

組込みシステムにおける シリアル通信の検証手法

はじめに

一般的な組込みシステムは、マイクロコントローラと、外部機器とのインタフェースのためのペリフェラル・デバイスで構成されています。さまざまな設計が行われており、マイクロコントローラとペリフェラル・チップ間のやりとりはシリアル通信リンクで行われています。

シリアル・バスは、パラレル・バスに比べてPCBで占有する面積が少なく、プロトコルはしっかりとした規格で定義されており、優れたインターオペラビリティ（相互運用性）を実現しています。シリアル・リンクを使用するペリフェラル・チップには、センサ/シグナル・コンディショナ、A/DコンバータやD/Aコンバータなどの変換デバイス、通信アダプタ、外部メモリ、リアルタイム・クロック、ディスプレイ・サブシステムなどがあります。

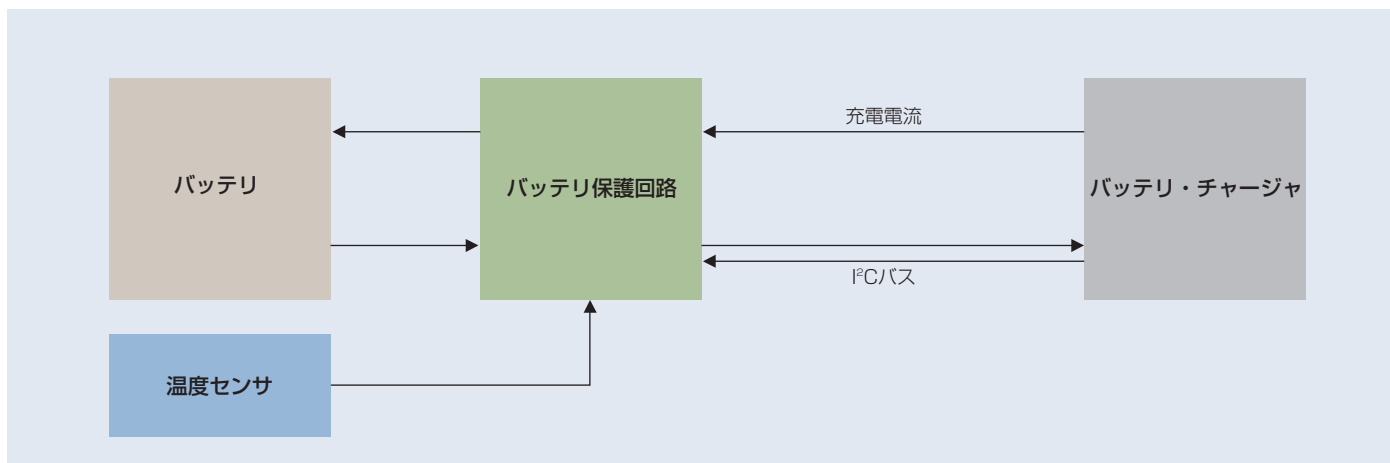


図1.1. バッテリー・チャージャ・システムのブロック図

ファームウェアの作業に取り掛かる前に、ペリフェラル・チップが正しく動作していることを確認しなければなりません。これは、ペリフェラル・チップの入出力と、チップとマイクロコントローラ間で伝送されるシリアル・バス・データを同時観測することで確認できます。しかし、従来のテスト・ソリューションでは、シリアル・バスが正しく機能し、正しいデータを送っていることを検証するのは難しいことでした。このアプリケーション・ノートでは、3種類のアプリケーションを紹介し、ペリフェラル・チップの信号とシリアル・バス・データを同時観測するための測定テクニックを説明します。

アプリケーション1：シリアル通信による バッテリー・チャージャの制御

リチウム・イオン・バッテリーの充電は非常に重要です。過充電はバッテリーにダメージを与え、発火の危険性をもたらします。リチウム・イオン・バッテリーのバッテリー・パックにはバッテリー保護回路が組込まれており、セルの電圧、温度、電流、充電状態を監視しています。バッテリー保護回路の中には、バッテリー・チャージャや機器本体と通信して、これらの情報にアクセスできるものがあります。バッテリー・チャージャがバッテリーの温度や個々のセルの電圧を知ることができれば、過充電やその他の危険な状態を回避することができます。機器本体がこの情報を利用することで、バッテリー残量が負荷に対応できなくなりつつあることをユーザに知らせることができます。最も一般的な通信方法が、I²Cプロトコルを

使用したSMBusと呼ばれるものです。この例では、バッテリー・チャージャはI²Cバスを使ってバッテリー温度（同時にセル電圧も）を定期的に問い合わせます。さらに、バッテリー保護回路の応答と、アナログ温度入力に対するチャージャの応答を観測します。図1.1に、バッテリー、バッテリー保護回路、バッテリー・チャージャの構成を示します。このバッテリー・チャージャは特に充電電流を制御し、バッテリー温度がスレッシュホールドを超えないようにして、バッテリーの劣化を防ぐよう設計されています。

