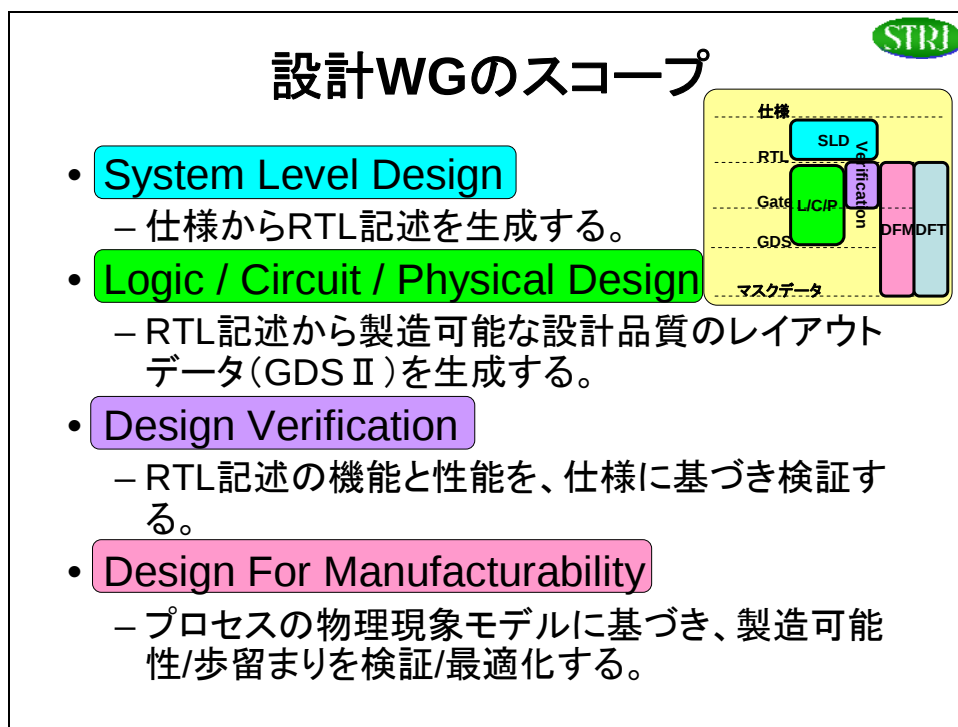


第3章 WG1 設計

3-1 はじめに

WG1(設計 WG)は、国際活動として、ITRS の System Drivers 章と Design 章を設計 TF と協力して担当している。また、国内活動として、①SOC(System On a Chip)の構造・規模を時間軸で定量化し、ロードマップ検討の基礎として提示する活動と、②設計技術課題を時間軸で定量評価し、解決策を提示する、所謂、「設計技術ロードマップ」作成の活動を中心に取り組んでいる。

ITRS での設計のスコープは、SOC の「仕様」から「製造可能なマスクデータ」を作成するまでの一連の設計作業工程であり、図表 3-1 に示すように、ITRS では、SLD(System Level Design)、L/C/P(Logic/Circuit/Physical Design)、Design Verification、DFM(Design for Manufacturability)、DFT(Design For Test)の 5 つの枠組みで構成される。WG1(設計 WG)はこのうち、SLD、L/C/P、Verification、DFM をスコープとしており、DFT については WG2 が担当することになっている。



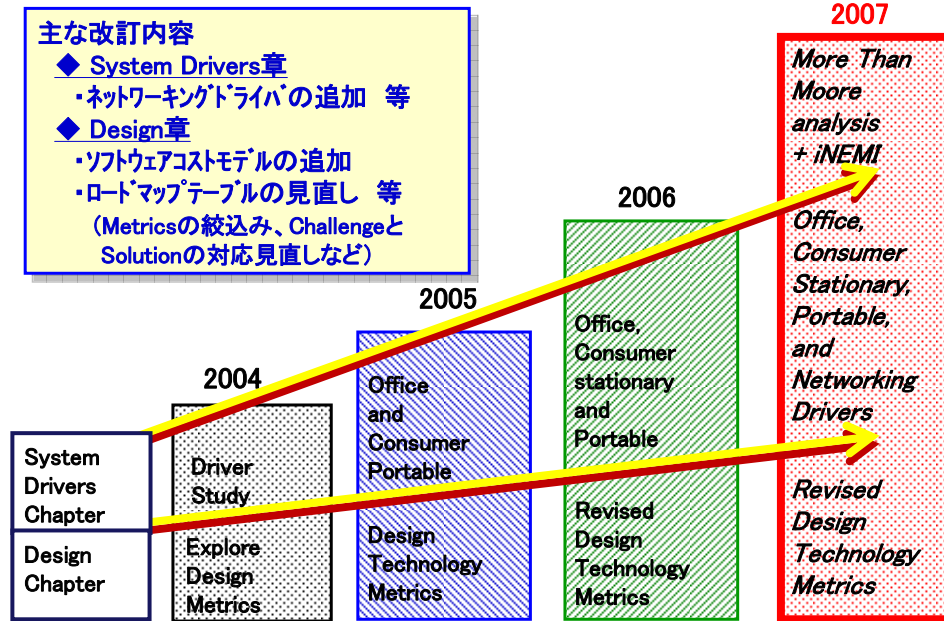
図表 3-1 WG1 設計のスコープ

2007 年度活動として、一つは国際担当を中心とした国際活動を行い、ITRS2007 に対して貢献を行った。もう一つの国内活動は、「SOC 設計技術ロードマップの詳細化・定量化」というテーマで、「論理検証」と「物理設計」の 2 つのサブワーキングによる「設計生産性向上」に向けた課題抽出と解決策の詳細化と定量化の活動を行った。

3-2 国際活動(ITRS2007 の概要)

ITRS2007 で改訂された System Drivers 章と Design 章の概要を紹介する。図表 3-2 で示したように System Drivers 章には新たにネットワーク機器向けのネットワーキングドライバーが追加された。また、Design 章では設計技術に対する Requirement table の Metrics の絞込みと Requirement と Solution の項目の対応付けの見直しが行われた。また、新たにソフトウェアコストモデルも追加された。

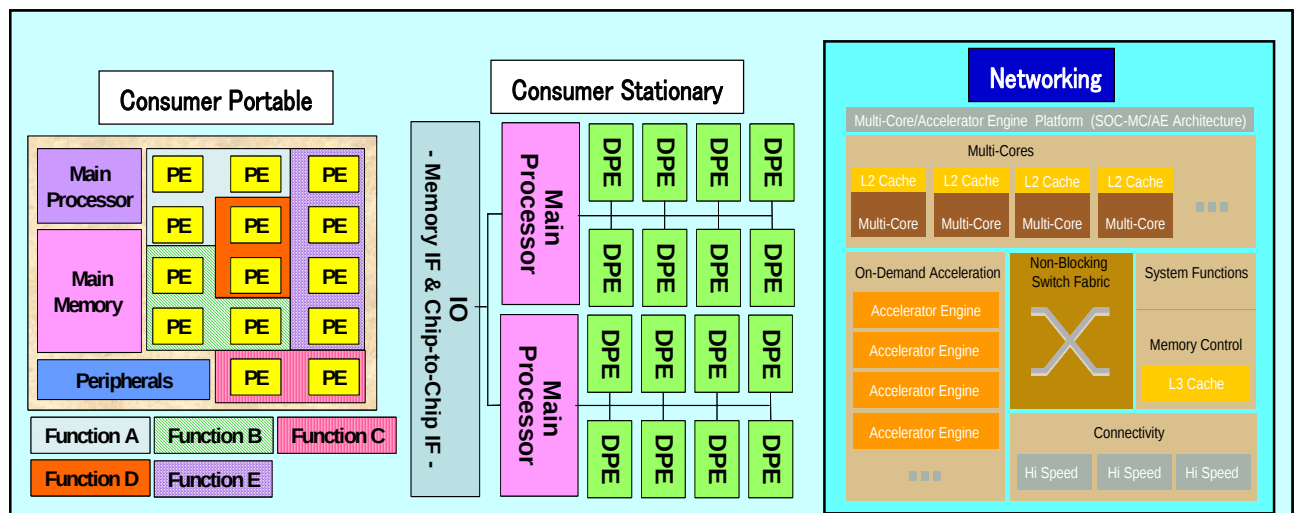
ITRS2007 Overview



7

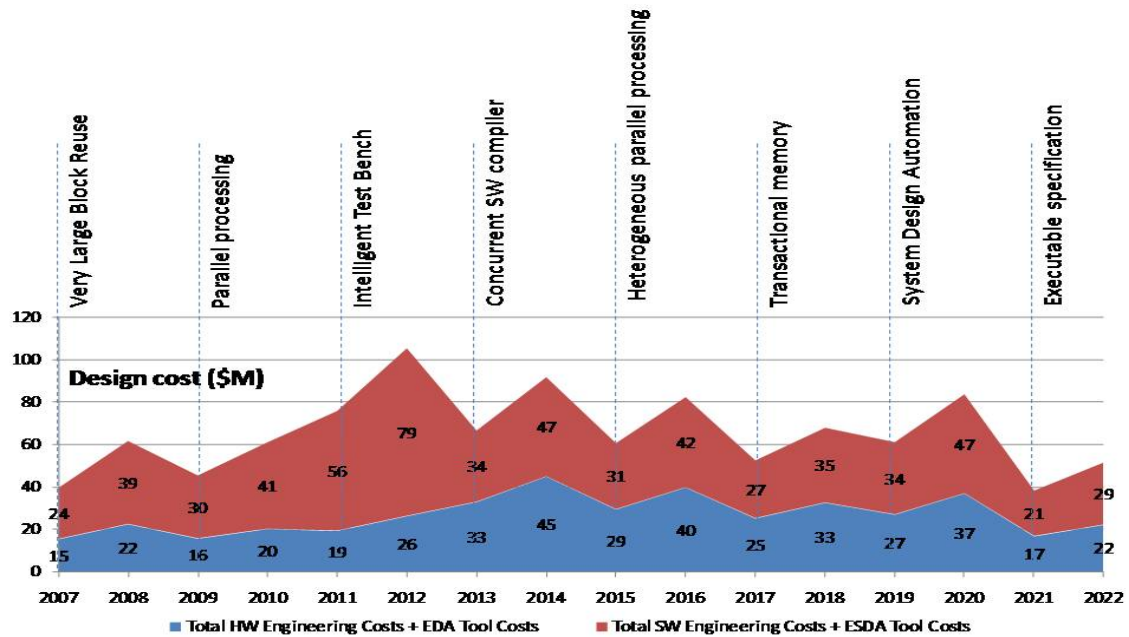
図表 3-2 ITRS2007 の System Drivers 章と Design 章の概要

