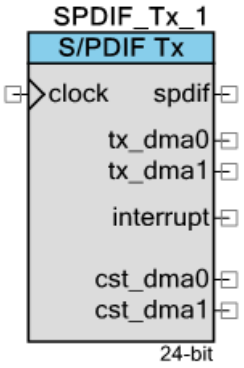
S/PDIF 发射器（SPDIF_Tx）

1.20



特性

- 符合线性 PCM 音频传输的 IEC-60958、AES/EBU、AES3 标准
- 采样率支持频率为（时钟/128）（最高频率为 192kHz）
- 可配置的音频采样长度(8/16/24)
- 消费应用的通道状态位发生器
- DMA 支持
- 独立的左声道和右声道 FIFO 或交叉立体声 FIFO

概述

SPDIF_Tx 组件可以提供在任何设计上添加数字音频输出的简单方法。它格式化输入的音频数据和原数据，用以创建适用于光纤或同轴数字音频的 S/PDIF 位流。该组件支持交叉及独立的音频模式。

SPDIF_Tx 组件接收来自 DMA 的音频数据及通道状态信息。在大部分时间内，通道状态 DMA 均由组件来管理；然而，可以选择单独指定此数据，以便更好地控制系统。

何时使用 SPDIF_Tx

无论何时需要 S/PDIF 发射器，SPDIF_Tx 组件均提供一个快速的解决方案，其中包括某些应用，例如：

- 数字音频播放器
- 计算机音频接口
- 音频主设备

组件的使用情况可能是：

- 编程时，PSoC 3 枚举为 USB 音频 HID。PSoC 3 是计算机声卡，它通过数字音频连接来播放。

- 该组件可与 I2S 组件和外部 ADC 结合使用，以将模拟音频转换为数字音频。

输入/输出连接

本节介绍 SPDIF_Tx 组件的各种输入和输出连接。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。

时钟 — 输入

提供的时钟速率必须是 Spdif 输出所需数据速率的 2 倍。例如，要生成 48 kHz 的音频，时钟频率将是：

2 × 48 kHz × 64 = 6.144 MHz

Spdif — 输出

串行数据输出。

Sck — 输出

输出串行时钟。

中断 — 输出

中断输出。

tx_DMA0 — 输出

DMA 请求音频 FIFO 0（通道 0 或交叉式）。

tx_DMA1 — 输出

DMA 请求音频 FIFO 1（通道 1）。如果选中了 **Audio Mode**（音频模式）参数的 **Separated**（独立），它将被显示。

cst_DMA0 — 输出 *

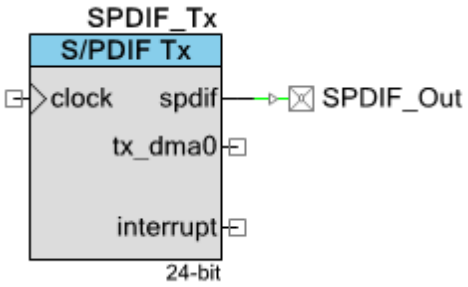
DMA 请求通道状态 FIFO 0（通道 0）。如果取消了 **Managed DMA**（管理 DMA）参数中的该复选框，它将被显示。

cst_DMA1 — 输出 *

DMA 请求通道状态 FIFO 1（通道 1）。如果取消 **Managed DMA**（管理 DMA）参数中的该复选框，它将被显示。

原理图宏信息

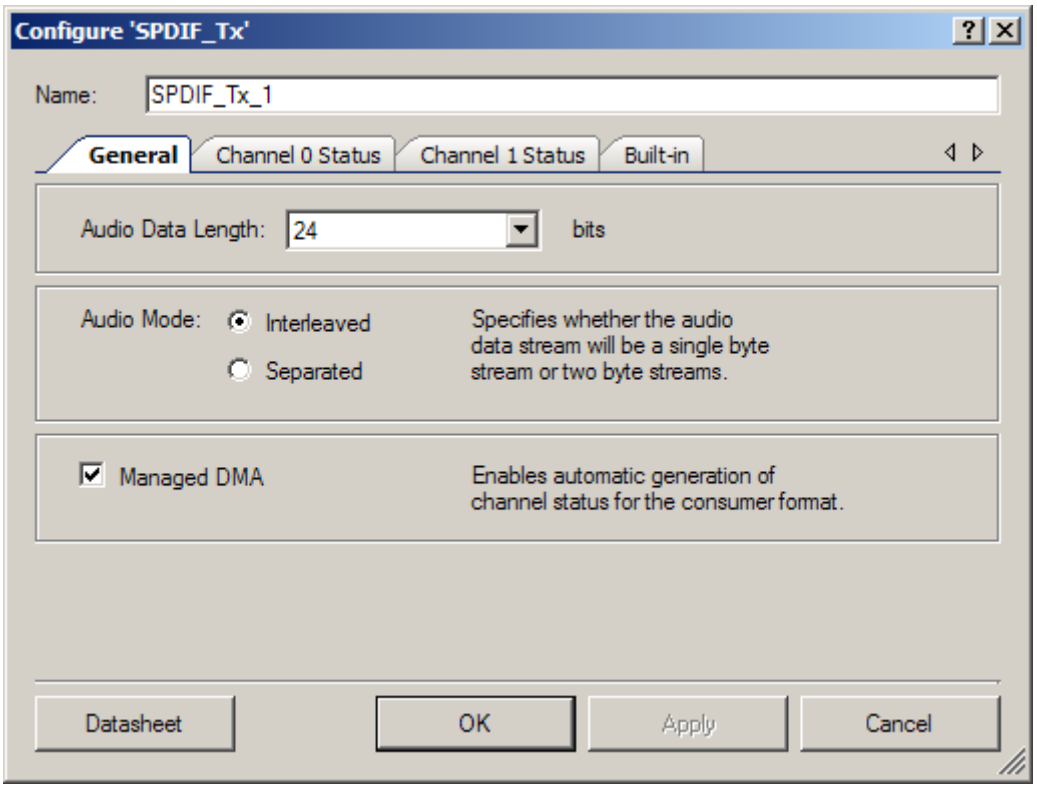
在组件目录中，默认设置的 SPDIF 发射器是使用 SPDIF 组件的原理图宏。它与数字输出引脚组件连接。针对引脚的 API 生成程序被关闭。



Component Parameters（组件参数）

将 SPDIF_Tx 组件拖入设计中，双击该组件，打开 **Configure**（配置）对话框。该对话框有 3 个选项卡，可引导您完成 SPDIF_Tx 组件的设置。

