

# ゲート論理図生成における見やすさの基準と実現方法†

新 舎 隆 夫\*\* 碓 谷 幸 夫\*\*\*  
男 澤 康\*\*\*\* 久 保 隆 重\*\*

論理設計が機能レベルで行われるとき、機能論理の入力後に、この機能論理からゲート論理が論理生成システムにより生成される。このゲート論理はテキスト形式で記述されているため、ゲート論理図はこのゲート論理記述から自動的に生成されなければならない。本論文は超大型計算機用のゲート論理図生成に関するものである。論理図生成の主な課題は生成される論理図の見やすさであり、この見やすさは見やすさの基準により規定される。本論文では、見やすさの基準として12項目を選択し、これらの項目を実現するゲート論理図生成方法を提案する。本方法は、超大型計算機 M 68 X の設計への適用を通じて、その有効性が実証されている。

## 1. ま え が き

ゲート論理図は論理設計の主要なドキュメントの一つとして長年にわたり使用されてきた。近年、論理設計自動化技術の進歩に伴い、論理設計は機能レベルで行われるようになったため、ゲート論理図は論理設計における重要性を低下させているが、それでも論理設計以降の実装・診断設計には必要である。

論理設計が機能レベルで行われるとき、機能論理の入力後に、この機能論理からゲート論理が論理生成システムにより生成される。このゲート論理はテキスト形式で記述されているため、設計者がこのゲート論理記述からゲート論理図を作成することは非常に困難であり、ゲート論理図は人手の介入なしで生成されなければならない。それゆえ、このような論理図生成は対話形式の論理図入力・編集<sup>1),2)</sup>と扱う問題の性質が本質的に異なっている。

本論文は超大型計算機用 DA システム<sup>3),4)</sup>のゲート論理図生成に関するものである。論理図生成に関してこれまでに、計算機用 CAD システムのゲート論理図生成<sup>5)</sup>、マイクロプロセッサ用 MODEL システムの機能論理図生成<sup>6)</sup>、デジタルシステム用 IITD-CAD システムのシステムレベルからゲートレベルまでのいずれかのレベルの論理図生成<sup>7)</sup>等が発表されている。これらの論理図生成はいずれも、その課題は生成され

る論理図の見やすさであり、そのアプローチは、現在見やすさの定量的評価方法が確立されていないため、見やすさの基準を規定して実現する方法が採られている。本論文の論理図生成の課題とアプローチも同様である。

一方、見やすさの基準は一般に以下の三つの基準からなる。

(1) システムレベルの見やすさの基準：これは DA/CAD システムの仕様に依存してその採否が決まる見やすさの項目からなる。

(2) シート分割（論理分割）の見やすさの基準：これは取り扱う論理規模に依存してその採否が決まる見やすさの項目からなる。

(3) シンボル配置・配線の見やすさの基準：これは論理図生成に共通な見やすさの項目と論理素子を表現するシンボルの仕様に依存してその採否が決まる見やすさの項目からなる。

それゆえ、見やすさの基準は DA/CAD システムの仕様、取り扱う論理規模、シンボル仕様に依存して多くの見やすさの項目が存在しうる。一方、見やすさの基準の実現方法も、見やすさの項目または項目グループごとにその実現方法が必要であり、また、見やすさの基準が同一であってもシンボル仕様に依存してその実現方法が異なるために、多くのものが存在しうる。

本論文では、最初に第2章で文献 6), 7) と本論文の論理図生成の見やすさの基準と実現方法を比較し<sup>\*</sup>、本論文の位置づけを明確にする。次に本論文のゲート論理図生成に関して、第3章でシステム概要を、第4章で課題を、第5章で生成方法を順に述べ、第6章で

† Readability Elements in Gate-Level Logic Diagrams Generation and Their Implementation by TAKAO SHINSHA (Systems Development Laboratory, Hitachi, Ltd.), YUKIO IKARIYA (Kanagawa Works, Hitachi, Ltd.), KOO OTOKOZAWA (Hitachi Software Engineering Co., Ltd.) and TAKASHIGE KUBO (Systems Development Laboratory, Hitachi, Ltd.).

\*\* (株)日立製作所システム開発研究所

\*\*\* (株)日立製作所神奈川工場

\*\*\*\* 日立ソフトウェアエンジニアリング(株)

\* 文献 5) の論理図生成は比較が可能な詳細レベルの記載がないので除外する。

代表的な生成例を示す。最後に第7章で超大型計算機 M 68 X の設計への適用結果を述べる。

## 2. 論理図生成の比較

MODEL システム, IITD-CAD システムの論理図生成を順に論理図生成 1, 2 と略記して, これらと本論文の論理図生成の見やすさの基準と実現方法を比較し, 本論文の位置づけを明確にする。

### 2.1 見やすさの基準の比較

見やすさの基準は一般にシステムレベルの見やすさの基準, シート分割の見やすさの基準, シンボル配置・配線の見やすさの基準からなる。

#### (1) システムレベルの見やすさの基準

システムレベルの見やすさの基準は DA/CAD システムの仕様に依存してその採否が決まる見やすさの項目からなり, 本論文のみが以下の2項目を採用している。

- ・論理図作成基準の遵守: 従来, ゲート論理図は後述の論理図作成基準に則って作成されていた。そのため, この論理図作成基準の遵守はゲート論理図生成の前提条件になる。
- ・論理図間の対応づけ: 後述の機能論理図を使用する論理検証とゲート論理図を使用するディレイ検証を並行に行えるようにするため, 両者の論理図間の対応づけは必須条件になる。

#### (2) シート分割の見やすさの基準

シート分割の見やすさの基準は取り扱う論理規模に依存してその採否が決まる見やすさの項目からなる。一般に, 生成される論理図は見やすさの観点から1枚(複数シートの継貼りを許す)であることが望ましいが, 対象論理を1枚の論理図で表現することは物理的に限界がある。そこで, 対象論理規模が相対的に小さい場合は, 複数シートの継貼りを前提にして1枚の論理図が生成される。このとき, シート分割はシンボル配置・配線の後にシートサイズに合わせて機械的に行われるので, シート分割の見やすさの考慮は不要である。これに対して, 対象論理規模が相対的に大きい場合は, 複数枚の頁づけした論理図と論理図頁間にわたる信号線のクロスリファレンスが生成される。このとき, 見やすさを考慮したシート分割がシンボル配置・配線の前に行われる。

論理図生成 2 は前者のシート分割方法が使用されているのに対して, 論理図生成 1 と本論文は後者のシート分割方法を使用している。ここで, 本論文が後者の

シート分割方法を採用している理由は 100 万ゲートを超える大規模論理を対象にしているからである。

論理図生成 1 で採用されているシート分割の見やすさの基準は以下の3項目であり, 本論文が採用している基準も同様である。

- ・最少切断数: 最少切断数でシート分割された論理図は, 論理図頁間にわたって参照する信号線が少なくなるので見やすい。
- ・適切なシンボル密度 (その1): シンボル密度が適切な論理図は見やすい。
- ・論理の進行性 (その1): 論理は入力側から出力側へ進行する。論理図頁間でこの論理の進行性を保証した論理図は見やすい。

