

コンフィギャラブル・プロセッサ を理解する

——コンパイラ検証のための基礎知識

小林真輔

対象となるアプリケーションに特化したカスタム命令を実装できるコンフィギャラブル・プロセッサが複数のベンダから提供されている。ここへ来て脚光を浴びているコンフィギャラブル・プロセッサだが、このチップを利用する場合、検証が難しいという問題がある。ユーザは、生成されたプロセッサの動作だけでなく、コンパイラなどのソフトウェア開発環境の動作についても検証・評価を行う必要がある。 (編集部)

コンフィギャラブル・プロセッサとは、ある環境やツールを用いて命令セットなどをカスタマイズできるプロセッサのことです。設計者は用途に応じてカスタマイズすることで、要求する仕様に合ったプロセッサを設計します。プロセッサと命令セットだけでは、その上で動作するプログラムを開発することは難しいため、通常、カスタマイズしたプロセッサ用のソフトウェア開発環境(命令セット・シミュレータ、Cコンパイラ、アセンブラ、リンカなど)も同時に提供されています。

1. 柔軟性と性能要求のジレンマ

では、どのような場合にコンフィギャラブル・プロセッサを用いればよいのでしょうか。例えば、次のような場合を想定してみましょう(図1)。

新しい動画の規格に対応した機器を開発することになりました。仕様のほとんどは決まっていますが、今後、規格の決定過程において変更される可能性があります。しかし、会社としては製品をいち早く開発して製品展開を有利に進めたいので、ASICを設計している余裕はありません。そこで、ソフトウェアで機能を実現することを前提にしたいと考えました。ところが、市販のプロセッサだけでは要求

性能を満たせないことがわかりました。プロセッサをチューニングして、できるだけ性能面で有利なプロセッサを使いたいと考えています。

このように、システムに柔軟性を持たせつつ、さらに厳しい要求性能を満たしたいという場合に、コンフィギャラブル・プロセッサは有効です。コンフィギャラブル・プロセッサを用いると、ソフトウェアによる実装を行いつつ、対象とするアプリケーションに最適なカスタム命令を組み込むことで、高い性能を実現できます。

●自分の応用に最適な設計を求める

コンフィギャラブル・プロセッサを設計するうえで重要な問題は、対象となるアプリケーションを効率良く動作させる命令やプロセッサ構成を探し出すことです。このことを設計空間探索(design space exploration)と呼びます。

コンフィギャラブル・プロセッサを用いた設計フローを図2に示します。最初に、システムの要求仕様からプロト

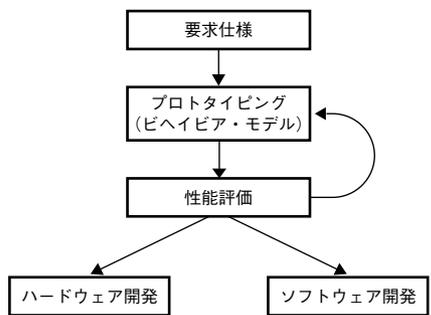


〔図1〕システム設計時のジレンマ
柔軟性も欲しいが、性能も欲しい…。

タイプを作成します。ここで言う「プロトタイプ」とは、システムの動作を解析するために用いるビヘイビア・レベルのシミュレーション・モデルのことです。この段階で、プロセッサのプロトタイプを作成し、そのプロセッサ(モデル)を用いて対象となるアプリケーション・プログラムを動作させること(いわゆるハードウェア・ソフトウェア協調検証)で、プロセッサの性能評価を行います。

性能評価を行った結果、目標を達成できる場合は個々のハードウェアやソフトウェアの詳細設計を行います。目標を達成できない場合は、プロトタイプを変更して、再度、性能評価を行います。

このように、さまざまなプロトタイプを作成して評価す



〔図2〕 コンフィギュラブル・プロセッサを用いた設計フロー
 コンフィギュラブル・プロセッサであれば、プロトタイピングと性能評価を繰り返して最適なプロセッサを設計することが可能である。

