

64M DRAMの自動設計手法

鶴岡義文* 奥秋勝己* 松原靖* 奥田高** 塚田浩二郎*** 関根順一**** 安田由美子****

*NEC LSI事業本部メモリ事業部

**NEC ULSIデバイス開発研究所

***NEC LSI事業本部ULSIシステム開発研究所

****NECアイシーマイコンシステム(株)

*〒229 神奈川県相模原市南橋本3-1-35 NEC橋本技術センター

電話 (0427) 72-1122

Fax. (0427) 72-7733

あらまし

今回開発した64MDRAMは0.35 μ m、CMOS2層アルミプロセスを使用し、10.1 \times 20.97mm²のチップ上に約1億5千万個の素子を集積し、電源電圧3.3Vにおいて読み出し時間50ナノ秒の高性能を実現した。

本DRAM設計にあたっては回路設計、回路シミュレーション、レイアウト設計、レイアウト検証に従来手法を生かした、メモリ設計に最適なCAD技術を導入し、設計のTATを大幅に短縮すると共に機能、動作速度等の目標性能を達成することが出来た。

和文キーワード DRAM、CMOS、CAD、回路シミュレーション、自動レイアウト

A 64Mega Bit Dynamic RAM and it's design techniques.

Yoshitake Tsuruoka

LSI MEMORY DIVISION

LSI OPERATIONS UNIT

NEC Corporation

1-35,MINAMI-HASHIMOTO,3-CHOUME

SAGAMIHARA,KANAGAWA 229,JAPAN

TEL (0427)72-1122

FACSIMILE (0427)72-7733

Abstract

An advanced 64Mega Bit DRAM and it's design technique are described in this paper. The 64MDRAM is fabricated by using 0.35 μ m CMOS double metal layer process technology to integrate 150,000K devices on a 10.1 \times 20.97mm² die. In design of the 64MDRAM, we used design automation techniques. The 64MDRAM was analyzed by circuit simulation and layout verification before fabrication. Due to the application of the design automation we achieved target access time and all functions in minimized design term.

英文 key words DRAM,CMOS,CAD,circuit simulation,automatic layout

1. はじめに

メモリICの集積度向上は近年も衰えることを知らず、特にダイナミックRAMの記憶容量推移は現在でも3~4年で4倍というペースを継続している。ダウンサイジングによるワークステーションの高性能化、あるいはパーソナルコンピュータ上のWindowsの普及等により大容量メモリに対する市場要求は今後とも増大していくと予想される。

またノートパソコンに代表されるパーソナル向け電子機器の発達に伴い低消費電力、低動作電圧がメモリICに対しても強く要求されるようになってきている。

一方従来のDRAMでは出力ビット幅は1bitあるいは4bitで構成されるものが主流であったが、大容量化にとめない、出力を多ビット化する方向が求められている。このため各種ビット幅および各種のアクセス方式への対応を1チップで実現するための仕組をチップ上に組み込むことが必要になってきている。

このように高機能・高性能なチップを実現するため、設計は益々複雑になっているにもかかわらず、従来と同等もしくはそれ以下の時間で設計することが求められている。

従来の全てを人手に頼る設計の継続では膨大な設計工数が必要になる。また一般に自動設計をメモリに適用すると、大量生産に耐え得るチップサイズと要求される高性能を実現することは困難であると考えられてきた。今回我々は64MDRAMの設計に当りこのような要求を満

たす自動設計手法の実現を目標とした。

2. 64MDRAMの概要

今回開発した64MDRAMは $0.35\mu\text{m}$ 、CMOS2層アルミプロセスを使用し、 $1.0 \times 20.97\text{mm}^2$ のチップ上に約1億5千万個の素子を集積し、電源電圧3.3Vで読み出し時間50ナノ秒、の高性能を実現した。(写真1)

本64MDRAMは低電圧動作と高速動作を実現するためにトリプルウェル構造および新たな昇圧回路を採用した。

またLOC(lead on chip)技術を採用し、ファーストページ、スタティックカラム、ライトパービットなどの機能切り替えおよび $\times 4$ 、 $\times 8$ 、 $\times 16$ のビット幅選択が同一チップで出来るよう設計されている。このため機能評価、信頼性評価を効率的に行うことが可能になった。