この例で難しいのは、バッテリー温度の変化に対するバッテリー・チャージャの応答を測定することです。MSO4000シリーズ・ミックスド・シグナル・オシロスコープを使用することで、以下の項目を測定することができます。

- 温度の変化からI²Cバス経由でデータが利用できるようになるまでの時間
- データが通信されてから電流が制御されるまでの時間
- 電流制御の時間と安定度
- 制御前後の実際の電流値

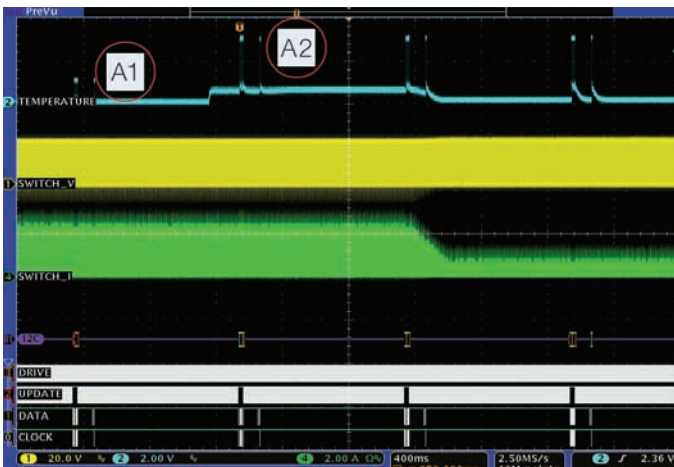


図1.2. 4秒間の充電動作

図1.2はバッテリー・チャージャの4秒間の動作を示しており、温度変化に対する応答がわかります。

- TEMPERATUREとある一番上の波形は、バッテリーのアナログ温度の読み値を示しています。
- その下のSWITCH_VとSWITCH_Iの2つのアナログ波形は、それぞれ充電制御回路の電圧とパワー・トランジスタの電流であり、充電電流はパルス幅変調で制御されていることを示しています。
- デコードされたI²C通信バスは、DATA、CLOCKの2つのデジタル・ラインで画面下に表示されています。I²CバスからのデコードされたデータはI²Cと書かれた波形で電流波形の下に表示されており、通信ストリームのデバッグが簡単になります。
- パワー・トランジスタのゲートへのデジタル信号はDRIVEの波形で取込まれ、電力制御を確認することができます。
- さらに、UPDATEはマイクロコントローラがコントロールする信号で、バッテリー・チャージャが電流制御補正を行うための制御ラインです。これにより、システム応答を測定することができます。

このように、出力信号（温度）、通信（I²C）、バッテリー・チャージャのスイッチング電源の影響（電圧または電流）を同時にモニタします。また、制御ループ（UPDATE）をモニタするテクニックを使っています。

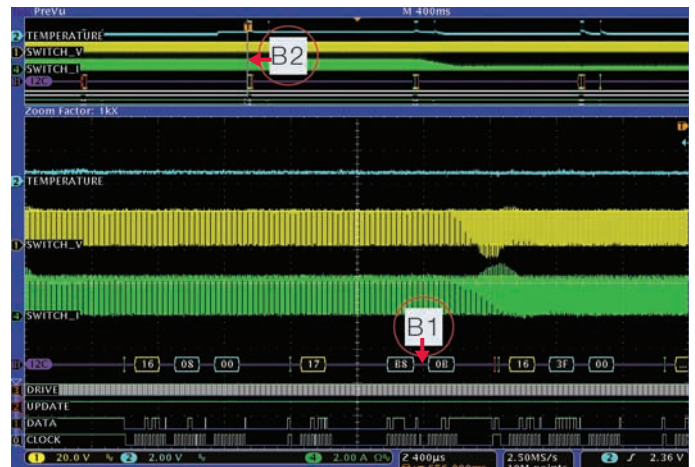


図1.3. 取込んだ波形を拡大表示することでデコードされた温度が読み取り、I²Cバスとデコードされたデータを詳細に観測することができます。

まず図1.2の波形から、区間A1と区間A2の間で変化した温度がどの時点で読み取られたかを知る必要があります。温度は、約1秒ごとにバッテリー保護回路により、TEMPERATURE上に散在するごく短いパルスで読み取れます。A1は温度が高いことを示しています（サーミスタを使用しているため、高い温度では抵抗は小さくなり、電圧は小さくなります）。A2は、温度が低いことを示しています。I²Cバスは、TEMPERATURE波形の上に散在するパルスのうちより長いパルスのときに読み取ることができ、それは同期してクロックとデータのデジタル波形が変化することからもわかります。

