



アナログ・デバイセズの DSP、プロセッサ、開発ツール用テクニカル・ノート
<http://www.analog.com/jp/ee-notes>、<http://www.analog.com/jp/processors>にはさまざまな情報を掲載しています。

SHARC®プロセッサ採用システムのデザインとデバッグ

著者: Aseem Vasudev Prabhugaonkar および Alberto Comaschi

Rev 1 - 2006 年 11 月 13 日

はじめに

この EE ノートでは、SHARC®プロセッサを使用するシステムのデザインに関して非常に重要な情報を提供します。これらのガイドラインは、デザイン・サイクルの短縮でハードウェア技術者とファームウェアまたはソフトウェア技術者を支援することを目的としています。以下に示す幾つかの推奨事項は、それぞれの SHARC プロセッサのハードウェア・リファレンス・マニュアルにも記載されています。この EE ノートは、ハードウェア・ボードのデザイン・ガイドライン、ソフトウェア手順とこつ、デバッグのヒントの 3 つのセクションから構成されています。特に指定がない限り、このヒントはすべての SHARC プロセッサに適用できます。

ハードウェアとボード・デザインのチェック・ポイント

このセクションでは、ボード技術者向けのヒントを提供します。

SPI インターフェース

*ADSP-2126x および ADSP-21362/3/4/5/6 プロセッサでの SPI ブート・インターフェース対
ADSP-21367/8/9 および ADSP-2137x SHARC プロセッサ*

上記 SHARC プロセッサは、SPI メモリ・デバイスからのブート・ロードをサポートしています。これらのプロセッサは SPI フラッシュ・ブートに設定されると、ボード上の SPI メモリからアプリケーション・イメージをブート・ロードすることができます。

一般に使用されている大部分の SPI フラッシュ・デバイスでは、パワーアップ後に命令を実行する前にチップ・セレクト(/CS)に立ち下がりエッジを必要とします。ADSP-2126x と ADSP-21362/3/4/5/6 プロセッサの場合、4.7kΩの抵抗を使って SPI フラッシュのチップ・セレクトをプルアップしてください。これらの SHARC プロセッサでは、この信号に内部プルアップがないため、自分自身でロジック・ハイからロジック・ローへの変化を行いません。このため、SPI フラッシュのセレクトはプロセッサから最初の命令が出力されるまでロー・レベル(または不定)を維持します。ADSP-21367/8/9 と ADSP-2137x プロセッサを使用するデザインでは、このプルアップ抵抗は不要です。これは、SPI フラッシュ・ブート・ロード動作時に立ち下がりエッジ要求を満たすための内部プルアップ抵抗がこれらのプロセッサに内蔵されているためです。この要求を満たす SPI フラッシュ・メモリ・デバイスの一例は、ST Microelectronics 社の MP25P80 シリアル・フラッシュ・デバイスです。

MOSI と MISO

SPI インターフェースでは、すべての MOSI ピンを相互に接続し、さらにすべての MISO ピンを相互に接続することが必要です。信号の競合とそのピンの損傷を防止するために、これらのピンが置き換わっていないことをダブルチェックしてください。MISO と MISO を、MOSI と MOSI を、それぞれ接続してください。ペリフェラル・ピン名が DIN または DOUT である場合は、それぞれのマスター機能またはスレーブ機能に応じて、これらの信号を接続してください。適切な回路図信号名を使用すると、混乱を防止できます。

JTAGデザインとブート問題

大部分のシステムは、プロトタイプと量産前のユニットをJTAG ICE (インサーキット・エミュレータ)を使ってテストとデバッグを行えるように、最初からJTAG接続を持つようにデザインされています。この場合、JTAG /TRST信号(TAPリセット)はICEから駆動されますが、ブート動作によりシステムがスタンドアロン・モードで動作する場合、またはICEを使用しない場合には、/TRST信号をボードのグラウンドに接続してください。/TRST信号をグラウンドに接続しないと、ブート・エラーやランタイム時にメモリ・アクセス・エラーが発生する原因になります。さらに、/TRSTにプルダウン抵抗を接続することは推奨されません。これは、SHARCプロセッサがこの信号のプルアップ抵抗を内蔵しているためです。JTAGシステムのデザイン・ガイドラインについては、「*Analog Devices JTAG Emulation Technical Reference (EE-68)*^[1]」を参照してください。

2 線式インターフェース(TWI)

内蔵の 2 線式インターフェースは、I²C 互換ペリフェラルです。SCL と SDA はオープン・ドレイン出力であるため、両 TWI 信号には、I²C 規格で規定されているようにプルアップ抵抗が必要です(プルアップ抵抗値を選択するときは、I²C 規格と I²C 互換デバイスのデータ・シートを参照してください)。

