



プリント基板自動配線システム OSACA*

坂本 明雄** 千葉 徹** 白川 功** 尾崎 弘**
杉田 定嗣*** 栗本 卓治**** 西岡 郁夫****

Abstract

The recent technical advancements of printed wiring board design make the interconnection density become higher and the designing cost of a digital system has become more expensive than that of its components. Therefore, an efficient automated design system of printed wiring boards is expected. Motivated by this, in the present paper, we describe a software system for designing two-layer printed wiring boards, OSACA (Organized System for Automated Connection-routing Algorithm).

In OSACA, there are three different algorithms, each of which has the features to compensate the other's defects, for finding connection-routes among a set of pins to be made electrically common. The computing results for some examples with OSACA show its effectiveness, particularly with respect to the wiring rate.

1. ま え が き

回路の小型化, 高密度化のためプリント基板が広範に用いられるようになり, そのパターン設計の自動化の重要性はますます高まりつつある。

このパターン設計のうち特に配線問題に注目すれば, 従来から多くの手法が提案されているが^{1)~4)}, 個々の手法はそれぞれの特徴をもち, どの手法が最も適当であるかを判断することはできない。そこで本文では, それらのなかから3つの代表的な手法を選んで改良し, その特徴を生かすよう有機的に結合した一つの自動配線システム OSACA (Organized System for Automated Connection-routing Algorithm) を提案し, その概要と実験結果を報告する。

2. システム構成の概要

OSACA のシステム構成の概要を Fig. 1 (次頁参照) に示す。記憶容量は 384 K バイト, 使用機種は IBM 370/168 で, 使用言語は迷路法の一部でアセンブラを用いているが他はすべて FORTRAN である。

2.1 基板の分割

このシステムは, 2層プリント基板の配線を目的とし, 部品実装面をA面, 他方をB面と呼ぶ。基板全体を 1.27 mm (1/20 inch) 単位に区切って x, y の正整数座標 (原点(1,1)を基板の左下隅に置く) を設定し, これを f メッシュ (fine mesh) と呼ぶ。配線は, A面には x 軸に平行 (すなわち水平), B面には y 軸に平行 (垂直) な線分を引くことを原則とするが, 特別なプリアサイン線, 長さが1メッシュの線分および迷路法で生成される線分についてはこの限りではない。また, 両面にある線分を結ぶスルーホール, および部品取付穴は隣接する f メッシュに同時に存在してはならないが, 対角位置には存在可能とする。

実装部品をつぎの4種類に分類する。

(1) 標準 IC (以下では単に IC と呼ぶ): 14, 16,

* OSACA; A System for Automated Routing on Two-layer Printed Wiring Board by Akio SAKAMOTO, Toru CHIBA, Isao SHIRAKAWA, Hiroshi OZAKI (Faculty of Engineering, Osaka University). Sadatsugu SUGITA (Toyo Information System), Takuji KURIMOTO and Ikuo NISHIOKA (Central Lab., Sharp Corporation).

** 大阪大学工学部

*** 東洋情報システム(株)

**** シャープ(株)中央研究所

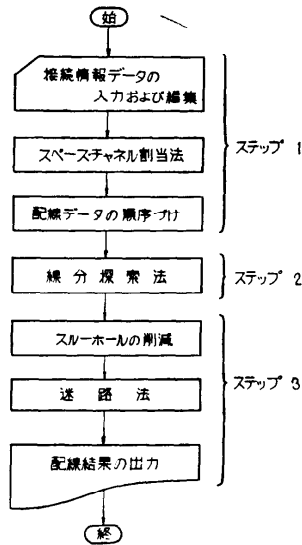


Fig. 1 Block flowchart of OSACA

18 ピンの dual-in-line package IC.

- (2) LSI: 上記以外の dual-in-line package IC または向い合う 2 つの端子列から成る部品.
- (3) 個別部品: 1 つの端子列から成る部品.
- (4) ターミナル: 基板の外部から電気的にアクセス可能な端子列で基板の端に位置するもの.

