

ウェーハ集積回路の一配置手法

金杉昭徳 佐藤真司 山下公一

(株)富士通研究所

本手法は、あらかじめ1つの共通配置パターンを用意しておき、それをブロックの故障分布に適合するように当てはめる方法である。共通配置パターンは、使用するブロックが密集せず、かつ総配線長が短くなるように決定する。また、当てはめ方法は線型順序法を改良して用いる。FFTシステムに適用した結果、1000種類の故障分布に対する総配線長の平均値は共通配置の11%増と短く、良好な結果が得られた。また計算複雑度は、ウェーハ上の回路ブロック数をMとしたとき $O(M)$ で、高速である。適用対象は、同一の回路ブロックの繰り返しから構成されるアレイ状のシステムであり、多くのウェーハ集積回路に適用可能である。

A Placement Algorithm for Wafer Scale Integration

Akinori Kanasugi, Shinji Sato, and Koichi Yamashita

Fujitsu Laboratories Ltd. Atsugi

10-1 Morinosato-Wakamiya, Atsugi 243-01, Japan

The algorithm presented in this paper stands on a common placement pattern and its fitting procedure to the actual failure map. The common placement pattern is determined under the condition that the distribution of used blocks is sparse and the total line length is minimum. Modified linear ordering is used for the fitting to the failure map. An example of FFT system shows that the average sum of total line length for one thousand random failure maps is larger than that of the common placement pattern by only 11%. The calculation time is in the order of $O(M)$ (M : sum of blocks on the wafer). The algorithm is applicable to array systems which are configured by the repetition of the same block. Most WSI systems belong to this category.

1. はじめに

ウェーハスケール集積回路 (WSI) [1]-[3] は、一枚の半導体ウェーハ上に形成された複数の回路ブロック (LSIチップ) の中から、正常に動作するものを選び、それらを切り離すことなくウェーハ上で相互結線することにより製造される大規模集積回路である。WSIは、

- (1) 高集積度
- (2) 高信頼性
- (3) 低コスト

という優れた特長を有し、実用化が期待されている。しかしながらWSIの実現には次のような新しい技術開発が不可欠である。即ち、

- (1) ウェーハ上の回路ブロックをテストする技術
 - (2) 不良ブロックを避け、良品ブロック間を配線する再構成技術
 - (3) 巨大なウェーハを冷却を考慮して実装する技術
- 等である。

本論文では、再構成技術、特に配置問題を扱っている。一般に配置アルゴリズムは、セル個数の1.3~2乗に比例する計算時間がかかるといわれ、配線問題以上に効率的な手法が必要とされている[4]。LSIの製造においては、配置・配線は1品種に対し1度だけ行えばよい。しかしながらWSIにおいては、各ウェーハ上の回路ブロックの故障分布がウェーハ1枚毎に異なるため、目的とする回路とバルク形状が同一にも関わらず、生産するウェーハ枚数だけ配置・配線を実施する必要がある。したがって、従来LSIで用いられている手法をWSIの回路ブロック配置にそのまま適用することは、効率的とは言えない。そこで、本論文はWSIに適した効率のよい配置アルゴリズムを提案し、その有効性を示すことを目的としている。

提案する手法は、(1) 予め1つの配置パターンを用意しておく (これを以下、共通配置パターンと呼ぶことにする)、(2) それをブロックの故障分布に適合するように当てはめる方法である。共通配置パターンは、配線長・配線混雑度・発熱分布・歩留り分布等を総合的に考慮して決定する。一方、当てはめ方法には線型順序法を応用する。このとき線型順序は共通配置パターンにおけるネットカット数を計算することにより定めている。提案する手法の計算複雑度は、ウェーハ上の回路ブロック数をMとしたときO(M)で、計算量も少ないため非常に高速であり、さらに計算機プログラムの作成も容易である。

適用可能なシステムは、同一の回路ブロックの繰り返しから構成されるアレイ状のシステムであり、アレイプロセッサや高速フーリエ変換 (FFT) システム、大容量メモリシステム等が該当する。WSIでは不良ブロックの置き換えの容易さから、上記のようなシステムが適しており、多くのシステムに適用可能である。