図1.3は、取込んだ波形をズームし、デコードされた温度を表示しており、I²Cバスとデコードされたデータを詳細に示しています。リード（0x17）コマンドの後のB1における2バイトは、0x0BB8（下位ビットが先）と表示されています。これがA1における温度の値です。バス・デコードの位置は画面上部のB2に示されており、常に波形全体の時間関係を把握することができます。この時点のI²Cの値は、バッテリー保護回路がA2における新しい温度を測定する前のものであるため、温度が上がったのか下がったのかわかりません。

図1.4は、区間A2において、バッテリー保護回路によってバッテリーの温度変化が測定された後のI²Cバスとデコードを示しています。(0x17) コマンドの後のC1における温度は0x08D4と変化して、温度が下がったことがわかります。これは、区間A2においてバッテリー保護回路によって温度が読み取られた後の最初のI²Cデータであり、結果は許容範囲です。波形全体の中での時間フレームは、画面上部に示されています。このズーム波形は、全体の波形のC2の部分になります。

図1.5は、図1.4でI²Cの値が読み取られた前後の期間(0.4秒)を表示したもので、徐々に電流制御されていることがわかります。電流制御プロセスは、UPDATEと書かれた波形で見とれます。さらに低い温度にするため、電流を小さくしなければなりませんので、UPDATEパルスごとに電流は徐々に補正されています。大きな電流から小さな電流への穏やかな制御と、直前のI²C通信が行われてすぐ、非常に短い時間経過後(約10ms)、電流が減少しだすことが観測できます。バッテリー・チャージによって新しい温度が読み取られた後、約200msで電流制御が完了しています。変化した電流の値も見てとれ、約2Aから約0.25Aへと減少しています。

図1.6は、このオシロスコープのロング・メモリの利点を示しています。スイッチング電源の充電制御における波形が拡大され、電圧波形と電流波形が詳細に表示されています。パワー・トランジスタの電流は、電流プローブで測定されています。ドライブ信号も電圧電流波形と同じ画面上に表示されており、ドライブの遅延やスイッチングの回数を測定することもできます。この電流波形では各充電電流のリップルも見ることができ、大きな電流から小さな電流へのトランジションにおいて電源のインダクタによるリップルが十分に抑えられていることがわかります。

