


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。

QI151021-11.0.0

Qsys インタコネクは、Avalon® インタフェースを使用するコンポーネントを接続するための高帯域幅の構造です。この章では、Qsys インタコネクを説明します。インタコネクは、Qsys システムを実装するにはインタコネクコンポーネントを挿入するアルゴリズム変換を使用しています。また、この章では、インタコネクを実装する Qsys インタコネク・コンポーネントについて簡単に説明します。すべての Qsys インタコネク・コンポーネントは、独自のデザインで使用することが可能です。Qsys インタコネクは、以下の Avalon インタフェースのタイプを接続します。

- Avalon-ST—Avalon-ST ソースおよびストリーム単方向データそのシンクを接続します。
- Avalon-MM—Avalon-MM マスタをリードおよびライトのコマンドと使用して通信するスレーブを接続します。
- Avalon Tri-state Conduits (Avalon-TC)— トライステート情報の三信号のエンコーディングを使用して PCB 上のトライステート・デバイスに FPGA でのトライステート・コンジット・コントローラを接続します。
- Avalon Interrupts— サービスするコンポーネントの割込み送信側と割込みレシーバを接続します。
- Avalon Clocks— クロック・ソースとクロック・シンクを接続します。
- Avalon Resets— リセット・ソースとリセット・シンクを接続します。
- Avalon Conduits— ポイント・ツー・ポイントのコンジット・インタフェースを接続します。他の Avalon インタフェースのカテゴリのいずれに適合しない信号の任意のコレクションを定義するためにコンジット・インタフェースのタイプを使用することができます。

 Avalon インタフェースについて詳しくは、[Avalon Interface Specifications](#) を参照してください。

Avalon-ST インタフェースの場合、Qsys はのポイント・ツー・ポイント接続の作成に柔軟性を可能にするアダプタを提供しています。例えば、Avalon-ST データ・フォーマット・アダプタは、異なる幅のストリーミング・インターフェースを接続することができます。

Avalon-MM インタフェースの場合、Qsys インタコネクの実装は、ネットワーク・オンチップ・アーキテクチャをベースにしています。マスタとスレーブ間のトランザクションは、パケットにカプセル化し、マスタとスレーブの間でパケットを伝送するネットワーク上で送信されます。マスタのコマンド・ネットワークは、マスタ・インタフェースからスレーブ・インタフェースにリードおよびライトのコマンド・パケットを転送します。スレーブの応答のネットワークでは、スレーブ・インタフェースからマスタ・インタフェースへのリード応答パケットを転送します。

© 2010 Altera Corporation. All rights reserved. ALTERA, ARRIA, CYCLONE, HARDCOPY, MAX, MEGACORE, NIOS, QUARTUS and STRATIX are Reg. U.S. Pat. & Tm. Off. and/or trademarks of Altera Corporation in the U.S. and other countries. All other trademarks and service marks are the property of their respective holders as described at [www.altera.com/common/legal.html](http://www.altera.com/common/legal.html). 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. 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.



この章では、以下の項で構成されています。

- 7-2 ページの「Avalon-MM インタフェース・コンポーネント」
- 7-19 ページの「Avalon-ST インタフェース」
- 7-22 ページの「トライ・ステートのコンジット・コンポーネント」
- 7-27 ページの「割り込みインタフェース」
- 7-29 ページの「クロック・インタフェース」
- 7-29 ページの「リセット・インタフェース」
- 7-30 ページの「Avalon コンジット」

## Avalon-MM インタフェース・コンポーネント

メモリ・マップされたインタフェースのための Qsys インタコネクは、Avalon-MM マスタとスレーブ・インタフェースを接続します。それは、以下の項目をサポートしています。

- 任意のマスタとスレーブのコンポーネントの数。マスタとスレーブ間の関係は一對一、一對多、多対一、または多対多にすることができます。
- 異なるデータ幅のマスタとスレーブ。
- 異なるクロック・ドメインで動作しているコンポーネント。
- 異なるインタフェース・プロパティと信号付きのコンポーネント。Qsys は、以下のタイプの差のあるインタフェースが接続できるように、コンポーネント・インタフェースを適応させる場合もあります。
  - アクティブ High およびアクティブ Low 信号を使用するインタフェース
  - 異なるバースト特性付きのインタフェース
  - 異なるレイテンシ付きのインタフェース
  - 異なるポート・シグネチャ付きのインタフェース

図 7-1 には、複数のマスタとの Avalon-MM システムの Qsys インタコネクの簡略化した表現です。この図に示すように、マスタとスレーブの接続の基礎となる実装は、ネットワーク・トポロジを使用しています。Qsys システムを生成すると、Qsys により、ネットワークのトポロジとの **Connections** カラムで作成されたポイント・ツー・ポイント接続を置き換えるには、指定されたインタコネクを実装しています。

図 7-1. Qsys インタコネク — システムの例

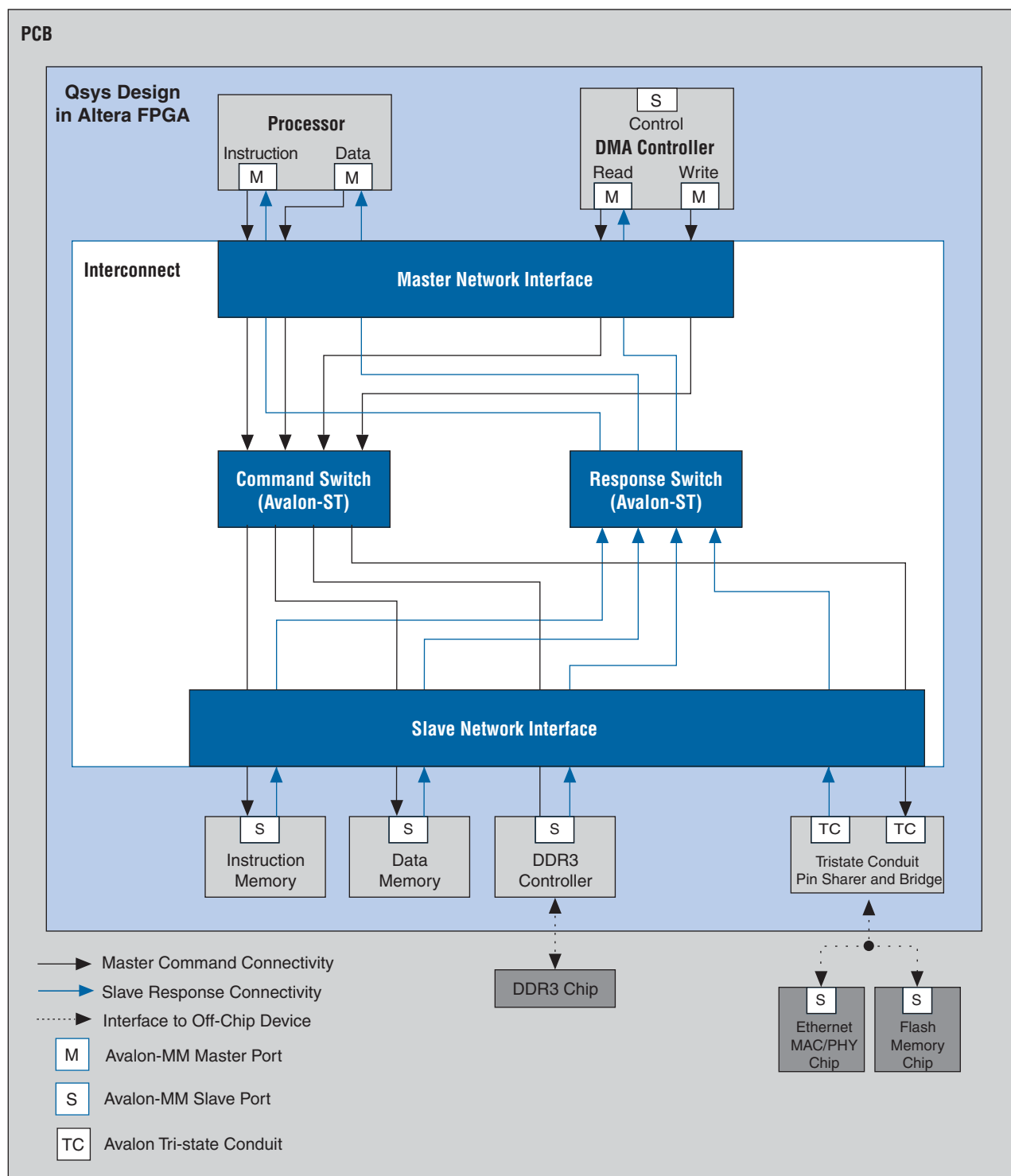


図 7-2 に、Avalon-MM マスタ・コマンドと Avalon-MM スレーブの応答をカプセル化する Qsys パケットのフォーマットを示します。

図 7-2. Qsys パケットのフォーマット

Address	Transaction type	Data	Byte enable	Source ID	Destination ID	Byte count	Burstwrap	Protection
---------	------------------	------	-------------	-----------	----------------	------------	-----------	------------

表 7-1 には、Qsys パケットのフィールドを説明します。

表 7-1. Qsys パケットのフォーマット (その 1)

フィールド	説明
Address	現在のサイクルの最下位バイトのバイト・アドレスを指定します。
Transaction_type	トランザクション・タイプを示します。表 7-2 には、5 種類のトランザクション・タイプをリストします。
Data	コマンド・パケットの場合、書き込まれるデータを運びます。リード応答パケットの場合、読み出されたデータを運びます。
Byteenable	有効なデータのシンボルを指定します。以下の値は、32 ビット・データを転送する Avalon-MM マスタとスレーブには有効です。 <ul style="list-style-type: none"> <li>■ 1111                   フル 32 ビットを書き込み</li> <li>■ 0011                   最下の 2 バイトを書き込み</li> <li>■ 1100                   最上の 2 バイトを書き込み</li> <li>■ 0001                   バイト 0 のみ書き込み</li> <li>■ 0010                   バイト 1 のみ書き込み</li> <li>■ 0100                   バイト 2 のみ書き込み</li> <li>■ 1000                   バイト 3 のみ書き込み</li> </ul>
Source_ID	コマンドまたは反応を開始したマスタまたはスレーブの ID。
Destination_ID	コマンドまたは反応が向けられる ID マスタまたはスレーブ。

表 7-1. Qsys パケットのフォーマット (その 2)

フィールド	説明
Burstwrap	<p>バースト・ラップ値は現在のバーストのラップの動作を指定します。バースト・ラップ値はフォーム <math>2^{m-1}</math> があります。以下のタイプは定義されます。</p> <ul style="list-style-type: none"> <li>■ 変数ラップ-変数ラップ・バーストは 2 値の任意の整数乗でラップすることができます。バーストがラップの境界に達する時には、それは、アドレッシングのために、低いオーダ・ビットだけが使われるように、前のバーストの境界をラップします。例えば、32 バイトのバースト・ラップの境界および 20 バイトのバースト・サイズのあるアドレス 0x1C から開始するバーストは、アドレス 0x1C、0x0、0x4、0x8、および 0xC に書き込む場合もあります。&lt;math&gt;m&lt;/math&gt;サイズのバースト・ラップの境界の場合、Burstwrap = &lt;math&gt;m-1&lt;/math&gt; であり、この場合の Burstwrap = <math>(32-1) = 31</math> では、&lt;math&gt;m&lt;/math&gt;は <math>2^5-1</math> になります。</li> <li>■ シーケンシャル-シーケンシャル・バーストは、バースト内の転送ごとにアドレスがインクリメントされます。シーケンシャル・バーストの場合、Burstwrap 欄がすべて 1 に設定されています。例えば、6 ビットの Burstwrap 欄で、シーケンシャル・バーストの値は 6'b111111 または 63 (すなわち <math>2^6-1</math> である)。</li> </ul> <p>Quartus II ソフトウェアのバージョン 11.0 では、適応ロジックが宣言されたマスタのバースト特性に従って、バースト・ラップのフィールドのハードワイヤードの値を設定します。例えば、破裂シーケンシャル宣言マスタのために、バースト・ラップ欄がすべて 1 に設定されています。同様に、ライン・ラップのバーストを宣言するマスタは、適切な定数値に設定されるバースト・ラップ欄があります。</p>
Protection	<p>0 の場合、パケットは通常のアクセスがあります。1 の場合、パケットは、特権アクセスを持っています。Avalon-MM インタフェースで、Avalon-MM マスタは、直接この欄を特権アクセスの信号にマップされ、オンチップ・メモリの ROM インスタンスに書き込むことができます。</p>

