

以 case2 为例

1.统一所有子 PST 的接口名称, 优先统一为源 Port, 比如 (VP1,VP2,VP3,VSS), 如果不存在 Port, 那么同一位源 Net (比如 VN4)。

2.识别最终的 merged PST 一共有几列, 可用可不用, 因为根据下面的方法, 应该最后合并出来的列数就是最终的列数, 不过可以作为检验用。

3.取用 pst 文件最上面的两个 PST, 以列数多的为优先 (自动分析哪张表列数多)

Situation1: 如果是包含关系, 即一个表所有的列数包含了另一列, 那么取列数较多的那个表作为基准表, 找出两个表中同时有效的行 (只需要比较同时存在状态的部分)。

Situation2: 如果是交叉关系, 一个表中有 VP1,VN4,VSS (PST of System/CPU_2), 另一个表中有 VP1,VP2,VSS(PST of System/CPU_1), 这个时候只需要比较 VP1, 如果以“PST of System/CPU_2”为基准表, 那么根据“PST of System/CPU_1”删除不适用状态后, 将 VP2,VSS 加到表后面, 注意这里的 VP2 和 VSS 加在后面要注意根据 VP1 在后面加状态。

4.每次取上一次的基准表和新的一个表进行合并操作, 最后得到 merged PST。

第1步

端口 (Supply Port) 上的供电状态 (Supply State)。如为上述设计中的 VP1, VP2, VSS, VP2, CPU_1/VP1_1, CPU_1/VP2_1, CPU_1/VSS, CPU_1/ALU/VP2_1_1, CPU_1/ALU/VSS, CPU_1/VP1_2, CPU_1/VP4_2, CPU_1/VSS, PMG/VP3_1, PMG/VSS 等端口定义供电状态。对于每个设计模块的供电状态,可以简单的通过表 1-5 来描述。以表 1 为例, 具体的 UPF 命令描述可参见附录部分。

优先统一-Port
无Port 其次统一-Net

PMG/pt3	VP3_1 VP3	VSS
S1	ON	OFF
S2	HV	OFF
S3	LV	OFF
S4	OFF	OFF

表 1. PST of System/PMG

pt	VN1 VP1	VN2 VP2	VN3 VP3	GN1 VSS
S1	ON	ON	ON	OFF
S2	HV	ON	LV	OFF
S3	LV	HV	LV	OFF
S4	ON	OFF	HV	OFF
S5	OFF	OFF	OFF	OFF

表 2. PST of System

CPU_1/pt1	VP1_1 VP1	VN2_1 VP2	GN2 VSS
S1	ON	ON	OFF
S2	ON	HV	OFF
S3	LV	HV	OFF
S4	OFF	HV	OFF
S5	OFF	ON	OFF
S6	OFF	OFF	OFF

表 3. PST of System/CPU_1

CPU_1/ALU/pt1_1	VP2_1_1 VP2	VSS
S1	ON	OFF
S2	HV	OFF
S3	OFF	OFF

表 4. PST of System/CPU_1/ALU

CPU_2/pt2	VP1_2 VP1	VP4_2 VP4	VSS
S1	ON	LV	OFF
S2	ON	OFF	OFF
S3	HV	LV	OFF
S4	LV	LV	OFF
S5	OFF	OFF	OFF

表 5. PST of System/CPU_2



CPU_1/ALU/VSS, CPU_1/VP1_2, CPU_1/VP4_2, CPU_1/VSS, PMG/VP3_1, PMG/VSS 等

端口定义供电状态。对于每个设计模块的供电状态,可以简单的通过表 1-5 来描述。

以表 1 为例,具体的 UPF 命令描述可参见附录部分。

第一步

PMG/pt3	VP3_1	VSS
S1	ON	OFF
S2	HV	OFF
S3	LV	OFF
S4	OFF	OFF

优先统一Port
无Port其次统一Net

表 1. PST of System/PMG

pt	VN1 VP1	VN2 VP2	VN3 VP3	GN1 VSS
S1	ON	ON	ON	OFF
S2	HV	ON	LV	OFF
S3	LV	HV	LV	OFF
S4	ON	OFF	HV	OFF
S5	OFF	OFF	OFF	OFF

表 2. PST of System

第二步
表1包含新表2

表1与表2合并
得新表2,新表2
不变
(此时只比较 VP3和VSS)

