

# システム・オン・チップ (SoC) デザインの 電力効率を高める全体的なアプローチ

米国シノプシス

システム・アーキテクト・グループ  
シニア・ディレクター

Piyush Sancheti

## はじめに

現在の社会活動は非常に多くのエネルギーによって支えられていますが、電力需要は日々増加しています。この傾向はエレクトロニクス分野では特に顕著で、自動化の進展と機器のインテリジェント化によって電力需要が絶え間なく増え続けています。半導体チップを使用する多くのアプリケーションが、消費電力の削減と電力効率の向上を強く求められています。これを受けて、半導体およびEDA業界はこうした要求に応えるための技術を幅広く開発してきました。本稿では、この問題の背景、および問題の解決に役立つテクノロジーをご紹介した後、アーキテクチャからサインオフまでシステム・オン・チップ (SoC) 設計フロー全体で電力効率を高める全体的なソリューションについてご説明します。

## エネルギー需要を押し上げる要因

単純に言って、電子機器の数が増えればそれだけ消費電力も増大します。ここ数十年で、日常生活のほとんどがデジタル・ガジェットに依存するようになっており、これら機器の動作は巨大なデータセンターによって支えられています。そうすると電力需要が増大するのは当然ですが、消費電力全体に占めるエレクトロニクス関連の消費電力の割合も大きくなっていきます。図1は、超小型のコンシューマ機器から大規模なコンピューティング・サーバーやネットワーク機器まで、ICT（情報通信技術）に関連する世界の電力需要の伸びを示したものです。ICTアプリケーションの消費電力だけでも2022年から2030年にかけてほぼ3倍に増え、2030年には全発電量の20%以上をICTアプリケーションが消費すると予想されています。

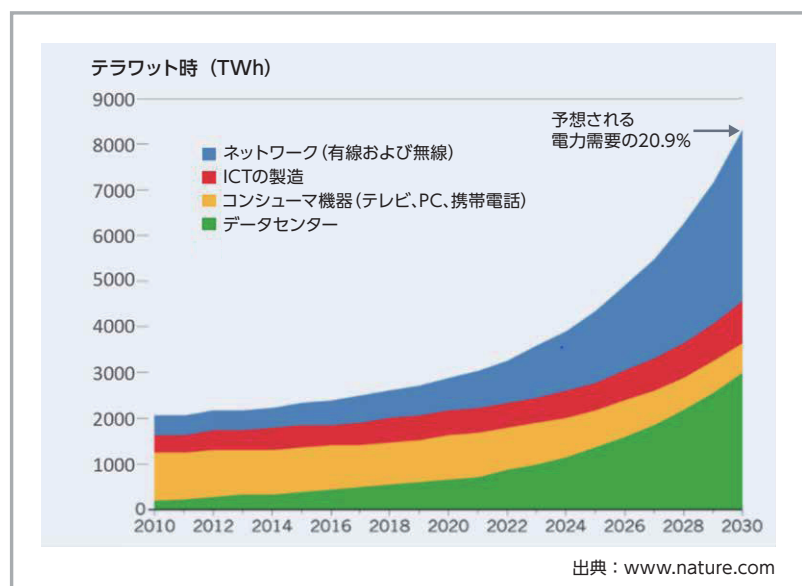


図1: ICTアプリケーションの電力需要の増大

増えた分の消費電力はどこからか供給する必要がありますが、その需要に応えるためのコストは金銭的な面でも社会的影響の面でも莫大なものになります。ただし、アプリケーションの主要な要件を満たしながら機器の電力効率を可能な限り高めることは、半導体メーカーとチップ設計者にとって明らかに利点があります。消費電力を抑えることは、特にバッテリーで動作する機器において製品の重要な差別化要因となるためです。歴史的に、電力効率を高めて消費電力を削減する取り組みのほとんどは、コンシューマ向け携帯機器を中心に進められてきました。ユーザーはバッテリーが長持ちする多機能な製品を求めているため、設計時と機器使用時の両方で性能と消費電力のトレードオフを行うことが非常に重要になってきます。

