

# セルベースICの活用例

## Examples of Application of ASICs

論理、レイアウトおよびテストデータが開発済みの既存モジュールとユーザーが設計するカスタムモジュールとを組み合わせ、カスタムLSIを開発するSBP (Silicon Back Plane)方式ASIC(Application Specific IC)の運用を開始した。本方法によれば、フルカスタムLSIと同等の高機能、高性能な素子をスタンダードセル並みの短時間で実現することができる。ASSP(Application Specific Standard Product)であるNPU(Network Processing Unit)のモジュールを用い、さらにユーザー固有のメモリ制御回路を付加してカスタムLSIを構成する応用例を示した。

堀田厚生\* Atsuo Hotta

脇本治己\*\* Harumi Wakimoto

### 1 緒言

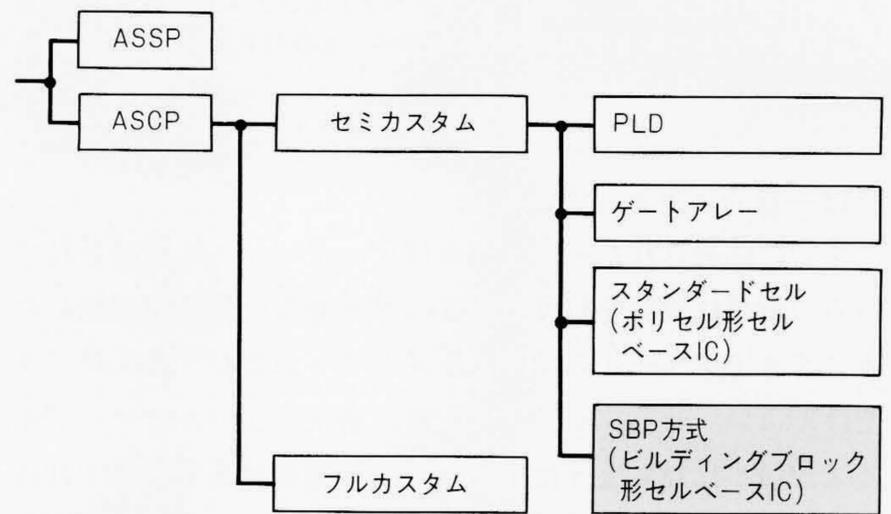
ワークステーションおよびパーソナルコンピュータなどの情報機器の需要が急速に拡大するに伴い、高性能かつ高機能なロジックLSIを短期間に開発することの重要性が増している。現在広く使われているASICは、ゲートアレーとスタンダードセル(ポリセル形セルベースIC)であるが、本論文では、フルカスタムLSIと同等の性能および機能を、スタンダードセル並みの開発期間で実現する新しいLSI設計方式であるSBP(Silicon Back Plane)方式ASIC(ビルディングブロック形セルベースIC)を紹介する。

### 2 ASICの分類

ASICの分類を図1に示す。

ASICではあるが、通信、ファイル、画像などの専門分野では、ほぼ標準ICとして使われるASSP(Application Specific Standard Product)と、ユーザーに特有の機能を盛り込んだASCP(Application Specific Custom Product)に分けられる。ASSPの例として、NPU(Network Processing Unit)およびMODEM(Modulator Demodulator)を挙げることができる。

ASCPは、セミカスタムとフルカスタムに大別できる。セミカスタムはさらにPLD(Programmable Logic Device)、ゲートアレー、スタンダードセル、ビルディングブロック形セルベースICに分けられる。ビルディングブロック形セルベースICは比較的新しい概念であり、スタンダードセルに似ている。日立製作所では、ビルディングブロック形セルベースICにSBP方式と称する設計手法を採用している<sup>1)</sup>。以下、これについて説明する。なお、日立製作所ではスタンダードセル方式とSBP



注：略語説明

ASSP (Application Specific Standard Product)

ASCP (Application Specific Custom Product)

PLD (Programmable Logic Device)

