

プログラマブルな布線論理とPARTHENONによる

新しいコンピュータ・アーキテクチャの実現を探る

小栗 清

1. コンピュータの性能向上

プログラム可能な装置で、高速に計算や処理を行いたいという要求がコンピュータ技術の発達の原動力となっています。プログラマブルな計算や処理の中心となっているCPUの性能に関しては、もう性能向上は難しいのではないかと、そのときどきの予想を超えて性能の向上が続いています。しかし、コンピュータの高速化技術は10年以上ほとんど変わっていません。

RISCやスヌーピング、投機的実行など目新しい言葉(概念)が少しずつ出てきていますが、性能向上のほとんどはLSIの集積度の向上とバランスの調整の結果で得られたものです。特に大規模なメモリの物理的なサイズが小さくなったことがもっとも性能向上に寄与しています。

コンピュータをもっとも単純に考えるとMEM(メモリ)とMEMに接続されたALU(算術論理演算ユニット)ということになり、その概念の基本はメモリです。メモリ・アクセスは単一の動作であり、並列性を持ち込むことはできません。

コンピュータの動作はこのメモリ・アクセスが逐次的に行われることを基本としているので、コンピュータの性能向上はメモリ・アクセスの高速化が中心となります。

高速化のための二大技術は、
(1)メモリのキャッシュ技術
と、キャッシュへのアクセスを休みなく行うための、
(2)パイプライン技術
です。

マルチプロセッサ・システムのようにメモリ・アクセスの要求元が複数の場合はインタリーブ・マルチバンク技術やキャッシュの一致処理技術も重要となります。

並列性を持たないメモリ・アクセスの高速化だけで、CPUのこれだけの高速化が達成されたことは驚異ですが、並列性の導入による有効な高速化が強く期待されているのも事実です。

2. プログラマブルな構成での性能向上

いっぽう、プログラマブルでなくてもよいのであれば、つまり、ある問題に特化して性能向上を行うのであれば、もっといろいろな工夫ができるはずです。

原理的にはこちらのほうがより高性能を達成できそうですが、現実には逆のことが多いようです。その理由はすべて経済的なところにあります。

プログラマブルであることは汎用であることであり、利用者が多く、その結果、大勢の人たちが競って改良を続けま

す。このことが、特定の問題だけを高速化すればよいという利点を越えてしまうのです。

まだまだ性能向上が止まりそうにないCPUとMEMですが、さらなる性能向上のための方策を考えておく必要があります。すぐ思いつくのは専用ハードウェアや並列計算機、そして超高速素子の開発などでしょう。その場合にプログラマブルであることは欠かせない条件となります。したがって専用ハードウェアによる性能向上は非常に狭い範囲に限らざるをえないと思われてきました。

3. プログラマブルな布線論理

ある論理的機能を実現しようとするとき、さまざまな方法を探ることが出来ます。もっとも手軽なのはCPUとMEMによるプログラム論理でしょう。

CPUとMEMを用意しておけば物理的な構成を変えないでさまざまな機能を実現することが出来ます。

論理回路により直接機能を実現する布線論理(ワイヤード・ロジック)を使えば、より高性能なものを得ることが出来ます。しかし、物理的にものを作る工程が必要となり、手軽というわけにはいきません。

CPUやMEMそして布線論理を実現するためのLSIは、今まで図1のように分類されてきました。

まず、内部の構造からメモリLSIと論理LSIが区別されます。次に、論理LSIは特定の機能を実現しているかどうかにより、汎用LSI(CPUやDSPなど)と特定用途向けLSIに区別されます。さらに特定用途向けLSIは、大量に使用するかどうかにより選択される製造方法である、フルカスタム、スタンダード・セル、ゲートアレイに分類されてきました。

フルカスタムはトランジスタの大きさや配置のレベルで詳細に最適化したもの、スタンダード・セルはあらかじめ用意された機能セルの組み合わせと配置を詳細に最適化したもの、ゲートアレイはあらかじめ配置されたゲート間の結線のみを最適化したものです。

ところが、これらのカテゴリとは異なるFPGA(Field Programmable Gate Array)と呼ばれる、メモリLSIと論理LSIに続く第3のLSIアーキテクチャが出現しています。