3. 設計フローの概要

64MDRAMの設計、検証フローを(図1)に示す。

メモリの設計においては回路設計とレイアウト設計はほぼ同時期に開始され、互いに設計のデータを交換しながら、平行に進行される。

通常回路設計は論理設計→論理検証→アナログ設計→SPICEシミュレーションの順序で進行する。それぞれの工程において、その時点の設計精度に応じた回路図・ネットリスト等がレイアウト設計担当者に提示される。

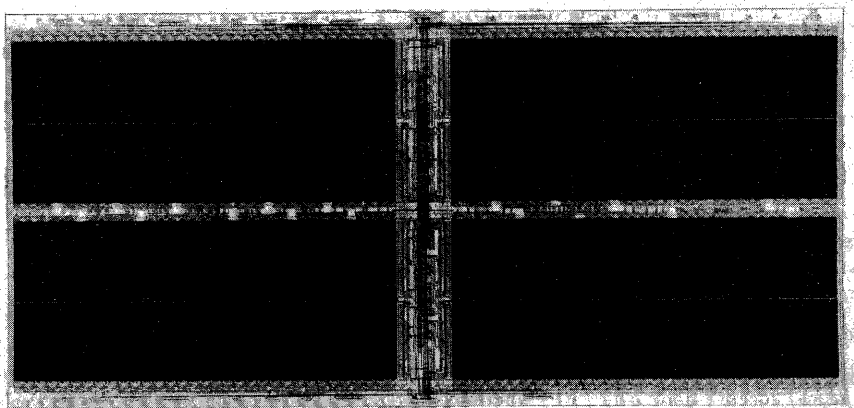


写真 1

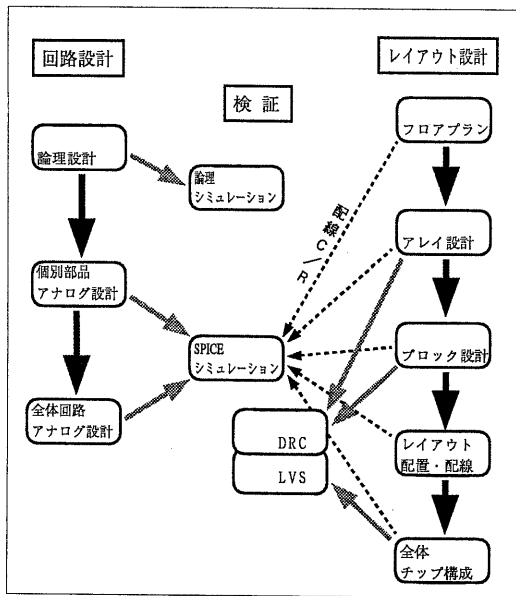


図1 メモリ設計のフロー

レイアウト設計は通常フロアプラン→アレイ設計→ブロック設計→周辺配置配線→全体チップ構成という順序で進行する。レイアウト設計側からは各信号の配線容量、配線抵抗等の配線負荷が回路設計側に提示される。

設計工程が進む毎に、これらの設計データは互いにフィードバックされ、回路・レイアウトの最適化が行なわれる。

4. 回路設計の自動化

4.1 回路設計

EWS上の回路図エディタの用いて回路図レベルで入力した。

入力された回路図から出力されるネットリストは

- (1) 論理シミュレーション
- (2) SPICEシミュレーション
- (3) 自動配置配線
- (4) レイアウト検証

に利用されるため、それぞれの目的に応じたさまざまな属性(property)を回路図に与えることにより出力されるネットリストを制御している。特にメモリの回路設計においては次項に述べるようにSPICEによるシミュレーションが主体となるが、全回路をSPICEでシミュレーション

