

概伦电子赛题——

标准单元电路的版图自动生成

赛题介绍

赛题背景

什么是标准单元库？

通常，一个数字VLSI设计的最终版图是由一堆晶体管 (*Transistor*) 构成的，但是如果真的将一个数以亿计的晶体管进行优化和设计，工程量是及其浩大的。例如我们盖房子，一般不会从沙子开始，我们会提前准备好砖头，来进行拼接、堆叠。而在VLSI设计里面，盖房子的“砖头”则是标准单元。

为什么标准单元的版图需要自动化？

因为目前业界实现标准单元版图的方式依然是以工程师手工实现为主，而随着工艺的演进，标准单元库中单元数量多达 *700 - 1000* 个，这将耗费大量的人力和时间。而且在一些全定制化的复杂标准单元电路中，人手工画的版图会存在欠优化的问题，算法可以帮助工程师优化版图质量。

对于任意标准单元 C ，我们定义：

$NETS = \{net_i | i = 0, 1, 2, \dots, N_{net}\}$ ，其中 N_{net} 为线网的数量。

$MOS = \{m_i = (x_i, y_i, s_i, g_i, d_i, w_i) | x_i \in N, y_i = 0 \text{ or } 1, s_i, g_i, d_i \in NETS, i = 0, 1, 2, \dots, N_m\}$

其中 x_i 为该晶体管的水平方向的相对位置， w_i 为沟道宽度。

y_i 为晶体管类型，0 表示 NMOS，1 表示 PMOS。

s_i, g_i, d_i 分别表示晶体管左侧、栅极和右侧所连接的线网。

$P = \{p | p \in NETS\}$ 为 pin 的集合，定义 N_{pin} 为 pin 的数量

要求确定所有晶体管 m_i 的状态 $s_i = (x_i, f_i)$ ，满足：

- (1) 若 $x_i - x_j = 1$ ，则 $s_i = d_j$ ；若 $x_j - x_i = 1$ ，则 $s_j = d_i$ ；
- (2) 若 $y_i = y_j$ ，则 $x_i \neq x_j$ ；
- (3) 若 $x_i = x_j$ ，则 $g_i = g_j$ ；其中 f_i 表示晶体管是否翻转源漏。

```

.SUBCKT NAND2V1 A1 A2 ZN VDD VSS

MM1 ZN A1 net18 VSS n09_ckt W=0.12u L=30.00n

MMN1 net18 A2 VSS VSS n09_ckt W=0.12u L=30.00n

MM0 ZN A2 VDD VDD p09_ckt W=0.12u L=30.00n

MMP1 ZN A1 VDD VDD p09_ckt W=0.12u L=30.00n

.ENDS

```

```

M1 = (1,0, net18, A1, ZN, 120nm)
MN1 = (0,0, VSS, A2, net18, 120nm)
M0 = (0,1, VDD, A2, ZN, 120nm)
MP1 = (2,1, ZN, A1, VDD, 120nm)