FPGAは、布線論理でありながらユーザの手元で機能変更が可能で、内部の構造はメモリに近いという特徴を持っています。この特徴は図1の×印の位置を占め、従来の分類にはあてはまらないこと

がわかります。これはFPGAでの布線論理の基本単位が小規模なメモリであるからです。

このFPGAの出現により、先に必須条件としたプログラマブルである範囲が大きく変わることを指摘しておく必要があります。すなわち、専用回路による性能向上が有効となるベースができたということです。

4. FPGAの構成と特徴

ここでFPGAの構成をまとめておきます。FPGAの布線論理はLUT(Look Up Table)と呼ばれる小規模メモリ[アドレスが数ビット(3ビットから5ビット程度)、出力が1ビットのメモリ]を基本単位として、これらを多段に接続することにより構成されています。

LUTはアドレスが4ビット(16ビットのメモリ)の場合でも2の2乗(4乗)乗、すなわち2の16乗(65536)個の論理関数のうちの任意の一つを表現できます。

これまでの論理LSIの基本単位である論理ゲート(ANDゲートやORゲートなど)が単純な論理関数を表現しているこ

とくらべると、かなり趣が異なります。

図2には4変数の論理関数のうちの三つ(A, B, C)を示します。定義域は4変数ですから0000から1111までの16通りで、値域は0か1です。定義域の各値に対して出力の値が何であるかを表すグラフが関数を表します。

Aはandを表す論理関数で、入力が1111のときのみ値が1でそれ以外は値が0となります。Bはorを表す論理関数で、入力が0000のときのみ値が0で、それ以外は値が1となります。

LUTはある定義域に対して関数の値を1とすることも0とすることもできるので、2の16乗種類の関数を表現することができるわけです。CはLUTで表現できる関数の一例です。

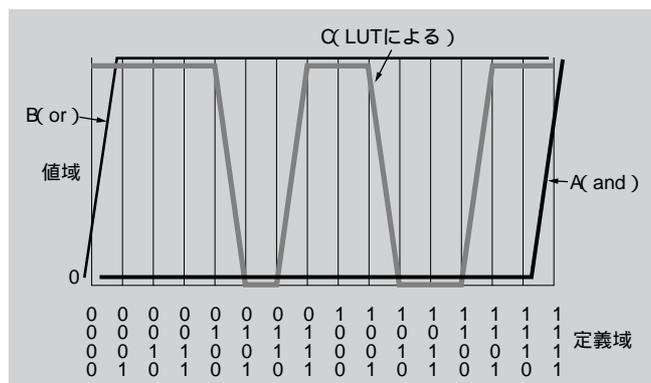
FPGAは図3のようにアレイ状に配置された多数のLUTを、これまた構成が変更できる相互接続部で結合する構造となっています。

相互接続部は現状では2階層または3階層とされることが多く、LUTには1ビットのレジスタが接続してあることが多いようです。

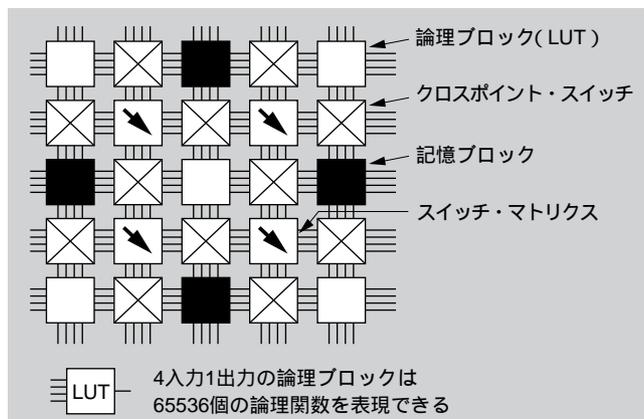
また、布線論理の構成要素は結局、デコーダや加算器、インクリメントなどであるとの考えのもとに、キャリアやカスケードのための専用の構造を用意したり、さらには、乗算テーブルなどに利用できる、もう少し規模の大きいメモリ(アドレス8ビット、出力8ビット程度)を構成要素に含めることもあります(エンベ



〔図1〕LSIの分類



〔図2〕4変数の論理関数



〔図3〕FPGAの内部構造