/RESET入力の駆動

SHARC プロセッサの /RESET 入力信号を駆動するときは、RC (抵抗/コンデンサ)回路の使用を避けてください。アナログ・デバイセズは、プロセッサへのパワーオンとマニュアル /RESET を出力する電源モニターIC の使用を推奨します。RC 回路とシュミット・トリガー・レベル・ゲートとの組み合わせも、/RESET 入力の駆動に使用することができます。

バイパス・コンデンサ

内部電源(VDDINT)に適切なバイパス・コンデンサを使用することが、高速動作では不可欠になります。不要な寄生インダクタンスがコンデンサとパターンに存在すると、高周波での有効性が損なわれます。プロセッサが100 MHz以上で動作する場合は、次の2つのことが必要になります。1つ目は、コンデンサは小型で、インダクタンスを小さくするためリードが短い必要があります。サイズ0402の表面実装コンデンサは、それより大きいサイズより優れています。2つ目は、容量値を小さくすると、LC回路の共振周波数が高くなることです。幾つかの0.1 uFコンデンサは50 MHz以下で優れていますが、500 MHz範囲でのVDD_INTに対しては、0.1uF、0.01uF、0.001 uF、100 pFの組み合わせが望まれます。

AVDD電源のフィルタ回路

このセクションは、ADSP-2116x、ADSP-2126x、ADSP-21362/3/4/5/6 SHARC プロセッサに適用されます。これらのデバイスのデータ・シートでは、内蔵 PLL で使用している AVDD 電源に対するフィルタ回路を推奨しています。データ・シートの旧レビジョンではノイズ耐性と PLL の安定性のために 10Ω直列抵抗の使用を推奨していましたが、この定格は高インピーダンス(100 MHzで 600~1000Ω)のフェライト・ビーズを使うように更新されました。これらのタイプのフェライト・ビーズの DC 抵抗は 1Ω以下です。

未使用入力信号

未使用プロセッサ入力はフローティングのままにしないでください。入力信号のアクティブ極性に応じて、プルアップ抵抗またはプルダウン抵抗を使用してください。プルアップとプルダウンの推奨抵抗値は、それぞれ 10kΩと 100Ωです。プルアップ/プルダウン抵抗を内蔵している入力のみ、フローティングのままにすることができます。デフォルトでプルアップ/プルダウン抵抗を内蔵している入力については、デバイスのデータ・シートで確認してください。未使用のバス要求(/BRx)やホスト・バス要求(/HBR)のような信号をフローティングのままにすると、ブート・エラーやアプリケーションのランタイム時にその他の問題が発生します。

EZ-KIT Lite®の回路図

EZ-KIT Lite®評価システムの回路図は、開始時の参考資料として利用することができます。EZ-KIT Liteボードは評価と開発を対象としているため、余分な回路が設けられていることがあります。アクセスなどを容易にするため、部品の実装があることも、ないこともあるため、EZ-KIT Liteボードの回路図には注意が必要です。SHARCプロセッサEZ-KIT Liteボードのデザイン・データベースはオンラインで提供しており、デザイン、レイアウト、製造、組み立てに必要なすべての電氣的条件が含まれています。

ftp://ftp.analog.com/pub/tools/Hardware/Reference_Designsから提供しています。

テスト・ポイントと信号アクセス

CLKOUT/RSTOUT、SDRAM クロック(SDCLK)、/MSx メモリ・バンク・セレクト、/BMS、/RESET のような信号にテスト・ポイントを追加すると、プロトタイプ・ボード上でのデバッグ・プロセスで役立ちます。ブート・モード(BOOTCFG)やコア・クロック・レート(CLKCFG)のようなセレクト・ピンが電源またはグラウンドに直接接続されている場合、BGA パッケージ・チップではこれらの信号をアクセスすることができません。デバッグ用として、電源またはグラウンドへ直接接続する代わりにプルアップ抵抗とプルダウン抵抗を使うと役立ちます。

信号インテグリティのヒント

高速な立ち上がり時間と立ち下がり時間は、信号インテグリティ問題の基本的な原因になります。SHARC プロセッサのこれらのエッジ・レートは、ピンごとに異なります。同様に、幾つかのピンは他のピンよりノイズと反射に対して高い感度を持ちます。異常なクロック信号と同期信号を発生させる伝送線反射を防止するシンプルな信号インテグリティ方法を使用してください。次のペリフェラルと信号では、パターン長を短くすることと直列終端が不可欠です。