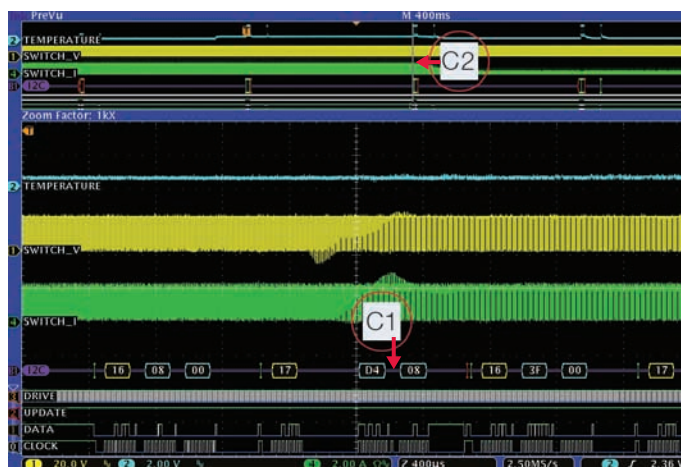


図1.4. バッテリー保護回路によって温度変化が測定された後の次のサンプルにおけるI²Cバスとデコードを示しています。

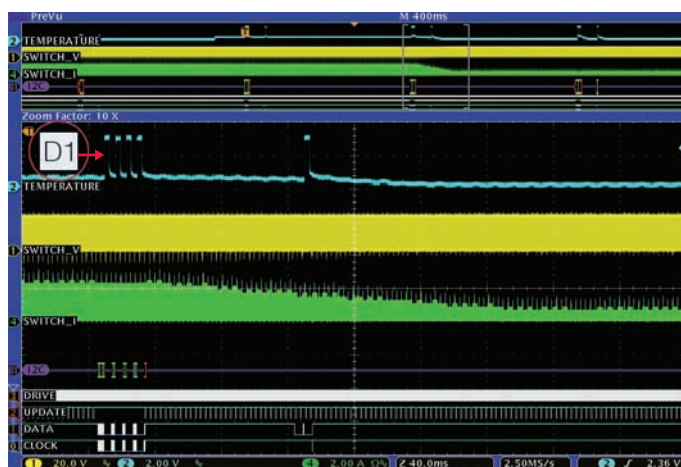


図1.5. I²Cの読み取り直後をズーム表示しており、徐々に電流制御されていることがわかります。



図1.6. ロング・メモリにより、スイッチング電源の充電制御における波形が拡大され、電圧波形と電流波形が詳細に表示されています。

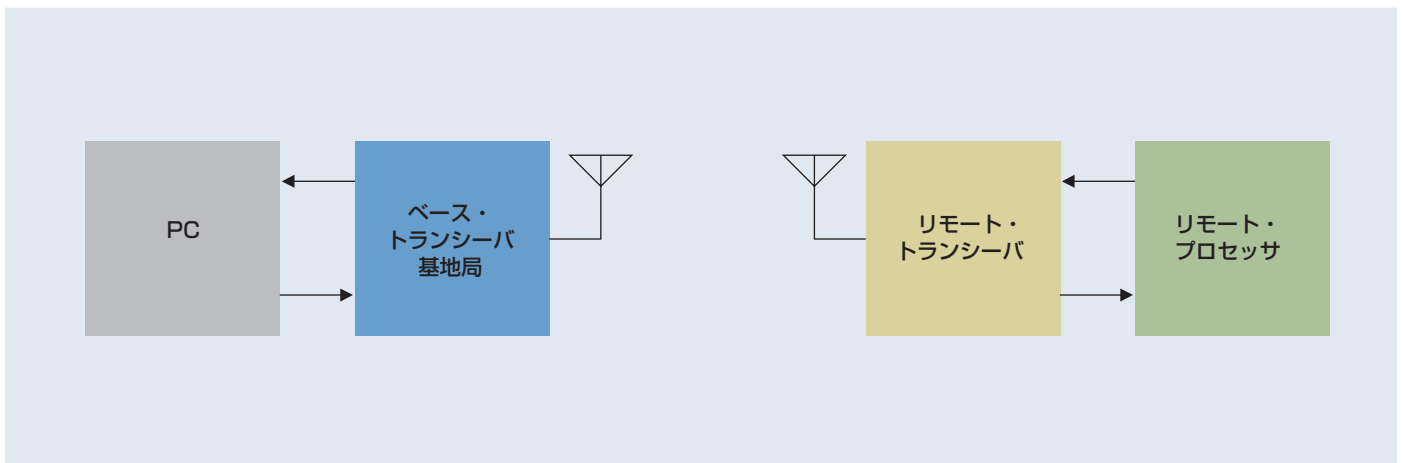


図2.1. 双方向無線システムのブロック図

この例では、バッテリー保護回路が温度変化を正しく取込み、バッテリー・チャージャはI²C経由でこの温度を正しく読み取り、徐々に充電電流を制御することによりこの温度を低下させる様子を観測しました。制御システムのイベントの発生する時間と電流応答は1つの波形となって表示されるため、エンジニアは必要に応じてすべての情報を検証することができます。

アプリケーション2：近距離のデータ無線通信

この例では、正しいパケット伝送と、双方向無線通信システムの各ステージのタイミングを確認します。図2.1は、RS-232シリアル通信によるPCから基地局トランシーバへの通信を示しています。次に、基地局は独自の無線プロトコルでリモート・トランシーバと通信します。ここでは、RS-232デコード機能により、PCから基地局トランシーバ、および基地局トランシーバからPCに送られるメッセージを観測します。基地局トランシーバとリモート・トラン

シーバにおける双方向無線信号は、アナログ・チャンネルとデジタル・チャンネルを使用して観測します。この例で使用する無線方式は、リモート・トランシーバで回路が簡単になるAM方式を使用しています。

パケットは、次に示す8バイトで構成されます。

- ヘッド・バイト = 0x21 (コマンド) (ロング・データ・パケットでは0x28を使用しますが、この例では説明していません。)
- 2つのシリアル番号バイト = 0x0000 (基地局用、リモートはその他の値)
- コマンド・バイト = 0x30~0x3F
- 2つの引数またはデータ・バイト = 0x0000~0xFFFF (下位ビットが先)
- 2つのチェックサム・バイト = 最初の6バイトの合計 (下位ビットが先)

