

《SoCの設計とプロセスは完全分離しているか?》

## ファブレスとファンドリーの協調関係 擦り合わせに最適な組織関係が鍵



働エフエーサービス 半導体事業部 技術主幹 湯之上 隆

SoCにおいて、設計とプロセスが完全分離したことにより垂直統合型（IDM）が衰退し、ファブレスとファンドリーが成長したという説がある。しかし、微細化が進むほど、Design for Manufacturing（DFM）が必要不可欠になってきており、設計とプロセスを完全分離するという理想的形態は、全く不可能となっている。実際に、SoCの各設計間および設計とプロセス間には、密接な擦り合わせが必要である。Taiwan Semiconductor Manufacturing（TSMC）は、ファブレス、IPベンダー、設計ツールベンダーとともに、極めて巧みに擦り合わせることができる仕組みを構築し、日本IDMを凌駕した。日本IDMは、“擦り合わせが上手くいくような組織体”を、初心に戻って構築するべきである。

### 世間の半導体産業に対する認識

世間では、半導体産業を次のように認識している。

LSIの製造技術は各種製造装置に一体化されているため、世界の主要装置メーカーから最先端装置を購入して据え付ければ、容易に最先端LSIが製造できる（要するに、装置を買って並べてボタンを押せば最先端LSIが製造できる）。

韓国Samsung Electronicsは、資本力、迅速で断固たる経営判断、政府支援、およびウォン安により、世界最大のメモリーメーカーになれた。

SoCにおいて、設計とプロセスが完全分離したことにより、垂直統合型（IDM）モデルが衰退し、ファブレス・ファンドリーモデルが成長した。

本誌の読者の方々は、どのような意見をお持ちであろうか？

### 全て誤認識

については、少しでも半導体の生産に関わった人であれば、「そんな馬鹿な」と思うことであろう。しかし、半導体業界を外側から眺めた場合、高額な装置を買って並べている量産工場とその歩留り程度のことしか認識できない。装置と装置の間をつなぐインテグレーション技術は見えない。その結果、著名な大学に在籍する著名な社会学者ですら、その著作の中で、堂々とこのような論説を主張している<sup>2)</sup>。これが誤認識であることを、筆者は学术论文<sup>3)</sup>や既報<sup>4)</sup>で詳述してきた。

については、Samsungの競争力の源泉として上

に挙げた要因が関係していることは否定しない。しかし、これだけではない。コンピュータ業界がメインフレームからPCへパラダイムシフトしたことを正確に捉えたマーケティング力<sup>5)</sup>、それを基にPC用のDRAMを安く大量生産した破壊的技術<sup>6)</sup>、それらを実現した組織能力などの要因が抜けている。むしろ筆者は、後者の方が重要性は高いと考えている。

については、どうだろうか？ファブレスとファンドリーの分業構造が上手く機能していることから、一見正しいように見える。これらを論じた学術論文や書籍も出版されている<sup>7)</sup>。しかし、実はこれも正しいとは言い難い。

そこで本稿では、の認識が正しくないこと、すなわち、SoCの設計とプロセスは完全分離などしていないこと、および、日本IDMよりもファブレスとファンドリーの方が、組織的に巧みに擦り合わせを行える仕組みを構築していることを示す。

### SoCの設計と製造フロー

図1に基づいて、SoCの設計と製造フローを説明する<sup>8)</sup>。

#### システム設計

SoCの設計では、まず製品のアーキテクチャを決定するシステム設計が行われる。最初に製品の要求仕様に基づいて、システムのビヘイビア（動作）が記述され、次にアーキテクチャ設計により、どの動作をソフトウェアで実現し、どの動作をハードウェア（LSI）で実現するかを決定する。

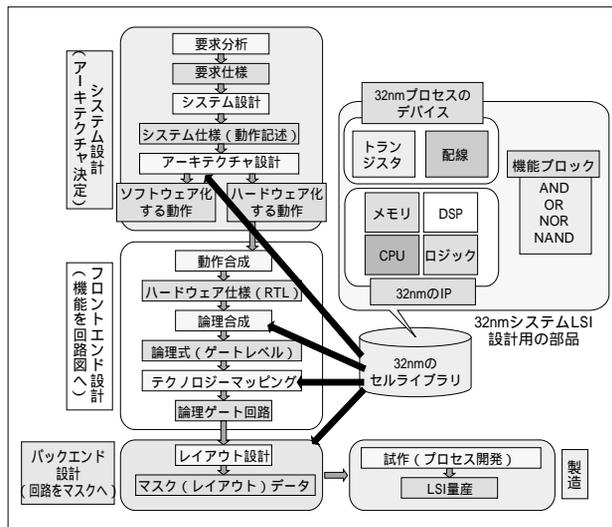


図1 SoCの設計・製造フロー

### フロントエンド設計

ハードウェア化する動作が決まると、それを基に、LSIの機能を回路図に落とし込むフロントエンド設計に移る。そこでは、まずハードウェア化する動作を、記憶素子であるレジスタとデータ処理(演算)で記述する。この記述方法をレジスタ転送レベル( Register Transfer Level : RTL )と呼ぶ。