図1 ASICの分類 SBP(Silicon Back Plane)方式はスタンダードセル方式ASIC(Application Specific IC)の発展形と考えられる。SBP方式とスタンダードセル方式を合わせてセルベースICと呼ぶこともある。

方式を合わせて「セルベースIC」として、HG52シリーズを商品化している<sup>1)</sup>。

### 3 SBP方式ASIC

本方式はフルカスタム並みの機能および性能をスタンダードセル並みの開発期間で提供することを目的としている。

SBP方式ASICの概念図を図2に示す。チップ全体は数個のモジュールで構成しており、それらは標準バスによって結合している。各モジュールはそれぞれ一定の大きさの長方形のシリコン領域を占め、その論理、レイアウトおよびテストデー

\* 日立製作所 デバイス開発センタ 工学博士 \*\* 日立製作所 半導体設計開発センタ

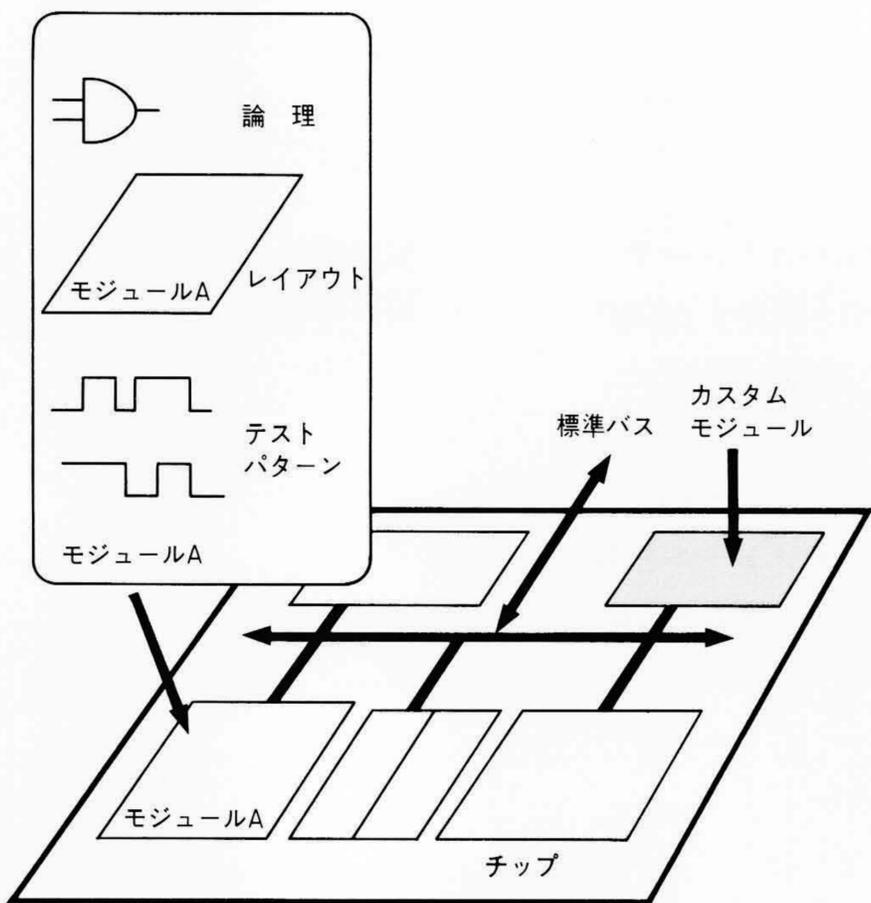
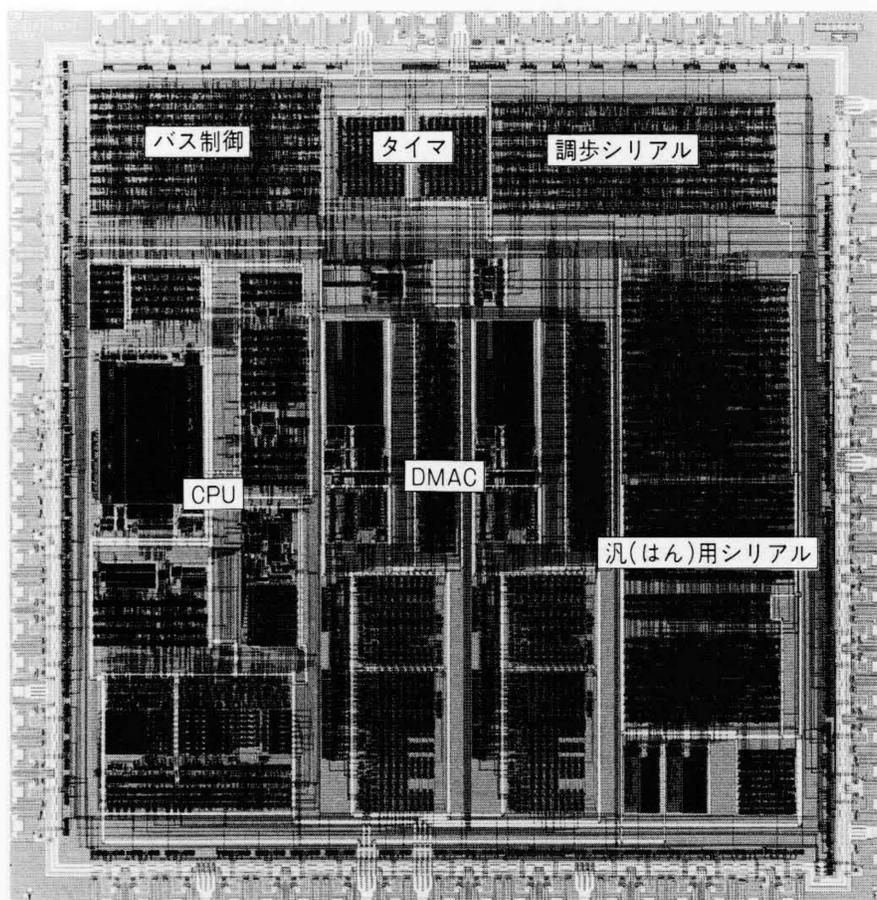