各 f メッシュをその座標を用いて (x, y) と表わし, 基板の右上隅の f メッシュを (x_{max}, y_{max}) とするとき*,

$$(x,) = \{(x, \alpha) | 1 \leq \alpha \leq y_{max}\}, \quad (1)$$

$$(, y) = \{(\beta, y) | 1 \leq \beta \leq x_{max}\}. \quad (2)$$

で定義される f メッシュの集合をチャンネルという. 必要に応じて (1) 式を垂直チャンネル, (2) 式を水平チャンネルと呼んで区別する.

いま基板全体を m 列 n 行の 2 次元配列状に分割し, それぞれの領域に属す f メッシュの集合をブロックと呼ぶ. ただし各列各行には等分にチャンネルを分配するものとし, 左から a 個目で下から b 個目のブロックを $\langle a, b \rangle$ と書く. さらに各ブロックを 3 列 4 行の領域に細分割してこれらの各領域に属す f メッシュの集合を s メッシュ (space mesh) と呼び, 基板全体の左から i 個目, 下から j 個目にある s メッシュを $[i, j]$ で表わす. このとき $1 \leq a \leq m, 1 \leq b \leq n$ なる a, b に対し

* 基板は矩形であるとし, x_{max} および y_{max} の値は次節で述べるパネル・データで指定する. ただし, プログラム上両者は共に 256 以下でなければならない.

て, 左から $(3a-1)$ 個目の s メッシュは 7 本の垂直チャンネル, 下から $(4b-2)$ 個目および $(4b-1)$ 個目の s メッシュはそれぞれ 6 本および 7 本の水平チャンネルを含み, それ以外には残りのチャンネルが配分される. このようにチャンネルを分配するのは, ひとつのブロックに 14 ピンの IC を搭載するとき, そのブロックの中央の 2 つの s メッシュに IC のすべての端子が位置するようにするためである (Fig. 2 参照). また 16 ピン (18 ピン) の IC の場合は, 一対 (二対) の端子が上 (上と下) の s メッシュにはみだして位置することになる.

いま $x_{max}=50, y_{max}=75$ とし $m=n=3$ の場合を考えてみよう. このとき $\langle 1, 1 \rangle$ は Fig. 2 のように 12 個の s メッシュに分割され,

$$\langle 1, 1 \rangle = \{(x, y) | 1 \leq x \leq 16, 1 \leq y \leq 25\},$$

$$[1, 1] = \{(x, y) | 1 \leq x \leq 4, 1 \leq y \leq 6\},$$

$$[1, 2] = \{(x, y) | 1 \leq x \leq 4, 7 \leq y \leq 12\},$$

⋮

$$[3, 4] = \{(x, y) | 12 \leq x \leq 16, 20 \leq y \leq 25\}.$$

となる.

さて, 次式で定義される集合

$$[i,] = \cup_{\alpha=1}^{a_i} [i, \alpha], \quad (3)$$

$$[, j] = \cup_{\beta=1}^{b_j} [\beta, j], \quad (4)$$

をスペースといい, 必要に応じて (3) 式を垂直スペース, (4) 式を水平スペースと呼んで区別する.

Fig. 3 (次頁参照) は f メッシュ, ブロック, s メッシュおよび水平, 垂直スペースを図式的に表わしたものである.

2.2 接続情報データ

- (1) パネル・データ: 基板の f メッシュ数とブロック数およびターミナルの位置を指定する.
- (2) ターミナル・データ: ターミナルの名称, 位

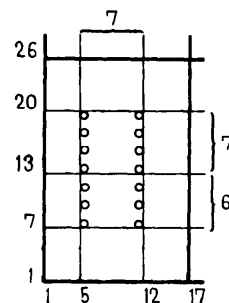


Fig. 2 Block $\langle 1, 1 \rangle$ and the pin position of a standard IC

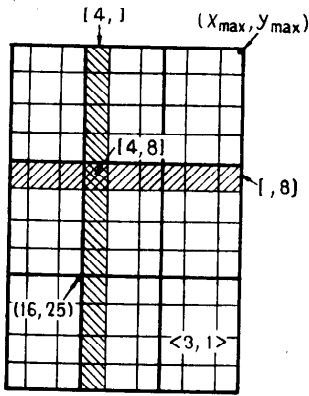


Fig. 3 An illustration for f-meshes, blocks, s-meshes and spaces with $x_{max}=50$ and $y_{max}=75$

置, 形状などの指定.