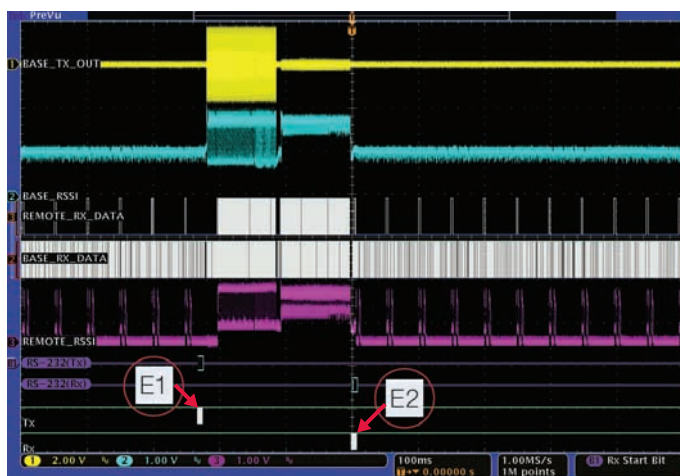


図2.2. PCから出てPCに戻るまでのすべての通信サイクル

図2.2は、基地局に送られたRS-232信号 (E1) からリターン・メッセージ (E2) までのすべてのトランザクションを示しています。RS-232信号は、TxとRxの2つのデジタル・ラインで取込まれています。これに対応するデコード情報は、RS-232(Tx)、RS-232(Rx)のラインで表示されています。実際にデコードされた信号は、後ほど説明します。BASE_TX_OUTの黄色の波形は、アナログの送信信号です。BASE-RSSIの青色の波形は、基地局で受信したアナログ信号です。この信号のデジタル・バージョンは、BASE_RX_DATAのデジタル信号で取込まれています。リモート・サイドでは、REMOTE_RSSIの紫色の信号が受信した信号のアナログ・バージョンであり、REMOTE_RX_DATAの信号がそのデジタル・バージョンです。この画面からもわかるように、すべてのコマンド・パケットとリターン・データの全トランザクションで約230msのやりとりになっています。この取込みでは、すべてのトランザクションが取込めるように、戻りのRS-232 (Rx) 信号のスタート・ビットでトリガしています。



図2.3. PCからの初期RS-232のデコード

図2.3は、PCから基地局に送信されたTxデータをデコードしたものをズーム表示しています。このデコード表示では、パケットが正しく送られていることを容易に確認することができます。上の波形の [] で囲まれた部分が、下の波形でズーム表示されています。このデコード・データから、パケットは正しく8ビットになっていることが確認できます。最初のバイトはコマンド0x21に対応しています。これは基地局であるため、アドレスは0x0000となります。コマンド0x3Aは、バッテリー電圧のリクエストです。このリクエストにおいては引数はなく、常にゼロになります。チェックサム0x5Bはそれまでのバイトの正しい合計値となります。

図2.4は、ほとんどの起動プリアンブルが送られた後の基地局トランシーバの送信波形を示しています。データは、キャリア信号の短い周期、長い周期で構成されています。リモート・トランシーバで受信したアナログ信号 (REMOTE_RSSI) とそのデジタル・バージョン (REMOTE_RX_DATA) は、期待通り全く同じパターンになっています。



図2.4. 基地局トランシーバの送信波形とリモート・トランシーバの受信信号

図2.5は、リモートの応答を示しています。メッセージの受信から応答の送信までは、REMOTE_RSSI信号上で観測でき、わずか10ms程度です。基地局のトランシーバで受け取ったリターン・メッセージは、基地局トランシーバ・アナログ信号 (BASE_RSSI) とデジタル・バージョン (BASE_RX_DATA) で表示されています。PCにリターンされるRS-232 (Rx) 信号は、リモートからのメッセージの受信 (BASE_RSSI) 後からわずか5ms程度の遅れとして観測されています。

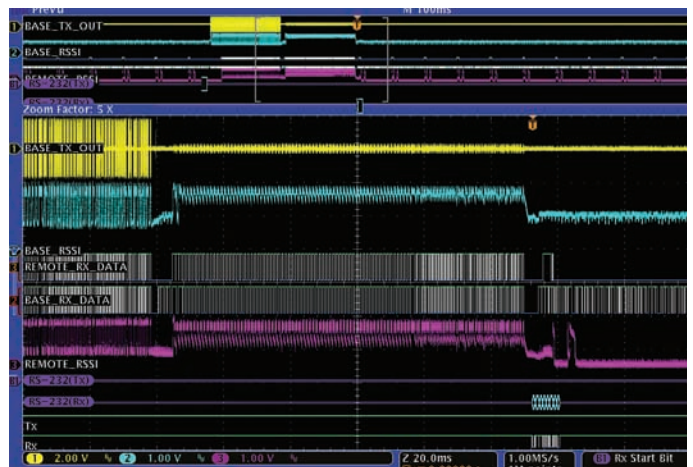


図2.5. 送信信号に応答した受信信号

図2.6はデコードされたRS-232メッセージ (Rx) が表示されており、リターン・データが期待通りのものであることを確認することができます。ここでも、このデコード・データから、パケットは正しく8ビットになっていることが確認できます。最初のバイトはここでもコマンド・ヘッダの0x21です。アドレスは0xFFFFであり、正しい値です。コマンド・バイトの0x3Aはエコーされているため、応答は送信されたコマンドによるものであり、読み違ったコマンドではないことを確認することができます。リターンのデータ値は0x21Fで、これは2.5Vに相当し、加えられた値そのものです。チェックサムは最初の6バイトの正しい合計値になっています。 $0x21 + 0xFF + 0xFF + 0x3A + 0x2F + 0x01 = 0x0289$

