

# PCBパターンエディタのパソコン化

中城一明, 西山順啓, 日高博文, 上村裕明  
(沖電気工業株式会社)

## 1. まえがき

PCB(アリント配線板)の自動設計は、従来のバッチ型システムから会話型システムへと変化してきた。

それら会話型システムの多くは、大型計算機からミニコンクラスにグラフィック端末を接続した、高価格で高機能なシステムである。

しかし、これらのシステムは操作方法の学習が必要であり、又高価なため設計者一人ひとりが自由に使えるまでには至っていない。

今回開発したPCBパターンエディタは、パターン修正機能専用にすることにより、設計者が自由に使用できる設計ツールになることを目標としている。

本エディタはSIDE/boardと呼ばれ、低価格で大量導入の可能な当社パソコンiF800上で実現したので報告する。

## 2. システム概要

本システムは大量導入の可能なハードウェアであり操作上学習の必要無いことをめざし、パソコンによるパターンエディタの開発を行なった。

以下に本システムの開発課題を示す。

(1) ハードウェアは極力最小構成で実現し、オフラインで実行できること。

(2) 操作は学習を必要とせず、ユーザのレベル、ニーズに合わせて変更できること。

(3) 実用上問題のないレスポンスを確保すること。

(4) 大規模な基板にも適用可能なこと。

## 2. 1 構成

図1にシステム構成、図2にハードウェア仕様を示す。

パソコン側ハードウェア構成は当社iF800モデル50を使用し、これに1Mバイトのメモリ拡張とタブレットを加えた形を基本としている。

ホストとのデータ交換はRJE端末として同じくiF800を使用してある。

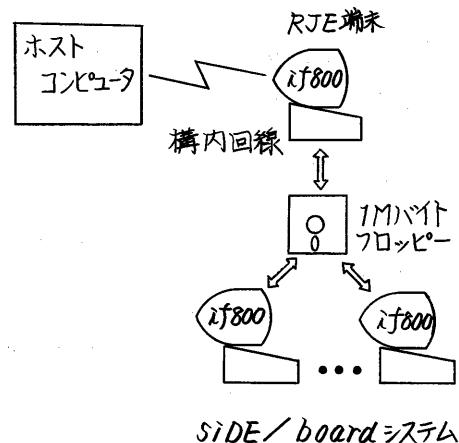


図1 システム構成図

項目	仕様	備考
CPU	8086 (8MHz)	
メモリ	1Mバイト	768Kバイト増設
CRT	カラ-640×475ドット	12インチ
タブレット	A3サイズ	RS232Cインターフェース
FDD	81/1Mバイト×2	

図2 ハードウェア仕様

## 2. 2 特徴

パソコン上にプログラムを開発するにあたり、操作性、レスポンス、基板規模の問題に対し本システムではパターン修正専用のシステムとすることを解決した。

次に本システムの特徴を示す。

### (1) リアルタイムパターンチェック

パターンの修正作業に伴ない発生する間陳エラー、接続ミスを即座に検出しマ表示する。

### (2) 高速図形処理

全データをメモリ内に格納し、階層化領域分割方式による図形管理を行なうことで、各処理において十分なレスポンスを確保している。

### (3) 大規模基板対応

必要最小限のデータをコンパクトに持つことにより  
基板サイズ 80cm×80cm  
基板層数 12層  
パターン 2万線/分  
部品ピン 5千ピン  
まで対応できる。

## (4) システムのカスタマイズ

ユーザのニーズによりメニュー構成、画面構成、メニュー構成処理制御等を自由に外部より変更できる。

### 3. プログラム構成と制御

本システムは将来への拡張、移植及びユーザ要求の対応が容易に行なえることを考えて、階層化とモジュール化された構成となっている。

プログラムの階層は図3に示す様に3階層、7部から成り立っている。

第1階層はユーザインタフェースとプログラム制御を行なう。

第2階層はパターン修正専用であり機能毎にモジュール化されているため追加変更が容易にならでいる。

第3階層は共通に使用するモジュール群であるが、特にエ／オモジュール部は、ハードウェアに依存する部分である。

### 3. 1 構成

図4はユーザインタフェース部、マスターコントローラ部、機能モジュール部、図形操作部、テーブル操作部、及びエ／オモジュール部の各部について関連を示すものである。

共通モジュール部は各部から共通に呼び出されるので省略してある。

## (2) マスタコントローラ部

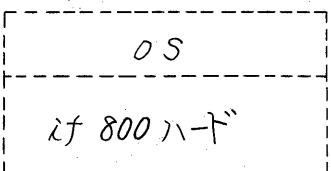
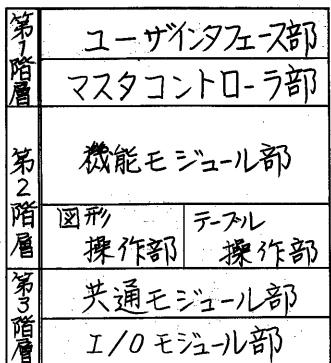