スマートフォン、タブレット、ノートPC、ウェアラブル機器などの小型デバイスでは、発熱も大きな課題となります。ユーザーがこれらの機器を手にとった時に不快にならないようにするには、熱を拡散してホットスポットを解消する必要がありますが、そうするとコストもかかり、製品の容積も大きくなります。これを防ぐには、開発の初期段階から消費電力を管理していく以外にありません。ここで注意しておきたいのは、小型機器だからと言って必ずしも小型のチップが搭載されているわけではないということです。IoTアプリケーションの多くは比較的単純ですが、スマートフォンには非常に大型で複雑なSoCが搭載されています。自動運転車は小型アプリケーションではありませんが、コンシューマ機器の1つであり、複雑なSoC、センサー、およびその他の電子部品を必要とします。

デスクトップPCやサーバーなど比較的大型のシステムでは、放熱と熱管理に関して比較的多くの選択肢があります。多くのデザインでは、先進のチップ・パッケージ、大型のヒートシンクやファン、水冷などを利用できますが、そうするとコストが大幅に増大します。消費電力を極限まで減らすことは求められませんが、電力管理は必要です。データセンター・サーバーの使用開始から廃棄までの間に消費される電力のコストは、サーバーのハードウェア初期費用を上回るという調査結果もあります。これらのシステムに使用されるSoCの電力効率を高めることは、金銭的にも大きな意味があります。購入者の側も、地球の気候変動に対する個人的な関心、あるいはデータセンターや各種機械の消費電力を制限する環境規制の遵守のために、なるべく二酸化炭素排出量を削減したいと考えています。特に人工知能（AI）や機械学習（ML）を使用した新しいアプリケーションでは、消費電力が増大の一途をたどっています。そこで、これまで以上に消費電力を意識した設計プロセスが必要とされています。

## エンド to エンドのローパワー設計

最適な結果を得るには、SoC設計のあらゆる段階で電力効率に対処する必要があります。消費電力の削減および管理については、長年にわたってさまざまな手法が開発されてきました。図2は、物理レベルから始まるこれら手法のうち、特に重要なものを示したものです。一般に、半導体の同義語として「シリコン」という用語がよく使われますが、それ以外にも半導体材料は存在しており、材料の選択に当たっては消費電力が考慮されることがあります。例えば、かつてガリウムヒ素（GaAs）は一部の高性能アプリケーションにおいてシリコンをしのぐ存在になると考えられていましたが、シリコンの方が熱伝導率が高く放熱性に優れるため、過熱を防ぎながらより大きな電力を扱うことができます。

設計段階	ストラテジ
ソフトウェア・レベル	電力を考慮したアプリ
OS/ファームウェア・レベル	電力を考慮したOS/ファームウェア
アーキテクチャ・レベル	電力管理、DVFS
	システム・コンポーネント
RTL(デザイン)レベル	マイクロ・アーキテクチャ
	電力効率に優れたRTL
インプリメンテーション・レベル	ロジック・レベル
	トランジスタ・レベル
	レイアウト・レベル
プロセス・テクノロジー・レベル	ライブラリ、マクロ
	デバイス/トランジスタ
	材料

図2: エンド to エンドのローパワー設計フロー

こうした素材だけではなく、トランジスタおよびその他デバイスの構造も電力効率に大きな影響を与えます。チップ開発の黎明期から、エンジニアはターゲット・デザインに最も適した特性を持つトランジスタを選択することによってPPA（消費電力/性能/面積）のトレードオフを行っていました。16nm以下のプロセス・ノードでFinFET（Fin Field-Effect Transistor）デバイスが主流になっているのは、その代表例です。ディープ・サブミクロン時代になると、それまであまり大きな問題ではなかったリーク電流が消費電力の大半を占めるようになりました。この問題を緩和するため、デバイスのチャネル制御性を高めたFinFETが導入されるようになりました。