表 7-2 に、トランザクション・タイプのエンコーディングを示します。

表 7-2. トランザクション・タイプ

ビット	名称	定義
0	PKT_TRANS_READ	アサートされると、リード・トランザクションを示します。
1	PKT_TRANS_COMPRESSED_READ	リード・トランザクションで、リード・コマンドは、1 サイクルで表現することを指定します。つまり、すべての byteenables がサイクルごとにアサートされているかどうかを指定します。
2	PKT_TRANS_WRITE:	アサートされると、ライト・トランザクションを示します。
3	PKT_TRANS_POSTED	アサートされると、応答は必要ありません。
4	PKT_TRANS_LOCK	アサートされると、アービトレーションがロックされていることを示します。パケットを書き込むに適用されます。

Qsys パケット・フォーマットの欄が使用されるリソースを最小限に抑えるための可変長です。しかし、大部分のデザイン内のコンポーネント、例えば 32 ビットおよび 64 ビットのシングルのデータ幅のある単一のコンポーネントの場合に、Qsys は 64 ビット転送に対応するために幅のアダプタを挿入します。

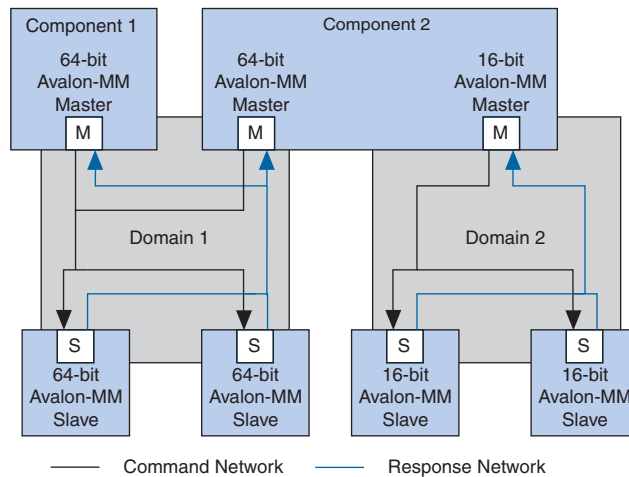
## コンポーネントのインタコネク・ドメイン

接続される Avalon-MM マスタおよびスレーブのグループは、インタコネク・ドメインと呼ばれています。単一のインタコネク・ドメイン内のコンポーネントは、同じパケットのフォーマットを共有します。以下の二つの例は、この点を示しています。

## 2つの別々のドメインの使用

図 7-3 は、2つの別々のドメインの使用方法を示しています。最初のドメインは2つの64ビット・スレーブに接続している2つの64ビット・マスターが含まれています。2番目のドメインは2つの64ビット・スレーブに接続している1つの16ビット・マスターが含まれています。Domain 1とDomain 2のインタフェースが接続を共有していないため、Qsys 2つの別々のドメインのためにパケット・フォーマットを最適化することができます。この例では、最初のドメインは、64ビット・データ幅を使用し、2つ目のドメインは、16ビットのデータを使用しています。

図 7-3. 2つのドメイン

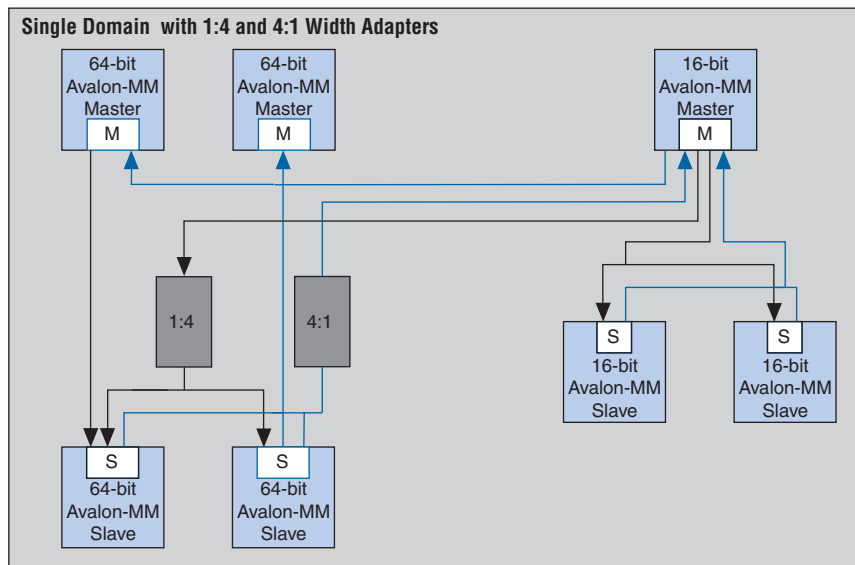


### 幅適応と1つのドメインの使用

図 7-4 に、2 つの 64 ビット・スレーブをアクセスする 2 つの 64 ビット・マスタを含む Qsys システムを示しています。また、2 つの 16 ビットスレーブ及び1つの 64 ビットスレーブをアクセスする一つの 16 ビットのマスタを含めています。マスタのうちの一つがすべてのスレーブに接続するため、Qsys には 2 つのパケット・フォーマットのあるシングル・ドメインを作成します：64 ビットのデータおよび 16 ビットのデータ付き。

幅のアダプタは 16 ビットのマスタおよび 64 ビットのスレーブ間のアクセスを管理します。

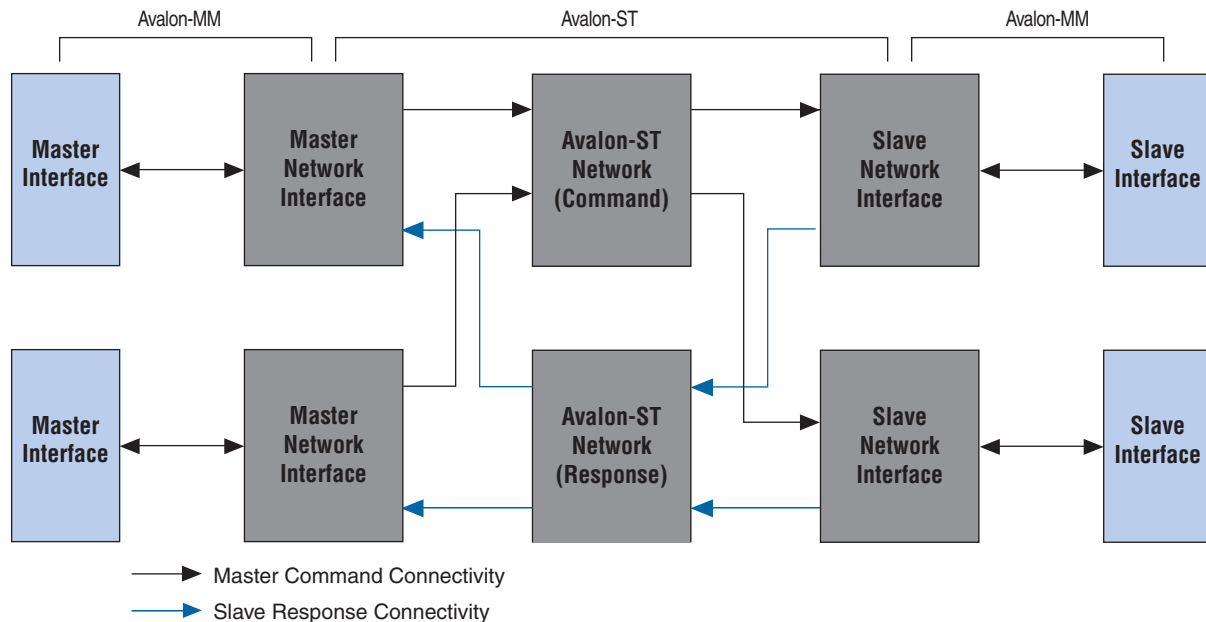
図 7-4. 1:4 および 1:4 の幅適応の 1 つのドメイン



### Qsys の変換

図 7-5 に、Avalon-MM マスタとスレーブ・コンポーネント付きの Qsys システムを生成するときに発生する変形をより詳細に示しています。この図に示すように、Avalon-MM マスタとスレーブ・コンポーネントは、Avalon-ST パケットでトランザクションをカプセル化するネットワーク・インターフェース・モジュールに接続します。Avalon-MM インタフェースは、Avalon-MM プロトコルに従って、*Avalon Interface Specifications* で定義されるようにリードとライト信号および転送を使用して、カプセル化についての情報、またはパケットおよび単に操作を輸送しているレイヤーの機能を全然ありません。

図 7-5. Avalon-MM から Avalon-ST への Qsys の変換



## マスタ・コマンドおよびスレーブの応答ネットワーク

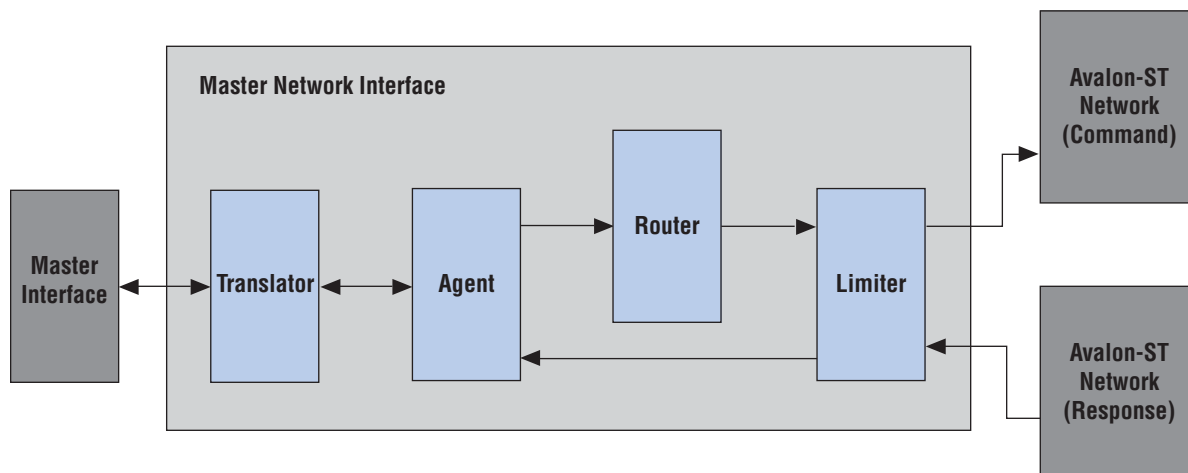
多くの Qsys コンポーネントは、Qsys インタコネクおよび図 7-5 での Avalon-ST Network (Command) と Avalon-ST Network (Response) のブロックで表されるネットワーク・インタフェースを実装します。これらの Qsys すべてのコンポーネントは、アルテラが提供され、Qsys 内の利用可能なコンポーネント・ライブラリに含まれています。これらはデザインでのスタンド・アロンを使用するために用意されています。例えば、デザインのクロック周波数を増加させ、データ・パス・ストリーミング接続はパイプラインに Avalon-ST パイプライン・ステージを含めることができます。