以下本論文では、2.において提案する手法について述べ、3.において適用例を示し、有効性を確認する。

2. アルゴリズム

2.1 基本的な考え方

一般にLSIの製造においては、配置・配線は1品種に対し1度だけ行えばよいが、WSIにおいては、生産するウェーハ枚数だけ配置・配線を実施する必要がある。それは、各ウェーハ上の回路ブロックの故障分布がウェーハ1枚毎に異なるためであるが、各ウェーハとも目的とする回路とバルク形状は同一である。したがって、従来LSIで用いられている手法をWSIの回路ブロック配置にそのまま適用することは、効率的とは言えない。各ウェーハ毎に大型計算機を長時間使用して配置・配線を行うことは実用的でなく、コストの上昇も招く。したがってWSIには効率のよい配置アルゴリズムが必要である。WSIの場合、同一の回路 (システム) を同一のバルク (但し、ウェーハ1枚毎に故障分布は異なる) に配置・配線するのであるから、何らかの共通化が可能である。そこであらゆる故障分布に適用するための配置パターン (共通配置パターン) を用意し、これを故障分布に適合するように修正して当てはめる方法を提案する。

共通配置パターンは、ウェーハ上の全ての回路ブロックが良品と仮定したときの最も望ましい配置として求める。即ち、総配線長、コンタクトホール数、配線混雑度、発熱分布、歩留り分布、等の要素から評価関数を作成し、最も良い結果を与える配置パターンとして従来LSI配置手法により求める。なお、特殊な故障分布 (例: 良品ブロックがウェーハの左半分に集中した場合、等) に対して求めた配置パターンを共通配置としても、平均的に良い結果は期待できないと考えられる。

一方、当てはめ方法にはさまざまな方法が考えられるが、この作業に多くの労力 (計算量) を裂いては予め共通配置パターンを用意しておく効率化を計る意味が薄れる。そこで、簡単上に効率と配置結果が良いと言われる[4],[5] 線型順序法 (ジグザグ配置法、つづら折り法) の考え方を応用する。概要を以下に述べる (図1)。

- (1) 共通配置パターンおよび故障分布を一定順序 (線型順序; 後述) により1次元マップに変換する。
- (2) 故障分布上で冗長なブロック (良品が使用しないブロック) の位置を決定する。
- (3) 共通配置パターンを故障分布に順番に当てはめてゆく (但し、不良ブロック及び冗長なブロックは避けて配置する)。
- (4) (1) の順序により、1次元マップを2次元配置に変換する。

この方法の特徴は、使用するブロック位置を決定する代わりに、冗長なブロック位置を決定してから、共通配置パターンと同じ順番でブロックを当てはめる点にある。一般に冗長なブロック数は、使用するブロック数に比べて少ないので、短い時間で決定できる。

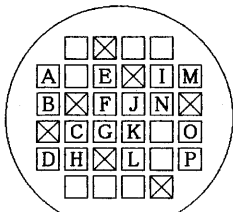
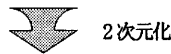
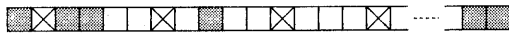
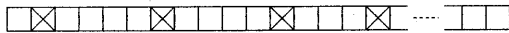
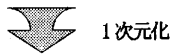
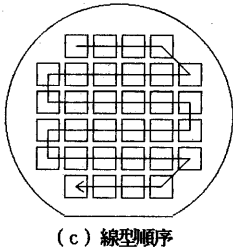
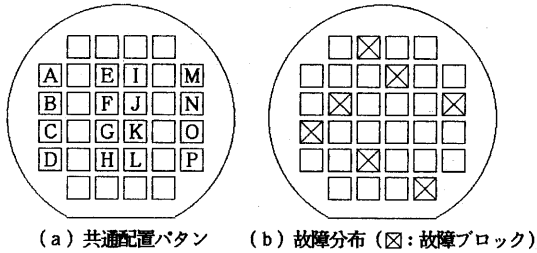