SoC設計のほとんどはトランジスタ・レベルではなく、RTL（Register Transfer Level）で行われるか、またはそれより高いレベルで共通機能のセル・ライブラリを使用してコードを合成して行われます。数十年前にスタンダード・セルとゲート・アレイが導入されて以来、ライブラリには機能が同じでPPA特性のみが異なるセルが複数含まれるようになってきました。一般的なライブラリには、多くのセルについて少なくとも「低消費電力」バージョンが含まれており、クリティカルパスでない機能にはこれらのバージョンを選択できます。論理合成ツールを使用すれば、セルに対する何種類ものマッピングを短時間で試すことができるため、ライブラリに多くの選択肢があれば、設計者の手を煩わすことなくPPAの目標を容易に達成できます。先端ノードでは物理的影響が支配的になるため、あらゆる解析および最適化テクノロジーでこうした物理効果を考慮できる必要があります。高品質なライブラリとIPを使用する必要があります。消費電力はSoCのパワー・インテグリティと熱特性に影響を与えるため、フィジカル・インプリメンテーション（合成およびレイアウト）時に対処し、サインオフ時に確認する必要があります。

電力効率は、RTLデザインおよびそこから論理合成によって生成されるゲート・レベルのネットリストでも対処する必要があります。ここでは、設計者が消費電力の目標を考慮に入れながらマイクロ・アーキテクチャを定義することが必要です。一般的な例として、SoCの非アクティブな部分をシャットダウンしてスタンバイ・モードに移行させたり、動的電圧周波数スケーリング（DVFS）によって動作を臨機応変に制御したりする機能があります。インプリメンテーションの終盤には自由度が低くなるため、そこでボトルネックとならないように、この段階で最適なマイクロ・アーキテクチャを見つけることが重要です。DVFSなどの高度な手法はマクロ・アーキテクチャ・レベルでのサポートが必要なため、全体的なシステム電力管理を高レベルのチップ・モデリングの一部に含める必要があります。SoCアーキテクトは電力制御構造を定義し、最終的にエンド・システム上で動作するソフトウェアによって操作できるようにフックを用意しておく必要があります。

ソリューションの最後のピースとなるのがソフトウェアです。現在のほとんどのSoCでは、純粋にハードウェア・レベルで決定できる電力管理も一部存在しますが、制御の大半は消費電力を考慮したファームウェア、オペレーティング・システム、およびアプリケーション（アプリ）が担っています。例えば、オペレーティング・システムは実行中または実行予定のアプリケーションやタスクをすべて把握しているため、最大限の性能が必要でない場合はチップの一部の電源をオフにしたり実行速度を落としたりするなど、インテリジェントな判断が可能です。SoCによっては、あまりにも多機能化が進み、チップ全体を同時に最高速度で動作させると、高温による損傷を招くことがあります。オペレーティング・システムは、このことを考慮して電力管理の方法を選択する必要があります。製造フロアでダイやチップのテストに使用するプログラムも、電力を考慮して過熱を防ぐようにする必要があります。

図3は、図2に示した手法がSoCプロジェクト全体でどのように適用されるかを時系列で示したものです。ローパワー設計フローでは、実際のソフトウェア・ワークロードを使用して、高速プロファイリングによってアクティビティ（ベクター）を駆動することにより、消費電力の検討/分析/最適化を行う必要があります。しかし一般的なソフトウェア・ワークロードは十億（B）単位のサイクルで数テラバイトのデータが生成されるため、プロセスの段階ごとに関心のある主要なウィンドウを絞り込み、データ・サイズを扱いやすいものにするアプローチをとります。この結果、アーキテクチャからRTL、インプリメンテーション、サインオフまでのフローの各段階で、実際のソフトウェア・ワークロードに基づいて、消費電力の検討/解析/最適化を実施できるようになります。