以下のセクションでは、Avalon-ST マスタ・コマンド、Avalon-ST スレーブ応答のネットワークおよび以下のコンポーネントを含む一部となっているコンポーネントについて説明します。

- Merlin Master Translator
- Merlin Master Agent
- Merlin Router
- Merlin Traffic Limiter
- Merlin Slave Translator
- Merlin Slave Agent

図 7-6 に、Merlin Master Translator、Agent、Router、および Limiter を示す Master コマンドのネットワークのためのブロック図を示します。

図 7-6. マスタ・コマンド・ネットワークでの Qsys コンポーネント



### Merlin Master Translator

Avalon-MM マスタ・コンポーネントの Merlin Master Translator のインターフェースです。それは、Qsys ネットワークで使用される単純な図への Avalon-MM マスタ・インタフェースを変換します。それは以下の機能を実行します。

- アクティブ High 信号にアクティブ Low 信号を変換する機能
- Avalon-MM マスタが無効なデータの読み出しを防ぐためにウェイト・ステートを挿入する機能
- ワードやシンボル・アドレスを変換する機能
- ワードやシンボルのバースト・カウントを変換する機能
- バースト・カウントのタイミングとシーケンスを処理する機能
- 不必要なアドレス・ビットを削除する機能

### Merlin Master Agent

Agent は、Qsys コマンド・パケットに Avalon-MM マスタのトランザクションを変換し、Avalon-MM 応答に Qsys Avalon-MM スレーブの応答パケットを変換します。

### Merlin Router

Router は、マスタからスレーブにコマンドパケットをルーティングし、スレーブからマスタに応答パケットをルーティングします。マスタ・コマンド・パケットの場合、Destination\_ID および Avalon-ST のチャンネルを設定するためには、Router は Avalon-MM アドレスを使用しています。スレーブの応答パケットの場合、Avalon-ST のチャンネルを設定するには、Router は、Destination\_ID を使用しています。ディマルチプレクサは、正しい宛先にルーティングするパケットには、Avalon-ST のチャンネルを使用しています。

## Merlin Traffic Limiter

Traffic Limiter は、応答が順序で到着を保証します。応答がすでに発行されているコマンドの応答と競合する可能性がある場合は送信されないように任意のコマンドは阻止されます。イン・オーダーの応答を保証することによって、Traffic Limiter は、応答のネットワークを簡素化します。

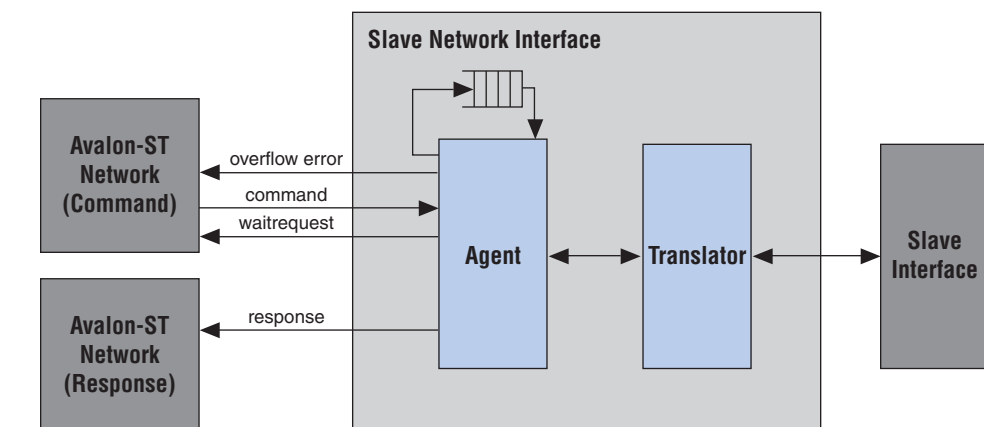
## Merlin Slave Translator

図 7-7 に示すように、Merlin Slave Translator は Avalon-MM スレーブ・コンポーネントにインタフェースします。それは、Qsys ネットワークで使用される簡略された図への Avalon-MM スレーブ・インタフェースを変換します。Avalon-MM Merlin Slave Translator は以下の機能を実行します。

- begintransfer、beginbursttransfer、および writebyteenable 信号をドライブする
- 有効なデータを識別するための readdatavalid 信号を使用してスレーブを使用する固定タイミングおよびスレーブを動作する Avalon-MM スレーブをサポートする
- Avalon-ST スレーブの応答のネットワークで使用される図に read、write、および chipselect 信号を変換する
- アクティブ High の信号へのアクティブ Low の信号を変換する
- 単語やシンボル・アドレスとバースト・カウントを変換する
- バースト・カウントのタイミングとシーケンスを処理する
- 不必要なアドレスビットを削除する

図 7-7 に、スレーブの応答ネットワークを構成する Qsys コンポーネントを示しています。

図 7-7. スレーブの応答ネットワークの Qsys コンポーネント



## Merlin Slave Agent

Slave Agent は、Avalon インタフェースにコマンド・パケットが受け入れられ、結果トランザクションを発行します。パイプライン化されたスレーブの場合、Avalon-ST FIFO は、保留中のトランザクションに関する情報が格納されます。この FIFO のサイズは、スレーブ・コンポーネントを作成するときに指定した保留中の応答の最大数です。

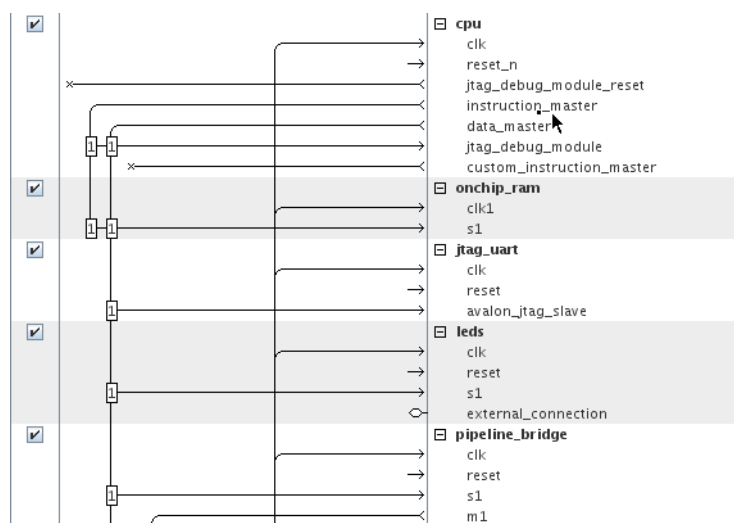
また、Merlin Slave Agent、スレーブ・コンポーネントが waitrequest 信号が含まれている場合、FIFO が完全であるときに、Avalon-MM マスタ・コマンドがバックプレッシャされます。

## アービトレーション

複数のマスタがスレーブへのアクセスに競合する場合、自動的にアービトレーション・ロジックを挿入し、公正ベース、ラウンド・ロビンの順序でアクセスを許可します。公平ベースのアービトレーション手法では、各マスタはスレーブに対する転送株のシェアの値があります。一つのシェアは、1回の転送を実行する権限を表します。デフォルトのアービトレーション手法は、すべての要求マスタに等しく、シーケンシャル・アクセスを付与するイコール・シェア・ラウンド・ロビンです。特定のスレーブにアクセスするマスタにアービトレーション・シェアアービトレーション・シェアの相対的な数を指定することにより、重み付けラウンド・ロビンにアービトレーションの手法を変更することができます。アービトレーションの設定を表示するには、View メニューの **Show Arbitration** をクリックします。

図 7-8 に、アービトレーション・シェアアービトレーション・シェアを示します。

図 7-8. System Contents タブでのアービトレーションの設定

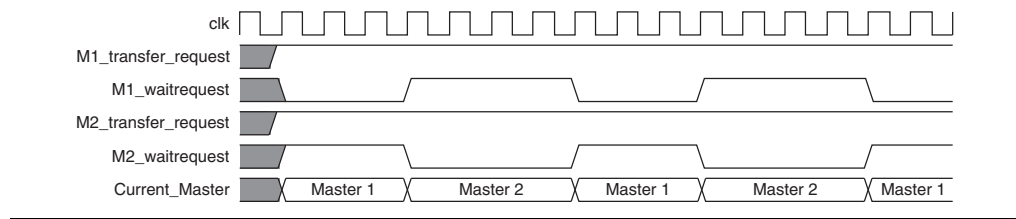


### Arbitration の例

図 7-9 に、バック・ツー・バック転送を実行するために、単一の Avalon-MM スレーブに継続的にアクセスする 2つの Avalon-MM マスタのタイミングを示します。Master 1 には、3つのシェアがあり、Master 2 は4つのシェアがあります。Merlin Arbiter は、Master 1 のアクセスには3つの転送でを許可し、Master 2 に4つの転送を許可します。

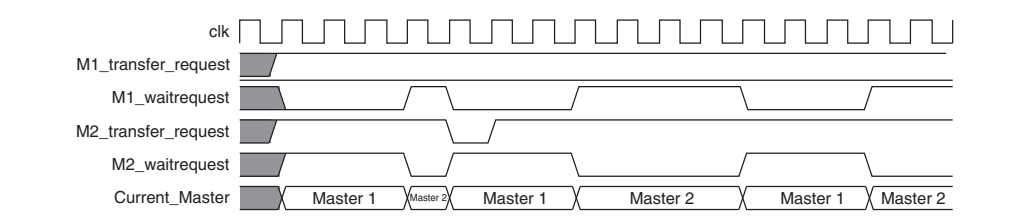
このサイクルは無期限に繰り返します。

図 7-9. 2つのマスタから継続的な転送の要求のアービトレーション



マスタがシェアを使い果たす前に転送の要求を停止した場合、[図 7-10](#) に説明されるとおり、その残留シェアのうちのすべて、および別の要求マスタへのアクセスを許可する Merlin Arbiter を喪失します。1 回の転送を完了した後、Master 2 は、1 クロック・サイクルの要求を停止します。その結果、Arbiter は、Master 1 にアクセスを許可します (3 つのシェアを取得する)。