図2 SBP方式ASIC 論理、レイアウトおよびテストパターンの固定した既存モジュールとカスタムモジュールとを合わせてLSIを開発する。



注：略語説明 DMAC (Direct Memory Access Control)

図3 NPUのチップ写真 NPU(Network Processing Unit)は数個のモジュールで構成している。これらのモジュールは再利用できる。

タは一意的に決まっている。

すでに開発されたモジュールはデータベースに登録されている。新規に設計を行うのはユーザー固有の機能を集積したカスタムモジュールの論理、レイアウトおよびテストデータだけでよい。チップレイアウトは、各モジュールをチップサイズが最小になるように配置した後、標準バスで結線すればよい。

本方式はASSPの開発にも使える。その一例として、NPUの例を示す。

NPUのチップ写真を図3に、NPUチップのモジュール配置を図4にそれぞれ示す。NPUはCPU、DMAC(Direct Memory Access Control)(2個)、汎(はん)用シリアル、調歩シリアル、タイマ(2個)、バス制御モジュールなどで構成している。これらのモジュールは、Si-バスと呼ばれる標準バスによって結合されている。

このように、ASSPを設計するときにはSBP方式で設計しておけば、そのときに新規開発したモジュールを後でASCPに流用することができる。

NPUを用いた通信アダプタボードのブロック図の一例を図5に示す。通信回線を介して受信されたデータは、NPUでデータリンクレベルのプロトコル処理を受けた後、共有メモリを介してシステムバス上に送り出される。その後システムバスに接続されたCPUが、そのデータに対し上位のプロトコル処理を行う。データを送信する場合は、この逆のルートを通る。すなわち、CPUが送信データを用意し、共有メモリに書き込む。そのデータをNPUが取り出し、必要なプロトコル処理を