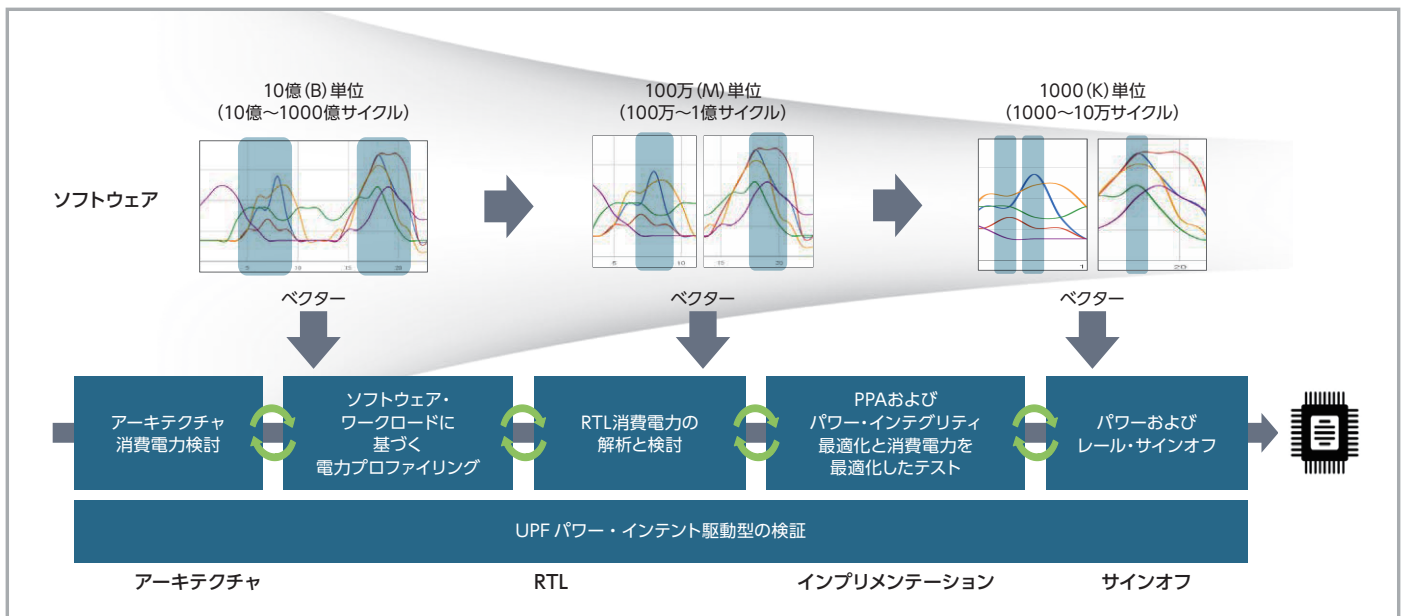


図3: ローパワー設計フロー

ローパワー設計フロー全体で電力効率を管理することは、UPF (Unified Power Format) 規格 (IEEE 1801-2018) の登場によって容易になりました。UPFは、SoCの電源制御ネットワークについて以下のように多くの側面を規定します。

- 電源ネットおよびパワー・スイッチ
- 電源/電圧ドメイン
- これらドメイン間のレベル・シフトとアイソレーション・セル
- 電源ステートとこれらステート間の遷移
- チップの一部の電源をオフにした場合のメモリー・リテンション

UPFファイルは、パワー・インテントの仕様を記述したものです。設計ツールはこのファイルを読み出し、その内容に基づいてロジック合成から配置配線までのインプリメンテーションをガイドします。アーキテクチャ・ツールはUPFを使用して仮想モデルに電力管理を反映させ、マクロ・アーキテクチャのトレードオフを支援します。多くの検証ツール (シミュレーション、エミュレーション、フォーマル解析、サインオフ・チェック) でも電力構造が考慮されます。

## シノプシスが提供するエンド to エンドのローパワー・ソリューション

エンド to エンドのローパワーSoC開発フローを必要としているアーキテクト、設計者、検証エンジニアにとっては現在、多くのチップ・プロジェクトで実証済みのソリューションを利用できる環境が整っています。シノプシスのローパワー・ソリューションは、アーキテクチャからサインオフまでソフトウェア・ワークロードを考慮した電力の検証/検討/解析および最適化をサポートします。図4は、図3に示したフローにシノプシス製品を重ね合わせたもので、これにより業界で最も包括的かつ効果的なソリューションが実現します。これには、以下の製品が含まれます。

