

自動レイアウト用

5U-4

ライブラリデータ作成プログラム

佐野葉子 松田庸雄

日本電気 株式会社

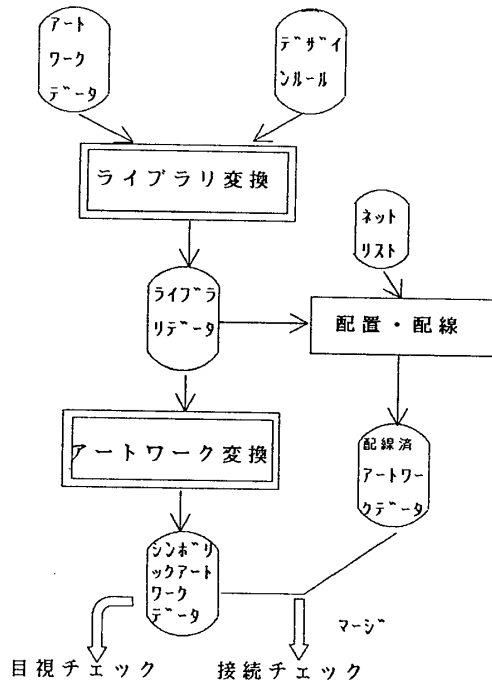
1. はじめに

近年のLSIの高集積度化に伴う設計自動化の浸透において、エラーフリーなセルライブラリの自動生成や、他システムとのインターフェースをとるためのライブラリ記述の規約が必要である。本稿では、自動レイアウトシステムPYRAMID<sup>[1]</sup>において、他システムと共通化可能なアートワークデータ記述の規約を設定し、本記述を元にライブラリデータ自動生成、及び出力機能を作成したのでこれを報告する。

2. プログラムの概要

本プログラムの構成は、大別して2つの部分より成る。第1図は、本プログラムの構成図である。

- 1) アートワークデータ→セルライブラリ (ライブラリ変換)
- 2) セルライブラリ→疑似アートワークデータ (シンボリックアートワーク変換)



第1図 プログラム構成図

3. ライブラリ変換

PYRAMIDは、ブロックライブラリに、

1. ブロック外形情報
2. 端子情報
3. 配線通過禁止情報
4. 固定クランプ情報

が必要である。ライブラリ変換処理部では、デザインルールと実際のアートワークデータより、上記情報を持つライブラリデータを作成する。アートワークデータのライブラリデータへの変換則は、アートワークデータのレイヤー、並びに、データフォーマットへの指定をパラメータとして変換プログラムに与えることによって定義する。以下に上記情報の作成方法、及び、データフォーマットを示す。

3-1. ブロック外形情報

指定されたアートワークレイヤーの矩形データをブロックの外形とする。

3-2. 端子情報

端子情報として、以下のものを自動認識する。

- a. 名前
- b. 層
- c. 属性 (入出力電源別)
- d. 位置
- e. 端子幅 (電源端子のみ)

a. ~ c. の認識 テキストから識別できるようにテキスト列の文字位置に次のような意味をもたせた。

先頭1文字目→端子テキスト識別ヘッダ

- 2文字目→属性
- 3文字目→端子層
- 4文字目以降→端子名

【ex.】Z12A1: 端子名“A1”、入力、2層

本プログラムは、パラメータとして、テキスト列の各位置における識別文字を与えられることにより、この情報を得る。

d. の認識 次の2つの方法をサポートしている。

- ① 端子テキストと認識されたテキストの座標を採用。
- ② 端子テキストの座標を含む配線データと等電位のデータを抽出し、そのうち、外枠と接する箇所を全て端子位置とする。

e. の認識 PYRAMIDは、電源端子のみ、幅を許している。d. で示す方法により、テキストが付加され、これを含む配線データと等電位のパターン、外枠に接する辺長を端子幅とする。

### 3-3. 配線禁止領域情報

PYRAMIDのルータは、ブロック上で配線可能であるため、ブロック上でのショートを防ぐべく、配線ショートを起こすアートワークデータを禁止領域として読み取る必要がある。このルータは、ブロック上のみグリッドルータであるため、ブロック上にグリッドを設定し、クリアランスを考えながら、そのグリッド上の配線通過の可否を判定する。

