

論理合成システムを用いた
シミュレーションエンジン用LSIの設計手法

稲田俊夫 今村真人 北村幸太
浜崎良二 白木昇 平川和之

沖電気工業株式会社 超LSI開発センタ

あらまし 本報告では、論理合成技術を導入した短TATなLSI設計手法を提案する。本設計手法は、特に小量生産型大規模カスタムLSIの設計に有用である。

近年のLSIの高集積化、大規模化はLSI設計期間の増大を引き起こしている。我々は、このことに対処するため、LSI設計手法の検討とその際に必要なCADシステムの開発を行なっている。その第一段階として、現在開発中の論理シミュレーションエンジン用の大規模カスタムLSIの開発に論理合成システムを適用した結果、論理設計段階以降の設計TATを大幅に削減した。本設計手法の特徴は、①LSIチップ全体を論理合成の対象とした、②機能検証に用いたテストデータを後の論理検証、レイアウト検証の際に利用できるようにしたことである。これにより、機能設計に注力可能な設計手法を実現した。

An LSI Design Method Applying Logic Synthesis System
- Development of Logic Simulation Engine Processors

Toshio INADA Masato IMAMURA Kohta KITAMURA
Ryoji HAMAZAKI Noboru SHIRAKI Kazuyuki HIRAKAWA

VLSI Research & Development Center,
OKI Electric Industry Co., Ltd.

Abstract This paper presents a QTAT LSI design method applying logic synthesis system in developing logic simulation engine processors. This method is suited for very large scale ASIC design.

With semiconductor circuit density increasing, LSI design become very complex and time consuming task. Solving this problem needs high level design method and CAD system to support it. We intend to establish an LSI design method using logic synthesis technology.

As developing logic simulation engine processor LSIs, we applied logic synthesis system to entire chip design. The results showed good efficiency of this design method. LSI designer can focus on function level design. Logic level design TAT is decreased to one eighth compared with a design method without logic synthesis technology.

1. はじめに

近年の半導体微細加工技術の進歩に伴い、LSIの高集積化、大規模化が急速に進んでおり、数百Kトランジスタを搭載したLSIが出現している。しかし一方で、LSIの高集積化、大規模化はデータ量の増大をもたらし、その設計作業を複雑化している。さらに、ASIC：小量生産型カスタムLSIの需要の増加と開発コスト低減化、開発期間短縮化の要求により、設計における短TAT化の必要性は強くなる一方である。

この問題を解消するためには、大規模カスタムLSIを短TATで設計可能とするLSICADシステムの開発、および設計手法の確立が不可欠である。

LSICADシステムについては、

- ① 抽象度の高いレベルでの設計を可能とするCADシステム
- ② 大量データを高速に処理するCADシステム

が必要である。我々は、①の要求に応えるため、レジスタ転送レベルからの設計を可能とする論理合成システムCLS^[1-3]を開発済みである。CLSは、LSIの高性能化、高集積化を目的に開発されたモジュールジェネレータベースのレイアウトシステムにインタフェース可能なシステムである。さらに、②の要求に応えるため、論理シミュレーションエンジンの開発を行なっている。

一方、設計手法については現在検討段階であり、論理合成技術の導入を目的に上述した論理シミュレーションエンジン用大規模カスタムLSIの設計を通じてCLSの実用化を試行している。

本報告では、論理シミュレーションエンジン用に設計したLSIの内、Event Management Unit (EMU)、Event Procedure Unit (EPU)と呼ぶ2種類のLSIの設計例を示し、その際に適用したCLSを用いた短TATな設計手法の有用性について述べる。以下、2章では本設計手法の概略を示す。次に3章で現在開発中の論理シミュレーションエンジンの構成とEMU、EPUの位置づけを示す。4章ではEM