- Platform Architect™: RTL前のアーキテクチャ・モデルとソフトウェア・ワークロードを使用したアーキテクチャ検討および早期段階での性能/電力トレードオフが可能です。
- ZeBu® Empower: ソフトウェア・ワークロードのプロファイリングに必要な容量と性能を備えた電力エミュレーション・システムで、関心の高いウィンドウを特定することによってさらなる解析と検討を可能にします。
- SpyGlass® Power: RTL開発の早期段階におけるRTL電力検討を短TATで実行できます。
- PrimePower RTLとRTL Architect: RTLが成熟した段階で、高精度のRTL電力解析が行えます。
- Fusion Compiler™: RTLからGDSIIインプリメンテーションまでをサポートし、最高のPPA結果が得られます。
- PrimePowerとAnsys® RedHawk™のサインオフ・エンジンのフュージョン: 収束までの時間を最短化し、最高の結果品質 (QoR) を達成します。
- PrimePowerおよびAnsys RedHawk: ゴールデン・サインオフ
- TestMAX™: 消費電力を最適化した自動テストパターン生成 (ATPG) により、チップのテスト時にも消費電力を適切に管理できます。

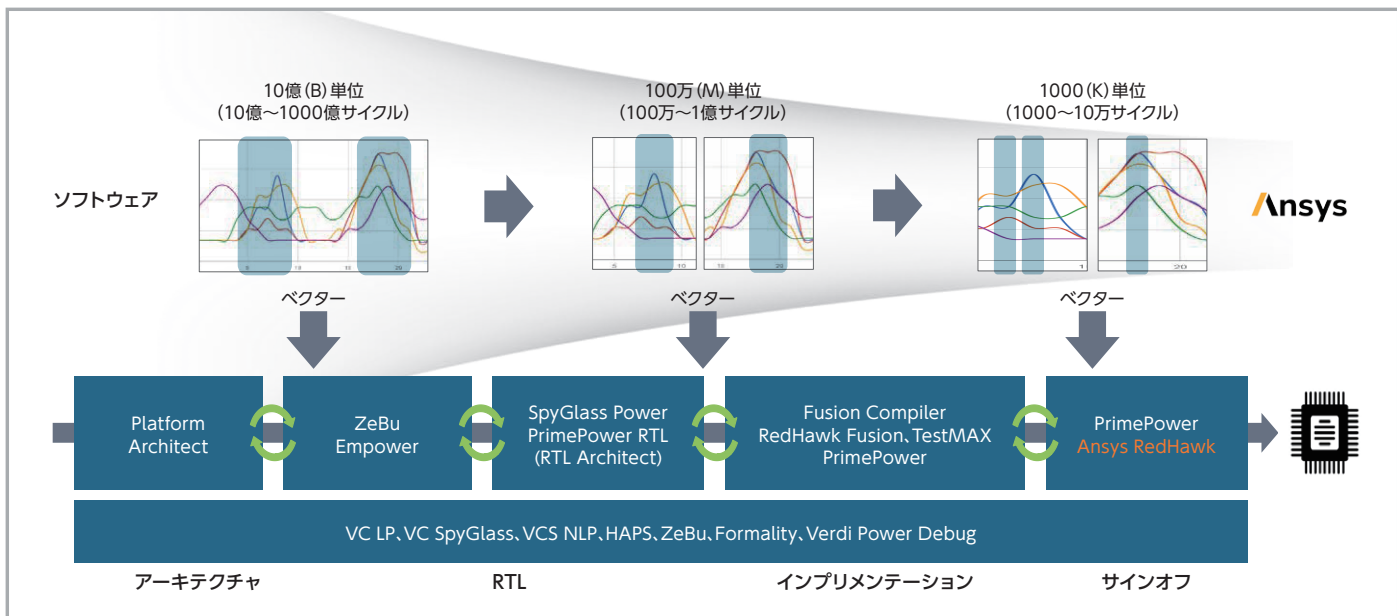


図4: シノプシスのローパワー・ソリューション

シノプシスの開発フローでは、設計の各段階で消費電力削減の機会が最大限に提供されるため、電力効率に優れた最適な結果を得ることができます。実際のソフトウェア・ワークロードに基づいて電力と性能のトレードオフが行えるため、SoCの立ち上げ時に予想とかけ離れた結果となることがありません。また、早期段階で正確な電力解析が行えるため、PPAの目標を予測可能な形で短時間に達成できます。図5はシノプシスのフローの詳細を表したもので、ソリューションの各コンポーネントがデザインの異なるレベルでどのように適用されるかを示しています。アーキテクチャ段階では、Platform Architectで抽象モデルを使用してマクロ・アーキテクチャのオプション検討および電力と性能のトレードオフを行います。一部のIP開発は並行して進めることができますが、RTL設計のほとんどはマクロ・アーキテクチャが確定した後に行われます。