General（常规）选项卡



Audio Data Length（音频数据长度）

确定为各个采样配置的数据位数（编译的硬件）。可将该值设置为：8、16 或 24。默认设置为 **24**。

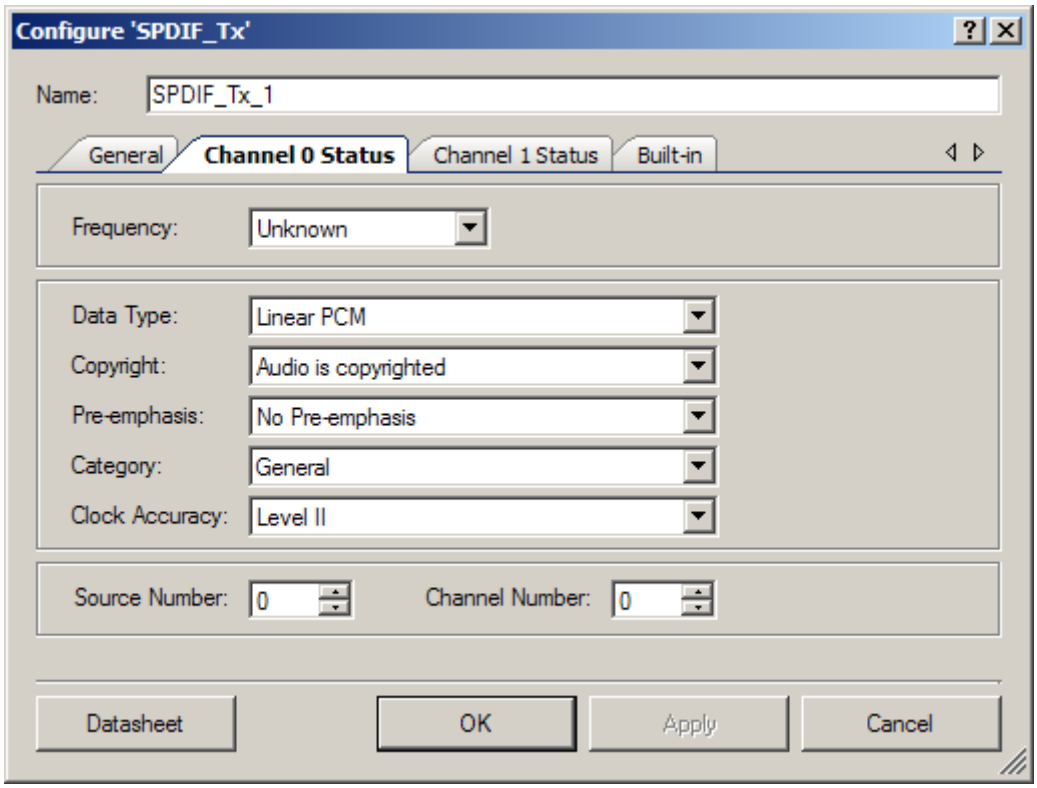
Audio Mode（音频模式)

通过此参数，您可以选择音频数据是 **Interleaved**（交叉数据）（默认设置）还是 **Separated**（独立数据）（编译的硬件）。

Managed DMA（管理 DMA）

通过此参数，您可以选择该组件是否将管理通道状态 DMA（编译的硬件）。如果选择 **Managed DMA**，则使能 **Channel 0 Status**（通道 0 状态）和 **Channel 1 Status**（通道 1 状态）选项卡。默认情况下，系统使能该选项。

Channel 0 Status（通道 0 状态）选项卡



Frequency（频率）

通过此字段，您可以为指定频率选择通道状态值。此值适用于双通道。频率源可以是：22 kHz、24 kHz、32 kHz、44 kHz、48 kHz、64 kHz、88 kHz、96 kHz、192 kHz 或未知。默认值为 **Unknown**（未知）。

Data Type（数据类型）

为通道状态 0 指定数据类型值。可将该值设置为 **Linear PCM**（线性 PCM）（默认）或其他数据。

Copyright（版权所有）

通过此字段，您可以选择 **Audio is copyrighted**（音频受版本保护）（默认值）或 **Audio is not copyrighted**（音频不受版权保护）。

Pre-emphasis（预侧重）

确定通道状态 0 的 PCM 预侧重值。可将该值设置为：**No Pre-emphasis（无侧重）**或 **50/15 µs**。默认设置为 **No Pre-emphasis**。

Category（类别）

指定通道状态 0 的类别类型。可将该值设置为 **General**（常规）（默认值）或 **Digital to Digital**（数字到数字）。

Clock Accuracy（时钟精度）

通过此字段，可以选择通道状态 0 的时钟精度。可将该值设置为：**Level I**（等级一）、**Level II**（等级二）或 **Level III**（等级三）。默认设置为 **Level II**。

