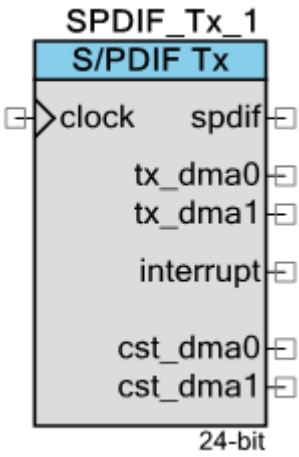
S/PDIF トランスミッタ (SPDIF_Tx)

1.0

特長



- リニア PCM オーディオ送信について IEC-60958、AES/EBU、AES3 基準に準拠
- クロック/128 のサンプルレートをサポート (最大 192 kHz)
- 設定可能なオーディオサンプル長 (8/16/24)
- 民生用アプリケーションのチャネルステータスビットジェネレータ
- DMA サポート
- 独立した左右のチャネル FIFO または交互送信によるステレオ FIFO

概要説明

SPDIF_Tx コンポーネントにより、簡単な方法で設計にデジタルオーディオ出力を追加できます。光または同軸デジタルオーディオに適切な S/PDIF ビットストリームを作成するため、入力オーディオデータおよびメタデータをフォーマットします。コンポーネントはインターリーブ、及びセパレートオーディオをサポートします。

SPDIF_Tx コンポーネントは DMA からのオーディオデータと同様にチャネルステータス情報を受信します。大半の場合、チャネルステータス DMA はコンポーネントにより管理されます。ただし、システムをより良く制御するために、このデータを分離して指定するオプションがあります。

SPDIF_Tx を使用するタイミング

SPDIF_Tx コンポーネントは、以下のようなアプリケーションなど S/PDIF トランスミッタが必要な場合には必ず高速ソリューションを提供します。

- デジタルオーディオプレーヤー
- コンピュータオーディオインタフェース
- オーディオマスタリング機器

コンポーネントの使用例:

- プログラムした場合、PSoC 3 は USB オーディオ HID としてエニュメレートされます。PSoC 3 はコンピュータのサウンドカードとして、デジタルオーディオ接続経由で再生します。
- このコンポーネントを I2S コンポーネントおよび外部 ADC と組み合わせて、アナログオーディオからデジタルオーディオに移行するのに使用できます。

入出力接続

ここでは、SPDIF_Tx コンポーネントのさまざまな入出力接続について説明します。I/O リストのアスタリスク (*) は、I/O が、その I/O の説明でリストされている条件において、シンボル上から隠されている可能性があることを示します。

clock – 入力

提供されるクロックレートは spdif 出力で必要とされるデータレートの 2 倍にする必要があります。たとえば 48 kHz オーディオを生成するには、クロック周波数は以下となります。

2 × 48 kHz × 64 = 6.144 MHz

spdif – 出力

シリアルデータ出力。

sck – 出力

出力シリアルクロック。

interrupt – 出力

割り込み出力。

tx_DMA0 – 出力

オーディオ FIFO 0 の DMA リクエスト (Channel 0 または インターリーブ)。

tx_DMA1 – 出力

オーディオ FIFO 1 の DMA リクエスト (Channel 1)。**Audio Mode** パラメータ下で **Separated** を選択すると表示されます。

cst_DMA0 – 出力*

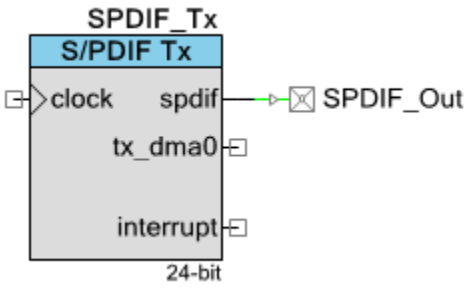
チャネルステータス FIFO 0 の DMA リクエスト (Channel 0)。**Managed DMA** パラメータ下のチェックボックスを選択解除すると表示されます。

cst_DMA1 – 出力*

チャネルステータス FIFO 1 の DMA リクエスト (Channel 1)。**Managed DMA** パラメータ下のチェックボックスを選択解除すると表示されます。

回路図マクロ情報

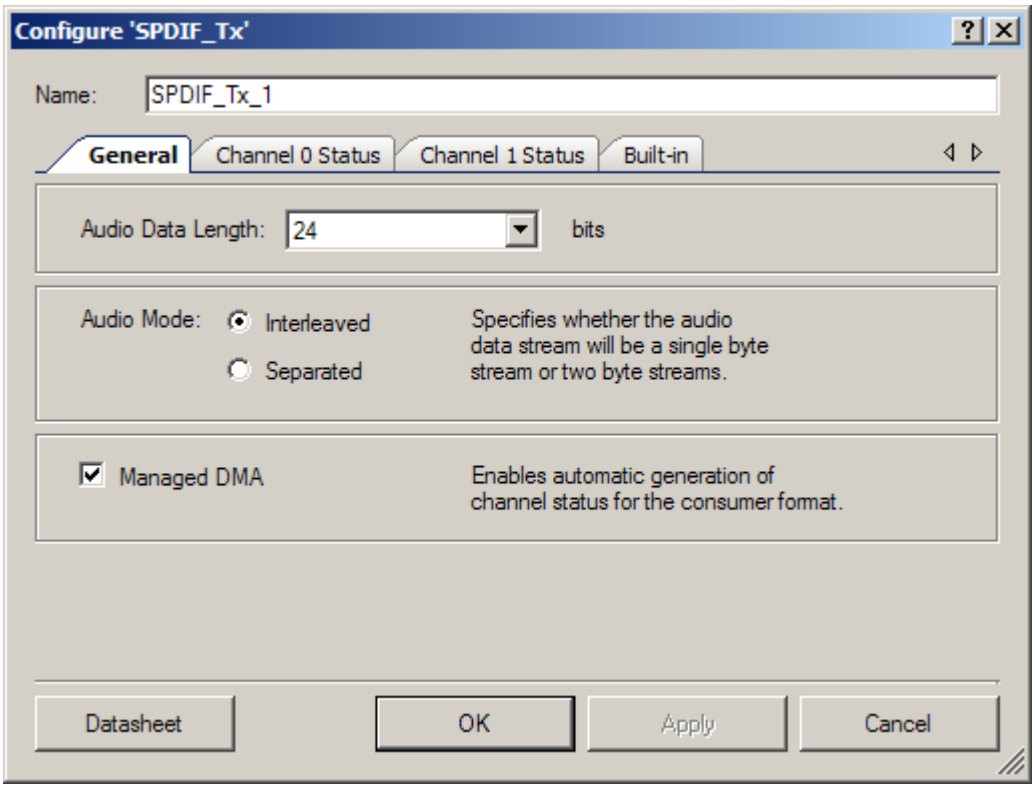
コンポーネントカタログ内のデフォルトの SPDIF トランスミッタは、デフォルト設定の SPDIF コンポーネントを使用した回路図マクロです。これは、デジタル出力ピンコンポーネントに接続されています。ピン用の API の生成はオフになっています。



