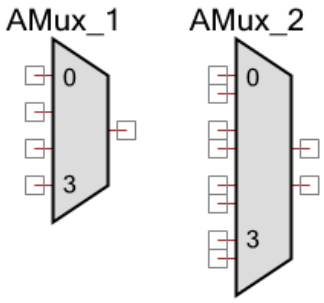
模拟复用器（AMux）

1.80

特性



- 单端或差分连接
- 连接数 1 到 64 之间可调节（单端 AMux），连接数 1 到 32 之间可调节（差分 AMux）。
- 软件控制
- 连接可以采用引脚或内部源
- 多个同时连接
- 双向（被动）

概述

模拟复用器（AMux）组件可用于将一个或多个模拟信号连接到另一个通用模拟信号，也可以不连接模拟信号。同时连接多个模拟信号的能力提供了交叉交换支持，这是对传统复用器功能的扩展。

何时使用 AMux

任何时候需要将多个模拟信号复用到一个源或目标时都可使用 AMux。由于 AMux 是被动的，它可用于复用输入或输出信号。

输入/输出接口

本节介绍 AMux 的各种输入和输出接口。I/O 列表中的星号（*）表示：在 I/O 说明部分中所列出的情况下，该 I/O 可能不可见。

0-63 — 模拟

AMux 能够具有 1 到 64 种模拟可交换连接。

0-32（成对）— 模拟*

只有当 **MuxType** 参数设置为 **Differential**（差分）时，才能使用成对可交换连接。

共模信号 — 模拟

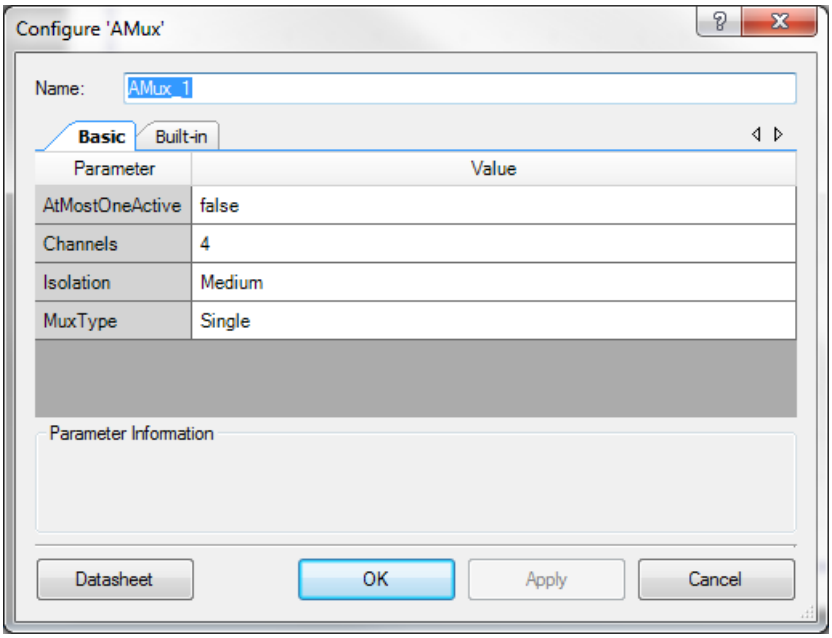
共模信号采用通用连接；它没有标签。使用 AMux_Select()函数选择的通道将连接到此终端。

共模信号（成对）— 模拟*

当使用差分复用器时，“共模（成对）”信号采用共模成对连接。使用 AMux_Select()函数选择的通道将连接到此终端。

组件参数

将 AMux 组件拖动到设计中，双击它以打开 **Configure**（配置）对话框。



AMux 提供下列参数。

通道

此参数根据 **MuxType** 选择可交换连接数。1 到 64 之间的任何数值都有效（单端 AMux），1 到 32 之间任何值均有效（差分 AMux）。

MuxType

此参数用于在 **Single**（单一）可交换连接复用器和 **Differential** 差分可交换连接复用器之间进行选择。当可连接信号都相对于同一参考信号（如 V_{SSA}）时，将使用 **Single**（单一）。当两个或多个信号与不同的参考源相对应时，要选择 **Differential**（差分）选项。“差分”模式最常用于提供差分输入的模数转换器（ADC）。