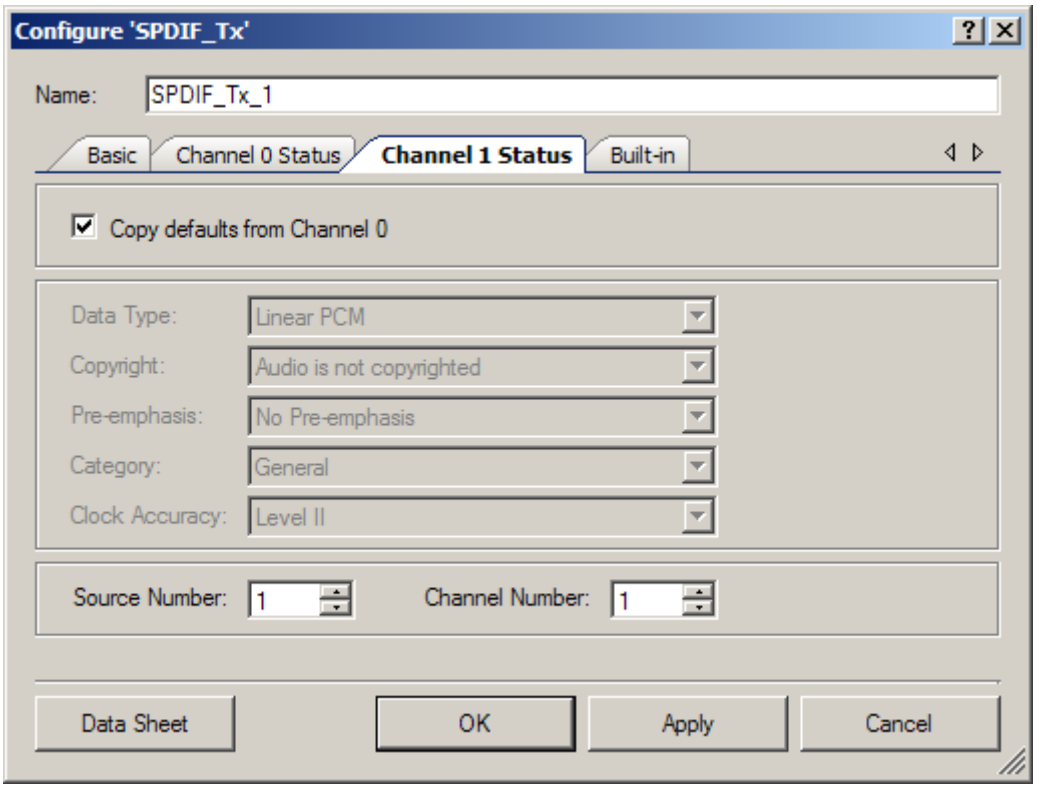
Source Number（源编号）

确定通道状态 0 的源编号。此值可设置在 0 至 15 之间。默认设置为 **0**。

Channel Number（通道编号）

确定通道状态 0 的通道编号。此值可设置在 0 至 15 之间。默认设置为 **0**。

Channel 1 Status（通道 1 状态）选项卡



Copy defaults from Channel 0（从通道 0 复制默认值）

通过此字段，可以选择通道 1 的通道状态与通道 0 相同。如果复选框被选中，**Channel 1 Status** 选项卡中所有的下拉框将被禁用。默认情况下，选中该设置。

Channel 1 Status（通道 1 状态）选项卡的剩余参数与 **Channel 0 Status**（通道 0 状态）选项卡相同。

Clock Selection（时钟选择）

该组件中没有内部时钟。您必须添加一个时钟源。提供的时钟速率必须是 Spdif 输出所需数据速率的 2 倍。

应用编程接口（API）

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

默认情况下，PSoC Creator 将给设计中第一个组件命名为“SPDIF_Tx_1”。您可以将该实例重新命名为符合标识符语法规则的任意唯一值。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。出于可读性考虑，下表中使用的实例名称为“SPDIF”。

函数	说明
SPDIF_Start()	启动S/PDIF接口。
SPDIF_Stop()	禁用S/PDIF接口。
SPDIF_Sleep()	保存配置和禁用S/PDIF接口。
SPDIF_Wakeup()	恢复S/PDIF接口的配置。
SPDIF_EnableTx()	使能S/PDIF位流中的音频数据输出。
SPDIF_DisableTx()	禁用S/PDIF 位流中的音频输出。
SPDIF_WriteTxByte()	向音频FIFO写入一个字节。
SPDIF_WriteCstByte()	向通道状态FIFO写入一个字节。
SPDIF_SetInterruptMode()	设置S/PDIF中断的中断源。
SPDIF_ReadStatus()	返回S/PDIF状态寄存器中的状态。
SPDIF_ClearTxFIFO()	清除音频FIFO。
SPDIF_ClearCstFIFO()	清除通道状态FIFO。
SPDIF_SetChannelStatus()	在运行时设置通道状态的值。
SPDIF_SetFrequency()	设置指定频率通道状态的值。

函数	说明
SPDIF_Init()	初始化或恢复S/PDIF默认配置。
SPDIF_Enable()	使能S/PDIF接口。
SPDIF_SaveConfig()	保存S/PDIF接口配置。
SPDIF_RestoreConfig()	恢复S/PDIF接口配置。

全局变量

变量	说明
SPDIF_initVar	SPDIF_initVar表示S/PDIF组件初始化是否完成。变量将初始化为0，并在第一次调用SPDIF_Start()时设置为1。这样，第一次调用SPDIF_Start()子程序后，组件不用重新初始化即可重启。 如果需要重新初始化该组件，则须首先调用SPDIF_Init()函数，然后再调用SPDIF_Start()或SPDIF_Enable()函数。
SPDIF_wrkCstStream0[]、SPDIF_wrkCstStream1[]	通道0和通道1各自的通道状态阵列。对这些阵列进行有条件编译，只在组件管理通道状态DMA时，才提供给该组件。 将通道状态流存储在两个不同缓冲区，这两个缓冲区要足够大以用来存储整个SPDIF模块流（2x24字节）。 注意：设置通道状态的值，建议使用SPDIF_SetChannelStatus()函数及通道状态宏。 例如，要设置通道0状态的类别字段，通常使用下列函数调用： SPDIF_SetChannelStatus（SPDIF_CHANNEL_0、SPDIF_CAT_GEN）；

void SPDIF_Start(void)

说明： 启动S/PDIF接口。如果配置组件以处理通道状态DMA，则启动通道状态DMA。根据需要，此函数启用活动模式电源模板位或门控时钟。通过通道状态启动S/PDIF输出的生成，而将音频数据全部设置为0。这样，S/PDIF接收器便可以锁定到组件时钟。

参数： 无

返回值： 无

其他影响： 无

void SPDIF_Stop(void)

说明： 禁用S/PDIF接口。根据需要，禁用活动模式电源模板位或门控时钟。将S/PDIF输出设置为0。清除音频数据和通道数据FIFO。SPDIF_Stop()函数调用SPDIF_DisableTx()，并停止管理通道状态DMA。

参数： 无

返回值： 无

其他影响： 无

void SPDIF_Sleep(void)

说明： 这是组件准备进入睡眠模式时的首选子程序。SPDIF_Sleep()子程序保存当前组件状态。然后调用SPDIF_Stop()函数和SPDIF_SaveConfig()以保存硬件配置。根据需要，禁用活动模式电源模板位或门控时钟。Spdif输出设置为0。

调用SPDIF_Sleep()函数，然后再调用CyPmSleep()或CyPmHibernate()函数。有关功耗管理函数的详细信息，请参考PSoC Creator *《系统参考指南》*。

参数： 无

返回值： 无

其他影响： 无

void SPDIF_Wakeup(void)

说明： 恢复SPDIF配置和非保留寄存器值。进入睡眠前，该组件停止，不考虑其状态如何。必须明确调用SPDIF_Start()函数，以便再次启动组件。

参数： 无

返回值： 无

其他影响： 如果在调用SPDIF_Wakeup()函数前未调用SPDIF_Sleep()或SPDIF_SaveConfig()函数，可能产生意外行为。

void SPDIF_EnableTx(void)