図表 3-3 に System Drivers 章に掲載された各種のシステムドライバーを示す。日本から提案して掲載済みの Consumer Portable と Consumer Stationary のシステムドライバーに、US が担当の Networking のシステムドライバーが新たに加わった。本ドライバーの特徴は、マルチコアと各種のアクセラレーションエンジンと高速なデータ転送のための高バンド幅のスイッチング回路を搭載した構成になっていることである。



図表 3-3 各種システムドライバー

図表 3-4 に Design 章に新たに掲載されたハード設計とソフト設計の両方を含めたコストモデルを示す。ハードウェア設計コスト以上にソフトウェアの設計コストがかかることが見て取れ、ハード設計だけではなくソフトウェア設計用の設計技術の革新も必要である。



図表 3-4 Impact of Design Technology on SOC Consumer Portable Implementation Cost

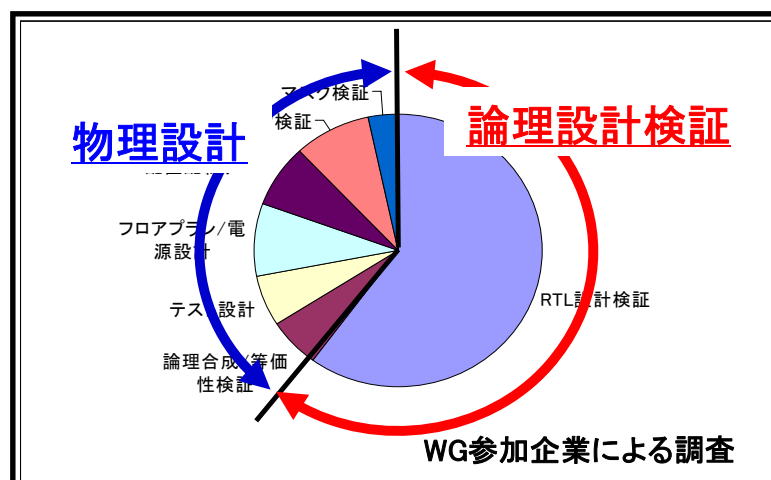
設計 WG からの ITRS2007 への主な貢献内容としては

- ①System Drivers 章の Consumer Portable と Consumer Stationary の各種テーブル数値の見直し
- ②Design 章の L/C/P と DFM の課題の項目値の確認と修正案の提示による完成度向上が上げられる。

3-3 国内活動

3-3-1 背景

2005 年度活動として、SLD、L/C/P、Verification、DFM の4つ枠組み毎にサブワーキング形式で SOC 設計技術ロードマップの見直しを行った。2007 年度では、2005 年度の活動をベースに「SOC 設計技術ロードマップの詳細化・定量化」というテーマで「設計生産性」の観点から活動を行った。



図表 3-5 SOC 設計工程の工数分布

図表 3-5 に設計 WG 参加企業による調査での SOC 設計工程の工数分布を示す。この SOC の設計工程を

「論理設計検証」と「物理設計」の二つの工程に分割し、それぞれに対して「論理検証」と「物理設計」の 2 つのサブワーキングで、「設計生産性向上」に向けた課題抽出とその解決策の詳細化と定量化の活動を行った。

3-4 論理検証サブワーキング活動

本節では、論理検証サブワーキングの活動について報告する。

論理検証サブワーキングでは、次の 2 つを目標に活動を行った。

- ① 論理検証の課題を分析し、中長期で実現すべき技術要求と解決策を提示する
- ② 2010 年の技術要求が実現された場合の生産性向上率を定量化する

なお、設計生産性の定義は ITRS で定義されている次の式とした。

$$\text{設計生産性} = \frac{\text{システム複雑度}}{\text{設計工数}}$$

ここで、システム複雑度は回路規模で表現されるものである。また、検討する論理設計検証工程の範囲(スコープ)は、2005 年度の活動では SLD の範疇としていた高位レベルの性能検証も含めて、HW/SW 分割後の機能仕様から RTL 設計完了までのハードウェア機能検証工程とした。

3-4-1 中長期での論理検証の技術課題の抽出と分類

まず、最初に 2007 年時点での機能(論理)検証の課題の洗い出しを行った。2005 年度からの変化を確認する意味も含めて、現状の課題を各社の設計現場から持ち寄り、分析を行ない、重要課題を抽出した。具体的には、サブワーキングに参加の 7 社から持ち寄った合計 75 項目の課題を図表 3-6 の 15 項目に整理・統合した。

明確で誤解がない仕様の作成
検証対象の状況に応じた最適な検証戦略の策定
最適な検証のマネジメント
網羅的な検証項目の抽出と検証順序の最適化、および仕様変更への対応
IP モデルの整備と検証品質
シミュレーションの高速化
クロックドメイン検証(*)
デバッグの効率化
ランダム検証(*)
設計レベル間検証
機能カバレッジの統一(*)
設計初期段階での性能検証
高精度時間モデルが必要となる回路の検証(*)
機能検証スキルを持つ人材／専門家の育成
検証事例／トラブル情報の共有(*)

図表 3-6 論理検証の重要課題抽出の中間結果

この中で、(*)の項目は既に解決策があり対応中のものや論理検証の範疇外ということで削除し、最終的には、図表 3-7 のように重要課題として 10 項目にまとめることができた。また、それらは「戦略策定」「仕様・設計」「検証実行」の 3 つの設計レベルに分類することができた。

設計レベル	重要課題
戦略策定	検証対象の状況に応じた最適な検証戦略の策定
	最適な検証のマネジメント
	IP モデルの整備と検証品質
	機能検証スキルを持つ人材／専門家の育成
仕様設計	明確で誤解がない仕様の作成
	網羅的な検証項目の抽出と検証順序の最適化、および仕様変更への対応
	シミュレーションの高速化
検証実行	デバッグの効率化
	設計レベル間検証
	設計初期段階での性能検証

図表 3-7 論理検証の重要課題抽出の最終結果

3-4-2 論理検証の課題と解決すべき技術要求

10 項目の重要課題それぞれに対して、「課題の説明」、「2007 年時点での対応策(取り組み)」、「2010 年に実現すべき技術要求／解決策」、「2011 年以降の技術要求」の 4 項目でまとめた。

◆ 戦略策定に分類した項目

【重要課題①】： 検証対象の状況に応じた最適な検証戦略の策定

●課題の説明

検証対象に要求される設計品質や適用可能な設計手法に応じて、最適な検証戦略を立案することが難しい。さらに、設計品質や検証レベルの異なったブロックが混在する場合、個々に検証ゴールや手法の選択が必要となり、全体最適な検証戦略を立案することが難しい。仕様変更時に変更内容の影響を正確に把握して、適切な検証戦略に修正することが難しい。

●2007 年時点での対応策(取り組み)

- ・プロジェクトリーダーが全体の検証指針を示して、それに基づいてトップはトップ検証担当が、サブブロックは各サブブロック・リーダーがそれまでの経験に基づいて、検証対象の特性を考慮した検証環境を構築している。
- ・過去の経験や他の設計事例から起こりうる問題点を設計レビューの中で確認して、その防止策や効率的な検証方法を検証手法や環境に取り入れている。

●2010 年に実現すべき技術要求／解決策

【技術要求】

- ・個々の経験やスキルのばらつきによらず高品質な検証環境が構築できる仕組みの実用化。

【解決策案】

- ・標準となる検証手法を確立する。(種々の検証事例に基づいた網羅的な検証手法マニュアルの

作成)

●2011 年以降の技術要求

【技術要求】

- ・検証ターゲットに応じた検証手法が用意され、検証環境が容易に構築できる。

【重要課題②】： 最適な検証のマネジメント

●課題の説明

利用可能なリソースを有効活用し、検証過程での種々の問題を解決して、リスク管理を行いながら最適な QCD のバランスの取れたマネジメントを実行することが難しく、プロジェクトリーダーとプロジェクトメンバのスキルに依存している。

●2007 年時点での対応策(取り組み)

- ・過去の事例から問題点、対処を考え、検証戦略を改良していく。
- ・プロジェクトによっては、独立した検証技術チームが検証を実施したり、コンサルティングを行い検証戦略や検証環境構築を支援することもある。