- **SPORT インターフェース信号(TCLK、RCLK、RFS、TFS):** これらの信号にノイズまたはグリッチがあると SPORT が正しく機能しなくなります。これらの信号にグリッチがあると、SPORT のロックアップ状態、チャンネル・スワップ、チャンネル・シフト、データ破壊のような症状が現れます。このため、パターンが長い場合またはシミュレーション結果で反射が示された場合には、これらのラインに終端抵抗を使用してください。
- **CLKIN ソース:** CLKIN のソースとして水晶を使う場合には、水晶メーカーが推奨するコンデンサを水晶に使用してください。できるだけ基本モード水晶を使用してください。CLKIN のソースとして発振器を使う場合は、プロセッサの XTAL ピンを N/C (接続なし) にしてください。XTAL/CLKIN 回路のデザインと推奨部品値については、データ・シートを参照してください。高速信号(たとえば SDRAM クロック、外部ポート・データ・バス、またはアドレス・バス)の配線は、XTAL/CLKIN 信号と回路に近づけないでください。クロストークによりノイズが混入して、PLL 性能に影響を与えます。外部発振器を使って CLKIN を駆動するときは、拡散スペクトル発振器を使ってクロック・ソースによる EMI 放出を削減してください。複数の SHARC プロセッサを持つシステムをデザインする場合は、スキューの小さいクロック・バッファ/ドライバを使用して CLKIN を 1 つのクロック発振器から得てください。
- **SDRAM クロック、コントロール、アドレス、データの各信号でもパターンを短くして、直列終端を使用して反射を防止すると、不要な EMI を削減することができます。**
- **メモリ・チップのような IC にはできるだけソケットを使用しないでください。**ソケットを使用すると寄生が増えるため、信号インテグリティ性能が低下します。複数のソースを持つ信号の場合、パターンを短くすることは困難であるため、シミュレーションによる対応が適します。信号シミュレーションに役立つ IBIS モデルをアナログ・デバイセズのウェブ・サイトから提供しています。

一般的なガイドラインとポイント

- SHARC プロセッサのパワーアップ: 両電源(VDDINT [コア電源]と VDDEXT [IO 電源])を同時にパワーアップさせてください。同時にパワーアップできない場合は、安定な電源間の時間差がデータ・シートの仕様を超えないようにしてください($t_{VDDDEVDD}$ [VDDINT から VDDEXT まで])。
- 共用される信号: 共用される信号が I/O 機能を持っていることに注意してください。これらの信号は、/RESET でパワーオン後デフォルトの機能を持ちます。デフォルト機能から所望の機能へ変更するときは、ソフトウェア・プログラミングが必要です。システム・デザインの点から、信号の競合が生ずる原因になります。幾つかの例を次に示します。
 - ADSP-21367/8/9 SHARC プロセッサでは、/MS2 ストローブと/MS3 ストローブが、フラグ・ピンおよび割り込みピンと共用されています。パワーオン後、これらの信号は入力に設定されます。このため、これらの信号をメモリ・セレクト信号として使う場合、外部ポートとの競合を避けるために、これらの信号にプルアップ抵抗が必要になります。たとえば、外部フラッシュ・メモリからブート・ロードを行うときに、/MS1 (バンク 1)を使います。/MS2 または/MS3 にインターフェースしているメモリ・デバイスが、チップ・セレクト・ピンのロジック・レベルを誤判断してしまい、バスの駆動を開始すると、バス競合が発生します。
 - もう 1 つの例は、ADSP-2137x SHARC プロセッサ上のランニング・リセット機能と共用される RSTOUT/CLKOUT 信号です(図 1 参照)。/RESET でのパワーオン時に、この信号は RSTOUT 信号として動作します。ソフトウェアを実行してこの信号を入力として設定し、ランニング・リセット信号を動作させます。ADSP-2137x プロセッサ上の RSTOUT/CLKOUT 信号は、ランニング・リセット機能を実行する際に使用して、ホストの OD (オープン・ドレイン)出力から駆動される必要があります。パワーアップ時、この信号は出力として設定されます。この状態は、ソフトウェアからこの信号を入力として設定して、ランニング・リセット信号を動作させるまで続きます。この信号をホストのアクティブ・ドレイン出力に接続すると競合が発生して、ドライバが損傷されます。
- ブート・メモリ・チップ・セレクト: ADSP-2106x と ADSP-2116x SHARC プロセッサは、パラレル・ブート・メモリ・デバイスを駆動する専用信号(/BMS [バイト・メモリ・セレクト])を持っています。ADSP-2126x と ADSP-21362/3/4/5/6 プロセッサの場合、専用のブート・メモリ・セレクト信号を持っていません。したがって、メモリ・セレクトはプロセッサから駆動されるパラレル・ポート・アドレスで駆動される必要があります。ADSP-21367/8/9 と ADSP-2137x プロセッサの場合、/MS1 (メモリ・バンク 1 セレクト)は、ブート・メモリ・セレクト信号として使う必要があります。ブートはバンク 1 から行われます。ADSP-21367/8/9 と ADSP-2137x SHARC プロセッサには/BMS のような専用信号はありません。