AtMostOneActive

当设为真时，该参数从 AMux 中移除交叉开关支持。这限制 AMux 到最多一个通用连接。设置这个值为真时会从已生成的代码中移除“连接”API。可优化 AMux 的性能。

隔离

通过此参数可以从下列隔离模式中选择某一项：

- **Minimum（最低） —** 采用单个外开关。这样可保证开关时间最短。
- **Medium（中等（默认）） —** 试图采用双开关，仅配有外开关和专用的内开关。如果专用的内开关均不可用，则采用单个外开关。双开关不仅会增强隔离，还会延长开关时间。
- **Maximum（最大） —** 采用双开关，配有外开关和可能共用的内开关。内开关不一定是专用的。参考计数允许共用一个内开关。当不使用专用内开关时，切换时间会进一步受到影响。

以下各图说明在无臂连接、底臂连接以及双臂连接时，Amux 适用的三种开关备选方案。

图 1. 单开关（无内开关）

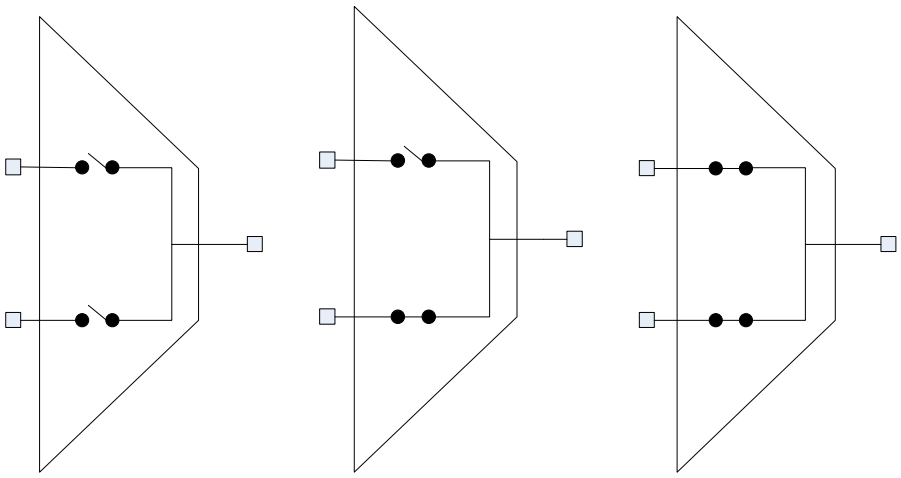


图 2. 双开关（专有内开关）

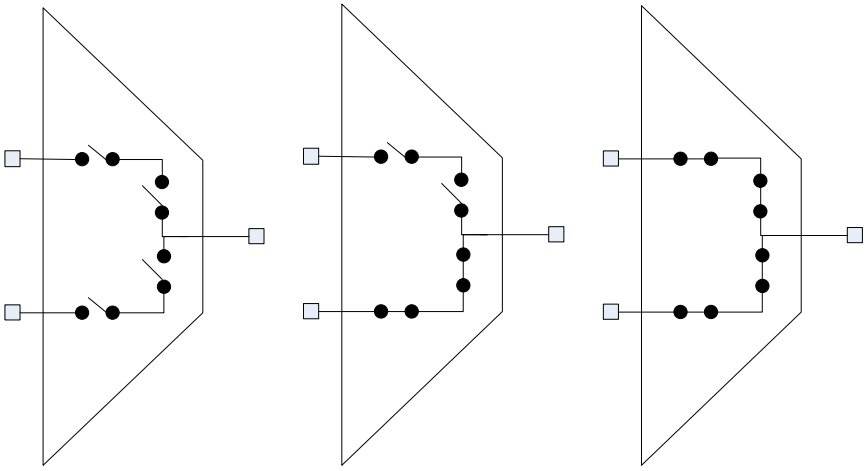
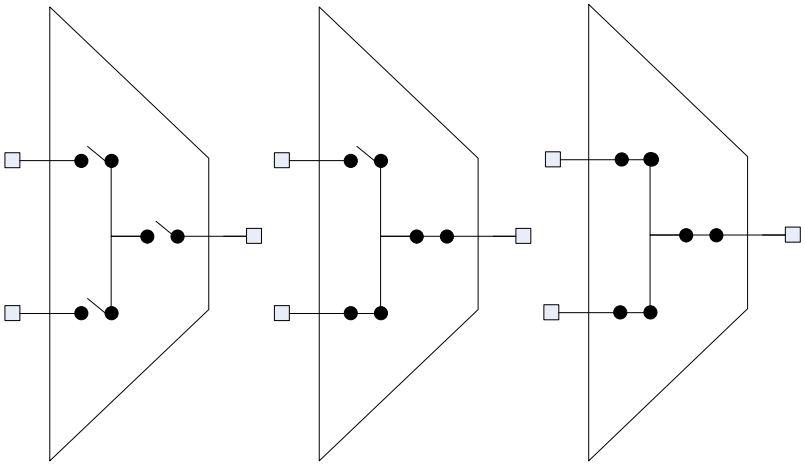


