

目次

1.0 目的	2
2.0 SPIインターフェース	2
2.1 概要	2
2.2 詳細な説明	4
2.3 Okikaデバイスにおけるデータ転送用SPI波形デバイス	6
3.0 複数のFPAAへのアドレス指定	7
3.1 FPAA チェーン	7
3.2 並列構成	9
4 その他の考慮事項	10
4.1 クロックバッファリング	10
4.2 レイアウト	10

1.0 目的

本ドキュメントの目的は、SPIプロトコルを説明し、複数のFPAAデバイスをアドレス指定するためにどのように使用できるかを記述することです。

SPIプロトコルについて詳細に説明した後、直列および並列の両方で複数のFPAAをアドレス指定するためにそれをどのように使用するかを説明します。最後に、バッファリングとレイアウトに関する考慮事項についてのセクションがあります。

2.0 SPIインターフェース

2.1 概要

SPIはシリアル周辺機器インターフェース（Serial Peripheral Interface）の略称であり、PC筐体内部のような短距離向けの低コスト・低速通信オプションです。全二重通信に対応し、2つのホスト間での実装が非常に容易です。SPIは数十MHzのデータレートを実現可能ですが、一般的なプロセッサベースのSPIインターフェースは通常1～3MHzの範囲のデータレートをサポートします。

SPIプロトコルを使用する場合、マスターと1つ以上のスレーブが定義されます。Okika Devicesデバイスを使用するシステムでは、マスターは常にホストプロセッサです。Okika Devicesデバイスは常にSPIスレーブです。図1はシングルマスター・シングルスレーブ構成を示しています：

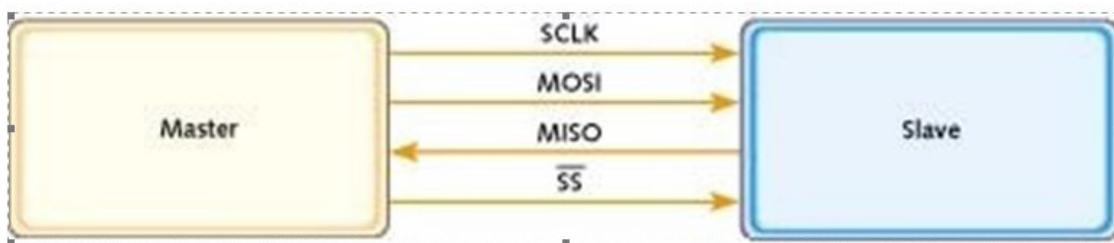


Figure 1: Single master / single slave

信号名	説明	AN231E04 ピン名
SCLK	クロック	SCLK
MOSI	マスター出力 - スレーブ入力	SI
MISO	マスター入力 - スレーブ出力	SO
SS	スレーブ選択	CS2b

表1: 信号名

クロックはマスターによって生成され、すべてのスレーブに送信される。スレーブセレクトラインは、マスターがどのスレーブと通信しているかを示すために使用される。図2は複数のスレーブ（例えばOkika Devices社のFPAA）が存在するケースを示している。

