セルベース設計環境を利用した最適設計技術

京都大学大学院情報学研究科
通信情報システム専攻
小野寺秀俊

セルベース設計環境を利用した最適設計技術

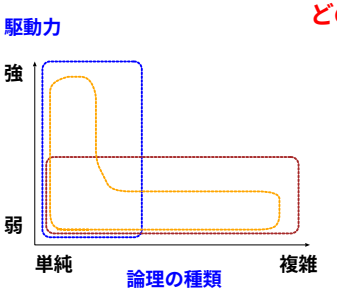
- セルベース設計における設計最適化
  - ライブラリ構成法
  - タイミング解析技術
  - 設計フロー
- セルベース設計環境を利用した最適設計
  - オンデマンドライブラリを用いた最適設計
  - データパス設計

セルベース設計環境を利用した最適設計技術

- セルベース設計における設計最適化
  - ライブラリ構成法
    -- 実験的検討 --
  - タイミング解析技術
  - 設計フロー
- セルベース設計環境を利用した最適設計
  - オンデマンドライブラリを用いた最適設計
  - データパス設計

セルライブラリ構成の実験的検討

どの領域が重要か



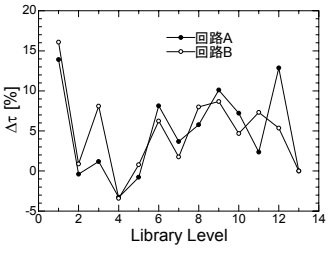
- セル生成の対象を妥当な範囲に制限
  - ライブラリ開発コスト低減
  - 生成ライブラリの品質向上
- 最低限必要かつ妥当なセル構成
  - 論理の種類
  - 駆動力の種類

セル論理の種類

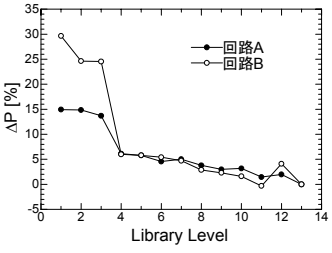
(1) NAND2, NOR2
(2) AND2, OR2
(3) XOR2, XNOR2
(4) AOI21, OAI21
AOI22, OAI22
(5) AO21, OA21
AO21, OA22
(6) NAND3, NOR3
(7) AND3, OR3
(8) AOI211, OAI211
AO211, OA211
(9) NAND4, NOR4
AND4, OR4
(10) XOR3, XNOR3
(11) AO222, OA222～
AO44, OA44
(12) Inverted Input
AOI21, 22
OAI21, 22
NAND2～4
NOR2～4
(13) NAND5, 6, 8
NOR5, 6, 8

集合の番号は以降の図で
Library Levelと呼ぶ。

最大パス遅延と消費電力の変化



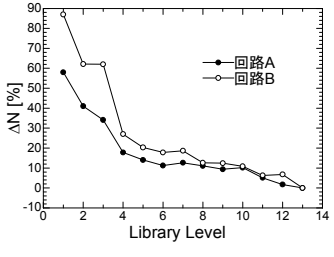
最大パス遅延の変化



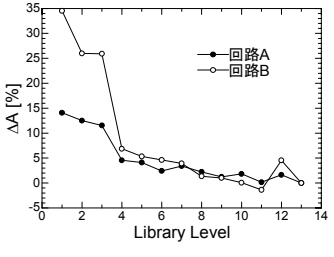
消費電力の変化

回路A:演算回路(12kG)　回路B:8bit CPU(4kG)

ネット数と面積の変化



ネット数の変化

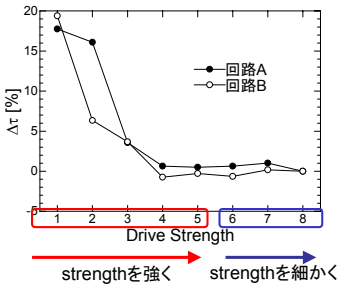


面積の変化

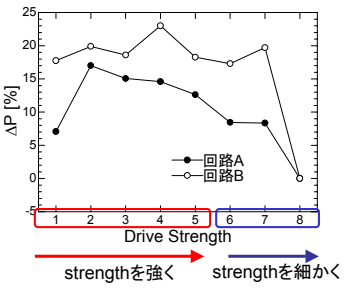
駆動能力の構成

(1) x1
(2) x1, x2, (x4)
(3) x1, x2, x4, (x8)
(4) x1, x2, x4, x8, (x16)
(5) x1, x2, x4, x8, x16, (x32)
(6) x1, x1.5, x2, x3, x4, x8, x16, (x32)
(7) x1, x1.5, x2, x3, x4, x6, x8, x12, x16
(x24, x32)
(8) x0.65, x1, x1.5, x2, x3, x4, x6, x8,
x12, x16, (x24, x32)