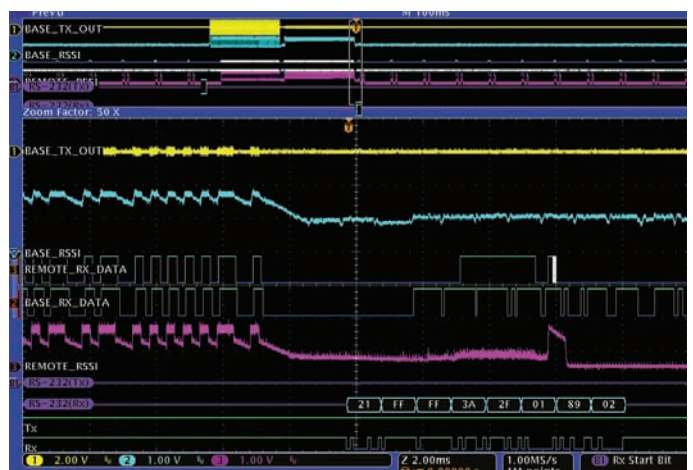


図2.6. PCに戻ったリターン・メッセージのデコード

オシロスコブのRS-232データ・デコード機能を使用し、この無線システムの送受信通信が正しく行われていることを確認しました。パケットが正しく構成されており、ヘッダ、コマンドの値、チェックサムが正しいことを確認しました。送受信のアナログ信号とデジタル情報を一つの波形として取込み、ズーム表示することでアナログ信号の品質と受信した無線データを詳細に観測することができました。また、トランシーバの応答時間が期待する範囲に入っており、双方向のトランザクション時間が期待通りに約230msであることを確認しました。

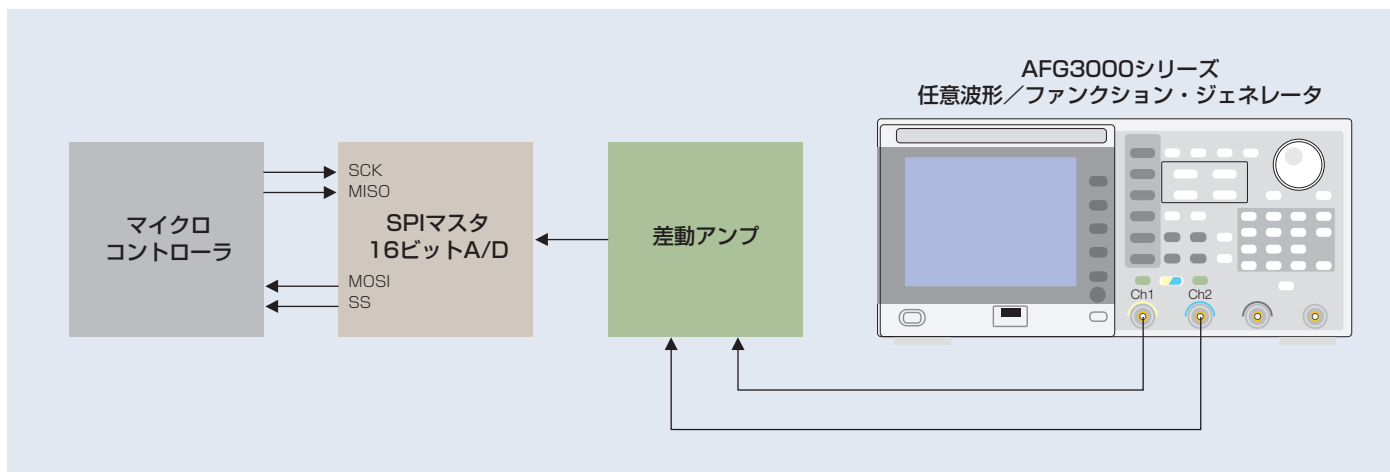


図3.1. データ・アキュイジション・サブシステムのブロック図

アプリケーション3：SPIベースのA/Dコンバータの評価

この例では、SPIタイプのシリアル・バスで通信する、SPIベースの16ビットA/Dコンバータの性能を調べます。AFG3000シリーズなどの2チャンネル出力のシグナル・ジェネレータの2つのチャンネルを使用して真の差動入力波形を生成し、入力信号からA/D変換したデータをMSO4000シリーズ・オシロスコープのSPIデコード機能を利用して、観測します。データ・アキュイジションのハードウェアは図3.1に示すように、差動入力アンプ、A/Dコンバータ、マイクロコントローラで構成されています。