- (3) IC データ: 搭載する各 IC について, 各称, ピン数, 向きおよび位置 (ブロック名) を指定する.
- (4) LSI データ: 搭載する各 LSI を定義する. 位置は 1 番ピンの座標を, ブロック $\langle X, Y \rangle$ と相対座標 $^*(\bar{x}, \bar{y})$ を用いて指定する (Fig. 4 参照).
- (5) 個別部品データ: 搭載する個別部品を定義する. 位置の指定は前記 LSI の場合に準じる.
- (6) 禁止領域データ: 配線が禁止されているブロックを指定する.
- (7) プリアサイン・データ: 電源線やアース線など, 基板面上にあらかじめいくつかの線分をプリアサイン線としてここで定義することができる. この指定は s メッシュを用いて行う.

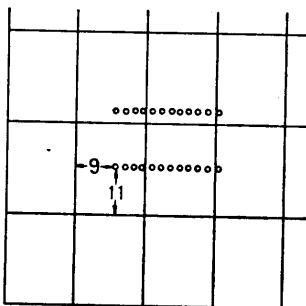


Fig. 4 Example of the position of an LSI, $\langle X, Y \rangle = \langle 2, 2 \rangle$, $^*(\bar{x}, \bar{y}) = (9, 11)$

* ブロック $\langle X, Y \rangle$ の左下隅の f メッシュを $(1, 1)$ としたときの 1 番ピンの f メッシュ座標.

- (8) 第 1 優先配線データ: 各プリアサイン線に結線する端子を, 部品名とピン番号で指定したデータ.
- (9) 第 2 優先配線データ: ターミナルの各端子に結線する端子を指定したデータ.
- (10) 第 3 優先配線データ: 上記 (8), (9) 以外の同一信号として結線されるべき端子を指定したデータ.

3. 配線アルゴリズム

3.1 スペース・チャンネル割当法^{5),6)} (space and channel assignment)

スペース・チャンネル割当法 (以下 SC と略記する) は, スペース割当とチャンネル割当とから成り, 前者は経路となる線分を生成して水平または垂直スペースへ登録するものであり, 後者は各スペースに登録された線分をそのスペース内のチャンネルに割当てるものである. この SC では, 一つの配線データの適当な 2 端子間の配線のみを行う (4.2 参照).

スペース割当の手順の概要は, 以下のようである. ただし, 下端が $[i, k]$ で上端が $[i, l]$ なる垂直線分を $[i,]_k^l$ と書き, 左端が $[k, j]$ で右端が $[l, j]$ なる水平線分を $[, j]_k^l$ と書く.

- 操作 1: $1 \leq i \leq 3m, 1 \leq j \leq 4n$ なるすべての i, j に対して, $H_{i,j} \leftarrow$ (水平スペース $[, j]$ に含まれるチャンネル数), $V_{i,j} \leftarrow$ (垂直スペース $[i,]$ に含まれるチャンネル数) とおき, 操作 2 へ.
- 操作 2: 一つの配線データ内の適当な 2 端子を選び, それらが位置する s メッシュを $[x_1, y_1]$ および $[x_2, y_2]$ とする.
- 操作 3: s メッシュ上で, $[x_1, y_1]$ と $[x_2, y_2]$ を結ぶ経路で, $H_{i,j} \geq 1$ なる $[i, j]$ を水平に通り, $V_{k,l} \geq 1$ なる $[k, l]$ を垂直に通るものを求める. 求めれば操作 4 へ, 求められないとき操作 5 へ.
- 操作 4: 求められた経路を構成する線分をそれぞれの所属するスペースへ登録し, 各線分 $[i,]_k^l$ ($[, j]_k^l$) に対して $V_{i,p} \leftarrow V_{i,p} - 1$ ($H_{l,p} \leftarrow H_{l,p} - 1$) とする. ただし $k \leq p \leq l$. 操作 5 へ.
- 操作 5: 取り扱うべき配線データが残っていれば操作 2 へ, 残っていなければ操作完了.

一方チャンネル割当は, 同一スペースに登録されている線分を最小個のチャンネルに割当てるものであり, この操作は各スペースごとに独立に行うことができる.