最大パス遅延と消費電力の変化



最大パス遅延の変化



消費電力の変化

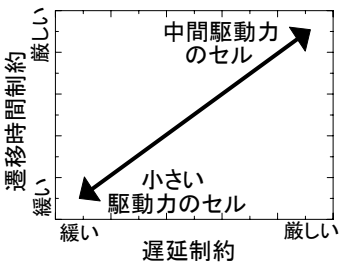
セルライブラリ構成実験のまとめ (1/2)

- 論理の種類(論理の種類は少なくてよい)
  - 基本論理(NAND, NOR, XOR)
    - 簡単な構造のセルで十分(2～3入力)
  - 複合論理(AOI, OAI)
    - 消費電力/ネット数/面積の改善に非常に有効
    - 簡単な構造のセルで十分(3～4入力)
  - 正論理(AND, AO等)
    - 遅延の改善に有効

セルライブラリ構成実験のまとめ (2/2)

- 駆動能力(多様な駆動レベルが望ましい)
  - 高駆動力
    - 遅延の改善に有効
    - 必要な駆動力は負荷の分布に拠る
  - 中間駆動力
  - 低駆動力
    - 消費電力の低減に非常に有効

駆動力の多様化による消費電力削減(実験的検討)



設計制約(遅延、遷移時間)によって消費電力削減に効果的なセルの種類は異なる。

Lib.0: ×1, ×2, ×4

Lib.2: ×0.65, ×1, ×1.3, ×2, ×2.6, ×4

15 – 20 % の消費電力削減効果

セルベース設計環境を利用した最適設計技術

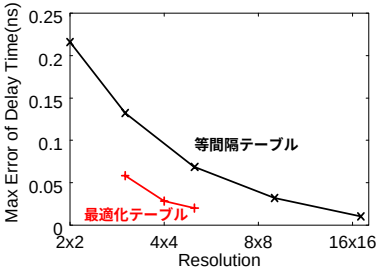
- セルベース設計における設計最適化
  - ライブラリ構成法
  - タイミング解析技術
    Static Timing Analysis
  - 設計フロー
- セルベース設計環境を利用した最適設計
  - オンデマンドライブラリを用いた最適設計
  - データパス設計

タイミング解析技術(STA)

- 遅延テーブル作成時の検討課題
  - テーブル分割の最適化
  - 入力波形の選択 ---波形の多様性による誤差
- STA実施時の検討課題
  - 信号波形歪みへの対応 ---等価入力波形の伝播
- Statistical STA
  - ゲート遅延の局所的なばらつきへの対応

遅延テーブルの分割最適化

0.6μm INV1 Cl: 0.02 - 2.00 pF tr: 0.02 - 2.00 ns

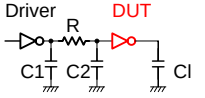


分割最適化により補間誤差十分小さくできる

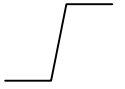
波形近似による誤差の方が大きい

入力波形の選択

実際の回路



直線



直線+ 指数関数
(Chang, DAC88)

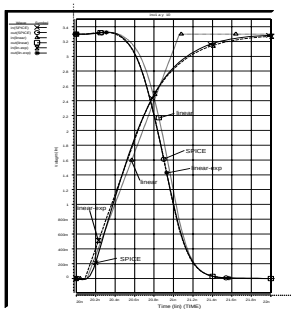
Driver, C1, R, C2 の組合せにより遅延異なる

種々の波形に対する遅延時間の評価誤差

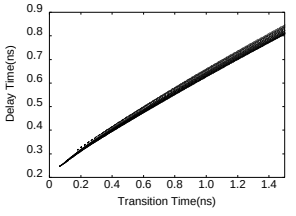
	平均誤差	最大誤差
直線	3.0 %	8.6 %
直線+指数	1.4 %	2.8 %