コンポーネント・パラメータ

SPDIF_Tx コンポーネントをデザイン上にドラッグし、ダブルクリックして **Configure** ダイアログを開きます。このダイアログには、SPDIF_Tx コンポーネントのセットアップをガイドする 3 つのタブがあります。

General タブ



オーディオデータ長

各サンプルに設定されたデータビット数を決定します (ハードウェアコンパイル)。この値は以下の値に設定できます。8、16、または 24。デフォルトの設定は **24** です。

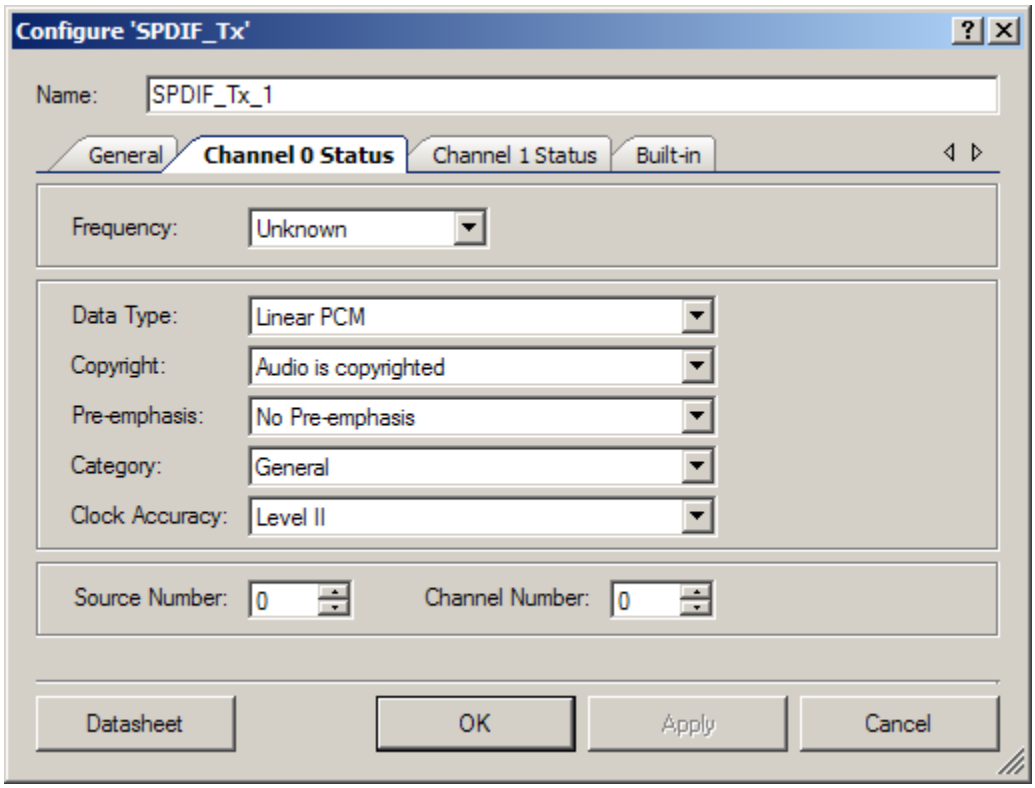
オーディオモード

オーディオデータに **Interleaved** (デフォルト) または **Separated** (ハードウェアコンパイル) のいずれかを選択できます。

管理 DMA

コンポーネントがチャネルステータス DMA (ハードウェアコンパイル) を管理するかどうかを選択できます。**Managed DMA** が選択されていると、**Channel 0 Status** タブおよび **Channel 1 Status** タブが有効になります。このオプションはデフォルトで有効になっています。

Channel 0 Status タブ



周波数

指定した周波数のチャネルステータスの値を選択できます。この値は両方のチャネルに適用されます。指定できるソース周波数：22 kHz、24 kHz、32 kHz、44 kHz、48 kHz、64 kHz、88 kHz、96 kHz、192 kHz、または Unknown。デフォルトは **Unknown** です。

データの種類

チャネルステータス 0 のデータタイプを指定します。この値を **Linear PCM** (デフォルト) または他のデータに設定できます。

Copyright

Audio is copyrighted (デフォルト) または **Audio is not copyrighted** のいずれかを選択できます。

プリエンファシス

チャネルステータス 0 の PCM プリエンファシス値を決定します。この値は以下に設定できます。**No Preemphasis** または **50/15 µs**。デフォルト設定は **No Pre-emphasis** です。

カテゴリ

チャネルステータス 0 のカテゴリの種類を指定します。この値は **General** (デフォルト) または **Digital to Digital** に設定できます。

クロック精度

チャネルステータス 0 のクロック精度を選択できます。この値は以下に設定できます。**Level I**、**Level II**、または **Level III**。デフォルトの設定は **Level II** です。