U、EPUの設計例をもとに、設計結果の諸元と本設計手法の有用性について述べる。最後に5章で今後の課題について触れる。

2. 設計手法

本章では、論理シミュレーションエンジン用LSIの開発に適用した設計手法について述べる。

最初に本設計手法を確立した背景について述べ、次に設計フローの説明を行なう。

2.1 背景

2.1.1 LSI設計の現状

近年のLSIの高集積化、大規模化はCPU等のシステムを1チップ化することを可能としたが、一方で、設計データ量の増大をまねき、LSI設計作業を複雑化し、人手での論理設計を困難にしている。また設計途中でのLSIの仕様変更は、小規模な回路で実現していたものに比べ、その及ぼす影響範囲が広がり、論理・レイアウトの再設計等莫大な時間と労力を必要とするなどの弊害をもたらした。

さらに、小量生産型カスタムLSIの需要が急増したため、チップ単価に含まれる設計コストの割合が増大し、かつ需要に対するLSI設計の技術者不足をまねく結果となってきている。

これらの問題を解決するためには、より上流工程からのCAD化が必要不可欠となっている。CAD化技術は、膨大なデータを扱う下流工程から実用化が進み、レイアウト設計以降に関しては、その効果が安定してきている。一方、より上流工程に位置する機能・論理設計に関しては、検証技術を中心にCAD化が進み、現在では、論理合成技術を用い論理回路を自動生成する商用システムが出現するまでになっている。

2.1.2 設計手法の確立

LSI設計の現状に対応するため、より抽象度の高いレベルからLSI化できる設計手法を確立する必要がある。

論理設計段階からのCAD化を目的に、論理合成技術の研究が行なわれている。論理合成技術は、設計言語から、その記述動作と同一の論理回路を生成する技術であり、設計言語の仕様記述のレベルにより、サポート範囲は異なる。そこで我々は、設計者の設計意図を容易に記述でき、かつそれらを反映した論理回路を得ることが必要と考え、レジスタ転送レベルの設計言語を入力する論理合成システムCLSを開発した。

CLSの適用により、機能設計結果を設計言語により明確化するとともに、論理回路作成負担をなくすことが可能となった。また設計言語からレイアウト生成システムへのインタフェースが可能となり、LSI設計特有の回路設計技術から設計者を解放し、機能設計段階からのLSI化を実現する設計手法を確立した。

2.2 設計フロー

現在開発中の論理シミュレーションエンジン用大規模カスタムLSIの開発に用いた設計手法の概略設計フローを、図1に示す。

本設計手法の特徴は、

- ・ 機能設計に、記述が容易なレジスタ転送レベルの機能記述言語HSL-FX^[4]を用い、論理合成システムCLSとのインタフェースを図る
- ・ CLSをLSIチップ全体に適用し、機能設計からレイアウト設計に至る工程の自動化を図り、回路設計の専門技術を持たない設計者でも、LSI設計を可能とする
- ・ 機能検証で用いたテストデータを以下の設計工程へもインタフェース可能としたことにより、テストデータ作成負担の軽減を図る

等である。

以下に、設計フローの工程毎の概略について述べる。

2.2.1 機能設計／機能検証

機能設計では、LSIの仕様をもとにLSIのブロック分割や、クロックを基準としたデー

タの参照・演算・転送タイミング等の設計をHSL-FXを用いて行ない、それを検証するためのテストデータを作成する。

次に、作成したHSL-FX記述とテストデータを機能シミュレータの入力とし、HSL-FX記述が設計者の意図どりの動作表現になっているかを確認する。

なお、HSL-FXは、大規模なLSIを効率良く設計できる階層記述やオートマツン記述が可能な言語仕様を持つ、LSI設計用に開発された設計言語である。

2.2.2 論理合成CLS／論理検証

CLSは機能検証済みのHSL-FX記述を入力し、レイアウト生成システムへ論理ブロック割付済みのネットリストを出力する。

この処理過程でHSL-FX記述から、同一タイミングで動作する資源を併合してデータバス部を抽出した後、FSM(有限状態機械)部とそれ以外のランダム回路部との分類処理を行