このチャンネル割当を行うことにより、 s メッシュ座標で定義されていた各線分が f メッシュ座標で定義されることになる。すなわち、垂直（水平）スペースに登録されている線分は、割当てられた垂直（水平）チャンネルの $x(y)$ 座標が軸座標となり、両端の $y(x)$ 座標はそれぞれの端点でスルーホールを介して連結されるべき水平（垂直）線分の軸座標で定まる。

3.2 線分探索法⁷⁾ (line search method)

線分探索法とは、指定されたいくつかの f メッシュ間を連結する経路を線分ごとの‘おおまか’な探索によって見いだそうとする接近法であり、これに対していくつかの手法が提案されている⁷⁾⁻⁹⁾。

本システムにおける線分探索法（以下 LS と略記する）は、基板上の2つの指定された f メッシュ間の配線経路を求める点对点ルーチン（以下 PPR と略記する）が基本となる⁷⁾。この PPR は、注目する2つの f メッシュ間に3本以下の水平線分との2本以下の垂直線分から成る配線経路が可能ならば、その一つを必ず見いだすという特徴をもっている。このような形状をした経路のみに限定することで、効率のよい探索アルゴリズムが可能になった。また、そのような経路が不可能な場合には、それがあつた程度まゝもって判別されるので、無益な探索操作を未然に防止することができる。

この PPR の応用として点对線ルーチン (PLR) があり、これは基板上の一つの f メッシュと線分群* が与えられたとき、その f メッシュから線分群上のどれかの1点への配線経路を求めることを目的とする。

3.3 迷路法¹⁰⁾ (maze routing method)

LS は前述のように効率のよい探索を行う反面、実際には配線可能であるにもかかわらず、その経路が見いだされないという場合が生じる。従つてそのような配線経路を最終的に探索して見いだす配線ルーチンが必要となる。

迷路法とは、いくつかの指定された f メッシュ間を連結する経路を f メッシュごとの探索によって見いだそうとする手法であり、最初 Lee¹¹⁾ により提案され、のちに Geyer¹²⁾ が多層プリント基板にも適用するべく拡張した。この手法の特徴は、配線経路が可能の場合には、その一つを必ず見いだすことにある。しかしながら、上記のいずれの手法においても2つの f メッシュ間の結線を目的としたものであり、そのまま

* 互いに連結された線分の集まりを線分群という。

** その配線データ内の端子でまだ線分群に含まれていないものを孤立端子という。

は3個以上の f メッシュ間の配線経路決定には不適である。本システムにおける迷路法（以下 MZ と略記する）は、2層プリント基板上のいくつかの指定された f メッシュ間を連結する配線経路が可能ならば、その一つを必ず見いだすことを目的とする¹⁰⁾。

本システムで MZ を適用する段階では、取り扱う配線データのいくつかの端子間に、すでに SC あるいは LS による配線経路が存在している場合が多い。そこで MZ では、いくつかの孤立端子** と線分群間の結線を行う問題としてとらえる。LS ではこの問題に対して孤立端子をひとつずつ取り扱い、線分群へ経路を順次求めていったが、MZ では複数個の孤立端子がある場合でも一つの配線データの結線を同時に行うところに特徴があり、その概要はつぎの通りである。

操作 1: 探索の出発点を孤立端子の中から適当に選び、出発点以外の孤立端子および線分群を目標として記憶する。

操作 2: 出発点を集合 S の元とする。

操作 3: S に含まれる各 f メッシュに対し、それに隣接する、 S に含まれない配線可能な f メッシュを新たに S に含める。この操作を繰り返して、すべての目標が S に含まれるか、新たに S に含むべき f メッシュがないとき次の操作へ。

操作 4: S に含まれた各目標から、操作 3 の探索順序の逆をたどつて出発点に至る経路を求める。

上記操作 3 で線分群上の一つの f メッシュが集合 S に含まれたとき、線分群を構成するすべての f メッシュを同時に S に含めることで孤立端子と同様に取り扱う。また各目標からの経路の逆探索（操作 4）は、最後に S に含まれた目標から行き、線分群上の一つの f メッシュに到達したとき、操作 3 で最初に S に含まれた線分群上の f メッシュからさらに逆探索を進めれば、出発点に至る経路が求められる。

4. 本システムにおける配線処理

4.1 配線データの処理順序

本システムでは配線データの処理順序について、グループによる順序づけと、同一グループ内の順序づけを考慮している。前者は、データ入力の際にユーザによってなされるもので、3つの配線グループを考える（2.2 参照）。後者は、それらのうち第2および第3優先配線グループに含まれる配線データ間の順序づけで

