

# 双金属层门阵列跨单元行布线问题与算法

应昌胜 洪先龙

(清华大学计算机系,北京, 100084)

1991年4月28日收到, 1992年3月20日修改定稿

在双金属层门阵列布图中, 跨单元行的走线可以直接在单元上进行, 这些单元被称为“走线块”。充分地利用走线块上的两层走线空间, 可以降低通道密度并减少通孔数。本文给出走线块布线问题及求解算法。包括三个过程: 首先确定走线块引线端的相对位置关系, 其次是定位引线端, 最后实现走线块内的连接。给出的算法已经用C语言实现并嵌入一门阵布图系统。实例运行结果表明提出的算法可以有效地降低通道密度和提高通道布通率。

CCACC: 7410D, 5120

## 一、引言

在单层金属门阵列布图中, 通常是在单元行内间隔地嵌入冗余通道(Feed-through)以实现跨越单元行的走线, 各冗余通道的宽度随门阵母片设计的完成而确定, 在冗余通道内的走线是单层的, 且只走垂直线。给定一个门阵母片, 其上的冗余通道位置与尺寸是固定的。而在双金属层门阵列布图中, 跨单元行的走线可以直接在空单元上进行, 不需要在母片上设置固定的冗余通道, 我们称单元行上用做跨单元行走线的各区域为“走线块(Wiring-tile)”, 走线块的宽度根据总体布线的结果确定, 调整布局就可以改变走线单元的位置和宽度。到目前为止发表的门阵列自动布图系统(包括双金属层门阵布图)都是只采用垂直走线段

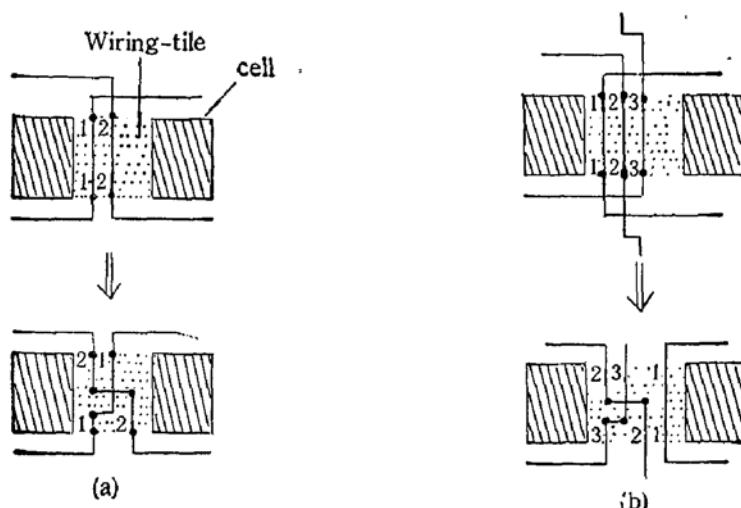


图1 走线块引线端的位置影响通道密度

实现跨单元行的线网连接，还没有见到有线网跨单元连接在单元行内既走垂直线又走水平线的算法或系统发表。而有效地利用走线块内的水平走线空间，显然可以降低通道密度和提高通道布通率，如图 1 所示。

在我们开发的双层金属门阵设计系统 MALS2<sup>④</sup> 中，引入并实现了一个称为“走线块引线端优化与走线块布线”的过程，它做为通道布线的预处理安排在通道布线之前进行，旨在降低通道密度提高通道布通率。

## 二、基本定义

给定一个走线块  $\Phi$ ，定义它的宽度为  $w_\phi$ ，高度为  $h_\phi$ ， $w_\phi$  为  $\Phi$  包含的垂直走线道数， $h_\phi$  为  $\Phi$  内可用的水平走线道数（它是一个固定值，与基本单元的高度和电源、地线的位置相关）。穿越  $\Phi$  的线网（由总体布线确定）记为  $N_\phi = \{n_i | i = 1, k\}$ ，其中  $k$  为穿越的线网数。每个线网  $n_i$  对应  $\Phi$  上下边上的两个引线端  $t_i$  和  $b_i$ ， $t_i$  位于  $\Phi$  的上边， $b_i$  位于  $\Phi$  的下边；相应地， $\Phi$  上下边上的引线端可用两个线性表表示：