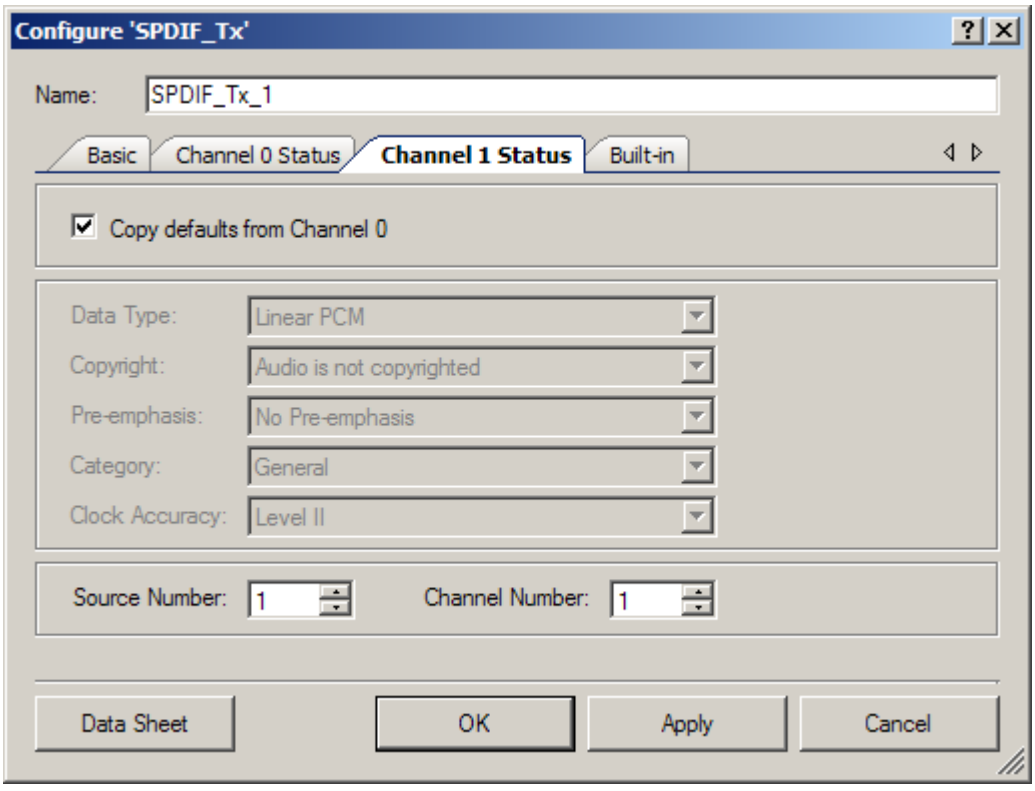
ソース数

チャネルステータス 0 のソース数を決定します。この値は 0 ～ 15 に設定できます。デフォルトの設定は **0** です。

チャネル数

チャネルステータス 0 のチャネル数を決定します。この値は 0 ～ 15 に設定できます。デフォルトの設定は **0** です。

Channel 1 Status タブ



Copy defaults from Channel 0

チャネル 1 のチャネルステータスがチャネル 0 と同一かどうかを選択できます。チェックボックスが選択されている場合、**Channel 1 Status** タブの全てのドロップダウンボックスは無効になります。この設定はデフォルトでチェックされています。

Channel 1 Status タブの残りのパラメータは **Channel 0 Status** タブと同一です。

クロック選択

このコンポーネントには、内部クロックはありません。クロックソースを必ず取りつけてください。提供されるクロックレートは spdif 出力で必要とされるデータレートの 2 倍にする必要があります。

配置

SPDIF_Tx コンポーネントは、UDB アレイ全体に配置され、すべての配置情報は、*cyfitter.h* ファイルを通して API に提供されます。

リソース

リソース	リソースのタイプ						API メモリ（バイト）		ピン（外部入出力ごと）
	データパスセル	PLD	ステータスセル	Control/Count7 セル	Drqs	Irqs	フラッシュ	RAM	
DMA取扱い有	4	9	1	2	2	2	1420	106	0
DMA取扱い無し	4	9	1	2	0	0	290	2	0

アプリケーション プログラミング インタフェース

アプリケーション・プログラミング・インターフェース (API) ルーチンにより、ソフトウェアを使用してコンポーネントをコンフィギュレーションできます。次の表は、各関数へのインターフェースとその説明を示しています。続くセクションでは、各関数について詳しく説明します。

デフォルトでは、PSoC Creator はインスタンス名「SPDIF_Tx_1」を設計上のコンポーネントの最初のインスタンスに割り当てます。インスタンス名は、識別子の構文ルールに従った独自の値に変更できます。インスタンス名は、すべてのグローバル関数名、変数名、定数名のプリフィックスになります。読みやすいように、次の表で使用されているインスタンス名は「SPDIF」です。

関数	説明
SPDIF_Start()	S/PDIFインターフェースを起動します。
SPDIF_Stop()	S/PDIFインターフェースをディスエーブルにします。
SPDIF_Sleep()	コンフィギュレーション情報を保存し、S/PDIFインターフェースをディスエーブルにします。
SPDIF_Wakeup()	S/PDIFインターフェースのコンフィギュレーションを復元します。
SPDIF_EnableTx()	S/PDIFビットストリームのオーディオデータ出力をイネーブルにします。
SPDIF_DisableTx()	S/PDIFビットストリームのオーディオ出力をディスエーブルにします。
SPDIF_WriteTxByte()	オーディオFIFOにシングルバイトを書き込みます。
SPDIF_WriteCstByte()	チャネルステータスFIFOにシングルバイトを書き込みます。
SPDIF_SetInterruptMode()	S/PDIFインタラプトに対して、インタラプトソースを設定します。
SPDIF_ReadStatus()	S/PDIFステータスレジスタの状態を戻します。
SPDIF_ClearTxFIFO()	オーディオFIFOをクリアします。
SPDIF_ClearCstFIFO()	チャネルステータスFIFOをクリアします。
SPDIF_SetChannelStatus()	ランタイム時のチャネルステータスの値を設定します。
SPDIF_SetFrequency()	指定された周波数に対するチャネルステータスの値を設定します。
SPDIF_Init()	デフォルトのS/PDIFのコンフィギュレーションに初期化、もしくは復元します。
SPDIF_Enable()	S/PDIFインターフェースをイネーブルにします。
SPDIF_SaveConfig()	S/PDIFインターフェースのコンフィギュレーションを保存します。
SPDIF_RestoreConfig()	S/PDIFインターフェースの構成を復元します。

グローバル変数

変数	説明
SPDIF_initVar	SPDIF_initVarはS/PDIFコンポーネントが初期化されたかどうかを示します。変数は 0 に初期化され、SPDIF_Start()が初めて呼び出されたときに 1 に設定されます。これにより、コンポーネントはSPDIF_Start()ルーチンへの最初の呼び出し後、再初期化なしに再起動できます。 コンポーネントの再初期化が必要な場合、SPDIF_Start()またはSPDIF_Enable()関数の前にSPDIF_Init()関数を呼び出すことができます。

変数	説明
SPDIF_wrkCstStream0[], SPDIF_wrkCstStream1[]	チャネル0およびチャネル1それぞれのチャネルステータス配列 これらの配列は条件付きでコンパイルされ、コンポーネントがチャネルステータスDMAを管理している場合にのみ、コンポーネントに示されます。 チャネルステータスストリームは2つの異なるバッファに保存され、それは完全なSPDIFブロック（2x24バイト）を保存できる大きさです。 注：チャネルステータスの値を設定するには、チャネルステータスマクロと共に、SPDIF_SetChannelStatus()関数を使用することが強く推奨されます。 例えば、チャネル0ステータスのカテゴリフィールドを一般使用に設定するには、次の関数を呼び出します。 SPDIF_SetChannelStatus(SPDIF_CHANNEL_0, SPDIF_CAT_GEN);

void SPDIF_Start(void)