入力波形の選択

入力波形近似の比較

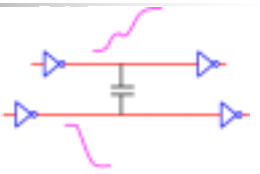


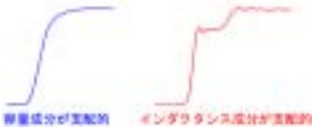
波形の多様さによる遅延時間の不確かさ

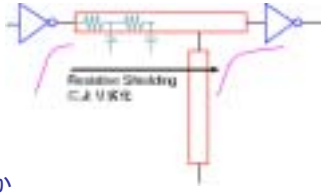


STAの課題: 信号波形の歪

- クロストークノイズ
- 抵抗性容量遮蔽効果
- インダクタンス



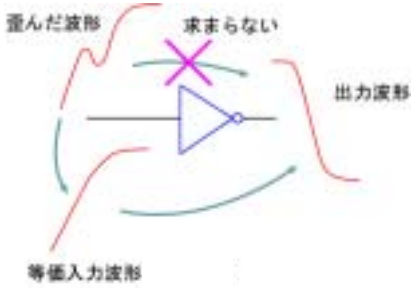




歪んだ波形をどのように扱うか

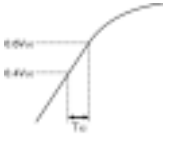
等価入力波形

- 正しい出力波形が得られる、等価ゲート入力波形を求める

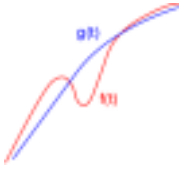


重み付き最小二乗法による合わせ込み

- 波形全体を合わせ込む
  - ゲートに入力される関数 f(t)
  - 合わせ込む関数 g(t): [Chang, DAC88]
  - 積分範囲[t_1 t_2] : 入力が遷移をはじめてから、入力もしくは出力の遷移が終わるまで

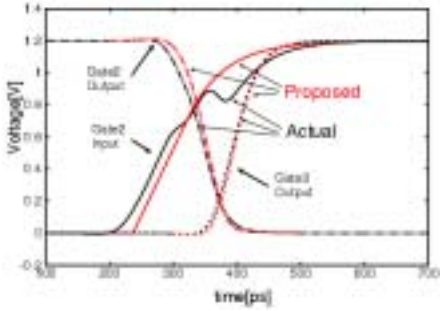


$$\int_{t_1}^{t_2} \left| \frac{dVout}{dVin} \right| \{f(t)-g(t)\}^2 \, dt$$



等価入力波形の例
（クロストーク、隣接配線2本）

- 隣接配線2本の場合の入出力波形

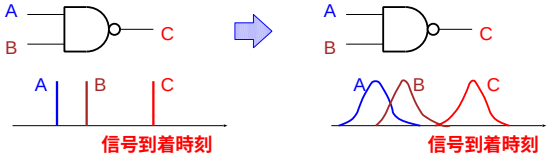


ゲート遅延の局所的ばらつき

大域的ばらつき:　同一の変動変数を全素子に適用

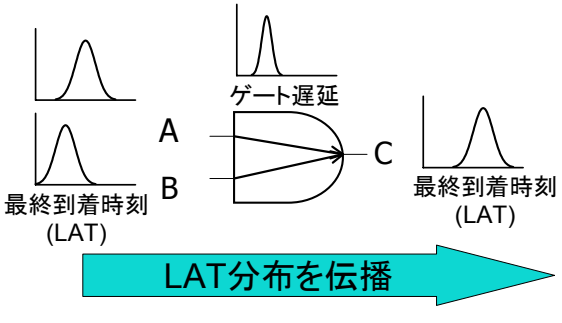
局所的ばらつき:　素子毎に変動変数異なる
特性のランダムな変動　ホワイトノイズ

(狭義)　Vth, C, R, etc. の局所的ばらつき
(広義)　波形の不確かさ、C/Rの見積り誤差、局所的なノイズ



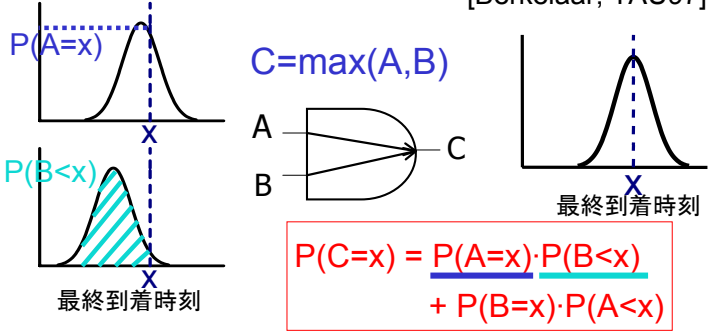
Statistical Static Timing Analysis
静的統計遅延解析

素子遅延(ゲート,配線)
信号の最終到着時刻(LAT)　は分布を持つ



最終到着時刻(LAT)分布の伝播

[Berkelaar, TAU97]



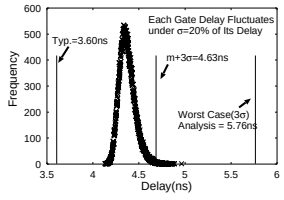
$$P(C=x) = P(A=x)\cdot P(B<x) + P(B=x)\cdot P(A<x)$$

