

この資料は英語版を翻訳したもので、内容に相違が生じる場合には原文を優先します。こちらの日本語版は参考用としてご利用ください。設計の際には、最新の英語版で内容をご確認ください。

2007 年 12 月 ver 1.0

Application Note 487

## はじめに

このアプリケーション・ノートでは、アルテラ MAX<sup>®</sup> II CPLD を使用してプロトコル変換を実行し、インター IC サウンド (I<sup>2</sup>S) バス上のオーディオ・デバイスへのデータ・フローをシリアル・ペリフェラル・インタフェース (SPI) を通じて制御する方法について説明します。

## I<sup>2</sup>S および SPI インタフェース

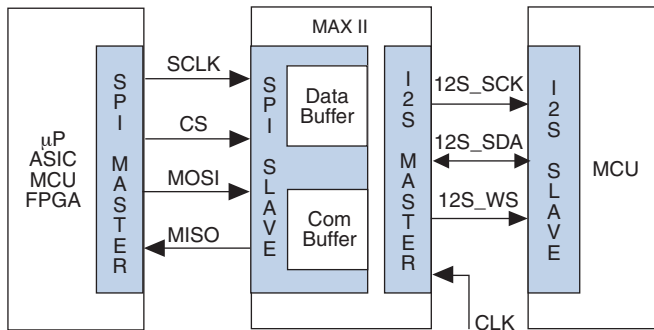
I<sup>2</sup>S は、I<sup>2</sup>S バスとして知られる通常の 3 線式バスを使用して、システムのオーディオ・デバイスを相互接続するために広く使用される、3 線式半二重シリアル・インタフェースです。I<sup>2</sup>S デバイスとバスは、各オーディオ・チャンネルに対応する 2 つの時分割多重オーディオ・データを搬送するシリアル・データ (SDA)、シリアル・クロック (SCK)、および I<sup>2</sup>S バス上の各デバイス間で交換されるデジタル・オーディオ・データを制御するワード・セレクト (WS) の 3 本の線を使用します。I<sup>2</sup>S システムは、ジッタを除去するために、シリアル・クロックからのシリアル・オーディオ・データを個別に処理するように設計されています。

SPI は、システムの内部または外部のペリフェラル・デバイス (スレーブ) にシステム・プロセッサ (マスタ) を接続するときに一般的に使用される、4 線式全二重シリアル・インタフェースです。SPI はクロック (SCLK) とスレーブ・セレクト信号またはチップ・セレクト信号 (CS) と併せて、独立したデータ出力信号 (マスタ・アウト・スレーブ・イン、すなわち MOSI) ラインと、データ入力信号 (マスタ・イン・スレーブ・アウト、すなわち MISO) ラインで通信します。

このデザインでは、SPI インタフェースを備えたホストは、A/D コンバータ、データ信号プロセッサ、デジタル・フィルタ、オーディオ・プロセッサ、PC マルチメディア・オーディオ・コンバータなどのデバイス、および I<sup>2</sup>S バス上のその他のデバイスへのデータ・フローを制御することができます。オーディオ・データ通信用の I<sup>2</sup>S インタフェースを内蔵しないで、SPI インタフェースを内蔵したエンベデッド・システムがいくつかあります。このデザインは、次のような状況で使用すると効果的です。

## MAX II CPLD による SPI-I<sup>2</sup>S 変換

SPI ホストへのブリッジ・インタフェースは、4 線 (CS、SCLK、MISO、MOSI) を持つ SPI スレーブです。I<sup>2</sup>S 側のインタフェースは、3 本の線 (I2S\_SCK、I2S\_SDA、I2S\_WS) を持つ I<sup>2</sup>S マスタです。CLK は I<sup>2</sup>S マスタの外部のマスタ・クロックです。図 1 に、MAX II CPLD を使用した SPI-I<sup>2</sup>S インタフェースの実装を示します。