●2010 年に実現すべき技術要求／解決策

【技術要求】

- ・プロジェクトリーダー/メンバが目指すべきベストプラクティス(やるべき項目、手順、リスク管理など)を明確するためのガイドラインの策定。
- ・プラットフォームベースの検証では、プロジェクトリーダー/メンバのスキルに依存せず最適な検証環境が構築できるよう、変更追加が起こる部分の修正方法が定型化できていること。

【解決案】

- ・検証マネジメントに必要な検討項目をプラットフォームに対するテンプレートとして用意して、各項目の記載内容の詳細定義と事例を示したガイドを作成する。

●2011 年以降の技術要求

【技術要求】

- ・プロジェクトリーダー/メンバのスキルに依存せず、最適な検証のマネジメントができる。

【解決策】

- ・マネジメントガイドに事例等の知識ベースを追加し、様々な条件を入力することで最適なマネジメント手法を生成するシステムを構築する。

【重要課題③】： IP モデルの整備と検証品質

●課題の説明

使用している IP のモデル(ex.高位動作モデル)や検証 IP が不足していたり、提供される IP や検証 IP の品質が十分でないことにより、検証が効率的に行えない。

●2007 年時点での対応策(取り組み)

- ・実績がある IP を、その使用条件のもとで情報共有して使っている。
- ・高位モデルなど不足しているモデルは自前で開発することもある。

●2010 年に実現すべき技術要求／解決策

【技術要求】

- ・高位モデルと RTL モデルがセットで販売され、SoC 開発に十分な IP 品種が流通する。
- ・検証 IP は利用環境によらず網羅的に検証され、利用者の環境で品質確認が容易。

【解決案】

・IP 業界として品質を示すための基準を定めて、定量的な品質表示を可能とする。例えば、標準となる検証手法(Verification Methodology Manual(VMM) など)を決めて、これに基いた検証 環境を IP と合わせて提供し、検証レベルがユーザー側でも確認できるようにする。

●2011 年以降の技術要求

【技術要求】

・完全に検証され、仕様通りに動作する IP が流通する。

【重要課題④】： 機能検証スキルを持つ人材／専門家の育成

●課題の説明

機能検証にはダイナミック検証やスタティック検証など複数の技術が必要で、検証用言語も複数存在するため、設計者が検証技術もカバーするのは困難で、専門家の育成が必要になっている。

●2007 年時点での対応策(取り組み)

・機能検証ガイドの作成や、検証技術講座などで基本知識を教えて、実務では検証技術担当がプロジェクトに参加して、OJT で対応したり、テストベンチやアサーションのテンプレートを作成して、設計者に引き継ぐことで、検証ノウハウを伝えている。

●2010 年に実現すべき技術要求／解決策

【解決案】

・検証エンジニアのスキル要件を明確にする。
・体系的な教育プログラムを整備する。
・各社からの知識、ノウハウを持ち寄り、共同でガイドライン作成、教育講座開設を行う。
・受講の容易さのために Web 教育による基本レベルの裾野展開を行う。
・トランジスタや論理回路の物理現象理解の基礎からの教育も実施する(現在、論理合成等 EDA ツールの利用により、電気の基本的な振る舞いを理解する部分が空洞化している)。

●2011 年以降の技術要求

【一般的要求】

・知識ベースを利用したコンサルティングシステムや専門家による支援を充実させ、個々のスキルのばらつきを補う仕組みを用意する。

◆ 仕様・設計に分類した項目

【重要課題⑤】： 明確で誤解がない仕様の作成

●課題の説明

SOC 規模拡大により様々な機能が組み合され、仕様が複雑になってきた。このため、個々の機能仕様は正しくても全体としては矛盾している「仕様の誤り」や、条件が網羅されてなかったり、暗黙の前提に基づいた記述による「仕様の漏れ」などが生じている。「仕様の誤り」がある場合は、仕様通りに設計しても SOC は意図通りの動作をしない。「仕様の漏れ」がある場合は、設計者の解釈が入り込むことにより、SOC は意図通りの動作をしない。

●2007 年時点での対応策(取り組み)

・仕様を設計と検証の別々の立場から読み下し、相互確認をすることで仕様の妥当性を検証している。
・シミュレーションによる機能検証とともに、プロトタイプングによる実機評価をおこなうことで、二重確認を行っている。
・システムレベル検証ツールで、シミュレーションによる動作確認で、設計の初期段階で仕様を検証

している。

●2010 年に実現すべき技術要求／解決策

【技術要求】

- ・機能仕様が誰が表現しても画一的な表現になるような仕組みを用意する
- ・仕様記述の標準化により理解が共通になり、誤解による設計誤りを防ぐとともに、相互確認が容易となる。
- ・仕様記述を設計資産化出来るので、再利用により記述ミスが軽減され仕様の完成度が上がる。

【解決案】

- ・機能仕様については、機能動作とその入出力で定義する。機能カテゴリ(制御(状態遷移系)、信号処理(データパス系)など)に対して仕様テンプレートが用意され、入出力についても、画像、音声、制御などのライブラリから選択して機能仕様を表現する。制約条件として必要な情報のみ追記する。機能はキーワード毎に動作を記述する。キーワード毎に各機能モジュール間で相互チェックを支援する仕組みが提供される。

●2011 年以降の技術要求

【技術要求】

- ・仕様の矛盾、網羅性が容易にチェックできること。例えば、キーワード毎に矛盾がないことが言語解析により確認される。非機能仕様も、機能仕様に対応する形で、フォーマルに記述できること。

【重要課題⑥】： 網羅的な検証項目の抽出と検証順序の最適化、および仕様変更への対応

●課題の説明

仕様から検証項目を網羅的に抽出する手法や適切な検証順序を決めることが定式化されておらず、仕様と検証項目の対応を確認出来ないため、プロジェクトメンバーのスキルに依存している。

このため、仕様変更時に仕様の変更内容と検証項目との対応が個別対応となり、全体の整合性を確認することが困難となっている。

●2007 年時点での対応策(取り組み)

- ・設計レビューで検証項目の確認を行い、抽出の妥当性や網羅性を確認している。
- ・網羅性を上げるために、過去の不具合事例から検証項目を追加する。
- ・一部のツールには、検証回路にランダムなバグを埋め込み、その検出結果からテストベンチの検証項目網羅性を評価するものがある。
- ・仕様変更では、担当者が変更内容をそれぞれ確認して、必要に応じて検証項目を修正した上で検証を行う。検証項目の妥当性や、検証環境の変更箇所は、設計レビューで確認し共有化している。

●2010 年に実現すべき技術要求／解決策

【技術要求】

- ・仕様と検証項目の対応が確認できる仕組みを用意する。

【解決案】

- ・検証項目毎に仕様との対応をひも付けできる仕組みを作り、検証項目に漏れが出ない仕様表記方法を確立する。仕様変更があった場合に、変更履歴をたどることで、変更が必要な検証項目を見つけられる仕組みも用意する。

●2011 年以降の技術要求

【技術要求】

- ・検証項目の漏れを自動チェックできること。

【解決案】

- ・IP ベース及プラットフォームベース設計の進展で、多くは検証項目が各 IP とリンク付けされたデータとして整備可能である。IP の組み合わせについては、各 IP の接続情報を元に、検証項目の候補をリストアップする。

◆ 検証実行に分類した項目

【重要課題⑦】： シミュレーションの高速化

●課題の説明

大規模化と機能の複合化により、検証項目が膨大になり、シミュレーションが検証期限までに終わらない状況にある。高位検証やプロトタイピング等の検証手段はあるが、(RTL)サインオフとしての機能検証は RTL ダイナミックシミュレーションに基づいたものとなるため、検証時間の増大を抑制する決め手にはなっていない。

●2007 年時点での対応策(取り組み)

- ・シミュレーションの高速化としてエミュレータやアクセラレータを利用しているが、エミュレータやアクセラレータで回路を動作させるまでの立ち上げに時間がかかったり、回路によっては修正が必要になり、高速化が十分行えないこともある。
- ・マルチ CPU の計算機や複数台の計算機を利用して、複数 CPU での分散実行による高速化が、比較的低コストで実現可能となってきた。また、マルチコア計算機をターゲットとした論理シミュレータが商品化されている。

●2010 年に実現すべき技術要求／解決策

【技術要求】

- ・2 桁以上の高速化。

【解決案】

- ・マルチ CPU 分散実行対応のシミュレータとその低コスト化。
- ・大規模高速でプログラミングが容易なプロトタイピングの実現。

●2011 年以降の技術要求

【技術要求】

- ・リアルタイムシミュレーションの実現。

【重要課題⑧】： デバッグの効率化

●課題の説明

問題の場所の特定は、シミュレーション結果などの目視確認が主となるため、時間がかかり、効率化が難しい。

●2007 年時点での対応策(取り組み)

- ・目視作業を軽減するために、アサーションの利用やフォーマル検証などを使う試みをしている。

●2010 年に実現すべき技術要求／解決策

【技術要求】

- ・アサーションの生成やプロトタイプ・プロービングの容易化。

【解決案】

- ・擬似エラーの解析支援技術の向上。
- ・アサーション記述を統一して、ツールの使いやすさを向上させる。

- ・波形表示ツールのトレーサビリティ強化などのデバッグ機能の向上。
- ・FPGA プロトタイプの内蔵ノードのモニタ機能やデバッグ機能の向上。

●2011 年以降の技術要求

【技術要求】

- ・仕様と検証項目から不良箇所を特定する技術の開発。

【重要課題⑨】：設計レベル間検証

●課題の説明

高位設計(C レベル)の導入により検証を効率化しようとしているが、高位レベルと RTL レベルの等価性検証ができないことや、高位のテストベンチを、RTL ではそのまま利用出来ないため、RTL のための修正が必要となり、効率化が阻害されている。