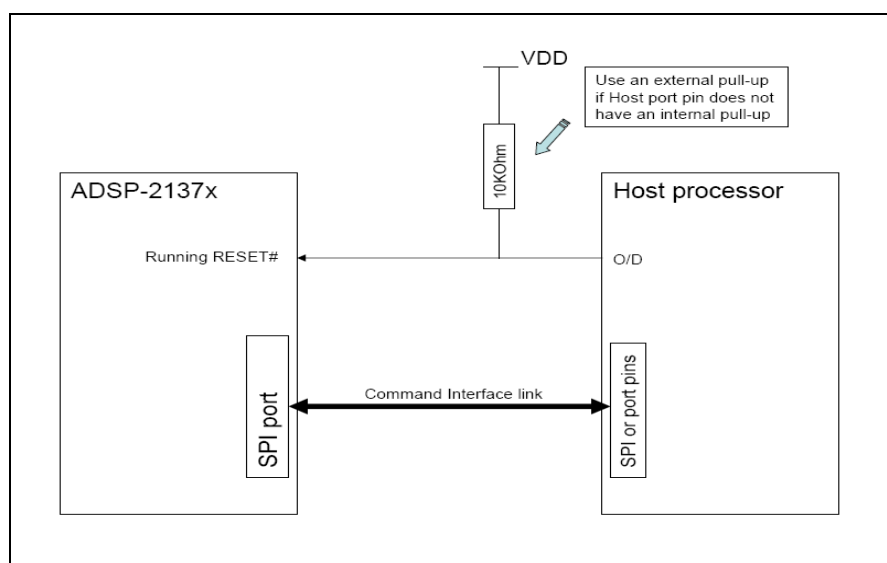


図1. リセット方式の実行を示す一般的なシステム・ブロック図

- SDRAM アドレス・ピンのマッピングと DQM 信号: ADSP-21367/8/9 と ADSP-2137x SHARC プロセッサのアドレス・ピン・マッピングは、ADSP-21161 プロセッサと異なっています。32 ビット・モードで使用する場合は ADSP-21367/8/9 と ADSP-2137x プロセッサの ADDR1 を SDRAM の ADDR0 に接続してください。16 ビット・モードで使用する場合は、ADDR0 を SDRAM の ADDR0 に接続してください。ADSP-21161 プロセッサとは異なり、ADSP-21367/8/9 と ADSP-2137x プロセッサには、DQM のサポートはありません。SDRAM のパワーアップ・シーケンスで特別な DQM 条件を持つ SDRAM デバイスもあります。このような SDRAM メモリの場合、SDRAM のパワーアップ・シーケンスで DQM 信号をプロセッサのフラグ・ピンを使って駆動する必要があります。ADSP-21367/8/9 プロセッサとの SDRAM インターフェースの詳細については、「*Interfacing 133 MHz SDRAM Memory to ADSP-21367 SHARC Processors (EE-286)*^[2]」を参照してください。

ソフトウェア手順とヒント

このセクションでは、幾つかのペリフェラルと SHARC プロセッサのユニットに対する推奨手順について説明します。

PCG (ペリフェラル・クロック・ジェネレータ)の設定

このセクションでは、PCG (高精度クロック・ジェネレータ)を使って SHARC SPORT の信号を発生する際の考慮事項について説明します。

- PCG を使って I²S モードに設定された SPORT の FS (LRCLK としてのフレーム同期)や SCLK (シリアル・ビット・クロック)のような信号を発生するときは、信号タイミングを I²S プロトコル・タイミングに準拠させるために PCG の遅延レジスタを設定する必要があります。LRCLK 信号は、シリアル・クロックの立ち下がりエッジで駆動する必要があります。
- TDM モードに設定された SPORT の信号を PCG を使って発生させるときは、フレーム同期はシリアル・クロックで 1 ビット・サイクル間だけアクティブを維持する必要があります。TDM モードでのフレーム同期は、エッジ検出ではなくレベル検出であることに注意してください。PCG パルス幅レジスタ(PCG_PW)には値“1”をロードして、フレーム同期がシリアル・クロックで 1 サイクル間アクティブになるようにする必要があります。この設定では、フレーム同期がシリアル・クロックで 1 サイクル間アクティブであるため、何時 SPORT がイネーブルされるか、または再イネーブルされるかは問題ではありません。フレーム同期が非アクティブのとき SPORT がイネーブルされた場合には、次のアクティブ・フレーム同期を待ちます。PCG 設定で 50%のデューティ・サイクルを必要とするシステムでは、内部 DAI 割り込みを使って SPORT をイネーブルしてください。

