POWER VIA IMPROVEMENT SCRIPT

Thimma Raju Vadde

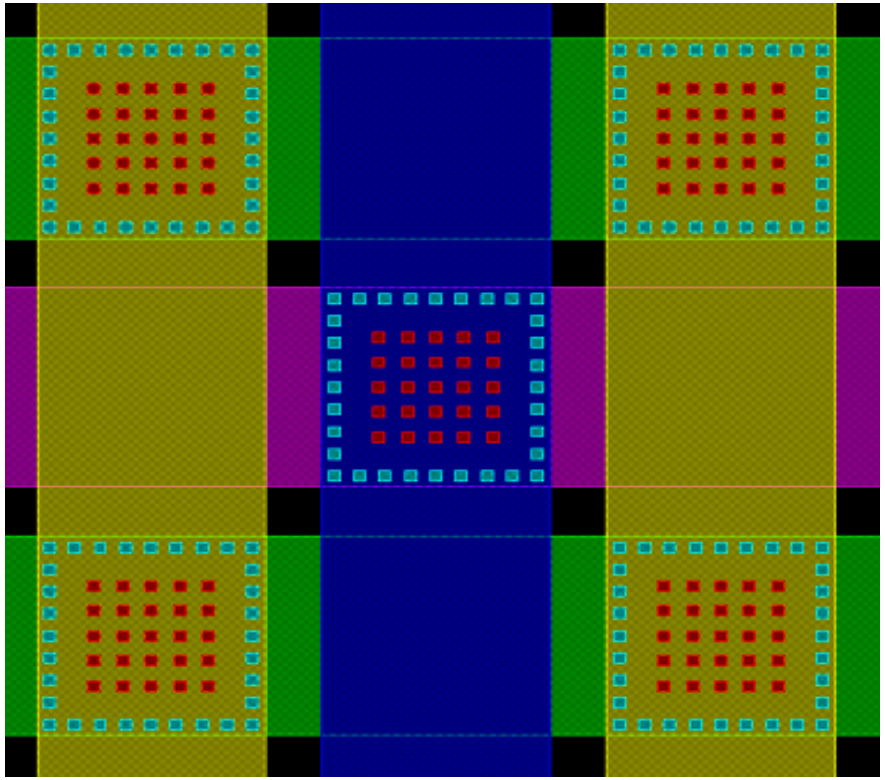
20 August 2020

Agenda

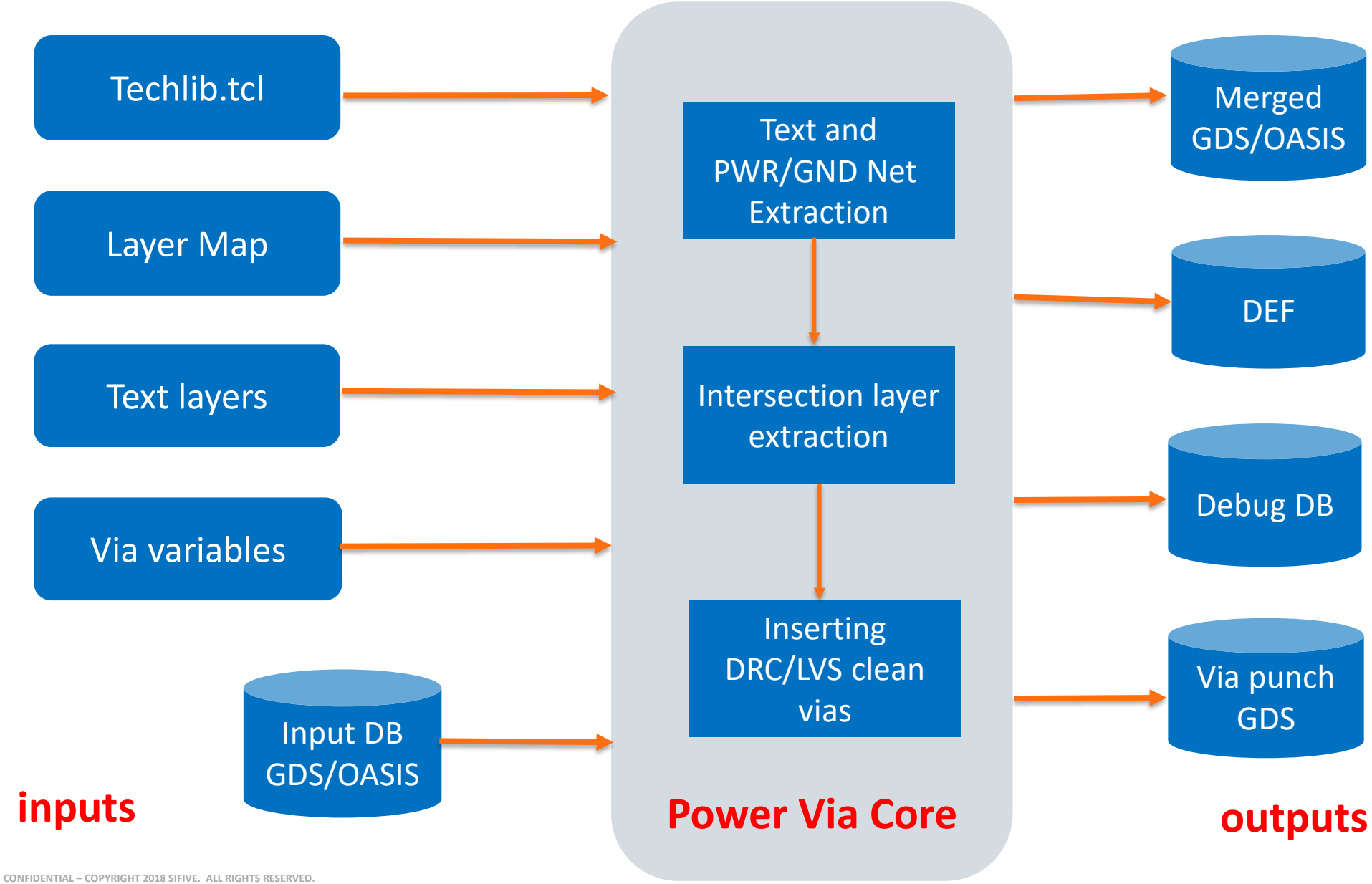
- Introduction to power via improvement flow using Calibre script
- Why this script is needed and scope of it