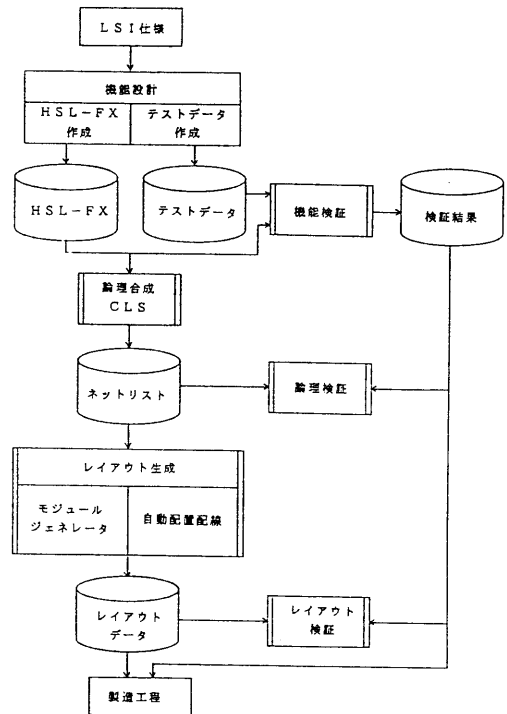


図1. 概略設計フロー

ない、各部毎に面積、遅延等を考慮した論理割付・最適化を行なう。この分類処理は、HSL-FX記述に表現されているデータの流れや制御単位等の設計者の意図を、合成結果に反映することを目的としている。

CLSが出力したネットリストと機能検証時の実行結果をもとに、論理シミュレータを用いて論理検証を行なう。ここでは、論理ブロックの遅延を考慮した検証を行ない、大まかなクロックサイクルの見積もりや、合成された論理回路が適切であるか確認を行なう。

2.2.3 レイアウト生成/レイアウト検証

レイアウト生成では、CLSが出力したネットリストをもとに論理ブロックの自動配置配線を行ない、LSIのレイアウト結果を得る。本設計では、レイアウト生成システムとしてモジュールジェネレータとスタンダードセル方式の自動配置配線システムを用いた。

CLSの処理過程で分類されたデータバス部およびFSM部は、あらかじめモジュールジェネレータによりレイアウトを生成して、自動配置配線システムの中でモジュールとして扱い、全体のレイアウトを生成する。このことにより、人手による信号の流れやタイミングを考慮した配置や機能のかたまり毎の階層化レイアウトに近づけることができる。

レイアウト検証では、生成されたレイアウトからトランジスタや配線容量等のパラメータを抽出しポストレイアウトシミュレーションを行なう。ここで、ハザードの状態やクリティカルバスの検出、内部に搭載するメモリのアクセス動作の確認等のLSIの動作タイミングの最終的な検証を行なう。また、DRCシステムにより、設計基準違反の検出も行なう。

3. 論理シミュレーションエンジンの構成と

EMU, EPUの位置づけ

3.1 論理シミュレーションエンジンの構成

論理検証の高速化を目的に論理シミュレーションエンジンを開発中である。概略構成を図2に示す。本シミュレーションエンジンは、シミュレーション制御部、ネットワーク管理部、シミュレーション実行部より構成されている。

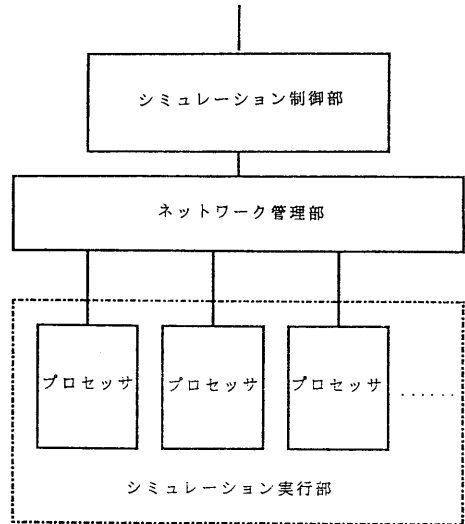


図2. 論理シミュレーションエンジン概略構成

シミュレーション制御部、ネットワーク管理部、シミュレーション実行部より構成されている。

シミュレーション制御部は、シミュレーションエンジンとホストとのデータの授受、シミュレーションエンジンの初期化、およびシミュレーションの開始・実行・終了等の制御を行なう。