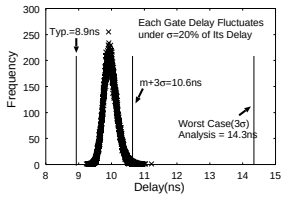
最終到着時刻分布を正規分布で近似

統計的な遅延解析

モンテカルロ解析　　Statistical Static Timing Analysis

解析例　　**各ゲートの遅延が、σ = 20% でばらつくとした**





Skew corner(**各ゲートの遅延を 3σ の値で定義**)**解析は悲観的すぎる**

⇨ Over Design　　**正しい 3σ の遅延値が必要**

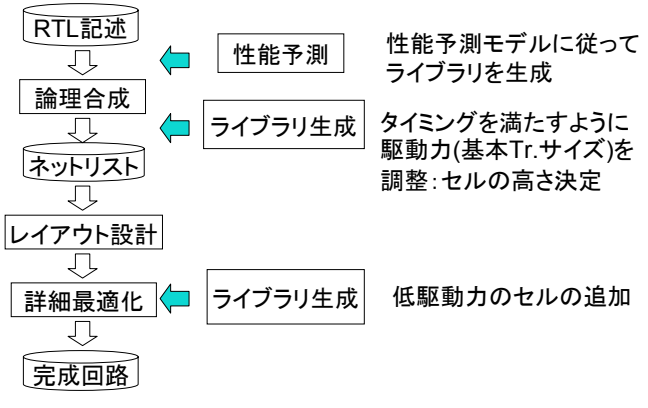
Statistical STA

- 素子毎のランダムなばらつきや、設計量(遅延など)の不確かさの影響を統計的に取り扱う
- ばらつきにより各信号の最終到着時刻が分布を持つ(正規分布に近似)
- この分布を、入力から出力に伝播させ、出力信号最終到着時刻の分布を求める
- 通常のSTAと同様の処理

セルベース設計環境を利用した最適設計技術

- セルベース設計における設計最適化
  - ライブラリ構成法
  - タイミング解析技術
  - 設計フロー
    実験的検討
- セルベース設計環境を利用した最適設計
  - オンデマンドライブラリを用いた最適設計
  - データパス設計

性能最適化設計フロー

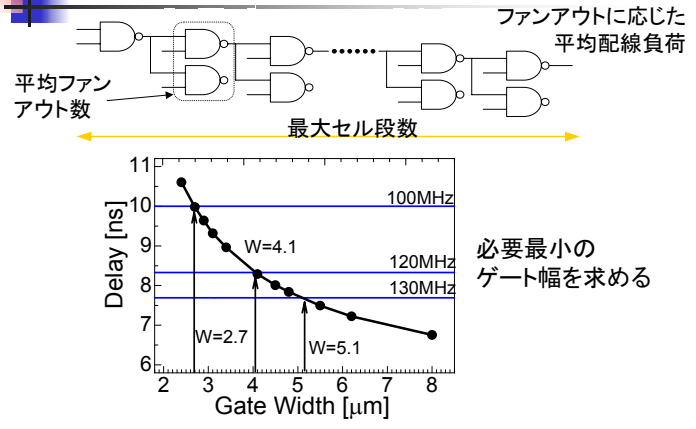


性能最適化設計フロー

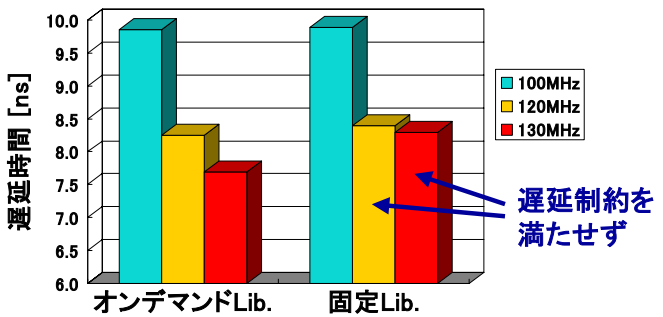
32-bit RISCコア設計実験

- 設計回路
  32-bit CPU　回路規模 9kセル、面積の約40%がFF
- 設計仕様
  100MHz, 120MHz, 130MHz
- 設計に用いるライブラリ
  - オンデマンドライブラリ　セル高さ(基本Tr.サイズ)可変
  - 固定ライブラリ　　基本Tr.サイズ W=4.45μm

性能予測



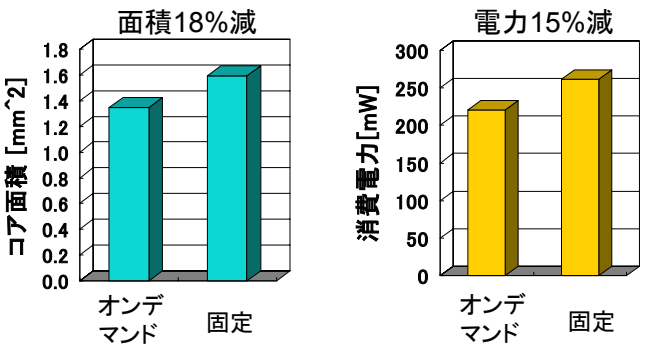
設計回路の評価(遅延)



オンデマンドLib.ではすべての遅延制約で設計可能

設計回路の評価(面積、電力)