図 7-10. 転送の要求での 2 つのマスタのアービトレーション

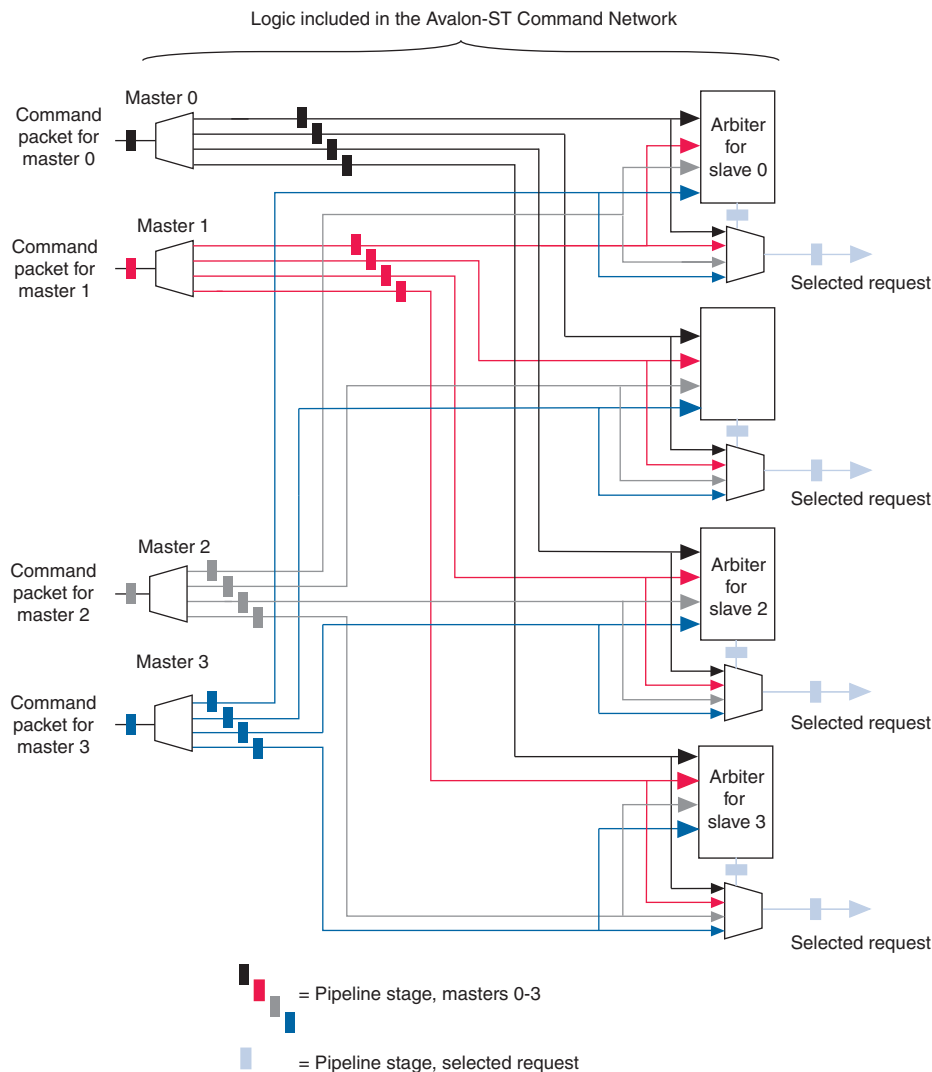


## Merlin Arbiter


Merlin Arbiter への入力は、特定のスレーブにアクセスを要求するすべてのマスタのための Avalon-MM マスタのコマンド・パッケージです。Arbiter は、選択したマスタのためのチャンネル番号を出力します。このチャンネル番号は、スレーブ・デバイスを選択するマルチプレクサの出力を制御します。[図 7-11](#) に、このロジックを示しています。

図 7-11 には、4 の Avalon-MM マスタは、4 つの Avalon-MM スレーブに接続します。各サイクルでは、各 Avalon-MM スレーブの前に配置する Arbiter は、の Avalon-MM マスタ要求の中から選択します。

図 7-11. アービトレーション・ロジック



**Qsys Project Settings** タブのゼロより大きい **Limit interconnect pipeline stages to** パラメータを指定した場合、Arbiter の出力が登録されています。この出力を登録すると、マスタとファブリック間の組み合わせロジックの量が低減され、システムの  $f_{MAX}$  を増やします。

 **Limit interconnect pipeline stages to** パラメータについて詳しくは、「Quartus II ハンドブック Vol 1」の「Creating a System with Qsys」の項の「Project Settings」のセクションを参照してください。

## インタコネク・パイプライン

**Project Settings** タブの 0 より大きい値に **Limit interconnect pipeline stages to** パラメータを設定した場合、デザインを生成するときに、Qsys は Avalon-ST のパイプライン・ステージを自動的に挿入します。パイプライン・ステージは、組み合わせロジックの容量を低下することによって、デザインの  $f_{MAX}$  を増加させます。コストは、追加のレイテンシとロジックです。

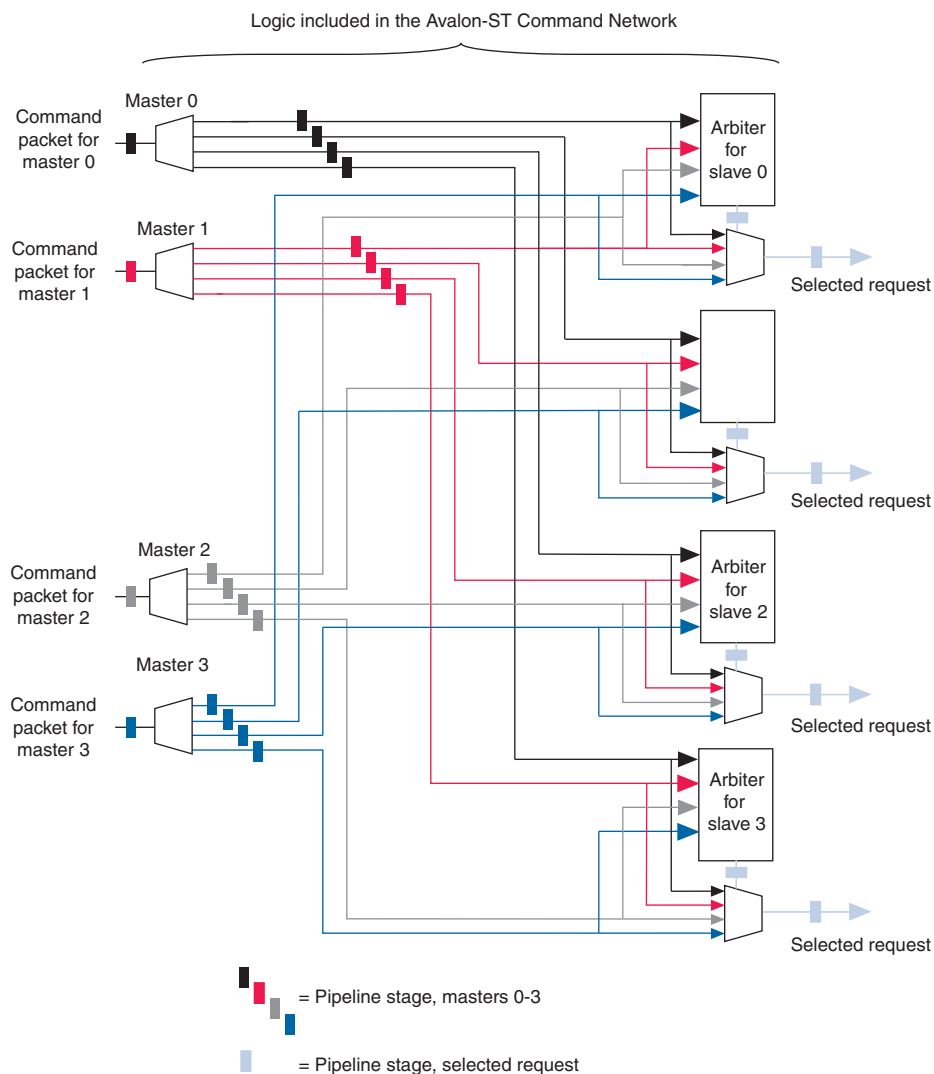
図 7-12 に、以下の場所に Qsys によって挿入された 4 つの潜在的なパイプライン・ステージへの配置を示しています。

- デマルチプレクサへの入力前に
- マルチプレクサの出力で
- アービタとマルチプレクサの間
- デマルチプレクサの出力で



パイプライン・ステージの挿入は、特定のインタコネク・コンポーネントの存在に依存しています。例えば、1 つのスレーブ・システムでは、マルチプレクサが存在しないので、マルチプレクサのパイプラインは発生しません。1 つのスレーブ・システムにシングル・マスタの極端な場合では、**Limit interconnect pipeline stages to** の値は関係なく、パイプラインが発生しません。

図 7-12. アービトレーション・ロジックのパイプラインの配置



## 追加の Qsys インタコネク・コンポーネント

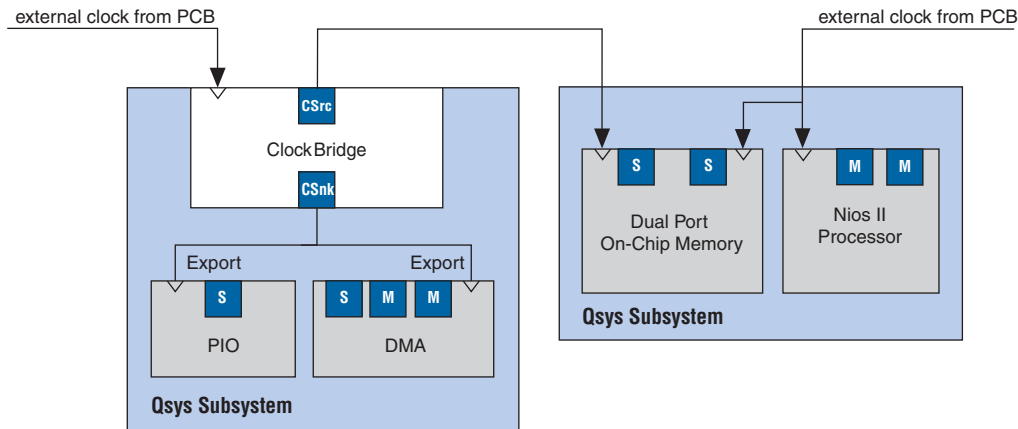
以下のセクションでは、Qsys インタコネクで使用される追加コンポーネントについて説明します。すべてのコンポーネントは、デザインで使用するためのコンポーネント・ライブラリにあります。

- 7-16 ページの「Clock Bridge」
- 7-16 ページの「Avalon-MM Clock Crossing Bridge」
- 7-16 ページの「Avalon-MM Pipeline Bridge」
- 7-17 ページの「Merlin Width Adapter」

## Clock Bridge


Clock Bridge は、Qsys サブシステム間のクロックを配線することができます。複数の Qsys サブシステムの入力クロックに 1 つのクロック・ソースを接続するために、このブリッジを使用することができます。図 7-13 に、このブリッジの使用を示しています。

図 7-13. Clock Bridge



## Avalon-MM Clock Crossing Bridge

Avalon-MM Clock Crossing Bridge は、非同期クロック・ドメイン間の Avalon-MM コマンドと応答を転送します。これはクロック・クロッシング・ロジックを実装するために、非同期 FIFO を使用しています。Avalon-MM Clock Crossing Bridge は、マスタとスレーブのクロック・ドメインの両方でコマンドと応答の FIFO の容量を制御するパラメータを含むパラメータの数があります。フライト中のリードの数は、応答の FIFO の容量を超える場合は、Avalon-MM Clock Crossing Bridge はリードの送信を停止します。高性能アプリケーションのスループットを維持するためには、デフォルトの最小の容量から応答の FIFO の容量を 2 倍の最大のバースト・サイズで増大させてください。

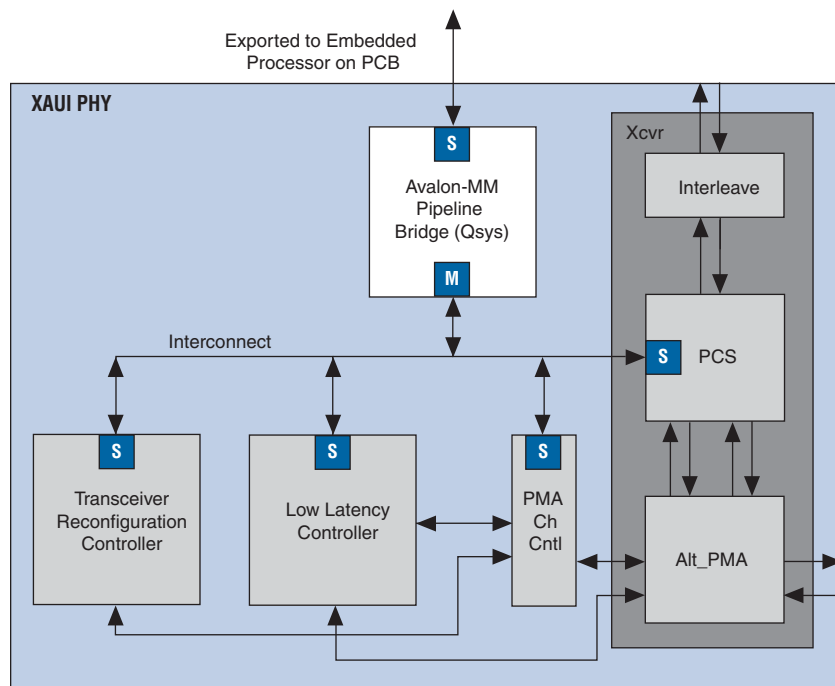
 Avalon-MM Clock Crossing Bridge のコアは、Qsys インタコネクで動作するように実装されています。レガシー Avalon-MM Clock Crossing Bridge のコアは、SOPC Builder システムに使用できます。Qsys に Avalon-MM Clock Crossing Bridge アービトレーション・シェアを含む SOPC Builder デザインのをポートする場合、Qsys は自動的に Qsys バージョンに古いバージョンを変更します。

