

# 組込みシステムとハードウェア/ソフトウェア・コデザイン

大阪大学 大学院基礎工学研究科 情報数理系専攻 今井 正治

## ■ はじめに

半導体集積回路の集積度は、1970 年代以来 18 ヶ月で2倍という高い割合で伸びている。現在では数千万トランジスタ規模の商用マイクロプロセッサも製造されている。この傾向は今後 10 年程度は継続すると見られており、米国 SIA (Semiconductor Industry Association) の技術予測によれば、約 10 年後には 10 億トランジスタ規模の集積度を持つマイクロプロセッサが実現可能になると見られている[1]。

このような技術を用いると、従来複数のコンポーネントで構成されていた複雑な電子システムが単一のシリコンチップ上に集積化して実装可能になる。いわゆるシステム LSI (または、システム・オン・シリコン) 時代の幕開けである。システム LSI 上には、図 1 に示すように、プロセッサコア、メモリ、周辺回路、FPGA(Field Programmable Gate Array) などのデジタル・コンポーネントだけでなく、アナログ回路や高周波 (RF) 回路などのコンポーネントも集積化可能になりつつある。システム LSI 技術を用いれば、組込みシステムを高性能化できるだけではなく、小型化・軽量化・低消費電力化することも可能になる。

## ■ HW/SWコデザインとは

システム LSI の設計を行うためには、ハー

ドウェアのみならずソフトウェアも含めて最適化を行う必要がある。システム LSI は性質の異なる複数種類のコンポーネント(プロセッサ、メモリ、専用ハードウェア回路など)から構成される大規模かつ非均質 (heterogeneous) なシステムだからである。したがって、それぞれのコンポーネントを個別に最適化したとしてもシステム全体の最適化が達成できるとは限らない。

設計の品質は設計品質指標 (design quality metrics) [2] によって評価できる。設計品質指標の中で現在重要視されている項目は以下の 3 つである。設計品質指標に含まれ得る項目には、下記以外にも、テスト容易性、パッケージのピン数などがある。コデザイン手法は、これらの設計品質指標を定量的に評価して最適化を行う設計手法である。

### (1) 面積 (ハードウェア・コスト)

チップ面積は、システム LSI のハードウェア・コストに直接影響を与える指標である。チップ面積は最終的には製造後の製品のダイサイズによって評価できるが、設計の途中の段階では、トランジスタ数やゲート数、またはこれらと配線領域の大きさなどによって見積りが行われる場合もある。

## (2) 性能

特定の応用を考えた場合のシステムの性能は、システムのクロック周波数( $F_c$ )とその応用を実行するのに必要なクロックサイクル数( $T_{ex}$ )を用いて次の式で評価される。

$$\text{性能} = F_c / T_{ex} \quad (1)$$

式(1)で定義される性能は、特定の応用についての評価である。「汎用」プロセッサのように複数の応用を対象とする場合には、これらの応用に対する性能はそれぞれの応用に対する性能を加重平均することによって評価できる。

## (3) 消費電力

バッテリーで駆動される携帯用の情報処理機器・通信機器などでは、消費電力が重要な評価尺度となる。また、商用電源などで駆動される製品でも発熱によるVLSI内部およびパッケージの温度上昇が大きな問題となりつつあるので、消費電力が重要な評価尺度となる。前者と後者の違いは、前者では平均消費電力の低減が重要であるのに対して、後者ではピーク時消費電力がより大きな問題となる点である。

設計最適化問題は、上記の設計品質指標を用いて定式化できる。システムLSIの設計最適化問題は上記の設計品質指標のうちの一つを目的関数とし、他の2つを制約条件として、面積最小化問題、性能最大化

問題、消費電力最小化問題の3種類に分類できる。

## ■ 理想的なコデザイン手法

理想的なコデザイン手法の概念図(設計フロー)を図2に示す。同図のコデザイン手法は、次のような特徴を持っており、現在のコデザイン手法よりも広い範囲での実現方法(設計空間)の探索が可能になり、設計品質がより高いシステムの設計が可能になる。

- (1) 設計対象システムは、単一の言語で記述される。
- (2) 設計の機能分割は、設計品質指標の定量的な見積りに基づいて行われる。機能分割は設計者が行っても良いし、最適化アルゴリズムを用いて自動分割を行っても良い。
- (3) 機能分割の結果は、CPUの記述、専用ハードウェアの記述、およびアプリケーションプログラムの記述の3種類になる。
- (4) CPUの記述から、そのCPUのオブジェクトコードを生成するコンパイラを生成する。また、CPUの記述から、CPUと専用ハードウェアのインタフェース回路のHDL記述が生成される。
- (5) CPUおよび専用ハードウェアの記述は論理合成可能なHDL記述として出力される。
- (6) アプリケーションプログラムのオブジェクト・コードは、(1)で述べたシステムの記述の一部を(4)で生成されたコンパイラを用