PCG のフレーム同期とクロックが開始された後に、SPORT がイネーブルまたは再イネーブルされる場合について考えてみます。フレーム同期とクロックが開始された後には何時でもシリアル・ポートをイネーブルできるため、シリアル・ポートがデータのラッチを開始する場合(レシーバの場合)、または有効なフレーム同期がサンプルされるたびにデータの駆動を開始する場合があります(トランスミッタの場合)。TDM モードでは、フレーム同期はエッジ検出ではなくレベル検出であることに注意してください。このような状況は、シリアル・ポートがディスエーブルされると、および再イネーブルされるとに繰り返されるので、同期が失われる原因になります。これにより、チャンネル・シフトやデータ喪失のような症状が発生します。シリアル・ポートを開始させる前の正しい手順は、フレーム同期の非アクティブ・エッジを待った後にフレーム同期がまだ非アクティブの間にシリアル・ポートをイネーブルすることです。これにより、シリアル・ポートが確実にアクティブなフレーム同期レベルを待ち、データをサンプルすることが保証されます。

この手順は、シリアル・ポートがディスエーブルされ、さらに再イネーブルされるときにも実行して、シリアル・ポートがアクティブ・フレームの途中でデータをサンプルしないようにする必要があります。これは次の手順を使って実現されます。PCG フレーム同期を DAI 割り込みに割り当てます。フレーム同期の非アクティブ・エッジにより DAI 割り込みが発生します。これは、フレーム同期がある期間非アクティブであることを示しています。フレーム同期が非アクティブである間のこの期間にシリアル・ポートをイネーブルします。シリアル・ポートがディスエーブルされ再イネーブルされるときにも同じ手順を実行します。

IDPを使ったI2Sのイネーブル(入力データ・ポート)

IDP を使って外部デバイスからデータを受信する場合、I2S モードでデータを受信するように設定された IDP ポートをイネーブルする際に従うべきシーケンスがあります。このシーケンスに従わないと、チャンネル・シフトまたはスワップが発生します。

1. SRU (信号ルーティング・ユニット)を使って、フレーム同期を DAI 割り込みに内部で接続します。
2. DAI 割り込みをフレーム同期の非アクティブ・エッジ用に設定します。
3. DAI 割り込みを待ち、DAI 割り込みサービス・ルーチン内部で IDP ポートをイネーブルします。
4. DAI 割り込みラッチ・レジスタを読み出して、DAI 割り込みをクリアします。

この手順により、IDP ポートが正しいタイミングでイネーブルされるようになり、受信データ内でチャンネル・シフトやスワップのような問題の発生が回避されます。

DAI割り込み

DAI 割り込みは他の割り込みとは異なり、DAI 割り込みサービス・ルーチン内で自動的にクリアされません。割り込みをラッチした後、DAI 割り込みラッチ・レジスタを読み出すことにより、ラッチ・ステータスがクリアされるまでそのラッチ状態が続きます。DAI 割り込みをクリアするときは、DAI 割り込みラッチ・レジスタを読み出す必要があります。これを行わないと、DAI 割り込みサービス・ルーチンを終了するごとに、再度起動されることが続きます。

DMA割り込みサービス・ルーチン内でのペリフェラルのディスエーブル

SHARC プロセッサ・ペリフェラルは、データ転送に対して DMA モードをサポートしています。DMA 転送カウンタがカウントアウトすると DMA 割り込みが発生します。ペリフェラルをデータ受信に設定した場合、データの受信が完了して内部メモリへ転送されたとき、DMA カウンタがカウントアウトします。ペリフェラルをデータ送信に設定した場合、DMA カウンタがカウントアウトして割り込みが発生しても、データはペリフェラルの DMA FIFO 内に留まります。ソフトウェアから ISR 内で DMA とペリフェラルをディスエーブルしようとするときは、DMA FIFO のステータスを最初にポーリングする必要があります。FIFO がエンプティ・ステータスを表示したときには、ペリフェラルを安全にディスエーブルできますが、そうでない場合はデータが失われることがあります。