#### (3) シンボル配置・配線の見やすさの基準

シンボル配置・配線の見やすさの基準は論理図生成に共通な見やすさの項目とシンボル仕様に依存してその採否が決まる見やすさの項目からなる。

論理図生成に共通な見やすさの項目は以下の6項目である。

- ・適切なシンボル密度 (その2): 同上。
- ・論理の進行性 (その2): 1枚の論理図上で論理の進行性を保証した論理図は見やすい。
- ・最少交差数: 信号線の交差数が最少の論理図は見やすい。
- ・近接性: 接続しているシンボルを近くに配置した論理図は見やすい。
- ・直線性: 信号線の折れ曲がりが少ない論理図は見やすい。
- ・適切な配線密度: 配線密度が適切な論理図は見やすい。

一方, シンボル仕様に依存してその採否が決まる見やすさの項目も存在する。ここで, シンボル仕様はシンボルサイズの変異性, 信号線を接続するシンボルの入出力端子の位置, 例外的に扱われる特殊シンボルの有無を意味する。

論理図生成 1, 2 は, シンボルの両方向の長さが可変であり, 入出力端子の位置はシンボルの種類に依存し, 固定されていない。これに対して, 本論文の論理図生成は, シンボルの縦方向の長さは可変であるが, 横方向の長さは固定であり, 入力端子の位置はシンボルの左側に, 出力端子の位置はシンボルの右側に固定されている。そのため, 本論文はシンボル仕様に依存する見やすさの項目として以下の2項目を採用している。

- ・階層性：最も出力側のシンボルを起点にしてシンボル段数を基準にシンボルを階層的に配置した論理図は見やすい。
- ・対称性：出力側シンボルをそれが接続されている入力側シンボルの重心位置に配置した論理図は見やすい。

また、論理図生成1は特殊シンボルが扱われていないが、論理図生成2はバスを、本論文はECL回路系のゲート論理を対象にしているために結線論理を特殊シンボルとして扱っている。そのため、論理図生成2と本論文はシンボル仕様に依存する見やすさの項目として特殊シンボル固有の見やすさを採用している。

## 2.2 実現方法の比較

システムレベルの見やすさの基準の実現方法はその見やすさの基準に依存する。これに対して、シート分割とシンボル配置・配線の見やすさの基準の実現方法はシンボル仕様に依存し、見やすさの基準が同一であっても異なっている。本論文では、システムレベルの見やすさの基準は本論文のみが採用しているので、その実現方法は比較の対象外とし、シート分割とシンボル配置・配線の見やすさの基準の実現方法を方式とアルゴリズムの2レベルで比較する。

### (1) 方式の比較

論理図生成2はシート分割の見やすさの考慮が不要であり、シンボルの両方向の長さの可変幅が共に大きいシンボルを対象にしている。そのため、この論理図生成は、同一シンボルサイズを仮定した粗いシンボル配置・配線、この結果を使用して実際のシンボルサイズを考慮した細かいシンボル配置・配線、機械的なシート分割の順序で行う方式が採用されている。

これに対して、論理図生成1と本論文はシート分割の見やすさを考慮しており、論理図生成1はシンボルの両方向の長さの可変幅が共に小さいシンボルを、本論文の論理図生成はシンボルの縦方向の可変幅が最大4倍で、横方向の長さが固定のシンボルを対象にしている。そのため、これらの論理図生成はシート分割、シンボル配置・配線の順序で行う方式を採用している。

したがって、両者のグループは方式レベルで異なっている。

### (2) アルゴリズムレベルの比較

論理図生成1の対象シンボルは両方向の長さが可変であり、入出力端子の位置はシンボルの上下左右のいずれの側にも存在する。そのため、この論理図生成はシンボルの両方向の長さの可変性が取り扱えるアルゴ

リズムが使用されている。具体的には、シート分割にクラスタ成長法<sup>9)</sup>の応用が、シンボル配置に組立て式配置法<sup>9)</sup>とクラスタ成長法の応用が、シンボル配線に二段階配線法<sup>10)</sup>の応用が各々使用されている。

これに対して、本論文の対象シンボルは縦方向の長さは可変であるが、横方向の長さは固定であり、入力端子の位置はシンボルの左側のみ、出力端子の位置はシンボルの右側にのみ存在する。そのため、この論理図生成はシンボルの縦方向の長さの可変性が取り扱えるアルゴリズムを使用することになる。

このアルゴリズムは上述の論理図生成1のアルゴリズムが適用可能であるかもしれないが、シンボルの両方向の長さの可変性を取り扱うアルゴリズムとシンボルの一方の長さの可変性を取り扱うアルゴリズムはその考え方が基本的に異なっている。そのため、本論文はシンボルの一方の長さの可変性を取り扱うアルゴリズムを使用している。具体的には、シート分割に分離度法とレベル内・レベル間分割法を、シンボル配置につづら折り法と重心法<sup>11)</sup>の応用と優先度法<sup>11)</sup>の応用を、シンボル配線に線分探索法<sup>12)</sup>を各々使用している。ここで、重心法と優先度法はシステム構造化を目的とした平面グラフ生成のシンボル配置方法であり、同一サイズのシンボルを対象にしている。そのため、本論文はこれらの方法がシンボルの縦方向の長さの可変性を取り扱えるように拡張している。

したがって、両者はアルゴリズムレベルで異なっている。

## 3. ゲート論理図生成のシステム概要

超大型計算機用論理DAシステム<sup>3),4)</sup>(ゲート論理図生成に関する部分)の概要を図1に示し、ゲート論理図生成のシステム概要を述べる。

本システムの入力はFL(Function Logic Diagram)<sup>13)</sup>と呼ばれる機能論理図であり、この論理図は以下の3種類のモジュールを構成要素として記述される。

(1) 汎用モジュール：複数ゲートからなる一つの論理単位で、その機能がブール式または真理値表で個別に定義されるモジュール。

(2) 標準モジュール：複数ゲートからなる一つの論理単位で、標準的なマクロ論理として、そのゲート論理記述とシンボル配置情報がライブラリに登録されているモジュール。

(3) 基本モジュール：ゲートと1対1に対応する

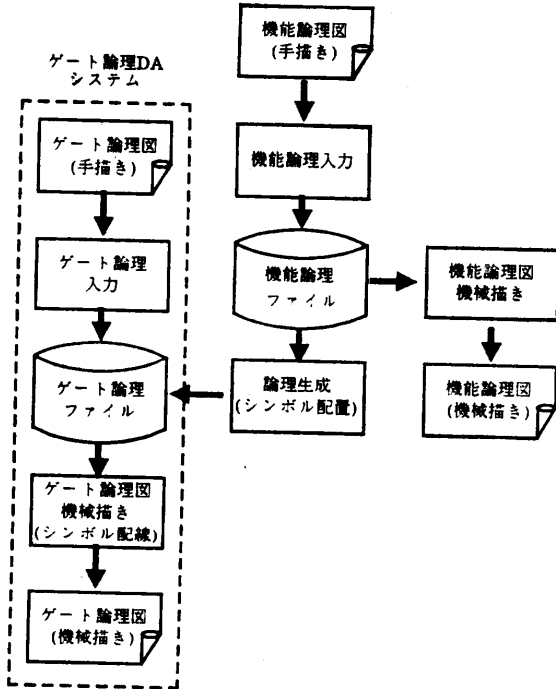


図1 超大型計算機用論理 DA システム  
Fig. 1 The logic design automation system for very large scale computers.