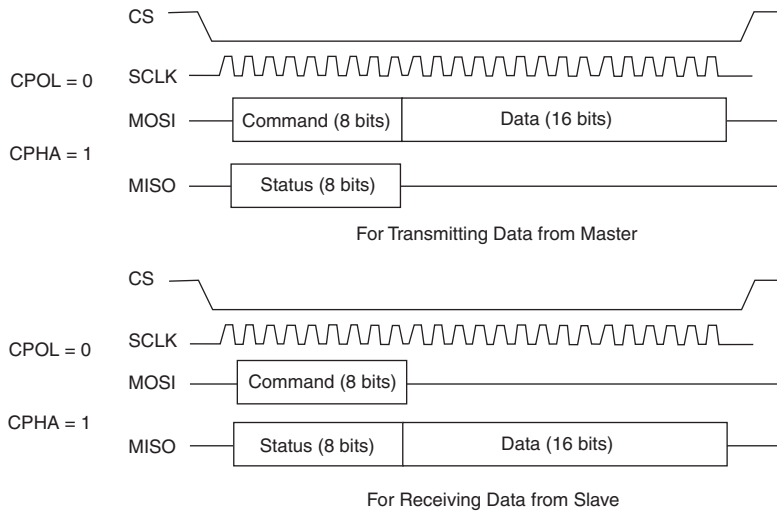
図 1. MAX II CPLD を使用した SPI-I<sup>2</sup>S インタフェースの実装

このデザインは、(システム内のデジタル・オーディオ・デバイスとの相互接続に使用される比較的帯域幅が低いプロトコルの) I<sup>2</sup>S バス上のオーディオ・デバイスに (マイクロプロセッサやホストにシリアル・ペリフェラル・インタフェースで接続するのに用いられる、より一般的な 4 線式のシリアル・インタフェースである) SPI インタフェースを通じてデータ・フロー制御のためのプロトコル変換を提供します。このデザインでは、I<sup>2</sup>S バス上で接続されるデバイスと通信する SPI ホストをブリッジするために、MAX II CPLD を使用しています。図 1 では、MAX II CPLD ブリッジが SPI マスタ (ホスト・プロセッサ) に対する SPI スレーブとして認識され、ホストを I<sup>2</sup>S バスにブリッジする I<sup>2</sup>S マスタの機能を実行します。

送信時には、SPI データには最初の 8 ビットにコマンド・バイトがあり、その後 16 ビット・データが続きます。これらのデータ・ビットは、I<sup>2</sup>S へのデータ送信が要求される場合にのみ有効です。受信時には、ステータス・バイトとして認識される最初の 8 ビットの後に 16 ビット・データ・ワードが続きます。このデータ・ワードの有効性は、ステータス・バイトのデータ有効ビットにより指定されます。

図 2 に、SPI のタイミング図を示します。

図 2. SPI のタイミング図



## SPI インタフェース

通常、SPI バスには 1 つのマスタとそれに接続される多数のスレーブがあります。CPLD ブリッジは、SPI マスタ・デバイスへのスレーブの 1 つとして機能します。図 1 に示す SPI スレーブ・ラインを表 1 で説明します。

ピン名	入力 / 出力	説明
CS	入力 (アクティブ Low)	チップ / スレーブ・セレクト
SCLK	入力	シリアル・クロック
MISO	出力	マスタ・イン・スレーブ・アウト
MOSI	入力	マスタ・アウト・スレーブ・イン

図 2 に示すように、SPI ホストは最初の 8 ビット・コマンド・バイトを送信した後、16 ビット・データを送信します。このデータは送信時にのみ有効です。データが受信される場合、または I<sup>2</sup>S マスタがコントローラ役を担う場合、これらの 16 ビットは無効です。

図 3 に、コマンド・バイト・フォーマットを示します。各コマンド・バイトについては、次の項で説明します。

図 3. コマンド・バイト・フォーマット

Check	RST	-----	Controller	WS	RC	Clock	TR
-------	-----	-------	------------	----	----	-------	----

### Check

最初に送信されるこのビットは、SPI マスタから送信されるワードの有効性を指定します。このビットが 0 の場合、後続の他のビットは無効です。このビットが 1 の場合、後続の他のビットは有効です。このビットは、スLEEPからステータスが読み込まれる場合のみ使用されます。