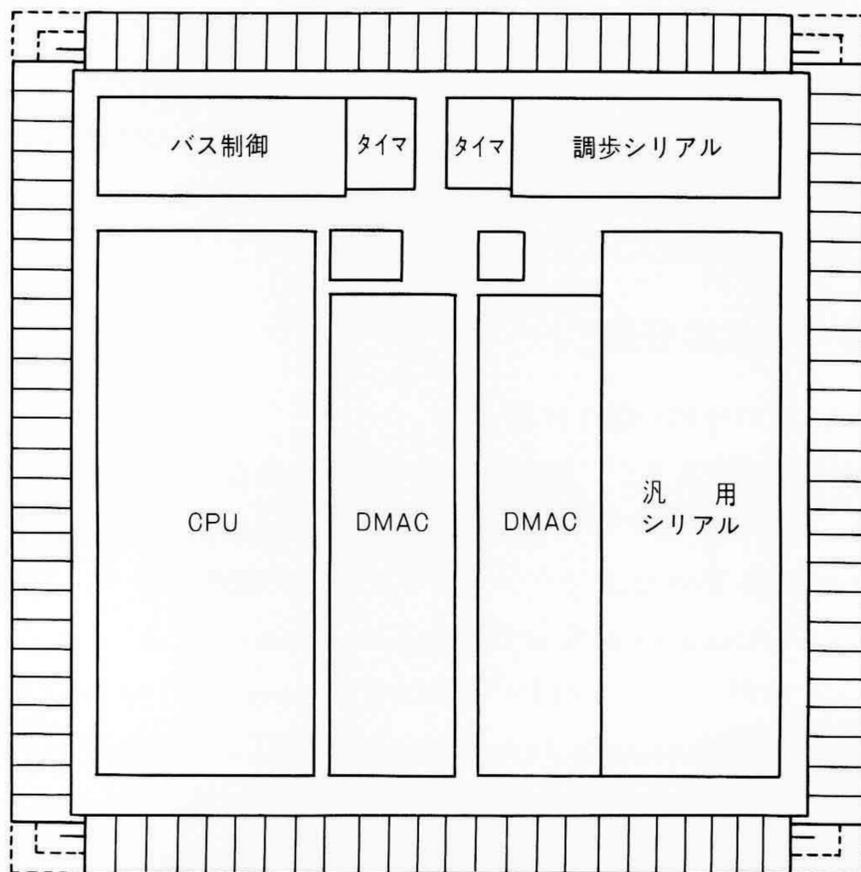
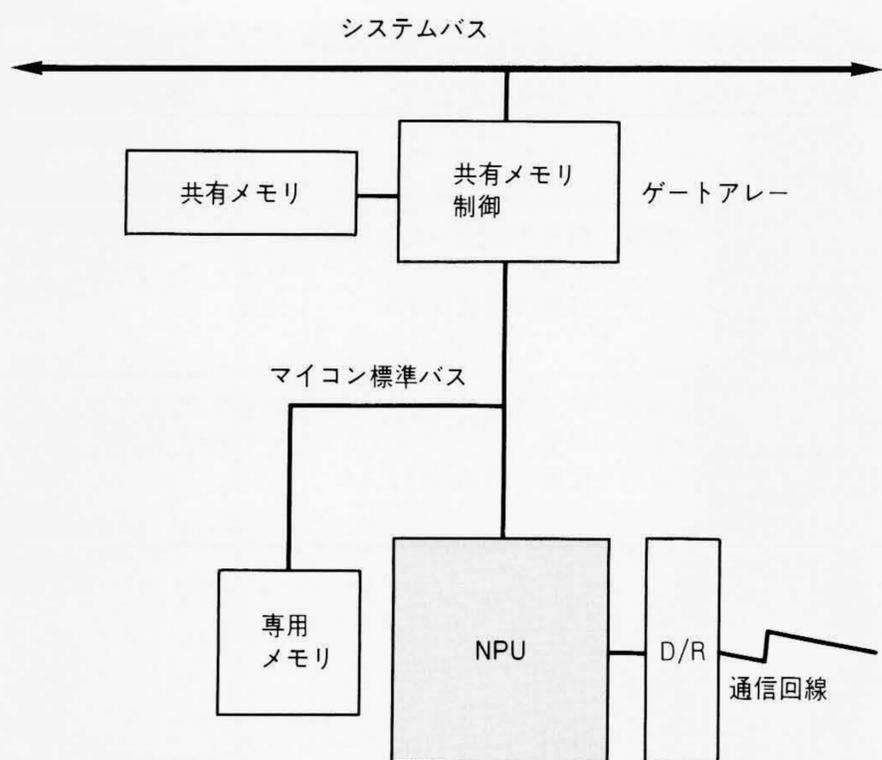