次に、論理合成により、RTLをセルライブラリ中のAND、OR、NOR、NANDの機能ブロック単位の論理式に変換する。機能ブロックは、0と1の2進数であるゲートから構成されることから、この論理式をゲートレベルという。このゲートは、LSIのトランジスタに対応する。この論理合成の際、演算時間、遅延時間、占有面積、消費電力などの具体的なゲート性能が制約条件として入力され、その結果、SoCの性能がほぼ決定される。

さらに、このゲートレベルで記述された論理式に対して、セルライブラリの中に準備されているセル(IP)の中から最適な回路ブロックを選択して割り当てる(テクノロジーマッピングという)。

以上によって、機能および動作が具体的な回路図に変換される。このように、回路図の設計は一見、既成コンポーネントの組み合わせの外観を呈している。

### バックエンド設計

回路図が完成すると、セルをチップ上に配置し、配線をつなぐレイアウト設計(バックエンド設計)を行う。以上により、回路図はマスクデータに変換され、これを基に20~30枚、場合によっては50枚を超えるフォトマスクセットが製造される。

### 試作・製造

このフォトマスクセットを用いて、SoCそのもの

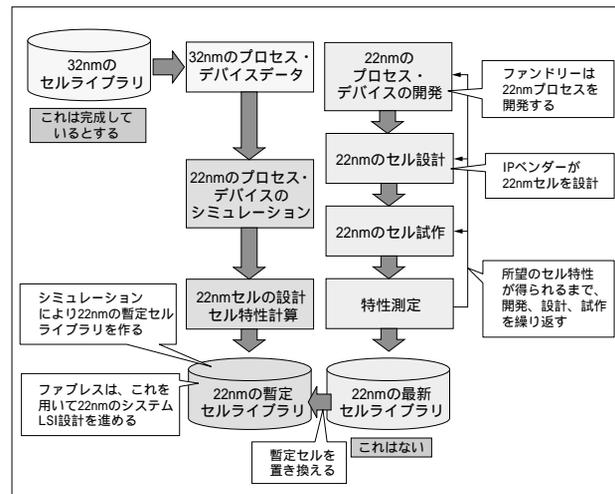


図2 22nmのセルライブラリの構築

の製造のためのプロセス開発と試作が行われる。試作チップができたならトランジスタや配線が予定通りできているか、回路が正常に動作するかなどが測定される。完全動作品が得られるようになったら、歩留りを向上させるためにプロセスを最適化し、量産に移行する。

### 最先端SoCを設計し製造するには

2010年現在、最先端の半導体メーカーでは、45~32nmのLSIが量産されており、ITRSロードマップに従って、22nmのLSIが開発段階にある。そこで、32nmのセルライブラリが完成していると仮定した上で、ファブレスが22nmの最先端SoCを設計し、これをファンドリーが量産する場合を想定してみよう。

### セルライブラリの更新

トランジスタや配線を微細化するなどプロセス条件が変更になると、デバイスの大きさや集積度は当然のこと、動作速度、消費電力などが変化する。それに伴って、トランジスタや配線の集合体であるゲートレベルの機能ブロックやIPなどの回路ブロックは、上記トランジスタや配線などのデバイスデータを基に、再設計し、再試作しなくてはならない。つまり、セルライブラリは、微細化などのプロセス条件の変化のたびに、全て更新されなくてはならない。

### セルライブラリ更新の波及効果

セルライブラリの更新が必要な場合、SoCの設計はどうなるのか? SoCの設計過程の中には、セルライブラリのデバイスデータおよびIPデータがないと正しい設計ができない工程が多数存在する。まず、システム設計の中のアーキテクチャ設計では、IPデータがなければ、どこまでハードウェアで実現可能か判断できない。また、フロントエン

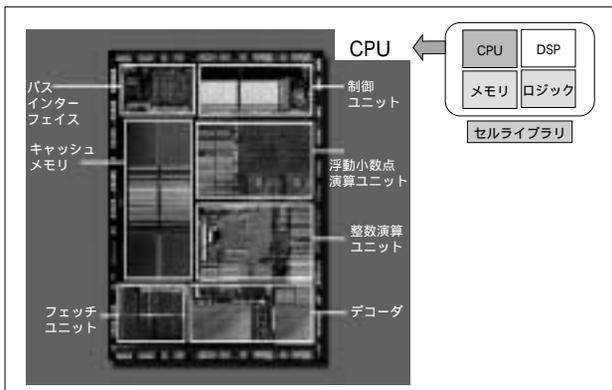


図3 大規模IPはSoCそのもの

ド設計の中の論理合成をする際には、ゲートの演算時間、遅延時間、占有面積、消費電力などの特性がわからなければ論理式への変換ができない。さらに、具体的なIPの回路データがなければ、最適なテクノロジーマッピングもできない。その結果、レイアウト設計もできない。