### RST

I<sup>2</sup>S マスタのリセット信号です。このビットが 1 の場合、I<sup>2</sup>S マスタはリセットされます。このコマンド・ビットが I<sup>2</sup>S マスタにリセットを指示する場合、マスタは 1 クロック・サイクルの間にリセットし、次のクロック・サイクルの間に新しいコマンドに備えます。したがって、リセット・プロセスを完了するには 2 クロック・サイクルが必要です。



マスタの機能の変更が要求されるたびに、コマンドと同時に RST が送信されて、I<sup>2</sup>S マスタがリセットされます。特定の機能が完了した後、最後のコマンドで再び I<sup>2</sup>S マスタがリセットされます。リセット・コマンドが送信される時は常に、コマンド・バイトの残りのビットには必ず前のコマンドが、1 に設定された RST とともに含まれています。これが必要な理由は、データ・トランザクション・サイクル全体が完了する（16 クロック・サイクルの 15 番目）前に、コマンド・バッファからコマンドが読み込まれるためです。

### Controller

I<sup>2</sup>S マスタがコントローラとして機能しているときには、このビットは High です。



I<sup>2</sup>S マスタがコントローラとして機能するには、RC とコントローラ・ビットの両方が High でなければなりません。I<sup>2</sup>S マスタがレシーバとして機能するには、RC ビットのみ High です。

## WS

このビットは、受信または送信されるデータの各ワードのワード・チャンネルを指定します。

## RC

I<sup>2</sup>S マスタがレシーバとして機能している場合、このビットは High です。

## Clock

I<sup>2</sup>S マスタのクロック・イネーブルです。

## TR

I<sup>2</sup>S マスタがトランスミッタとして機能している場合、このビットは High です。

## I<sup>2</sup>S ステータス・バイト・フォーマット

CS が Low になるたびに、SPI マスタによって 24 ビットが送信または受信されます。これらの 24 ビットの最初の 8 ビットはステータスを表し、次の 16 ビットは受信されるデータに一致します。

図 4 に、8 ビット・ステータス・フォーマットを示します。各コマンド・バイトについては、次の項で説明します。

図 4. I<sup>2</sup>S ステータス・バイト・フォーマット

Current WS	TR/RC	WS/RC	Data Valid	Data Buff	Data Buff	Command Buff	Command Buff
------------	-------	-------	------------	-----------	-----------	--------------	--------------

### Current WS

このビットはマスタで現在選択されているチャンネルを指定します。

### TR/RC

このビットは、I<sup>2</sup>S マスタのステータスを指定します。このビットが 1 に設定されている場合、I<sup>2</sup>S マスタは現在トランスミッタとして機能しています。このビットが 0 に設定されている場合、レシーバとして機能しています。

### WS/RC

このビットは、データが受信されるチャンネルを指定します (I<sup>2</sup>S マスタがレシーバの場合)。

### Data Valid

このビットは 16 ビット・データの有効性を指定します。I<sup>2</sup>S マスタがレシーバで、このビットが 1 に設定されている場合、受信される 16 ビットは有効です。I<sup>2</sup>S マスタがレシーバで、このビットが 0 に設定されている場合、受信される 16 ビットは無効です。

### Data Buff

これらの 2 ビットは、データ・バッファの現在のステータスを指定します（送信されるデータをバッファする各 2 バイトのバッファが 3 個あります）。これらのバッファで可能な 4 つの状態を以下に示します。

- 00: すべてのバッファがエンプティ
- 01: バッファ 1 と 2 がエンプティ、バッファ 3 がフル
- 10: バッファ 1 がエンプティ、バッファ 2 と 3 がフル
- 11: 全バッファがフル

### Command Buff

これらの 2 ビットは、コマンド・バッファの現在のステータスを上記のデータ・バッファに類似したフォーマットで指定します。



コマンド・バッファのステータスは、SDA ライン上のデータ・トランザクションの 15 番目のクロック・サイクルで変化し、データ・バッファのステータスは 16 番目で変化します。