をすることは現実的でもないし、設計のTATを考慮すれば不可能である。このためメモリセルに代表される繰り返しの回路のSPICEシミュレーションは畳み込んだ形(縮退と呼ぶ)で行われる。一方、自動レイアウト用にネットリストを出力するため、あるいは最終的な接続検証を行なうための回路図には、全ての繰返しが表示されていなければならない。この繰返しを効率的にかつミス無く行うために回路図エディタのframe機能(for表現)を最大限に活用している。

またSPICEシミュレーションでは計算精度を高めるためにレイアウトから読み取られた配線抵抗、配線容量が回路図に付加している。しかし自動配置配線、レイアウト検証用にはこのような容量、抵抗は不要となるため回路図エディタのframe機能(case/if表現)を用いてネットリスト抽出時に削除するようにしている。

4.2 回路シミュレーション

(a) 論理シミュレーション

従来メモリICにおいては回路機能が限定されているため、論理シミュレーションは大きな重要性を持っていなかった。しかし近年のメモリの高機能化、各種ビット幅製品の1チップ化、高度なテスト機能の内蔵化などにより、1チップ上に集積される回路の複雑度は高度化している。このため各種入力パターンに対する正常動作を確認するための論理シミュレーションが不可欠になっている。本設計においてはEWS上の論理シミュレーターを用いて約1000種類のパターン、タイミングの組み合わせでシミュレーションを実行し全ての機能が全ての入力条件において正常動作することを設計段階で確認した。

(b) SPICEシミュレーション

アナログ動作のシミュレーションはメモリにとっては更に重要である。本64MDRAMの設計においては全て社製のSPICEシミュレーターを使って行なった。

SPICEシミュレーションは基本的に

- (1) 個別回路
- (2) 各系回路
- (3) 全体回路

の階層に分けて実施される。

個別回路は各機能ブロックについて動作速度、消費電流などの特性について最高の性能を追求される。個別回路は通常、制御信号単位に

数十から数百のトランジスタにより構成されておりEWS上のSPICEシミュレータにより上記性能が最適化される。

各系回路はDRAMの場合Xアドレス制御系(RAS系)、Yアドレス制御系(CAS系)、等に分割されクリティカルパスの動作速度あるいは各制御系信号のタイミングに関してSPICEにより最適化される。

これらの各系別回路は最終的に全体回路として、レイアウトデータおよびリードフレームから読みとられた電源・GNDのインピーダンスを回路図に付加され、電源ノイズの影響を含めた形でシミュレートされる。

64MDRAMの設計に当たっては、社製並列コンピュータであるCenju/Cenju2をSPICEシミュレーションのマシンに用いることにより従来不可能であった数万デバイス規模のSPICEシミュレーションを可能とした。

5. レイアウト設計手法

レイアウト設計は各部分の性格にあわせ最適な設計手法を採用した。図2にレイアウトデータの階層構造を示す。

5.1 レイアウト分割と設計手法

(a) アレイ部設計

メモリセルおよびそれに付随しアレイ状に構成されるXデコーダ、Yデコーダ、センスアンプ等を含む部分を「アレイ部」と呼ぶ。今回のアレイ部設計に当っては社製のポリゴンレベルエディタを用いてデータ作成した。

このアレイ部はまず最小単位の個別部品の設計から始まり、この部品を最小の繰り返し単位

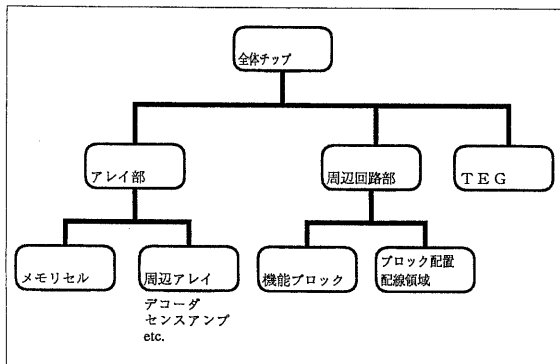


図2 レイアウトデータ構造

で構成した「ミニアレイ」を作る。このミニアレイでデザインルールチェック(DRC)および接続検証(LVS)が実施され、部品内および部品間接続の正当性が検証される。その後ミニアレイを構成する部品は実際の個数分繰返し配置され「本アレイ」と呼ぶマクロブロックとして完成される。今回の64MDRAMでは、16Mビット分の本アレイを作成し、それを4ブロック配置する構成をとった。