モジュール。

本システムでは、機能論理図（手描き）の入力により機能論理記述の機能論理ファイルが作成される。その後、このファイルを入力として、機能論理図（機械描き）が出力され、一方、ゲート論理記述が論理生成システムにより生成される。この論理生成はモジュール単位に行われ、汎用モジュールは POLARIS<sup>14)</sup> により、標準モジュールはライブラリ展開により、基本モジュールはコピーにより、各々のゲート論理記述が生成される。論理生成システムは、ゲート論理記述を生成後に、各ゲートのシンボル配置情報（シンボルの配置頁と配置座標）を生成してゲート論理記述に付加し、ゲート論理ファイルを生成する。その後、既存のゲート論理図機械描きシステムがこのファイルを入力し、シンボル配線情報を生成し、ゲート論理図（機械描き）を出力する。

本論文は論理生成システムのシンボル配置機能を取り扱っており、シンボル配線機能は従来より実用化されているので対象外とする。

#### 4. ゲート論理図生成の課題

本論文のゲート論理図生成で取り上げている見やす

さの項目は以下のとおりである。

(1) システムレベルの見やすさの項目：これは論理図作成基準の遵守と論理図間の対応づけの2項目である。

(2) シート分割の見やすさの項目：これは最少切断数、適切なシンボル密度（適切なシンボル密度（その1））、論理の進行性（論理の進行性（その1））の3項目である。

(3) シンボル配置・配線の見やすさの項目：これは階層性、最少交差数、近接性、対称性、直線性、適切な配線密度、結線論理シンボル固有の見やすさの7項目である。

本論文では、適切なシンボル密度（その2）は論理図作成基準の遵守に、論理の進行性（その2）は階層性に各々含まれているので、これらの項目を取り上げない。また、第3章で述べたようにシンボル配置のみを取り扱うので、シンボル配置・配線の各見やすさの項目の実現方法はシンボル配置側の実現方法のみを取り扱う。このとき、適切な配線密度は論理図作成基準の遵守に含まれるので、この項目を取り上げない。また、結線論理シンボル固有の見やすさの実現方法はその記述に多くの紙面を要するので割愛し、文献15)に委ねる。

上述の前提の下で、見やすさの項目の優先順序を述べる。第1番目はシステムレベルの見やすさの項目であり、論理図作成基準の遵守と論理図間の対応づけ（両立可能）である。第2番目はシート分割の見やすさの項目であり、優先順序は最少切断数と適切なシンボル密度（競合するため順不同）、次に論理の進行性である。第3番目はシンボル配置の見やすさの項目であり、優先順序は階層性、次に最少交差数、次に近接性と対称性と直線性（競合するため順不同）である。ここで、階層性を最優先にしている理由は、超大型計算機用のゲート論理図では、シンボル段数がディレイの推定に有効であるからである。

#### 5. ゲート論理図生成方法

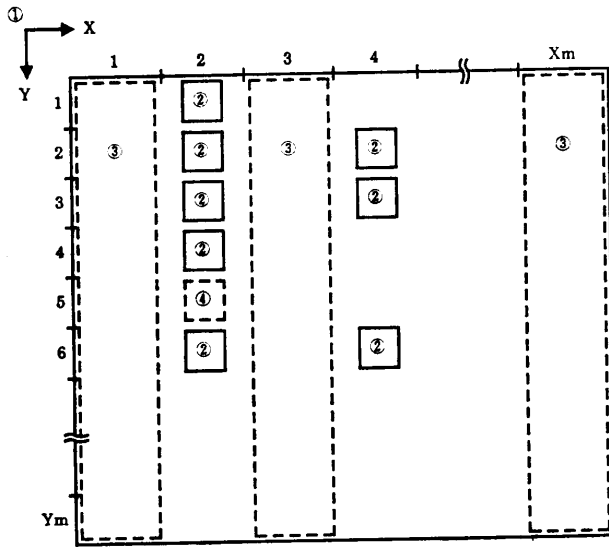
上述の見やすさの項目の優先順序に従って、これらの項目を実現するゲート論理図生成方法を述べる。

##### 5.1 システムレベルの見やすさの項目の実現方法

システムレベルの見やすさの項目は論理図作成基準の遵守と論理図間の対応づけ（両立可能）である。

(1) 論理図作成基準の遵守（前提条件）

ゲート論理図作成基準はゲート論理図生成の前提条



- ①: マクロ座標系
- ②: 配置シンボル
- ③: 信号名表示領域
- ④: シンボル配線領域

図 2 ゲート論理図作成基準

Fig. 2 Drawing standards of gate-level logic diagrams.

件である。その主なものを図 2 に示し、以下に述べる。

- ・本論理図はマイクロ座標系  $(x, y)$  とマクロ座標系  $(X, Y)$  の 2 種類が存在する。ここで、マイクロ座標系は  $x, y$  方向の単位長が同一であるのに対して、マクロ座標系は  $X, Y$  方向の単位長が異なっており、 $X$  方向の 1 マクロ長は 20 ミクロ長、 $Y$  方向の 1 マクロ長は 8 ミクロ長である。このように  $X$  方向の単位長が  $Y$  方向の単位長より大きい理由は信号名表示のためである。このとき、シンボルはマクロ座標系  $(X, Y)$  ( $X=1, \dots, X_m, Y=1, \dots, Y_m$ ) にのみ配置可能である。
- ・ $X=1, X=X_m$  の 2 領域は信号名 (クロスリファレンスの信号名を含む) 表示領域であり、シンボルは配置不可能である。
- ・ $Y$  方向において、4 マクロ長連続してシンボルを配置した場合、シンボル配線領域として 1 マクロ長以上を確保する。
- ・ $X$  方向において、隣接するシンボルの間は信号名表示領域として 1 マクロ長以上を確保する。

なお、これらのゲート論理図作成基準は適切なシンボル密度 (その 2) と適切な配線密度を考慮して決められている。

(2) 論理図間の対応づけ方法

第 3 章で述べたように、機能論理図はモジュールを

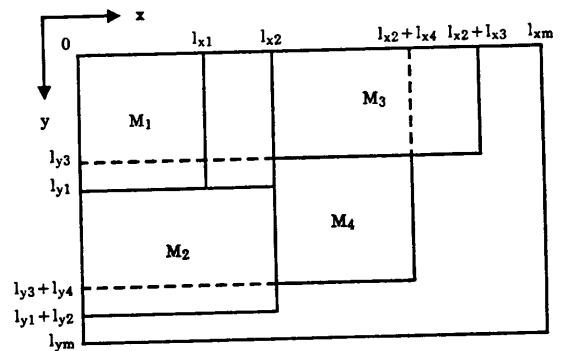
構成要素として記述され、論理生成はモジュール単位に行われる。ここで、各モジュールはユニークなモジュール ID を有しており、この ID は生成される各ゲートのゲート ID の一部として引き継がれる。それゆえ、モジュール ID が機能論理図とゲート論理図を対応づけるキーになる。このとき、シンボル配置 (シート分割を含む) はこの対応関係を保持するためにモジュールベースが基本になる。

