

Aurora 活用 チュートリアル

~Xilinx 社の FPGA 間通信プロトコルを
使いこなす

久保田新二

ここでは、FPGA 間インターフェースで用いるデータ・リンク層通信プロトコルの一つである米国 Xilinx 社の「Aurora」の活用法を解説する。Aurora 通信ブロックの生成方法やユーザ論理とのインターフェース法、検証手法について具体的に説明する。
(編集部)

Aurora は、米国 Xilinx 社のデータ・リンク層通信プロトコルです。FPGA が搭載する高速シリアル通信ブロック RocketIO の MGT ブロックを使用します(図1)。通信したいデータにヘッダとフッタを自動的に付加して送受信を行います(図2)。

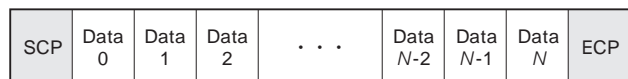


図2 シリアル・データの packets のイメージ
通信したいデータにヘッダとフッタを自動的に付加する。

RocketIO の MGT ブロックは、さまざまなシリアル通信仕様に柔軟に対応しています。しかし、特性の設定やデータ同期への考慮が難しいため、とても使用しにくいものです。これを簡単に使用できるように開発されたプロトコルが、Aurora です。

Aurora モジュールは、Xilinx 社の FPGA 開発ツールの一つで、IP コアの設定・生成を行う「CORE Generator」を使って作成します。ウィザードに従ってユーザ・インターフェースのパラメータと通信速度の値を設定してだけです。難しい MGT の設定は自動で行ってくれるので、設計者は MGT をほとんど意識せずに高速シリアル通信回路を設計できます。後は Aurora 通信ブロックに対し、送受信したいデータを Aurora で決められたユーザ・インターフェースの仕様通りのタイミングで受け渡しすればよいのです。

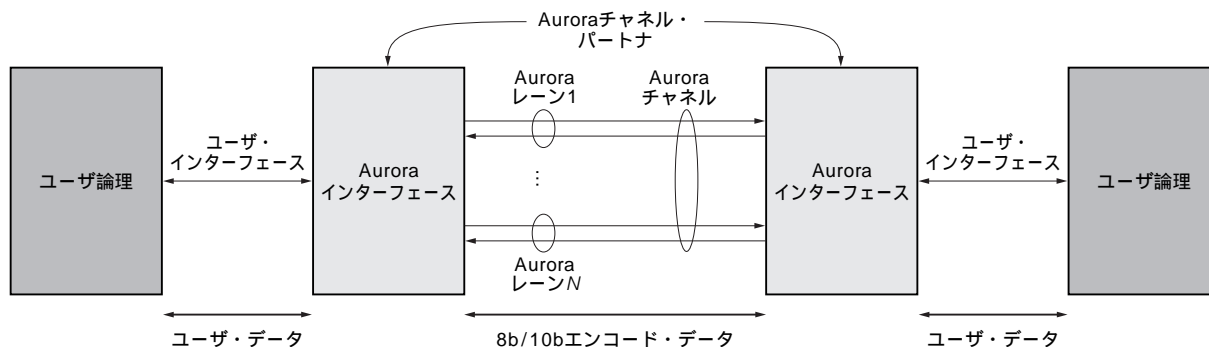


図1 Aurora を使ったシステム構成
米国 Xilinx 社の FPGA が搭載する高速シリアル通信ブロック RocketIO の MGT ブロックを使用する。

KeyWord	FPGA, 高速シリアル通信, RocketIO, Aurora, CORE Generator, MGT, クロック・コレクション, クロック, リセット, ModelSim
----------------	--

しかし、このインターフェースの仕様を正しく理解していないと、予想外の事態を引き起こす可能性があります。

1. 高速シリアル通信ブロック活用の基礎

Auroraを使用すれば、設計者はMGTをほとんど意識せずに設計を行えると述べました。しかし、MGTの基本構成を全く知らずに設計はできません。ここではVirtex-4をターゲットに、Auroraの設計に必要な最低限の知識についてまとめます。

