

超並列計算の自動ハードウェア化でどこまで高速にできるか?

Altera OpenCLでJPEG/MPEG定番の画像圧縮処理をハードウェア化する

山際 伸一 Shinichi Yamagiwa

ここでは、最近の高位合成の流行となりつつあるOpenCLを使った高性能ハードウェア開発に関して解説します。今ではその代表格となりつつあるAltera OpenCLフレームワークに焦点を当て、実際に、OpenCLのソフトウェアとしての開発手法からAltera OpenCLへ適合するプログラムの書き方までを、簡単なアプリケーションを使って性能まで解説していきます。

1. OpenCLとその開発手法

● 超並列計算向け言語OpenCLとは

OpenCLはもともとApple社が提唱していましたが、現在はクロノス・グループ(<http://jp.khronos.org/>)と呼ばれる非営利団体が仕様策定と互換性認証を行っています。クロノス・グループはOpenGLをサポートする団体として有名です。OpenCLはOpenGLとの親和性も高いインターフェースとなっていて、グラフィックス処理も含んだインターフェースを提供しています。つまり、元来、OpenCLはグラフィックス処理の一部の数値計算部分を補助するための機能言語として開発されましたが、現在では、GPUをはじめとするメニーコア・アクセラレータを制御するプログラム言語として有名になりました。

OpenCLが前提とするアーキテクチャを図1に示します。OpenCLはCompute Deviceと呼ばれるホスト・プロセッサからコントロール可能なアクセラレータでプログラム実行することを前提としています。Compute DeviceはDevice Unitと呼ばれる単位の集合でできており、これがプログラム実行をするプロセッサ・コアとなります。

すべてのDevice Unitは共通のメモリ空間を参照することで計算を進めます。このとき、各Device Unitが固有のIDを持ちます(NDRangeと呼ばれる)。この固有のIDはOpenCLのプログラムの中で参照可能になっています。このIDの次元は事前に決めることができます。例えば、2次元の場合は(X, Y)、3次元で指定した場合は(X, Y, Z)のように振られます。これらのIDを使うことで担当部分の計算を実施します。例えば、 $A[i, j] = B[i, j] + c[i, j]$ という行列

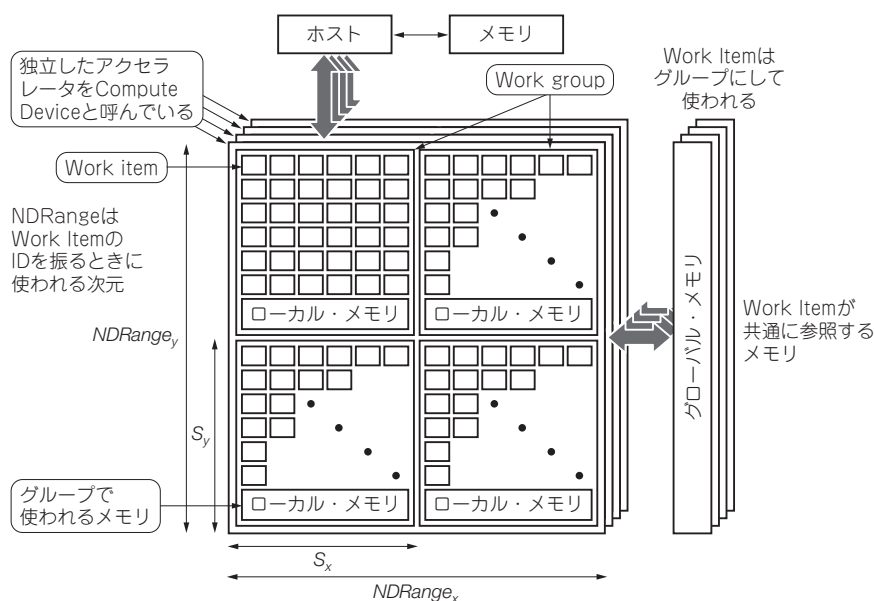


図1 OpenCLが対象とするアーキテクチャ