図1 本配置手法の概要

2.2 共通配置ボタン

提案する手法を実際に適用する際に共通配置ボタンを準備することが必要である。ここでは、共通配置ボタン設定の指針について述べる。

通常LSIの配置では、

(1) 未結線が出ないこと

(2) 総配線長が短いこと

等が基本的な要求であるが、WSIの場合はさらに以下の点に留意する必要がある。

(3) ウェーハ周辺部は配線が混雑しやすいため、非周辺部から使用すること

(4) ウェーハ上の発熱分布を均一化すること

これらの条件を満たす共通配置ボタンを求めるためには、条件を反映する評価関数を作成して最も好ましい値を与える配置を探せばよい。条件(1), (2)は各々異なる要求であるが、ここでは簡単のために、総配線長で代表させ、

$$\text{評価関数} = \text{総配線長} + \alpha \cdot \text{周辺使用度} + \beta \cdot \text{局所集中度} \quad (1)$$

なる評価関数を作成すると、この値を最小にする配置ボタンを共通配置とすればよい (α, β : 重み係数)。

ここで周辺使用度は、ウェーハ上の周辺ブロックが使用されている割合を意味する。しかしながら、周辺ブロックの配線混雑の割合は位置によって異なる。即ち、注目するブロックの上下左右4方向のうちブロックが存在していない方向数 D ($= 0 \sim 3$) が大きい程配線が混雑すると予想される。いま D を周辺係数と呼ぶことにし、一例を図2に示す。但し、 $D=0$ の場合は空白にしている。このようにすれば、周辺使用度は例えば次式で与えることができる。

$$\text{周辺使用度} = \sum \text{周辺係数} \quad (2)$$

次に、局所集中度について考察する。ウェーハの冷却を容易にするため、発熱分布を一様にするのが望ましい。

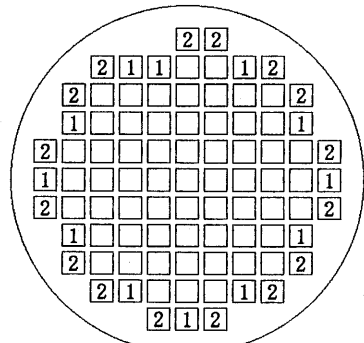


図2 周辺係数

特にバイポーラ素子のように発熱が大きい回路の場合は重要な問題になる。局所集中度を表す方法には、

(a) ウェーハ上を分割したときの各領域中の最大使用ブロック個数

(b) 各行および列中の最大使用ブロック個数
等が考えられる。(a)の場合、例えば図3のような分割を行う。この例では $6 \times 6 = 36$ の領域に分割している。いずれの場合も、各領域(あるいは、行、列)内の最大使用ブロック個数により、局所集中度は次式で表すことができる。

$$\text{局所集中度} = \max(\text{領域内使用ブロック個数}) \quad (3)$$

以上の検討により、共通配置パタンの評価が可能になる。従って、式(1)を最小化する配置パターンを求めれば、共通配置を得ることが可能である。また、式(2)、(3)より図4に示すような冗長ブロックの指定を行った後、総配線長を最小化するようにブロックを配置することも可能である。この例では88個のブロック中、40個を冗長ブロック(良品であっても使用しないブロック)に指定している。

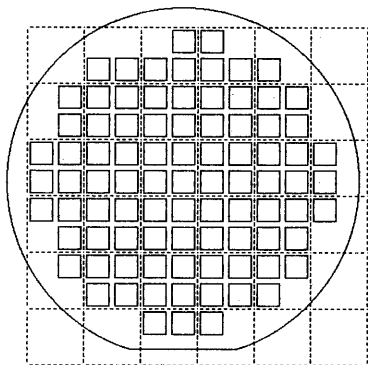


図3 ウェーハの分割例

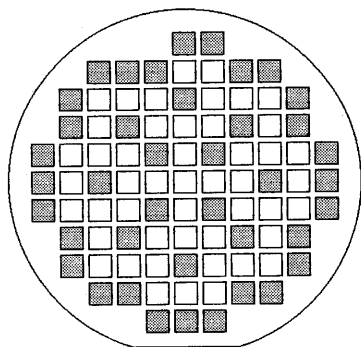
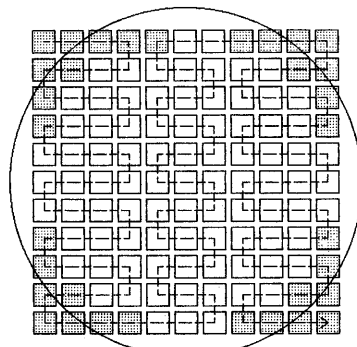