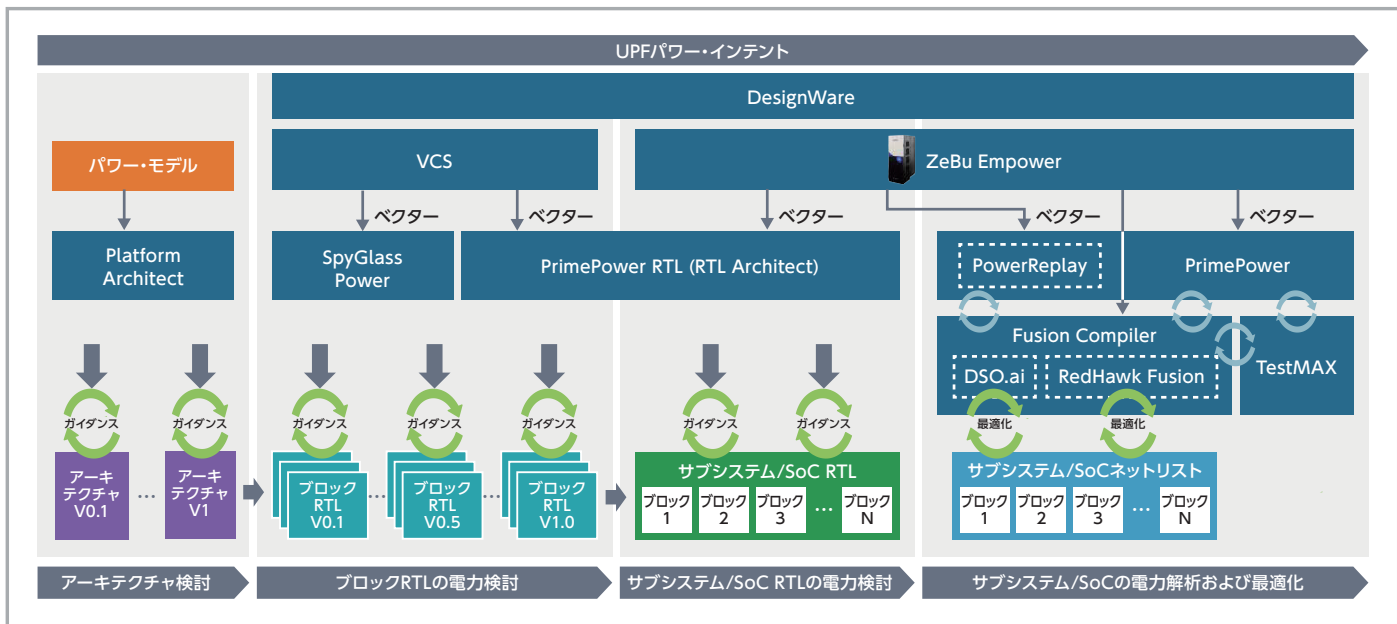


図5: ソフトウェア・ワークロードを考慮した電力の検討/解析/最適化

RTLブロック開発の早期段階でマイクロ・アーキテクチャを定義する際に、シミュレータVCS®で生成したベクターとSpyGlass Powerを使用して早期電力検討を実施します。RTLブロックが成熟してインプリメンテーションに近付いてくると、フィジカルとタイミングを考慮したRTL Architectの予測エンジンおよびPrimePowerのサインオフ・エンジンを内蔵したPrimePower RTLにより、さらに正確な解析を行います。SoCまたはサブシステムがエミュレーション可能な状態になると、ZeBu Empowerを使用してソフトウェア・ワークロードをプロファイリングし、特に関心の高いウィンドウ（ピーク電力、平均電力の高い領域など）

を特定します。このウィンドウをPrimePower RTLで使用することにより、さらに詳細な解析が可能となります。ZeBu Empowerは、数十億サイクルのSoCソフトウェア・ワークロードを完全に処理できるだけの容量と性能を備えた業界唯一の電力エミュレーション・システムです。