● MGTの構成

MGTはFPGAの品種によって搭載数が異なります。また、MGTと接続されるシリアル信号線も専用ピンとして決められており、パッケージによって異なります。

Virtex-4では、物理的にMGTは左右に分かれて配置されています(図3)。同様に、MGTへ供給するクロック・ドライバも左右それぞれに配置されています。この物理的要因から、供給するリファレンス・クロックは、左右それぞれにあるMGTでは共通に使用可能ですが、左右にまたがるMGTで共有することはできません。

MGTは2個(MGTA, MGTB)で一つのMGTタイル(ペア)を構成しています。そのためMGTタイルの片側のMGTのみを使用する場合、それとペアとなっているMGTには未使用の処理が必要になります。CORE GeneratorでAuroraを作成したときに、一緒に出力されるunused_mgtというモジュールを組み込みます。さらにペアのMGT同士のコミュニケーション用にCOMBUSIN, COMBUSOUTという信号があるので、互いに接続する必要があります。

● クロック・コレクションの動作

クロック・コレクションとは、対向して接続されたAuroraの送信側から一定サイクル期間内にアイドル・シーケンスを挿入する処理のことです。

MGTの受信側には、受信データを一時蓄えるためのエラスティック・バッファというFIFOメモリが存在しています。このバッファは、書き込みと読み出しのクロックが異なります。書き込みは受信データから抽出したクロックで、読み出しはユーザ・インターフェースから供給されたクロックにて行われます。SerDesチップの場合では、受信

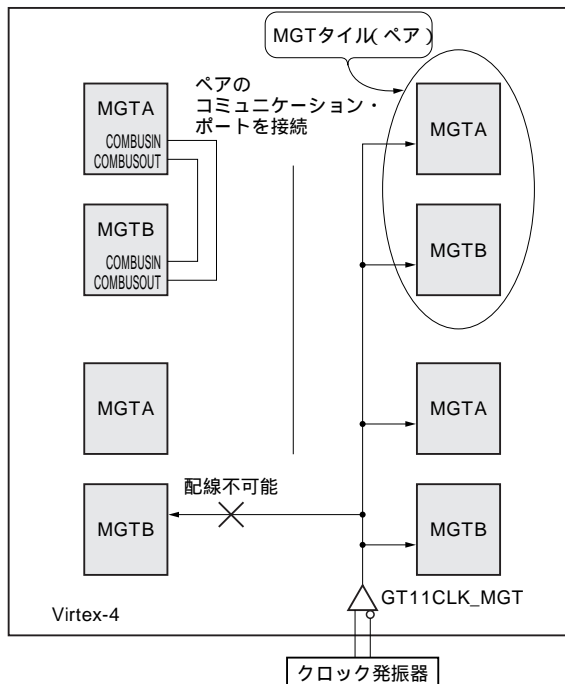


図3 MGTの構成

Virtex-4では、物理的にMGTは左右に分かれて配置されている。MGTへ供給するクロック・ドライバも左右それぞれに配置されている。MGTは2個(MGTA, MGTB)で一つのMGTタイル(ペア)を構成する。

側は従属モードとして受信したデータから抽出したクロックをユーザ・インターフェースで使用しますが、ここではクロックの乗せ換えを行っているようです。

送信側と受信側で同じ仕様の水晶発振器を使用しても、水晶発振器の精度の範囲で周波数が異なります。従って、これが原因で受信側のエラスティック・バッファにオーバフローまたはアンダフローが生じる可能性があります。すると、HARD_ERRORに至って、Lane_UPとChannel_UPがダウンし、通信が途切れてしまいます。

この問題を回避するため、一定期間内にアイドル・シーケンスを送信するサイクルが必要になります。これが送信側からのクロック・コレクションです。クロック・コレクションは、Auroraの入力ポートにあるDO_CC信号を一定期間以上アサートすることによって行います。Lane_UP, Channel_UP確立後から常に必要なサイクルです。

クロック・コレクションのアサート間隔とアサート期間は、ユーザ・インターフェースのデータ幅や水晶発振器の精度によって異なります。100ppm(ppmは百万分の1)精度では表1のようになります。

クロック・コレクションを挿入するためのタイミング設