$$T_\phi = \{t_i\} \text{ and } B_\phi = \{b_i\}, i = 1, n.$$

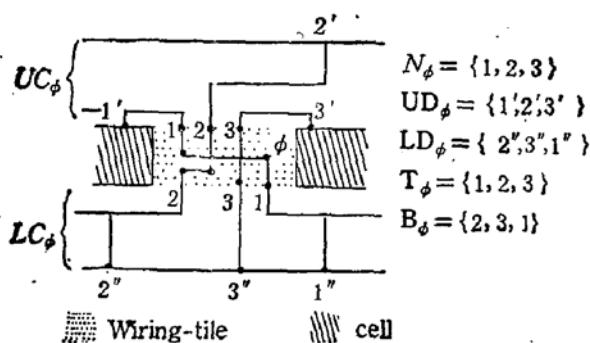


图 2 走线块上的基本定义

$\Phi$  的上下边分别落在两个通道内，它们是  $\Phi$  的上通道  $UC_\phi$  和  $\Phi$  的下通道  $LC_\phi$ 。为表示线网进入或退出  $\Phi$  的方位，分别在  $UC_\phi$  和  $LC_\phi$  中为  $N_\phi$  中各线网  $n_i$  定义一个“上目标”  $ud_i$  和一个“下目标”  $ld_i$ 。 $ud_i$  表示在上通道  $UC_\phi$  中与  $t_i$  有互连关系且距离  $t_i$  最近的引线端， $ld_i$  表示下通道  $LC_\phi$  中与  $b_i$  有互连关系且距离  $b_i$  最近的引线端；换

言之， $ud_i$  最可能与  $t_i$  相连， $ld_i$  最可能与  $b_i$  相连。相应地定义了以下两个集合：

$$UD_\phi = \{ud_i\} \text{ and } LD_\phi = \{ld_i\}, i = 1, n.$$

$\Phi$  边上的引线端位置是可变的，在它的位置被确定之前假定它落在  $\Phi$  上/下边的中心。各定义如图 2 所示。

总体布线之后，各线网的路径可以描述为它所经过的通道段和走线块的一个序列。对每个走线块  $\Phi$  而言，跨越它的线网集合  $N_\phi$  已经确定，要求确定各线网对应的引线端位置及在走线块内的连接。问题分三步求解：引线端定序、引线端定位和走线块布线。

## 三、走线块引线端定序

引线端定序的目标是使引线端的安置顺序有利于减小通道内的布线密度和通孔(via)数。基本策略是：分配从左边进入或离开走线块的线网的引线端在走线块的左端头，从右边进入或离开走线块的线网的引线端在走线块的右端头；这样的安置顺序能够避免这些线网在通道中发生交叉。走线块上下边引线端定序是分别进行的。每个线网，

$n_i$ , 对应的两个引线端  $t_i$  和  $b_i$ , 它们在上/下边上所处相对顺序的可以是不同的, 例如,  $n_i$  在  $UC_\phi$  从  $\Phi$  的左边进入  $\Phi$  而在  $LC_\phi$  从  $\Phi$  的右边离开  $\Phi$ , 合理的安置是  $t_i$  在  $\Phi$  上边的左端而  $b_i$  在  $\Phi$  下边的右端。

上下边上引线端定序算法是对称的。不仿以上边为例, 如图 3, 走线块  $\Phi$  和它的上通道  $UC_\phi$ ,  $AB$  为  $\Phi$  的上边。把整个上通道  $UC_\phi$  视为一个闭区域  $CR$ ,  $UC_\phi$  的四边构成了  $CR$  的边界。首先对集合  $UD_\phi$  排序, 从  $A$  开始按顺时针方向沿  $CR$  边界遍历  $UD_\phi$  中的引线端, 根据被访问的先后顺序得到  $UD_\phi$  中引线端的一个排序。而  $UD_\phi$  中的各引线端分别对应于  $T_\phi$  中的一个引线端, 由此可以对应地得到  $T_\phi$  中引线端的一个排序。 $T_\phi$  中引线端将按照该顺序从左往右安置在边  $AB$  上。这样安置  $T_\phi$  中的引线端有利于减小上通道中覆盖  $\Phi$  部分的通道密度和通孔数。参见图 2, 输入:  $N_\phi = \{1, 2, 3\}$ ,  $T_\phi = B_\phi = \{1, 2, 3\}$ ,  $UD_\phi = \{1', 2', 3'\}$ ,  $LD_\phi = \{1'', 2'', 3''\}$ , 先排序  $UD_\phi$  和  $LD_\phi$ , 得  $UD_\phi = \{1', 2', 3'\}$ 、 $LD_\phi = \{2'', 3'', 1''\}$ , 然后相应地得到定序的  $T_\phi = \{1, 2, 3\}$  和  $B_\phi = \{2, 3, 1\}$ 。