フォーマル検証技術は制約(回路規模、使い方)があり、効率的な使い方が難しい。

●2007 年時点での対応策(取り組み)

- ・設計レベル間の検証は主として動的シミュレーションに頼っている。テストベンチの流用については、高位テストベンチを修正することで、RTL レベルで使えるようにしたり、ベクタをファイルダンプしてテスト系列として利用している。
- ・フォーマル検証、等価検証については、回路規模や使い方の制約を理解した専門家のサポートを得ながら使っている。

●2010 年に実現すべき技術要求／解決策

【技術要求】

- ・等価検証が実用的に使える。

【解決案】

- ・動作合成が回路タイプによらず処理できて、動作合成の単位では等価検証を実用化
そのためには、以下のことが必要となる。
 - ー抽象レベルの厳密な定義と変換ルールの規程。
 - ー動作合成のレジスタ生成情報などを利用して等価検証の適用範囲を拡大。
 - ーノウハウ、経験を盛り込んだ知識ベースの自動分割と並列処理による全体検証。

●2011 年以降の技術要求

【技術要求】

- ・機能仕様と制約条件からアーキテクチャが合成され、通信部分はインタフェース合成で生成される。機能仕様の正当性確認と ゲートレベル以降のタイミング検証のみで実装可能となる。動作合成、論理合成によりゲートレベルまで自動的に生成できること。

【重要課題⑩】：設計初期段階での性能検証

●課題の説明

設計初期段階(上流)での性能見積もりとその検証手法が未確立であり、特に、電力、システム性能、動作周波数等は下流工程で初めて精度高い値が算出(抽出)されるため、設計手直し等の設計の非効率化を招いている。

●2007 年時点での対応策(取り組み)

- ・上流では、設計経験、ノウハウに基づく人手による電力見積もりや高位(C 言語など)モデルによるシミュレーションでシステム性能検証を実施している。最終確認はゲートレベルの検証で行っている。

●2010 年に実現すべき技術要求／解決策

【技術要求】

- ・RTLで消費電力や性能の確認が高精度に出来ること。

【解決策】

- ・論理合成アルゴリズムを内蔵するなど、RTLレベルでゲートレベル精度の消費電力検証(見積もり)や動作周波数の確認が出来るようにする。

●2011 年以降の技術要求

【技術要求】

- ・仕様書の非機能要求として指定した電力、システム性能記述の制約に基づいて高位合成(アーキテクチャ、動作、論理)で回路を生成する。

3-4-3 2010 年時点での技術要求が実現された場合の生産性向上率の見積もり

10 項目の重要課題に対して 2010 年時点の技術課題の解決策が実現された場合に、どの程度設計生産性の向上が見込めるかについて検討した。各社の現場の意見も取り入れて見積もった結果を図表 3-8 に示す。戦略策定で 25%、仕様・設計で 35%、検証実行で 50%の改善が見込まれるとなった。

設計レベル	重要課題の解決策	生産性向上率の見積もり内容	向上割合
戦略策定	最適な戦略を作るための共通な手順が確立される	・戦略策定とマネジメント改善 →10%向上 ・人材育成 →10%向上 ・IPモデル整備 →5%向上	25%
	ベストプラクティスに基づいたマネジメント・ガイドラインが策定される		
	IP利用に必要な各種モデルが用意され、その品質が保証されている		
	各エンジニアのスキル要件が明確になり、教育プログラムが用意される		
仕様・設計	共通な仕様表現方法が確立される	・仕様の明確によるミス削減 →50%向上 ・網羅性向上で検証作業増 →25%悪化 ・仕様変更対応効率化 →10%向上	35%
	仕様と検証項目の対応が確認できるようになる		
検証実行	2桁以上の高速化が実現される	・検証実行の高速化、デバッグ効率化 →50%向上 ・手変換ミスによる手戻りが5回→2.5回でシミュレーション回数半減 →50%向上 ・見積もり誤差による設計手戻りが2回→1回に半減 →50%向上	50%
	アサーションが自動生成され、プロトタイプの内外部状態が容易に観測できる		
	等価検証が実用化される		
	RTLで性能や消費電力の確認が出来る		

図表 3-8 2010 年時点の生産性向上率の見積もり

次に、これらの向上率が論理設計検証工程全体にどの程度影響するかを算出した。(図表 3-9)

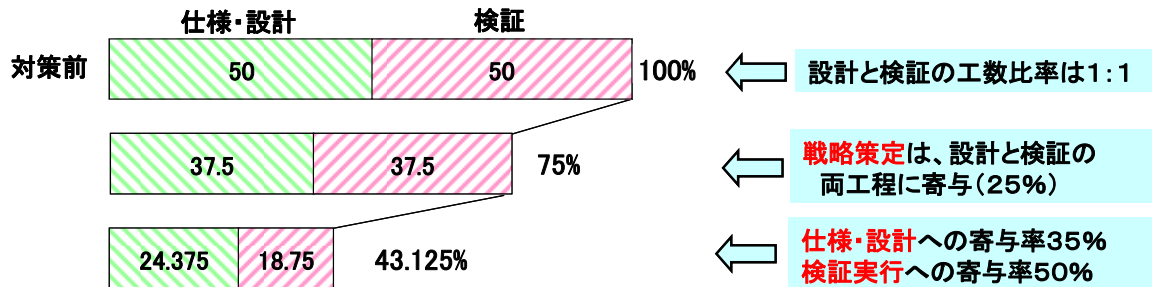
算出に際して、仕様・設計に費やす工数と機能検証に費やす工数がそれぞれどの程度の割合を占めるかを、最初に検討した。一般には、検証作業が設計工数に占める割合が 70%といわれるが、RTL までの設計検証に限って検討したところ、各社の状況や専門誌などで公表されているデータを元にとすると 1:1 の割合で工数がかかっているという結論を得た。

これに対して、まず、戦略策定は、設計と検証にかかわらず改善が影響すると考え、結果、75%の改善が見込める。一方、仕様・設計の項目は設計に、検証実行の項目は検証の工数にのみ影響すると考えて計算したところ、2010 年の解決策の実現により、全体として機能検証に対して「2.3 倍(=100/43.125)」の改善が得られるこ

とがわかった。

ITRS のロードマップによれば、2010 年に求められる設計生産性は 2007 年比 1.96 倍であるので(図表 3-10)、2010 年の技術要求を実現することで、ロードマップ要求値に対してさらに約 20%の改善が実現できることになった。

しかし、これを実現することは容易ではなく、業界全体での標準化の活動や EDA ベンダーとの協業推進による技術開発の加速など多くの努力が必要である。



図表 3-9 2010 年時点の論理検証の生産性向上率

	2007	2008	2009	2010	2011	2012	2013	2014
Trend: SOC total logic size (normalized to 2007)	1.00	1.29	1.62	2.12	2.64	3.24	4.07	5.29
Requirement: % of reused design	38%	42%	46%	50%	54%	58%	62%	66%
Requirement: Productivity for new designs (normalized to 2007)	1.00	1.25	1.54	1.96	2.38	2.84	3.47	4.37

図表 3-10 ITRS2007 の設計生産性要求テーブル

3-5 物理設計サブワーキング活動

本節では、物理設計サブワーキングの活動について報告する。

物理設計の現状を見ると、設計生産性を阻害している要因として、回路規模で表現されるシステム複雑度要因だけではなく、プロセスの微細化の進展に伴う消費電力対策や製造ばらつき対策といった要因も大きいと考えられる。