## Avalon-MM Pipeline Bridge

Avalon-MM Pipeline Bridge は、レジスタ・ステージを Avalon-MM コマンドと反応パスに挿入します。それは、Avalon-MM スレーブ・ポート上でコマンドを受け取り、その Avalon-MM マスタ・ポートに伝播します。コマンドと応答のネットワークでパイプラインをオンにするには別々のパラメータが提供されています。

このブリッジのパイプラインの機能をオフにすることができるため、複数の Avalon-MM スレーブ・デバイスを制御する 1 つの Avalon-MM スレーブ・インタフェースをエクスポートするための Avalon-MM ブリッジを使用することができます。このコンフィギュレーションでは、その Avalon-MM マスタ・ポートへの Avalon-MM スレーブ・インタフェースで受信したコマンドを転送します。図 7-14 は、その使用方法を示しています。

図 7-14. Avalon Bridge



Avalon-MM スレーブ・インタフェースがデバイスのピンにエクスポートされるので、シングル of Avalon-MM スレーブ・ポート（個々の Avalon-MM スレーブ・デバイスのための別個のポートというよりも）を持っていることが、FPGA のピン数を削減します。

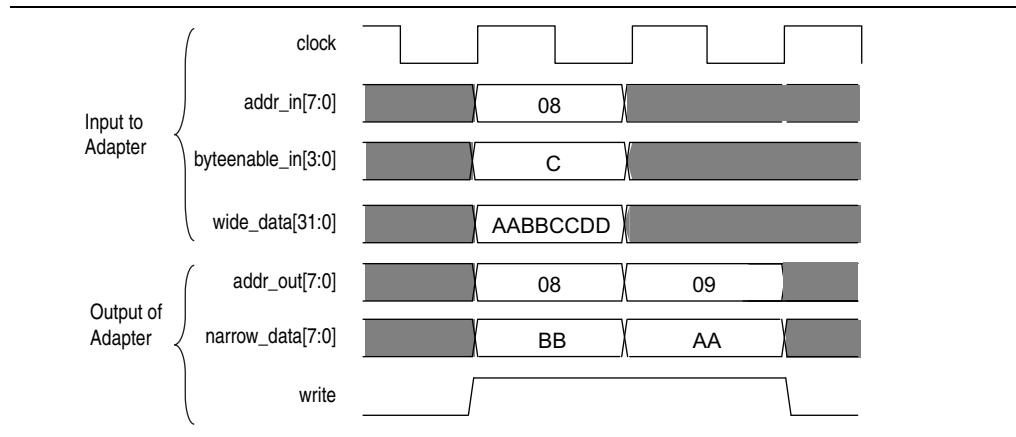
### Merlin Width Adapter

Merlin Width Adapter は、異なるデータおよびバイト・イネーブル幅の Avalon-MM マスタとスレーブ間の変換を実行します。このアダプタは、Avalon-ST のドメインで使用され、4 ページの図 7-2 に示されるパケット・フォーマットに含まれている情報で動作します。その 1 つのデータ幅付きのシンク・インタフェースを受け取り、異なるデータ幅で、ソース・インタフェースで出力パケットを生成します。狭い幅と広いデータ幅の比率は、4:1、8:1、および 16:1 のような 2 の累乗でなければなりません。このアダプタは、入力および出力パケットの順序フィールドが同じであることを前提としています。唯一の違いは、データの幅と、付属のバイト・イネーブル信号です。

Width Adapter が狭いデータに幅のデータから変換するとき、幅の狭いデータはいくつかのビットを介して送信されます。最初の出力のビットは、入力データとバイト・イネーブルの最下位アドレス・セグメントが含まれています。図 7-15 に、4:1 幅アダプタのタイミングを示しています。

Width Adapter は、幅の狭いデータか広いデータから変換するとき、各入力ビットのデータとバイト・イネーブルは、広い出力データとバイト・イネーブル信号の適切なセグメントにコピーされます。

図 7-15. 4:1 アダプタの幅のアダプタ・タイミング



## Burst Transfers

Avalon-MM バースト・トランザクションは、Avalon-MM スレーブに指定される数の転送のマススタ連続アクセスを許可します。burstcount 信号でバーストを開始する時には、マススタは転送の数を指定します。いったんマススタとスレーブ・ペア間でバーストが開始したら、アービタ・ロジックは、バーストが完成するまでロックされます。バースト・マススタでは、バースト・サイズは、マススタがスレーブにアクセスできるサイクルの数、および選択されたアービトレーション・シェアは効果が全然ありません。

### Merlin Burst Adapter

Qsys インタコネクは、Qsys Merlin Burst Adapter を使用して、バースト転送をサポートしていないインタフェースを含むシステム内の各インタフェースのバースト機能を対応します。各インタフェースの最大バースト・レンジは、コンポーネント・インタフェースのプロパティであり、システム内の他のインタフェースとは無関係です。したがって、特定のマススタは、スレーブのサポートされる最大バースト・レンジよりも長いバーストを開始させることができる場合もあります。このケースでは、スレーブはバーストがサポートされていない場合、バースト・アダプタは、大きいマススタ・バーストから小さなバーストに、または個々のスレーブの転送に変換します。マススタは、バーストが完了するまで、アービタのロジックは、他のマススタをターゲット・スレーブへのアクセスから防止します。たとえば、マススタが 8 の最大バースト長とスレーブ 16 転送のバーストを開始する場合、バーストアダプタはスレーブに長さ 8 の 2 バーストを開始します。

Avalon-MM マスタは、常に、転送のサイズにアラインメントされるアドレスを発行します。しかし、場合によっては、狭いから広いの幅適応が使用されている場合、結果のアドレスは、アラインドされていないことがあります。アラインメントされていないアドレスの場合は、Burst Adapter は、適切なバイト・イネーブルのある可能な最大サイズのバーストを発行して、バースト・イン進行をアラインメントされるスレーブ・アドレスまでを立ち上げります。その後、アラインメントされるアドレスでのバーストが完了します。

## Burst Types

バースト・アダプタは、Avalon-MM マスタのさまざまなプロパティに対応するために変数ラップまたはシーケンシャル・バースト・タイプをサポートしています。これらのバースト・タイプの定義については、4 ページの表 7-1 を参照してください。いくつかバースト・マスタは、複数のバーストのタイプを発行することができます。

## Avalon-ST インタフェース

Avalon-ST のインタコネクは、Avalon-ST インタフェースを使用する高帯域幅と低レイテンシのコンポーネントを接続します。このインタコネクは、マルチチャネル・ストリーム、パケット、および DSP データを含む単方向トラフィックのデータ・パスを作成します。Avalon-ST インタコネクは柔軟性を備えており、Ethernet、Interlaken やビデオなどの業界標準の通信およびデータ通信コア向けに、オンチップ・インタフェースを実装するのに使用できます。すべてのケースで、バス幅、パケット、およびエラー条件を定義することができます。

Qsys で、Avalon-ST のソースおよびシンク・ポートの接続方式を指定します。ソースとシンク・インタフェースが、種々のプロパティがある場合、Tools メニュー Qsys の **Insert Avalon-ST adapters** を選択することにより、表示される必要なアダプタが **System Contents** タブに挿入されます。

## Avalon-ST の例

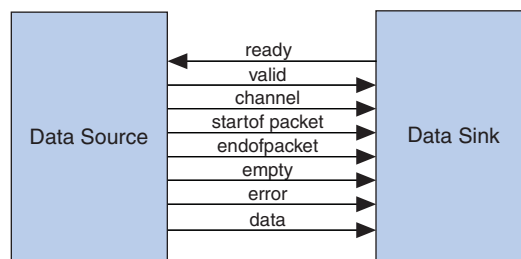
図 7-16 に、ソースとシンク間の Avalon-ST の接続で最も単純なシステムのサンプルを示します。このソース/シンク・ペアは、data 信号のみが含まれています。シンクは、ソース・インタフェースがリセットから終了するとすぐにデータを受信することができます。

図 7-16. シンプル Avalon ストリーミングのソース/シンク・ペアのためのインタコネク




図 7-17 に、パケットの開始と終了を示す信号、チャンネル番号、エラー条件、バック・プレッシャを含むより包括的なインタフェースを示します。

図 7-17. パケット・データのための Avalon ストリーミングのインタフェース



Avalon-ST インタコネクトを使用するすべてのデータ転送は、関連するクロック・インタフェースの立ち上がりエッジに同期して行われます。システムのスループットおよび周波数は、コンポーネント間の接続方式によって決まります。

 Avalon-ST インタフェースのプロコルについて詳しくは、[Avalon Interface Specification](#) を参照してください。

## Avalon-ST コンポーネント

Qsys コンポーネント・ライブラリで、異なるプロパティのある入力ストリームおよび出力ストリームのデータパスが含まれて、データパスを作成するために使用できるの Avalon-ST コンポーネントがいくつかあります。生成プロセスは、その構造が [7-7 ページの「Qsys の変換」](#) で説明するようにネットワーク・トポロジのようなインタコネクトを作成するため、Avalon-MM マスタとスレーブのコンポーネントが含まれる生成されたシステムは、これらの Avalon-ST のコンポーネントを使用することができます。以下のセクションは Avalon-ST コンポーネントを導入します。


### Avalon-ST Handshake Clock Crosser

Avalon-ST Handshake Clock Crossing アダプタは、異なる周波数で動作ストリームを接続します。このアダプタは、簡単なハンド・シェイク・プロトコルを使用して、他の方向のクロック境界と応答間の転送コントロール信号および応答を伝播します。次の転送を開始する前に、各転送が安全にターゲット・ドメインに伝播されるため、この方法論は、より少ない FPGA リソースを使用しています。ハンド・シェイクがパフォーマンスを制限し、すべてのリード・コマンドの往復レイテンシの少なくとも 4 つのサイクルを招くため、Avalon-ST Handshake Clock Crossing は、低スループットの接続に適用されます。

パラメータ値を指定するには、Avalon-ST Handshake Clock Crosser のパラメータ・エディタを使用することができます。指定できるパラメータの中のデータ幅は、パケット・サポートとシンクロナイザの容量を含めるかどうかのデータ幅です。

### Avalon-ST Pipeline Stage

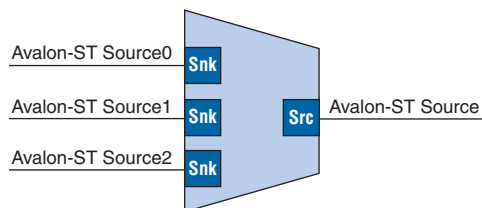
Avalon-ST コマンドと応答のデータパスの Avalon-ST のパイプライン・ステージは、必要に応じてシングル・パイプライン（レジスタ）のステージを挿入します。それは、Avalon-ST のシンク・インタフェース上でデータを受信し、その Avalon-ST ソース・インタフェース上でそれを変更せずにドライブします。