(b) 周辺レイアウト設計

周辺レイアウトはブロックベース2層メタル対応の自動配置配線プログラムを用いて、ネットドリブン配線機能によりインタラクティブに配置配線を行った。詳細は後述する。

(c) 全チップ設計

ポリゴンレベルのエディタで作成したアレイ部と、自動配置配線プログラムで作成した周辺レイアウトは最終的にポリゴンエディタを用いて合体され、チップ製造に必要なスクライプライン、特性チェック用TEG等を付加し最終的なEBテープ作成用データに組み上げられる。

5.2 レイアウト自動設計手法

メモリの設計においては通常ゲートアレイ、スタンダードセル等のASICに適應される全自動の配置配線ツールは使えない。これは、

- ・生産コストのミニマム化
- ・最高の動作速度
- ・メモリアレイが大きな面積を占めるため周辺回路領域の形状が限定される

等の理由によりブロック配置位置、配線経路の細かい指定が必要なためである。このため今回の64MDRAMの設計に当って人手設計と比較して遜色のないチップサイズ・特性が実現出来るように、インタラクティブに配置、配線およびコンパクションが行えるツールを選択した。

(a) ブロック配置

ブロックの配置は上記の理由により、人手で決定する事とした。レイアウト上へのブロックの配置は自動配置配線プログラムのマニュアル配置機能を用い、回路図に記述されている回路ブロックを人手により決定されたフロアプランにしたがって配置した。この配置の際に後述のコンパクション機能を有効にするために実際の配置位置より倍程度の配線領域を確保した位置に配置した。

また本来の回路機能と無関係なオンチップT

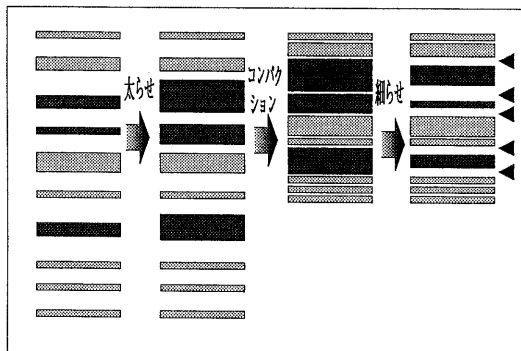


図3 配線のコンパクト化

このため図3に示す手順により指定配線の幅、間隔が所望の寸法になるようにした。

- (1)間隔を広げる配線の太らせ
- (2)コンパクト化実行
- (3)太らせた配線を元の太さに戻す

という処理により図中「◀」で示された部分を所望の間隔にすることが出来る。

この配線の太らせおよび細らせに関してもスクリプトプログラムにより一括実行することでミス無く効率的に行うことが出来た。

6. レイアウト検証手法

レイアウト検証は物理的な寸法チェックを行うDRC (Design Rule Check)、電気的な接続の正当性を検証するERC (Electrical Rule Check)、回路図とレイアウトの一致を検証するLVS (Layout VS. Schematic)に分けられる。

今回の開発に当っては、全て社製ツールを用い全チップにおいてミスがないことを確認した。

6.1 デザインルールチェック (DRC)

レイアウトデータの階層毎にDRCが実施される。したがって上位階層のチェックで引っかかるエラーは基本的に下位階層の境界部にしか存在しない。

特にアレイ部に関しては前述のように個別部品、ミニアレイといった設計階層毎にDRCによる検証を行い最終的には本アレイ全体に対してDRCのウィンドウ機能を用い切り出した形で検証した。

また周辺レイアウトにたいしても同様に階層毎に検証が行なわれる。特に機能ブロックはポリゴンレベルでマニュアル設計されるため慎重

EGやアセンブリ上規定される配置禁止領域はフィードセルとして定義をしたダミーブロックを配置した。

メモリ設計においてはチップサイズの極小化、極限までの動作速度の高速化のため設計変更が繰り返される。その度に人手により再配置を行っているため効率が悪い。そのため今回の設計においては一回配置を行った後の再配置には前回の配置結果のデータベースよりブロックの配置位置、方向を抽出し前回と同じ配置を自動的に実行するプログラムを作成し、再配置時の効率向上をはかった。