そこで、回路規模以外の要因を「シリコン複雑度」要因と定義し、物理設計の設計生産性の式を

$$\text{設計生産性} = \frac{\text{システム複雑度} + \text{シリコン複雑度}}{\text{設計工数}}$$

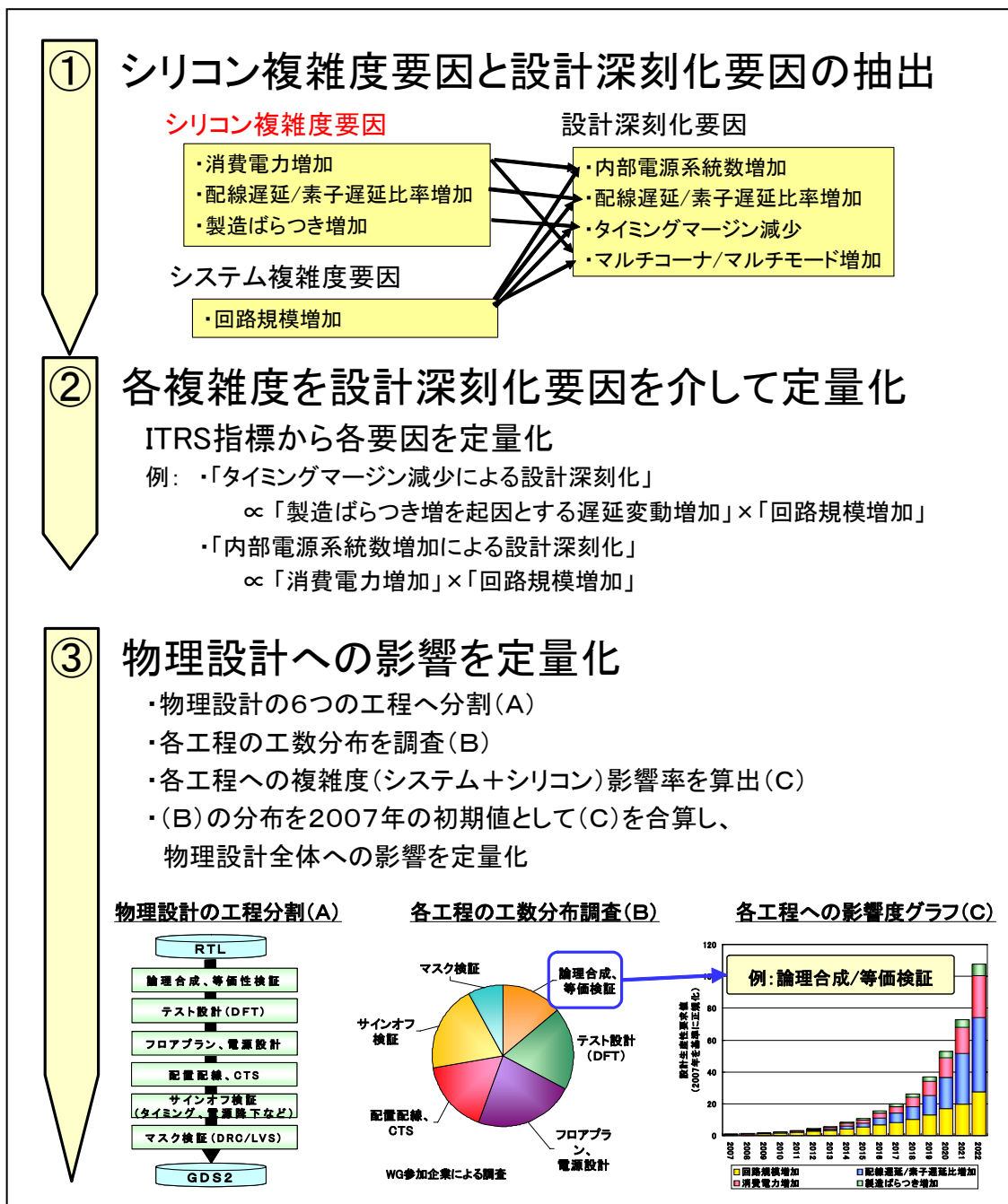
と定義した上で、次の 2 つを目標に活動を行った。

- ① シリコン複雑度の影響を加えた物理設計の生産性要求値を定量化する。
- ② シリコン複雑度を加えた際の設計生産性向上のための解決策を提示する。

なお、物理設計工程の範囲としては、RTL 記述からレイアウトデータ(GDS II)を生成するまでとした。

3-5-1 シリコン複雑度を考慮した設計生産性要求の定量化の手順

シリコン複雑度を考慮した設計生産性要求の定量化の手順を順次説明していく。(図表 3-11)



図表 3-11 シリコン複雑度を考慮した設計生産性要求の定量化手順

3-5-1-1 シリコン複雑度要因とそれに関する設計深刻化要因の抽出

まず、現状の物理設計工程の工数を増加させている課題を調査した上で、シリコン複雑度に影響する項目をピックアップし、「消費電力増加」「配線遅延／素子遅延比率増加」「製造ばらつき増加」の3つの項目にまとめた。

次に、それらを具体的な物理設計工程の実作業の工数増に直接影響する項目に落とし込む作業を実施し、それを「設計深刻化要因」としてまとめた。具体的には、「内部電源系統数増加」「配線遅延/素子遅延比率増加」「タイミングマージン減少」「マルチコーナ/マルチモード増加」の4つとした。

たとえば、消費電力増加への対策として、回路毎に最適な電圧制御をしたり、ケイタイ電話などでは、パワーシャットダウン制御をしたりするために、内部電源系統数を増やす必要があり、そのことが、回路規模の増加とも関連して、設計複雑度を上げる要因になっているからである。

3-5-1-2 各複雑度を設計深刻化要因を介して定量化

次に、今回抽出した消費電力増などのシリコン複雑化による影響を ITRS の各指標から設計深刻化要因を介して定量化に取り組んだ。

ITRS から引用した各種指標を図表 3-12 に示す。図表 3-13 は、それらの指標化から、設計深刻化要因を定量化するための要素へ数値化するためのテーブルを示し、図表 3-14 は、最終的な設計深刻化要因の数値テーブルを示す。例えば、タイミングマージン減少による影響に関しては、まず、「製造ばらつき増を起因とする遅延変動増加」と「回路規模増加」による影響を掛け合わせたものに比例すると定義し、さらに「遅延変動増加」に関しては、「ゲート長の平方根に反比例する」という仮定から、また、「回路規模」に関しては、「チップ面積一定というモデル」と「配線ロードマップ(具体的にはメタル 1 の配線ピッチ)」から決定している。

★ITRSからの引用				Year		
要素	ITRS Tableでの名称または補足		Unit	2007	2008	2009
PIDS Tableから引用						
素子遅延	Intrinsic delay, τ		ps	0.64	0.55	0.51
電源電圧	Power supply voltage		V	1.10	1.00	1.00
ゲート長	Physical gate length		nm	25.00	22.00	20.00
Source/Drain Leakage密度	Source/Drain Subthreshold Off-State Leakage		$\mu\text{A}/\mu\text{m}$	0.34	0.71	0.70
Gate Leakage密度	Maximum Gate Leakage current density		A/cm ²	8.00E+02	9.09E+02	1.00E+03
Interconnect Tableから引用						
MET1ピッチ	Metal1: Wiring pitch		nm	136	118	104
配線遅延	Interconnect RC delay (ps) for 1mm Cu intermediate wire, assumes width-dependent scatt		ps	682	1039	1413
K値(配線間材料の比誘電率)	Interlevel metal insulator Effective dielectric constant (κ)			2.90	2.90	2.60
仮定に基づき算出(基本的には、System Driver Modelと同じ考え方)						

図表 3-12 ITRS から引用した各種指標テーブル

★ITRSからのトレンド予測									
深刻化要因の 定量化の要素	分類	比率 @2007	要素(x)	関係式	Unit	2007	2008	2009	
動作周波数	素子遅延に反比例と仮定				Nomalized	1.00	1.16	1.25	
総信号容量	K値に比例し、MET1ピッチに反比例と仮定 総信号容量は、配線容量が支配的であると仮定。信号数は回路規模(MET1ピッチの二乗に反比例)に比例するが、平均信号長はMET1				Nomalized	1.00	1.15	1.17	
平均ゲート幅	MET1ピッチに比例と仮定				Nomalized	1.00	0.87	0.76	
製造バラツキによる遅延変動率	ゲート長の平方根に反比例と仮定				Nomalized	1.00	1.07	1.12	
消費電力(改革・改善無し)	設計技術上の技術革新・技術改善が無い場合のトレンド				Nomalized	1.00	1.48	1.65	
	Switching Power	60%			Nomalized	1.00	1.11	1.22	
			電源電圧	x^2	Nomalized	1.00	0.91	0.91	
			動作周波数	x	Nomalized	1.00	1.16	1.25	
			総信号容量	x	Nomalized	1.00	1.15	1.17	
	Source/Drain Leakage Power	35%			Nomalized	1.00	2.19	2.45	
			Source/Drain Leakage密度	x	Nomalized	1.00	2.09	2.06	
			電源電圧	x	Nomalized	1.00	0.91	0.91	
			平均ゲート幅	x	Nomalized	1.00	0.87	0.76	
			回路規模	x	Nomalized	1.00	1.33	1.71	
	Gate Leakage Power	5%			Nomalized	1.00	1.05	1.19	
			Gate Leakage密度	x	Nomalized	1.00	1.14	1.25	
			電源電圧	x	Nomalized	1.00	0.91	0.91	
			平均ゲート幅	x	Nomalized	1.00	0.87	0.76	
			ゲート長	x	Nomalized	1.00	0.88	0.80	
			回路規模	x	Nomalized	1.00	1.33	1.71	

図表 3-13 深刻化要因の定量化のための各種要素テーブル

深刻化要因	要素(x)	比率 @2007	要素(x)	関係式	Unit	2007	2008	2009
回路規模の増加	回路規模	100%			Nomalized	1.00	1.33	1.71
			チップ面積	x	Nomalized	1.00	1.00	1.00
			MET1ピッチ	$1/x^2$	Nomalized	1.00	0.87	0.76
配線遅延/素子遅延比増加	配線遅延/素子遅延比	100%			Nomalized	1.00	1.77	2.60
			配線遅延	x	Nomalized	1.00	1.52	2.07
			素子遅延	$1/x$	Nomalized	1.00	0.86	0.80
内部電源系統数増加	消費電力(改革・改善無し)	100%			Nomalized	1.00	1.48	1.65
	内部電源系統数は、消費電力(改革・改善無し)に比例すると仮定							
タイミングマージン減少	遅延変動率 × 回路規模	100%			Nomalized	1.00	1.42	1.91
			製造バラツキによる遅延変動率	x	Nomalized	1.00	1.07	1.12
			回路規模	x	Nomalized	1.00	1.33	1.71
マルチコア/ マルチモード増加	内部電源系統数 × 回路規模	100%			Nomalized	1.00	1.97	2.81
			内部電源系統数	x	Nomalized	1.00	1.48	1.65
			回路規模	x	Nomalized	1.00	1.33	1.71