图 3. 动态开关（共用内开关）



应用编程接口

通过应用编程接口（API），您可以使用软件对组件进行配置。下表格列出并说明了每个函数的接口。以下各节将更详细地介绍每个函数。

默认情况下，PSoC Creator 将实例名称“AMux_1”分配给所提供设计中的第一个组件实例。可以在遵循标识符语法规则下，将实例重命名。实例名称会成为每个全局函数名称、变量和符号常量的前缀。为增加可读性，下表中使用了实例名称“AMux”。

函数	说明
AMux_Init()	断开所有通道的连接
AMux_Start()	断开所有通道的连接
AMux_Stop()	断开所有通道的连接
AMux_Select()	断开所有通道的连接，然后连接“chan”。当**AtMostOneActive**为真时，实现AMux_FastSelect()。
AMux_Connect()	连接“chan”信号，但是不断开其他通道的连接。当**AtMostOneActive**为真时，此函数不可用。
AMux_Disconnect()	仅断开“chan”信号的连接
AMux_FastSelect()	断开AMux_Select()或AMux_FastSelect()函数选择的上一通道，然后连接新信号“chan”。
AMux_DisconnectAll()	断开所有通道的连接

void AMux_Init(void)

说明： 断开所有通道的连接。

参数： 无

返回值： 无

其他影响： 所有寄存器都将被复位为初始值。

void AMux_Start(void)

说明： 断开所有通道的连接。

参数： 无

返回值： 无

其他影响： 无

void AMux_Stop(void)

说明： 断开所有通道的连接。

参数： 无

返回值： 无

其他影响： 无

void AMux_Select(uint8 chan)

说明： AMux_Select()函数首先断开所有其他通道的连接，然后连接给定的通道。当**AtMostOneActive**为真时，实现AMux_FastSelect()。

参数： chan：要连接到通用终端的通道。

返回值： 无

其他影响： 使用AMux_Select()，将断开通过AMux_Connect()或AMux_FastSelect()实现的连接。

void AMux_FastSelect(uint8 chan)

说明： 此函数首先断开AMux_FastSelect()或AMux_Select()函数进行的上一连接，然后连接给定的通道。AMux_FastSelect()函数类似于AMux_Select()函数，但速度快一些，这是因为断开的仅是上一选择的通道而不是选择所有可能的通道。

参数： chan：要连接到通用终端的通道

返回值： 无

其他影响： 如果在调用AMux_FastSelect()之前使用AMux_Connect()函数选择了通道，将不断开AMux_Connect()选择的通道。这在必须连接并行信号时非常有用。

void AMux_Connect(uint8 chan)

说明： 此函数将给定通道连接到通用信号，不影响其他连接。当**AtMostOneActive**为真时，此函数不可用。

参数： chan：要连接到通用终端的通道

返回值： 无

其他影响： 调用函数AMux_Select()后，会断开在将通过的通道连接到AMux_Select()命令之前使用AMux_Connect()函数连接的任何通道。

void AMux_Disconnect(uint8 chan)