図4 NPUチップのモジュール配置 モジュールは長方形をしている。

行った後、通信回線に送り出す。

図5は、多くの通信ボードに当てはまる。同図でシステムバスは、ユーザー固有のものが使われる場合が大多数である。したがって、共有メモリ制御のブロックは、ユーザー固有の論理を含むためゲートアレーで実現されるのが普通である。このカスタムブロックをNPUとともに集積したカスタムLSI



注：略語説明 NPU (Network Processing Unit)  
D/R (Drivers/Receivers)  
マイコン (マイクロコンピュータ)

図5 NPUを用いた通信アダプタ NPUは共有メモリ制御回路を介して、システムバスに接続される。

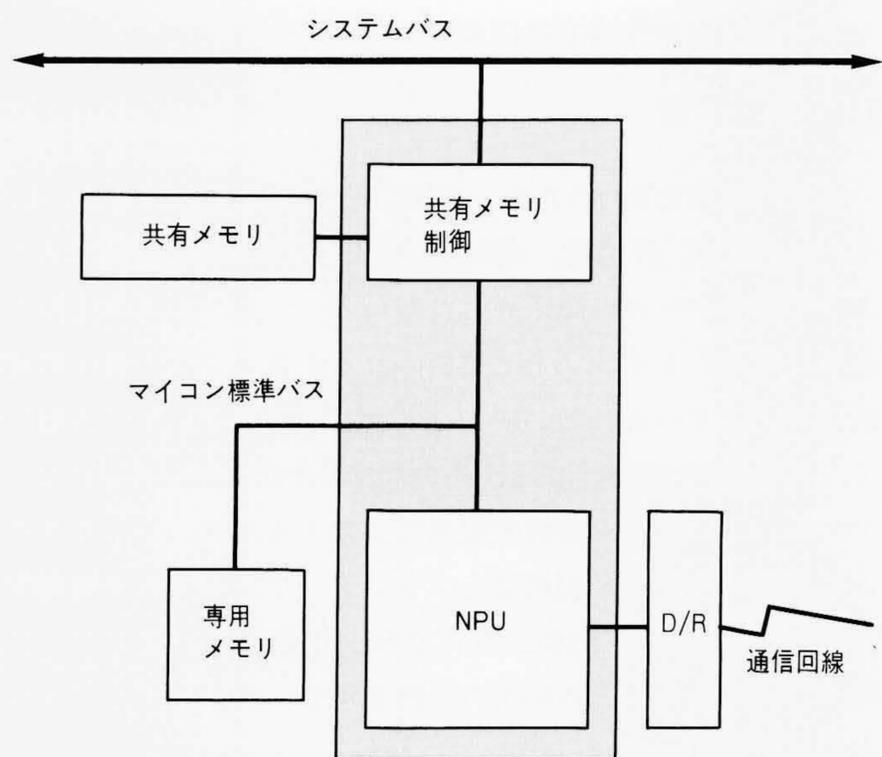


図6 SBP方式ASICを用いた通信アダプタ NPUと共有メモリ制御回路を同一チップに集積することで、通信アダプタボードの小形化が可能となる。

を用いた通信アダプタを図6に示す。同図に示す専用メモリは、NPUのプログラムおよびデータを格納するものである。

この結果、ボード上の部品点数が減り、ボードの小形化が可能となる。

#### 4 設計方法

設計フローを図7に示す。最初に方式設計を行う。この段階では、既存モジュールのリストの中から必要なモジュール

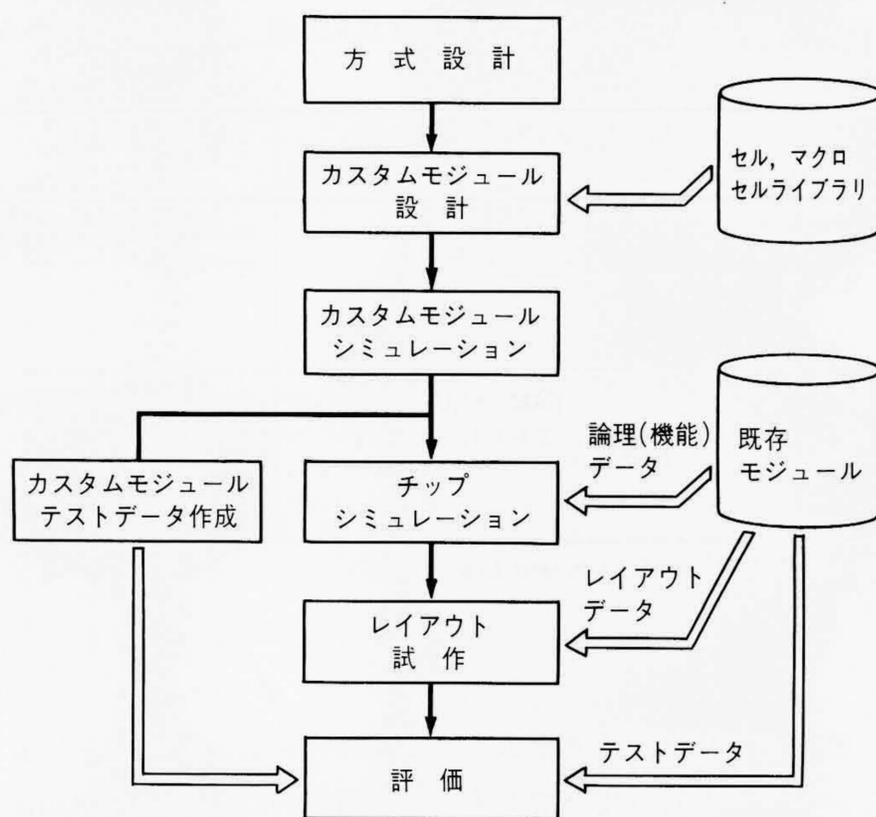


図7 設計方法 設計工数の大半はカスタムモジュールの設計に費やされるので、設計期間は大幅に短縮する。