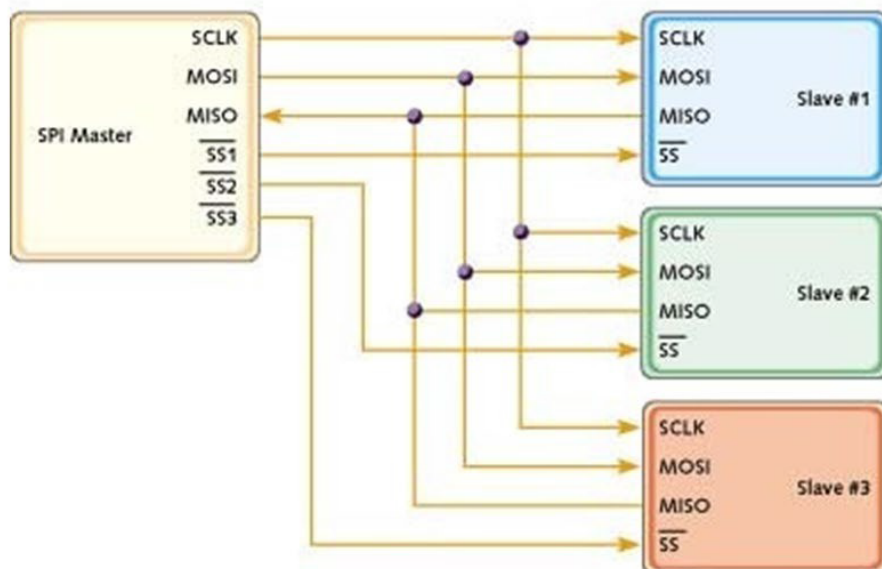


Figure 2: Single master / multiple slaves

図からわかるように、スレーブセレクトを除く全ての信号は全スレーブで共通です。したがって通信時には、マスターがどのスレーブをアクティブにするかを明示する必要があります。スレーブ数が多い場合、マスター側の必要出力数と基板上の配線数が膨大になり、このプロトコルの採用は現実的ではありません。しかしOkika Devices FPAAは、デバイスへの設定データに埋め込まれたIDフィールドを活用することでこの課題を克服しています。

Okika Devices FPAAでは、全てのCS2bチップセレクトを並列に接続でき、CS2bがローかつ該当デバイスのIDが正しい場合にのみ個別デバイスと通信します。もちろん、必要に応じて別々のセレクトラインを使用してOkika Devices FPAAをアドレス指定することも可能です。

注記：FPAAにはCS1bとCS2bの2本のセレクトラインがあります。CS1bはFPAAのチェーン接続に使用されるため、通常CS2bがSPIセレクトとして使用されます。詳細はセクション3で説明します。

2.2 詳細な説明

SPIは、クロック信号に対するデータの有効タイミングを決定するために、クロック極性（CPOL）とクロック位相（CPHA）と呼ばれる2つのパラメータを使用します。通信を機能させるためには、通常、マスターとすべてのスレーブでこれらの設定を同一にします。CPOLは、クロックの立ち上がりエッジまたは立ち下がりエッジのどちらをデータ定義の先頭エッジとするかを決定します（後端エッジについては逆の処理を行います）。CPHAは、立ち上がりエッジがセットアップ用かサンプル用かを決定します（立ち下がりエッジは逆）。表2に各種設定をまとめます：

CPOL/CPHA	Leading Edge	Trailing Edge	SPI Mode
0/0	Sample, rising	Setup, falling	0
0/1	Setup, rising	Sample, falling	1
1/0	Sample, falling	Setup, rising	2
1/1	Setup, falling	Sample, rising	3

Table 2: SPI settings

Okika DevicesデバイスはSPIモード0および3をサポートします。SPI EPROMのデータシートを参照すればすぐにわかるように、これらは最も一般的にサポートされているモードです。Okika Devicesデバイスでは、これらのモードのいずれかを物理的に選択する必要はありません。デフォルトで両モードがサポートされています。ただし、マスターはモード0またはモード3のいずれかでデータを送受信するように設定する必要があります。