設計仕様100MHzの場合



オンデマンドライブラリによる設計結果

(a) 100 MHz 9-pitch cell

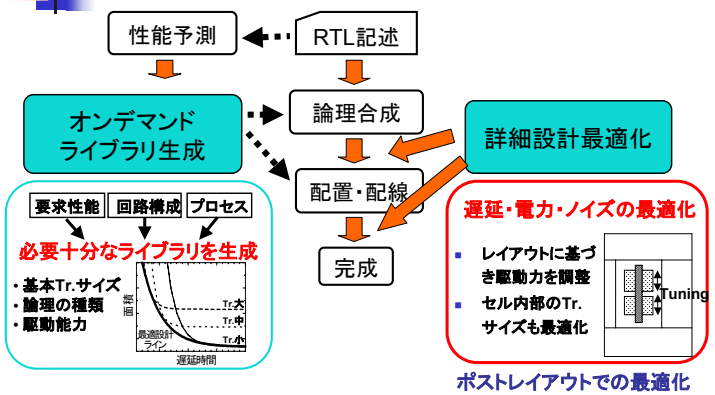
(b) 120 MHz 11-pitch cell

(c) 130 MHz 13-pitch cell

セルベース設計環境を利用した最適設計技術

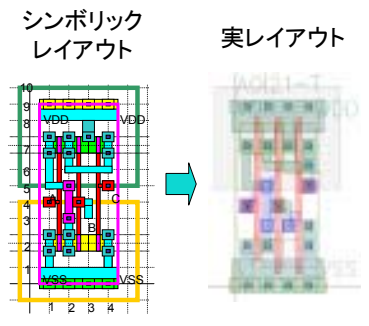
- セルベース設計における設計最適化
  - ライブラリ構成法
  - タイミング解析技術
  - 設計フロー
- セルベース設計環境を利用した最適設計
  - オンデマンドライブラリを用いた最適設計
  - データパス設計

ライブラリのオンデマンド生成による最適設計



駆動力可変レイアウト生成

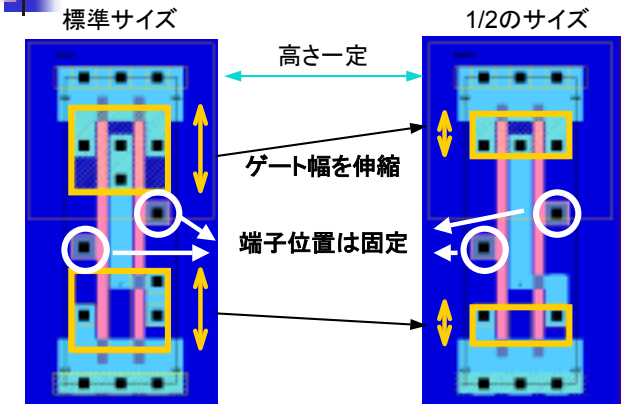
- プロセスに対して独立
- 十分なレイアウト密度
- 駆動能力が可変
  - セル高さ
  - 内部Tr.のゲート幅



階層的な仮想グリッドを用いたシンボリックレイアウト手法
(Hashimoto, SASIMI2000)

0.13, 0.18, 0.35, 0.6 μm ライブラリ生成に使用

レイアウト生成例
(端子位置固定してトランジスタ幅調整可能)

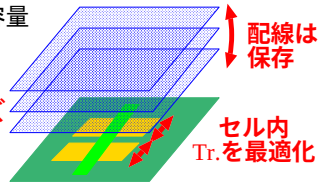


ポストレイアウトでの
トランジスタ寸法最適化

配線を保存した最適化

- 配線変更なく最適化結果を反映
- 詳細配線後の高精度な配線容量

目的関数:
- 消費電力
- クロストークノイズ

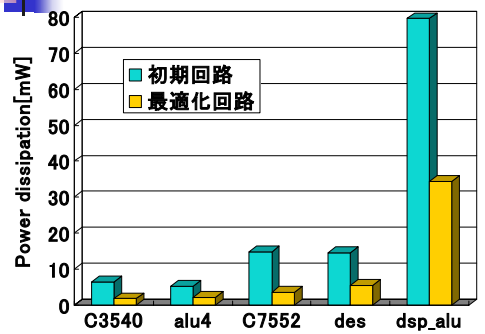


Tr.サイズを最適化(最小化)

- Tr.サイズは連続的に変更
- PN比も自由に変更
  (ノイズマージン制約考慮)