 また、Qsys コンポーネント・ライブラリに、そのデータ・インタフェースは、Avalon-ST プロトコルではなく、Avalon-MM プロトコルを使用する Avalon-MM Pipeline Bridge が含まれています。

### Merlin Multiplexer

Merlin Multiplexer は、Avalon-ST シンク・インタフェース上のデータを受け入れ、Avalon-ST ソース・インタフェースでトランスミッションのためのデータを多重化します。ソース・データをドライブしているシンクを示すために、ソースにチャンネル情報を付加するようにマルチプレクサをパラメータ化することができます。マルチプレクサは、ラウンド、ロビン、アービトレーション・アルゴリズムによる入力間で選択する内部アービトレーション・ロジックが含まれています。[図 7-18](#) に、Avalon-ST のマルチプレクサを示しています。指定可能なパラメータの中のパケットのスケジューリングを使用するオプションがあります。このオプションにより、このマルチプレクサは、パケットの端の入力のみ変更することが保証されます。

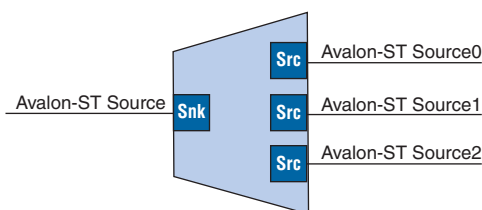
図 7-18. Merlin Multiplexer



### Merlin Demultiplexer

Merlin Demultiplexer は、そのシンク・インタフェース上でチャンネル・データを受け入れ、そのソースのいずれかのインタフェース上でデータを送信します。ソース・ストリームのチャンネル・ビットは、出力データをドライブするポートを示します。図 7-19 に、Merlin Multiplexer を示しています。指定可能なパラメータは、出力ポートの数およびチャンネルの信号の幅です。

図 7-19. Avalon-ST Demultiplexer



## Avalon-ST および Avalon-MM インタフェース

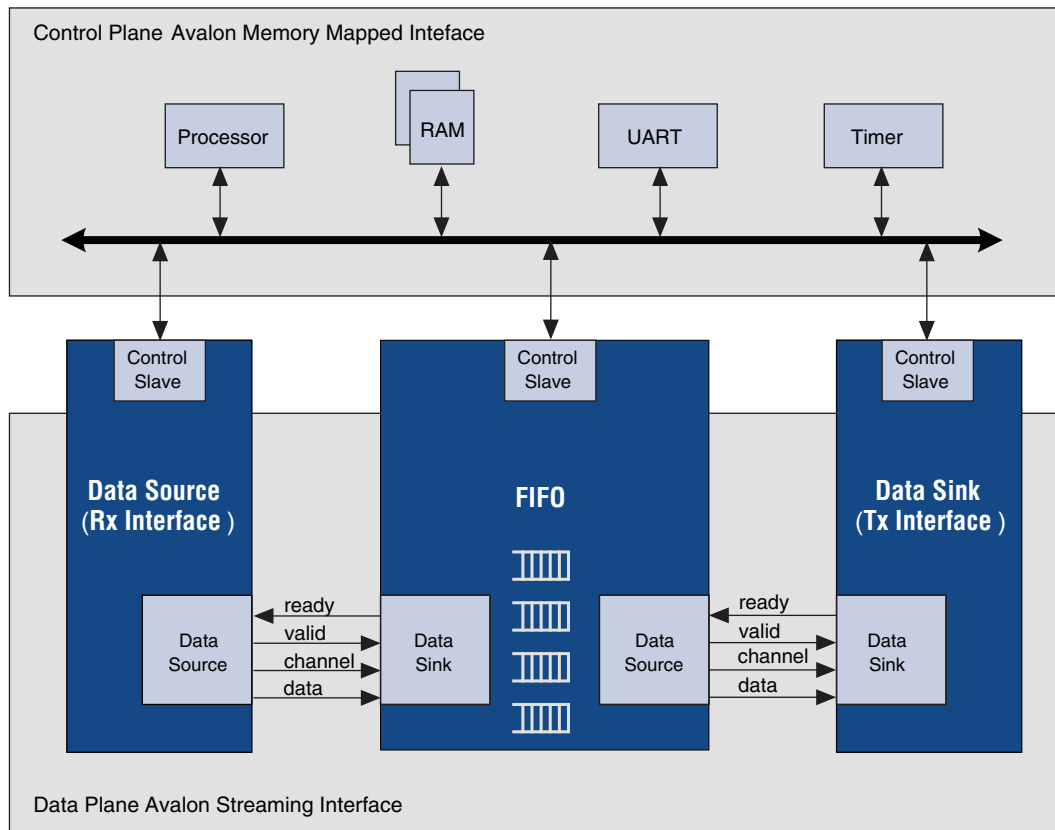
Avalon-ST および Avalon-MM インタフェースはコンプリメンタリです。ストリーミング・データを持つ高帯域幅のコンポーネントは、通常、高スループット・データパス用の Avalon-ST インタフェースを使用します。また、これらのコンポーネントは、コントロールのアクセス・ポイントを提供するための Avalon-MM 接続インタフェースを使用することができます。各種のトポロジを作成するために使用できる Avalon-MM インタコネクとは対照的に、Avalon-ST インタコネク・ファブリックは、図 7-20 に示すとおり、1 つのデータ・ソースとデータ・シンク間のポイン・トツ・ポイントを作成します。

この図には 2 つの接続ペアがあります。

- Rx インタフェースのデータ・ソースは、データを FIFO のデータ・シンクに転送します。
- FIFO のデータ・ソースは、データを Tx インタフェースのデータ・シンクに転送します。

図 7-20 には、プロセッサは Avalon-MM インタフェースによりデータ・ソース、FIFO、またはデータ・シンクにアクセスして、システム・コントロールを提供します。

図 7-20. Avalon-MM Avalon-ST インタフェースの使用



## トライ・ステートのコンジット・コンポーネント

Avalon-TC のインタフェース・タイプを使用すると、PCB 上のトライ・ステート・デバイスに接続する Qsys サブシステムをデザインすることができます。以下の 3 つのコンポーネントでは、トライ・ステート・コンジット機能を実装します。

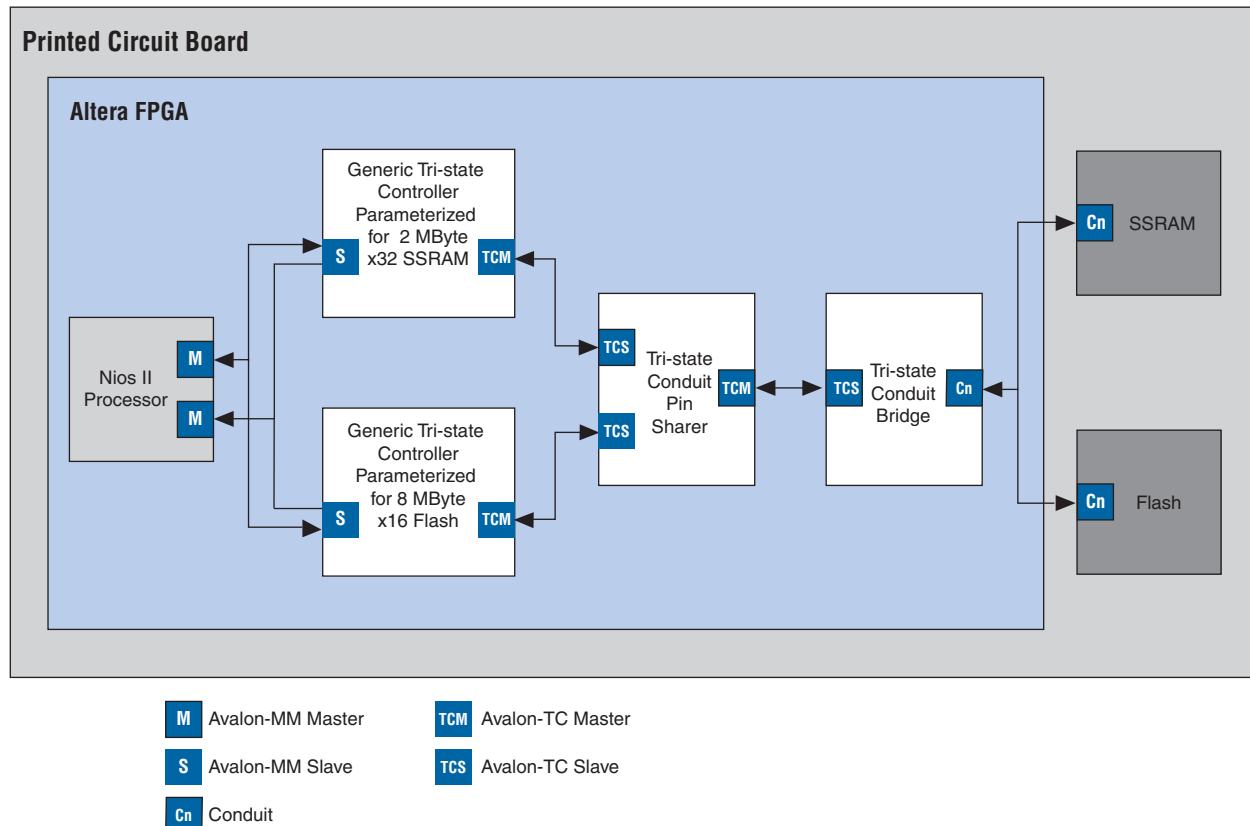
- [Generic Tri-state Controller](#)
- [Tri-state Conduit Pin Sharer](#)
- [Tri-state Conduit Bridge](#)

Avalon-TC 信号の種類で記述できるデバイスのためにピンの共有を実装、単方向および双方向の信号を変換、およびトライ・ステート・コントローラを作成するには、これらのコンポーネントを使用することができます。

- 🔗 Avalon-TC 信号の種類については、[「Avalon Interface Specifications」](#) の [「Avalon Tri-state Conduit Interfaces」](#) の項を参照してください。

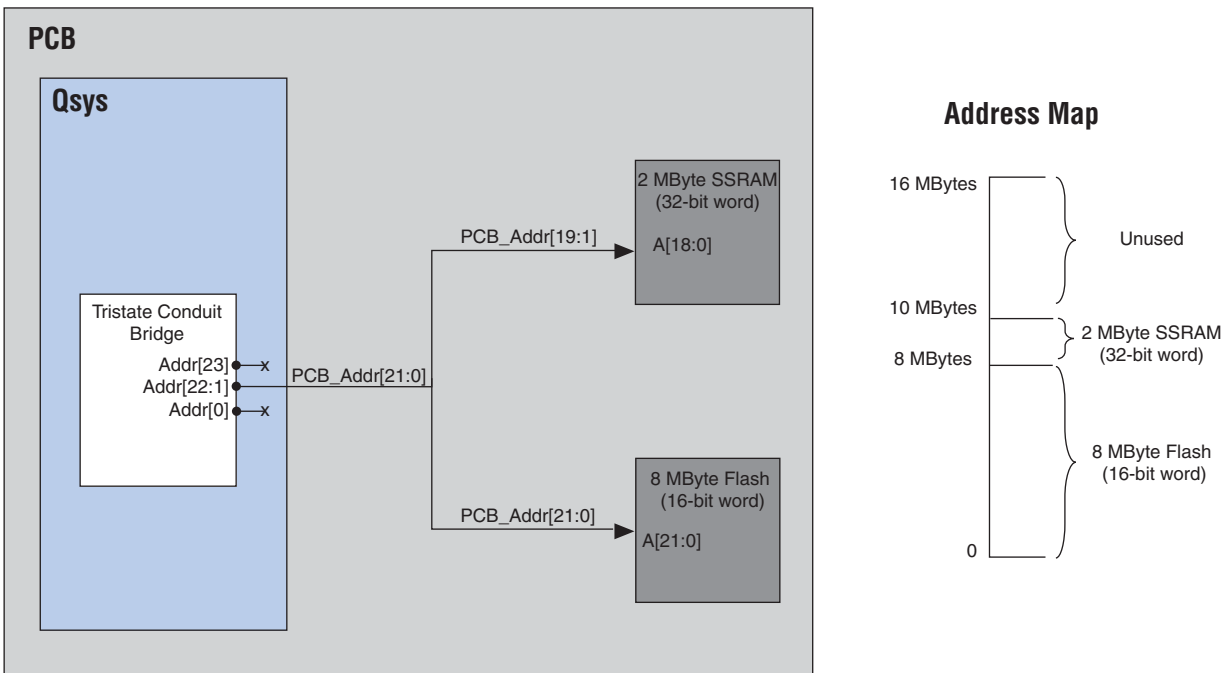
図 7-21 に、これらのコンポーネントの一般的な使用方法を示しています。この図は、2つの Generic Tri-state Conduit Controllers が含まれています。最初のものは、フラッシュ・メモリを制御するためにカスタマイズされています。2番目のものは、オフ・チップ SSRAM を制御するためにカスタマイズされています。Tri-state Conduit Pin Sharer はこれら 2つのコントローラ間を多重化し、そして Tri-state Conduit Bridge はトライ・ステート信号とトゥルーの双方向信号のオン・チップ・エンコードに変換します。