いてコンパイルすることによって生成される。

- (7) CPU は IP ライブラリの中であらかじめ用意されたモデルの中から選んでも良いが、応用分野により特化した CPU を新たに設計することも可能である。

## ■ おわりに

本稿では組込みシステムとその設計手法として注目されているハードウェア/ソフトウェア・コデザインについて述べた。組込みソフトウェアは組込みシステムの重要な要素であるにもかかわらず、わが国ではハードウェアの設計に従属して(?)開発されている、と言えなくもない。すなわち、開発スケジュールの遅れやハードウェア部分の機能の不具合などの、さまざまなトラブルの最後の帳尻を合わせるのはソフト屋さん、という構図が出来てしまっている。このような状況を変え、システム開発を効率良く行うためには、コデザインというシステムマティックなシステム設計手法が必要不可欠である。

現在の商用コデザイン・システムは、マニュアルでの機能分割の支援を行うツールという性格が強く、機能分割を自動的に行うというレベルには達していない。しかし徐々に機能分割の支援から自動分割へと研究が発展していくものと思われる。また、現在の

コデザイン・システムの多くは、ハードウェアの記述と応用ソフトウェアの記述に別の言語(たとえば HDL と C 言語)を用いている。しかし今後は、ハードウェアとソフトウェアを区別せず単一の言語でシステムの機能を記述しておき、このシステム記述から論理合成可能な HDL 記述と応用プログラムのオブジェクト・コードを生成するという方向に向かうと思われる。今後の技術の発展に期待するところが大きい。

### 参考文献

- [1] Semiconductor Industry Association: *The National Technology Roadmap for Semiconductors – Technology Needs*, 3rd Edition (1998).
- [2] D.D. Gajski, F. Vahid, S. Narayan, J. Gong: *Specification and Design of Embedded Systems*, Prentice Hall (1994).
- [3] 今井 正治、武内 良典: 「ハードウェア/ソフトウェア・コデザイン手法」、電子情報通信学会誌、Vol. 81、No. 11、pp. 1132-1138 (1998年11月)。

(いまい まさはる)