说明： 使能S/PDIF位流中的音频数据输出。传输将在下一个X或Z帧开始。

参数： 无

返回值： 无

其他影响： 无

void SPDIF_DisableTx(void)

说明： 禁用S/PDIF位流中的音频输出。数据传输将在时钟下一个上升沿停止，并将传输常量0值。

参数： 无

返回值： 无

其他影响： 无

void SPDIF_WriteTxByte(uint8 wrData, uint8 channelSelect)

说明： 向音频数据FIFO写入一个字节。在此调用前，应选中组件状态，以便确认音频数据FIFO未满状态。

参数： uint8 wrData：包含要传输的音频数据的字节。

uint8 channelSelect：包含要写入的通道常量的字节。请参见以下状态宏。在交叉模式下，此参数被忽略

返回值： 无

其他影响： 无

void SPDIF_WriteCstByte(uint8 wrData, uint8 channelSelect)

说明： 向指定通道状态FIFO写入一个字节。在此调用前，应选中组件状态，以便确认通道状态FIFO未满状态。

参数： uint8 wrData：包含要传输的状态数据的字节。

uint8 channelSelect：包含要写入的通道常量的字节。请参见以下状态宏。

返回值： 无

其他影响： 无

void SPDIF_SetInterruptMode(uint8 interruptSource)

说明： 设置S/PDIF中断的中断源。多个源可能均是“或”运算。

参数： 包含所选中断源常量的uint8字节。

SPDIF Tx中断源	Value（输入值）	类型
AUDIO_FIFO_UNDERFLOW	0x01	读取后清除
AUDIO_0_FIFO_NOT_FULL	0x02	透明
AUDIO_1_FIFO_NOT_FULL	0x04	透明
CHST_FIFO_UNDERFLOW	0x08	读取后清除
CHST_0_FIFO_NOT_FULL	0x10	透明
CHST_1_FIFO_NOT_FULL	0x20	透明

返回值： 无

其他影响： 如果将读取后清除的位作为中断生成的源使用，则中断输出将保持置位，直至读取SPDIF状态寄存器。

uint8 SPDIF_ReadStatus(void)

说明： 返回SPDIF状态寄存器中的状态。

参数： 无

返回值： SPDIF状态寄存器的uint8状态。

SPDIF状态掩码	数值	类型
AUDIO_FIFO_UNDERFLOW	0x01	读取后清除
AUDIO_0_FIFO_NOT_FULL	0x02	透明
AUDIO_1_FIFO_NOT_FULL	0x04	透明
CHST_FIFO_UNDERFLOW	0x08	读取后清除
CHST_0_FIFO_NOT_FULL	0x10	透明
CHST_1_FIFO_NOT_FULL	0x20	透明

其他影响： 清除“读取后清除”类型的SPDIF状态寄存器位。

void SPDIF_ClearTxFIFO(void)

说明： 清除音频数据FIFO。FIFO中提供的所有数据均将丢失。在独立音频模式下，两个音频FIFO将被清除。仅当禁用传输时，才调用此函数。

参数： 无

返回值： 无

其他影响： 无

void SPDIF_ClearCstFIFO(void)

说明： 清除通道状态FIFO。在任意一个FIFO中提供的所有数据均将丢失。仅当组件停止时，才调用此函数。

参数： 无

返回值： 无

其他影响： 无

void SPDIF_SetChannelStatus(uint8 channel, uint8 byte, uint8 mask, uint8 value)

说明： 在运行时设置通道状态的值。仅在该组件管理DMA时，此API才有效。

参数： uint8 channel：包含用来指定要修改通道的常量的字节。请参见以下状态宏。

uint8 byte：要修改的字节[0..23]。请参见以下状态宏。

uint8 mask：字节中的掩码。请参见以下状态宏。

uint8 value：设置的值。请参见以下状态宏。

返回值： 无

其他影响： 无

uint8 SPDIF_SetFrequency(uint8 frequency)

说明： 设置指定频率通道状态的值，并返回值1。此函数仅在组件被停止时才起作用。如果在启动组件时调用此函数，则返回0，并且不将修改任何值。仅在该组件管理DMA时，此API才有效。

参数： uint8：包含指定频率常量的字节。

名称	说明
SPDIF_SPS_22KHZ	为22 kHz音频设置的时钟频率
SPDIF_SPS_44KHZ	为44 kHz音频设置的时钟频率
SPDIF_SPS_88KHZ	为88 kHz音频设置的时钟频率
SPDIF_SPS_24KHZ	为24 kHz音频设置的时钟频率
SPDIF_SPS_48KHZ	为48 kHz音频设置的时钟频率
SPDIF_SPS_96KHZ	为96 kHz音频设置的时钟频率
SPDIF_SPS_32KHZ	为32 kHz音频设置的时钟频率
SPDIF_SPS_64KHZ	为64 kHz音频设置的时钟频率
SPDIF_SPS_192KHZ	为192 kHz音频设置的时钟频率
SPDIF_SPS_UNKNOWN	未指定时钟频率

返回值： uint8： 成功时返回值为1
失败时返回值为0

其他影响： 无

void SPDIF_Init(void)

说明： 初始化或恢复自定义程序提供的默认S/PDIF配置，因此在配置组件以用来处理通道状态DMA时定义组件的中断源和通道状态。

参数： 无

返回值： 无

其他影响： 配置组件以处理通道状态DMA时，仅恢复中断生成的掩码寄存器和通道状态。它不会从FIFO中清除数据，也不会复位组件硬件状态机。

void SPDIF_Enable(void)

说明： 激活硬件，并开始执行组件操作。无需调用SPDIF_Enable()，因为SPDIF_Start()子程序会调用此函数，这是开始执行组件操作的首选方法。

参数： 无

返回值： 无

其他影响： 无

void SPDIF_SaveConfig(void)

说明： 此函数保存组件配置。它将保存非保留寄存器。此函数还将保存当前“Configure”对话框中所定义的或通过相应API修改的组件参数值。通过SPDIF_Sleep()函数调用此函数。

参数： 无

返回值： 无

其他影响： 无

void SPDIF_RestoreConfig(void)

说明： 此函数恢复组件配置。这将恢复非保留寄存器。此外，该函数还将组件参数值恢复到调用SPDIF_Sleep()函数之前的值。此子程序由SPDIF_Wakeup()来调用，从而恢复退出睡眠状态时的组件。