説明: S/PDIFインターフェースを起動します。コンポーネントがチャネルステータスDMAを取り扱うように構成されている場合、チャネルステータスDMAを起動します。必要に応じて、アクティブ モードのパワー テンプレート ビットまたはクロック ゲーティングをイネーブルにします。チャネルステータスと共にS/PDIF出力の作成を起動しますが、オーディオデータはすべて0に設定されます。これによりS/PDIFレシーバは、コンポーネントのクロックにロックすることができます。

パラメータ: なし

返り値: なし

副作用: なし

void SPDIF_Stop(void)

説明: S/PDIFインターフェースをディスエーブルにします。必要に応じて、アクティブ モードのパワー テンプレート ビットまたはクロック ゲーティングをディスエーブルにします。S/PDIF出力は0に設定されています。オーディデータおよびチャネルデータFIFOはクリアされています。SPDIF_Stop()関数はSPDIF_DisableTx()を呼び出し、管理されているチャネルステータスDMAを中止します。

パラメータ: なし

返り値: なし

副作用: なし

void SPDIF_Sleep(void)

説明: これは、コンポーネントのスリープを準備するのに推奨されるルーチンです。SPDIF_Sleep() ルーチンは、現在のコンポーネントの状態を保存します。次にSPDIF_Stop()関数を呼び出し、SPDIF_SaveConfig()を呼び出し、ハードウェア構成を保存します。必要に応じて、アクティブ モードのパワー テンプレート ビットまたはクロック ゲーティングをディスエーブルにします。spdif出力は0に設定されています。

CyPmSleep() またはCyPmHibernate()関数を呼び出す前に、SPDIF_Sleep()関数を呼び出してください。パワーマネージメント関数については、PSoC Creator *System Reference Guide* を参照してください。

パラメータ: なし

返り値: なし

副作用: なし

void SPDIF_Wakeup(void)

説明: SPDIF設定およびノンリテンションレジスタ値を復元します。スリープ前の状態に関わらず、コンポーネントは停止されます。SPDIF_Start()関数は、コンポーネントを起動するために明示的に呼び出す必要があります。

パラメータ: なし

返り値: なし

副作用: 最初にSPDIF_Sleep()またはSPDIF_SaveConfig()関数を呼び出すことなく SPDIF_Wakeup()関数を呼び出すと、予期されない振る舞いにつながる可能性があります。

void SPDIF_EnableTx(void)

説明: S/PDIFビットストリームのオーディオデータ出力をイネーブルにします。送信は次のXまたはZフレームで開始します。

パラメータ: なし

返り値: なし

副作用: なし

void SPDIF_DisableTx(void)

説明: S/PDIFビットストリームのオーディオデータ出力をディスエーブルにします。クロックの次の立ち上がりエッジでデータの送信が中止され、定数0値が送信されます。

パラメータ: なし

返り値: なし

副作用: なし

void SPDIF_WriteTxByte(uint8 wrData, uint8 channelSelect)

説明: オーディオデータFIFOにシングルバイトを書き込みます。コンポーネントのステータスはこの呼び出しの前にチェックし、オーディオデータFIFOがフルでないことを確認します。

パラメータ: uint8 wrData: 転送するオーディオデータを含むバイト。

uint8 channelSelect: チャネルが書き込むための定数を含むバイト。下記のチャネルステータスマクロを参照してください。インターリーブモードでは、このパラメーターは無視されます

返り値: なし

副作用: なし

void SPDIF_WriteCstByte(uint8 wrData, uint8 channelSelect)

説明: 指定されたチャネルステータスFIFOにシングルバイトを書き込みます。コンポーネントのステータスはこの呼び出しの前にチェックし、チャネルステータスFIFOがフルでないことを確認します。

パラメータ: uint8 wrData: 転送するステータスデータを含むバイト。

uint8 channelSelect: チャネルが書き込むための定数を含むバイト。下記のチャネルステータスマクロを参照してください。

返り値: なし

副作用: なし

void SPDIF_SetInterruptMode(uint8 interruptSource)

説明: S/PDIFインタラプトに対して、インタラプトソースを設定します。複数のソースがあった場合、論理和（OR）がとられます。

パラメータ: 選択されたインタラプトソースの定数を格納したuint8バイト。

SPDIF Txインタラプトソース	値
AUDIO_FIFO_UNDERFLOW	0x01
AUDIO_0_FIFO_NOT_FULL	0x02
AUDIO_1_FIFO_NOT_FULL	0x04
CHST_FIFO_UNDERFLOW	0x08
CHST_0_FIFO_NOT_FULL	0x10
CHST_1_FIFO_NOT_FULL	0x20

返り値: なし

副作用: なし

uint8 SPDIF_ReadStatus(void)

説明: SPDIFステータスレジスタの状態を返します。

パラメータ: なし

返り値: SPDIFステータスレジスタのuint8状態。

SPDIF ステータスマスク	値	種類
AUDIO_FIFO_UNDERFLOW	0x01	読み込みの際クリア
AUDIO_0_FIFO_NOT_FULL	0x02	透明
AUDIO_1_FIFO_NOT_FULL	0x04	透明
CHST_FIFO_UNDERFLOW	0x08	読み込みの際クリア
CHST_0_FIFO_NOT_FULL	0x10	透明
CHST_1_FIFO_NOT_FULL	0x20	透明

副作用: ステータスレジスタのうち、Clear on Readタイプのビットはクリアされます。

void SPDIF_ClearTxFIFO(void)

説明: オーディオデータFIFOをクリアします。FIFOにあるデータはすべて失われます。セパレートオーディオモードの場合、両方のオーディオFIFOがクリアされます。転送がディスエーブルされている時だけこの関数を呼び出します。

パラメータ: なし

返り値: なし

副作用: なし

void SPDIF_ClearCstFIFO(void)

