

# LSI設計のコスト感覚と回路の考え方

LSIのメリットを引き出す全体設計



森岡澄夫

この章からは、開発者がハードウェアで処理を組む時にどのような考え方をを用いるのか、そこにどのような理由・背景があるのかを説明する。このような知識は、自動設計ツールを上手に使っていくうえで必須である。人手による設計がツールに頼った設計より良くなる場合が多いのは、ゲート・レベルなど低い抽象度で細かい最適化をするのが主な理由ではない。設計対象に応じて、ハードウェアのメリットを引き出す全体設計をすることからである。

(筆者)

## 1. ハード設計におけるコスト感覚

この章では現在のハードウェア(LSI)設計におけるコストの感覚について説明し、それを土台にしてハードウェア全体構成の概要へ話を進めます。具体的な例は、第3章と第4章で説明します。

### ● ハードの性能は処理内容によって著しく違う

第1章で述べたように、「ソフトウェア処理をハードウェア化する」という設計手法のニーズが最も高いのは、画像処理やセキュリティ処理などに代表されるデータ演算処理です<sup>1)</sup>。パソコン上でソフトウェアを組んだり、MATLAB(米国The MathWorks社)などのモデリング・ツールを使って製品に搭載するアルゴリズムの検討・評価を行っていく過程で、高速化、低消費電力化、知的情報保護などの必要性が見えてきて、LSI化を考え始めるケースは多くあります。また、最初からLSIの開発がゴールであっても、手軽なソフトウェアで処理を組んで試すところから研究、開発を始めるのも、よくある話です。

しかし、高速化一つ取っても、パソコン上で実行していた処理をそのままLSI上へ持ってきては、少しも速くならない場合が出てきます。この事実は、ハードウェア設計が初めてという人には意外に思えることでしょう。

図1に、いろいろなデータ処理をLSI化したときに得られるスループット(単位時間当たりのデータ処理量)を、同一クロック周波数のパソコン用ソフトウェアとの相対値で

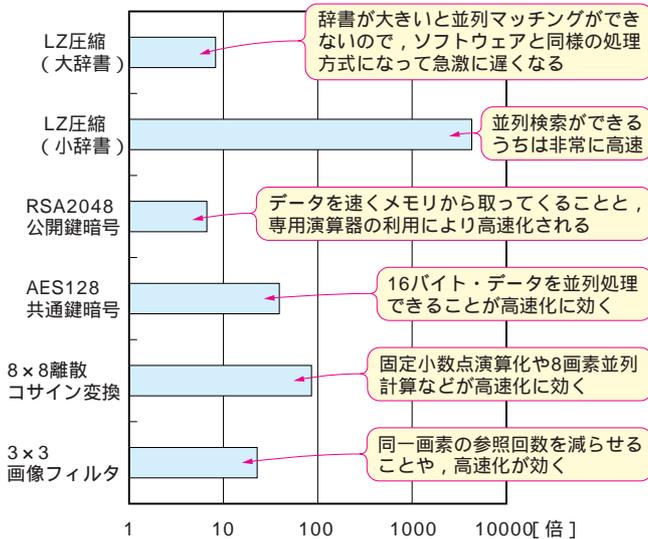


図1 各種データ処理をLSI化したものと、同一クロック換算のパソコン用ソフトウェアとのスループット比較

少し前の0.18 $\mu$ m～90nmプロセスのスタンダード・セルASICを想定したもの。パソコン用プロセッサの方が10倍ほど動作周波数が高いので、10倍のラインを下回っていると、実際のパソコンには負ける可能性がある。また、組み込み用プロセッサの場合、(同一周波数の)パソコン用プロセッサよりは遅いので、この図よりもう少し倍率が上がる。

### Keyword

全体設計, LSI, コスト, IPコア, 並列処理, ビット幅, 基本処理ユニット, 演算器, 処理構成図

示します<sup>2)</sup>。測定結果は設計や実装方法に大きく左右されるので、あくまで参考程度にとらえてください。ただし、非常に速くなる処理とそうでない処理の差は歴然としていることが分かります。

図1のデータは、ハードウェアで使う処理アルゴリズムや回路構成を十分最適化した場合のデータです。ソフトウェア処理をそのまま何も考えず LSI 化すれば、図1の性能よりもけた違いに悪くなる可能性が高くなります。