- Demonstration and analysis

Introduction

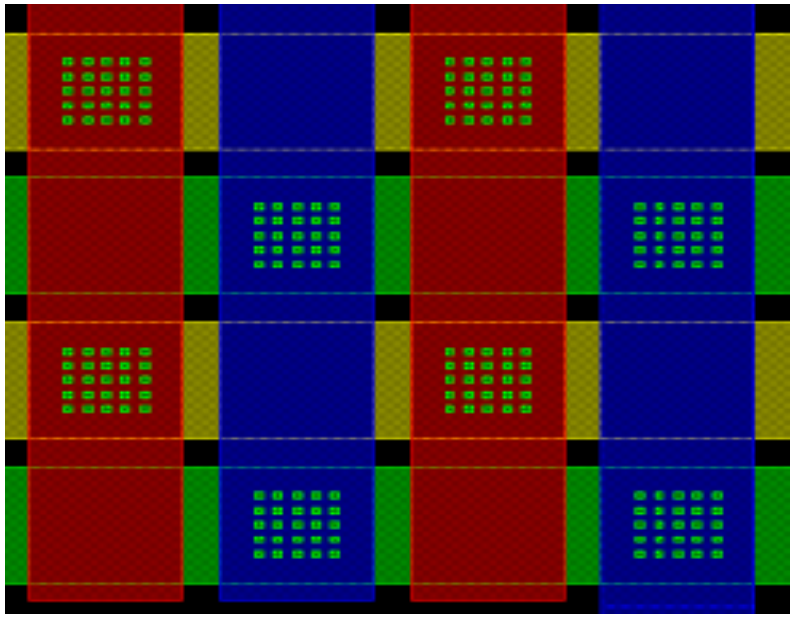
- **Power via flow inserts VIA to reduce IR and EM drops in chip design.**
- **Calibre power via flow doesn't disturb connectivity or cause new DRC's.**