説明: チャネルステータスFIFOをクリアします。FIFOにあるデータはすべて失われます。コンポーネントが停止されている時だけ、この関数を呼び出します。

パラメータ: なし

返り値: なし

副作用: なし

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

説明: 動作中にチャネルステータスの値を設定します。このAPIはコンポーネントがDMAを管理しているときのみ有効です。

パラメータ: uint8 channel: 変更するチャネルを指定している定数を含むバイト。下記のチャネルステータスマクロを参照してください。

uint8 byte: 変更するバイト [0..23]。下記のチャネルステータスマクロを参照してください。

uint8 mask: バイト上のマスク。下記のチャネルステータスマクロを参照してください。

uint8 value: 設定する値。下記のチャネルステータスマクロを参照してください。

返り値: なし

副作用: なし

uint8 SPDIF_SetFrequency(uint8 frequency)

説明: 特定の周波数に対するチャネルステータスの値を設定し、1を戻します。この関数はコンポーネントが中止されているときのみ作動します。コンポーネントが起動されたときにこれを呼び出すと、ゼロが戻され、値は変更されません。このAPIはコンポーネントがDMAを管理しているときのみ有効です。

パラメータ: uint8: 指定した周波数の定数を含むバイト。

名前	説明
SPDIF_SPS_22KHZ	クロックレートは22-kHzオーディオに設定されています
SPDIF_SPS_44KHZ	クロックレートは44-kHzオーディオに設定されています
SPDIF_SPS_88KHZ	クロックレートは88-kHzオーディオに設定されています
SPDIF_SPS_24KHZ	クロックレートは24-kHzオーディオに設定されています
SPDIF_SPS_48KHZ	クロックレートは48-kHzオーディオに設定されています
SPDIF_SPS_96KHZ	クロックレートは96-kHzオーディオに設定されています
SPDIF_SPS_32KHZ	クロックレートは32-kHzオーディオに設定されています
SPDIF_SPS_64KHZ	クロックレートは64-kHzオーディオに設定されています
SPDIF_SPS_192KHZ	クロックレートは192-kHzオーディオに設定されています
SPDIF_SPS_UNKNOWN	クロックレートは指定されていません

返り値: uint8: 成功の際は1
失敗の際は0

副作用: なし

void SPDIF_Init(void)

説明: コンポーネントがチャネルステータスDMAを取り扱うよう構成されている場合、コンポーネントとチャネルステータスのインタラプトソースを定義する、カスタマイザーで提供されているデフォルトS/PDIF構成に初期化または復元します。

パラメータ: なし

返り値: なし

副作用: コンポーネントがチャネルステータスDMAを取り扱うよう構成されている場合、インタラプト発生およびチャネルステータス用のマスクレジスタのみを復元します。FIFOからのデータはクリアされず、コンポーネント ハードウェア ステートマシンをリセットしません。

void SPDIF_Enable(void)

説明: ハードウェアの使用を開始し、コンポーネントの動作を開始します。SPDIF_Start()ルーチンがSPDIF_Enable()関数を呼び出すので、この関数を呼び出す必要はありません。これはコンポーネントの動作を開始する際に推奨される方法です。

パラメータ: なし

返り値: なし

副作用: なし

void SPDIF_SaveConfig(void)

説明: この関数は、コンポーネントの設定を保存します。保持されないレジスタも保存します。この関数は、Configure ダイアログで定義されている、または該当する API で変更される、現在のコンポーネントパラメータ値も保存します。この関数は、SPDIF_Sleep()関数に呼び出されます。

パラメータ: なし

返り値: なし

副作用: なし

void SPDIF_RestoreConfig(void)

説明: この関数は、コンポーネントの設定を復元します。維持されないレジスタも復元します。この関数はコンポーネントパラメーター値をSPDIF_Sleep()関数を呼び出す前のものに復元します。このルーチンは SPDIF_Wakeup() に呼ばれ、スリープが終了したときにコンポーネントを復元します。

パラメータ: なし

返り値: なし

副作用: SPDIF_SaveConfig()ルーチンの後においてのみ呼び出します。それ以外の場合、コンポーネントのコンフィギュレーションは初期設定で上書きされます。

チャネルステータスマクロ

チャネルステータスマクロはマスクと特定のチャネルステータス設定の値をカプセル化します。チャネルマクロと共に、これらのマクロはランタイム時にチャネルステータスの値を SPDIF_SetChannelStatus() API で設定するために使用されます。

推奨される最も良い実行方法は、同時に両方のチャネルを設定することです。

下記に例を示します。

SPDIF_SetChannelStatus(SPDIF_CHANNEL_0, SPDIF_DATA_TYPE_LINEAR_PCM);

チャネル名の定数

名前	説明
SPDIF_CHANNEL_0	チャネル0
SPDIF_CHANNEL_1	チャネル1

チャネルステータスの定数

名前	バイト	マスク	値
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

名前	バイト	マスク	値
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
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

名前	バイト	マスク	値
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

ファームウェア・ソースコードのサンプル

PSoC Creator は、Find Example Project ダイアログに数多くのサンプルプロジェクトを提供しており、そこには回路図およびコード例が含まれています。コンポーネント固有の例を見るには、コンポーネントカタログまたは回路図に置いたコンポーネントインスタンスからダイアログを開きます。一般例については、「Start Page (スタートページ)」または **File** メニューからダイアログを開きます。必要に応じてダイアログにある **Filter Options** を使用し、選択できるプロジェクトのリストを絞り込みます。

詳しくは、PSoC Creator ヘルプの「Find Example Project（サンプルプロジェクトを検索）」を参照してください。

機能の説明

このコンポーネントは入ってくるオーディオデータとメタデータをフォーマットし、S/PDIF ビットストリームを作成します。コンポーネントは DMA からオーディオデータを受け取るだけでなく、チャネルステータス情報も受け取ります。多くの場合、チャネルステータス DMA はコンポーネントにより管理されています。しかしながら、システムをよりよくコントロールするために、このデータを別途指定するオプションがあります。

データストリームフォーマット

オーディオおよびチャネルステータスデータは独立したバイトストリームです。バイトストリームは最下位バイトの下位ビットから先にパック化されます。各サンプルで使用されるバイト数は、サンプルを持つための最小バイト数です。未使用のビットは、左端のビットから始めて、0 が付加されます。

オーディオデータストリームはシングルバイトストリームまたは 2 つのバイトストリームの場合があります。シングルバイトストリームの場合、左右のチャネルは最初に左チャネル、そして次に右チャネル用のサンプルが交互配置されています。2 つのストリームの場合、左右のチャネルバイトストリームは別の FIFO を使用します。ステータスバイトストリームは必ず 2 つのバイトストリームです。