参数： 无

返回值： 无

其他影响： 必须在调用SPDIF_SaveConfig()子程序之后调用。否则，该组件配置将被其原始设置所覆盖。

通道状态宏

通道状态宏用来封装指定通道状态设置的掩码和值。这些宏及通道宏用于在运行时通过SPDIF_SetChannelStatus() API 设置通道状态值。

建议最好在相同时间内设置这两个通道值。

如下例所示：

```
SPDIF_SetChannelStatus(SPDIF_CHANNEL_0, SPDIF_DATA_TYPE_LINEAR_PCM);
```

通道名称常量

名称	说明
SPDIF_CHANNEL_0	通道0
SPDIF_CHANNEL_1	通道1

通道状态常量

名称	字节	掩码	Value（输入值）
SPDIF_DATA_TYPE_LINEAR_PCM	0	0x02	0x00
SPDIF_DATA_TYPE_OTHERDATA	0	0x02	0x02
SPDIF_COPY_HAS_CP_RIGHT	0	0x04	0x00
SPDIF_COPY_NO_CP_RIGHT	0	0x04	0x04
SPDIF_PREEMP_NO_PREEMP	0	0x38	0x00
SPDIF_PREEMP_PREEMP50	0	0x38	0x08
SPDIF_CAT_GEN	1	0xFF	0x00
SPDIF_CAT_D2D	1	0xFF	0x02
SPDIF_SRC_NUM00	2	0x0F	0x00
SPDIF_SRC_NUM01	2	0x0F	0x01
SPDIF_SRC_NUM02	2	0x0F	0x02
SPDIF_SRC_NUM03	2	0x0F	0x03
SPDIF_SRC_NUM04	2	0x0F	0x04
SPDIF_SRC_NUM05	2	0x0F	0x05
SPDIF_SRC_NUM06	2	0x0F	0x06
SPDIF_SRC_NUM07	2	0x0F	0x07
SPDIF_SRC_NUM08	2	0x0F	0x08
SPDIF_SRC_NUM09	2	0x0F	0x09
SPDIF_SRC_NUM10	2	0x0F	0x0A
SPDIF_SRC_NUM11	2	0x0F	0x0B
SPDIF_SRC_NUM12	2	0x0F	0x0C
SPDIF_SRC_NUM13	2	0x0F	0x0D
SPDIF_SRC_NUM14	2	0x0F	0x0E
SPDIF_SRC_NUM15	2	0x0F	0x0F

名称	字节	掩码	Value（输入值）
SPDIF_CH_NUM00	2	0xF0	0x00
SPDIF_CH_NUM01	2	0xF0	0x10
SPDIF_CH_NUM02	2	0xF0	0x20
SPDIF_CH_NUM03	2	0xF0	0x30
SPDIF_CH_NUM04	2	0xF0	0x40
SPDIF_CH_NUM05	2	0xF0	0x50
SPDIF_CH_NUM06	2	0xF0	0x60
SPDIF_CH_NUM07	2	0xF0	0x70
SPDIF_CH_NUM08	2	0xF0	0x80
SPDIF_CH_NUM09	2	0xF0	0x90
SPDIF_CH_NUM10	2	0xF0	0xA0
SPDIF_CH_NUM11	2	0xF0	0xB0
SPDIF_CH_NUM12	2	0xF0	0xC0
SPDIF_CH_NUM13	2	0xF0	0xD0
SPDIF_CH_NUM14	2	0xF0	0xE0
SPDIF_CH_NUM15	2	0xF0	0xF0
SPDIF_CLKLVL_1	3	0x30	0x10
SPDIF_CLKLVL_2	3	0x30	0x00
SPDIF_CLKLVL_3	3	0x30	0x20
SPDIF_STDLEN	4	0x0F	0x00
SPDIF_24BLEN	4	0x0F	0x0B

MISRA 符合性（compliance）

本节介绍了MISRA-C:2004符合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 — 仅适用于此组件的偏差

本节介绍有关组件特定偏差的信息。*《系统参考指南》*的“MISRA 符合性”章节中介绍项目偏差以及有关 MISRA 符合性验证环境的信息。

SPDIF_Tx 组件具有以下特定偏差：

MISRA-C：2004 规则	规则类别（必须/建议）	规则说明	偏差描述
10.1	必须	某些情况下，整数类型表达式的值不应隐式转换为不同的底层类型，如果： a) 没有转换到相同符号的更宽的整数类型，或 b)表达式是复合的，或 c)表达式不是一个常量，而是函数参数，或 d)表达式不是一个常量，而是返回表达式。	库函数memcpy具有通用的int参数，该参数用于设置需要复制的字节数。 无符号的8位整数作为此函数的参数传递。 这个操作不会引起其他任何影响，因为要复制的字节数总是小于256。 DMA组件提供了各个通用整数类型的定义。
11.5	必须	不执行转换操作，以删除由指针寻址的类型中所有的常量或易失性资质。	库函数memcpy具有void参数的指针指向源和目标。 指向易失性阵列的指针作为源和目标参数传递并失去其资质。 memcpy函数可以确保正确的操作并安全使用它。
13.2	建议	对一个非零值进行显性测试，除非操作数是有效的布尔（Boolean）值。	DMA组件提供了对各个通用整数类型的定义。通过对这些定义进行“或”（OR）运算，可以得到各函数的正确参数。这些定义为正数，运算结果不会产生其他影响。
17.4	必须	数组索引是唯一允许的指针运算形式。	为了访问DMA寄存器，使用数组索引的指针运算形式。 DMA组件提供对这种类型的访问。

此组件具有以下嵌入式组件：DMA。MISRA 合规性与特定偏差的相关信息，请参见相应的组件数据手册。

固件源代码示例

在 Find Example Project 对话框中，PSoC Creator 提供了大量的示例项目，包括原理图和示例代码。要获取组件示例，请打开组件目录中的对话框或原理图中的组件实例。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** 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”的内容。

功能说明

此组件用来格式化输入的音频数据和原数据以创建 S/PDIF 位流。该组件接收来自 DMA 的音频数据及通道状态信息。在大部分时间内，通道状态 DMA 均由组件来管理；然而，您可以选择单独指定此数据，以便更好地控制系统。

数据流格式

音频和通道状态数据是独立的字节流。字节流从最低有效字节和有效位开始被压缩。每个采样所使用的字节数是保持采样的最少字节数。任何未使用的位从最左侧的位开始使用 0 填塞。