を選び出し、チップのブロック設計を行う。

次にカスタムモジュールの設計を行う。本モジュールは、通常のスタンダードセル方式のLSIの設計と同じである。すなわち、セルライブラリにあるセルおよびマクロセルを用いて論理設計を行う。また、設計の正しさを確認するためにシミュレーションを行う。

その後、チップ全体のシミュレーションを実施する。このときに必要な既存モジュールの論理データはライブラリから持ってくる。次に、カスタムモジュールおよびチップのレイアウトを行い、マスクを製作し試作する。

最後に評価を行う。このとき、カスタムモジュールの部分のテストデータは新たに作成したものをを用いる。既存モジュールについては、すでに完成しているテストデータを用いる。

以上は設計のあらましであるが、設計工程をLSIユーザーとメーカーとでいかに分担するかについて、いくつかの方法がある。

第一の方法は、方式設計だけユーザーが行い、ユーザーが出した仕様書に従って、カスタムモジュールをメーカーが設計する場合である。本方法は、通常のフルカスタムLSIの開発方法と変わらず、メーカー側の工数低減は、既存モジュールの設計をしなくてよい点だけである。また、カスタム部分の設計は、ユーザーが実施したほうが効率的である場合が多いので、本インタフェースはSBP方式ASICのねらいを生かすものとは言いがたい。

第二の方法は、カスタムモジュールの設計までユーザーが行う場合である。この論理図インタフェースは、ゲートアレーおよびスタンダードセルLSIで広く行われている方法である。