図3および図4はこれらのモードの動作を示しています：

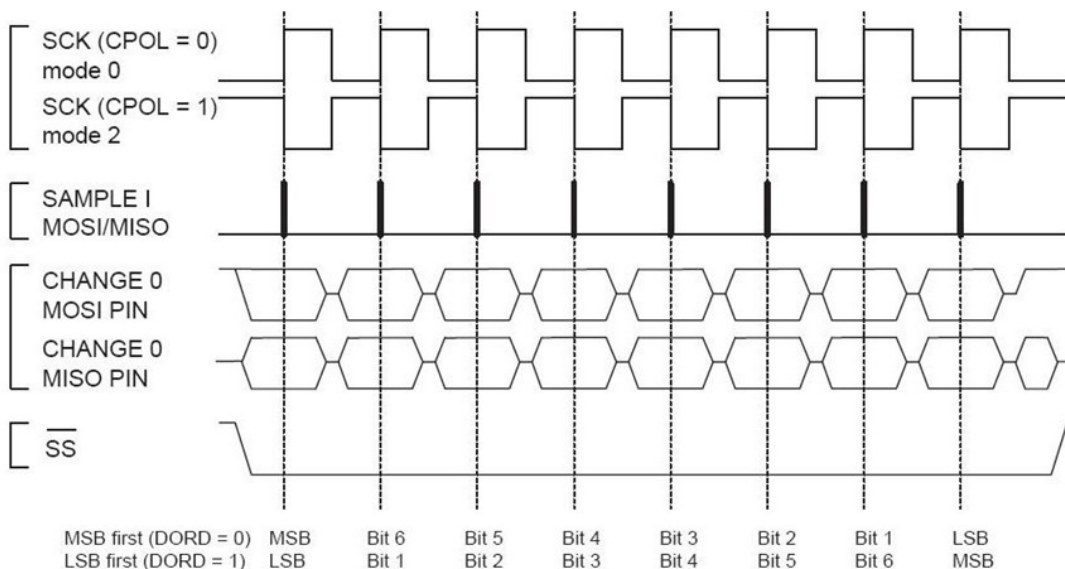


Figure 3: SPI formats with CPHA=0

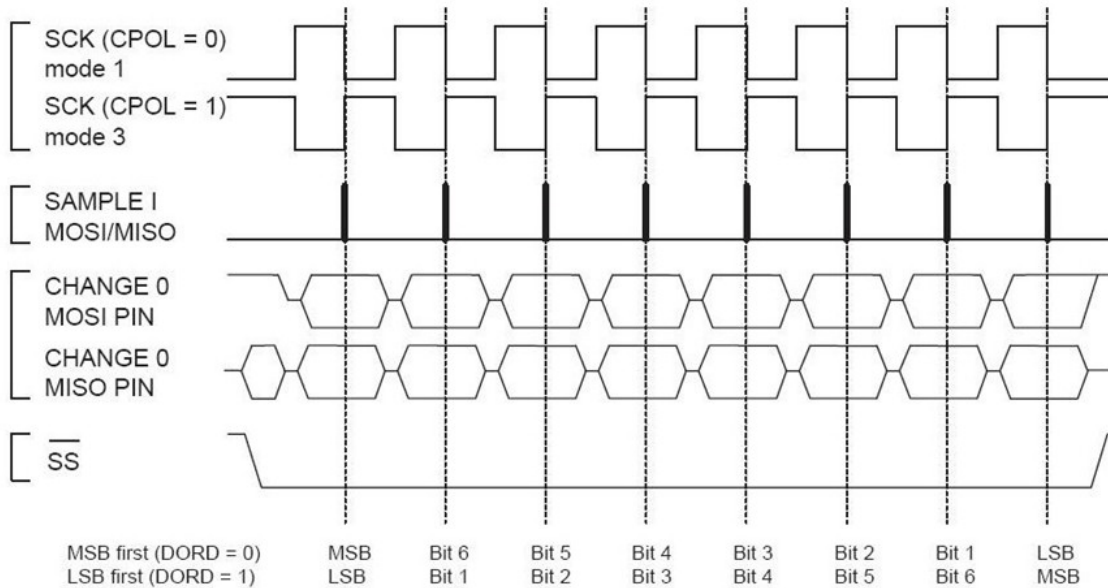


Figure 4: SPI formats with CPHA=1

これらの例では、SPIハードウェアの設定方法に応じて、最上位ビット（MSB）または最下位ビット（LSB）のどちらを最初に送信できることに注意してください。Okika Devicesデバイスでは、常にMSBが最初に送信されます。

マスターとスレーブ間の通信は以下のように行われる：

1. マスターは対象のスレーブデバイスのスレーブセレクトラインをローレベルにします。これにより各デバイスは通信準備状態に入ります。
2. マスターはSPIモードに応じてクロック信号を生成する。マスターとスレーブはクロックサイクルごとに1ビットずつ送信する。
3. 1バイトの送信が完了すると、マスターはスレーブ選択ラインをハイレベルにします。

ここで説明した内容がSPIプロトコルが提供する全てです。ハンドシェイク、確認応答、その他の高度な通信規格は含まれていないため、必要に応じてこのレイヤーの上に実装する必要があります。

Okika Devicesデバイスでは、CRCまたは基本エラーチェックによりマスターからスレーブへの送信データ検証が可能です。またシリアルビットストリームにIDを組み込んでおり、必要に応じて個別のセレクトラインを共通セレクトラインに置き換えることが可能です。デバイスはチップセレクトラインをアサートし、プライマリ設定時にロードされたIDと一致するIDを含むデータを送信することで「選択」されます。