在排序后的  $T_\phi$  和  $B_\phi$  中, 排在前面的引线端趋于连接到  $\Phi$  的左边, 而排在后面的引线端趋于连接到  $\Phi$  的右边。一个线网,  $n_i$ , 如果它的引线端  $t_i$  和  $b_i$  都趋于连接往  $\Phi$  的左边, 则称该线网为 LL 型线网; 相反地, 如果  $t_i$  和  $b_i$  都趋于连接往  $\Phi$  的右边, 则称该线网为 RR 型线网。由于上下边引线端的排序是分别进行的, LL/RR 线网的引线端对很可能是非对准的。接下来, 对  $T_\phi$  和  $B_\phi$  头尾中属于 LL/RR 线网的引线端相对顺序做局部调整, 使各 LL/RR 线网的引线端对对准, 这样可简化  $\Phi$  内的布线问题, 同时又不改变通道密度。调整后 LL/RR 线网在  $\Phi$  内只需用垂直线段连接, 如图 5 的线网 a、线网 b。

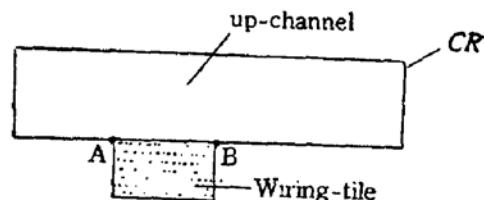


图 3 引线端定序

#### 四、引线端定位和走线块布线

确定了走线块上下边引线端排列顺序之后, 接下来就要在走线块内实现线网的连接。它包括定位和互连两步。定位过程是一个“垂直约束图 (VCG<sup>(2)</sup>) 环检测”与“VCG 环消除”的迭代过程, 它确定引线端的位置保证线网间的垂直约束不存在回路; 互连过程确定各线网在走线块中水平走线的位置并连接各线网。假定在各走线块中每个线网的走线至多只占用一个水平道, 即走线块内的走线不含 ‘Dogleg’。给定走线块  $\Phi$ ,  $\Phi$  中 LL/RR 线网的引线端在  $\Phi$  内对准安置, LL 型线网的引线端靠近  $\Phi$  的左边安置, RR 型线网的引线端靠近  $\Phi$  的右边安置, 在这之后的讨论中将不考虑它们。其他线网的引线端在开始时先按照它们的相对顺序依次从左到右安置在  $\Phi$  的上下边上, 位于 LL 线网引线端之后。

##### (a) 环检测

给定两个线网, 如果它们的引线端出现在同一列上, 则它们之间存在垂直约束, 要求

一个线网的水平走线道必须安置在另一线网水平走线道的上面(或是下面)。从左往右逐列扫描  $\Phi$ , 可以得到线网的垂直约束关系图 VCG。VCG 是有向图, 如果 VCG 上出现回路就不能在  $\Phi$  内 100% 完成布线(前提是不用 Dogleg)。在实现布线之前要求消除 VCG 上的回路。由于走线块内的线网为二端线网, VCG 上每个顶点的出度与入度都小于或等于 1, 换言之, VCG 由无分枝的路径组成。故在 VCG 上检测回路只需简单地遍历一遍图上各路径即可。每次任意选择一个尚未访问过的顶点为起始顶点, 交替地访问后继顶点, 直到再次访问到起始顶点(表明 VCG 上含回路)或路径上没有后继顶点为止; 重复这个过程, 直到找到一个回路或所有顶点都已被访问为止(表示 VCG 上无回路)。

### (b) 回路消除

VCG 上的回路可以通过移动引线端消去。算法简单叙述如下: 当在 VCG 的某一路径上发现回路时, 在  $B_\phi$  中选择一个位于最右边且属于回路上线网的引线端, 将其右移一个位置(开始时引线端是自左往右安置的, 右边空出), 如果其右边一列有另一引线端存在, 则先移动它右边的引线端而后再移动该引线端。右移过程可以简单实现如下:

MoveRight(t) /\* move a bottom terminal t one column right \*/

```
IF a bottom terminal t' one column right to t exist THEN
    MoveRight(t');
    move t one column to its right;
end.
```

右移过程在被移动引线端的右边没有空位置时失败。重复回路检测与回路消去操作, 直至 VCG 上不存在回路(去回路成功)或右移过程失败(去回路失败)。

### (c) 水平走线道分配

消除 VCG 回路的同时也就确定了引线端的位置。消去 VCG 上的回路之后, 各线网水平走线段的位置根据线网的垂直约束关系相应确定。线网水平走线道的分配算法描述如下:

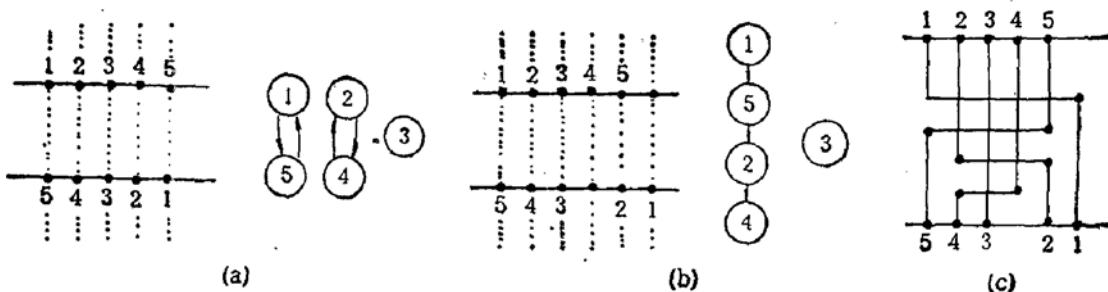


图 4 VCG 回路消除与走线块布线

### Track-Assignment()

```
y = 1; /* relative position of the horizontal tracks */
```

```
FOR each path on VCG DO
```

```
{visit each net along the path start from the beginning;
```

```

FOR each node DO
  {assign 'y' to the net; y = y + 1;}
}
end.

```

图4给出了引线端定位和走线块布线的一个例子。在走线块 $\Phi$ 中，定序后的引线端集合为： $T_\phi = \{1, 2, 3, 4, 5\}$  和  $B_\phi = \{5, 4, 3, 2, 1\}$ ，图4(a)是引线端的初始安置方案和对应的VCG；右移引线端  $b_5$  ( $b_5 = 1$ ) 消去线网1和线网5之间的垂直约束回路，再右移引线端  $b_4$  ( $b_4 = 2$ ) 消去线网2和线网4之间的垂直约束回路，如图4(b)所示；接下来，根据VCG上线网的垂直约束关系确定出各线网的水平走线位置，布线结果见图4(c)。

走线块布线的失败可能是因为走线块的宽度不够(不足以消去VCG上的所有回路)或水平走线道不够。在这种情形下采用的方案是只用垂直走线完成线网的连接。由此要求各线网在走线块内的引线端必须垂直对准。 $T_\phi$  和  $B_\phi$  是分别排序的，以哪个为基准取决于走线块上、下通道的密度。不防假定上通道  $UC_\phi$  较下通道  $LD_\phi$  更拥挤(即密度较大)，则选择  $T_\phi$  中的引线端顺序为线网穿越走线块的顺序，即让  $b_i = t_i, i = 1, n$ ，这样有利于降低上通道  $UD_\phi$  的密度。为充分发挥走线块布线的效用，系统MALS2在布局阶段就考虑分配各走线块以较穿过线网数更大的宽度。

## 五、实验结果与小结

上述各算法用C语言实现并嵌于MALS2门阵布图系统中，整个MALS2系统在SUN4/110工作站上基于UNIX和X-Window实现。通道布线过程参见[3,4]。测试的部分电路实例列于表1。对每个完成的设计，我们统计各通道上的走线密度和实际占

表1 部分测试电路及母片数据

电路名	元件数	线网数	母 片	
			中心通道数	规模(门)
add1	221	238	9	650
add2	442	474	12	1500
add3	408	490	10	1500
motor	159	191	9	650
synt	327	399	9	650