(b) 配線

メモリのような複雑なレイアウトには自動配線機能が使えない。そのため基本的に配線処理は自動配置配線プログラムが有する、インタラクティブな配線機能を用いて行った。この方法では全自動で配線するのに比べ効率は低下するがネットドリブンに配線が行えるため、ミスはなくまた後述するコンパクト化機能の活用により、配線自身はラフに引く事が可能になりポリゴンエディタを用いて配線する場合と比較して効率良くデータ作成ができた。

またメモリのレイアウトでは、アレイの繰り返しに対応し周辺部にも多くの繰り返しブロックが存在する。これらの繰り返しブロックへの配線は同じ電気特性が要求される。この繰り返しを最適化するために自動配置配線プログラムが有するコマンド自動実行機能を利用し、繰り返し配線を実行するためのスクリプトプログラムを自動発生するプログラムを作成し効率向上をはかった。

(c) コンパクト化

今回の設計においては基本的にラフなサイズで配置配線を手により短時間に行い、その後コンパクト化によりデザインルールの値まで全体の寸法を縮小する手法をとった。これは今回利用した自動配置配線プログラムのコンパクト化機能が高速で、細かい制御が可能であるためである。

64Mクラスのメモリにおいては配線間容量による信号間のクロストークおよび配線抵抗による信号の減衰が大きな問題になるため配線の幅および間隔は詳細に規定する必要がある。しかし現状のコンパクト化では全ての配線をデザインルールで設定された間隔に一律に詰めるため細かい配線間隔の設定が出来ない。

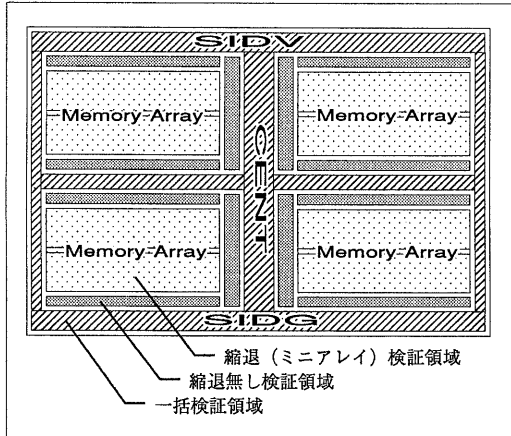


図4 LVS検証の領域分け

な検証が必要である。

6.2 接続検証 (ERC、LVS)

レイアウトの接続検証はERCおよびLVSを併用して行なわれるが、DRAMの設計においては通常のCMOSのルールに対する正当性をチェックするERCは、疑似エラーが多発するため有効ではない。これはメモリの場合高速動作を追求するためにアナログ回路を多用していることが原因である。このためERCは電源線、信号線のショート、オープンをチェックすることを主な目的に行なった。

LVSについてもアナログ回路の検証のため単純な接続検証に加えて

- ・トランジスタのL/W寸法
- ・トランジスタの基板電位
- ・抵抗値/容量値
- ・複数系統の電源、GND配線

を検証する必要がある。またメモリ特有の素子として

- ・リダグダンシ用ヒューズ素子

の検証も必要である。このため検証用の接続情報にはSPICEベースのネットリストを用いた。

メモリの特徴であるメモリセル回りの繰返しブロックの検証は素子数の多さから非常に困難である。今回の設計においては回路図の作成および、レイアウトデータ作成においてそれぞれの工夫によりこの問題点を解決した。

回路図の作成においては、前述の回路図エディタの有する繰返しを表現するためのfor frame機能を最大限に活用した。このためアレイ

部検証用回路図の作成工数、入力ミスを低減することが出来た。

レイアウトデータの作成においては、ワード線・ビット線に代表される繰返し信号の検証時のキーになる"TEXT"データを自動的に付加するプログラムを作成しデータ入力工数、ミスの低減をはかった。このプログラムの特徴はサフィックスの単純なインクリメントだけではなく"0→1→3→2"等の変則的な並びを自由に設定できるようにしたことにある。