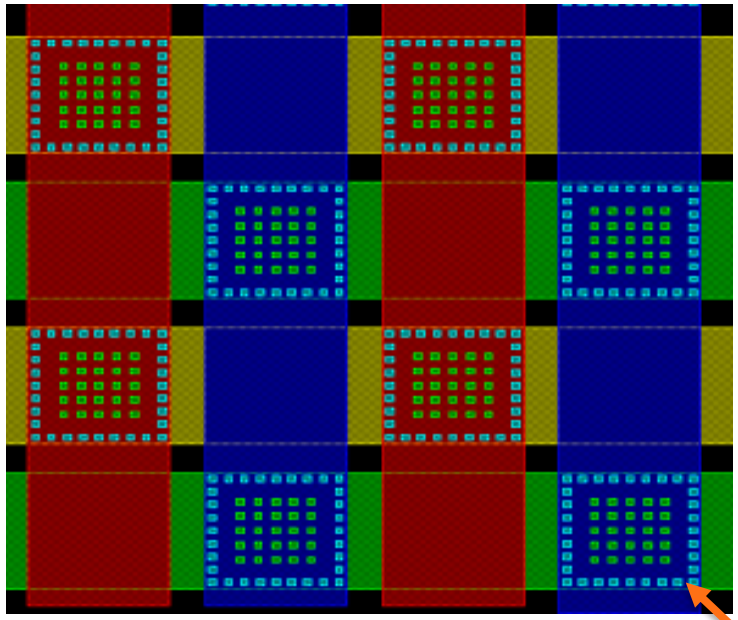
Power Via Flow



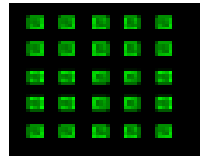
Sample design flow



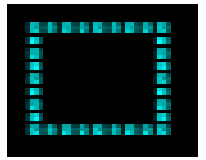
initial DB



improved DB



original vias



inserted vias

inserted vias

Run Results

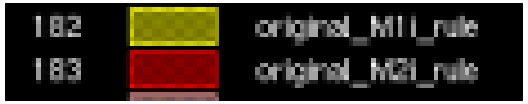
- **Outputs**
  - merged DB (GDS/OASIS)
  - via_punch_fill.gds (new vias DB added by power via flow)
  - incremental.def
  - net_aware_PWR_GND.oas (Debug DB)
- **Runtime is less than an hour for top level block with 16 CPU's.**
- **Vias that are added by power via flow**

```
RULECHECK VIA1iCFILLR_count ..... TOTAL Result Count = 166218
RULECHECK VIA2iCFILLR_count ..... TOTAL Result Count = 224830
RULECHECK VIA3iCFILLR_count ..... TOTAL Result Count = 15450
RULECHECK VIA4iCFILLR_count ..... TOTAL Result Count = 88874
RULECHECK VIA5iCFILLR_count ..... TOTAL Result Count = 297270
RULECHECK VIA6iCFILLR_count ..... TOTAL Result Count = 382236
RULECHECK VIA7iCFILLR_count ..... TOTAL Result Count = 547592
RULECHECK VIA8iCFILLR_count ..... TOTAL Result Count = 553888
RULECHECK VIA9iCFILLR_count ..... TOTAL Result Count = 857
RULECHECK VIA10iCFILLR_count ..... TOTAL Result Count = 1692920
```

Debug DB

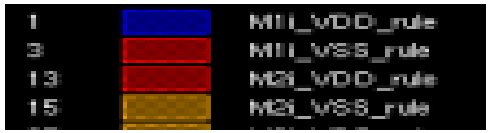
- **Debug DB : net_aware_PWR_GND.oas**
- **We can identify original and added vias by analyzing layers in the debug DB.**

- Original metal layers



- Original via layers

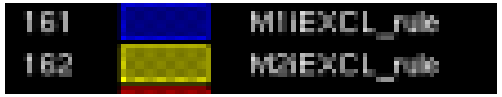
- Metal layers identified by flow for each net



- same net metal layer intersection

- Fill area for each via layer

- Exclusion layer for metals/vias



- Inserted vias