デザインがインプリメンテーションに進むと、アクティビティ・ウィンドウをさらに絞り込んでFusion Compilerでのインプリメンテーションを実施します。Fusion CompilerのパワードリブンRTL-to-GDSIIフローでは、人工知能エンジンDSO.ai™ (Design Space Optimization AI) を活用することで、PPAの最適な結果を短時間で得ることができます。業界初のチップ設計向け自律型アプリケーションであるDSO.aiは、SoCデザインの膨大な解空間を探索し、より優れた最適化ポイントを見つけます。さらに、ZeBu Empowerから得られたアクティビティ・ウィンドウによってPrimePowerでのサインオフを駆動します。PrimePowerのPowerReplay機能は、VCSによるRTLシミュレーションからのベクターをインプリメンテーション後のゲート・レベル・ネットリストに対して再利用します。PrimePowerによるゴールデン・パワー・サインオフには、グリッチ解析およびデバッグ、タイミング精度を高める遅延シフト、先端プロセス・ノードに対応したモデリングなどの重要なテクノロジーが含まれます。最後に、TestMAXが電力を考慮に入れて製造テストを生成します。

SoCには自社開発のRTLだけでなく、商用IPも使用されています。シノプシスは、プロセッサ、インターフェイス、センサー、アナログ/ミックスドシグナル (AMS)、メモリーおよびロジック・ライブラリに関して、幅広いDesignWare® ローパワーIPポートフォリオを提供しています。これらのIPには、設計者から提供されるファイルを補完するものとして、事前に定義済みのUPF記述が付属します。すべてのライブラリとIP、および図4に示したすべてのツールは、一貫したUPFサポートによって相互にリンクされます。図6に、この統一されたアプローチでローパワー検証に使用されるシノプシスのツールの詳細を示します。この組み合わせにより、ブロック・レベル、サブシステム・レベル、そして完全なSoCレベルでUPFパワー・インテントに基づいた包括的なローパワー検証が可能になります。