音频数据流即可以是一个字节流，也可以是两个字节流。在一个字节流的情况下，左通道和右通道与采样互相交错，左通道后面紧接右通道。在双流情况下，左右通道字节流均使用各自的 FIFO。该状态字节流始终为双字节流。

DMA

S/PDIF 接口是要求未中断数据流的连接接口。对大多数应用而言，这需要使用 DMA 传输，以便防止音频数据或通道状态 FIFO 的下溢。通常情况下，在该组件中使用两个通道状态阵列完整地执行通道状态 DMA 并通过使用宏可修改它；然而，您可以使用自定义 DMA 或 CPU 来提供数据，从而提高灵活性。

根据组件配置，S/PDIF 可以驱动高达 4 个 DMA 组件。DMA 向导用于配置如下 DMA 操作：

在DMA向导中DMA源/目录的名称	方向	DMA请求信号	DMA请求类型	说明
SPDIF_TX_FIFO_0_PTR	目标	tx_dma0	电平	传输通道0的FIFO，或交错音频数据
SPDIF_TX_FIFO_1_PTR	目标	tx_dma1	电平	传输通道1音频数据的FIFO
SPDIF_CST_FIFO_0_PTR	目标	cst_dma0	电平	传输通道0通道状态数据的FIFO
SPDIF_CST_FIFO_1_PTR	目标	cst_dma1	电平	传输通道1通道状态数据的FIFO

在所有情况下，DMA 请求信号上的高电平信号表示可以传输一个附加字节。

错误处理

如果音频为空，并出现后续读取事件（传输下溢）或通道状态 FIFO 为空，而发生后续读取事件（状态下溢），则该组件可能发生两个错误条件。

如果出现传输下溢，则组件强制传输音频数据常量 0，并继续修正所有帧和状态数据的生成。再次启动传输前，必须禁用传输，并不应清除 FIFO，必须缓冲传输数据，然后重新使能传输。使用组件状态位 AUDIO_FIFO_UNDERFLOW 通过 CPU 来监控此下溢条件。此外，还可以针对此错误条件配置中断。

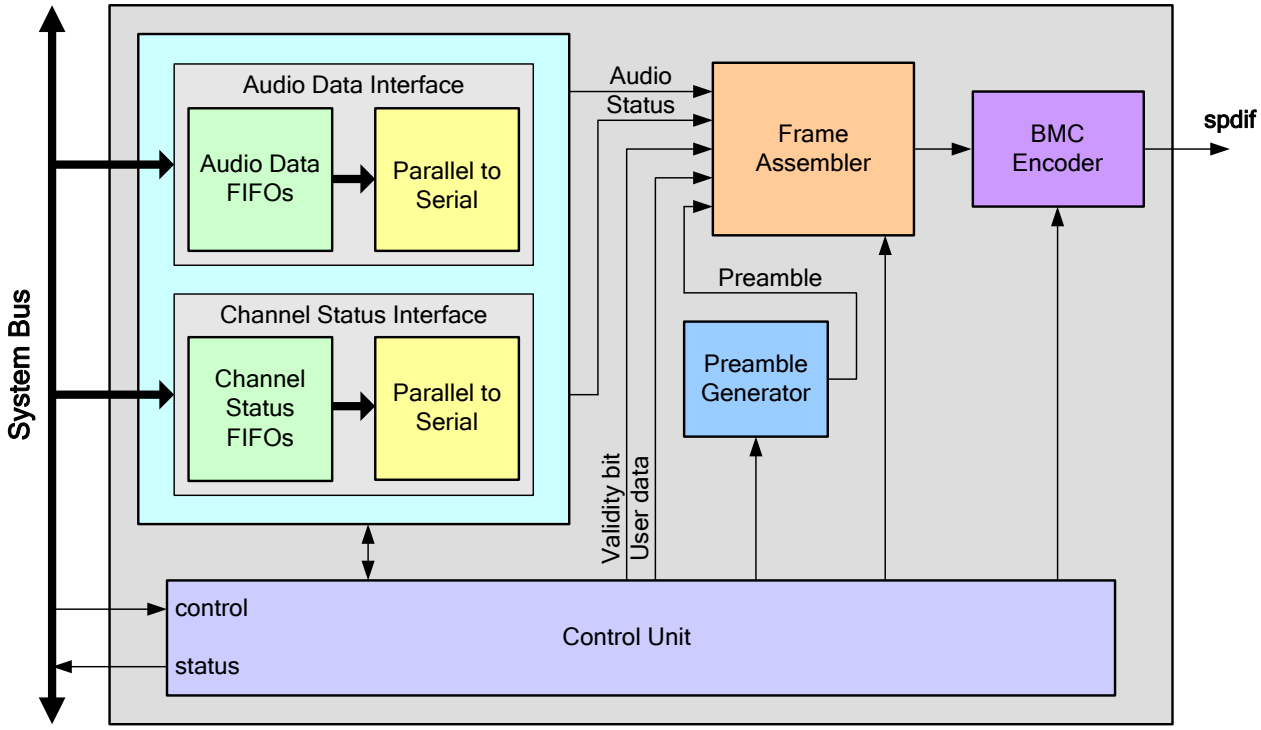
启动组件时，如果发生状态下溢，则该组件将通过正确生成 X、Y、Z 帧和正确的奇偶校验位来为通道状态发送所有 0 值。音频数据是连续的，而不受影响。要正确传输通道状态，必须停止组件，然后再启组件。使用状态位 CHST_FIFO_UNDERFLOW 通过 CPU 来监控此下溢条件。此外，还可以针对此错误条件配置中断。如果组件未管理 DMA，则必须在重启组件前缓冲状态数据。

使能系统

音频数据传输具有专用的使能系统。启动组件而非使能组件时，在通道状态下生成 S/PDIF 输出，而将音频数据全部设置为 0。这样，S/PDIF 接收器便可以锁定到组件时钟上。在 X 或 Z 帧发生时，切换为使能状态。

框图和配置

SPDIF_Tx 可以实现为一组配置 UDB。在以下框图中显示该实现。



通过系统总线接口接收输入的音频数据，并通过 CPU 或 DMA 提供该音频数据。该数据是从最低有效字节开始的字节宽，存储在音频缓冲区（一个或两个 FIFO，这取决于组件配置）中。

通道状态流具有自己的专用接口。与音频数据类似，它也有两个通道状态 FIFO。此外，通道状态为从最低有效字节开始的字节宽数据。从这些 FIFO 中，每 8 个采样使用一个字节。音频数据和状态数据均从并行格式转变成串行格式。