2.3 OKIKAデバイスにおけるデータ転送用SPI波形

以下は、Okika Devicesデバイスへのデータ転送シミュレーションから直接取得した2つの波形である。図5の波形では、モード0 SPIフォーマットが使用されている。SIライン上でスレーブに転送されているバイトは、左から順に2E、2F、30、2Aである。MEMSETUP_SOライン上でスレーブからマスターへ読み戻されるバイトは00,2D,2E,2F（左から右）です。チップセレクト線CS_Bがハイレベルになると、MEMSETUP_SOピンはトライステート状態になることに注意してください。シミュレーションで使用されるこのフォーマットは、上記のモード0フォーマットと一致することが確認できます。ただし、Cs_b信号はデータ最初のビットがセットアップされる前に、公称1クロックサイクル前にローレベルになることに注意してください。これはシミュレーションには影響しません。

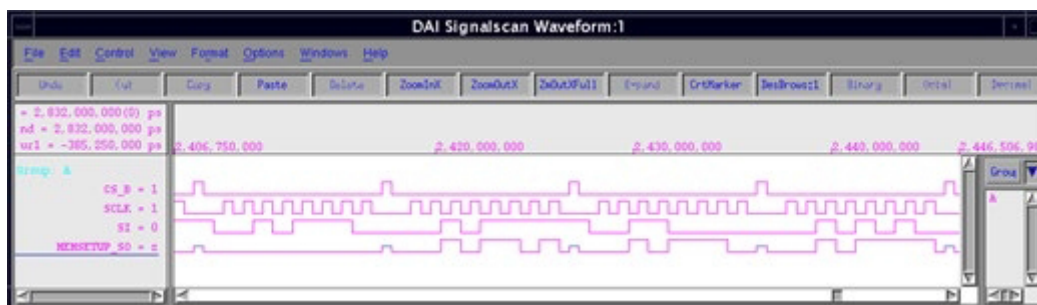


図5: FPAAを用いたモード0データ転送

図6の波形では、モード3 SPIフォーマットが使用されている。SIライン上でスレーブに転送されるバイトは左から順に2E、2F、30、2Aである。MEMSETUP_SOライン上でスレーブからマスターに読み戻されるバイトは左から順に00、2D、2E、2Fである。チップセレクト線CS_Bがハイレベルになると、MEMSETUP_SOピンはトライステート状態になることに注意してください。シミュレーションで使用されるこのフォーマットは、上記のモード3フォーマットと一致することが確認できます。ただし、Cs_b信号はデータ最初のビットがセットアップされる前に、公称1クロックサイクル前にローレベルになることに留意してください。これはシミュレーションには影響しません。

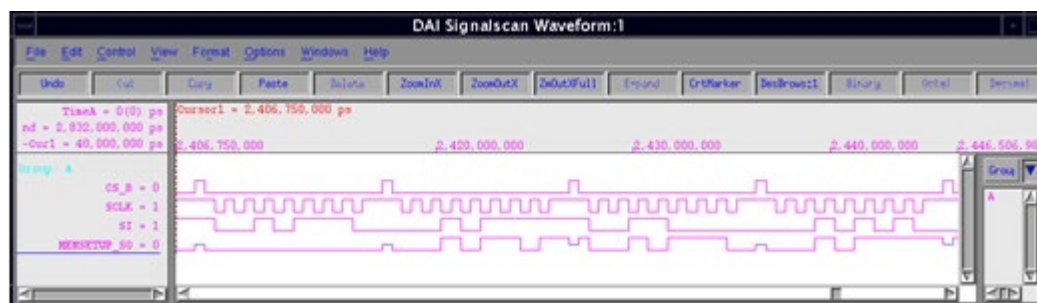


図6: FPAAを用いたモード3データ転送

注記：上記セクションではSPIポート上の双方向通信について説明しています。Okika Devices製FPAAからのデータ読み出しは、データ出力ピンであるMEMSETP_SOを使用します。実際には、FPAAからデータを読み戻す必要性はほとんどなく、有用ですらありません。したがって、次のセクションでは一方通信、すなわちFPAAの設定のみについて説明します。

3.0 複数FPAAのアドレス指定

3.1 FPAAチェーン

本節では、SPIポートから複数のAN231E04 FPAAをアドレス指定する方法を説明します。図7は2つのFPAAチェーンを示しています。各チェーンは最大255個のFPAAで構成され、それぞれが独自の8ビットアドレス（0x00～0xFE）を持ちます。アドレス0xFFは、チェーン内の全FPAAに再構成更新をブロードキャストするために使用されます。