図4 冗長ブロックの割当例

2.3 線型順序

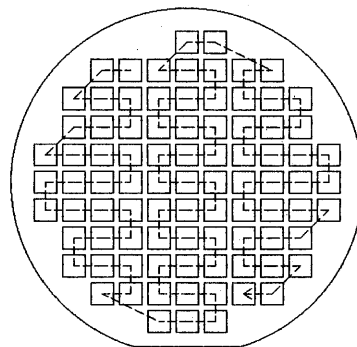
2.1節で述べた手法を実行する際、変換順序(線型順序)の決定方法が必要である。本手法の場合、通常の線型順序法と比較して考慮すべき点が2つある。第1番目は、丸いウェーハ内に線型順序を取める方法、そして第2番目は、共通配置パターン上で結合の強いブロック同士がなるべく近くに配置されるようにすることである。

ここで提案する方法は、ウェーハ上のブロックをすべて包含するような仮想的な矩形領域を想定して線型順序を作成し(図5(a))、その後実在しないブロックをバイパスすることによって目的とする線型順序を得る方法である(図5(b))。こうすることにより複雑な形状の中に収まる線型順序曲線を容易に発生することができる。この例では第4、5列目の間および第7、8列目間でウェーハが分割されているが、実際に適用するときは、共通配置パタンのネットカット数を計算し、局所最小箇所を分割する。これは、第2番目の条件を満たすようにするためである(詳細は3.1において例を用いながら述べる)。



(a) 仮想ウェーハ上の線型順序

(■: 実在しないブロック)



(b) 実際に得られた線型順序

図5 線型順序の設定

2.4 当てはめ方法

ここでは、2.1 で概説した共通配置パターンを当てはめるための手順を以下にアルゴリズムとしてまとめる。

[step 1] 良品ブロック数をP, その中で使用するブロック数をNとすると、冗長なブロック数Rは、

$$R = P - N$$

となる。もし、

- ・ $R < 0$ なら、終了 (システム構成不可能)。
- ・ $R = 0$ なら、冗長なブロック位置を決定する必要がないので [step 3] へ。
- ・ $R > 0$ なら、[step 2] へ。

[step 2] 冗長ブロックの位置を決定する。関数 $Scp(i)$, および変数Sを各々、

$Scp(i)$: 一次元共通配置上で左端から第i番目までの冗長ブロックの総和

S : 一次元故障分布上で左端から第i番目までの故障および冗長ブロックの総和

としたとき、両者の差が最少になるよう冗長ブロックを配置する。そのためのアルゴリズムを以下に示す。

```

S ← 0
for ( i = 1 ~ M )
  if ( 第 i 番目のブロックが良品 )
    if ( S < Scp(i) and R > 0 )
      ・ 第 i 番目を冗長ブロックに指定
      S ← S + 1
      R ← R - 1
    }
  else
    S ← S + 1
    if ( S > Scp(i) and
      既に冗長指定したブロックが存在する )
      ・ 直前に冗長指定したブロックを
      使用予定にする
      S ← S - 1
      R ← R + 1
    }
  }

```

[step 3] 以上により、冗長ブロックの位置が決定したので、最後に空きブロック位置に共通配置パターンと同じ順序で使用するブロックを配置してゆく。

以上のアルゴリズムから、本手法が $O(M)$ であり、また僅かな加減算だけで実行できることがわかる。

3. 適用例

最初に、高速フーリエ変換 (FFT) システムに適用した例について述べる。

図6にシステム構成を示す。各回路ブロックは同一の乗算累積器であるが、外部配線により異なる係数を与える。対応する共通配置パターンの一例を図7に示す。この例では88個のブロックが搭載されており、48個の良品を選んでシステムを構成する。