```

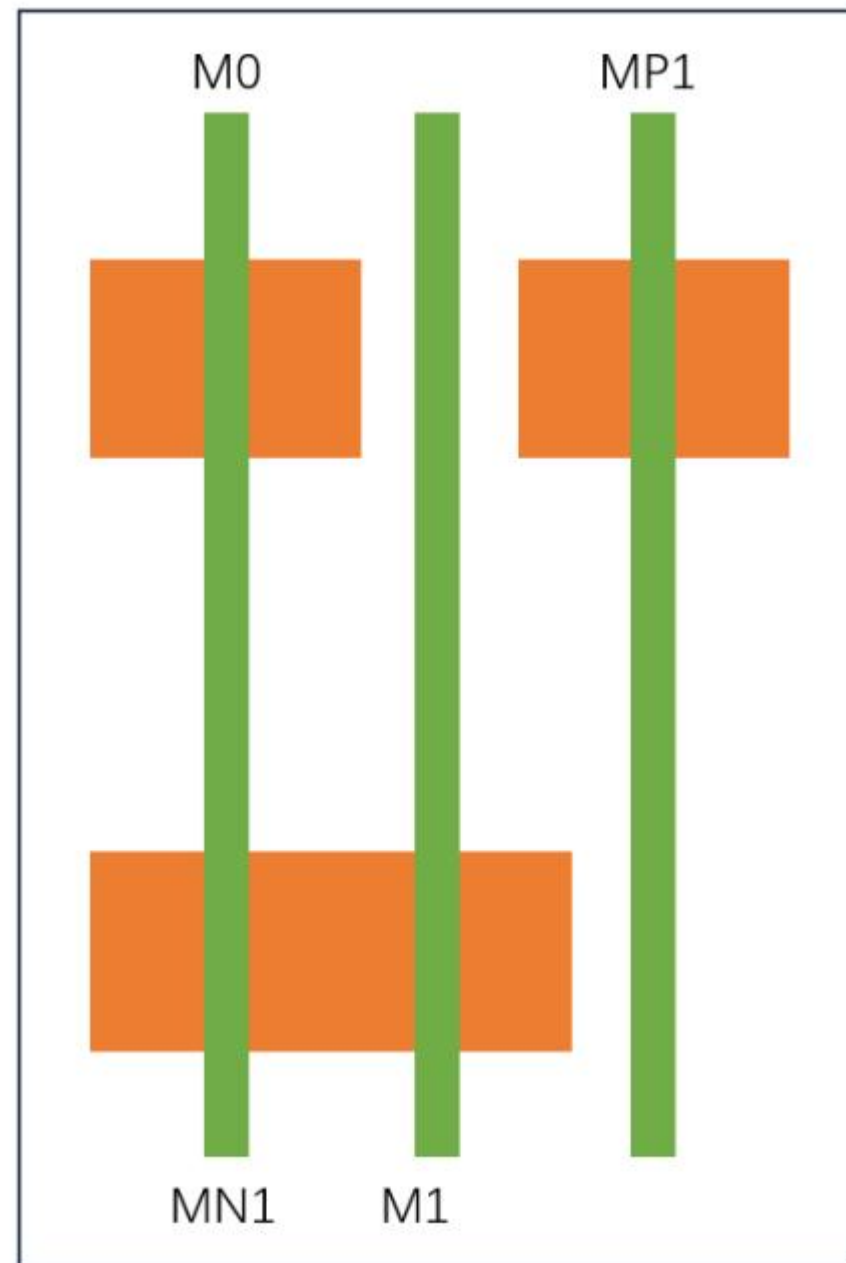


图 3 NAND2V1 布局示意图

数据格式

以每个 MOS 晶体管的名字为 key (若为折叠后的晶体管, 加上_finger\${id}后缀名用

以区分, 具体格式参考 example.json), value 为字典, 各属性含义如下表:

属性字段	含义	值范围
x	水平方向的相对位置, 值为整数。 放在最左侧的 MOS 晶体管 x=0	≥0, 整数
y	0 表示 NMOS, 1 表示 PMOS	0 或 1
source	MOS 晶体管左侧所连接线网	字符串
gate	MOS 晶体管栅极所连接线网	字符串
drain	MOS 晶体管右侧所连接线网	字符串
width	MOS 晶体管的沟道宽度	≥0, 整数, 单位为 nm

```
1 {
2   "placement": {
3     "M0": {
4       "x": "0",
5       "y": "1",
6       "source": "VDD",
7       "gate": "A2",
8       "drain": "ZN",
9       "width": "200"
10    },
11    "M0_finger1": {
12      "x": "1",
13      "y": "1",
14      "source": "ZN",
15      "gate": "A2",
16      "drain": "VDD",
17      "width": "200"
18    }
19    "M1": {
20      "x": "2",
21      "y": "0",
22      "source": "NET18",
23      "gate": "A1",
24      "drain": "ZN",
25      "width": "200"
26    },
27    "M1_finger1": {
28      "x": "3",
29      "y": "0",
30      "source": "ZN",
31      "gate": "A1",
32      "drain": "NET18",
33      "width": "200"
34  },
35  "MN1": {
36    "x": "0",
37    "y": "0",
38    "source": "NET18",
39    "gate": "A2",
40    "drain": "VSS",
41    "width": "200"
42  },
43  "MN1_finger1": {
44    "x": "1",
45    "y": "0",
46    "source": "VSS",
47    "gate": "A2",
48    "drain": "NET18",
49    "width": "200"
50  },
51  "MP1": {
52    "x": "2",
53    "y": "1",
54    "source": "VDD",
55    "gate": "A1",
56    "drain": "ZN",
57    "width": "200"
58  },
59  "MP1_finger1": {
60    "x": "3",
61    "y": "1",
62    "source": "ZN",
63    "gate": "A1",
64    "drain": "VDD",
65    "width": "200"
66  }
67 }
68 }
```

评分标准:

1. 宽度

```
# get cell width
width = 0
for transistor_name, properties in placement.items():
    if width < int(properties["x"]) + 1:
        width = int(properties["x"]) + 1
```

2. 布线长

```
for net, r in self.net_bbox.items():
    if not is_power(net):
        self.bbox += r.x2 - r.x1
```

3. Pin Access

```
self.pin_access = numpy.std(numpy.array(pin_coords))
```

4. DRC (设计规则检查)

5. 运行时间

```
rs = 10 * (1 / (1 + math.exp(runtime / 3600 - 1)))
```

评价指标	评分权重	计算公式	备注
布局宽度	50	$\max_{0 \leq i \leq N_m} x_i + 1$	
布线复杂度	10	$\sum_{n \in NETS} \max_{i, j \in M_n} x_i^n - x_j^n $, 其中 $M_n = \{m_i m_i \in MOS \text{ 且被 } n \text{ 连接}\}$	所有线网所占宽度之和, 不考虑 VDD, VSS。
对称性	10	满足: $ \{x_i x_i = x, i = 0, 1, \dots, N_m\} = 1$ 的整数 x 的个数	栅极不配对的晶体管数量, 具体描述见图 4。越小表示对称性越好。
pin 密度	10	$\frac{1}{N_{pin}} \sum_{i=0}^{N_{pin}} (p_i - \mu)^2$	pin 位置的方差。当存在多个位置时, 选择距离周围 pin 最远的 x
DRC	10	存在三个连续的晶体管 i, j, k, 使得: $x_i = x_j - 1 = x_k - 2$ $y_i = y_j = y_k$ $w_j < w_i, w_j < w_k$	有源区凹槽。若存在该情况, 则为 0 分, 否则满分。具体描述见图 5。
运行时间	10	$1 - \text{sigmoid}(t - 1)$	单位: 小时 要求内存消耗 < 10G

测试用例:

1. 选手可见的电路: 340
2. 所有供测试的标准单元: 1813