ネットワーク管理部は、シミュレーション制御部と後述するシミュレーション実行部、さらにはシミュレーション実行部内のプロセッサ間のデータの授受を管理する。

シミュレーション実行部は、プロセッサ群より構成されている。各プロセッサは、入力されるテストデータをもとにシミュレーションを実行する。

なお、本シミュレーションエンジンの詳細については後日発表予定である。

3.2 EMU, EPUの位置づけ

EMU, EPUは、上述したプロセッサを構成する数10Kトランジスタ規模のデジタルLSIである。プロセッサの高性能化のためには、専用LSIの開発が不可欠であるため両LSIをカスタム化することとした。

図3にプロセッサ内の構成を示す。プロセッサはEMU、EPU、およびメモリで構成されており、タイムマップを用いたイベント駆動方式のシミュレーション^{〔6〕}を実行する。EMUはイベントの登録、取り出し等を中心としたデータ処理を、EPUはイベント評価のための演算処理を行なう。メモリには、EMU、EPUで用いる回路モデル（イベント、回路モデルの接続データ、およびその状態値等）が格納されている。

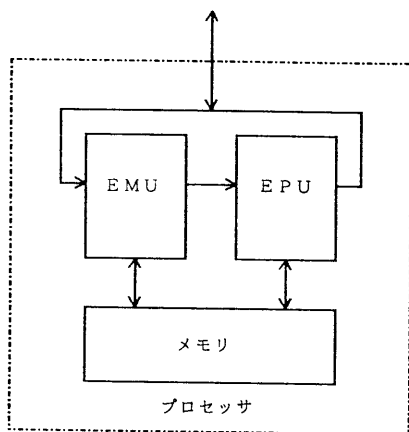


図3. プロセッサ内の構成

4. 設計結果

本章では、EMU、EPUの設計結果の諸元と本設計手法の有用性について述べる。

表1にEMU、EPUの設計結果の諸元を、図4にレイアウト生成結果を示す。

両LSIとも第1回目のレイアウト生成において、その正常動作が検証された。ただし、EPUの動作周波数については目標性能を若干下

回る結果となった。この原因としては、①論理ブロック、特に加算器の性能不足、②レイアウト生成システムの配置・配線処理機能の不備等が挙げられる。これらのことは、論理合成システム単独では対処できないため、レイアウト生成システムと組み合わせて解決を図る必要がある。また、チップ面積においては両LSIともコア部におけるデッドスペースが少なく、チップ面積は入出力ポートで制限されている。このため、チップ全体に対する論理合成システム適用による面積増大の弊害はほとんど認められなかった。さらに、HSL-FX記述量と生成されたトランジスタ数の関係は約30~100トランジスタ/行であり、その記述効率の良さも示された。

次に設計工数について述べる。

図5にEMU、EPUの設計工数比を各設計段階に分けて示す。図は、全LSI設計工数に占める論理設計段階以降の工数比が20%以下であることを示している。図より、本設計手法は設計者が機能設計に注力可能な設計手法であるといえる。また、機能設計・検証工数の大半はLSIの仕様変更のために生じたものであり、その都度修正を行なったHSL-FX記述の機能検証用テストデータの作成に費やされた。今後、テストデータ作成工数の低減化を行なう必要がある。