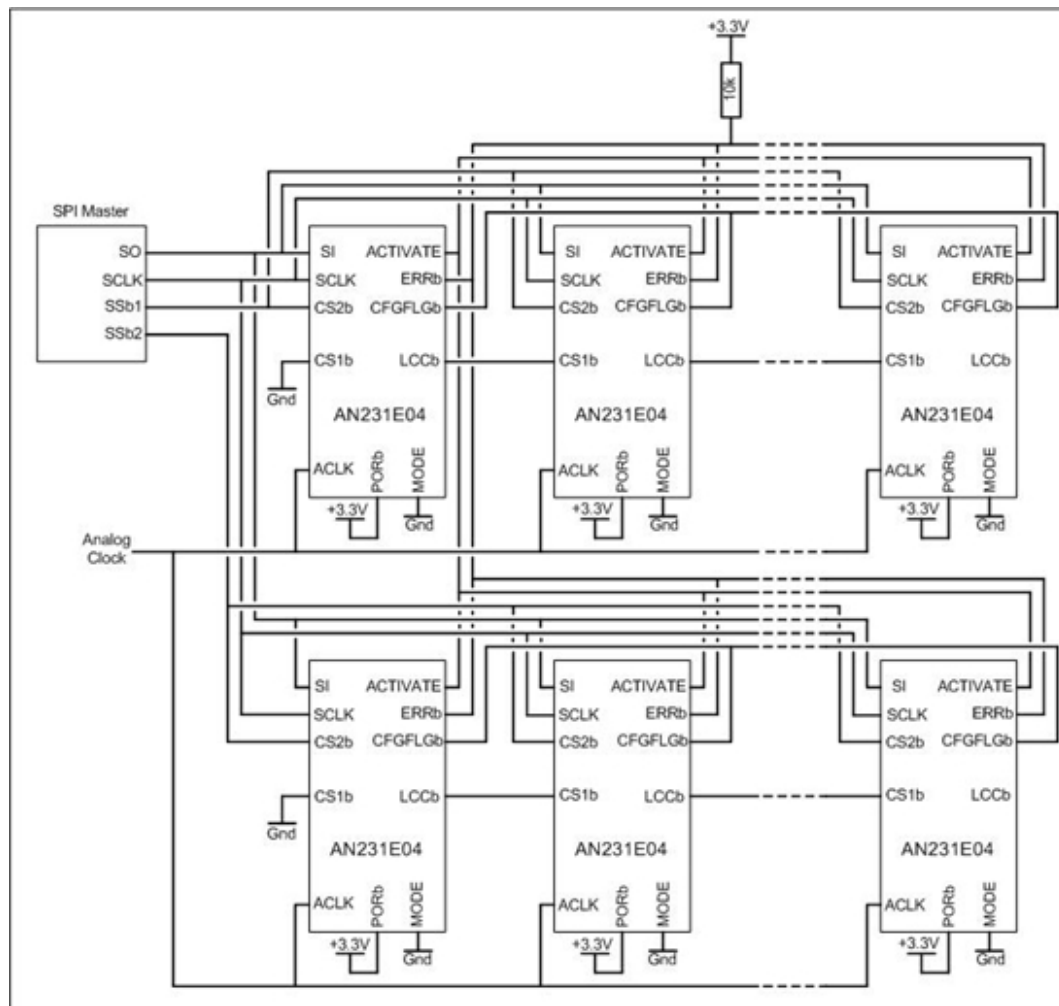


Figure 7: Multiple FPAA's in more than one chain

FPAAアドレスはプライマリ構成中に割り当てられる。チェーンのプライマリ構成時には、適切なセレクトラインがアクティブ化される（例：図7の上位チェーンではSSb1がローレベル駆動される）。その後、N個のプライマリ構成が送信される。ここでNは当該チェーン内のFPAA数（<256）である。プライマリ回路は左側から順に各FPAAにロードされます。各FPAAが構成されると、プライマリ構成ヘッダーで割り当てられたアドレスを採用します。プライマリ構成が完了すると、各FPAAは自身宛ての再構成更新に応答します。

FPAAのチェーンは以下のように特徴付けられる：

1. すべてのCS2bピンに共通する共通選択ライン。
2. 全SCLKピンへの共通クロックライン。
3. 全SIピンへの共通データライン。
4. 各FPAAのLCCbは、次のFPAAのCS1bに接続される。最初のFPAAはCS1bをローに固定し、最後の