说明： 仅断开指定通道与通用终端的连接。

参数： uint8 chan：要与通用终端断开连接的通道

返回值： 无

其他影响： 无

void AMux_DisconnectAll(void)

说明： 断开所有通道的连接。

参数： 无

返回值： 无

其他影响： 无

示例固件源代码

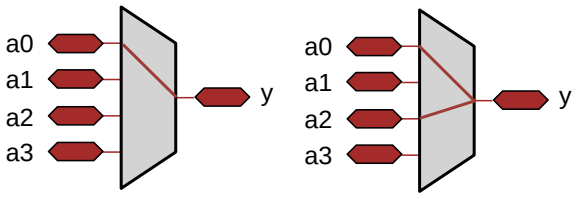
PSoC Creator 在“Find Example Project”（查找示例项目）对话框中提供了很多包括原理图和代码示例的示例项目。要查看特定组件实例，请打开“Component Catalog”中的对话框或者原理图中的组件样例。要查看通用示例，请打开 Start Page 或 **File** 菜单中的对话框。根据要求，可以通过使用对话框中的 **Filter Options** 选项来限定可选的项目列表。

更多有关信息，请参考《PSoC Creator 帮助》部分中主题为“查找示例项目”中的内容。

功能说明

AMux 与大多数硬件复用器不一样。下面是 AMux 与标准固定硬件复用器的两个不同之处。第一，它是独立开关的集合；第二，它由固件而不是硬件控制。

由于这两点不同，AMux 更加灵活，允许多个信号同时连接到通用信号。当 **AtMostOneActive** 参数设为假时，在任意给定时间两个或两个以上信号可连接至通用信号。



性能

模拟复用器由软件控制，因此切换性能取决于提供的 API 的执行时间。此性能取决于设计中复用器的精确配置。表 1 为了提供有关切换性能的指南。

所有的性能测量都是在 CPU 频率为 48 MHz 的情况下测得的。性能指标与 CPU 频率接近线性关系。编译器优化被配置为 PSoC Creator 捆绑的编译器所能提供的最高级别。对于 PSoC 3，编译器设置是针对大小处于优化级别为 5 的 Keil 优化。对于 PSoC 4 和 PSoC 5LP，GNU 编译器设置是针对大小的优化。

表 1. 性能

函数	复用器单输入	PSoC 3 （µs）	PSoC 4 （µs）	PSoC 5LP （µs）
Connect*	2	4.1	1.3	1.9
	4	3.9	1.3	1.9
Disconnect	2	3.9	1.4	1.9
	4	3.8	1.4	1.8
Select*	2	14.6	3.5	5.5
	4	22.6	5.8	8.1
FastSelect	2	9.3	2.6	4.0
	4	9.3	2.6	4.0

*当“AtMostOneActive”设为真时，Connect 函数不可用，而 Select 函数的性能与 FastSelect 相同。

资源

AMux 使用单独开关将模块和引脚连接到模拟总线。

API 存储器的使用情况

根据不同的编译器、器件、所使用的 API 数量以及组件的配置情况，组件所用的存储空间大小也不一样。下表提供了在某一器件配置中的所有 API 使用的存储器大小。

通过使用“释放”模式下相应的编译器，可以完成测量操作。在该模式下，存储器的大小得到了优化。有关特定的设计，可分析编译器所生成的映射文件以确定内存的使用情况。

配置	PSoC 3 （Keil_PK51）		PSoC 4 （GCC）		PSoC 5LP （GCC）	
	闪存 字节	SRAM 字节	闪存 字节	SRAM 字节	闪存 字节	SRAM 字节
单端	91	1	88	1	88	1
差分	141	1	136	1	140	1

MISRA 合规性

本节介绍了 MISRA-C:2004 合规性和本组件的偏差情况。定义了两种类型的偏差：

- 项目偏差 — 适用于所有 PSoC Creator 组件的偏差
- 特定偏差 — 仅适用于该组件的偏差

本节介绍了有关组件特定偏差的信息。《系统参考指南》的“MISRA 合规性”章节中介绍了项目偏差以及有关 MISRA 合规性验证环境的信息。

MISRA-C:2004 规则	规则类别（必须/建议）	规则说明	目标器件	违规的辩解
19.7	A	使用时，函数应优先于类函数宏。	所有器件	由于使用了函数宏以实现更高效的代码，所以出现了偏差。

