labeled. Whichever input (0-255) signal is selected, it will be connected to this terminal. Paired “common” terminals are present when the **MuxType** parameter is set to **Differential**.

Component Parameters

Drag an AMuxHw Component onto your design and double-click it to open the **Configure** dialog.



The AMuxHw provides the following parameters.

Channels

This parameter selects the number of inputs or paired inputs depending on the **MuxType**. You may choose any value between 1 and 256. The default is **4**.

Mode

This parameter selects the mode of the AMuxHw Component, as follows:

- **Mux** (default) – Only one channel is connected at a time. Channel switching is synchronized with the clock. Select lines are binary encoded. Use Mux mode when the multiplexing must be “break before make,” making mux switching less prone to signal leakage.
- **Switch** – Multiple channels can be connected to the common connection simultaneously. Channel switching is unsynchronized. Select lines are one per channel. Use Switch mode when an arbitrary set of channels must be connected at the same time to the common connection. The select line of the particular channel is made high or low to connect or disconnect that channel from common, regardless of the other channels connected.

MuxType

This parameter selects between a single input per connection (**Single**) and a dual-input **Differential** input mux. **Single** is used when the input signals are all referenced to the same signal. In cases where two or more signals may have a different signal reference, select **Differential**.

ShowEnable

This parameter shows or hides the “enable” terminal on the Component. By default, this parameter is set to **false**, so the “enable” terminal is hidden.

Clock Selection

See clock – Input *.

Resources

The AMuxHw uses the individual switch at a pin.

When in Mux mode, AMuxHw also consumes macrocells.

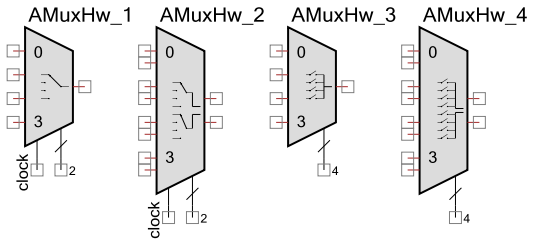
Total macrocells = number of inputs + number of select inputs + 1 (if number of inputs <= 16) or 2 (if number of inputs > 16).

Functional Description

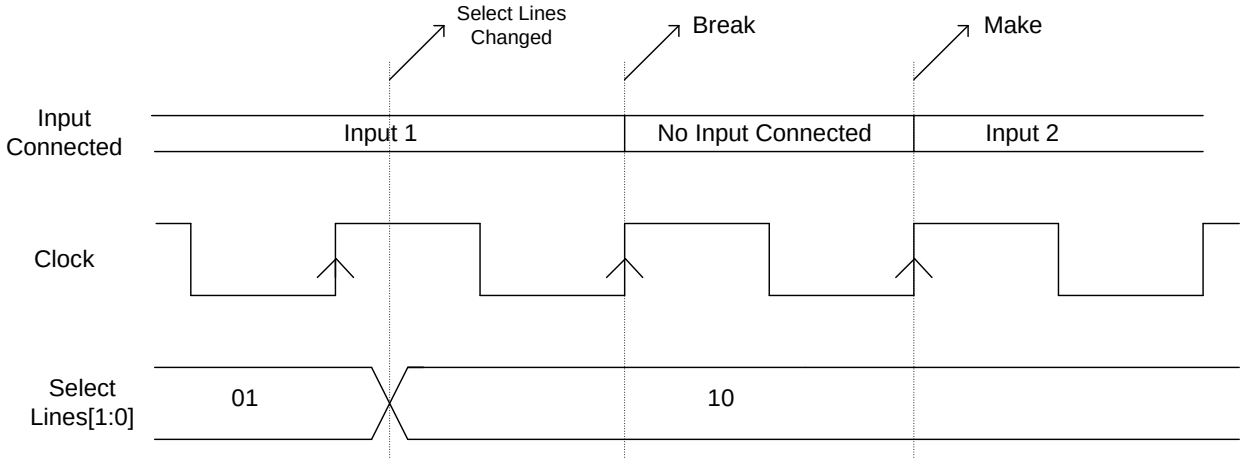
The AMuxHw is controlled by hardware. You can only connect one signal at a time to the common signal when it is set to Mux mode; you can connect multiple signals to the common signal when it is set to Switch mode.

In Mux mode, the AMuxHw Component is synchronized with the clock using the “break before make” feature. On each rising edge of the clock cycle, it looks for a change on the select lines. If there is any change, it disconnects the present channel (break). On the next rising edge of the clock, the new channel is connected (make).