図3 プログラム階層

各種コントローラデータを参照し、又入力されたコマンドを解析し、必要に応じて各機能モジュールを起動する。

## (3) 機能モジュール部

パターンの追加、削除、図形の拡大、縮小等コマンドに対応する形で独立して作成されており起動された時の状態に応じて、それぞれの機能を実行する。

## (4) 図形操作部

機能モジュール部から指示パタメータにより図形の拡大、縮小等を行ない画面表示を行なう。

## (5) テーブル操作部

パソコン上に取り込まれた基板設計データに対し、参照、追加削除等を行なう。特に任意の図形領域内の図形参照には、階層化領域分割方式を考慮し図形データの高速検索を実現している。

## (6) 共通モジュール部

各部より呼びだされ、ハードウェア、パターン修正機能等に依存しない機能を実行する。

## (7) I/Oモジュール部

ハードウェアに依存し、デバイスエ/オに関係する機能を実行する。

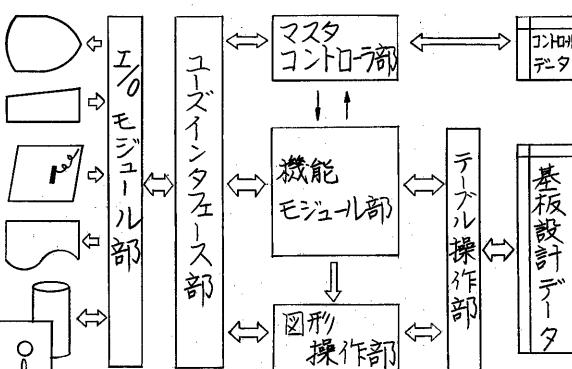


図4 プログラム構成

次に各部の主な機能について述べる。

### (1) ユーザインタフェース部

操作者に対し、図形、メニュー、メッセージの表示と、操作コマンドの入力を行なう。座標値については実座標と論理座標との変換を行なっている。

### 3. 2 制御

プログラムの制御方法についてパタン追加中に画面拡大を行なう場合を例に説明する。

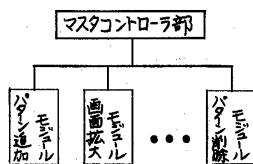
この場合、使用する機能モジュールにはパターン追加、画面拡大モジュールであり図5に示す構成となっている。

実際のメニュー構成は図6のように階層的になつてゐるため、マスターコンピュータ部で以下のように制御する。

今、パターン追加モジュールが実行中である。 図7

- ① ユーザからの画面拡大コマンドはユーザインターフェース部を通り、実行中である追加モジュールへ渡される。
  - ② 追加モジュールでは、このコマンドを割込コマンドとしてコマンドスタックへ格納する。
  - ③ 次に画面拡大処理が終了後に再び追加モジュールに戻るため、パターン追加コマンドをコマンドスタックへ格納する。
  - ④ 現在のパターン追加処理を終了する。
  - ⑤ マスター制御部ではコマンドスタックが空でなければ、ユーザへのコマンド入力要求を出さず、コマンドスタックに格納されている画面拡大コマンドを実行する。
  - ⑥ 拡大コマンド実行後、再びコマンドスタックの内容よりパターン追加モジュールが起動される。

このように本システムではコマンド内より任意のコマンドを自由に実行ができるので、メニュー構成を容易に変更できる。



### 図5 プログラム構成

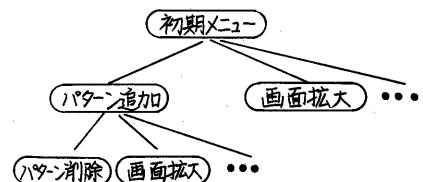
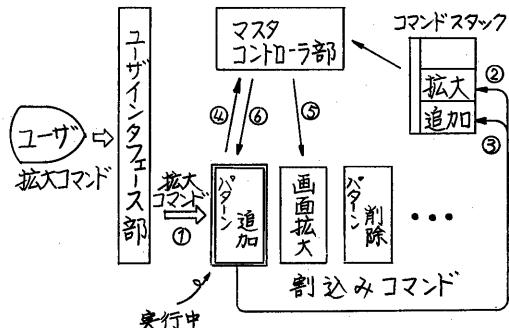