#### 暫定セルライブラリの構築

そこで、22nmのセルライブラリが存在しない場合、ファブレスは、32nmのセルライブラリのプロセス・デバイスデータを基に、シミュレーションにより22nmのプロセス・デバイスデータを作る(図2)。次に、それを基に、やはりシミュレーションにより、CPUやメモリなどのIPデータを構築する。そして、これらを22nmの“暫定”セルライブラリとして、SoC設計を進めるのである。

#### 最新セルライブラリの構築

それと並行して、ファンドリーは、まず22nmのトランジスタおよび配線の開発を行う。この開発には、通常、3年程度の期間が必要となる。この最新のデバイスデータを、逐次、ファブレスに伝えて設計に反映させる。

#### 大規模IPはSoCそのもの

セルライブラリ中の22nmのCPUやメモリなどのIPは、IPベンダーやIP専門のファブレスが設計を行う。この設計および製造フローは、基本的にSoC(図1)と同じ過程を必要とする。つまり、図3に示すように、CPUなどの大規模セルは、SoCそのものなのである。

ファンドリーが開発した22nmのトランジスタや配線などのデバイスデータと、それらを使って形成した基本動作(NAND、NOR、ORなど)回路を基に、IPベンダーなどが、CPUなどのIPのシステム設計、フロントエンド設計、バックエンド設計を行う。この結果、製造されたフォトマスクセットを用いて、ファンドリーが個々のIPに対応するプロセス開発と試作を行う。

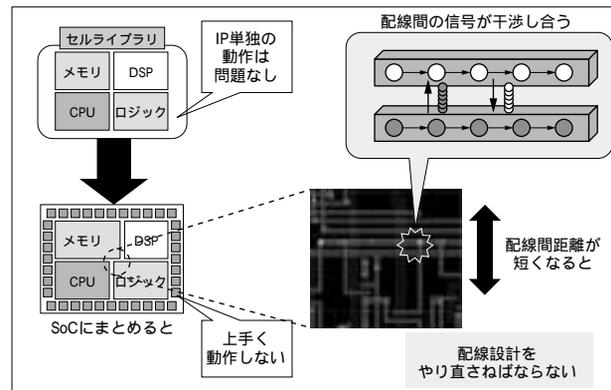


図4 SoCが動かない例：クロストーク

#### IP構築における設計とプロセスの擦り合わせ

しかし、最初のフォトマスクセットによる試作でIPが動作するとは限らない(動作しないことの方が多い)。また、動作しても要求性能を満たしていない場合が多い。そこで、IPベンダーは、試作の結果を基に設計をやり直す。まず、レイアウトを修正してファンドリーが再試作を行う。その試作結果が好ましくない場合は、論理合成、動作合成、アーキテクチャ設計と高次の設計修正を行う場合もある。最悪の場合は、トランジスタなどのデバイス修正が行われることもある。

このように、IPベンダーの設計とファンドリーの試作を何度も繰り返して、要求仕様に近い動作が可能なIPを構築する。以上のような擦り合わせ過程を経て、IPベンダーとファンドリーは、CPUやメモリなどのIPを1つずつ構築していく。

#### 暫定セルライブラリの置き換え

最新のIPが構築されたら、順次、暫定のセルライブラリを更新していく。そして、その情報を、SoCを設計するファブレスに伝えて、最新の設計にフィードバックする。

このファンドリーからファブレスへのデバイスデータおよびIPデータの効率的な伝達のために、ファブレスの技術者がファンドリーに駐在したり、ファンドリーの技術者がファブレスに駐在したりする。または、IPベンダーがその情報伝達を行うこともある。

ここ最近、多くのファブレスがプロセス技術者を雇用し、またファンドリーのTaiwan Semiconductor Manufacturing (TSMC) が大量の設計技術者を中途採用している。この技術者の転職状況は、ファブレスとファンドリーの双方向の情報伝達が極めて重要になってきていることを示唆している。

#### セルライブラリを更新するケース

セルライブラリの一更新を要求する事情は、以上に示したようにITRSロードマップに従って微細化

が一挙に進む場合だけではない。システムを高速化するために少しだけ微細化したり、デバイスの材料、構造、プロセスが少しでも変化すると、トランジスタや配線の特性が変化し、その結果、セルライブラリの全てのIPデータが変わってしまう。