CPLD ブリッジがコマンド・バイトを受信すると、最上位ビット（MSB）が 1 かどうかチェックされます。このコマンド・バイトは、MSB が 1 の場合にのみコマンド・バッファに送信されます。次にこれらの 8 ビットに続く 16 ビット・データは、データ・バッファに格納されます。コマンド・バッファとデータ・バッファは、いずれも最大 3 セットのコマンド・バイトと最大 3 セットのデータ・ワードを格納できます。ホストは毎回ステータスをチェックし、バッファのステータスを認識し、新しいコマンド・バイトおよびデータ・ワードを送信するかどうかを決定します。

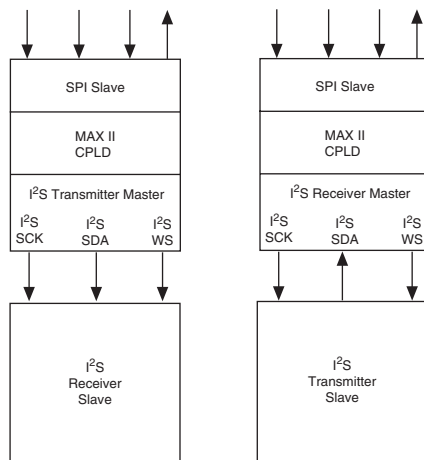
I<sup>2</sup>S マスタのデータ受信トランザクションが完了すると、ステータス・レジスタのデータ有効ビットが High に設定され、I<sup>2</sup>S データの SPI スレーブへの転送が開始されます。このデータは MISO ピンを通じて SPI マスタに転送できます。この MISO トランザクションが完了すると、データ有効ビットは 0 に設定されます。

## I<sup>2</sup>S インタフェース

MAX II CPLD の I<sup>2</sup>S 部は、I<sup>2</sup>S バス上の I<sup>2</sup>S スレーブに接続された I<sup>2</sup>S マスタとして機能します。表 2 に、SPI-I<sup>2</sup>S ブリッジの I<sup>2</sup>S ラインを示します。

ピン名	入力 / 出力	説明
I2S_SCK	出力	I <sup>2</sup> S シリアル・クロック
I2S_SDA	双方向	I <sup>2</sup> S シリアル・データ
I2S_WS	出力	I <sup>2</sup> S ワード・セレクト
CLK	入力	入力マスタ・クロック入力

I<sup>2</sup>S マスタはまず、コマンド・バッファおよびデータ・バッファから SPI スレーブが設定したデータを読み出します。対応するコマンド・バイトがデータ送信を指定している場合は、対応するデータ・バッファから読み出されたデータが I2S\_SDA ライン上をシリアルに送信されます。15 番目のデータ・ビットが送信された後、I<sup>2</sup>S マスタはコマンド・バッファから新しいコマンド・バイトを読み出し、この新しいコマンド・バイトに指定される次の処理に備えます。これが特に重要になるのは、次のデータ・トランザクションの前に 1 クロック・サイクルだけ変化し、新しいデータを別のチャンネルに割り当てるための WS ビットです。ここで I<sup>2</sup>S マスタは、図 5 に示すように、I<sup>2</sup>S スレーブ・レシーバへのトランスミッタとして機能しています。

図 5. I<sup>2</sup>S マスタの予測されるシナリオ

I<sup>2</sup>S マスタによりコマンド・バッファから読み出されたコマンド・バイトがデータ受信を指定する場合、データは I<sup>2</sup>S-SDA ラインから受信されます。このラインで 16 ビットのデータが受信された後、データは SPI スレーブに送信され、同時にステータス・レジスタのデータ有効ビットに 1 が設定されます。これにより、SPI スレーブは MISO ライン上で SPI ホストにデータを送信できます。ステータス・バイトの WS ビットも更新され、受信されたデータのチャンネルを指定します。