あって、SCによる配線が終了した時点でそれらの順序を定める。これは、LSでは配線の可否がその時点の基板の状態、すなわち他の配線データのための線分の分布に大きく依存していることによる。以下でその手法を考察する。

いま、配線データ i の孤立端子を p_1, p_2, \dots, p_{k_n} とし、各 p_j は部品 C_j に属し、かつそれは s メッシュ $[X_j, Y_j]$ の上にあるとする。このときつぎの3つの関数を定義する。

$$f_1(i) = \left| \bigcup_{j=1}^{k_n} [C_j] \right| + \alpha_i, \quad (5)^*$$

$$f_2(i) = (\max_j X_j - \min_j X_j) + (\max_j Y_j - \min_j Y_j), \quad (6)$$

$$f_3(i) = \left| X_0 - \frac{1}{2}(\max_j X_j + \min_j X_j) \right| + \left| Y_0 - \frac{1}{2}(\max_j Y_j + \min_j Y_j) \right|. \quad (7)$$

ここで α_i は、配線データ i の2つの端子がSCで結線されているとき1、そうでないとき0の値をとり、 $[X_0, Y_0]$ は基板中央の s メッシュ座標である。

この関数の値 $f_1(i)$ は、 i の配線を完成するために、SCで作られた線分群があればそれを一つの部品とみなしたとき、 $f_1(i)$ 個の部品間の配線を行わなければならないことを意味している。さらに $f_2(i)$ は、最も効率よく配線されたときの線分の長さの総和を s メッシュ単位で与えるものであり、 $f_3(i)$ は孤立端子の拡がりの中心と基板中央との直角距離である。

これらの関数を用いて、配線データ i に対して3つの実数の組を値とする $F(i) \triangleq (f_1(i), f_2(i), f_3(i))$ を対応させる。同一グループ内の2つの配線データ i, j に対して、 $F(i) > F(j)$ ** であるとき j を i の前に処理する、という条件でグループ内の処理順序を定める。いくつかの基板に対する実験の結果、この順序づけによって配線率に良い効果をもたらすことが分った¹³⁾。

4.2 各アルゴリズムの配線処理

本システムにおいて前述の3つの配線アルゴリズムが相互にどのような関連をもつかについて、実験結果をもとにして概説する。各アルゴリズムが処理する配線データは以下のものである。ただし、第 n 優先配線グループを G^n で表わす ($n=1, 2, 3$)。

* $|S|$ は集合 S の元の個数である。

** $A=(a_1, a_2, a_3), B=(b_1, b_2, b_3)$ に対して、つぎのいずれかの条件が満たされるとき $A > B$ と定義する。(i) $a_1 > b_1$; (ii) $a_1 = b_1$ かつ $a_2 > b_2$; (iii) $a_1 = b_1, a_2 = b_2$ かつ $a_3 > b_3$ 。

SC: G^3 内の各配線データについて、適当な2端子を選び、それら間の配線を行う。適当な2端子を選べないとき、その配線データについては何もしない。

LS: (1) G^1 内の各データを、入力された順に配線する。(2) G^2 内の各データを、前節で定められる順に配線する。(3a) G^3 内の各データでSCによって全く配線されなかったものについて、最も遠い直角距離にある2端子間の配線を行う。(3b) G^3 内の未完成の配線データについて前節の順序で配線を完了する。

MZ: SCおよびLSによって完成されていない配線データを、LSが処理した順序で経路探索する。

上記から明らかなように、本システムではLSが配線アルゴリズムの中心的な役割を果たしている。多くの場合、全体の90%前後の配線がここで実現されているが、これにおける配線処理についてまず考察する。

G^1 の各配線データに含まれる端子は、一般に基板全体に分布して、ユーザの指定した最寄りのプリアサイン線に通常1本の線分で配線される。もし基板上に多くの線分が存在している段階でこの配線を実行すれば、1本の線分のみでは配線できないことが生じる。一方、この配線を優先しても他の配線データの経路探索には直接の影響を与えない。