DMA

S/PDIF インターフェースは、連続するインターフェースで、中断のないデータストリームを要求します。多くのアプリケーションの場合、DMA 転送の使用を要し、オーディオデータまたはチャネルステータス FIFO のアンダーフローを防ぎます。通常、チャネルステータス DMA は 2 つのチャネルステータス配列を使用してコンポーネント内で完全に行われ、マクロを使用して変更することができます。しかしながら、自前の DMA または CPU を利用してデータを提供し、柔軟性を持つことができます。

S/PDIF はコンポーネントのコンフィギュレーションにより、最高 4 つの DMA コンポーネントを駆動することができます。DMA Wizard は次に示のように DMA 操作を設定するのに使用できます。

DMA ウィザードでの DMA ソース/デスティネーションの名前	方向	DMA 要求信号	DMA 要求タイプ	説明
SPDIF_TX_FIFO_0_PTR	Destination (転送先)	tx_dma0	Level（レベル）	チャネル0またはインターリーブされたオーディオデータ用にFIFOを送信
SPDIF_TX_FIFO_1_PTR	Destination (転送先)	tx_dma1	Level（レベル）	チャネル1オーディオデータ用にFIFOを送信
SPDIF_CST_FIFO_0_PTR	Destination (転送先)	cst_dma0	Level（レベル）	チャネル0チャネルステータスデータ用にFIFOを送信
SPDIF_CST_FIFO_1_PTR	Destination (転送先)	cst_dma1	Level（レベル）	チャネル1チャネルステータスデータ用にFIFOを送信

すべての場合、DMA リクエスト信号の'High'信号は、次のシングルバイトが転送される可能性があることを示します。

Error Handling（エラーの取り扱い）

コンポーネントには 2 つのエラー状態があり、これはオーディオデータが空で、その後読み取りが起きた場合（送信のアンダーフロー）またはチャネルステータス FIFO が空で、その後読み取りが起きた場合（ステータスのアンダーフロー）に生じることがあります。

送信のアンダーフローが生じた場合、コンポーネントは強制的にオーディオデータとして0の固定値送信を行い、すべてのフレームおよび）ステータスデータの適切な生成を継続します。再度送信が始まる前に、送信を必ずディスエーブルにし、FIFO をクリアし、転送用データをバッファし、その後に送信を再度イネーブルする必要があります。このアンダーフロー状態は、コンポーネントステータスビット AUDIO_FIFO_UNDERFLOW を使用して、CPU でモニターすることができます。割込みも、このエラー状態に対して設定することもできます。

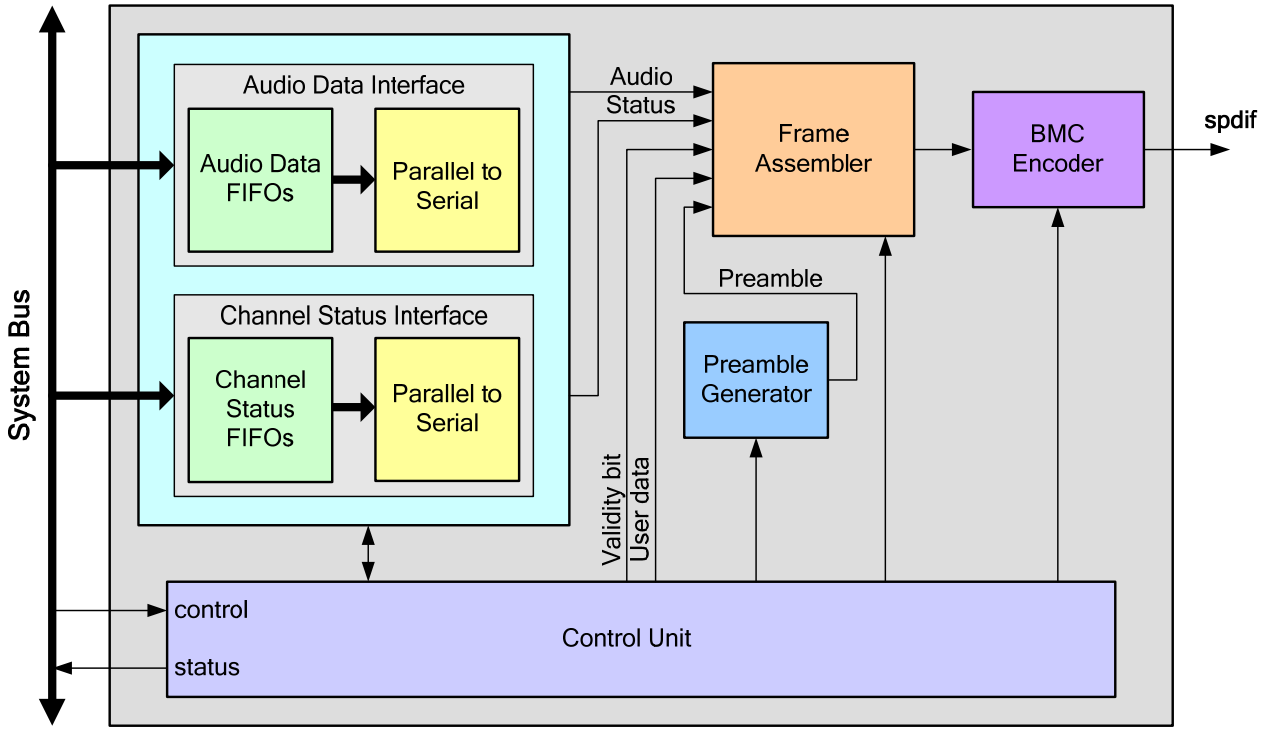
コンポーネントを起動中、ステータスアンダーフローが生じた場合、コンポーネントは X、Y、Z フレームの適切な発生と適切なパリティと共に、チャネルステータスとしてすべて0を送信します。オーディオデータは継続し、影響を受けません。チャネルステータスデータ送信を修復するには、コンポーネントを停止し、再起動する必要があります。このアンダーフロー状態は、ステータスビット CHST_FIFO_UNDERFLOW を使用して、CPU でモニターすることができます。割込みをこのエラー状態に対して設定することもできます。コンポーネントが DMA を管理しない場合、コンポーネントを再起動する前に、ステータスデータをバッファする必要があります。

イネーブル

オーディオデータの送信は、専用のイネーブルがあります。コンポーネントをイネーブルされない状態で起動すると、チャネルステータスと共をに S/PDIF 出力が生成されますしますが、オーディオデータはすべて 0 に設定されます。これにより S/PDIF レシーバは、コンポーネントのクロックにロックオンすることができます。イネーブル状態への移行は X または Z フレームで起きます。