表1 SBP方式とスタンダードセル方式の比較 SBP方式を用いることにより、フルカスタム並みの高機能、高性能をスタンダードセルと同等の期間で開発できる。

方式	項目	機能性能	論理設計	シミュレーション	レイアウト	テストデータ作成
SBP方式		フルカスタムLSIと同等	カスタム部分だけの設計で可	既存モジュールは機能記述で可 ハードウェアシミュレーション可	既存モジュールは配置するだけ。 任意の配置は不可	カスタムモジュールのテストデータだけ新規作成
スタンダードセル方式		RAM, ALUなどのマクロセル使用可	チップ全体の論理設計が必要	チップ全体をトランジスタゲートレベルでシミュレーションするのが原則	チップ全体レイアウト必要 配置の融通性は高い。	チップ全体のテストデータの作成が必要

注：略語説明 ALU(Arithmetic Logic Unit)

第三の方法は、カスタムモジュールのシミュレーションまでユーザーが行う場合である。論理設計のDA(Design Automation)環境を備えたユーザーの場合、自社のDA上で設計の検証を実行することを好むからである。

第四の方法は、チップシミュレーションまでユーザーが実行する場合である。カスタムモジュールのデバッグの品質を上げるために、チップ全体でシミュレーションを行う場合である。この場合、既存モジュールの論理ファイルか、あるいは機能記述をユーザーに提供しなければならない。またはメーカーが提供する既存モジュールのICを用いて、プリント板上にチップ全体の論理を実装し、デバッグを行う。このとき、カスタムモジュールのICをゲートアレーで製作する方法と、カスタムモジュールの論理は計算機上に記述し、計算機と既存モジュールのLSIとを連動させて、シミュレーションを実行する方法がある。

### 5 SBP方式とスタンダードセル方式の比較

SBP方式は新しいLSIの開発方法であり、スタンダードセル方式の発展形と考えられる。両者の得失を表1に示す。

機能および性能の点では、SBP方式のほうがCPUコアを始めとしフルカスタム技術で設計した各種の高機能かつ高性能なモジュールを利用できる点で優れている。

論理設計では、SBP方式がカスタムモジュールだけを新規設計すればよいのに対し、スタンダードセルではチップ全体を設計する必要がある。

カスタム部のシミュレーションでは両者の差はないが、チップシミュレーションではSBP方式の既存部分を機能記述で表現することにより、シミュレーション時間を少なくすることができる。また、既存モジュールのICを用いて、ハードウェアシミュレーションを実行することもできる。

レイアウトについては、SBP方式では、カスタム部を長方

形状にレイアウト設計した後、チップ上に全モジュールを配置し、それからモジュール間を結線する。したがって、チップのレイアウトは簡単になるが、既存モジュールの形状があらかじめ固定されているので、チップの形に制限が生じる。一方、スタンダードセルでは、チップの形状は任意にできるが、チップ全体のレイアウトを行わなければならない。

テストデータについては、SBP方式ではカスタム部のテストデータだけを作成すればよい。

### 6 結 言

装置の差別化を行うため、ASICを採用する例が増えている。そのためには、高機能、高性能のASICを短期間に開発することが半導体メーカーに求められている。一方、半導体技術の進歩に伴い、1チップ上に集積されるトランジスタ数は飛躍的に増大している。この膨大なトランジスタをいかに活用して、高性能かつ高機能なLSIを開発するかが重要な課題になっている。このような時代の要求にこたえるため、フルカスタムIC並みの高性能、高機能をスタンダードセル並みの短期間で開発する新しいLSI設計方式であるSBP方式ASICについて紹介した。また、その応用の一例を示し、スタンダードセルとの比較を行った。

本開発法の応用例を増やすとともに、改良を行っていくことが今後の課題である。

### 参考文献

- 1) 小林, 外:セルベースIC「HG52シリーズ」, 日立評論, 71, 12, 1223~1230(平1-12)
- 2) 堀田, 外:通信制御プロセッサ, 日立評論, 70, 12, 1275~1280(昭63-12)