次に線型順序を求める過程について述べる。図7に示されているグラフは、各行および列の間でカットしたときの交差ネット数 (ネットカット数) である。この例では、行方向でカットしたときの方が列方向よりも大きく変化して

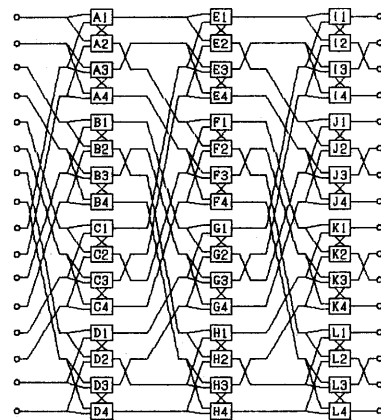


図6 FFTシステム

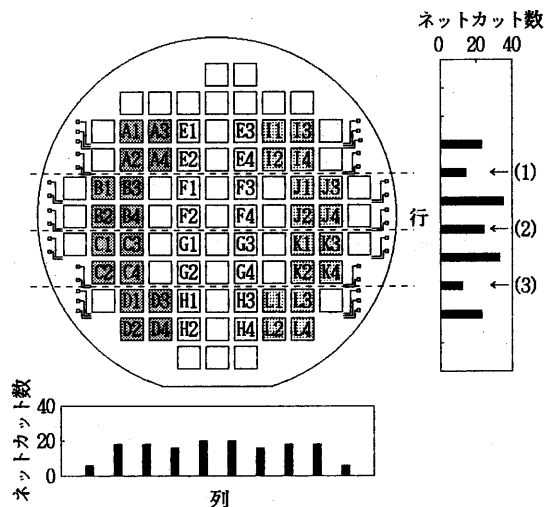


図7 共通配置パターンとネットカット数

いる。そして(1)-(3)の矢印で示す箇所で局所最小になっている。そこで図7に示されている破線によりウェーハ上の領域を複数のサブ領域(この例では4つ)に分割する。そして2.3で述べた手順により線型順序を作成する。得られた線型順序を図8に示す。同図で下に示されているグラフは、線型順序曲線上の任意の点から1つ先に進んだときの移動距離をマンハッタン距離で表したものである。理想的には、すべて1であることが望ましいが、2もしくは3進んでいる箇所が存在している。このような箇所は1次元上と2次元上での距離に差が生じるため好ましくない。そこで図9に示すような変更(円で囲んでいる箇所)を行うことにより、1度に3進む箇所をなくしている。

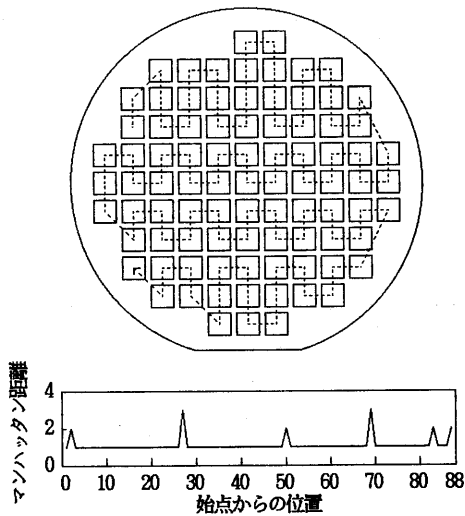


図8 線型順序1 (改良前)

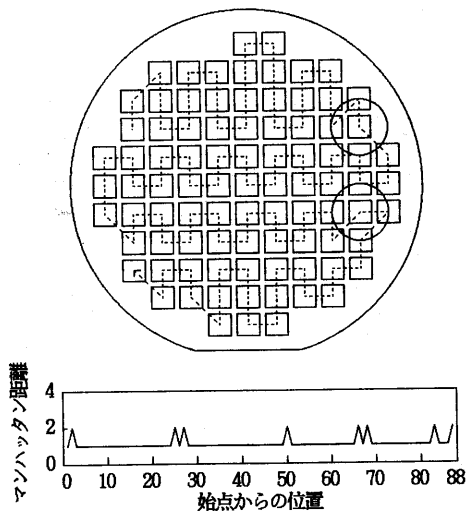


図9 線型順序1 (改良後)