在 S/PDIF 标准中未定义用户数据，某些接收器可能忽略该用户数据，因此这些数据将作为常量 0 发送。

当有效位为低位时，表示音频采样适合于模拟转换。此位作为常量 0 发送。

在前导码发生器模块中生成前导码模式，然后以串行格式传输。

这是构成 SPDIF 子帧结构的所有必要数据，但奇偶校验位除外。将所有输入汇集到子帧结构时，在帧汇编程序模块中计算奇偶校验位。

帧汇编程序模块的输出转到 BMC 编码器，在此将数据编译到 Spdif 输出中。

控制单元模块从系统总线接口获得控制数据，并将组件操作状态返回到总线。传输数据过程中，它控制其他所有模块。

寄存器

SPDIF_Tx_CONTROL_REG

位	7	6	5	4	3	2	1	0
值	Reserved（保留）						enable（使能）	Txenable

- 使能：使能/禁用 SPDIF_Tx 组件。未使能时，组件处于复位状态。
- txenable：使能/禁用 S/PDIF 位流中的音频数据输出。

SPDIF_Tx_STATUS_REG

位	7	6	5	4	3	2	1	0
值	保留		chst1_fifo_not_full	chst0_fifo_not_full	chst_fifo_underflow	audio1_fifo_not_full	audio0_fifo_not_full	audio_fifo_underflow

- chst1_fifo_not_full：如果设置此位，通道状态 FIFO 1 未满
- chst1_fifo_not_full：如果设置此位，通道状态 FIFO 0 未满
- chst_fifo_underflow：如果设置此位，会发生通道状态 FIFO 下溢事件
- audio1_fifo_not_full：如果设置此位，音频数据 FIFO 1 未满
- audio0_fifo_not_full：如果设置此位，音频数据 FIFO 0 未满
- audio_fifo_underflow:如果设置此位，会发生音频数据 FIFO 下溢事件

通过 SPDIF_Tx_ReadStatus() API 函数读取寄存器值。

注意：将状态寄存器的位 3 和位 0 配置为“读取后清除”。在此模式下，在状态寄存器时钟的每个周期完成后，会对输入状态进行采样。当输入为高电平时，设置寄存器位，并保持设置，而无需考虑输入的后续状态。在后续读取时，通过 CPU 清除寄存器位。

资源

SPDIF 组件放置在整个 UDB 阵列中。该组件利用以下资源。

配置	资源类型					
	数据路径单元	宏单元	状态单元	控制单元	DMA通道	中断
管理的DMA	4	23	1	2	2	2
未管理的DMA	4	23	1	2	–	–

API 存储器的使用情况

根据不同编译程序、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。下表提供了给定组件配置中的所有 API 的存储器使用情况。

下表中的存储器大小是在将相应编译器设置为 Release 模式并且优化选项为 Size 的情况下测得的。对于特定的设计，分析编译器生成的映射文件后可以确定组件占用存储器的大小。

配置	PSoC 3（Keil_PK51）		PSoC 5（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
管理的DMA	1507	107	1506	115	1402	111
未管理的DMA	276	1	534	5	430	1

直流和交流的电气特性

除非另有说明，否则这些规范的适用条件是：-40°C ≤ T_A ≤ 85 °C 且 T_J ≤ 100 °C。除非另有说明，否则这些规范的适用范围为 1.71 V 到 5.5 V。

直流特性

参数	说明	最小值	典型值 [1]	最大值	单位[2]
$I_{DD(DMA)}$	组件电流消耗（管理的DMA）				
	闲空电流[3]	–	34	–	µA/MHz
	工作电流[4]	–	39	–	µA/MHz
$I_{DD(NO_DMA)}$	组件电流消耗（未管理DMA）				
	空闲电流[3]	–	30	–	µA/MHz
	工作电流[4]	–	35	–	µA/MHz

交流特性

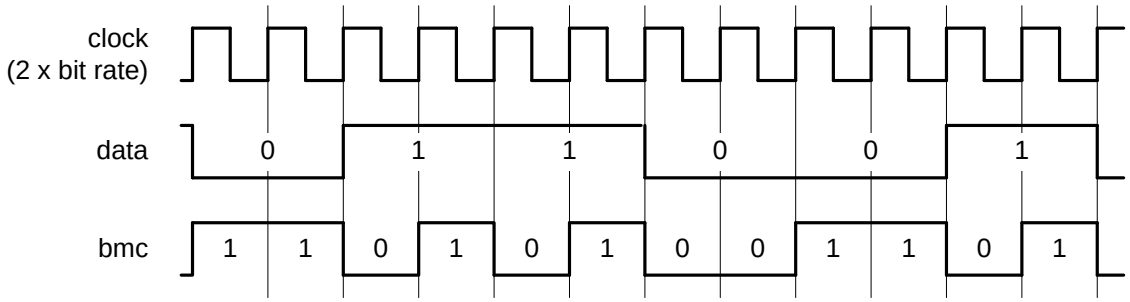
参数	说明	最小值	典型值	最大值	单位
f_S	采样频率	–	–	192	kHz
f_{CLOCK}	组件时钟频率	–	128 × f_S	–	MHz

S/PDIF 通道编码

S/PDIF 是单线串行接口。位时钟嵌入 S/PDIF 数据流中。使用双相符号编码（BMC）对数字信号进行编码，这是一种相位调制。时钟频率是位速率的两倍。原始数据的每一个位表示两个逻辑状态，这两个状态一起形成一个单元。在开始位的逻辑电平始终转换成上一个结束位的电平。要使

1. 未包括设备 IO 和时钟分配的电流。这些值是在温度为 25 °C 时的值。
2. 电流消耗与组件的输入时钟相关。
3. 组件使能但不传输数据时消耗电流。
4. 组件使能且传输数据时消耗电流。