### 3-4. 固定クランプ情報

固定クランプは先頭3文字を識別子、それ以降を端子名、と定義した特殊なフォーマットのテキストで示される。クランプ位置はテキストの座標より得る。本プログラムは、識別子により、ハイ/ロー別、及び、クランプセル名を認識する。

## 4. シンボリックアートワーク変換

本プログラムの目的は、2つある。

1. ライブラリデータの目視チェック
2. 接続チェックプログラム<sup>[2]</sup>とのインターフェース

本プログラムは、3. で示したPYRAMIDのライブラリとしての必要情報(アートワークのフォーマットとしては、テキスト及び矩形)に、目視用(接続チェック用)の端子位置図形を加えたアートワークデータを出力する。以下に、各使用方法を述べる。

### 4-1. ライブラリデータの目視チェック

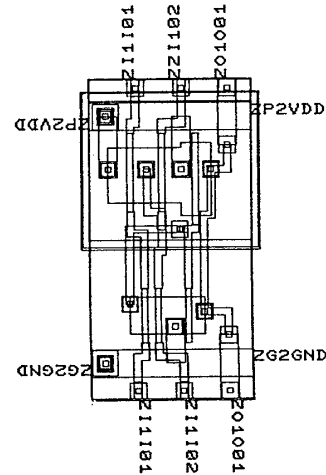
本プログラムの出力をアートワークエディタ等にロードすることにより、ライブラリデータの目視チェックができる。特に、端子用テキストに関しては、もとのアートワークデータになく、PYRAMID実行用に付加するため、人為的ミスが起こり得る。このようなエラーを早期発見し、後の配置・配線プログラムの誤動作を避けるためにも、実際入力されたライブラリのチェックは、必要である。

### 4-2. 接続チェックプログラムとのインターフェース

本プログラムの出力と配線プログラムによるレイアウト結果をマージすることにより、ブロックレベルの接続チェックが可能である。本手法は、ライブラリ変換プログラムの入力となるアートワークデータを使用して、それをトランジスタレベルにまで展開してチェックする場合に比べ、はるかにTAT短縮できる。

## 5. データ例

第2図にライブラリ変換の入力のアートワーク図、第3図に変換されたライブラリの一部、第4図にシンボリックアートワーク変換の結果を示す。



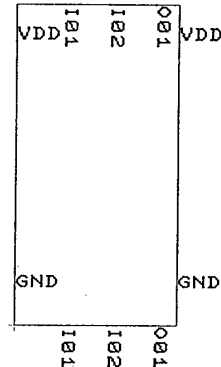
第2図 入力アートワークデータ

```

1000EX1        01B  12/10/8612/10/86        1
1100EX1        OVLAAD00A        3        5
1120        0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
1200R        1 0 0 0 0 350 700
1301        1 2 3
1401        1VDD        0. -1P        0. 0. 0. 0 0 2
1410        0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
1500        1 2 OX        0. 576 0. 0. 576 0. 664
1500        2 2 OX        350 576 350 576 350 664
1401        2 2GND        -1P        0. 0. 0. 0 0 2
1410        0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
1500R        1 2 OX        0 0 36 0. 36 0. 0. 124
1500        2 2 OX        350 36 350 36 350 124
1401        3I01        -1I        0. 0. 0. 0. 0. 0. 0. 0. 2
1410        0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
1500        1 1 OY        100 0 100 0 100 0 100 0
1500        2 1 OY        100 0 100 0 100 0 100 0 700
1401        4I02        -1I        0. 0. 0. 0. 0. 0. 0. 0.
1410        0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
1500        1 1 OY        200 0 200 0 200 0 200 0
1500        2 1 OY        200 0 200 0 200 0 200 0 700
1401        5001        -10 0. 0. 0. 0. 0. 0. 0. 0. 2
1410        0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
1500        1 1 OY        300 0 300 0 300 0 300 0
1500        2 1 OY        300 0 300 0 300 0 300 0

```

第3図 ライブラリデータ



第4図 シンボリックアートワークデータ

## 5. まとめ

本稿では自動レイアウトシステムPYRAMIDにおける、ライブラリ記述の規約に基づいた、ライブラリ自動生成、及び出力機能について報告した。これにより、ライブラリ入力の簡素化・エラー減少、及び、他システムとのインターフェースが容易に図れた。

### [参考文献]

- [1] 松田 他 第35回情報処全大、5H-1、1987
- [2] 原田 他 第35回情報処全大、7H-4、1987