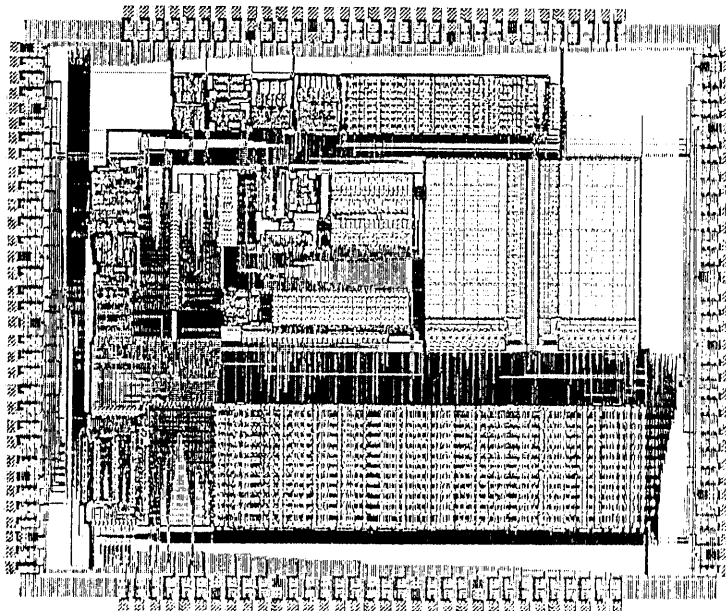
図6には、本手法を適用した20Kトランジスタ規模のEPU下位ブロックの設計工数と、同等規模の人手設計LSIの機能・論理設計段階の工数比較を示す。ただし、人手設計LSIは、論理回路図入力のポリセル型スタンダードセル方式のレイアウト生成システムを用いた場合

表1. EMU・EPU設計結果の緒元

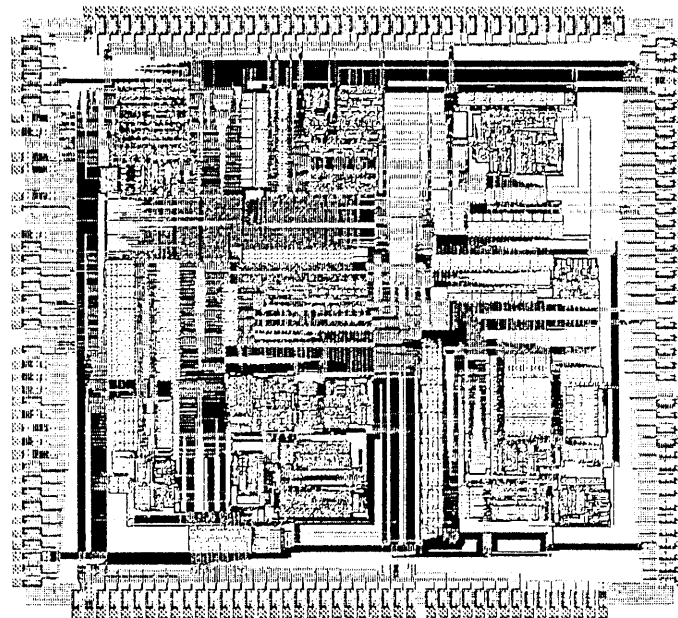
	EMU	EPU
HSL-FX記述量	600行	2700行
動作周波数 ¹⁾	40MHz	30MHz
プロセス	1.2μm CMOS	1.2μm CMOS
トランジスタ数	73K ²⁾	54K
チップサイズ	8.49mm×7.13mm	9.98mm×9.03mm
パッケージ	132Pin PGA	176Pin PGA
消費電力	1.9W	2.0W