モジュールベースのシンボル配置は各モジュールのシンボル相対配置情報を使用し、十字枝法により行う。ここで、シンボル相対配置情報はモジュールの配置開始座標を原点に設定したときのシンボル配置情報である。この情報は、汎用モジュールの場合は後述の方法により生成し、標準モジュールの場合はライブラリ内に格納されており、基本モジュールの場合は 1 ゲートのため特定の情報になる。十字枝法によるモジュールベースのシンボル配置例を図 3 に示し、その処理手順を以下に述べる。

Step 1:  $p_c=1, x_c=y_c=0$  とする。ここで、 $p_c$  は現在の頁を、 $x_c, y_c$  は現在の  $x, y$  座標を各々表す。

Step 2: 機能論理図上でモジュール配置座標の小さい順に (頁、次に  $X$  座標、次に  $Y$  座標) モジュール  $M_i$  を選択する。 ( $M_i$  がシート分割されている場合はサブモジュール番号  $j$  の小さい順にサブモジュール  $SM_{ij}$  を選択する。) ここで、選択する  $M_i$  がなければ本処理は終了する。

Step 3:  $M_i$  は矩形サイズとして扱う。  $y_c+l_{yi} \leq l_{ym}$  であつ  $x_c+l_{xi} \leq l_{xm}$  ならば、各シンボルの頁を  $p_c$  に



- $M_i$ : モジュール  $i$
- $l_{xi}, l_{yi}$ :  $M_i$  の  $x, y$  方向の長さ
- $l_{xm}, l_{ym}$ : 論理図面の  $x, y$  方向の長さ

図 3 十字枝法によるモジュールベースのシンボル配置例

Fig. 3 An example of module-based symbol placement using the cross chopping method.

し、 $x_c, y_c$  をマクロ長  $X_c, Y_c$  に各々換算し、各シンボルの  $X, Y$  座標に  $X_c, Y_c$  を各々加算し、 $y_c = y_c + l_{vi}$  とし、Step 2 へ分岐する。そうでなければ、次の Step へ進む。ここで、 $l_{xi}, l_{vi}$  は  $M_i$  の  $x, y$  方向の長さを、 $l_{xm}, l_{ym}$  は論理図面の  $x, y$  方向の長さを各々表す。

Step 4:  $y_c + l_{vi} > l_{ym}$  ならば、 $y_c = 0, x_c = x_c + \text{MAX}$  (配置開始  $x$  座標が  $x_c$  の各  $M_i$  の  $l_{xi}$ ) とし、Step 3 へ分岐する。そうでなければ、次の Step へ進む。

Step 5: ( $x_c + l_{xi} > l_{xm}$  の場合)  $p_c = p_c + 1, x_c = y_c = 0$  とし、Step 3 へ分岐する。

5.2 シート分割方法

シート分割の見やすさの項目の優先順序は最少切断数と適切なシンボル密度 (競合するため順不同)、次に論理の進行性である。最初の二つの項目は水平・垂直分割により、最後の項目はサブモジュールの通番管理により各々実現する。シート分割方法を以下に述べる。

(1) 全体の処理手順

Step 1:  $j = p = 1$  とし、当該モジュール  $M_i$  をサブモジュール  $SM_{ij}$  ( $j = 1$ ) とする。ここで、 $j$  はサブモジュール番号、 $p$  は現在のサブモジュール数を各々表す。

Step 2:  $SM_{ij}$  内の最も出力側の各シンボルを起点にファンイン・トレースを行い、各シンボルのレベル (シンボル段数) を設定する。

Step 3: ゲート論理図作成基準に準拠し、レベル順に、同一レベルのシンボルが1列になるように  $SM_{ij}$  の仮配置を行う。

Step 4:  $SM_{ij}$  の  $x, y$  方向の長さ  $l_{xi}, l_{vi}$  を算出する。

Step 5:  $l_{vi} > l_{ym}$  ならば次の Step へ進み、 $l_{vi} \leq l_{ym}$  かつ  $l_{xi} > l_{xm}$  ならば Step 7 へ、 $l_{vi} \leq l_{ym}$  かつ  $l_{xi} \leq l_{xm}$  ならば Step 9 へ各々分岐する。ここで、 $l_{xm}, l_{ym}$  は論理図面の  $x, y$  方向の長さを表す。

Step 6: 水平分割を試みる。分割可能ならば Step 8 へ分岐し、不可能ならば次の Step へ進む。

Step 7: 垂直分割を行う。

Step 8: 論理分割により生成されたサブモジュールの通番管理を行い、Step 2 へ分岐する。

Step 9:  $j = j + 1$  とする。  $j \leq p$  ならば Step 2 へ分岐し、  $j > p$  ならば本処理は終了する。

(2) 水平分割の処理手順

水平分割は以下の分離度法により行う。

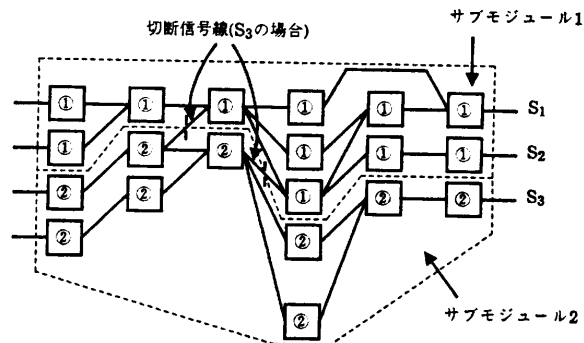
Step 1: サブモジュール  $SM_{ij}$  の起点シンボルを決定する。第1レベルのシンボル数が2以上ならば、これらのシンボルを起点親シンボルとし、1ならば、シンボル数が2以上の最上位レベルを見つけ、これらのシンボルを起点子シンボルとする。

Step 2: 分離度を算出する。各起点シンボル  $S_a$  について以下の処理を行う。最初に、 $S_a$  が起点子シンボルならば、 $S_a$  のファンアウト信号線を仮切断する。次に、 $S_a$  からファンイン・トレースを行い、切断信号線 (ファンイン・トレース中にたどった各シンボルの未トレースのファンアウト信号線) を認識する。最後に、これらの切断信号線を仮切断し、分離度  $\rho_a$  を次式により算出する。

$$\rho_a = \prod_{k=1}^{n_a} \frac{b_m}{b_k}$$

ここで、 $n_a$  は生成されるサブモジュールの個数を、 $b_k$  は生成される各サブモジュール内のシンボル数を、 $b_m$  は  $b_k$  の最大値を各々表す。

Step 3: 論理分割を試みる。あらかじめ指定されている許容分離度が  $\rho_m$  のとき、 $\text{MIN } \rho_a \leq \rho_m$  ならば、当該切断信号線を実切断し、論理分割を行う。MIN



①: サブモジュール1に属するシンボル (シンボル数=11)  
②: サブモジュール2に属するシンボル (シンボル数=9)  
(a) 適用例と分割結果

$S_a$	$n_a$	$b_k$ 's	$\rho_a$
$S_1$	3	14, 2, 4	$\frac{14}{14} \times \frac{14}{2} \times \frac{14}{4} = 24.5$
$S_2$	3	12, 4, 4	$\frac{12}{12} \times \frac{12}{4} \times \frac{12}{4} = 9$
$S_3$	2	11, 9	$\frac{11}{11} \times \frac{11}{9} = 1.2$

(b) 分離度

図4 水平分割例

Fig. 4 A horizontal decomposition example.

