

Altera社製FPGA Stratix VをOpenCLで開発できる OpenCL for FPGAの最適化テクニック

大澤 俊晴 Toshiharu Oosawa

FPGA マガジン No.4でOpenCL for FPGAの概要について紹介しました。今回は応用編としてOpenCL for FPGAの最適化テクニックについて解説します。最適化の柱は、パイプライン・スループットの向上、パイプライン・レイテンシの削減、メモリI/Oの効率化の3本です。

OpenCLを利用するメリットの一つは、プラットフォームに依存しない点でしょう。正しく実装されたOpenCLプログラムは、どのようなOpenCL対応デバイス上でも正しく動くことが期待できます。ただし、「パフォーマンス」に関しては、いかなる保証もありません。デバイスが変われば、適切なOpenCLプログラムの書き方も変わります。裏を返せば、デバイスの特性に合わせてコードを記述すれば劇的に性能を向上させることができるといえるでしょう。

今回は、OpenCL for FPGAのソース・コードを元に、FPGAにおけるOpenCLカーネルの最適化テクニックについて解説します。

1 最適化テクニックの概要

● 用語の定義

本題に入る前に、使用する用語の定義をしておきましょう。

- カーネル…OpenCLでは、ホストとデバイス(FPGAなど)で動作するプログラムを明確に区分しており、デバイス側で動作するOpenCLプログラムのことをカーネルと呼ぶ。
- レイテンシ…あるワーク・アイテムの処理がカーネル・パイプラインに入ってからパイプラインを出るまでのサイクル数
- スループット…単位時間あたりに処理可能なワーク・アイテム数
- ワーク・アイテム…一般的な「スレッド」と同義で、カーネルはワーク・アイテム単位で処理を実装する。
- ワーク・グループ…ワーク・アイテムの集合
- 演算のベクトル化…繰り返し処理で配列の要素をひとつひとつ計算しているような部分を、高速に演算できるように変形すること。

● 最適化項目

OpenCL for FPGAの最適化は、次の3項目に大き

く分けることができます。

(1) パイプライン・スループットの向上

スループットを向上させることは性能を向上させることと同義です。これ以外の最適化テクニックは、スループット向上のためのお膳立てといっても過言ではありません。スループットを向上させるには多くのリソースが必要となります。

(2) パイプライン・レイテンシの削減

レイテンシは、カーネル・パイプラインの深さ・複雑さと直結しています。これを削減することはリソース消費を減らすことにつながります。リソース消費が減れば、減った分のリソースをスループット向上に割り当てることができます。

(3) メモリI/Oの効率化

データ・インテンシブなアプリケーションでは、メモリI/Oの効率化が非常に重要になります。OpenCL規格には四つのメモリ空間があり、性能特性が異なるため、これらを使い分けることで効率の良いOpenCLプログラムを開発することができます。

2 パイプライン・スループットの向上

パイプライン・スループットを上げるには次の方法があります。

(1) カーネル・パイプラインの多重化

(2) 演算のベクトル化

● カーネル・パイプラインの多重化

`num_compute_units`アトリビュートをカーネル関数に付与すると、カーネル・パイプラインを多重化することができます(図1)。`num_compute_units`アトリビュートには引数を指定することができ、指定した数だけ多重化されます。リスト1はカーネル・パイプラインを二つに多重化する例です。

● 演算のベクトル化

演算のベクトル化ですが、OpenCL組み込み型のベクトル型を使用して記述を行います。512要素の配列の加算がベクトル化される様子を図2に示します。