図10に、20個の不良ブロックを含んでいる場合の配置結果の一例を示す。同図は共通配置パターンを良く反映している。

次に提案した手法の有効性を確認するために不良ブロック数を変えて計算機実験を試みた。図11は総配線長の不良ブロック数依存性を示している。ここで不良ブロック数は1~40個まで1つずつ増加させ、各点で100種類の故障分布を乱数によって作成し対応する配置結果および総配線長を求めている。同図でハッチングを施している部分が総配線長の分布範囲であり、実線は平均値を表している。実線の傾きの小さいことから本手法の有効性が確認できる。なお、不良ブロック数および故障分布を乱数によって1000種類発生させ、総配線長の平均を求めたところ、共通配置パターンの総配線長 320に対して、356となり、僅か11%増に収まった。

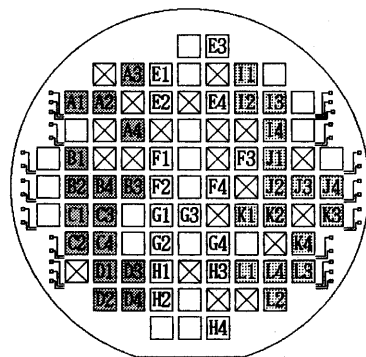


図10 配置結果の一例

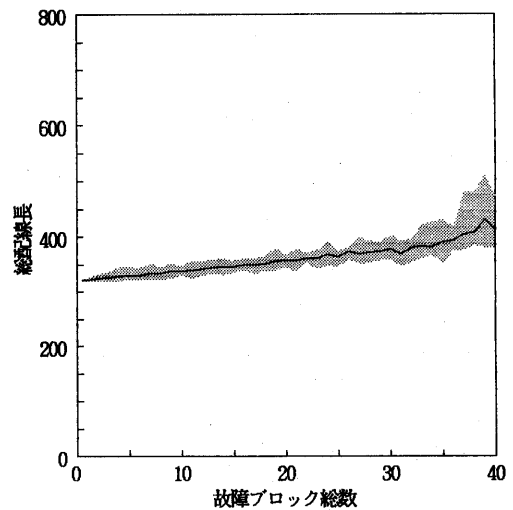


図11 総配線長の故障ブロック数依存性

同様に図9に示す線型順序を一行上にずらした線型順序(図12)を用いた結果を図13に示している。両者の比較から、図11の結果が傾きも小さく範囲も狭いことが明らかであり、提案した線型順序の設定方法の有効性、即ちネット交差数の局所最小箇所で分割した効果が確認できる。なお、1000種類の故障分布に対する総配線長の平均値は、共通配置パタンの22%増であった。

最後に、別の適用例を図14に示す。同図(a)はメモリシステムをモデルにして想定した仮想的なシステムであり、各ブロックはアレイ状に接続されている。同図(b)~(d)は各々、共通配置パターン、線型順序、配置結果の例である。この例からも本手法の有効性が確認できる。

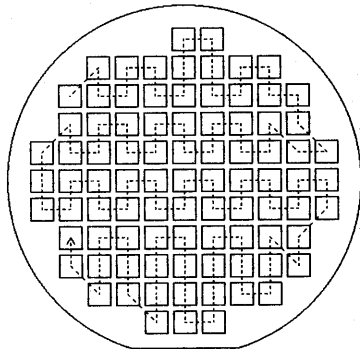


図12 線型順序2 (1行移動)