(a) An applicant and the decomposition result.

(b) The separation ratios.

$\rho_n > \rho_m$  ならば、論理分割は行わない。ここで、 $\rho_m$  は本論文のシート分割方法の試行評価を通じて決めており、 $\rho_m = 1.5$  を使用している。

$l_{xm} = 9$  マクロ長、 $l_{ym} = 5$  マクロ長、 $\rho_m = 1.5$ 、シンボルの  $x, y$  方向の長さが共に1マクロ長で、 $X, Y$  方向の1マクロ長が同一であると仮定した場合の水平分割例を図4に示す。

(3) 垂直分割の処理手順

垂直分割はレベル間分割法とレベル内分割法のいずれかにより行う。

Step 1: つづら折り法によりサブモジュール  $SM_{ij}$  の仮配置を行う。最初に、 $SM_{ij}$  内の各シンボルのレベルを設定する。次に、ゲート論理図作成基準に準拠し、レベル順に、同一レベルのシンボルを1列に、そのときの  $y$  方向の長さ  $l_{yj}$  が論理図面の  $y$  方向の長さ  $l_{ym}$  を超える場合は、同一レベルのシンボルが複数列にわたってつづら折りになるように  $SM_{ij}$  の仮配置を行う。そして、出力側のシンボル列を起点にして各シンボルのレベルを再設定する。

Step 2: サブモジュール  $SM_{ij}$  の分割数  $m$  を算出する。すなわち、 $SM_{ij}$  の  $x$  方向の長さ  $l_{xij}$  を算出し、 $l_{xij}$  を論理図面の  $x$  方向の長さ  $l_{xm}$  で除算し、その商を  $m$  とする。ここで、その端数が0ならば、 $m = m - 1$  とする。

Step 3: レベル間分割を試みる。最初に、 $SM_{ij}$  の各レベル  $r$  ( $r \geq 2$ ) について、 $(r-1)$  以下のレベルのシンボルと  $r$  以上のレベルのシンボルの間のファンアウト信号線総数  $f_r$  を算出する。次に、 $f_r$  の小さい順に  $m$  個の  $r$  のセットを順次選択し、分割制約条件「 $m$  回のレベル間論理分割により生成される各サブモジュールは論理図1頁内に配置可能である」を満たすかを判定する。この条件を満たす  $m$  個の  $r$  のセットが存在するならば、それらのファンアウト信号線をすべて実切断し、論理分割を行う。存在しなければ、次の Step へ進む。

Step 4: レベル内分割を行う。Step 1 で行った  $SM_{ij}$  の仮配置 (つづら折り配置) において、 $SM_{ij}$  の入力側からの長さが  $l_{xm}$  に等しくなる最小レベル  $r_1$  のシンボル列を見つけ、 $(r_1-1)$  以下のレベルのシンボルと  $r_1$  以上のレベルのシンボルの間のファンアウト信号線をすべて実切断し、論理分割を行う。

$l_{xm} = 9$  マクロ長、 $l_{ym} = 5$  マクロ長、シンボルの  $x, y$  方向の長さが共に1マクロ長であると仮定した場合のレベル間分割例を図5に、レベル内分割例を図6に

示す。ここで、図5の例は  $X, Y$  方向の1マクロ長が同一であると仮定しており、図6の例は配線を明示するために  $X, Y$  方向の1マクロ長の対応関係を無視している。

(4) サブモジュールの通番管理の処理手順

サブモジュール  $SM_{ij}$  が論理分割により  $q$  個のサブモジュールに分割されたとする。最初に、 $p = p + q - 1$  とする。次に、 $j$  より大きい番号をもつサブモジュールの各番号に  $(q-1)$  を加算する。最後に、生成された  $q$  個のサブモジュールの各番号が、 $j$  から  $(j+$

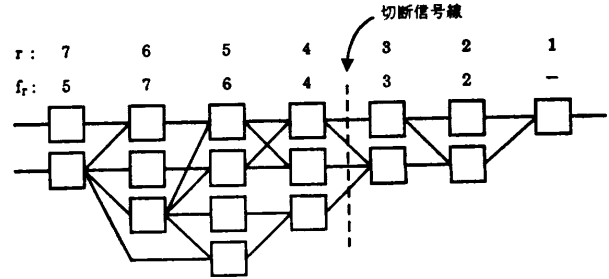
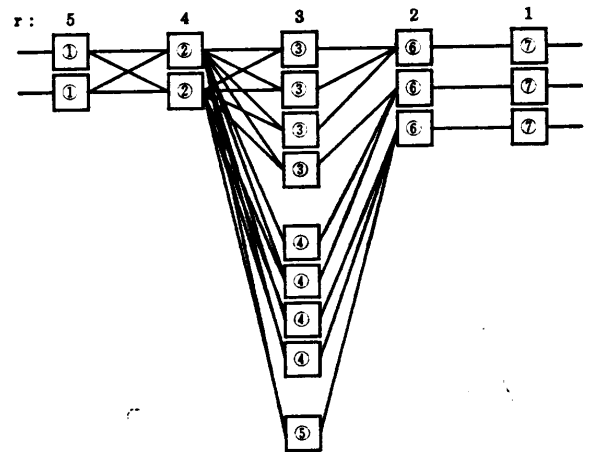
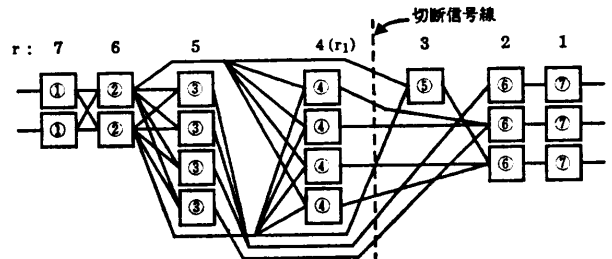


図5 レベル間分割例  
Fig. 5 A straight decomposition example.



(a) レベルに基づく配置例



(b) (a)のつづら折り配置

図6 レベル内分割例

Fig. 6 A zigzag decomposition example.  
(a) A rank-based placement example.  
(b) The re-alignment of example (a).

q-1) までの範囲で、切断信号線を出力信号線とするサブモジュールの番号がその切断信号線を入力信号線とするサブモジュールの番号より小さくなるように設定する。

5.3 シンボル配置方法

シンボル配置の見やすさの項目の優先順序は階層性、次に最少交差数、次に近接性と対称性と直線性(競合するため順不同)である。シンボル配置方法を以下に述べる。

Step 1: 上述のつづら折り法により、階層性を実現するようにモジュール  $M_i$  (サブモジュール  $SM_{ij}$ ) の仮配置を行い、各シンボルの  $X$  座標を決定する。

Step 2: 重心法<sup>11)</sup>の応用により、最少交差数を実現するように各同一  $X$  座標のシンボルの  $y$  方向の並び順序を決定する。