明らかに、「ハードウェア化するだけで何でも速くなるわけではない」といえます。以下、図1のような違いが現れてくる理由について、いくつかの視点から考えてみます。

### ● 専用 LSI の不得意なところ

表1に、専用 LSI(スタンダード・セルを用いた ASIC)を設計する上でのコスト感覚を、パソコン用ソフトウェアのそれと対比して示します。

パソコン用ソフトウェア処理をそのまま持って来ようとしたときにぶつかる障壁を、次の通り読み取ることができます。パソコン用ソフトウェアを組み込み用プロセッサへ移植する場合とも似ています。

#### (1) クロック周波数に厳しい制約がある

場合によりますが、クロック周波数がパソコンの1/10未満になってしまいます(FPGA ならば、さらにその数分の1)。まったく同じ処理アルゴリズムならば、それだけ実行が遅くなるわけで、これは結構なペナルティとなります。

なぜそれほどクロック周波数に差がつくのか不思議に思うかもしれませんが、これは、使える技術や設計にかけられるコストが同じではないからです<sup>注1</sup>。

#### (2) 使用可能なメモリ量に厳しい制約がある

一つのチップ上に何Gバイトものメモリを搭載することはできません。搭載可能なメモリ容量は、パソコンの1/1000未満になってしまいます(どんなに多くても数十Mバイトであり、数十Kバイト程度のこともある)。

画像処理などではこの制約はクリティカルで、これだけ

注1：同じプロセス・ルールであっても、パソコンなどの汎用プロセッサはスタンダード・セルなど使わずカスタム設計がなされるので、最大クロック周波数はASICと比べて高くなる。普通のシステムLSI全体をカスタム設計するのは、工数や工期などのコスト面で非現実的である。また、ASICのクロック周波数はチップの全体設計(特にチップ入出力インターフェースやオンチップ・バス)に強い束縛を受けて決まってくるし、チップの設計管理や歩留まりなどの点からも、一部の回路だけを特別に高い周波数で動かすような設計には無理がある。

注2：1個のプロセッサ・コアによるソフトウェア処理では、例えば複数プロセス/スレッドのマルチタスクといっても、実際にはある瞬間に走っている処理はわずかである。ハードウェアの場合は、同じ瞬間にほんとうに複数の処理が走っている点が異なる。

で処理の大幅見直しを余儀なくされる可能性があります。

### (3) 回路規模にも制約はある

LSIの製造プロセスが微細化して、コストさえかければ1億ゲート単位の回路でも1チップ化できる時代です。しかし、設計時にゲート数を気にしなくてよくなったわけではありません。

データ処理が主体の機能ブロック(IPコア)では、設計が良くないと、数万ゲート~数十万ゲート単位で規模が増加します<sup>1)</sup>。一つのLSIが10個かそれ以上のIPコアを搭載しているとき、各IPコアが10万ゲートを無駄にすれば、全体では100万ゲート以上の無駄になります。

また、先端のプロセスを使ってゲートも湯水のごとく使える、といった恵まれたLSI設計は珍しい方です。大量生産される民生機器向けのチップの場合、価格にも厳しい制約があります。このため2~3世代前の製造プロセスでコツコツ頑張っている設計部隊が多いのが実情です。そこでは、回路規模は常に重要な問題となっています。

### ● 専用 LSI の得意なところ

いっぽうで表1には、ハードウェアに向けた処理の組み方のヒントも含まれています。

図1において高い相対性能を達成できている処理では、以下のようなハードウェアの利点をうまく生かしているのです<sup>3)</sup>。

#### (1) 処理の並列性を非常に高くできる

並列処理は、ハードウェアの最大の強みです。扱うデータ量が比較的少量ならば、複数個のデータの参照、演算、更新を同時に並列して行えるので劇的に速くなります<sup>注2</sup>。

表1 専用ハードウェア(ASIC)の設計におけるコスト感覚

黄色い部分は有利な点を表す。

	パソコン用ソフトウェア	専用 LSI(最近の65nm ~ 180nm ASIC)
クロック周波数	1G ~ 3GHz	数十M ~ 数百MHz
使用可能メモリ	数百Mバイト ~ 4Gバイト	数十Kバイト ~ 数十Mバイト(オンチップの総量)
使える電力	数十 ~ 200W前後	数mW(モバイル機器) ~ 数W(据え置き機器)
回路設計方式	カスタム設計(プロセッサ)	スタンダード・セルやゲート・アレイを利用
回路価格の制約	数万円(プロセッサ)	数円 ~ 数千円
処理ビット幅	32ビットないし64ビット	1 ~ 数百ビットまで任意
基本数値演算	論理演算、整数四則、浮動小数点数四則(プロセッサ)	ほとんど制限なし
処理の同時並列性	処理粒度にもよるが、それほど高くはない	回路構成次第で非常に高い並列性