FPAAはLCCbをオープンにする。

5. すべてのCFGFLGbピンは互いに接続されています。
6. すべてのACTIVATEピンは互いに接続されています。
7. すべてのERRbピンは（プルアップ抵抗付きで）共通接続されています。

チェーン内の各FPAAが設定されると、そのLCCbピンをローレベルに駆動し、チェーン内の次のFPAAを選択します。次のプライマリ設定は、新たに選択されたFPAAを設定します（既に設定済みのFPAAは、その後のプライマリ設定を無視します）。

ACTIVATEピンはオープンドレインであり、共通接続されている。これは、最後のFPAAがプライマリ設定されるまでこのラインがロー状態を維持することを意味する（各FPAAは設定完了までACTIVATEをローに駆動し、その後解放する）。この時点で全てのFPAAがACTIVATEを解放し、その状態を監視し始めます。これによりラインはハイレベルに引き上げられ、FPAAがハイレベルを検出すると各回路が起動します。このプロセスにより全ての回路が同時に起動することが保証されます。マルチFPAAシステムでは、内部プルアップを設定すべきデバイスは1つだけです。多数のdpASPを使用し負荷が問題となる場合、CFGFLGbおよびACTIVATEネットへの外部プルアップ抵抗を、内部プルアップの代替として適切に配置してください。ACTIVATEおよびCFGFLGの内部プルアップを無効化するには、各FPAAで以下の操作を行います。DynAMx Design Labソフトウェアを使用し、各FPAAについて順次「チップ設定」を選択し、「チップタブ」内でプルアップのチェックを外す。CFGFLGネットも共通接続されている場合、同一の内部プルアップを選択する必要がある。CFGFLGbおよびACTIVATEについても同様の設定を行う。

いずれかのFPAAがプライマリ構成データにエラーを検知した場合、ERRbを19 SCLKサイクル間ローアサートします。これによりチェーン内の全FPAAがリセットされます（ERRbが共通接続されているため）。再構成更新中にFPAAがエラーを検知した場合もERRbをローアサートします。この場合、ERRbパルスは事前設定により短パルスまたは長パルスに設定可能である。短パルスの場合はローカルリセットのみが発生し、長パルスの場合はチェーン内の全FPAAがリセットされる。

初期設定が完了した各FPAAには8ビットのアドレスが割り当てられる。チェーンに送信された再構成更新データは、該当するアドレスを持つFPAAのみが受信する。チェーン内の他のFPAAは全てこの更新データを無視する。ただし、ある再構成データが別のFPAA向けの更新データの開始部のように見える場合がある。この誤ったアドレス指定を回避するため、チェーン内の全FPAAのCFGFLGbピンは共通化される。FPAAが受信した再構成が自身宛てであると認識すると、CFGFLGbをローレベルに駆動し、チェーン内の他の全FPAAに現在の再構成データを無視するよう通知する。

これが複数FPAAのアドレス指定方式である。FPAAは最大255個までチェーン接続され、各々に固有の8ビットアドレスを持つ。より多くのFPAAを使用する場合、複数のチェーンを設定可能であり、各チェーンはSPIMスタの異なるセレクトラインに接続される。

3.2 並列設定

チェーン内の全FPAAに同一の再構成更新を送信する必要がある場合、アドレス0xFFを使用すべきである。複数のチェーンに同一の再構成更新を送信する必要がある場合、再構成対象の全チェーンを選択しデータを送信すればよい。

チェーン内の全FPAAに同一のプライマリ設定を送信する必要がある場合、同一のプライマリ設定をチェーン内の各FPAAに対して複数回送信すればよい。FPAAが常に同一のプライマリ設定を含む必要がある場合、ハードウェア構成を変更してFPAAを並列に設定する方が簡便である（図8）。