Step 3: 優先度法<sup>11)</sup>の応用により、近接性、対称性、直線性を実現するように各シンボルの  $Y$  座標を決定する。

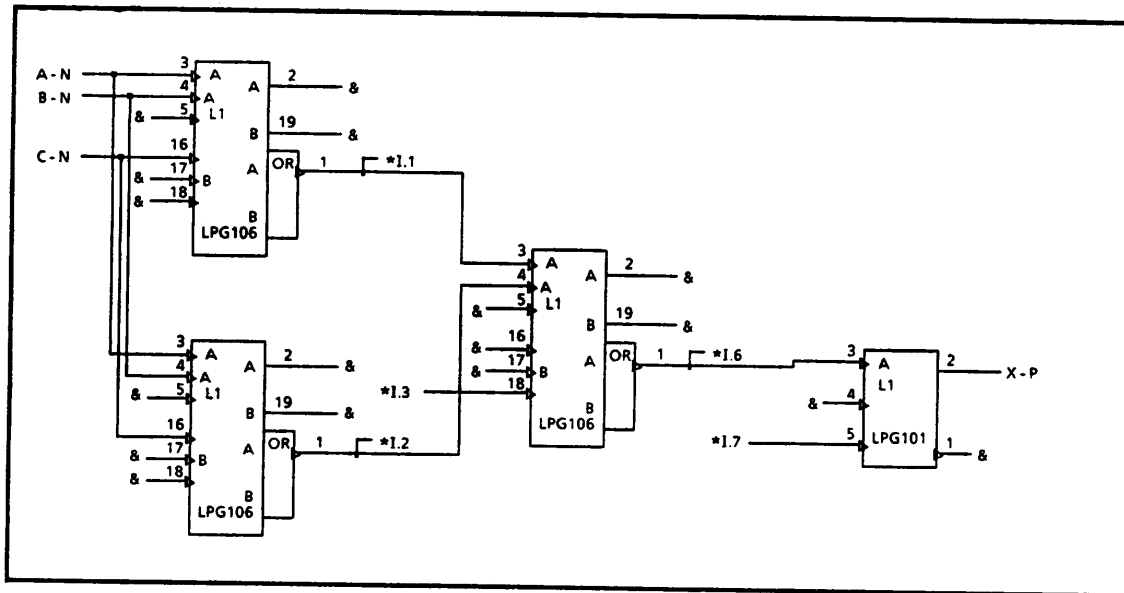
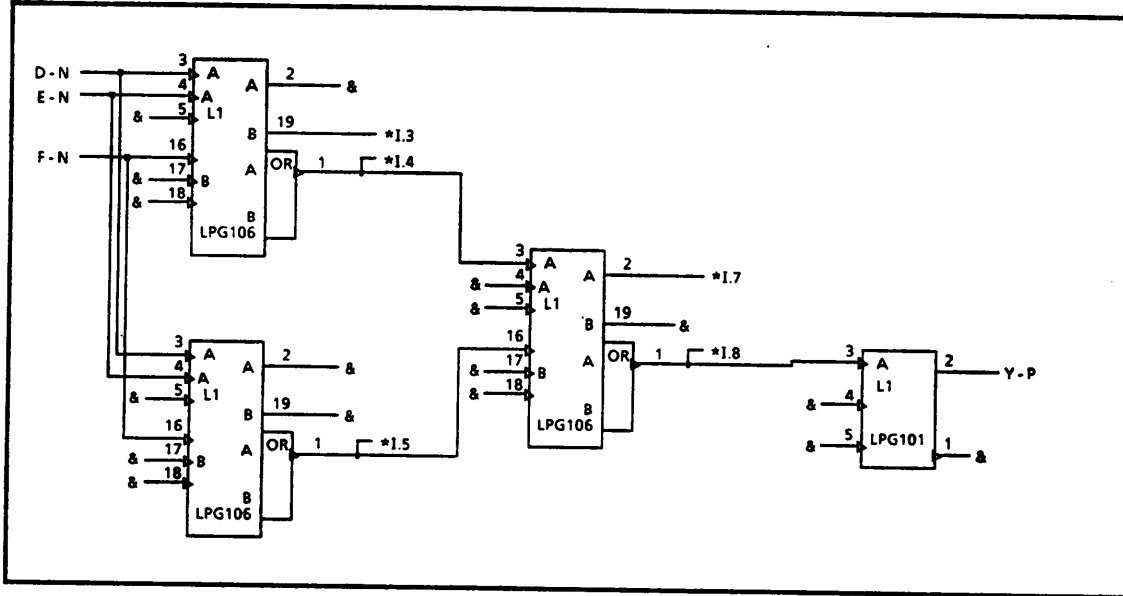


図 7 水平分割の適用例  
Fig. 7 A horizontal decomposition application.



重心法と優先度法は平面グラフ生成のシンボル配置方法である。これらの方法をゲート論理図生成に適用するために以下の機能拡張を行っている\*。

- (1) ゲート論理図作成基準に準拠している。
- (2) 平面グラフ生成の対象シンボルは同一サイズである。そのため、縦方向の長さの可変性を考慮して

いる。

- (3) 平面グラフ生成の対象シンボルの入出力端子数は各々1本である。これに対して、ゲート論理図生成の対象シンボルの入出力端子数は通常各々複数本である。そのため、最少交差数を実現するために、接続されるシンボル対の各入出力端子の相対位置(y座標)関係を考慮している。