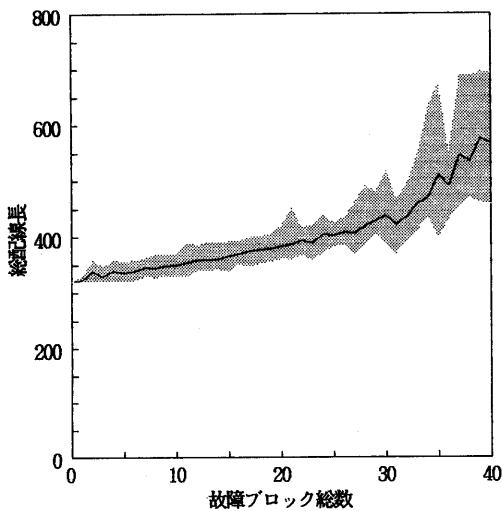
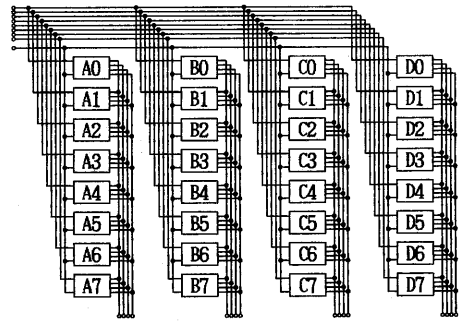
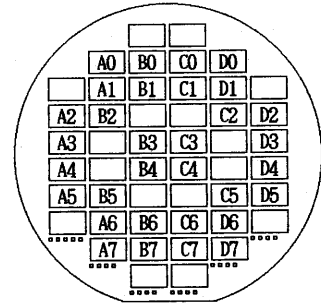


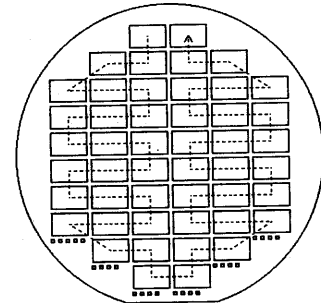
図13 総配線長の故障ブロック数依存性(図12に対応)



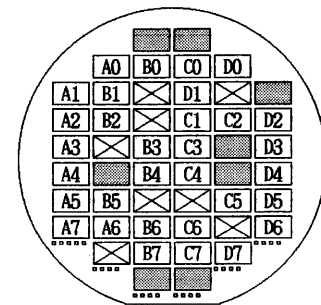
(a) システム構成



(b) 共通配置パターン



(c) 線型順序



(d) 配置結果

図14 適用例(2)

4. おわりに

ウェーハスケール集積回路に適した効率的な配置手法を提案した。手法は、予め共通配置パターンを用意しておき、それをブロックの故障分布に適合するように当てはめる方法である。計算複雑度は、ウェーハ上の回路ブロック数を M としたとき $O(M)$ で、計算量も少ないため非常に高速であり、さらに計算機プログラムの作成も容易である。

適用可能なシステムは、同一の回路ブロックの繰り返しから構成されるアレイ状のシステムであり、多くの $W S I$ システムに適用可能である。

文 献

- (1) Jesshope, C. and Moore, W. ed. : "Wafer Scale Integration", Adam Hilger, 1985.
- (2) Carlson, R.O. and Neugebauer, C.A. : "Future Trends in Wafer Scale Integration", Proceedings of the IEEE, vol. 74, no. 12, pp. 1741-1752 (1986)
- (3) 山下, 金杉, 土屋, 後藤 : "ウェーハスケール $L S I$ の可能性と限界", 『日経エレクトロニクス』, 1987年 6月1日号, no. 422, pp. 141-161.
- (4) 久保登, 西岡郁夫 : "レイアウト設計における $C A D$ 一分割・割付け・配置手法", 情報処理, Vol. 25, No. 10, pp. 1084-1089 (1984)
- (5) Kang, S. : "Linear Ordering and Application to Placement", Proc. 20th D.A. Conf., pp. 457 - 464 (1983).
- (6) Yamashita, K. et al. : "A Wafer Scale 170000-Gate FFT Processor with Built-in Test Circuits", IEEE J. of Solid-State Circuits, vol. 23, no. 2, pp. 336-342 (1988).