SPORT/SPIのダブル割り込み

このセクションは、ADSP-21367/8/9 と ADSP-2137x SHARC プロセッサのみに適用されます。SPI をコア・モードで使用してデータを送信する場合について考えます。コアが送信バッファ・エンプティ状態に遭遇すると、送信割り込みが発生します。送信割り込みサービス・ルーチン内で、ソフトウェアから新しい値を送信バッファに書き込み、割り込みからリターンします。このような状況では、ISR から戻ると、プロセッサは送信バッファ・エンプティ状態を再度検出してしまうため、SPI 送信 ISR が実行されます。これは、IOP 書き込みが高度にパイプライン化されているために発生します。送信バッファへの書き込み後、実際の値が書き込まれて、送信バッファ・ステータスが"非エンプティ"に更新されるまでにコア・クロックで 10 サイクルを要します。このため、割り込みルーチンからのリターンをコア・クロックで少なくとも 10 サイクル遅延させる必要があります。

SHARC PLLの設定

ADSP-2116x SHARC プロセッサの場合、外部 CLKCFG (クロック・コンフィギュレーション)信号により PLL が設定されます(コア・クロックと外部ポート・クロック周波数も設定されます)。ADSP-2126x、ADSP-2136x、ADSP-2137x SHARC プロセッサの場合、外部 CLKCFG 信号の他に、PLL をソフトウェアから設定することができます。したがって、ソフトウェアからは、PLL の通倍カウント値と分周カウント値を使った種々のプログラマブルな比率が可能で、これはパワー・マネジメント・コントロール・レジスタ(PMCTL)に書き込まれます。これはソフトウェアからコア周波数を変更する柔軟性も提供します。PLL の設定では、「*Managing the Core PLL on ADSP-2136x SHARC Processors (EE-290)*^[3]」に記載する推奨シーケンスに従ってください。DIVEN ビットをクリアした後に、バイパス・モードへの切り替えと、バイパス・モードからの切り替えを行うようにしてください。EE-290 には、elfar ユーティリティを使って PLL 設定の“C 言語呼び出し可能”関数をライブラリに組込むコード例とステップごとの手順も記載してあります。

SPORTとゲーティングされたクロック・デバイスとのインターフェース

システム・デザインによっては、データ・コンバータやホスト・プロセッサ SPI のようなゲーティングされたクロック・デバイスへ SHARC SPORT をインターフェースさせる必要がある場合があります。「*Interfacing Gated Clocks to ADSP-21065L SHARC Processors (EE-244)*^[4]」, *Interfacing AD7676 ADCs to ADSP-21065L SHARC Processors (EE-247)*^[5], and *Interfacing AD7676 ADCs to ADSP-21365 SHARC Processors (EE-248)*^[6]」では、ゲーティングされたクロック・デバイスとの正しい動作とデータ転送を確実にするガイドラインを、例を使って説明しています。

デバッグのヒント

治療よりは予防とよく言われます。システム・デザイン技術者とソフトウェア技術者は必ず、使用しようとしている SHARC プロセッサのアノマリ・ドキュメントをお読みください。既知のシリコン問題と対策を知ることができ、デザイン・サイクル・タイムを短縮することができます。ハードウェアに関する問題の場合(たとえばボード・デザイン)、高価で時間のかかるボードの再デザインを回避するのに役立ちます。

このセクションでは、一般的に遭遇する問題について検討し、これのデバッグ方法についてヒントを提供します。これらのヒントは、ユーザの他のデバッグ努力を補完するものです。最も一般的に報告されるケースについて以下に説明します。

ブート・エラー

これは、JTAG ICE を使ってアプリケーションをロードして実行できるのですが、ブート・ロードしたときにアプリケーションを実行できないケースとして報告されます。デバッグ時に次のポイントを検討してください。