The following shows the representation for an AMuxHw configured as Mux-Single, Mux-Differential, Switch-Single, and Switch-Differential, respectively.



Timing Diagram



Parasitics

The AMuxHW Component uses many routing sources on the chip. Due to the parasitics of these routing resources, the AmuxHW Component may appear to have current leak through an input to the output when the input is initially selected. Therefore, use strong voltage drive on AMuxHW Component inputs to achieve more accurate voltage reading on the output. If this is not possible, then reduce the switching speed to account for the increased parasitic effects.

PSoC Analog Coprocessor

For PSoC Analog Coprocessor devices, the AmuxHW component is only usable in “switch” mode and can only be hardware controlled using select TCPWM and CTB (Comparator) outputs.

DC and AC Electrical Characteristics

The AMuxHw operates at all valid supply voltages.

The AMuxHw clock supports up to the maximum operating frequency of the devices. You will need to validate the timing requirements with STA results for large configurations.

Component Changes

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

Version	Description of Changes	Reason for Changes / Impact
1.50.i	Updated Resources equation	
	Added note regarding possible parasitic effects	
1.50.h	Expanded the maximum number of channels to 256	Enable valid designs in select devices
1.50.g	The Component was made visible for PSoC 6.	
1.50.f	Minor datasheet edit.	
1.50.e	Added microcell usage for Mux mode. Added information about maximum clock frequency.	Datasheet defect fixed.
1.50.d	Minor datasheet edits and updates	
1.50.c	Minor datasheet edits and updates	
1.50.b	Minor datasheet edits and updates	
1.50.a	Minor datasheet edits and updates	

© Cypress Semiconductor Corporation, 2010-2019. This document is the property of Cypress Semiconductor Corporation and its subsidiaries, including Spansion LLC (“Cypress”). This document, including any software or firmware included or referenced in this document (“Software”), is owned by Cypress under the intellectual property laws and treaties of the United States and other countries worldwide. Cypress reserves all rights under such laws and treaties and does not, except as specifically stated in this paragraph, grant any license under its patents, copyrights, trademarks, or other intellectual property rights. If the Software is not accompanied by a license agreement and you do not otherwise have a written agreement with Cypress governing the use of the Software, then Cypress hereby grants you a personal, non-exclusive, nontransferable license (without the right to sublicense) (1) under its copyright rights in the Software (a) for Software provided in source code form, to modify and reproduce the Software solely for use with Cypress hardware products, only internally within your organization, and (b) to distribute the Software in binary code form externally to end users (either directly or indirectly through resellers and distributors), solely for use on Cypress hardware product units, and (2) under those claims of Cypress’s patents that are infringed by the Software (as provided by Cypress, unmodified) to make, use, distribute, and import the Software solely for use with Cypress hardware products. Any other use, reproduction, modification, translation, or compilation of the Software is prohibited.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS DOCUMENT OR ANY SOFTWARE OR ACCOMPANYING HARDWARE, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. To the extent permitted by applicable law, Cypress reserves the right to make changes to this document without further notice. Cypress does not assume any liability arising out of the application or use of any product or circuit described in this document. Any information provided in this document, including any sample design information or programming code, is provided only for reference purposes. It is the responsibility of the user of this document to properly design, program, and test the functionality and safety of any application made of this information and any resulting product. Cypress products are not designed, intended, or authorized for use as critical Components in systems designed or intended for the operation of weapons, weapons systems, nuclear installations, life-support devices or systems, other medical devices or systems (including resuscitation equipment and surgical implants), pollution control or hazardous substances management, or other uses where the failure of the device or system could cause personal injury, death, or property damage (“Unintended Uses”). A critical Component is any Component of a device or system whose failure to perform can be reasonably expected to cause the failure of the device or system, or to affect its safety or effectiveness. Cypress is not liable, in whole or in part, and you shall and hereby do release Cypress from any claim, damage, or other liability arising from or related to all Unintended Uses of Cypress products. You shall indemnify and hold Cypress harmless from and against all claims, costs, damages, and other liabilities, including claims for personal injury or death, arising from or related to any Unintended Uses of Cypress products.

Cypress, the Cypress logo, Spansion, the Spansion logo, and combinations thereof, WICED, PSoC, CapSense, EZ-USB, F-RAM, and Traveo are trademarks or registered trademarks of Cypress in the United States and other countries. For a more complete list of Cypress trademarks, visit cypress.com. Other names and brands may be claimed as property of their respective owners.