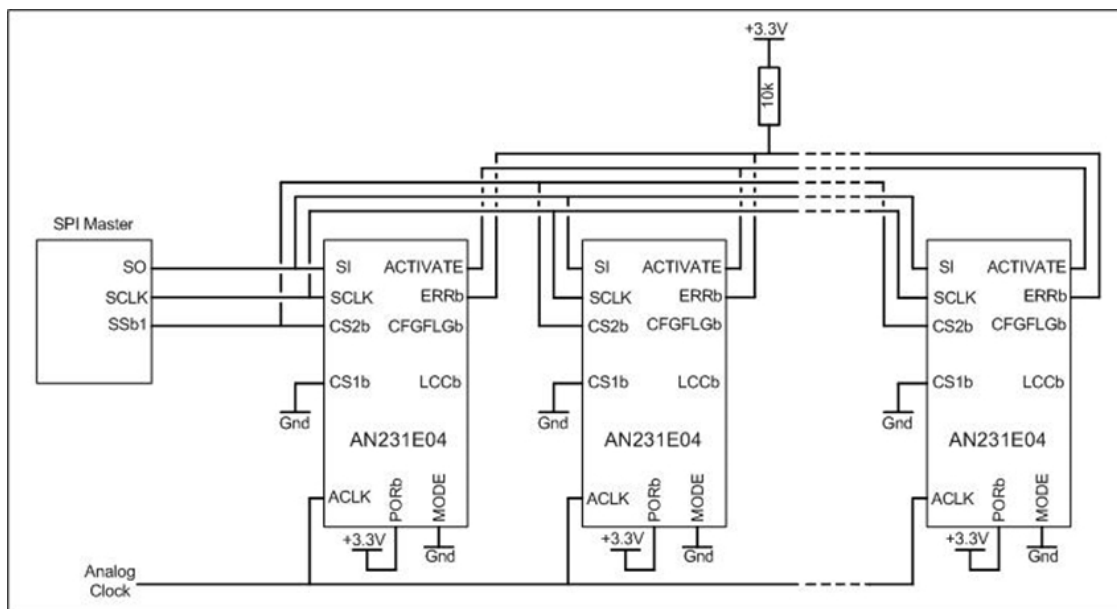


Figure 8: FPAA configured in parallel

この場合、単一プライマリ設定を全FPAAに送信可能である。選択ラインすら不要で、CS2bピンは全てローに固定できる（ただし、SPIポートをシステム内の他デバイスと共有する場合は選択ラインが必要となる）。

ただし、全てのFPAAを並列にプライマリ設定すると、全FPAAが同一アドレスを持つことになるため、再構成更新に対して全FPAAが反応する点に留意が必要である。この構成の利点は、理論上、この方法で設定可能なFPAA数に制限がないことである。

もちろん、シリアルとパラレルの構成を組み合わせることも可能です。図7において、両方のチェーンで同じセレクトラインが使用されていた場合、各チェーン内のFPAAが異なる回路を持っていたとしても、両方のチェーンは同じ回路で構成されることになります。

注記:

ACTIVATE: 上記で説明したように、FPAA内の回路を同時に起動させるためにはACTIVATEピンを共通化する必要があります。特定のFPAAまたはFPAAグループを同時に起動させる必要がない場合、負荷を軽減するために個別のACTIVATEラインを使用すべきです。例えば図7では、チェーン内の全回路が連動する一方、チェーン同士は独立している場合があります。この場合、各チェーンに個別のACTIVATEラインを設けることが最適である。

ERRb: 上記と同様の論理がERRbにも適用される。異なる回路グループまたはドメインには個別のERRbラインを設けるべきである。これにより負荷低減に寄与する。また、複数のFPAAに対してはプルアップ抵抗の値を低く設定する必要がある場合がある点に留意すること（デバイスマニュアル参照）。

4 その他の考慮事項

4.1 クロックバッファリング

多数のFPAAを使用する場合、クロックライン、場合によってはデータラインのバッファリングが必要となる可能性があります。クロックラインをバッファリングする理由は、負荷が大きくなるとエッジが遅延する可能性があるためです。また、開放端の長いクロックラインでは反射が発生し、クロック信号に二重エッジが生じる可能性があります。これらの状態はいずれも、同一データビットの多重ラッチングを引き起こす恐れがあります。過去において、Okika Devices社は高速かつ低スキュー特性を備えたTexas Instruments製CDCLVC11xxクロックバッファを採用してきました。例えば図7では、2つのチェーン間でSCLKラインを分割するためにCDCLCV1102を使用できます。ACLKラインについても同様です。チェーンが非常に長い場合、ツリー構造で接続された複数のバッファが必要となる可能性があります。

4.2 レイアウト

多数のFPAAを扱うレイアウトについては注意が必要です。最も重要なのはアナログ領域とデジタル領域を分離することです。図9は8個のFPAAを2列×4個で配置したチェーンを示しています。