基板上で配線のためのチャンネルが不足する場所は、多くの場合、基板中央とターミナル付近である。特に外部端子(ターミナルの端子)からは一方向にしか線分が引けないため、 G^2 内の配線データを G^3 のそれより優先させる。SCではこの G^2 のデータは全く取り扱わないため、LSでは各データについて、まず最も遠い2端子を選んでPPR(3.2参照)で配線し、それを目標とみなしてPLRにより配線を完成する。一つの配線データについての探索が終了してから次の配線データの処理に移る。

G^3 の経路探索は、最初に各配線データの2端子間の配線を、SCでその配線が行われていないデータについて実行する。これは、基板上の線分の密度が高くなればPPRでの経路探索が困難になることを考慮するため、この段階で2端子間の配線を行っておけば、他の孤立端子との配線にはPLRを用いることができ、目標が線分群であるため基板が密であってもその経路が見いだされる可能性が大きい。

前述のように基板上の線分が密のとき、配線データが孤立端子のみであればその経路探索は一般に困難である。そこでLSの補助的な役割をもつ配線アルゴリズムとしてSCが用いられる。これは最初に適用され

る配線アルゴリズムであるため、基板にはブリアサイン線のみが存在し、しかもここで生成される線分は、チャンネル割当によって同時にそれらの位置が決定されるため、比較的長い線分も容易に作る事が可能である。また各スペースに登録された線分をできるだけ少ない個数のチャンネルに集中して割当てることで、LSのために全く線分が存在していないチャンネルを残しておくことができる。SC では全体の 10% 以下の配線しか行わないが、OSACA において SC を用いない場合、配線率および演算時間の点で悪い結果を招くことが多い¹³⁾。基板サイズが大きく、配線データ数が多いとき特にその傾向が顕著に表われる。これは G³ の配線データを LS で処理する場合、すでに線分群が生成されていれば、基板がある程度密になっていても効果的に配線がなされることを示している。

MZ は、未完成の配線データについて経路探索を行うが、LS でその経路が見いだせなかったのは、3.2 で述べた形状の解がないことであり、より複雑な形状の経路が存在する場合が多く、もし存在すれば MZ がその一つを必ず見いだす。特に配線の最終段階では基板上の線分の密度は非常に高く、このことは MZ における探索範囲が必要以上に広がらないという点からも、最後の配線アルゴリズムとして適当である。

4.3 スルーホールの削減

本システムでは、MZ を用いる直前にそれまでに生成された線分を調べてスルーホールの個数を減少させる処理を行う。ここでは一つの線分がその両端のメッシュ以外で他の線分と同一のメッシュを共有しないときに限り、その線分を逆の面上に移すことでスルーホールを削減する。この処理は単純であるが、通常 10~20% のスルーホールを削減する効果がある。

4.4 実験例

ここでは、Table 1 に示した 3 つの例題についての

Table 1 Input data for examples.

例題番号	例 1	例 2	例 3	
サイズ(メッシュ数)	100×135	230×120	234×125	
搭載部品数	ターミナル	5	4	4
	IC	19	34	29
	LSI	2	2	19
	個別部品	7	24	5
合計	33	64	57	
配線データ数	94	144	271	

* 日本 IBM のプリント基板総合設計プログラム。

Table 2 Results of example 1.

スルーホール数		373
削減されたスルーホール数		47
水平線分数		365
垂直線分数		384
	配線率	所用時間
SC	8.05 %	3.1 sec
LS	88.14	11.0
MZ	3.81	5.3
合計	100 %	19.4 sec

Table 3 Results of example 2 and 3.

例題番号	例 2		例 3	
システム名	APD	OSACA	APD	OSACA
スルーホール数	329	405	833	794
線分数	961	895	1,839	1,614
配線率	SC (%)	8.20	5.18	
	LS (%)	97.53	90.98	70.38
	MZ (%)	2.19	0.82	17.95
	合計 (%)	99.73	100	88.33

本システムの適用結果を示す。

Table 1 の例 1 の結果を Table 2 に、プロッタの出力図を Fig. 5 (次頁参照) に示す。Table 2 における SC, LS および MZ の所用時間は、それぞれ Fig. 1 のステップ 1, 2 および 3 のルーチンで要した演算時間を示した。したがってそれらの合計である 19.4 sec は必ずしも CPU time とは一致しない。