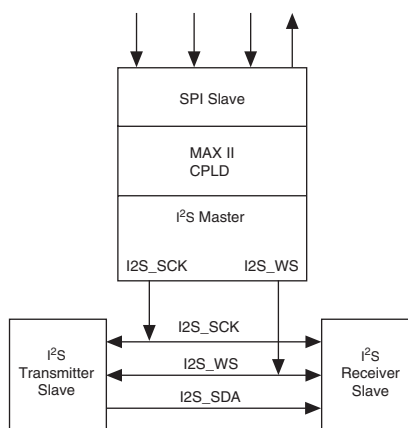
## コントローラとしての I<sup>2</sup>S マスタ

SPI マスタが I<sup>2</sup>S マスタに I<sup>2</sup>S コントローラとして機能する場合、コマンド・バイト内の対応するビット (controller, RC, clock) は **High** に設定されます。次に、I<sup>2</sup>S マスタは I<sup>2</sup>S\_SCK 信号と I<sup>2</sup>S\_ws 信号をイネーブルにし、I<sup>2</sup>S\_SDA ラインをトライステートにします。



図 6 に、コントローラ・モードの I<sup>2</sup>S マスタを示します。

図 6. コントローラ・モードの I<sup>2</sup>S マスタ



## 実装

このデザインは、EPM570 または EPM1270 を使用して実装することができます。デザインのソース・コードはコンパイルして、MAX II CPLD Map にプログラムすることができます。SPI インタフェース・ポートと、適切な汎用 I/O (GPIO) への I<sup>2</sup>S バス・ラインを図 1 に示します。

## ソース・コード

このデザイン例は Verilog を使用して作成されています。ソース・コード、テストベンチ、および完成した Quartus® II プロジェクトは、以下から入手可能です。

[www.altera.co.jp/literature/an/an487\\_design\\_example.zip](http://www.altera.co.jp/literature/an/an487_design_example.zip)

## まとめ

このデザインで示すとおり、MAX II CPLD は、SPI-I<sup>2</sup>S などのインタフェース・プロトコル・コンバータを実装するための有力な選択肢となります。低コスト、低消費電力、および使いやすいパワー・オン機能により、MAX II CPLD はインタフェース・プロトコル・コンバータ・アプリケーションに最適なプログラマブル・ロジック・デバイスになります。

## 関連情報

以下に、このアプリケーション・ノートに関連情報を示します。

- MAX II CPLD ホームページ：  
[www.altera.co.jp/products/devices/cpld/max2/mx2-index.jsp](http://www.altera.co.jp/products/devices/cpld/max2/mx2-index.jsp)
- MAXII デバイスの資料ページ：  
[www.altera.co.jp/literature/lit-max2.jsp](http://www.altera.co.jp/literature/lit-max2.jsp)

- MAX II パワーダウン・デザイン :  
[www.altera.co.jp/support/examples/max/exm-power-down.html](http://www.altera.co.jp/support/examples/max/exm-power-down.html)
- MAX II アプリケーション・ノート :  
「AN 428: MAX II CPLD のデザイン・ガイドライン」  
「AN 422: MAX II CPLD を使用したポータブル・システムにおける消費電力の管理」

## 改訂履歴

表 3 に、このアプリケーション・ノートの改訂履歴を示します。

日付 & ドキュメント・バージョン	変更内容	概要
2007 年 12 月 v1.0	初版	—



101 Innovation Drive  
San Jose, CA 95134  
[www.altera.com](http://www.altera.com)  
Literature Services:  
[literature@altera.com](mailto:literature@altera.com)

Copyright © 2007 Altera Corporation. All rights reserved. Altera, The Programmable Solutions Company, the stylized Altera logo, specific device designations, and all other words and logos that are identified as trademarks and/or service marks are, unless noted otherwise, the trademarks and service marks of Altera Corporation in the U.S. and other countries. All other product or service names are the property of their respective holders. Altera products are protected under numerous U.S. and foreign patents and pending applications, maskwork rights, and copyrights. Altera warrants performance of its semiconductor products to current specifications in accordance with Altera's standard warranty, but reserves the right to make changes to any products and services at any time without notice. Altera assumes no responsibility or liability arising out of the application or use of any information, product, or service described herein except as expressly agreed to in writing by Altera Corporation. Altera customers are advised to obtain the latest version of device specifications before relying on any published information and before placing orders for products or services.