- (4) つづら折りに配置される場合を考慮している。

\* プログラムレベルでは、重心法、優先度法の各基本機能と以下の拡張機能を一括して考慮し、これらの機能が容易に実現可能な方法を採用している。

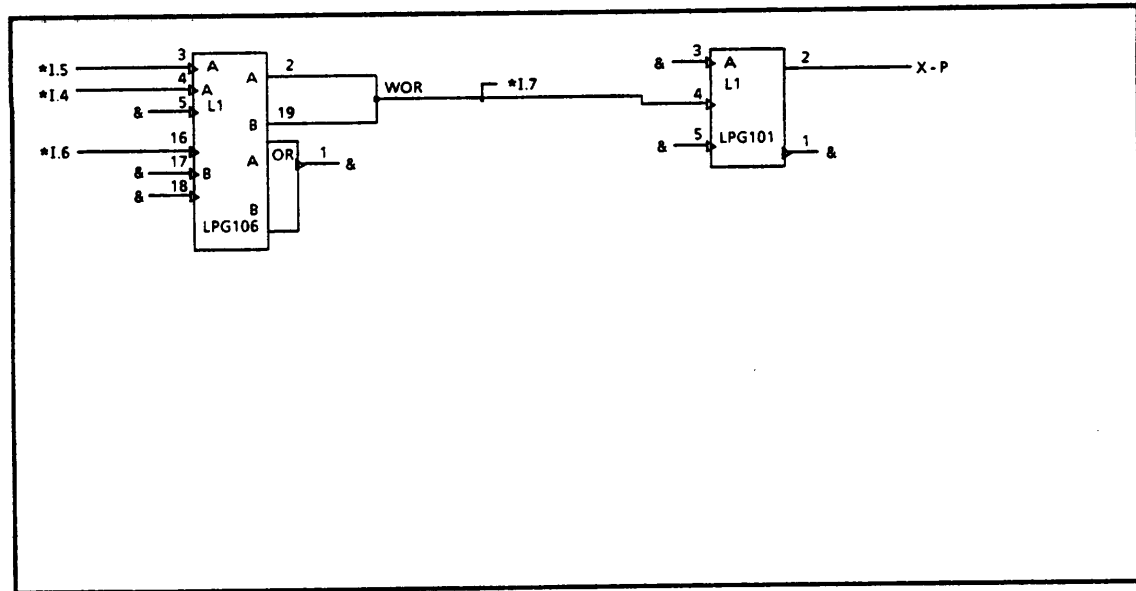
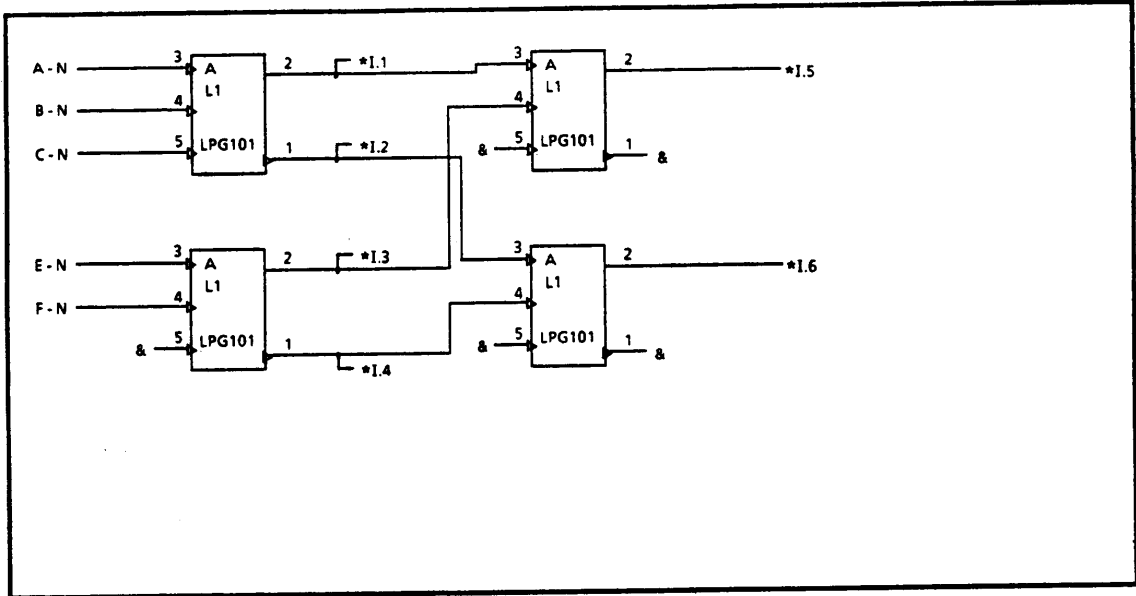


図 8 レベル間分割の適用例  
Fig. 8 A straight decomposition application.

(5) 結線論理の見やすさ<sup>15)</sup>を考慮している。

### 6. ゲート論理図生成例

代表的なゲート論理図生成例を図7~9に示す。上述のゲート論理図生成方法において、論理図面サイズはパラメータにより変更可能である。実際の論理図面はA3サイズであるが、これらの図は、論文のシートサイズに合わせて縮小した論理図面サイズを使用している。また、これらの図において、★で始まる信号名

の信号線はモジュールの内部信号線を表し、入力側シンボルまたは出力側シンボルと接続されていない内部信号線が切断信号線を表す。

### 7. 適用結果

上述のゲート論理図生成方法を超大型計算機 M68X の設計に適用した。本方法の中核をなすシート分割方法とシンボル配置方法の適用結果を以下に述べる。

(1) 適用対象は汎用モジュールであり、1汎用モ

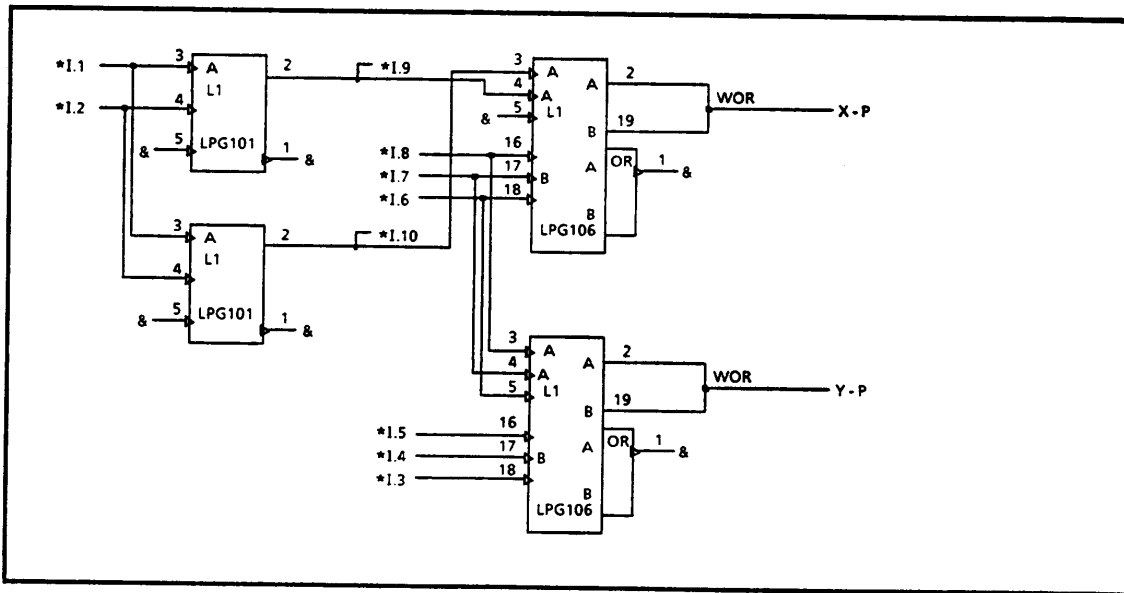
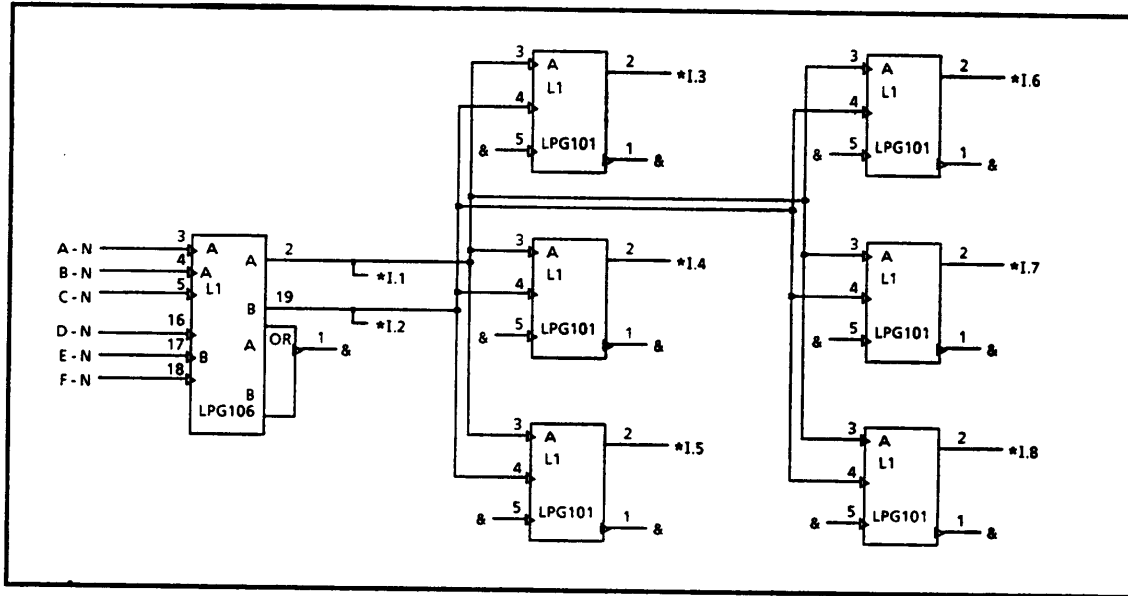


図9 レベル内分割の適用例  
Fig. 9 A zigzag decomposition application.