用此格式传输‘1’，在数据位边界中间有一次跃变。如果在此中间未发生跃变，则该数据被视为‘0’。



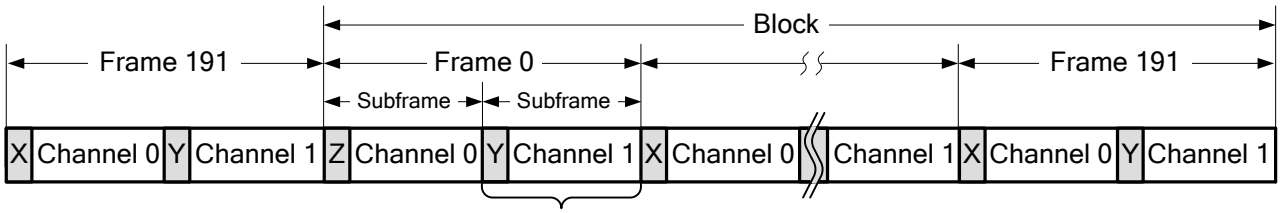
S/PDIF 协议层次

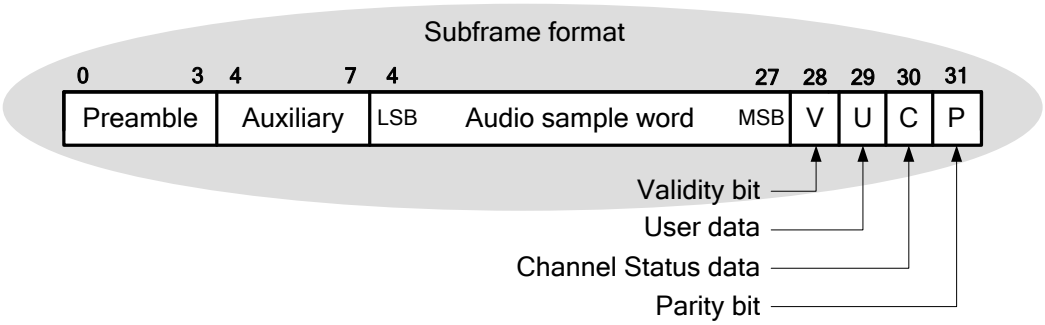
下图说明了 SPDIF 信号格式。音频数据以连续模块的格式进行传输。模块包含 192 个帧。每帧由两个子帧组成，这两个子帧是组织数字音频数据的基本组件。

子帧包含：

- 前导码模式
- 高达 24 位宽的单一音频采样
- 表示采样是否有效的有效位
- 包含用户数据的位
- 包含通道状态的位
- 该子帧的偶校验位

有三种前导码类型：X、Y 和 Z。前导码 Z 指示模块的起始位和子帧通道 0 的起始位。如果前导码 X 并非位于模块的起始位时，会指示通道 0 子帧。前导码 Y 始终表示通道 1 子帧的起始位。


Block
Frame 191
Frame 0
Frame 191
Subframe
Subframe
X
Channel 0
Y
Channel 1
Z
Channel 0
Y
Channel 1
X
Channel 0
Channel 1
X
Channel 0
Y
Channel 1


Subframe format
0
3
4
7
4
27
28
29
30
31
Preamble
Auxiliary
LSB
Audio sample word
MSB
V
U
C
P
Validity bit
User data
Channel Status data
Parity bit

组件更改

本节列出了各版本的主要组件更改内容。

版本	更改说明	更改原因/影响
1.20	已添加了MISRA符合性章节。	该组件具有所描述的特定偏差。
1.10	向“.cyre”文件中包括的所有组件API中添加了CYREENTRANT关键词。	并非所有API都是真正可重入的函数。组件API源文件中的注释指出了适用的函数。 对于采用了安全方式并且是不可重入的函数，则需要该项变更，这样可以消除编译器警告：通过标志或关键节防止同时调用。
	添加了PSoC 5LP支持	
	向数据手册中添加了特性部分。	

赛普拉斯半导体公司，2013-2016 年。本文件是赛普拉斯半导体公司及其子公司，包括 Spansion LLC（“赛普拉斯”）的财产。本文件，包括其包含或引用的任何软件或固件（“软件”），根据全球范围内的知识产权法律以及美国与其他国家签署条约由赛普拉斯所有。除非在本款中另有明确规定，赛普拉斯保留在该等法律和条约下的所有权利，且未就其专利、版权、商标或其他知识产权授予任何许可。如果软件并不附随有一份许可协议且贵方未以其他方式与赛普拉斯签署关于使用软件的书面协议，赛普拉斯特此授予贵方属人性质的、非独家且不可转让的如下许可（无再许可权）（1）在赛普拉斯特软件著作权项下的下列许可权（一）对以源代码形式提供的软件，仅出于在赛普拉斯硬件产品上使用之目的且仅在贵方集团内部修改和复制软件，和（二）仅限于在有关赛普拉斯硬件产品上使用之目的将软件以二进制代码形式的向外部最终用户提供（无论直接提供或通过经销商和分销商间接提供），和（2）在被软件（由赛普拉斯公司提供，且未经修改）侵犯的赛普拉斯专利的权利主张项下，仅出于在赛普拉斯硬件产品上使用之目的制造、使用、提供和进口软件的许可。禁止对软件的任何其他使用、复制、修改、翻译或汇编。

在适用法律允许的限度内，赛普拉斯未对本文件或任何软件作出任何明示或暗示的担保，包括但不限于关于适销性和特定用途的默示保证。赛普拉斯保留更改本文件的权利，届时将不另行通知。在适用法律允许的限度内，赛普拉斯不对因应用或使用本文件所述任何产品或电路引起的任何后果负责。本文件，包括任何样本设计信息或程序代码信息，仅为供参考之目的提供。文件使用人应负责正确设计、计划和测试信息应用和由此生产的任何产品的功能和安全性。赛普拉斯产品不应被设计为、设定为或授权用作武器操作、武器系统、核设施、生命支持设备或系统、其他医疗设备或系统（包括急救设备和手术植入物）、污染控制或有害物质管理系统中的关键部件，或产品植入之设备或系统故障可能导致人身伤害、死亡或财产损失其他用途（“非预期用途”）。关键部件指，若该部件发生故障，经合理预期会导致设备或系统故障或会影响设备或系统安全性和有效性的部件。针对由赛普拉斯产品非预期用途产生或相关的任何主张、费用、损失和其他责任，赛普拉斯不承担全部或部分责任且贵方不应追究赛普拉斯之责任。贵方应赔偿赛普拉斯因赛普拉斯产品任何非预期用途产生或相关的所有索赔、费用、损失和其他责任，包括因人身伤害或死亡引起的主张，并使之免受损失。

赛普拉斯、赛普拉斯徽标、Spansion、Spansion 徽标，及上述项目的组合，WICED，及 PSoC、CapSense、EZ-USB、F-RAM 和 Traveo 应视为赛普拉斯在美国和其他国家的商标或注册商标。请访问 cypress.com 获取赛普拉斯商标的完整列表。其他名称和品牌可能由其各自所有者主张为该方财产。