D
s

消費電力の最適化結果



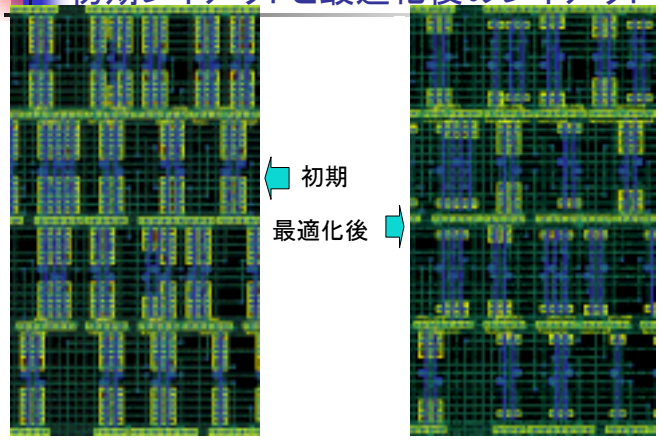
制約
- 最速回路
- 遷移時間0.5ns

遅延を増加させずに平均66%の消費電力削減

消費電力最適化具体例(des回路)

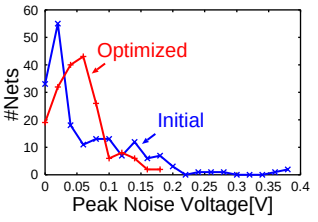
- 初期回路(x1,x2,x3,x4,,) 14.7mW
- セル寸法最適化(. + x0.15, x0.5) 11.3mW(-23%)
- トランジスタ寸法最適化 6.4mW(-56%)

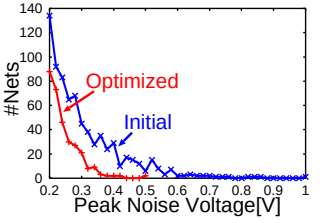
初期レイアウトと最適化後のレイアウト



クロストークノイズ削減結果

遅延時間を増加させることなくノイズを削減





des	回路	dsp_alu
0.40 -> 0.19	ノイズ最大値[V]	1.00 -> 0.50
41	CPU 時間[s]	1926
3.4k	#cells	13k

オンデマンドライブラリを用いた最適設計

- ライブラリのオンデマンド生成と、ポストレイアウトにおけるトランジスタ寸法調整
  - 消費電力最適化
    - 最大77%、平均66%の電力削減(実験結果)
    - 電源配線のピーク電流量も大幅削減(信頼性向上)
  - クロストークノイズ最適化
    - ピークノイズを50%削減(実験結果)

セルベース設計環境を利用した最適設計技術

- セルベース設計における設計最適化
  - ライブラリ構成法
  - タイミング解析技術
  - 設計フロー
- セルベース設計環境を利用した最適設計
  - オンデマンドライブラリを用いた最適設計
  - データパス設計

セルベース設計環境を利用したデータパス設計の検討内容

データパス回路では…

信号が規則的に伝搬

トランジスタレベルの設計、回路性能最適化

セルベース設計環境を用いて…

信号が規則的に伝搬するレイアウト設計

セル内部のトランジスタサイズの調整、最適化

それぞれにおいて回路性能を評価

セルベース設計環境を利用したデータパス設計

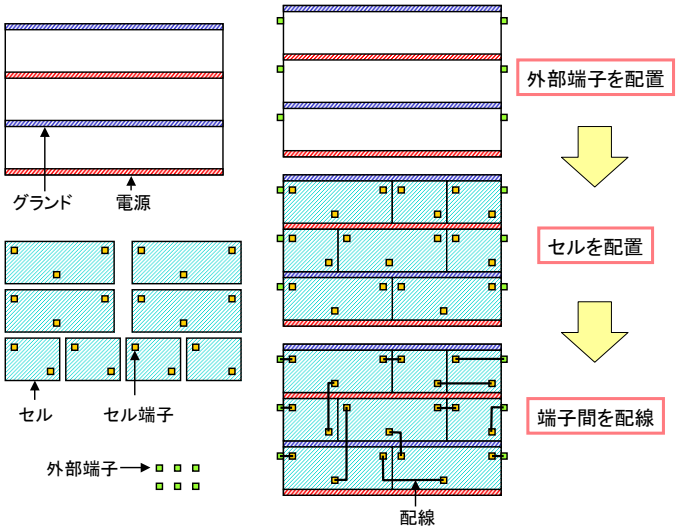
検討内容

- 信号が規則的に伝搬するレイアウト設計
- セル内部のトランジスタサイズの調整、最適化

3種類のレイアウト設計法

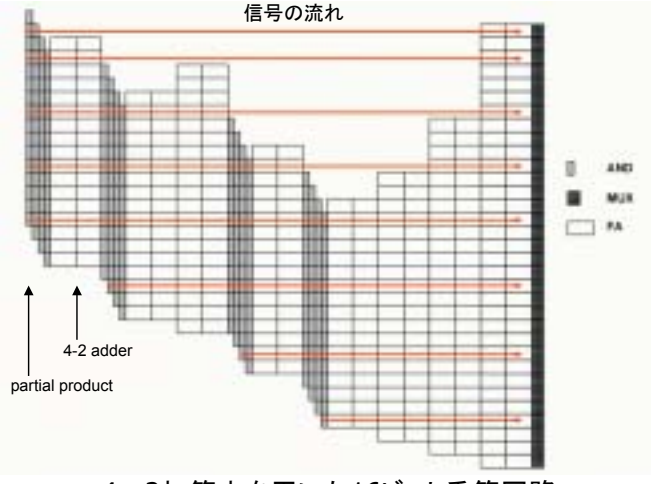
- 信号が規則的に伝搬するように外部端子とセルを手動配置、配線はCADを用いて自動配線
- 外部端子のみ手動配置し、セル配置と配線はCADを用いて自動
- 外部端子とセルの配置、配線の全てをCADを用いて自動