ることで、目標に合ったプロセッサを探索していくこととなります。従来、プロセッサはすべて人手で開発されていたため、このような探索を十分に行う時間はありませんでした。コンフィギュラブル・プロセッサでは、構成を容易に変更できるビヘイビア・モデルを用いることで設計空間探索を効率良く行うことができます。

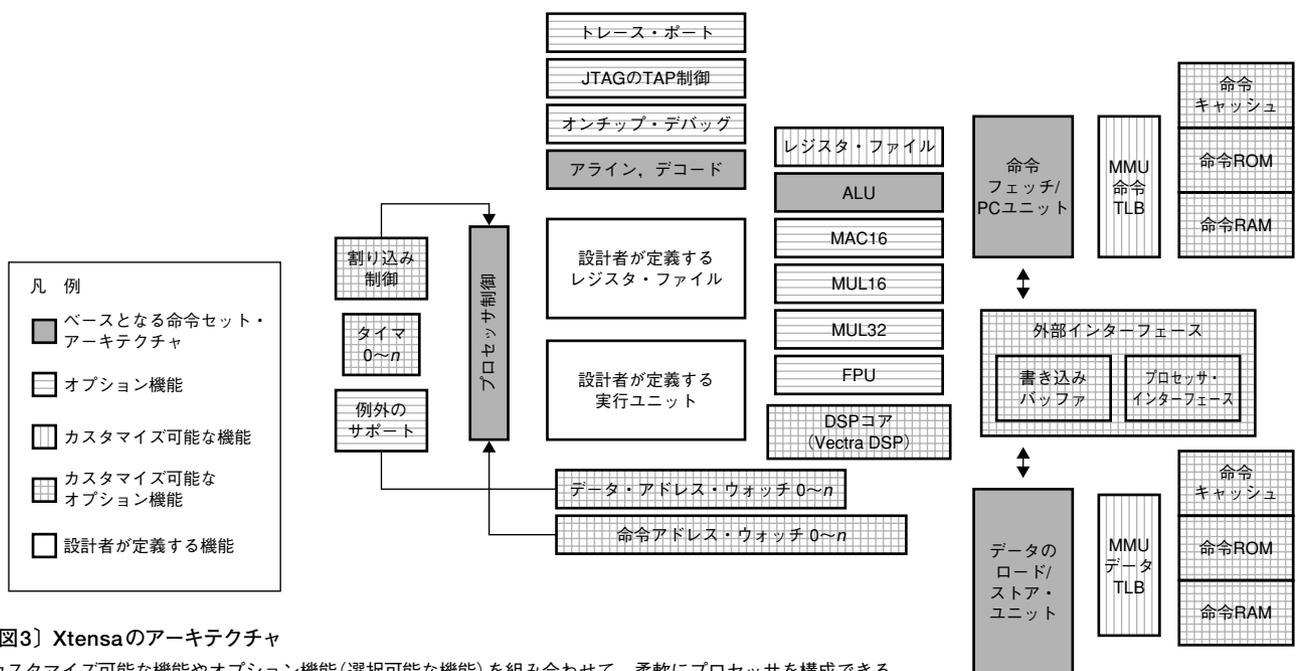
●命令追加方式と言語記述から生成する方式がある

コンフィギュラブル・プロセッサは、開発方法の違いによって以下の2種類に分類できます。

1) ベース・プロセッサ型

ベース・プロセッサ型は、あらかじめベースとなるマイクロプロセッサが与えられており、これに対して命令やレジスタを追加する方式です。この方式では、あらかじめ用意された最低限の命令セットに対して、アプリケーションに適した命令をユーザが追加していきます。

ベースとなるプロセッサがすでに用意されているため、Cコンパイラや命令セット・シミュレータなどを開発しやすいというメリットがあります。また、プロセッサ開発に必要な時間は比較的(後述するプロセッサ仕様言語型と比べて)短くなります。しかし、あらかじめベース・プロセッサが決められているため、柔軟性に欠ける部分があります。この方式のコンフィギュラブル・プロセッサとしては、



〔図3〕 Xtensaのアーキテクチャ
 カスタマイズ可能な機能やオプション機能(選択可能な機能)を組み合わせ、柔軟にプロセッサを構成できる。