図表 3-14 設計深刻化要因の数値テーブル

3-5-1-3 物理設計への影響を定量化

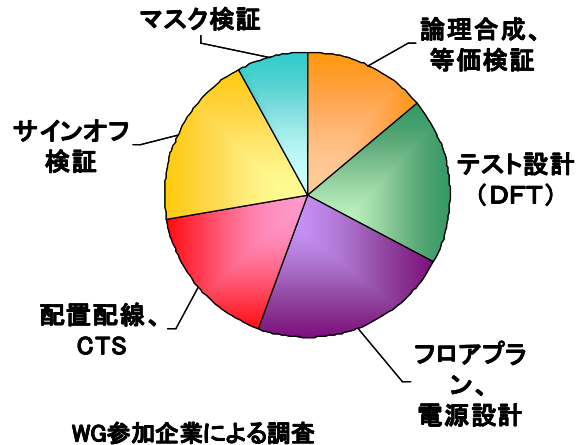
手順の最後は、複雑度の影響の物理設計工程への反映である。

まず、物理設計工程を図表 3-15 のように 6 つのサブ工程に分割した。また、各サブ工程の工数分布をサブワーキング参加企業で調査した。

物理設計の工程分割



各工程の工数分布調査



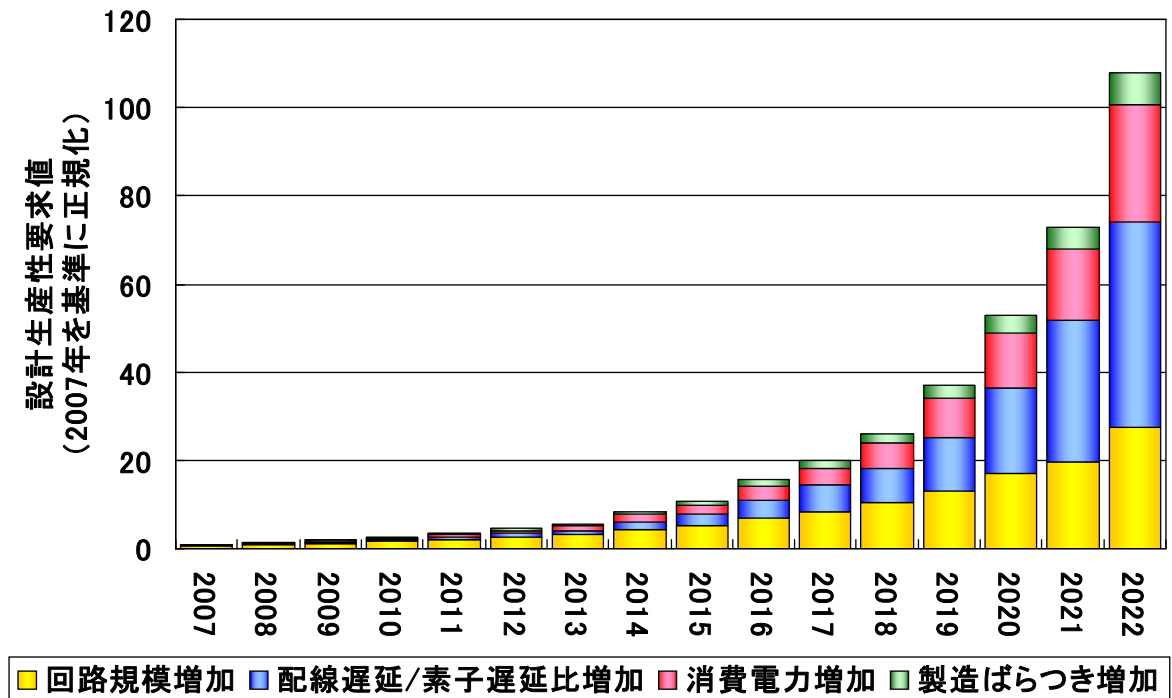
図表 3-15 物理設計の工程分割と各工程の工数分布調査結果

次に、それぞれのサブ工程をさらにその工程内の設計作業に分割し、それぞれの設計作業のサブ工程内の比率と、各設計作業の回路規模要因と4つの設計深刻化要因による影響の比率を、各社の設計現場の実態を調査して決定してテーブルとしてまとめた。「論理合成／等価性検証」工程での結果を図表 3-16 に示す。

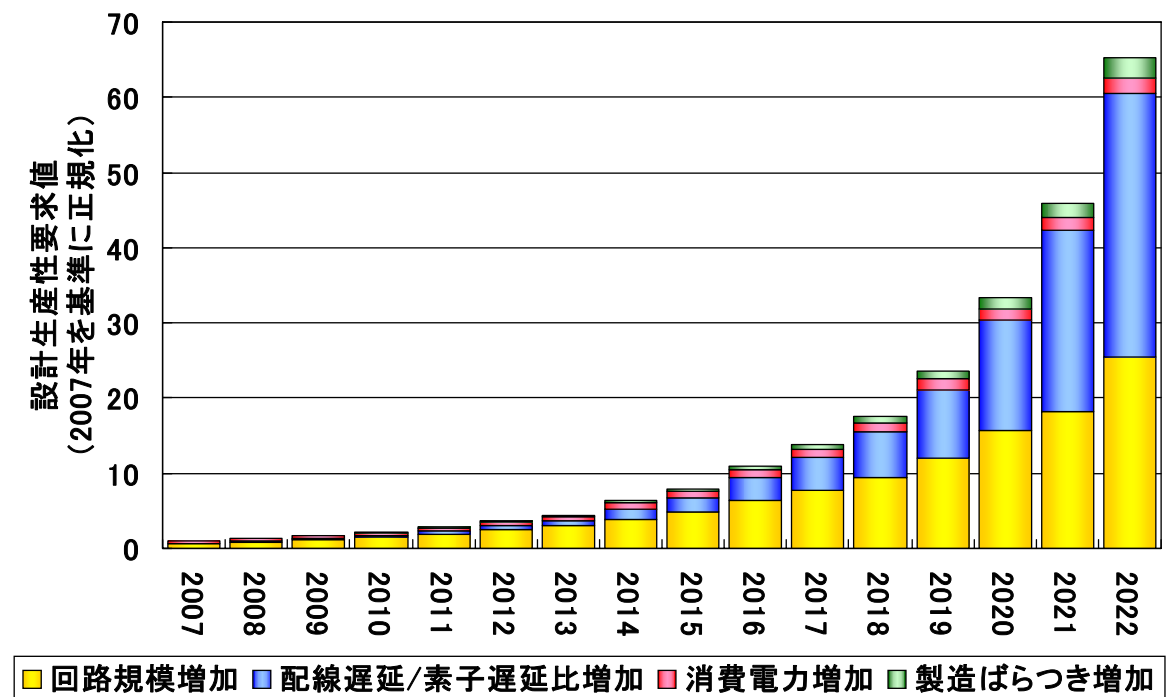
論理合成/等価検証		各設計作業の工数において、回路規模及び阻害要因が占める比率						各設計作業の工数		
設計作業	設計作業 工数比 @2007年	回路規模	設計深刻化要因				合計 (確認用)	2007	2008	2009
			配線遅延/ 素子遅延比 増加	内部電源 系統数 増加	タイミング マージン 減少	マルチコーナ マルチモード 増加				
1 合成スクリプトの作成	10%	100%					100%	0.1	0.1	0.2
2 タイミング制約(SDC)の作成と検証	20%	100%					100%	0.2	0.3	0.3
3 論理合成の実行	5%	100%					100%	0.1	0.1	0.1
4 合成結果の確認と合成スクリプト&SDCの修正または最適化	40%	30%	20%	10%	20%	20%	100%	0.4	0.6	0.9
5 論理回路ルール(入力オープン、他)の実行と解析	10%	100%					100%	0.1	0.1	0.2
6 等価検証の実施と結果解析	15%	100%					100%	0.2	0.2	0.3

図表 3-16 「論理合成／等価性検証」工程での設計作業毎の各種設計複雑度への影響比率テーブル

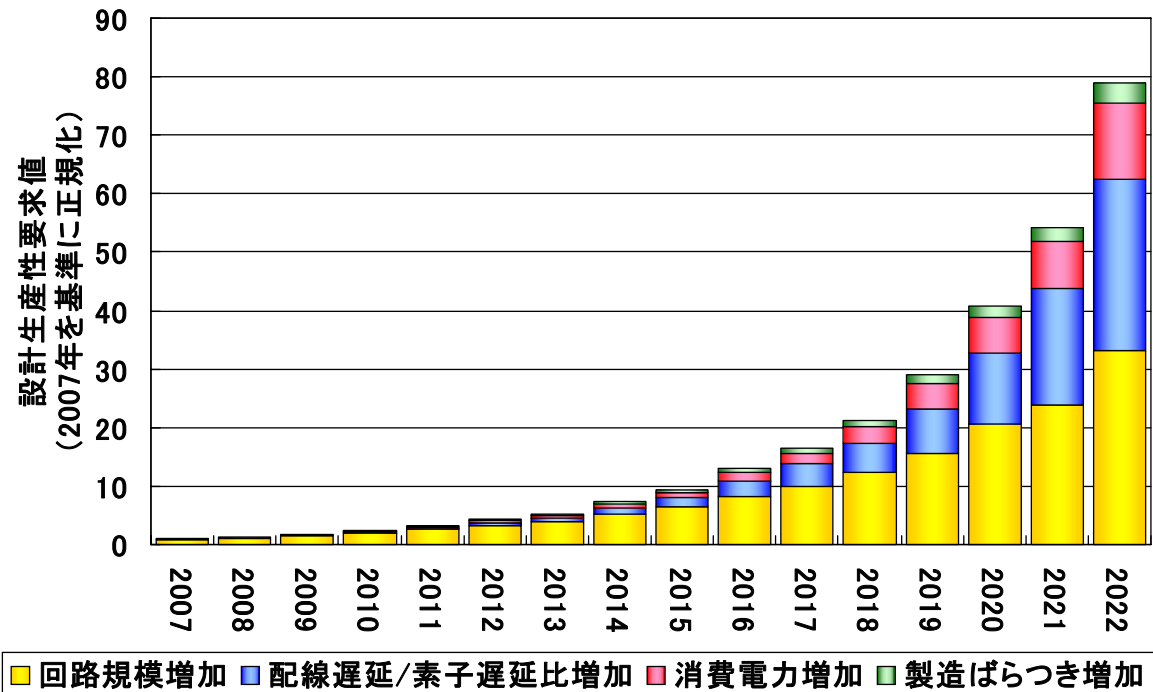
以上の作業の結果として算出された、物理設計のサブ工程毎のシステム複雑度(回路規模増)とシリコン複雑度を考慮した設計生産性要求を図表 3-17 から図表 3-21 に示す。