64MDRAM規模になるとチップ全体を一括で検証することは困難である。このため今回の設計に当たっては、図4に示す領域にチップを分割

- (1) ミニアレイによる個別部品および部品間
- (2) X/Yデコーダ列等の非繰返し部分全体
- (3) 周辺個別レイアウト
- (4) 周辺全体
- (5) 周辺全体+アレイ部境界部

という手順で検証を行ない、最終的にチップ全体に渡り全ての部分が検証できるようにした。

7. 終りに

今回の64MDRAM設計においてはCADによる設計を本格的に導入することにより、設計精度の向上、設計工数の低減が可能となり、

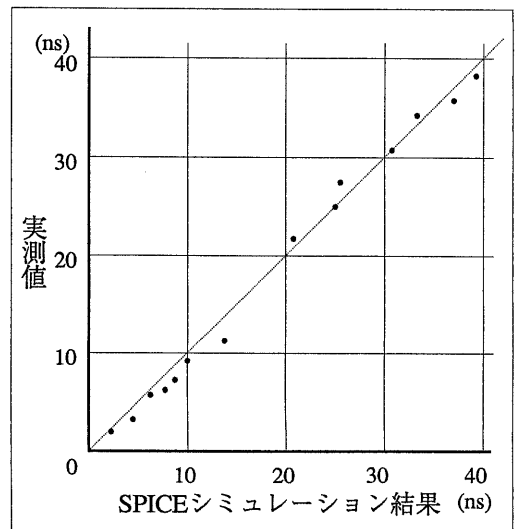


図5 SPICEシミュレーションと実測値の比較

化, 動作電圧 $3.3 \pm 0.3 \text{ V}$ で 50 ns の高速アクセスを実現」、日経マイクロデバイス、1993年4月号

[2] 中田 他「並列回路シミュレーションマシンCenju」、情報処理学会第31巻第5号PP.593-601,1990年

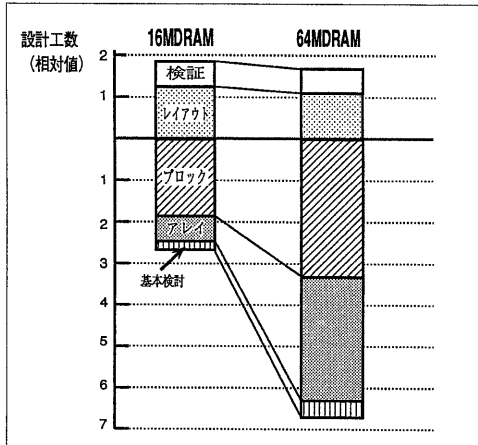


図6 16MDRAMとの工数比較

目標の機能・動作速度を短TATで達成することが出来た。以下に効果の代表例としてSPICEシミュレーションと実測値の比較、および過去の製品とのレイアウト設計工数の比較を示す。

7.1 SPICEシミュレーションの精度

SPICEシミュレーションの計算精度をチェックするため、試作された製品の内部波形と比較した。その結果を図5に示す。最終的に5%以下の誤差に収まっており十分な精度を有している。

7.2 自動レイアウトによる工数削減度

64MDRAMの設計工数と16MDRAMの設計工数比較を図6に示す。

16MDRAMと比べ設計規模は約2倍になっている。このためブロック設計、アレイ設計等従来手法により設計した部分において規模に比例した工数が必要であった。しかし今回の自動化手法を採用した周辺レイアウトの設計工数は16MDRAMより低減することができた。またネットドリブン機能により接続情報の正当性が保たれるため、その後の検証の収束を早くすることができた。

今後更なる工数低減のためには、ブロックおよびアレイ部設計のメモリ設計に適した自動化手法の確立が必要である。

8. 参考文献

[1] 奥田、辻本「64MDRAMを初めて実用