図 7-21. オフ・チップ SRAM およびフラッシュ・デバイスをコントロールするトライ・ステートのコンジット・システム



デフォルトでは、Tri-state Conduit Pin Sharer および Tri-State Conduit Bridge はバイト・アドレスを示します。多くのメモリ・デバイス内の各アドレス位置には複数のデータ・バイトが含まれています。図 7-21 に示した例では、フラッシュ・デバイスは、16 ビット・ワードで動作し、Avalon-MM アドレスの最下位ビットを無視する必要があります。そして、SSRAM メモリは 32 ビット・ワードで動作し、2 の下位メモリ・ビットを無視する必要があります。デバイスはバイト・アドレスを必要としないため、PCB 上で addr[0] が接続されていません。図 7-22 に、接続されていないとしての addr[0] を示します。

図 7-22. Qsys System から PCB へのアドレスの接続



このデザイン例で、フラッシュ・デバイスは、0 MBytes から 8 MBytes-1 の範囲を MAC アドレスに応答します。そして、SSRAM は、8 MBytes から 10 MBytes-1 の範囲に応答します。SSRAM は、32 ビットのワード・アドレスに応答するため、PCB の PCB 回路図は、SSRAM デバイスの `addr[18:0]` に `addr[20:2]` を接続します。8M バイトのフラッシュ・デバイスは、16 ビットのワード・アクセスに応答します。その結果、略図は `addr[0]` を接続しません。chipselect 信号は、2つのデバイス間を選択します。

 ワード・アラインされたアドレスによるカスタム・トライ・ステート・コンジットのマスタを作成する場合、Tri-state Conduit Pin Sharer は、アドレス信号を変更したり、整理することがありません。

図 7-23 に、Qsys のシステムのサンプルを示します。


図 7-23. Qsys のトライ・ステートのコンジット・システム

System Contents	Address Map	Clock Settings	Project Settings	System Inspector	HDL Example	Generation
<input checked="" type="checkbox"/>				<b>nios2_qsys_0</b>		
				data_master	Avalon Memory Mapped Master	Click to export
				instruction_master	Avalon Memory Mapped Master	Click to export
<input checked="" type="checkbox"/>				<b>flash_controller</b>		
				uas	Avalon Memory Mapped Slave	Click to export
				tcm	Tristate Conduit Master	Click to export
<input checked="" type="checkbox"/>				<b>ssram_controller</b>		
				uas	Avalon Memory Mapped Slave	Click to export
				tcm	Tristate Conduit Master	Click to export
<input checked="" type="checkbox"/>				<b>tristate_conduit_pin_sharer_0</b>		
				tcm	Tristate Conduit Master	Click to export
				tcs0	Tristate Conduit Slave	Click to export
				tcs1	Tristate Conduit Slave	Click to export
<input checked="" type="checkbox"/>				<b>tristate_conduit_bridge_0</b>		
				tcs	Tristate Conduit Slave	Click to export
				out	Conduit	tristate_conduit_out

## Generic Tri-state Controller

Generic Tri-state Controller は、オフ・チップ・デバイスの動作を反映するようにパラメータ化するコントローラ用のテンプレートを提供します。Generic Tri-state Controller は、以下の例のように、このコンポーネントのカスタマイズするのに使用できる多くのパラメータがあります。

- アドレスおよびデータ・信号の幅
- リードおよびライトの待ち時間
- バスのターンアラウンド・タイム

 遅延の計算では、Generic Tri-state Controller は、より大きいバスのターンアラウンド・タイムおよびリード・レイテンシを選択します。ターンアラウンド・タイムは以前が返されたデータを読み出した時間から測定されるのではなく、コマンドを受け入れる時間から測定されます。

- データのホールド時間

Generic Tri-state Controller には常に以下のインタフェースが含まれます。

- Avalon-MM スレーブ・インタフェース — このインタフェースは、Nios II プロセッサなどの、Avalon - MM マスタに接続します。
- Avalon-TC マスタ・インタフェースは、通常、トライ・ステート・コンジットのピンの共有のトライ・ステート・コンジットのスレーブ・インタフェースに接続します。
- Avalon Clock シンク — コンポーネントのクロックの基準。このインタフェースは、クロック・ソースに接続する必要があります。
- Avalon Resets シンク — このインタフェースは、リセット・ソース・インタフェースに接続します。

## Tri-state Conduit Pin Sharer


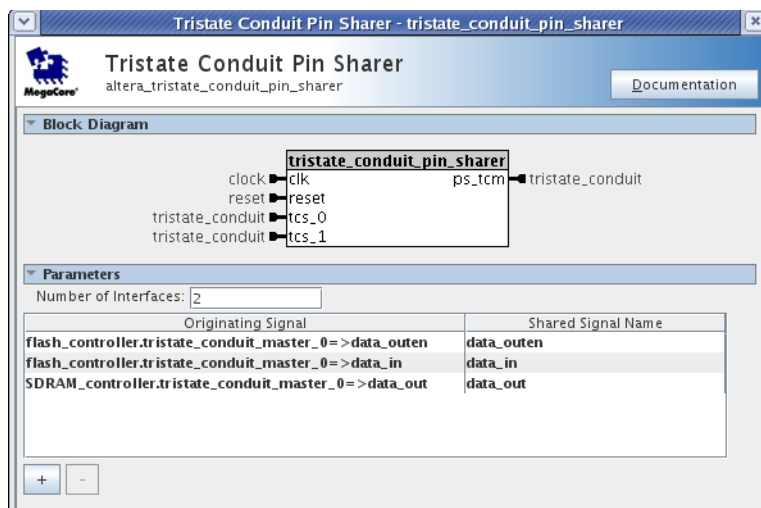
接続されたトライ・ステート・コントローラの信号間の Tri-state Conduit Pin Sharer 多重。Tri-state Conduit Pin Sharer にトライ・ステート・コントローラからすべての信号を接続し、共有されている信号を指定するパラメータ・のエディタを使用してください。パラメータ・エディタでは、 7-24 に示すように、共有の信号名を入力するための Shared Signal Name カラムが含まれています。

図 7-24. Tri-state Conduit Pin Sharer による共有信号の指定



共有信号の幅が異なる場合、信号はその第 0 ビットにアライメントされています。そして、小さな信号がバスのコントロールを持つたびに、高次のピンが 0 にドライブされています。非共有の信号は、常に Pin Sharer を介して伝播します。Tri-state Conduit Pin Sharer は、トライ・ステート・コンジット・コントローラ間で選択するために 7-11 ページの「アービトレーション」に記載されているラウンド・ロビン・アービタを使用しています。



特定の Pin Sharer に接続されているすべてのトライ・ステート・コンジットのコンポーネントは、同じクロック・ドメインに存在する必要があります。

## Tri-state Conduit Bridge

コンポーネントを介して直接に他のすべての信号を通過しながら、Tri-state Conduit Bridge は、それぞれのトライ・ステート信号用に双方向の信号をインスタンス化します。Tri-state Conduit Bridge は、リード要求のためにレイテンシの 2 サイクルが追加され、すべての発信の信号および着信の信号を登録します。カスタム・コントローラをデザインするときは、この追加のパイプラインを考慮する必要があります。リセット中に、全ての出力がハイ・インピーダンス状態に、配置されます。リセットがデアサートされた後の出力は、最初のクロック・サイクルでイネーブルされています。Quartus II ソフトウェアで、これらの出力の信号は双方向にラベルを付けます。

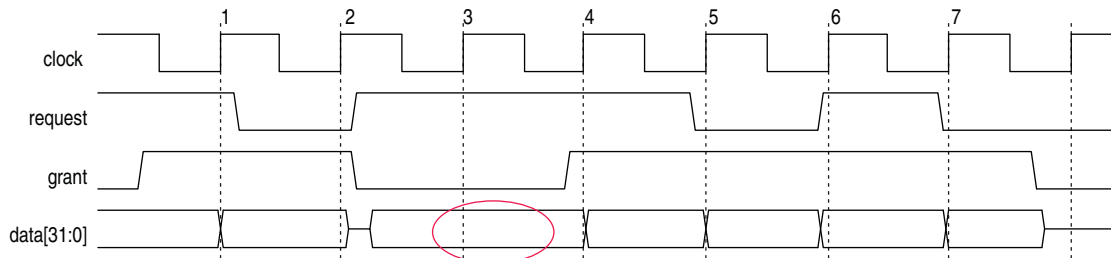
## タイミング

図 7-25 に、アービトレーションのタイミングを示します。この図に示すように、デバイスが付与されたサイクルで有効データをドライブすることができます。図 7-25 に、以下のイベントを示しています。

1. サイクル 1 においては、アービタは要求を許可します。付与されたデバイスは、サイクル 1 と 2 で有効データをドライブします。
2. サイクル 4 では、アービタは要求を許可します。付与されたデバイスは、サイクル 4 と 5 で有効データをドライブします。

3. サイクル 6 では、アービタは要求を許可します。付与されたデバイスは、サイクル 6 と 7 で有効データをドライブします。
4. サイクル 3 は有効データが含まれていないサイクルだけです。

図 7-25. アービトレーションのタイミング



## 割り込みインタフェース

割り込み送信のインタフェースのあるシステムでは、Qsys インタコネクは、割り込み処理を実装するためにいくつかのコンポーネントが含まれています。Qsys は個々のシングル・ビットの割り込み要求 (IRQ) を処理します。複数の送信側が同時にその IRQ をアサートした場合に、レシーバ・ロジックは、(通常はソフトウェアの制御下で) 最優先の IRQ を判定し、適切に応答します。

個々の要求を使用して、割り込みロジックは、各割り込みレシーバに接続されている 32 IRQ 入力を処理できます。このロジックで、receiver\_0 を中断するために接続されている割り込み送信者は最優先であり、シーケンシャル・レシーバは継続的に優先順位の低いレシーバになります。Merlin IRQ マップ・コンポーネントをインスタンス化することによって、割り込み送信側の優先順位を再定義することができます。7-28 ページの「Merlin IRQ Mapper」について詳しくは、を参照してください。

## Qsys 内の IRQ の割当

Qsys の **System Contents** タブで IRQ の接続をアサインします。システムにすべてのコンポーネントを追加した後、割り込み送信側と受信側を接続します。それぞれの受信に対して、IRQ 番号を指定したり、IRQ を接続しないように指定する IRQ カラムを使用することができます。