Table 1 の例 2 および 3 についてその結果を Table 3 に示す。例題の規模を知る目安として APD* の結果を併記した。ただしデータ入力形式に差異があるため両者の部品の配置には若干の違いがあり、特に例 3 については 24 ピンの LSI の配置を変更した。Table 3 の線分数は、水平、垂直線分の合計本数で、APD の配線率は Heuristic method および Maze method のそれを LS および MZ の欄にそれぞれ記した。OSACA の 3 つのステップの合計所用時間は、例 2 が 20.11 sec、例 3 が 43.50 sec であった。一方 APD では CPU time (IBM 370/165) が、例 2 で 27.72 sec、例 3 で 72.36 sec であった。これらの値は使用機種およびデータの取り方が違うため比較はできないが、少なくとも基板上の配線密度がある程度高くても OSACA はその特徴を発揮して効率よく配線すると考えることができる。

5. むすび

本文では、2層プリント配線基板の自動配線システ

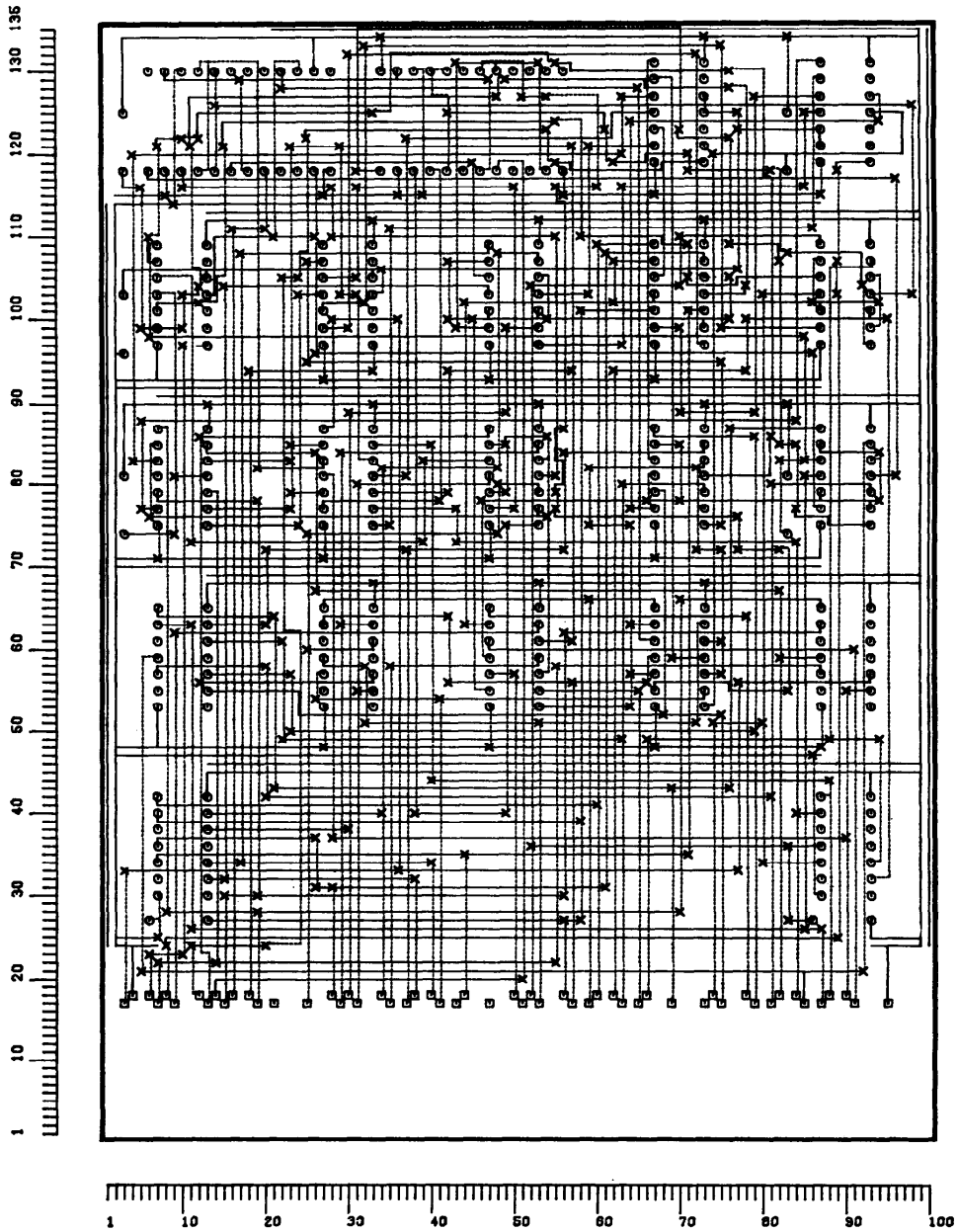


Fig. 5 Implemented result for example 1 (solid and dotted lines stand for conductor paths on layer A and B, respectively)