ブロックダイアグラムと設定

SPDIF_Tx は設定された UDB のセットとして実装されます。実装は以下のブロック図のように行われます。



入ってくるオーディオデータは、システムバスインターフェースを使用して受信され、CPU または DMA から提供されます。データは最下位バイトが先になるバイト幅データであり、オーディオバッファに保存されます（FIFO は 1 つまたは 2 つで、コンポーネントのコンフィギュレーションに依存します）。

チャネルステータスストリームはそれ自身の専用のインターフェースがあります。オーディオデータ同様、2つのチャネルステータス FIFO があります。ここでも、チャネルステータスはバイト幅のデータで、最下位バイトが先になります。8 つのサンプルごとに、これらの FIFO から 1 バイトが消費されます。オーディオおよびステータスデータは両方ともパラレルからシリアル形式に変換されます。

ユーザーデータは S/PDIF 規格では定義されておらず、レシーバによっては無視されることがあるため、定数ゼロとして送信されます。

有効ビットは'Low'のときはオーディオサンプルをアナログに変換できることを示します。このビットはゼロの固定値として送信されます。

プリアンブルパターンはプリアンブル発生ブロックで作成され、シリアル形式で送信されます。

これはパリティビットを除いて、SPDIF サブフレーム構造を形成するために必要なデータのすべてです。パリティビットは、すべての入力をサブフレーム構造に組み立てるとき、フレームアセンブラーブロックで計算されます。

フレームアセンブラーブロックの出力は、BMC エンコーダーに送られし、そこでデータは spdif 出力としてエンコードされます。

コントロールユニットブロックはシステムバスインターフェースからコントロールデータを取得し、コンポーネント動作のステータスをバスに戻します。データ送信中、他のすべてのブロックを制御します。

レジスタ

SPDIF_Tx_CONTROL_REG

ビット	7	6	5	4	3	2	1	0
値	予約済み						enable (イネーブル)	txenable

- イネーブル：SPDIF_Tx コンポーネントをイネーブル/ディスエーブルします。イネーブルでない場合、コンポーネントはリセット状態にあります。
- txenable：S/PDIF ビットストリームのオーディオデータ出力をイネーブル/ディスエーブルにします。

SPDIF_Tx_STATUS_REG

ビット	7	6	5	4	3	2	1	0
値	予約済み		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: セットされている場合、チャネルステータス FIFO 1 はフルではありません。
- chst1_fifo_not_full: セットされている場合、チャネルステータス FIFO 0 はフルではありません。
- chst_fifo_underflow: セットされている場合、FIFO アンダーフローイベントが生じたことを示します。
- audio1_fifo_not_full: セットされている場合、オーディオデータ FIFO 1 はフルではありません。
- audio0_fifo_not_full: セットされている場合、オーディオデータ FIFO 0 はフルではありません。
- audio_fifo_underflow: セットされている場合、オーディオデータの FIFO のアンダーフローイベントが生じたことを示します。

レジスタ値は SPDIF_Tx_ReadStatus() API 関数で読み取ることができます。

ステータスレジスタのビット 3 およびビット 0 はスティッキーモード（リード時クリア：clear-on-read）に設定されています。このモードでは、入力ステータスはステータスレジスタクロックの各サイクルごとにサンプルされます。入力が 'High'になると、レジスタビットがセットされ、後続する入力状態に関わらず、セットされたままになります。CPU がその後読み取りを行うと、レジスタビットはクリアされます。

特性

以下の値は、期待される性能を示しており、初期特性データを基にしています。

「公称ルーティングでの最大」タイミング特性

パラメータ	説明	Min	Typ	Max	単位
f_S	サンプリング周波数	–	–	192	kHz
f_{CLOCK}	コンポーネント クロック周波数	–	128 × f_S	35	MHz

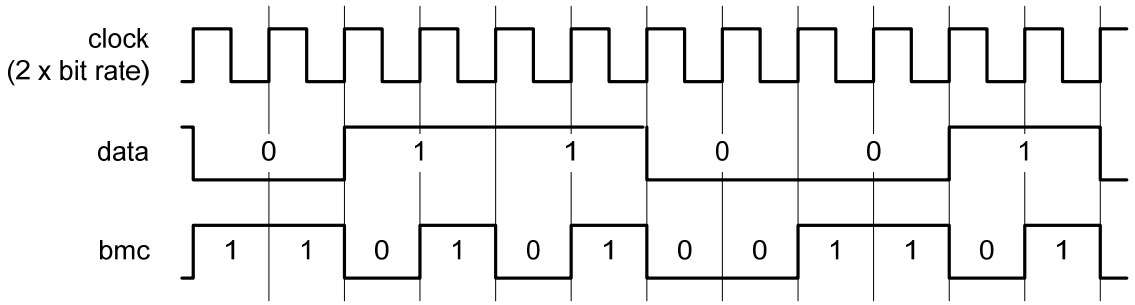
「すべてのルーティングでの最大」タイミング特性

パラメータ	説明	Min	Typ	Max[1]	単位
f_S	サンプリング周波数	–	–	96	kHz
f_{CLOCK}	コンポーネント クロック周波数	–	128 × f_S	17	MHz

[1]「すべてのルーティング」の最大値は、<公称値>/2 で最近似の整数に切り上げ/切り下げられます。この値は、コンポーネントがこのコンポーネント周波数以下で稼働している場合、ミーティングのタイミングについて心配しないためのベースを提供します。

S/PDIFチャネルエンコーディング

S/PDIF はシングルワイヤシリアルインターフェースです。ビットクロックは S/PDIF データストリーム内に組み込まれています。デジタル信号は位相変調の一種である、バイフェーズマークコード (BMC) を使用して暗号化されます。クロックの周波数は、ビットレートの 2 倍です。元のデータのすべてのビットが 2 つの論理ステートとして示され、共にセルを形成します。ビットのスタートの論理レベルは必ず前のビットの終わりのレベルから反転されています。この形式で「1」を送信するには、データビット境界の中央に遷移があります。中央に遷移がない場合、データは「0」と見なされます。