- ② 詳しくは、「Quartus II Help」の「[Connecting Qsys Components](#)」を参照してください。

Qsys は、割り込みの処理を実装するために以下の 3 つのコンポーネントを使用しています。

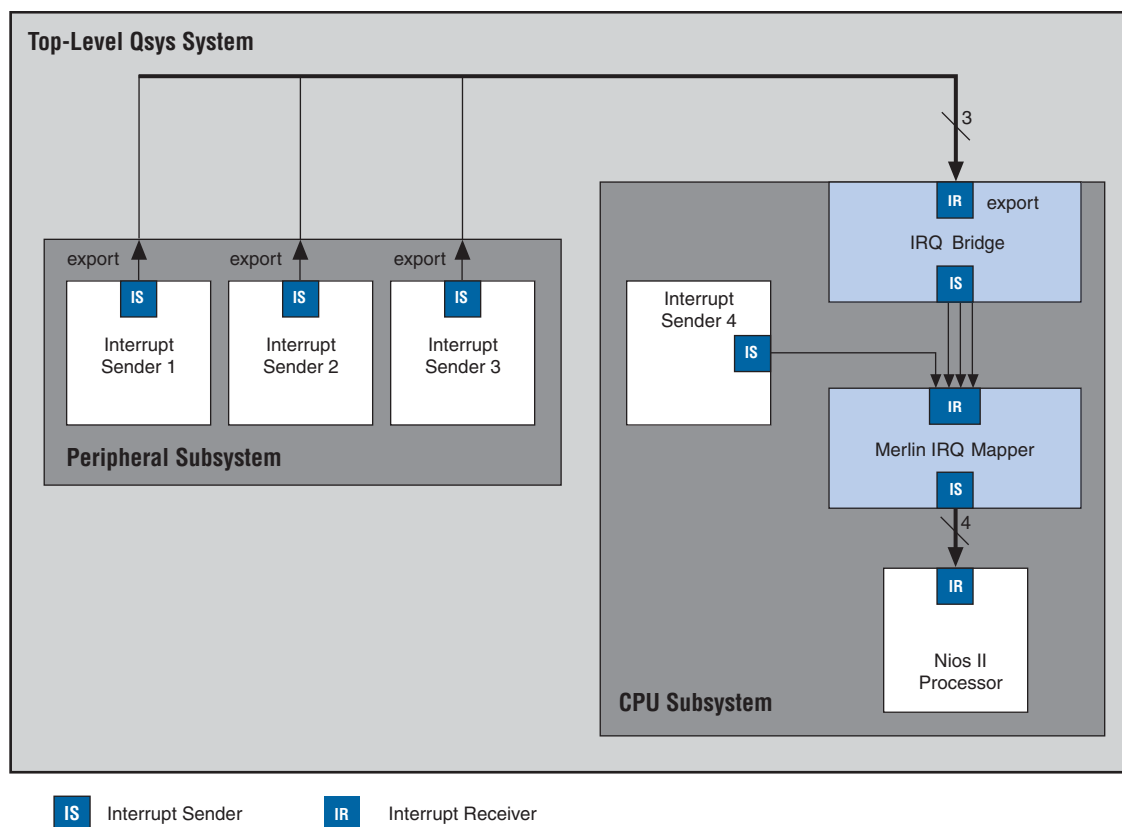
- [IRQ Bridge](#)
- [Merlin IRQ Mapper](#)
- [Merlin IRQ Clock Crosser](#)

以下の項では、これらのコンポーネントについて説明しています。

## IRQ Bridge

IRQ Bridge では、Qsys のサブシステム間で割り込みワイヤを配線することができます。図 7-26 に、Peripheral Subsystem はサブシステムの最上位レベルにエクスポートされる 3 つの割り込み送信者があります。これらの割り込みは、CPU Subsystem の Merlin IRQ のレシーバ・ブリッジに配線されます。

図 7-26. Qsys IRQ Bridge のアプリケーション



## Merlin IRQ Mapper

Merlin IRQ Mapper は、バスに個々の割り込みワイヤに変換します。また、割り込み番号を指定するには、IRQ Mapper を使用することができます。デフォルトでは、IRQ Mapper のレシーバ・インタフェースに接続されている割り込み送信側は、最優先であり、シーケンシャル・レシーバは継続的に優先順位の低いレシーバになります。優先順位を再マップするパラメータ・エディタで **IRQ Map** パラメータを使用することができます。例えば、図 7-26 での IRQ Mapper に接続されている 4 つの割り込み送信側の優先順位を逆回転するには、**IRQ Map** のパラメータは次の文字列を入力できます：0:3、1:2、2:1、0:3

## Merlin IRQ Clock Crosser

Merlin IRQ Clock Crosser は、割り込み送信側と異なるクロック・ドメインにある受信側を同期させます。このコンポーネントを使用するには、割り込み送信側と受信側のインタフェースに加えて、割り込み送信側と受信側の両方にクロックを接続してください。必要に応じて、Qsys は自動的にこのコンポーネントを挿入します。

## クロック・インタフェース

外部クロック・ソース（例えば、ボード上のオシレータ）を定義するために **Clock Settings** のタブを使用することができます。各クロック・ドメイン、すべてのクロックの1つのリセット・ソース、または両方に個別のリセット・ソースを定義することができます。

## リセット・インタフェース

Tools メニューの **Auto-Connect Resets** を選択することにより、Qsys により生成されたシングル・グローバル・リセット・ドメインを持つことを選択することができます。デザインに複数のリセット・ドメインを必要とする場合、自身のリセット・ロジックと接続性を実装することができます。

### Qsys で実装したシングル・グローバル・リセット信号

Tools メニューの **Auto-Connect Resets** を選択した場合は、Qsys インタコネクは、グローバル・リセット・バスを配布しています。リセット要求のすべては、**oRed** で統合され、各クロックド・メインに同期され、リセット入力に供給されています。リセット信号の期間は、少なくとも1つのクロック周期です。

Qsys インタコネクは、以下の条件でシステム・ワイド・リセットを挿入します。

- Qsys システムへのグローバル・リセット入力のアサートされる場合。
- いずれかのコンポーネントは、その `resetrequest` 信号をアサートする場合。

### 複数のリセット信号

Qsys のコンポーネント・ライブラリには、リセット・機能を実装するためにリセット・コントローラおよびリセット・ブリッジが含まれています。また、独自のリセット・ロジックをデザインすることができます。



リセット回路をデザインする場合には、慎重にシステムのロックアップを引き起こす可能性がある状況を考慮する必要があります。例えば、Avalon-MM スレーブがトランザクションの途中でリセットされる場合、Avalon-MM マスタはずっと待機する可能性があります。

#### Merlin Reset Controller

複数のリセット入力でシステムをデザインする場合、Merlin Reset Controller はすべてのリセット入力を OR して、単一のリセット出力を生成します。Reset Controller は、その動作をカスタマイズするために指定できる次の3つのパラメータがあります。

- **Number of inputs**— 信号のリセット出力を作成するには OR をインタフェースする個々のリセットの数を示します。

- **Output reset synchronous edges**— 同期のレベルを指定します。1つの以下のオプションを選択することができます。
  - **None**— リセット非同期にアサートとディアサートされます。内部同期回路をデザインされている場合、この設定を使用することができます。
  - **Both**— リセットが同期にアサートとディアサートされます。
  - **Deassert**— リセットは同期的にディアサートされ、非同期でアサートされません。
- **Synchronization depth**— シンクロナイザがメタステーブル・イベントの伝播を排除するために使用するレジスタ・ステージを指定します。

Qsys は、以下の条件でリセットされたシンクロナイザを自動的に挿入します。

- 複数のリセット・ソースはリセット流し台と接続されます。
- リセット・ソースの同時的なエッジおよびリセット・シンクの同時的なエッジの間にミスマッチがあります。

### Reset Bridge

Reset Bridge では、Qsys システムの 2 つ以上のサブシステムのリセット信号を使用することができます。ローカル・コンポーネントに 1 つのリセット・ソースを接続し、必要に応じて他のサブシステムに 1 つ以上をエクスポートすることができます。パラメータ・エディタを使用してリセット出力の数を指定します。

## Avalon コンジット

他の Avalon のインタフェース・タイプのいずれかに適合しないインタフェースのための Avalon コンジットのインタフェース・タイプを使用することができます。信号の任意の収集をグループ化するために Avalon コンジットインタフェースを使用することができます。他のインタフェース・タイプと同様に、コンジット・インタフェースをエクスポートまたは接続することができます。PCI Express IP コアの PCI Express リンクは、エクスポートのためのコンジット・インタフェースの使用例です。

Qsys 内の 2 つのコンジット・インタフェースを接続するには、以下の条件を満たす必要があります。

- インタフェースは、同じ信号の役割と幅で完全に一致する必要があります。
- インタフェースは、反対の方向でなければなりません。



コンジット接続は常にポイント・ツー・ポイント接続です。



Avalon Conduit について詳しくは、[Avalon Interface Specifications](#) を参照してください。

## 概要 : Qsys インタコネク・コンポーネント

表 7-3 に、Qsys インタコネクを実装するすべての Qsys のコンポーネントを示します。

表 7-3. Qsys インタコネク・コンポーネントの概要

コンポーネント名	一般的なアプリケーション	
	内部 Qsys インタコネク (注 1)	ユーザー・デザイン
<b>Avalon-MM マスタおよびスレーブ・ネットワークの変換</b>		
Merlin Master Translator	✓	—
Merlin Master Agent	✓	—
Merlin Router	✓	—
Merlin Traffic Limiter	✓	—
Merlin Slave Translator	✓	—
Merlin Slave Agent	✓	—
<b>Avalon-ST コンポーネント</b>		
Avalon-ST Handshake Clock Crosser	✓	✓
Avalon-ST Pipeline Stage	✓	✓
Merlin Multiplexer	✓	✓
Merlin Demultiplexer	✓	✓
<b>ブリッジ</b>		
Clock Bridge	—	✓
Avalon-MM Clock Crossing Bridge	—	✓
Avalon-MM Pipeline Bridge	—	✓
<b>アービトレーションおよびアダプタ</b>		
Merlin Arbiter	✓	—
Merlin Width Adapter	✓	✓
Merlin Burst Adapter	✓	✓
<b>トライ・ステート・コンジット</b>		
Generic Tri-state Controller	—	✓
Tri-state Conduit Pin Sharer	—	✓
Tri-state Conduit Bridge	—	✓
<b>割り込み</b>		
IRQ Bridge	—	✓
Merlin IRQ Mapper	✓	—
Merlin IRQ Clock Crosser	✓	✓

表 7-3. Qsys インタコネク・コンポーネントの概要

コンポーネント名	一般的なアプリケーション	
	内部 Qsys インタコネク (注 1)	ユーザー・デザイン
	リセット	
Merlin Reset Controller	✓	✓
Reset Bridge	—	✓

表 7-3 の注:



- (1) これらのコンポーネントは、Qsys インタコネクへの理解を拡張するために説明されています。自身のデザインでそれらを使用する必要はない場合もあります。
- (2) この表では、✓ は、コンポーネントが通常カラム・ヘッダで指定された目的のために使用されることを意味します。そして、— は、コンポーネントが通常カラム・ヘッダで指定された目的のために使用されていないことを意味します。

## 改訂履歴

表 7-4 に、本資料の改訂履歴を示します。

表 7-4. 改訂履歴

日付	バージョン	変更内容
2011 年 5 月	11.0.0	ベータのステータス削除。
2010 年 12 月	10.1.0	初版。

-  Quartus II ハンドブックの前バージョンには、[Quartus II Handbook Archive](#) を参照してください。
-  このハンドブックの章についてのフィードバックを提供するために、[オンライン調査](#)を取ってください。