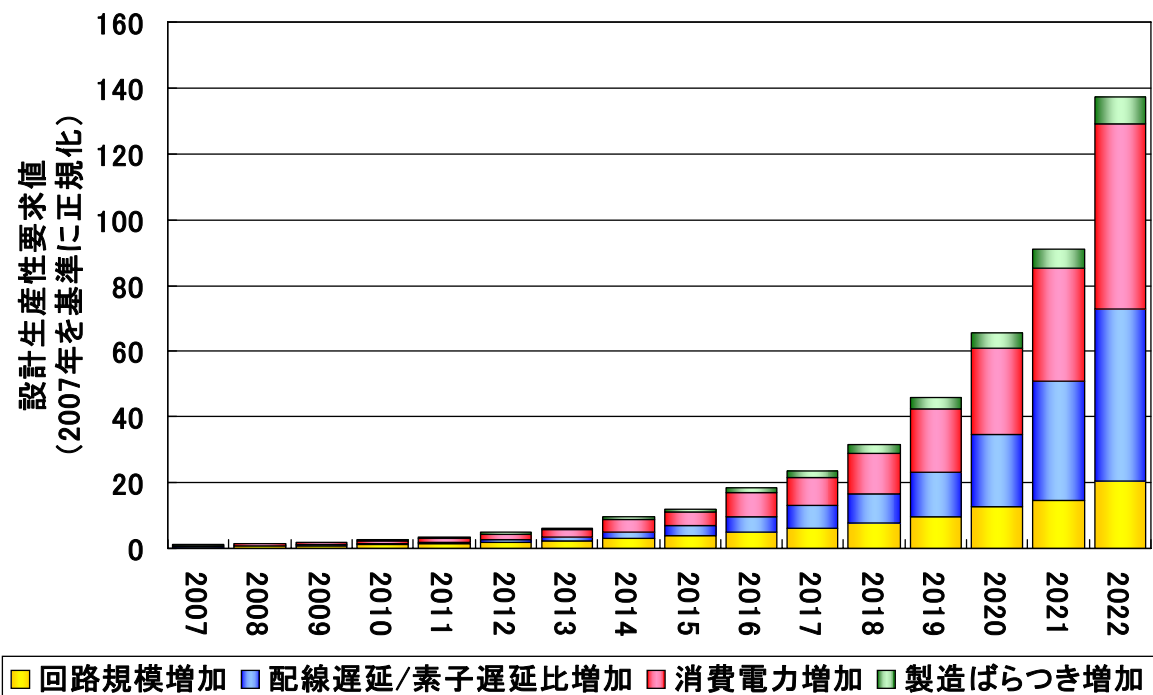
図表 3-17 「論理合成、等価性検証」工程の設計生産性要求



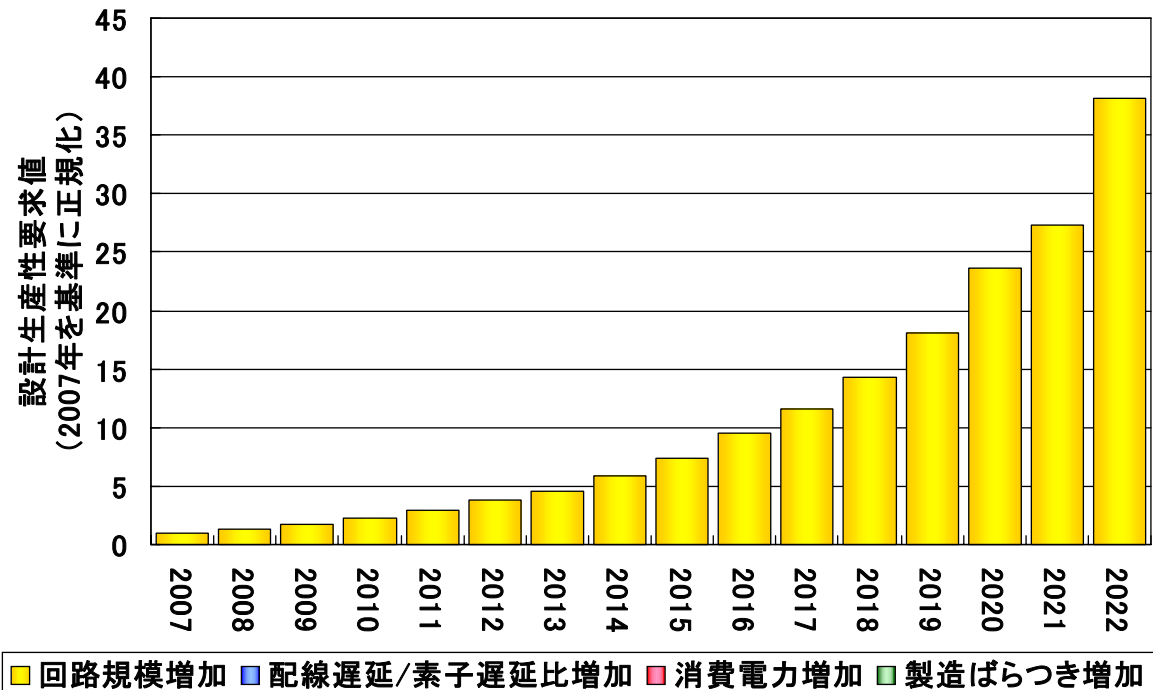
図表 3-18 「フロアプラン、電源設計」工程の設計生産性要求



図表 3-19 「配置配線、CTS」工程の設計生産性要求



図表 3-20 「サインオフ検証」工程の設計生産性要求

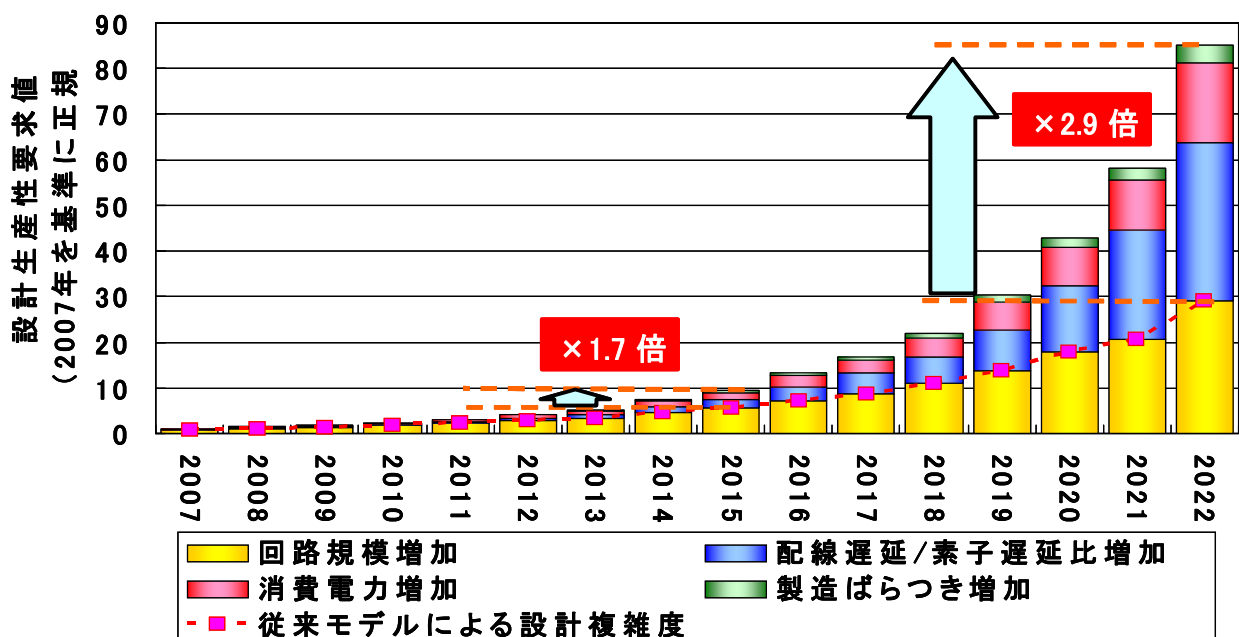


図表 3-21 「マスク検証」工程の設計生産性要求

3-5-1-4 物理設計全体へのシリコン複雑度を加えた設計生産性要求の定量化

最後に、図表 3-15 の工数分布を 2007 年度の初期値として物理設計全体への複雑度の影響を定量化した結果を図表 3-22 に示す。

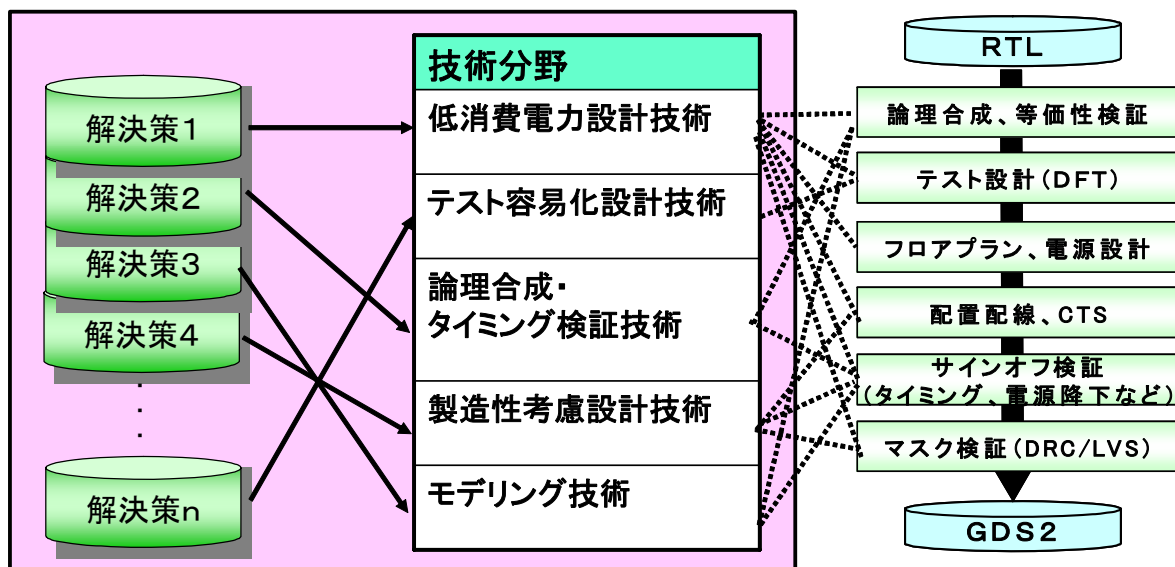
従来の「回路規模増加」という「システム複雑度」のみを考慮したものに、「消費電力増加」、「配線遅延／素子遅延比増加」、「製造ばらつき増加」という 3 つの「シリコン複雑度」を加えた結果、物理設計工程の生産性向上の要求値が、従来モデルに比べて、2015 年で「1.7 倍」、2022 年で「2.9 倍」となり、シリコン複雑度の影響が相当大きいことがわかる。



図表 3-22 物理設計へのシリコン複雑度を加えた設計生産性要求

3-5-2 物理設計の生産性向上を実現するための課題解決策の検討

今回新たに算出したシリコン複雑度を加えた物理設計の設計生産性要求を実現するための課題解決策の検討を行った。2005 年度にまとめた解決策をベースに、物理設計の 6 つのサブ工程と関連付けて、「低消費電力設計技術」「テスト容易化設計技術」「論理合成・タイミング検証技術」「製造性考慮設計技術」「モデリング技術」の 5 つの技術分野で整理を行った。(図表 3-23)



図表 3-23 設計生産性向上のための課題解決策の検討手順

5 つの技術分野毎に、解決策を 2010 年までの中期的なものとしてそれ以降のものに分けて分類してまとめたものを図表 3-24 と 3-25 に示す。

たとえば低消費電力設計技術に関しては、

- ・設計の手戻りをなくすための、設計の上流段階での各種の消費電力の見積もり精度を上げる技術が必要
- ・また、最近出てきたパワー記述の標準化の推進とそれを用いての設計の自動化の加速が必要と考えた。

技術分野	課題解決策	
	～2010	2011～
低消費電力設計技術	・クロックゲーティッドCTS、活性化率を考慮した消費電力見積もり技術 ・マルチVDDを考慮した消費電力見積もり技術 ・RTLレベルでの消費電力見積もりの高精度 ・パワー記述の標準化 ・電源系統考慮の論理合成技術 ・マルチVth/VDD考慮の物理合成技術 ・DVFS(ダイナミックな電圧と周波数のスケール)の自動化技術	・IR drop対応の電源ラインを考慮した見積もり技術 ・電源内蔵Chipによるバイアス可変考慮した消費電力見積もり技術 ・バイアス可変ブロック割り出し自動化技術 ・非同期回路適用の自動化技術 ・小振幅クロック考慮論理合成技術 ・Vth可変制御回路考慮の物理合成技術 ・上流言語(C等)からの多電源動作合成技術
テスト容易化設計技術	・テストバリティ解析技術 ・低故障検出率の論理抽出機能の精度向上 ・高速遅延故障検出ATPG技術 ・故障パターン増加を抑える高速圧縮技術	・高効率テストポイント挿入技術の向上 ・低故障検出率論理のRTL自動修正技術 ・上位言語→RTL時の低検出率回路回避技術 ・故障検出向上回路の自動生成技術 ・故障検出向上回路と論理合成/フロアプランとの連携

図表 3-24 物理設計の設計生産性向上の解決策(1)

技術分類	課題解決策	
	～2010	2011～
論理合成・ タイミング検証 技術	・配置考慮の論理合成技術 ・論理合成と物理合成の融合技術 ・CTS考慮の合成技術 ・クロックメッシュ対応技術 ・マルチコーナマルチモードのタイミング同時最適化技術 ・統計的タイミング解析技術	・仕様からの合成制約自動生成技術 ・制約条件の等価検証技術 ・複数電圧間におけるタイミング自動最適化技術 ・可変VDD/Vthにおけるクロック自動最適化技術 ・高位言語での動作合成からの配置合成技術 ・小振幅クロック考慮技術 ・非同期設計考慮技術