AMux 组件没有任何嵌入式组件。

直流和交流的电气特性

AMux 将在所有有效供电电压下运行。

组件更改

本节列出了各版本中主要组件的更改内容。

版本	更改说明	更改/影响原因
1.80	更新了MISRA-C:2004合规性的API代码。	
1.70a	添加了PSoC 4资源使用情况与性能信息。	
1.70	放宽了AMux输入连接的范围。“单端”AMux的输入范围为1-64，“差分”AMux的输入范围为1-32。	满足客户的要求。
	添加了MISRA合规性章节。未证明该组件符合MISRA-C:2004编码准则。	
1.60	更改了性能表中的数据。	之前发布的性能数值有误。
	添加了AtMostOneActive和隔离参数。更新了屏幕抓图。	AtMostOne参数允许从AMux中移除横杆开关支持。 隔离参数允许选择隔离模式，以控制开关时间。
1.50.c	在数据手册中添加了“性能”一节	
1.50.b	对数据手册进行了少量编辑及更新	
1.50.a	对数据手册进行了少量编辑及更新	
1.50	添加了AMux_Init函数。	为了符合公司标准，提供了API以便无需启动组件即可初始化或恢复组件。
1.20.a	向组件中添加了消息，以说明它与芯片修订版的兼容性。	如果组件在不兼容的芯片上使用，该工具将报告错误或警告。如果发生此类情况，请更新为支持您的目标组件的修订版。
1.20	更新了符号图片。	更新的目的是为了符合公司标准。

©赛普拉斯半导体公司，2013。此处所包含的信息可能会随时更改，恕不另行通知。除赛普拉斯产品的内嵌电路以外，赛普拉斯半导体公司不对任何其他电路的使用承担任何责任。也不会以明示或暗示的方式授予任何专利许可或其他权利。除非与赛普拉斯签订明确的书面协议，否则赛普拉斯产品不保证能够用于或适用于医疗、生命支持、救生、关键控制或安全应用领域。此外，对于可能发生运转异常和故障并对用户造成严重伤害的生命支持系统，赛普拉斯不授权将其产品用作此类系统的关键组件。若将赛普拉斯产品用于生命支持系统中，则表示制造商将承担因此类使用而招致的所有风险，并确保赛普拉斯免于因此而受到任何指控。

PSoC®是赛普拉斯半导体公司的注册商标，PSoC Creator™和可编程的片上系统（Programmable System-on-Chip™）是赛普拉斯半导体公司的商标。此处引用的所有其他商标或注册商标归其各自所有者所有。

所有源代码（软件和/或固件）均归赛普拉斯半导体公司（赛普拉斯）所有，并受全球专利法规（美国和美国以外的专利法规）、美国版权法以及国际条约规定的保护和约束。赛普拉斯据此向获许可者授予适用于个人的、非独占性、不可转让的许可，用以复制、使用、修改、创建赛普拉斯源代码的派生作品、编译赛普拉斯源代码和派生作品，并且其目的只能是创建自定义软件和/或固件，以支持获许可者仅将其获得的产品依照适用协议规定的方式与赛普拉斯集成电路配合使用。除上述指定用途外，未经赛普拉斯的明确书面许可，不得对此类源代码进行任何复制、修改、转换、编译或演示。

免责声明：赛普拉斯不针对此材料提供任何类型的明示或暗示保证，包括（但不仅限于）针对特定用途的适销性和适用性的暗示保证。赛普拉斯保留在不做出通知的情况下对此处所述材料进行更改的权利。赛普拉斯不对此处所述之任何产品或电路的应用或使用承担任何责任。对于合理的预计可能发生运转异常和故障，并对用户造成严重伤害的生命支持系统，赛普拉斯不授权将其产品用作此类系统的关键组件。若将赛普拉斯产品用于生命支持系统，则表示制造商将承担因此类使用而招致的所有风险，并确保赛普拉斯免于因此而受到任何指控。

产品使用可能受适用于赛普拉斯软件许可协议的限制。