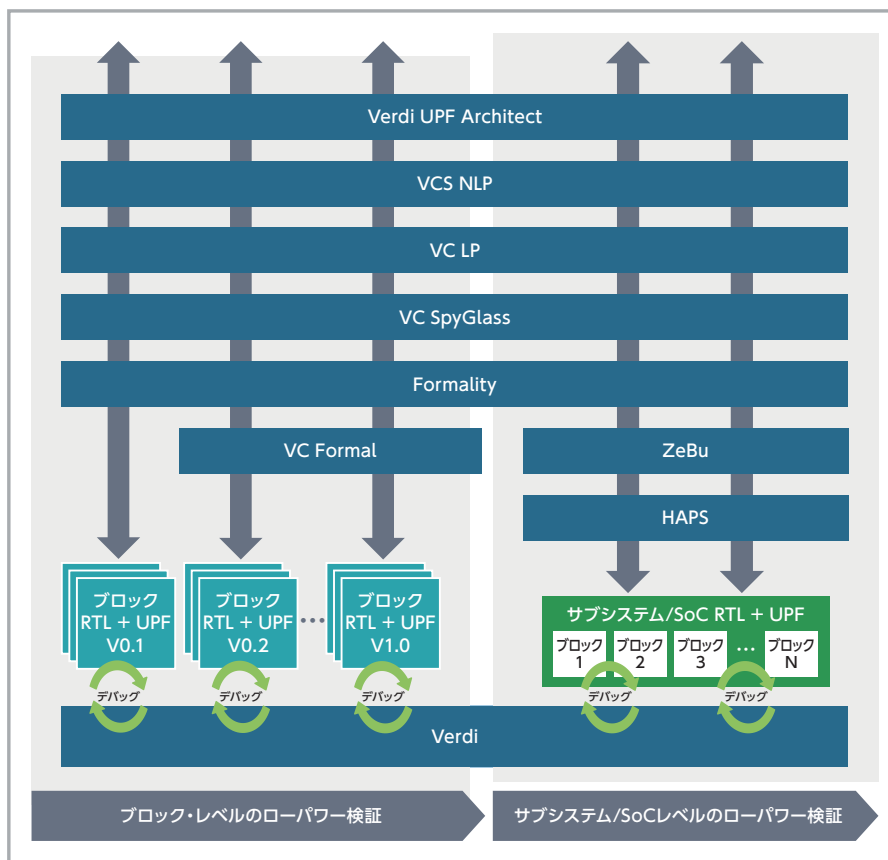


図6: シノプシスのローパワー検証

このプロセスは、まず業界をリードするVerdiデバッグ・ソリューションの1つであるVerdi® UPF ArchitectによってCorrect-by-ConstructionなUPFを生成することから始まります。Verdi UPF Architectでは、デザインのパワー・インテントを高い抽象度で記述し、これに基づいてUPFファイルを生成できます。このUPFを、デザインおよびテストベンチと一緒に

VCS® Native Low Power (NLP) に入力します。VCS NLPはUPFで記述されたパワー・ネットワーク全体をモデル化し、パワー・イベントを考慮して動的シミュレーションを実行します。例えば、ある電源ドメインが制御信号によってオフに切り替わった場合、VCS LPはそのドメインのすべての信号を不明 (Unknown) に設定します。スタティック・ローパワー検証ソリューションVC LP™には650種類以上のチェック機能があり、従来の手法よりも早い段階でローパワー・バグを迅速に検出できます。これらのチェック機能には、以下のようなものがあります。

- パワー・インテント整合性チェック：UPFの構文とセマンティクスをチェックし、インプリメンテーション前にUPFの整合性を確認
- アーキテクチャ・チェック：パワー・アーキテクチャの規則に違反している信号をRTLレベルでグローバルにチェック
- 構造およびパワー/グラウンド (PG) チェック：アイソレーション・セル、パワー・スイッチ、レベル・シフト、リテンション・レジスタ、常時オン・セルの挿入と接続をインプリメンテーション・フロー全体を通じて検証
- 機能チェック：アイソレーション・セルとパワー・スイッチの機能的な正しさを確認

これらのチェック機能は、RTLから最終レイアウト・ネットリストまであらゆる設計段階で実行できます。また、これらのチェック機能はRTL-to-GDSIIフローのさまざまな段階でFusion Compilerから直接呼び出すことも可能で、インプリメンテーション中にデザインのパワー・インテントが維持されているかを確認できます。RTLスタティック・サインオフ・プラットフォームのVC SpyGlass™はUPFの読み出しをサポートしており、消費電力を考慮したクロック・ドメイン・クロッシング (CDC) およびリセット・ドメイン・クロッシング (RDC) インスタンスの検証が可能です。Formality®による論理等価検証 (LEC)、およびVC Formal™による解析でも同様に消費電力が考慮されます。また、エミュレーション・システムZeBuとプロトタイプリング・ソリューションHAPS®でもUPFが考慮されます。これらすべてのツールおよびテクノロジーで使用される統合デバッグ・プラットフォームVerdiも、消費電力を考慮したデバッグ機能を数多くサポートしています。この結果、パワー・インテント仕様の記述から機能検証のあらゆる段階にわたってシームレスなフローが実現します。

## まとめ

多くのSoCアプリケーションでは、バッテリー動作時間延長のために消費電力を最小化することや、周到的な電力管理によって各種規制および市場の要求に応えることが要求されます。とはいえ、消費電力だけを重視して全体的なPPAの目標を達成できないようでは困ります。そこで、半導体業界ではあらゆる設計段階で電力効率改善を図ることのできるエンド to エンドの設計フローが必要とされています。まず早期段階で電源オプションのアーキテクチャ検討を実施した後、消費電力を考慮したインプリメンテーションと検証を行い、これらすべての工程をUPFで記述した共通のパワー・インテントおよび統合デバッグによって連携させることが必要です。全体的なアプローチを採用したシノプシスのエンド to エンド・ローパワー・ソリューションは、SoC開発のあらゆる段階で電力効率を改善する業界最先端の最も成熟した手段となるものです。あらゆるアプリケーションのあらゆるプロジェクトで、これらの機能をぜひご活用ください。