

Altera社製FPGA Stratix VをOpenCLで開発できる OpenCL for FPGAの概要と活用例

大澤 俊晴 Toshiharu Osawa

昨今、ヘテロジニアスなマルチプロセッサ環境が普及しています。一般的にアーキテクチャの異なるCPUやGPU(Graphics Processing Unit)にはそれぞれにプログラムが必要ですが、OpenCLでプログラムを記述しておけば、CPUでもGPUでもそのほかのアクセラレータでも実行できるようになります。この考えを推し進め、OpenCLで記述したコードでFPGAを動作させることも可能になりました。ここではAltera SDK for OpenCLを使ったOpenCLによるFPGA開発について紹介します。

1 OpenCLの概要

● ヘテロジニアスなマルチプロセッサが普及

OpenCLについて詳しく解説する前に、OpenCL登場の背景から解説しましょう。近年、GPUボードのように異なる種類のプロセッサによって計算機を構成(ヘテロジニアスな構成)するハイブリッド・システムが流行しています。

PCI Express接続で演算処理ボードを搭載するCPU + GPUの他に、PlayStation3のCPU(Cell/B.E.)のPPE + SPEや、ARM SoC(System on a Chip)も異なる演算コアを1チップに収めたヘテロジニアス環境といえます。

これらのヘテロジニアス環境におけるアプリケーション開発環境は各社各様であり統一されていません。

例えば、NVIDIA社が提供するCUDAは、GPUを制御するCPU側を「ホスト」、データ並列処理のみを実施するGPU側を「デバイス」と呼びます。またCPU側で動作するプログラムをホスト・プログラム、GPU側で動作するプログラムをカーネルと呼び区別しています。一般的なソフトウェア開発と最も異なる点は、このGPUで動作するカーネルはCUDA拡張言語を使って記述する必要があるところです。

ソフトウェア開発者から見ると、プラットフォーム開発環境が変化するたびに新たなAPIや言語を習得しなければなりません。ソフトウェアの可搬性を維持しつつ、年々進化するハードウェアへの乗り換えをもっと気軽にできるようにならないだろうか。その問いに答えてくれるのがOpenCLなのです。

● OpenCLとは

OpenCL(Open Computing Language)をできるだけ簡潔に表現すれば、「並列コンピューティング環境のためのフレームワーク」となります。単なる一つの並列プログラミング言語と思われがちですが、OpenCLの中身はOpenCL C言語だけでなく、OpenCL C言語によって書かれたプログラムを動作させるための

枠組みも含まれます。

OpenCLは、Khronos Groupという非営利団体によって仕様の標準化作業が行われています。この標準化メンバには、主要なプロセッサ・ベンダやマルチコア・ソフトウェア・ベンダが名を連ねており、CPU、GPU、DSP(Digital Signal Processor)、FPGAといったあらゆる種類のプロセッサによって構成される並列計算機環境において、共通的に使用できるコンピューティング環境を、OpenCLで実現できるようになっています。

フィクスターズもこの標準化グループに所属し、仕様策定にかかわっています。なお、2013年11月現在の最新の正式版はOpenCL 1.2となっています。

● OpenCLの概要

OpenCLは、並列計算機を構成するプロセッサの種類やベンダに依存することなく、統一された並列ソフトウェア開発環境を実現するために、次の二つの仕様を標準化しています。

(1) OpenCL C言語仕様

演算用プロセッサで動作する、C言語を拡張した並列プログラミング言語

(2) OpenCLランタイムAPI仕様

OpenCL C言語で作成したプログラムを演算用プロセッサで並列実行させるために、制御用プロセッサが利用するAPI

● OpenCLに必要な道具

具体的に「OpenCLを使う」とはどういうことでしょうか。OpenCLでソフトウェア開発するときに利用する道具は次の二つとなります。

(1) OpenCLコンパイラ

OpenCLコンパイラは、OpenCL C言語で作成した演算用プロセッサ側で動作させるプログラムをバイナリ・コードに変換します。

(2) OpenCLランタイム・ライブラリ

制御用プロセッサで、CあるいはC++から呼び出し可能なライブラリ形式となっています。

ソフトウェア開発者は、まず、演算用プロセッサで