- プロセッサの **B00TCFG** ピンをチェックし、デジタル電圧のハイ・レベルまたはロー・レベルに正しく接続されていることを確認してください。抵抗を使ってブート・モード変更の柔軟性を提供するときは、正しい抵抗が実装されていることを確認してください。プロセッサ・ピンでオシロスコープを使って電圧レベルをチェックするのが最善の方法です。
- JTAG ICE の **/TRST** 信号がボードのグラウンドに接続されていることを確認してください。この信号をフローティングのままにしないでください。この信号をフローティングのままにすると、ブート・エラーまたはメモリ・アクセス・エラーが発生する原因になります。
- 正しいブート・カーネルを使用していることを確認してから、ローダ(**.LDR**)ファイルを生成してください。変更されたブート・カーネルを使用している場合は、**VisualDSP++** およびアプリケーション例(たとえばフラグ・トグル)と一緒に提供されたデフォルト・ブート・カーネルを使って基本ブート・ロードを確認してみてください。
- **.LDR** ファイルを生成する際に正しいパラメータを選択していることを確認してください。不適切なパラメータを選択すると、ブート・エラーが発生する原因になります。
- ICE (インサーキット・エミュレータ)を使ってブート・ロードしたコードであるか否かをチェックしてください。ICE をターゲットに接続して、**VisualDSP++**内でシミュレータ・セッションを開始してください。ターゲットをターンオンして、ブート・ロードを開始します。セッションを“シミュレータ”から“エミュレータ”へ変更して、**Disassembly** ウィンドウを調べてください。このウィンドウに、外部ソースからアプリケーションがブート・ロードされたか否かが表示されます。期待したコードが表示されない場合、ブート・エラーがあったことを意味します。また、この方法を使って、プロセッサが最初の 256 命令をダウンロードしたか否かをチェックすることができます。
- データ・シートで規定された正しいパワーオン・リセット・タイミングであることを確認してください。
- **CLKCFG** 信号をチェックして、PLL がオーバードライブされていないことを確認してください。**CLKIN** 周波数との関係で選択した比が、規定値より大きな値に対してコア・クロックを超えていないことを確認してください。
- **/BMS** (ADSP-21367/8/9 と ADSP-2137x プロセッサでは **/MS1**)、**/RD**、**ADDR**、バス上のデータのようなブート関連信号を調べてください。ボード上のパターンが断線または短絡することがあるので、パターンが正常であることをチェックしてください。**RSTOUT** を監視して PLL のロックを確認することもできます。
- ブート・ロード動作時に外部バスにバス競合がないかチェックしてください。たとえば、ADSP-21367/8/9 と ADSP-2137x プロセッサの場合、**/MS2** と **/MS3** (使用時)に外部プルアップ抵抗が必要です。そうしないと、**/MS2** と **/MS3** にインターフェースしているメモリ・デバイスがバスを駆動すると、ブートアップ時にバス競合が発生する原因になります。

アプリケーションのクラッシュ

大部分の場合、この問題はアプリケーション実行中のソフトウェア・クラッシュとして報告されます。プロセッサが未知のロケーションに進んだとき単純にハングアップするか、またはアプリケーション全体が再起動するケースがあります。この発生原因には次が含まれます。

- プロセッサがインダイレクト・アクセスを実行しており、このインダイレクト・アクセスで **.LDF** ファイルで定義されたヒープまたはスタック・メモリ・ロケーションが壊される可能性があります。これにより、ランタイム時に必要な重要データが壊される可能性があり、最終的にソフトウェアのクラッシュが発生する可能性があります。破壊を生ずるこのようなメモリ・アクセスをアプリケーションで行っていないことを確認してください。ランタイム時は、ICE を使った **VisualDSP++** のハードウェア・ブレーク・ポイント機能を使ってこれをデバッグすることができます。これらの領域でプロセッサが違法アクセスを実行している場合、これらのアクセスを検出することができます。
- アプリケーションのハングまたは再起動を発生させる、システム・レジスタまたは **IOP** レジスタへの有害/偶発的なアクセスをアプリケーションが実行しているか否かをチェックしてください。たとえば、コード内の何処かで誤ってソフト・リスタート/リブートをアプリケーションが設定していないかを、チェックしてください。インダイレクト・ジャンプまたはコールをチェックしてください。
- デカップリング/バイパス・コンデンサが不十分なボード・デザインでは、特定のコードの実行と関連のある予期しない動作が発生することがあります。ボード上で、プロセッサやその他のデバイスに対して正しいデカップリング・コンデンサとバルク・コンデンサ(コンデンサのタイプ/値と配置場所が非常に重要)が使用されていて、システム・ランタイム時に必要な十分なスイッチング電流/電源を供給していることを確認してください。重いコアまたは **I/O** のスイッチングでは大きなスイッチング電流が必要になります。電流が供給されない場合には、システム・シャットダウン、アプリケーション・クラッシュ、または特定コードの不正な実行などの症状が現れます。
- ブート・ロードされたアプリケーションのチェックサムを確認して、データとコード全体がビット・エラーなしで正常にダウンロードされたことを確認することは有効です。ビット・エラーはノイズの多い環境で発生します。ビット・エラーのため、プロセッサが不正(または違法)命令を実行または不正データを処理して、予期しない動作が発生します。
- **DAG** (データ・アドレス・ジェネレータ)のレンジ・レジスタ(**Lx**)を循環バッファとして使用しない場合、ゼロに初期化していることを確認してください。ICE を使ってコードをダウンロードして実行する場合、これに関係する症状は見えません。これは、ICE がレンジ・レジスタをゼロに初期化しているため、コードがブート・ロードされてスタンドアロン・システム内で実行されたときにのみ症状が現れるためです。
- 特定のツール・バージョン(またはアップデート)に関する既知の **VisualDSP++** コンパイラ・バグについて調べてください。コンパイラが不正なコードを生成すると、予期しない動作が発生する原因になります。常に **VisualDSP++** の最新バージョンを使用することが推奨されます。