ジュールはA3サイズの機能論理図面1頁内に記述されるため、1汎用モジュールから生成されるゲート論理のシンボル数は高々200個である。

(2) 汎用モジュールのゲート論理図の見やすさは人手作成時とほぼ同等である。

(3) 1汎用モジュールのゲート論理図の平均シンボル密度は25~30個/頁であり、人手作成時とほぼ同等である。(1LSI全体では、十字枝法により空き領域が生じ、シンボル密度が低下するため、ゲート論理図面枚数は人手作成時と比べて5~10%増加する。)

(4) 1汎用モジュールのシンボル処理時間は300個以上/秒(on M200H)であり、実用上の問題はない。

## 8. むすび

見やすさの基準12項目を考慮したゲート論理図生成方法を開発し、超大型計算機M68Xの設計に適用した。

ゲート論理図の見やすさに関しては、当初の目標を十分に達成できており、設計者に受け入れられている。また、処理時間に関しても、現状の適用規模であれば十分である。今後は、高位論理生成技術の開発の一環として、対象シンボル数を1000個以上に拡大するために、シート分割方法の高速化を図る予定である。

最後に、論理図生成の実現方法はシンボル仕様に依存し、論理図の見やすさは、論理図自身の見やすさだけでなく、システムレベルの見やすさも考慮されなければならないことを付記しておく。

**謝辞** 本研究の機会を与えて頂いたシステム開発研究所の川崎淳所長と石原孝一郎部長ならびに神奈川工場の大野泰廣部長、また、本研究内容について有益な御意見・御討論を頂いた神奈川工場の土屋洋次主任技師と森田正人技師に深謝いたします。(所属は当時のもの)

最後に、本論文に対して有益なコメントを頂いた査読者に深謝いたします。

## 参 考 文 献

- 1) Berlig, D. E.: The Electronics Engineers Design Station, *Proc. of 17th DA Conf.*, pp. 422-428 (1980).
- 2) Ishii, M., Ito, Y., Iwasaki, M., Yamamoto, M. and Kodama, S.: Automatic Input and Interactive Editing Systems of Logic Diagrams, *Proc. of 18th DA Conf.*, pp. 639-645 (1981).
- 3) Ohno, Y., Miyoshi, M., Yamada, N., Odaka, T., Kozawa, T. and Ishihara, K.: Principles of Design Automation System for Very Large Scale Computer Design, *Proc. of 23rd DA Conf.*, pp. 354-389 (1986).
- 4) Tsuchiya, Y., Morita, M., Ikariya, Y., Tsurumi, E., Mori, T. and Yanagita, T.: Establishment of Higher Level Logic Design for Very Large Scale Computer, *Proc. of 23rd DA Conf.*, pp. 366-371 (1986).
- 5) 杉本, 兵頭, 堀内: 論理回路図の自動発生システム, 日本電気技報, No. 133, pp. 77-79 (1980).
- 6) 長谷川, 門倉, 深沢: 見やすさを重視した論理回路図作成システム, 電子通信学会論文誌, Vol. J68-D, No. 8, pp. 1513-1520 (1985).
- 7) Arya, A., Kumar, A., Swaminathan, V. V. and Misra, A.: Automatic Generation of Digital System Schematic Diagrams, *Proc. of 22nd DA Conf.*, pp. 388-395 (1985).
- 8) Weindling, M. N.: A Method for Best Placement of Units of a Plane, *Proc. of DA Workshop* (1964).
- 9) Breuer, M. A.: *Design Automation of Digital Systems, Vol. 1, Theory and Techniques*, Prentice-Hall, Inc., Englewood Cliffs, N. J. (1972).
- 10) Kani, K., Kawanishi, H. and Kishimoto, A.: ROBIN: A Building Block LSI Routing Program, *Proc. of Int. Symp. on CAS*, pp. 658-660 (1976).
- 11) Sugiyama, K., Tagawa, S. and Toda, M.: Methods for Visual Understanding of Hierarchical System Structures, *IEEE Trans. on SMC*, Vol. SMC-11, No. 2, pp. 109-125 (1981).
- 12) Mikami, K. and Tabuchi, K.: A Computer Program for Optimal Routing of Printed Circuit Conductors, *IFIP Congr. '68*, pp. 1475-1478 (1968).
- 13) Ohno, Y., Miyoshi, M., Kazama, Y., Tada, O. and Sakai, T.: Design Verification of Large Scale LSI Computers, *Proc. of Int. Symp. on CAS*, pp. 443-446 (1982).
- 14) Shinsha, T., Kubo, T., Hikosaka, M., Akiyama, K. and Ishihara, K.: POLARIS: Polarity Propagation Algorithm for Combinational Logic Synthesis, *Proc. of 21st DA Conf.*, pp. 322-328 (1984).
- 15) 男澤, 新舎, 碓谷, 植松, 久保: 見易い論理図のための結線論理シンボル配置方式, 第30回情報処理学会全国大会論文集, 6H-9, pp. 1973-1974 (1985).

(昭和63年10月12日受付)  
(平成元年6月13日採録)



**新倉 隆夫 (正会員)**

昭和 23 年生. 昭和 46 年横浜国立大学工学部機械工学科卒業. 昭和 48 年東京大学工学部産業機械工学科修士課程修了. 同年, (株)日立製作所システム開発研究所に入所. 現在, 第 3 部主任研究員. 論理装置の設計自動化に関する研究に従事. IEEE Computer Society 会員.



**碓谷 幸夫 (正会員)**

昭和 27 年生. 昭和 46 年新潟県立柏崎工業高校卒業. 同年 (株)日立製作所入社. 超大型機向け DA システムの開発に従事.



**男澤 康 (正会員)**

昭和 31 年生. 昭和 56 年慶應義塾大学工学部計測工学科卒業. 同年日立ソフトウェアエンジニアリング(株)入社. 以来コンピュータの設計自動化システムの開発に従事.



**久保 隆重 (正会員)**

昭和 17 年生. 昭和 40 年京都大学工学部数理工学科卒業. 昭和 42 年同大学院修士課程 (数理工学専攻) 修了. 同年 (株)日立製作所入社. 中央研究所にて汎用大型コンピュータの基本ソフトウェアの研究に従事. 昭和 55 年より同社システム開発研究所にて, コンピュータシステムの高信頼化技術, 論理設計自動化技術, 汎用コンピュータ方式技術に関する研究開発に従事. 現在同所第 3 部部长. ACM, IEEE, 電子情報通信学会各会員.