表2 Wiring-tile 布线结果

电路名	减少密度	减少走线道
add1	4	4
add2	2	4
add3	12	16
motor	7	9
synt	1	8

用的走线道数,就嵌入走线块布线过程和未嵌入走线块布线过程做比较,得到由于采用走线块布线过程而减少的通道密度和走线道数,结果列于表 2。可以看出本文提出的走线块布线算法有效地降低了通道密度并提高通道布通率。布图结果还显示出线网总的连线长度和通孔数(包括通道内和单元上的)并不因为引入走线块布线而有较大变化,这可以解释为线网的部分水平走线和通孔从通道‘移到’了走线块中。图 5 给出了两个较大的走线块布线实例。

在门阵布图设计中通道宽度是固定的,采用走线块布线技术虽不能减少电路布图面积但显然有助于提高设计的成功率。文中提出的走线块布线算法不做修改即可用于标准单元模式的布图设计,在标准单元设计中通道宽度是根据实际占用的水平走线道数来确定的,因此减少水平走线道数就意味着减小布图面积。此外,在积木块布图(BBL)模式

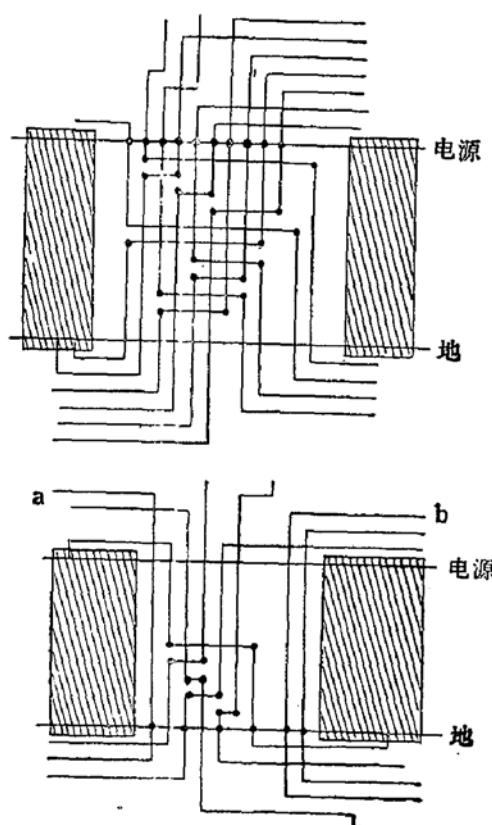


图 5 两个较大走线块布线

中也存在类似的引线端优化问题<sup>[5,6]</sup>,文中提出的算法可以在 BBL 布图中找到应用。

### 参 考 文 献

- [1] 洪先龙等, MALS2 系统研制报告,清华大学计算机系, 1991 年 1 月。
- [2] T. Yoshimura and E. S. Kuh, *IEEE Trans. on CAD*, CAD-1, 25 (1982).
- [3] C. S. Ying, X. L. Hong and E. Q. Wang, Proc. of ICCAD-87, Santa Clara, Nov., 1987.
- [4] 黄浦江,硕士学位论文,清华大学计算机系, 1991 年 3 月。
- [5] T. G. Waring, Ph. D. thesis, Chapter 2, Dept. of Comp. Sci., Univ. of Edinburgh, Oct., 1990.
- [6] X. Yao, M. Yamada and C. L. Liu, Proc. of 25th DAC, pp566—572, 1988.

## Over-the-Cell Routing in 2-Metal-Layer Gate Array Layout

Ying Changsheng and Hong Xianlong

(Department of Computer Science & Technology, Tsinghua University, Beijing, 100084)

### Abstract

In double-metal-layer technology based gate array layout, connection across cell row can be done over the cells. We call each area for across the cell row routing a wiring-tile. Two layers are available for routing over a wiring-tile. We find that carefully locating the terminals and exercising double-layer routing in wiring-tiles help to reduce both channel densities and the number of vias in channels. This paper investigates the problem of wiring-tile terminal assignment and wiring-tile routing for the connection across the cell row. Three steps are involved for solving the problem: first, order the wiring-tile terminals to avoid the intersections among the nets. Then, position the wiring-tile terminals to guarantee a feasible routing solution in wiring-tile. Finally, realize the connection in wiring-tile. They have been implemented and incorporated into a gate array system we developed. The experimental results on several industrial circuits show that the new heuristics help to improve the channel routability significantly.

CCACC: 7410D, 5120