図6 メニュー構成



## 図7 制御7口一

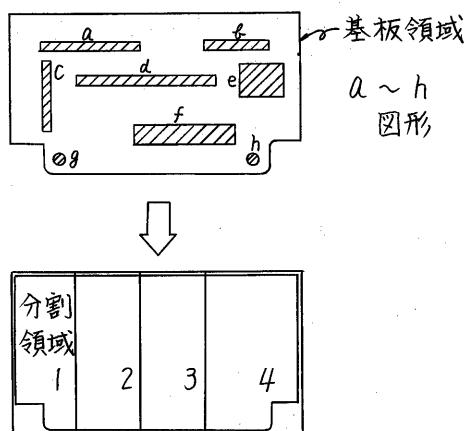
#### 4. 階層化領域分割方式

会話型システムでは、大量の図形データを高速に取扱う事が求められている。

本システムでは階層化領域分割と呼ぶ図形データの管理方法を考案し、高速化を達成している。

階層化領域分割方式は、領域分割方式（図8）で問題となる領域間にまたがる図形データの取扱いを解決したシンプルな管理方法である。

本方式の基本概念は図9に示す様に領域を階層的に分割して行くものであり、図形データはその図形を包含する最小の領域に登録、管理される。



領域	領域内図形
1	c (a) g
2	(a) d (f)
3	(d) f
4	f e (f) h

○は領域間にまたがる図形

図8 領域分割方式

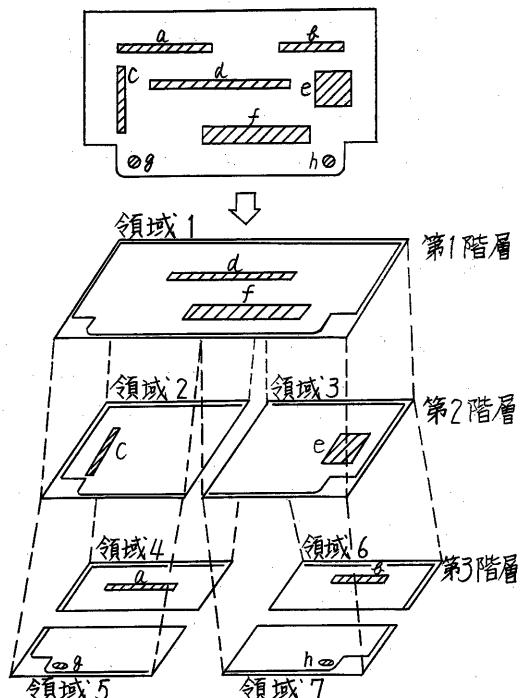


図9 階層化領域分割方式

領域	エリア	階層番号	図形枚数
1		1	4
2	対	2	3
3	角	2	5
4	座	3	1
5	標	3	7
6		3	2
7		3	8

図形データ	領域番号
a	0
f	0
c	0
d	6
e	0
f	0
g	0
h	0

図10 領域管理テーブル

階層化領域分割方式は次の3つの処理より構成される。

##### 第1処理

ここでは基板外形寸法より作業領域を求め第1階層を設定する。次に第1階層の領域1を領域の長辺で2分割して第2階層の領域2、領域3を設定する。

この様に各階層の領域について分割をくりかえし、最大6階層63領域を設定する。

各領域は領域管理テーブル(図10)にその領域の対角点座標と階層番号を格納する。

## 第2処理

次に基板上の図形データ(パターンについては線分単位)を領域管理テーブルを参照して包含する最小領域を求める。求められた領域の管理テーブルに図形データのポインタをリンクすることで図形の登録を行なう。