1) 目標動作周波数40MHz

2) 6KビットRAMを含む



(a) EMU



(b) EPU

図4. レイアウト生成結果

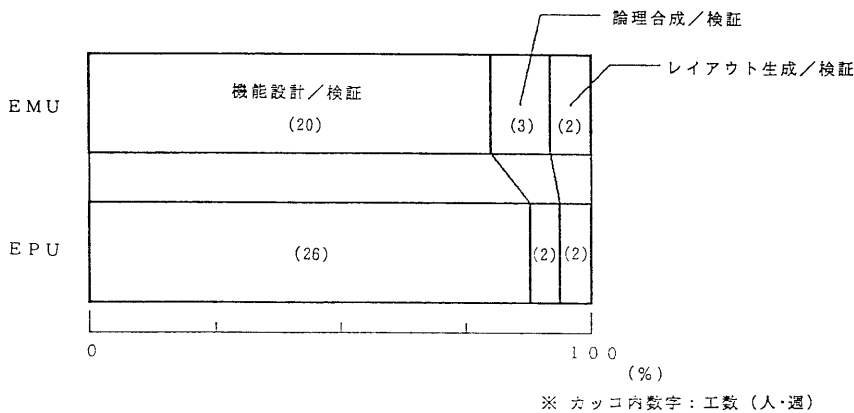


図5. EMU・EPU設計段階別工数比

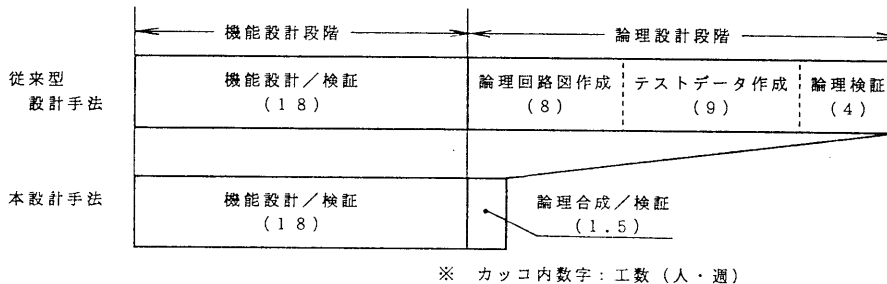


図6. 従来型設計手法と本設計手法の工数比較（20Kトランジスタ規模のLSI設計例）

の設計例で、どちらもLSIの仕様変更はないものとした。本設計手法は、従来型の設計手法に比べ論理設計段階の工数（論理検証用テストデータ作成工数を除く）を約1/8に短縮しており、その有用性が明らかとなった。

5. おわりに

本報告では、論理合成システムCLSを用いた短TATなLSI設計手法について、論理シミュレーションエンジン用LSIの設計例をもとに、その有用性を述べた。本設計手法の適用により論理設計段階以降の設計工数の大幅な削減が実現でき、設計者を機能設計段階に注力可能とした。このことにより、LSI設計者不足の解消についての見通しも得た。

今後の課題として、機能設計段階の工数削減、特にテストデータ作成負担の低減化を図ることが必要であると思われる。そのため、LSIのHSL-FX記述からの機能検証用テストデータ生成の検討を行なっている。また、論理合成技術は今後のLSI設計において重要性が増大することは確実であるが、LSIの高性能化、高信頼性化のためには下流工程との有機的つながりが不可欠である。そのためには、論理合成システムとレイアウトシステム等のより下流設計段階のCADシステム、さらには故障検出用テストデータ自動生成システム等の製造工程に関連したCADシステムとの効率的な結合を図っていくことが必要であると考えている。

最後に、本研究は基盤技術研究促進センターの融資により行なわれたものであり、関係された方々からの種々の御助言に対し紙面を借りて感謝いたします。

【参考文献】

- [1] 今村他：論理合成エキスパートシステム
CLS = システム構成と入力データ処理部
について =, 情報処理学会第36回全国大
会予稿集, 1988.
- [2] 北村他：論理合成エキスパートシステム
CLS = 合成過程 =, 情報処理学会第36
回全国大会予稿集, 1988.
- [3] 浜崎他：論理合成エキスパートシステム
CLS について, 情報処理学会設計自動化
研究会資料, 43-1, 1988.
- [4] NTT厚木電気通信研究所集積回路応用研
究室：拡張階層仕様記述言語HSL-FX
仕様書, 1986.
- [5] 樹下他：VLSIの設計II, 岩波講座マイ
クロエレクトロニクス4, 岩波書店