この測定の目的は、A/DコンバータがSPIマスタとして正しく機能していること、またすべての入力電圧範囲を正しく読み取っていることを確認することです。コンバータへのフルスケール入力の生成には、任意波形ジェネレータを使用します。オシロスコープのロング・メモリにより、図3.2に示すように、複数個のSPIサンプルを1つのアキュイジション波形として取込むことができます。オシロス

コープのSPIデコード機能により、A/Dで変換した値を読み取ります。次に、このアキュイジション波形の異なる点における、アナログ波形やA/Dコンバータによって生成され、デコードされたデジタル値をズームして表示します。図3.2において、IN+、IN-の黄色と青色の波形は、シグナル・ジェネレータからの電圧です。INPUT_DIFFの赤色の波形は、2つの入力信号の引き算波形であり、差動アンプへのトータルの入力を表しています。A/D_INの紫色の波形は、差動アンプ後のA/Dコンバータへの実際の入力です。これらのアナログ波形から、A/D前のアナログ回路が正しく動作していることを確認できます。CLOCK、DATA、COVERTのデジタル信号は、SPI制御信号、データ信号です。さらに、SPI (MOSI)とSPI (MISO)の波形は、データのデコードを示しています。

図3.3では、入力電圧波形振幅の一番下の部分をズーム表示しています。ズーム表示している波形の範囲は、画面の一番上で確認できます。データはデコードされてSPI (MOSI) のラインで確認できます。出力は0x00D8で、期待通りの非常に小さい値になっています。

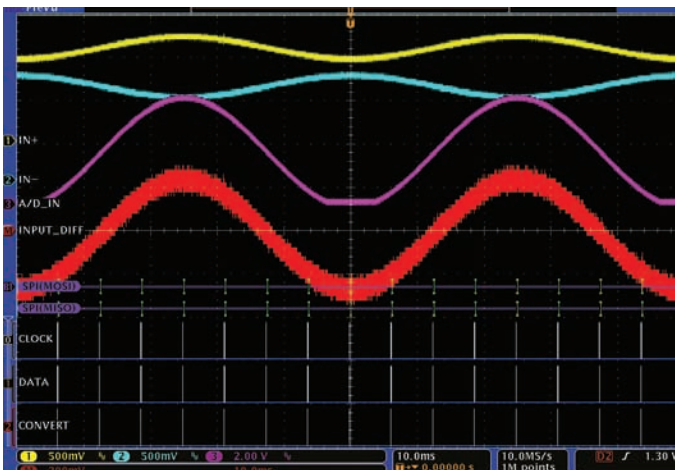


図3.2. サイン波形上でサンプルされた複数個のSPI信号

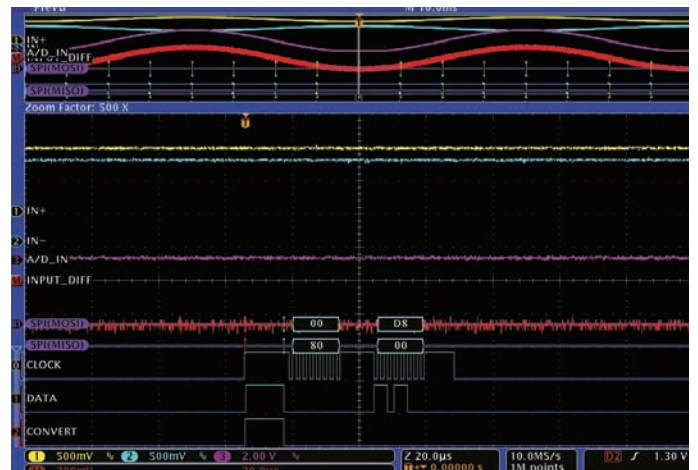


図3.3. 波形振幅の一番下で取込んだデータ

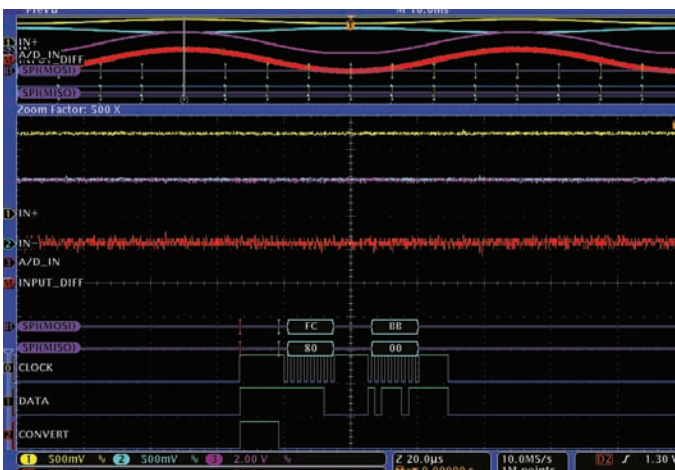


図3.4. 波形振幅の一番上で取込んだデータ

図3.4において、ズーム表示されている波形は、入力電圧波形振幅の一番上の部分になっています。デコードされた出力の値は0xFCBBであり、期待通りに16ビットのA/Dの上限近くの値になっています。