ムである OSACA について、その配線アルゴリズムの概要と、それらの配線処理について述べた。

OSACA は、LS を中心的な配線アルゴリズムとして持ち、SC および MZ がそれぞれの特徴を生かして LS を補助するという構成になっている。SC は、LS

が G^3 の配線データの配線を容易に探索できるように前もって 2 端子間の配線を行うという意味で LS を補助し、一方 MZ は、LS において配線不可能であった部分についてより詳細に再度探索するという意味で LS を補助している。このように 3 つの配線アルゴリ

ズムが互いにその特徴を生かすことにより、一つの自動配線システムとして OSACA はその有効性を発揮する。このことは、特にサイズの大きな基板上に高密度の配線が要求された場合その特徴が十分発揮され、最近のように高密度配線の自動化がますます要請されるに至っている今日、その実用的価値が高いものといえる。

最後に、本システムの製作過程において現 住友金属の山村春夫氏には線分探索法の手法の開発に従事して頂き、現 三菱重工の井手幹生氏には迷路法のプログラム化に従事して頂き、またシャープの三坂重雄氏には技術的な諸問題について御討論を頂き、さらには東洋情報システムの和田部長、林泰宏氏、および帝人の佐藤成生氏には計算機運用の過程で御助言、御協力を頂いたことに対し深く感謝の意を表する。

参 考 文 献

- 1) M. A. Breuer (ed.): *Design Automation of Digital Systems*, vol. 1. p. 420, Englewood Cliffs, N. J., Prentice-Hall, (1972).
- 2) H. Nakahara: Computer-aided interconnection routing: General survey of the state-of-the-art, *Networks*, vol. 2, no. 2, pp. 167~183, (1972).
- 3) D. W. Hightower: The interconnection problem - A tutorial, *Proc. Design Automation Workshop*, pp. 1~12, (1973).
- 4) H. C. So: Some theoretical results on the routing of multi-layer printed-wiring boards, *IEEE Inter. Symp. Circuit and Systems*, pp. 296~303, (1974).
- 5) 坂本, 山村, 千葉, 山下, 河田, 白川, 尾崎: 最適配置配線問題に関する一手法, 信学会, 回路とシステム研資, CST 73-70, (1973).
- 6) A. Hashimoto and J. Stevens: Wire routing by optimizing channel assignment within large apertures, *Proc. Design Automation Workshop*, pp. 155~169, (1971).
- 7) 山村, 白川, 尾崎: 二層プリント基板上の配線問題に対する線分探索の一手法, 信学論, vol. 57-A, No. 9, pp. 671~678, (1974).
- 8) K. Mikami and K. Tabuchi: A computer program for optimal routing of printed circuit conductors, *IFIP Congress 68*, pp. 1475~1478, (1968).
- 9) D. W. Hightower: A solution to line-routing problem on the continuous plane, *Proc. Design Automation Workshop*, pp. 1~24, (1969).
- 10) 千葉, 山下, 白川, 尾崎: 2層配線における迷路法, 信学会, 回路とシステム研資, CST 73-83, (1974).
- 11) C.Y. Lee: An algorithm for path connections and its applications, *IRE Trans.* Vol. EC-10, pp. 346~365, (1961).
- 12) J. M. Geyer: Connection routing algorithm for printed circuit boards, *IEEE Trans.* Vol. CT-18, pp. 95~100, (1971).
- 13) 坂本, 千葉, 井手, 白川, 尾崎, 山村, 杉田, 西岡, 栗本: プリント基板自動配線プログラム OSACA について, 信学会, 回路とシステム研資, CST 74-58, (1974).

(昭和50年6月2日受付)

(昭和50年11月25日再受付)