製造性考慮設計 技術	・容量セル挿入最適化技術 ・パーティクル/リソ/CMP考慮最適化技術 ・シグナルEM対応クロック配線技術 ・電源、信号EM解析・検証技術 ・マルチモードIRdrop解析技術 ・歩留り考慮ライブラリキャラクタライズ技術	・IR drop考慮の電源配線自動最適化技術 ・シグナルEM対応バス配置配線技術 ・ダイナミックIRdrop解析技術 ・歩留まり考慮論理合成技術 ・製造インタフェイス考慮技術
モデリング技 術	・電源記述仕様の標準化モデル ・マルチVDD/Vth/Ta及び可変電圧考慮タイミングモデル ・状態依存、ばらつき考慮消費電力ライブラリ生成技術	・インスタンス毎の電源認識対応のモデル ・可変VDD/Vth及びばらつき考慮のタイミングモデル適用 ・小振幅クロック対応のタイミング、消費電力モデル

図表 3-25 物理設計の設計生産性向上の解決策(2)

3-6 国内活動のまとめ

論理検証と物理設計の 2 つのサブワーキングで活動を行った結果をまとめる。(図表 3-26)

2007年度国内活動のまとめ

論理検証SWG活動

- ・重要課題を抽出し、中期的な技術要求と解決策を詳細検討
- ・検証戦略ガイド整備、仕様記述標準化、検証EDA技術向上が必要
- ・上記解決策の2010年時点の設計生産性向上率を定量化

2.3倍の生産性向上の見込み > ITRS要求(1.96)

物理設計SWG活動

- ・シリコン複雑度の影響を加えた設計生産性要求を定量化
 - Near Term(2015年)で**1.7倍**(従来比)
 - Long Term(2022年)で**2.9倍**(従来比)
- ・設計生産性向上のための課題解決策を提示
- ・高精度な見積もり/解析技術と各種最適化の自動化技術が必要

28

図表 3-26 国内活動のまとめ

論理検証サブワーキングとしては、

- ・論理設計検証の重要課題を抽出し、中期的な技術要求と解決策を詳細化してまとめた。
- ・生産性向上の解決策としては、「検証戦略ガイドの整備」「仕様基準の標準化の推進」「検証 EDA 技術の向

上」などが重要と提示することができた。

・これらの解決策が実現された際の 2010 年時点の生産性向上率を見積もったところ、「2.3 倍」となり、ITRS での要求値の「1.96 倍」を上回ることができるということを示すことができた。

次に、物理設計サブワーキングとしては、

・従来の回路規模増で表現したシステム複雑度に、新たにシリコン複雑度の影響を加えた設計生産性の要求値を定量化した。その結果、従来に比べて Near Term の 2015 年時点で「1.7 倍」、Long Term の 2022 年で「2.9 倍」の生産性の向上が必要という結果になった。

・その新たな設計生産性の向上の要求値に対する解決策をまとめた。結果として、消費電力やタイミングの高精度な見積もり・解析技術と、その見積もり値を実現し、更に最適化をおこなうための設計自動化技術の実用化加速が重要と提示することができた。

3-7 今後の予定

2005 年度の SOC 設計技術ロードマップの改訂に対して、今年度は設計生産性の観点から、ロードマップの詳細化と定量化に、論理設計検証と物理設計の 2 つの工程に分けて、それぞれ取り組んだ。

定量化という観点では、論理設計検証に関しては、2010 年時点の解決策が実現された際の生産性向上率を初めて定量化するという成果を上げることができた。また、物理設計に関しては、ITRS でも重要課題と記載されている、システム複雑度とシリコン複雑度の両方を加味した設計生産性の要求値を定量化することができた。今後も定期的に、それらの数値を見直していきたいと考える。また、今回の国内活動の成果を適宜、ITRS の改訂にフィードバックしていきたいと考える。

いずれにしても、半導体プロセスの微細化に伴い、SOC 設計の困難度は大幅に上昇しており、その生産性の向上は半導体事業の成否を左右する程、重要になってきている。そういう点でも、SOC 設計技術ロードマップの重要性は増していくと考えられ、今後もその改定を通じて、技術動向や技術要求を発信していきたいと考える。