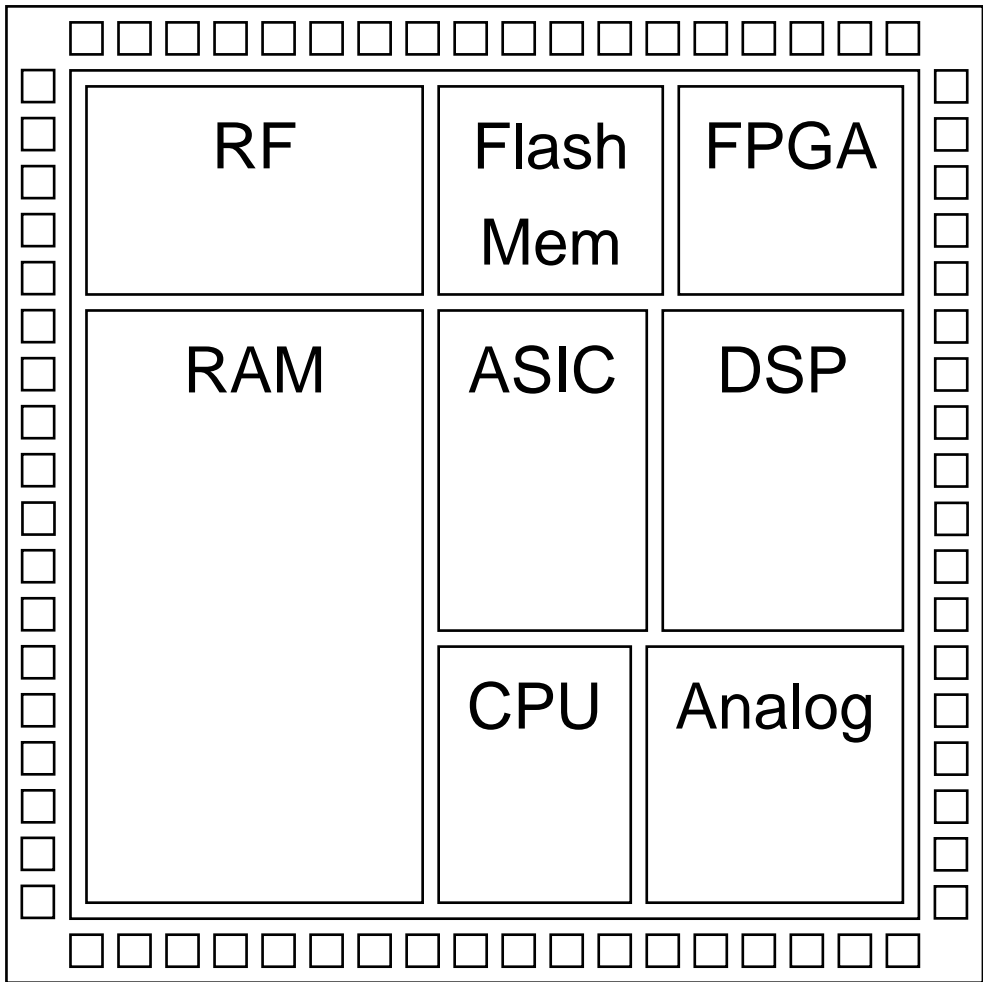


図 1 システムLSIの構成要素

Fig. 1 Components of a System LSI

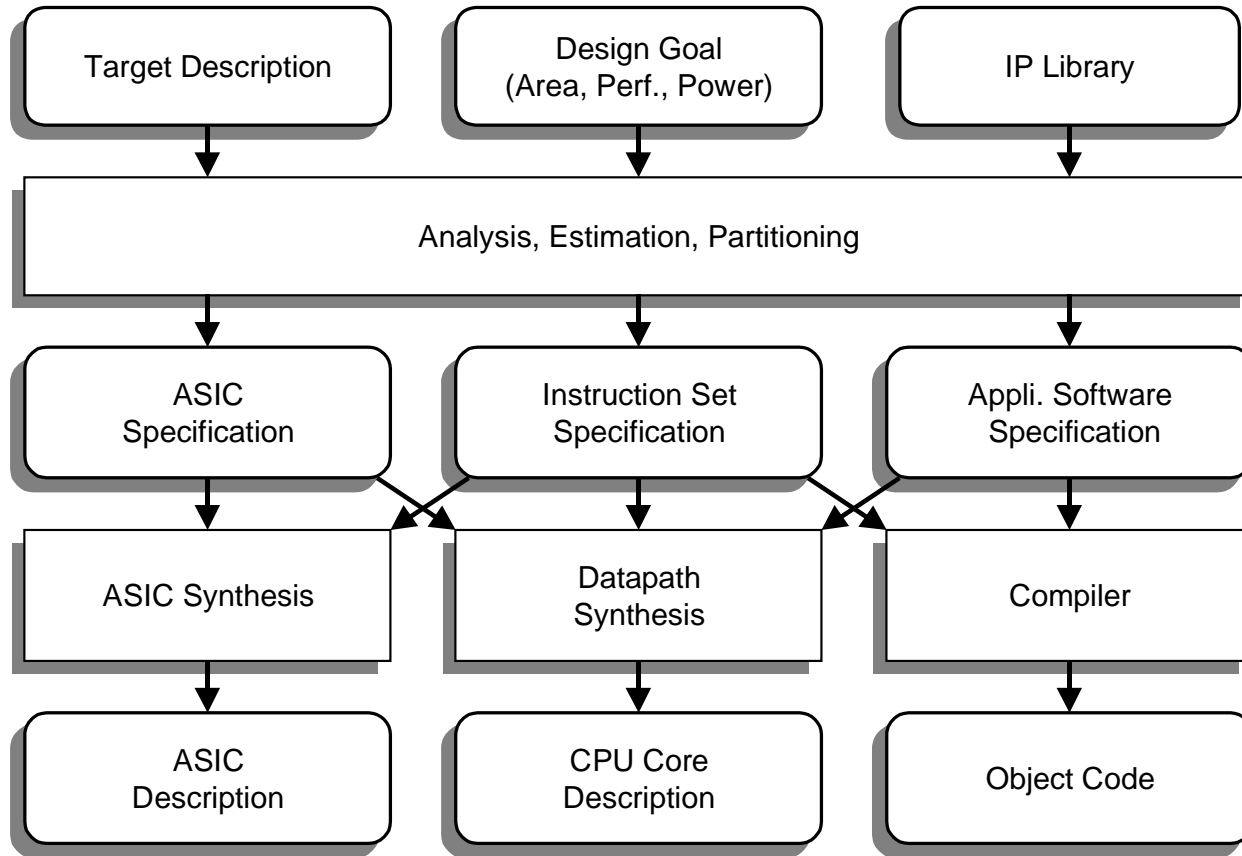


図 2 理想的なコデザイン手法での設計フロー

Fig. 2 System Design Flow in an “Ideal” Codesign Method