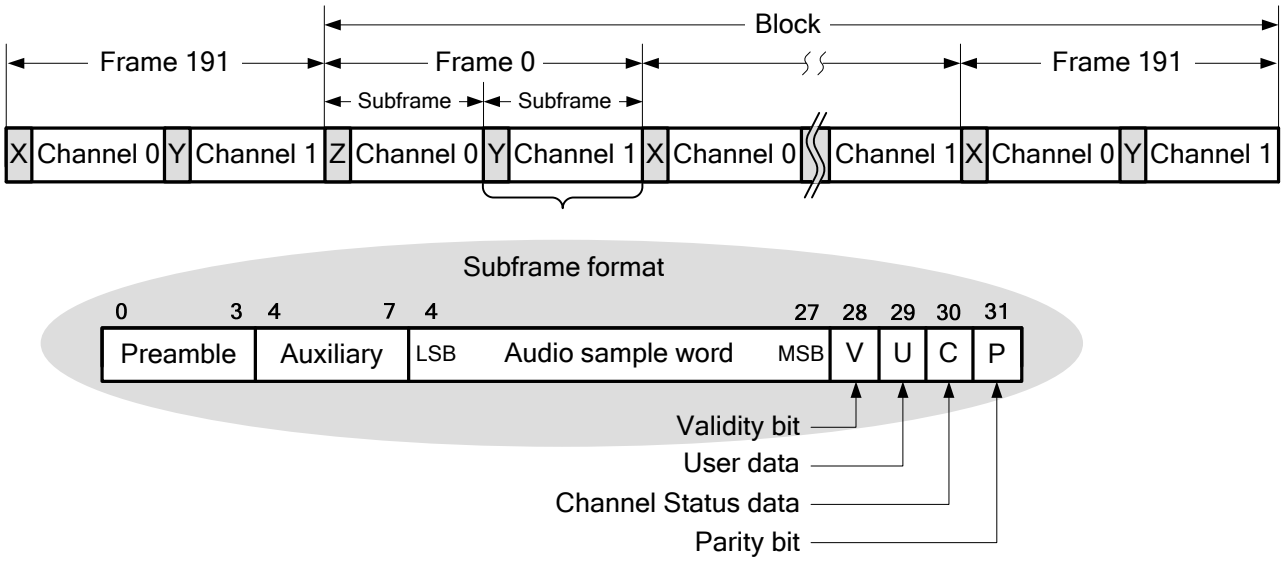
S/PDIFプロトコル階層

SPDIF 信号形式が下記の図に示されています。オーディオデータは連続するブロックで送信されます。ブロックには 192 のフレームがあります。各フレームには 2 つのサブフレームがあり、これはデジタルオーディオデータを構成する基本単位です。

サブフレームに含まれるもの:

- プリアンブルパターン
- 最高 24 ビット幅の単一のオーディオサンプル
- サンプルが有効であるかを示す、バリディティビット
- ユーザーデータを含むビット
- チャネルステータスを含むビット
- このサブフレーム用の偶数パリティビット

プリアンブルには 3 つの種類があります : X、Y および Z。プリアンブル Z はブロックの最初およびサブフレームチャネル 0 の最初を示します。プリアンブル X は、ブロックの最初ではないとき、チャネル 0 サブフレームの最初を示します。プリアンブル Y は必ずチャネル 1 サブフレームの最初を示します。



特性データ用の STA 結果の使用方法

f_{CLOCK} 最大コンポーネント・クロック周波数は、名前の付いた外部クロック (この場合は CLK) として、クロック要約でのタイミング結果で提供されます。内部クロック制限の例を下記に示します。

+Clock Summary

Clock	Actual Freq	Max Freq	Violation
BUS_CLK	24.000 MHz	52.845 MHz	
CLK	6.000 MHz	38.962 MHz	

コンポーネントの変更履歴

バージョン 1.0 は SPDIF_Tx コンポーネントの最初のリリースです。

Copyright © 2005-2012 Cypress Semiconductor Corporation 本文書に記載される情報は、予告なく変更される場合があります。Cypress Semiconductor Corporationは、サイプレス製品に組み込まれた回路以外のいかなる回路を使用することに対しても一切の責任を負いません。特許又はその他の権限下で、ライセンスを譲渡又は暗示することもありません。サイプレス製品は、サイプレスとの書面による合意に基づくものでない限り、医療、生命維持、救命、重要な管理、又は安全の用途のために仕様することを保証するものではなく、また使用することを意図したものでもありません。さらにサイプレスは、誤動作や故障によって使用者に重大な傷害をもたらすことを合理的に予想される、生命維持システムの重要なコンポーネンツとしてサイプレス製品を使用することを許可していません。生命維持システムの用途にサイプレス製品を供することは、製造者がそのような使用におけるあらゆるリスクを負うことを意味し、その結果サイプレスはあらゆる責任を免除されることを意味します。

PSoC Designer™及びProgrammable System-on-Chip™は、Cypress Semiconductor Corp.の商標、PSoC®は同社の登録商標です。本文書で言及するその他全ての商標又は登録商標は各社の所有物です。

全てのソースコード(ソフトウェア及び/又はファームウェア)はCypress Semiconductor Corporation (以下「サイプレス」)が所有し、全世界(米国及びその他の国)の特許権保護、米国の著作権法並びに国際協定の条項により保護され、かつそれらに従います。サイプレスが本書面によるライセンシーに付与するライセンスは、個人的、非独占的かつ譲渡不能のライセンスであって、適用される契約で指定されたサイプレスの集積回路と併用されるライセンシーの製品のみをサポートするカスタムソフトウェア及び/又はカスタムファームウェアを作成する目的に限って、サイプレスのソースコードの派生著作物を複製、使用、変更、そして作成するためのライセンス、並びにサイプレスのソースコード及び派生著作物をコンパイルするためのライセンスです。上記で指定された場合を除き、サイプレスの書面による明示的な許可なくして本ソースコードを複製、変更、変換、コンパイル、又は表示することは全て禁止されます。

免責条項:サイプレスは、明示的又は黙示的を問わず、本資料に関するいかなる種類の保証も行いません。これには、商品性又は特定目的への適合性の黙示的な保証が含まれますが、これに限定されません。サイプレスは、本文書に記載される資料に対して今後予告なく変更を加える権利を留保します。サイプレスは、本文書に記載されるいかなる製品又は回路を適用又は使用したことによって生ずるいかなる責任も負いません。サイプレスは、誤動作や故障によって使用者に重大な傷害をもたらすことが合理的に予想される生命維持システムの重要なコンポーネンツとしてサイプレス製品を使用することを許可していません。生命維持システムの用途にサイプレス製品を供することは、製造者がそのような使用におけるあらゆるリスクを負うことを意味し、その結果サイプレスはあらゆる責任を免除されることを意味します。

ソフトウェアの使用は、適用されるサイプレスソフトウェアライセンス契約によって制限され、かつ制約される場合があります。