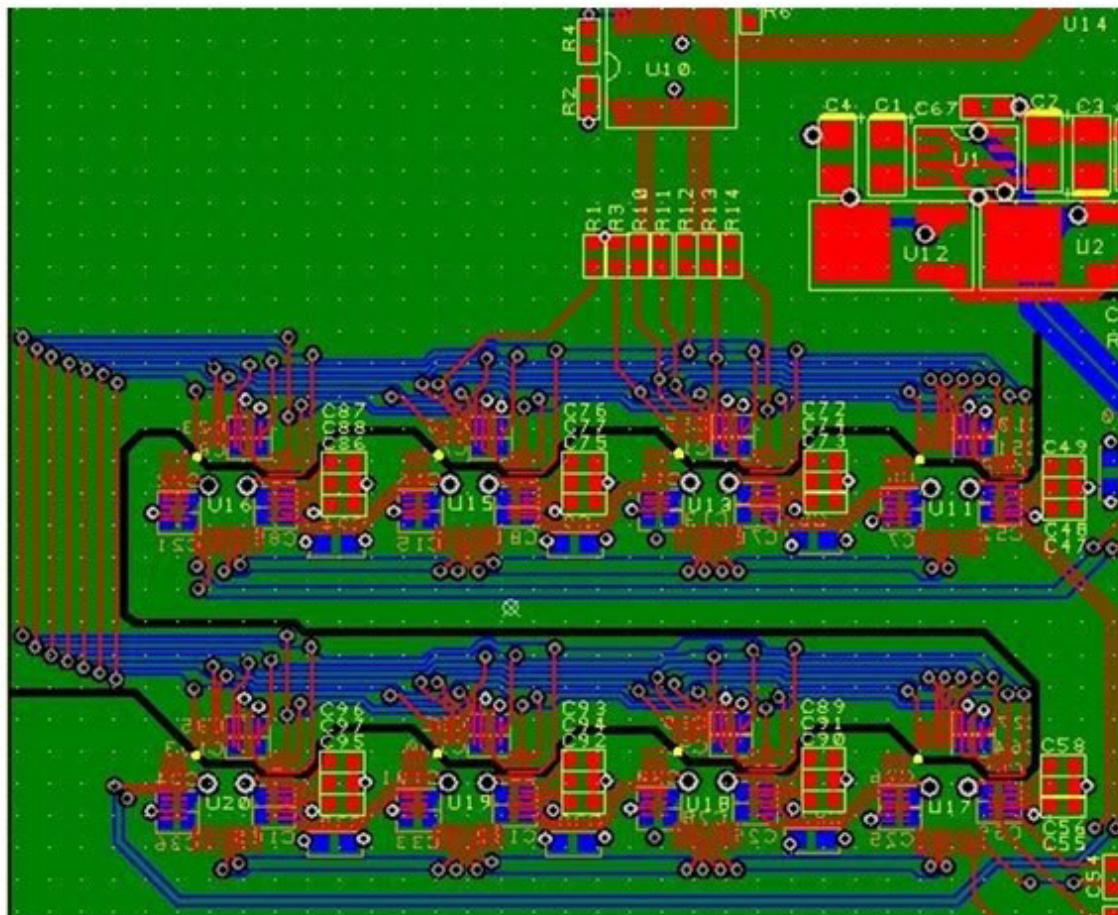


Figure 9: FPAA chain layout

FPAAは、デジタルピンを上側、アナログピンを下側に配置してレイアウトされています。これらはバスとして一緒に配線され、デジタル信号は上部のコントローラに接続され、アナログ信号は右側に取り出されています。緑色はアナログとデジタルで分割されたグランドプレーンを示します。図9は、デジタル信号がデジタルグランド領域上に、アナログ信号がアナロググランド領域上に配置されている様子を示しています。この設計では、電源プレーンも同様の方法で分割されていることに注意してください。分割された領域は通常、基板全体を横切って電源入力点まで延びています。もちろん、特にアナログ領域では十分なデカップリングが必要です。

クロックライン間、およびクロックラインとデータライン間の分離も重要です。特に長距離配線の場合に顕著です。図9ではデジタルラインが均等に配置されているように見えますが、Okika Devicesではクロックラインとデータラインを、クロストークの影響を受けにくい信号（例：PORb、CFGFLGb、ACTIVATE、LCCbなど）と交互に配置することを推奨しています。

例：PORb、CFGFLGb、ACTIVATE、LCCbなど...

詳細は www.okikadevices.com をご覧いただくか、Okika Devices テクニカルサポートまで
メールでお問い合わせください。 contact@okikadevices.com