このポインタ付け処理はパターン修正の前処理で全ての図形に対して行なわれるが、パターンの修正中に生じる図形の追加、削除では該当図形に対してのみ行なわれる。

## 第3処理

最後は登録された図形の参照であり、任意の領域内に存在する図形を求める処理である。

画面拡大処理を例にとって説明する。

図11に示す斜線領域を拡大しようとすると、その領域と重なる階層化領域(図10の領域1、3、6)を領域管理テーブルの対角座標より求める。

次に、求めた階層化領域に登録されている図形をポインタによりたどり、図形データ(図10、d、f、e、g)を求める。

これらの図形と拡大する領域との重なりにより表示する図形(図10、d、e、g)が求められる。

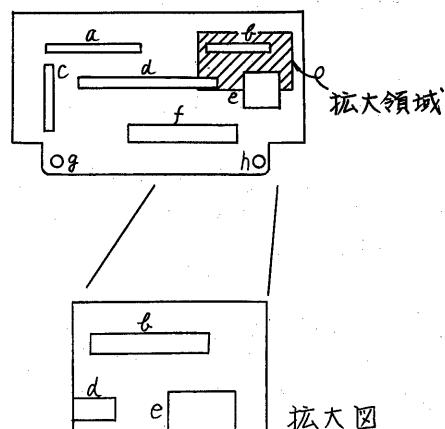


図11 画面拡大例

この方式での画面拡大レスポンスを示す。

表1

図形数	レスポンス
約7000	1~2秒
約14000	2~4秒

## 5. パターン修正機能

本システムで実現している主な修正機能を示す。

### (1) 追加

ぬりつぶしパターンについては矩形で入力する。

### (2) 削除

線分、区間、系列単位で行なう。

### (3) 層変更

パターンの線分単位に任意の層に移動させる。

(4) 線幅、スルーホール形状の変更

(5) 配線層自動切り替え

パターン追加時に層構成を参照して、自動的に層を切り替える。

(6) スルーホールの自動追加削除

修正によって必要又は不要になったスルーホールは自動的に処理される。

(7) パターンチェック

修正に伴なう、パターン、スルーホール、部品ピンとの接続ミス、配線禁止エリアへの接続ミス、及び間隙不良を即座に検出して画面上にプリント表示する。

(8) 未配線表示

論理的な接続があるにもかかわらず配線パターンが存在しない場合未配線表示されるが、その表示区間はパターン修正によって異和感なく更新していく必要がある。

本システムでは画面上に表示されている图形を未配線表示のターゲットとして優先することにより未配線表示区間を更新している。

(9) 信号名サーチ

信号名からパターンを、もしくはパターンから信号名を求める。

## b. 評価

SIDE/boardはパソコンを使用した実用的なパターンエディタとして満足のいくものである。実基板の適用例を図12～14に示す。

(1) 操作性

ユーザに合わせてメニューを変更できるため操作性は好評である。スパターンエディタ専用にしたことによりコマンドが少なくなったことも良い結果となっている。

(2) レスポンス

多くのコマンドで良好なレスポンスがあり、十分に実用に使用できる。

(3) 規模

IC150個を搭載した基板でもメモリ容量の約1/2を残しているので実用上問題は無い。

本システムは高密度基板を設計するユーザにとって特にリアルタイムのパターンチェック機能は有効であり、ホスト上にある詳細なパターンチェックでエラーとなることが少なくなった。

〈仕様〉

基板名	層数	サイズmm	IC数	系列数	印線数	部品点数
A	4	390X150	79	409	4858	1931
B	4	300X225	150	592	1021	3192
C	4	300X305	161	701	10893	3420
D	6	280X190	91	315	3490	2138

〈レスポンス〉

基板名	拡大	間隔検索	↓	追加	↓	削除
A	1~2秒	0.5~1秒	↓	0.5~1秒	↓	0.5~1秒
B	2~4秒	0.5~1秒	↓	0.5~2秒	↓	0.5~1秒
C	2~4秒	0.5~1秒	↓	0.5~1秒	↓	0.5~1秒
D	1~2秒	0.5~1秒	↓	0.5~1秒	↓	0.5~1秒

図12 適用例

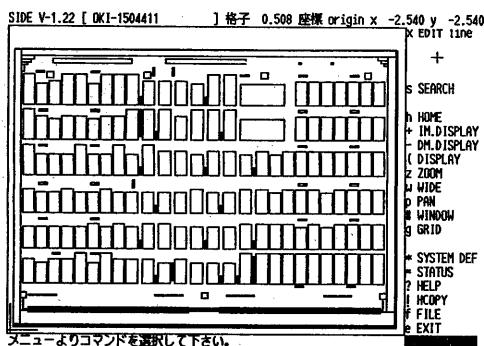


図13 例 1

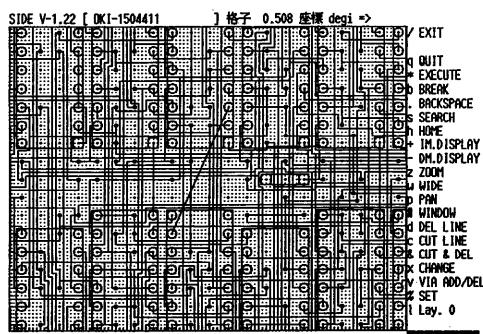


図14 例 2

## 7. あとがき

今回はPCBパターンエディタを作成したが、今後は部品配置、回路図入力をサポートする機能へも展開していく予定である。

なお、本プログラムの開発に当り御指導、御協力戴いた、技術本部 植村GL及び当部 中田部長、上田次長、原田課長 他関係者各位に感謝致します。