図3.5において、ズーム表示されている波形は、入力電圧波形振幅の真中の部分になっています。デコードされた出力の値は0x7902であり、A/Dレンジのほぼ中間の値になっています。

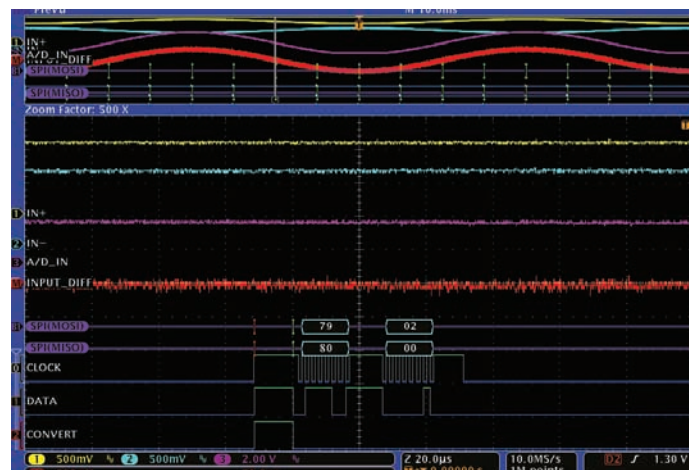


図3.5. 波形振幅の真中で取込んだデータ

この例では、デコードされたデジタル・チャンネルを使用して、A/Dコンバータがフル・レンジで入力電圧を正しく変換していることを確認しました。同時に、アナログ・チャンネルにより、入力アンプが期待通りに動作していることも確認できました。オシロスコープの持つロング・メモリにより、1回の波形取込みでこの解析が行え、十分な観測箇所を容易に確認でき、アナログ入力のさまざまな振幅におけるSPIデジタル出力を確認することができます。

まとめ

マイクロコントローラとペリフェラル・チップ間のやりとりをシリアル通信で行うような設計が増えてくるにつれ、シリアル通信が期待通りに機能していることを検証するためのテスト・ソリューションが必要になっています。このアプリケーション・ノートでは、特定の測定技術を紹介し、ペリフェラル・チップの信号とシリアル・バス・データの相関をとって表示する方法について説明しました。

当社MSO4000シリーズ・オシロスコープはロング・メモリを装備しており、十分な長さのシリアル・バス・データを取込み、デコードし、トリガし、検索することができます。AFG3000シリーズ任意波形／ファンクション・ジェネレータには2チャンネル出力の機種が用意されており、まったく異なった波形、周波数を出力することができるため、高い汎用性を実現しています。

Tektronix お問い合わせ先：

日本

本社 03-6714-3111

SA営業統括部 03-6714-3004

ビデオ計測営業部 03-6714-3005

大宮営業所 048-646-0711

仙台オフィス 022-792-2011

神奈川営業所 045-473-9871

東京営業所 042-573-2111

名古屋営業所 052-581-3547

大阪営業所 06-6397-6531

京都オフィス 075-323-9048

福岡営業所 092-472-2626

湘南カスタム・サービス・センタ 0120-7-41046

地域拠点

米国 1-800-426-2200

中南米 52-55-542-4700

東南アジア諸国／豪州 65-6356-3900

中華人民共和国 86-10-6235-1230

インド 91-80-2227-5577

欧州 44-0-1344-392-400

中近東／北アフリカ 41-52-675-3777

他30カ国

Updated 17 October 2007

詳細について

当社は、最先端テクノロジーに携わるエンジニアのために、資料を用意しています。当社ホームページ(www.tektronix.co.jp)またはwww.tektronix.comをご参照ください。



TEKTRONIXおよびTEKは、Tektronix, Inc.の登録商標です。記載された商品名はすべて各社の商標あるいは登録商標です。

12/07 DM

54Z-21285-0

Tektronix®

Enabling Innovation

日本テクトロニクス株式会社

東京都港区港南2-15-2 品川インターシティB棟6階 〒108-6106
製品についてのご質問・ご相談は、お客様コールセンターまでお問い合わせください。

TEL 03-6714-3010 E-mail ccc.jp@tektronix.com

電話受付時間／9:00～12:00・13:00～18:00 月曜～金曜(休祝日は除く)

当社ホームページをご覧ください。 www.tektronix.co.jp
製品のFAQもご覧ください。 www.tektronix.co.jp/faq/

●記載内容は予告なく変更することがありますので、あらかじめご了承ください。

© Tektronix