つまり、材料や構造の変化を含めて新しいプロセスが開発されれば、その都度、セルライブラリを全て構築し直す必要がある。それに応じて、SoCの設計もやり直さなくてはならない。デジタル家電品が非常に短いサイクルで新製品を発表しているが、それに使われているSoCにおいては、このように、プロセスと設計の密接な調整関係がほとんど恒常的に存在している。

#### プロセス開発と試作

これまでに示した過程を経て最新のセルライブラリが完成したとしよう。そして、あるSoCのマスクセットが製造されたとする。そのマスクセットを用いて、ファブリーがプロセス開発および試作を行う。しかし、このプロセス開発と試作により、すぐにSoCが量産できるわけではない。それは何故か？

図4に示したように、CPU、メモリなどのIP単体の動作は問題がなくとも、SoCにした途端に正常に動作しない場合がある。

例えば、パターンが密集するために、露光時に光の近接効果により正常なレジストパターンが形成できない場合がある。また、エッチング面積が変わるため、微細な孔が開けられない場合もある。さらに、パターンが形成できても、隣同士の配線が電氣的に影響し合うクロストークが起きたり、あるトランジスタや配線に過度な電流が流れて破壊される場合もある。さらに深刻な問題として、IP間の配線の仕方によっては、信号遅延が起きてタイミングが合わず、SoCとして機能しない場合もある。

このような不良が起きた場合、ファブリー側でのプロセスの最適化で対応できる場合もあるが、多くの場合は、ファブレスにおける設計に戻ってマスクを作り直すことになる。露光やエッチングなどの微細加工の問題は、レイアウト設計の修正で済む場合が多い。しかし、クロストーク、電氣的破壊、タイミングのずれなどの問題については、さらに上流の設計に遡って修正しなければならないケースも多い。場合によっては、トランジスタの修正が必要となり、結果的にセルライブラリの更新が必要になる場合さえあるという。

何故、TSMCは日本IDMを凌駕したのか？

微細化が進むほど、このような問題が多発しており、ファブリーが製造できるようにファブレ

スが設計すること(Design for Manufacturing : DFM)が必要不可欠になってきている。従って、SoCにおいては、設計とプロセスを完全分離するという理想的な形態は、まったく不可能となっている。

これまで見てきたように、SoCにおいて、各設計間、および、設計とプロセス間には、密接な擦り合わせが必要である。従って「設計とプロセスが完全分離したことにより、垂直統合型(IDM)モデルが衰退し、ファブレス・ファブリーモデルが成長した」という説は、SoCの実態とは大きくかけ離れた誤認識であると言える。

では、何故、擦り合わせが必要なSoCにおいて、日本IDMが凋落し、ファブレス・ファブリーモデルが成長しているのか？

設計ツールベンダー米Cadence Design Systemsの元副社長から、「日本IDMの擦り合わせ能力が低いのではないか？」という意見を伺った。研究所、開発センター、事業部を転々とした筆者も同感である。日本IDMの組織は縦割り式であり、各組織が全体最適を考えて行動できていない。要するに、擦り合わせに最適な組織体となっていないのである。

一方、TSMCは、ファブレス、IPベンダー、設計ツールベンダーとともに、IPの構築およびSoCの各設計と製造を、極めて巧みに擦り合わせることでできる仕組みを構築している。

“日本人は擦り合わせが得意”というのは単なる妄想である。そうではなくて、日本IDMは、“擦り合わせが上手くいくような組織体”を、初心に戻って構築するべきであると考えられる。

#### 参考文献

- 1) 湯之上隆、鈴木良始：システムLSIにおける設計とプロセスとの擦り合わせ、工業経営研究会第25回大会(2010.8)(本稿は本学会発表を基に執筆)
- 2) 藤本隆宏：日本もの造り哲学、日本経済新聞出版社(2004.6) p.146
- 3) 鈴木良始、湯之上隆：半導体製造プロセス開発と工程アーキテクチャ論 - 装置を購入すれば半導体は製造できるか -、同志社商学、第60巻3/4号(2008) pp.54-154
- 4) 湯之上隆：Electronic Journal(2009.1) pp.66-68
- 5) 湯之上隆：日本「半導体」敗戦、光文社、第3章
- 6) 湯之上隆：Electronic Journal(2010.4) pp.42-43
- 7) 立本博文、藤本隆宏、富田純一：第6章 プロセス産業としての半導体前工程 - アーキテクチャ変動のダイナミクス、藤本隆宏・桑嶋健一編『日本型プロセス産業 - ものづくり経営学による競争分析』有斐閣(2009)
- 8) 小谷教彦、西村正：LSI工学、森北出版(2005)