この3つの方法でレイアウト設計し、回路性能を比較

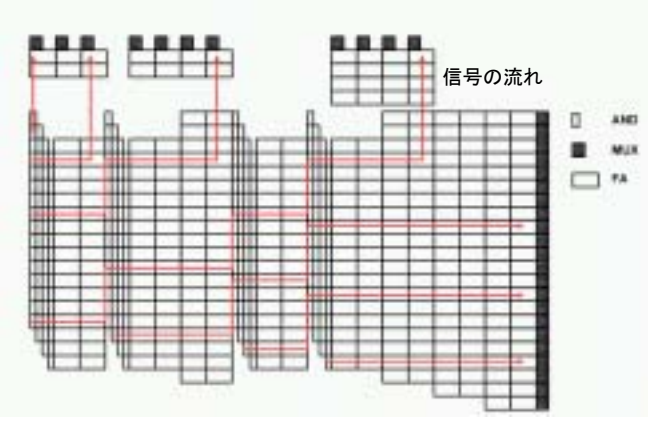


レイアウト設計する回路

- 0.35μmプロセスで設計
  - 8ビット、32ビットの桁上げ選択加算回路
  - 4-2加算木を用いた16ビットのツリー型乗算回路
- 設計手法に関わらず回路面積は全て一定とした
  - 8ビット加算回路　75×101　[μm²]
  - 32ビット加算回路　84×403　[μm²]
  - 16ビット乗算回路　515×378　[μm²]



4－2加算木を用いた16ビット乗算回路

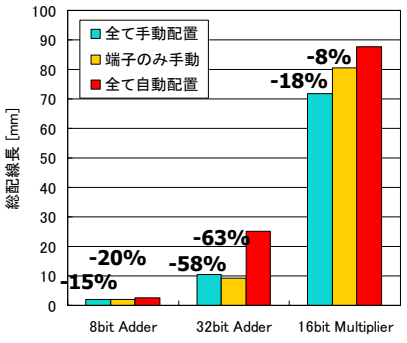


折り曲げを行った4－2加算木を用いた16ビット乗算回路

設計時間

- 全て手動配置で設計した場合
  - 8ビット加算回路　3時間
  - 32ビット加算回路　4時間
  - 16ビットツリー型乗算回路　5時間

設計結果(総配線長)

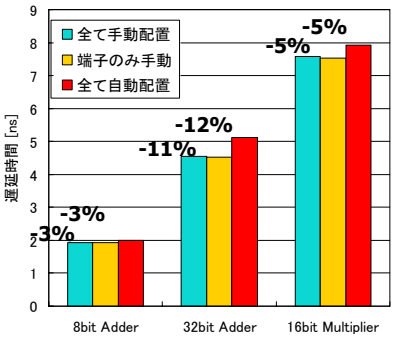


- 回路の総配線長は設計に使用したCADから得た

総配線長
最大 63%
平均 22% 減少

「全て手動配置」と「端子のみ手動」とであまり違いがない

設計結果(遅延時間)