CPU_1/ALU	VP1_1	VP2_1	VSS
-----------	-------	-------	-----

S1	ON	OFF
S2	HV	OFF
S3	LV	OFF
S4	OFF	OFF

表 1. PST of System/PMG

pt	VN1 VP1	VN2 VP2	VN3 VP3	GN1 VSS
S1	ON	ON	ON	OFF
S2	HV	ON	LV	OFF
S3	LV	HV	LV	OFF
S4	ON	OFF	HV	OFF
S5	OFF	OFF	OFF	OFF

表 2. PST of System

CPU_1/pt1	VP1_1 VP1	VN2_1 VP2	GN2 VSS
S1	ON	ON	OFF
S2	ON	HV	OFF
S3	LV	HV	OFF
S4	OFF	HV	OFF
S5	OFF	ON	OFF
S6	OFF	OFF	OFF

表 3. PST of System/CPU_1

pt	VP1	VP2	VP3	VSS
S1	ON	ON	ON	OFF
S2	LV	HV	LV	OFF
S3	OFF	OFF	OFF	OFF

合并过来

第3.1步

①发现表2更大件
为基准表

②只比较

新表2 VP1 VP2 VSS
只要查基准表中状态
是否存在即可

③由基准表
得到新表

由基准表
得到新表

pt	VPI	VP2	VP3	VSS
S1	ON	ON	ON	OFF
S2	LV	HV	LV	OFF
S3	OFF	OFF	OFF	OFF

- ① 列数更少
作为基准表
- ② 只需看
基准表中
状态是否存在

CPU_1/ALU/pt1_1	VP2_1_1	VP2	VSS
S1	ON		OFF
S2	HV		OFF
S3	OFF		OFF

表4 PST 叶 System/CPU-1/ALU.

③ 都存在, 得到新表与上一次基准表完全一样



pt	VPI	VP2	VP3	VSS
S1	ON	ON	ON	OFF
S2	LV	HV	LV	OFF
S3	OFF	OFF	OFF	OFF

① 列数更少
作为基准表

CPU_2/pt2	VP1_2 VP1	VP4_2 VN4	VSS
S1	ON	LV	OFF
S2	ON	OFF	OFF
S3	HV	LV	OFF
S4	LV	LV	OFF
S5	OFF	OFF	OFF

② 出现situation 2 ~~☆☆☆~~ (交叉情况)

(2.1) 先比较重叠的部分

如果出现交叉情况, 则两个表都要进行消除无效状态操作.

attention!
基座表不要忘记

pt	VP1	VP2	VP3	VSS
S1	ON	ON	ON	OFF
S2	LV	HV	LV	OFF
S3	OFF	OFF	OFF	OFF

CPU_2/pt2	VP1_2 VP1	VP4_2 VN4	VSS
S1 ✓	ON	LV	OFF
S2 ✓	ON	OFF	OFF
S3 ✓	HV	LV	OFF
S4 ✓	LV	LV	OFF
S5 ✓	OFF	OFF	OFF

(2.1) 发现删去另一张表的S3

(2.2) 将另一张表合并到基准表中, 注意
是根据所有重叠状态进行合并, 比如这
里是根据 VPI 和 VSS 相同的条件下, 加入 VN4

(2.3) 此时从基准表依次取得条目,

首先取基准表 $pt=S1$

	VPI	VP2	VP3	VSS
S1	ON	ON	ON	OFF

然后并上 VPI 与 VSS 取相同状态的另一张表的
VN4 的状态, 发现有 2 条.

CPU_2/pt2	VP1_1 VPI	VP4_2 VN4	VSS
S1	ON	LV	OFF
S2	ON	OFF	OFF

合并得

	VPI	VP2	VP3	VSS	VN4
S1	ON	ON	ON	OFF	LV
	ON	ON	ON	OFF	OFF

以此类推, 得到

	VP1	VP2	VP3	VSS	VN4
S1	ON	ON	ON	OFF	LV
S2	ON	ON	ON	OFF	OFF
S3	LV	HV	LV	OFF	LV
S4	OFF	OFF	OFF	OFF	OFF

注意这里每次对 PT S1, S2, S3, ..., 进行重新编号。

最后, 将 merged PST 中
进行替换.

ON - 1.2
 HV - 1.0
 LV - 0.6
 OFF - 0.4

得结果。

VP1	VP2	VN4	VP3	VSS
1.2	1.2	off	1.2	off
off	off	off	off	off
1.2	1.2	0.6	1.2	off
0.6	1.0	0.6	0.6	off

表 6. Merged PST