最新 **VisualDSP++** リリース・ノートはアナログ・デバイセズのウェブ・サイトから提供しています。

www.analog.com/processors/sharc/evaluationDevelopment/crosscore/toolsUpgrades/index.html

また、旧 **VisualDSP++** バージョンのリリースノートは、

www.analog.com/processors/sharc/evaluationDevelopment/crosscore/toolsUpgrades/archives.html から提供しています。

データの喪失

これは、SPORT、SPI、またはその他のシリアル・インターフェースを使用したデータの受信中に、データの先頭数ワードの喪失として報告されます。このようなケースでは、常にマスターをイネーブルする前にスレーブをイネーブルすることを確認してください。マスターがシリアル・クロックやフレーム同期のような信号を発生し、スレーブがこれらの信号をマスター・デバイスから取得します。

データの破壊

これは、SPORT、SPI、リンク・ポートなどのシリアル・リンクを使ってデータを受信しているときのデータの破壊として報告されます。このようなケースでは、信号の正常性を確認し、オーバーシュート/アンダーシュートに起因する問題とノイズを導入するクロストークまたはその他のソースに起因するノイズ・スパイクに注意してください。可能な場合には、内部ループ・バックを行って、プロセッサの機能的問題を除外します。無限繰り返しモードではオシロスコープを使って信号インテグリティ問題を探します。このようなケースでは、シリアル・クロックやフレーム同期などで終端を行って見てください(直列終端が望ましい)。これらの問題は、デザインでその他の機能的推奨事項を採用していても、受信されたチャンネルでシフト/スワップとして現れることがあります。

VisualDSP++のポータビリティ問題

これらは、VisualDSP++バージョン(またはアップデート)が変更されたときに見られる問題として報告されます。このようなケースでは、コンパイラ、既知のツール問題、その他での違いをチェックしてください。コード最適化をターンオフして、コンパイラ最適化関連問題の範囲をさらに絞ってください。できるだけプロジェクト開始時のVisualDSP++の最新バージョン/アップデートを使用してください。

アペンディックス

	Board Design Checklist
☐	Check the size and placement of decoupling capacitors
☐	Check the boot mode selection pins (B00TCFG)
☐	Check the core clock rate selection pins (CLKCFG)
☐	Check JTAG pins configurations. See <i>JTAG Emulation Technical Reference (EE-68)</i> ^[1]
☐	Check test points and probe point accessibility. Unused PF pins are very useful for hardware and software debug purposes (for example driving LEDs or probe points). If possible, pin out all DSP signals through vias.
☐	Check the reset circuitry. Reset should be generated by a reset supervisory IC, not by RC circuitry. RC network use is possible in combination with a Schmitt trigger gate to drive RESET/ of the processor.
☐	Check power supply ratings and thermal requirements. It is preferable to be able to adjust the core voltage by simply changing a resistor value.
☐	Check component placement, and avoid long traces
☐	Check the processor's data sheet and errata for up-to-date information

参考

- [1] Analog Devices JTAG Emulation Technical Reference (EE-68), *Rev 9, October 2004, Analog Devices, Inc.*
- [2] Interfacing 133 MHz SDRAM Memory to ADSP-21367 SHARC Processors (EE-286), *Rev 2, March 2006, Analog Devices, Inc.*
- [3] Managing the Core PLL on ADSP-2136x SHARC Processors (EE-290), *Rev 1, June 2006, Analog Devices, Inc.*
- [4] Interfacing Gated Clocks to ADSP-21065L SHARC Processors (EE-244), *Rev 1, September 2004, Analog Devices, Inc.*
- [5] Interfacing AD7676 ADCs to ADSP-21065L SHARC Processors (EE-247), *Rev 1, October 2004, Analog Devices, Inc.*
- [6] Interfacing AD7676 ADCs to ADSP-21365 SHARC Processors (EE-248), *Rev 1, October 2004, Analog Devices, Inc.*
- [7] *Hardware Design Checklist for the Blackfin Processors (EE-281)*, *Rev 1, October 2005, Analog Devices, Inc.*

ドキュメント改訂履歴

Revision	Description
<i>Rev 1 – November 13, 2006 by Alberto Comaschi and Aseem Vasudev Prabhugaonkar</i>	Initial Release