- PathMill(トランジスタレベル遅延解析ツール)でクリティカルパスの遅延時間を評価

遅延時間
最大 12%
平均 4% 減少

「全て手動配置」と「端子のみ手動」とであまり違いがない

セル手動配置設計する際の問題点

- セルを手動配置するため、設計に時間がかかる
- 回路の規則性と面積削減の両立が困難

全て手動配置で設計した
4-2加算木を用いた
16ビット乗算回路

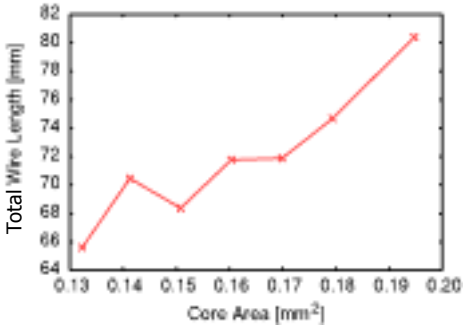
無駄な空きスペース
コア利用率=0.65

セル自動配置設計よる面積削減

- セル自動配置することで配線収容容量の許す限り簡単に面積削減可能
- 外部端子のみ適切に配置すればセルの配置はほとんど回路性能に影響しない

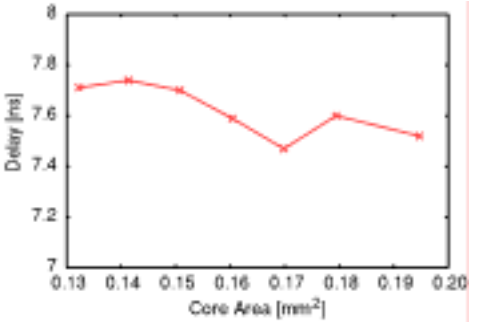
外部端子のみ手動配置してセルは自動配置で
乗算回路を設計して、回路面積と回路性能の関係を検討

結果(総配線長)



- コア利用率0.65から0.96に
- 総配線長は回路面積に比例して減少

結果(遅延時間)



- 回路面積とは無関係に遅延時間は7.6ns前後

外部端子のみ手動配置してセルは自動配置

セルベース設計環境を用いたデータパス設計

検討内容

- 信号が規則的に伝搬するレイアウト設計
- セル内部のトランジスタサイズの調整、最適化

ゲート幅最適化方法

- セルベース設計におけるゲート幅サイジングは以下の手法を想定
  - H. Onodera, M. Hashimoto and T. Hashimoto, “ASIC Design Methodology with On-Demand Library Generation,” *Proc. Symposium on VLSI Circuits*, 2001.
- 回路性能最適化問題はトランジスタのゲート幅を変数とする非線形最適化問題
- 回路を構成するセルの全トランジスタを対象
- 簡単のために、同じ論理のセルのトランジスタは全て共通させて変動させる

最適化手順

遅延時間が
最小になるよう最適化

遅延時間の3%
までの増加を制約にして

ゲート幅の総和が
最小になるよう最適化

回路内の容量を最小化
(～消費電力の最小化)

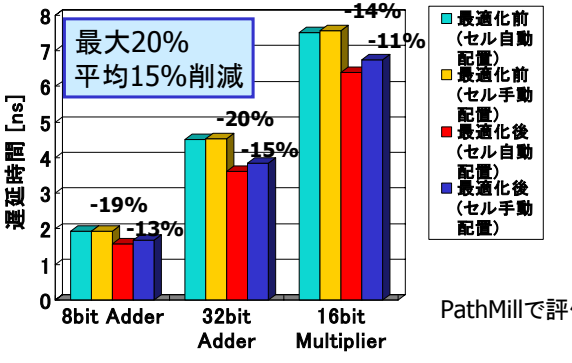
ゲート幅の最適化には非線形
数値最適化プログラムFSQPを
使用

対象回路は設計した
- 8ビット、32ビット加算回路　変数52
- 16ビットツリー型乗算回路　変数34

計算時間

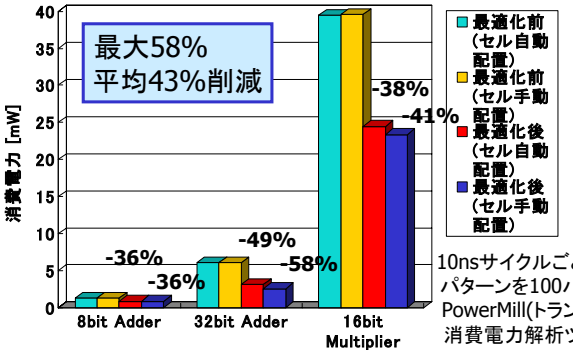
- 8ビット加算回路　4時間
- 32ビット加算回路　7時間
- 16ビットツリー型乗算回路　5日

最適化結果(遅延時間)



PathMillで評価

最適化結果(消費電力)



10nsサイクルごとにランダムな
パターンを100パターン与えて
PowerMill(トランジスタレベル
消費電力解析ツール)で評価

セルベース設計環境を用いたデータパス設計のまとめ

- セルベース設計環境を用いて高性能データパス設計する手法について検討
- セル配置が回路性能を与える影響は小さい
- トランジスタレベルの最適化がフルカスタム設計同様に重要である
  - 遅延時間　　最大20%、平均15%削減
  - 消費電力　　最大58%、平均43%削減

セルベース設計環境を利用した最適設計技術

- セルベース設計における設計最適化
  - ライブラリ構成法
    - 論理の種類は少なくてよい。駆動力は多数用意する。
  - タイミング解析技術
    - キャラクタライズに注意(テーブル分割、入力波形)
    - 等価入力波形、Statistical STA
- セルベース設計環境を利用した最適設計
  - オンデマンドライブラリを用いた最適設計
    - トランジスタ寸法の調節による消費電力･ノイズの最適化
  - データパス設計
    - 自動配置配線の活用
    - トランジスタ寸法の最適化