3D IC Analysis using RedHawk

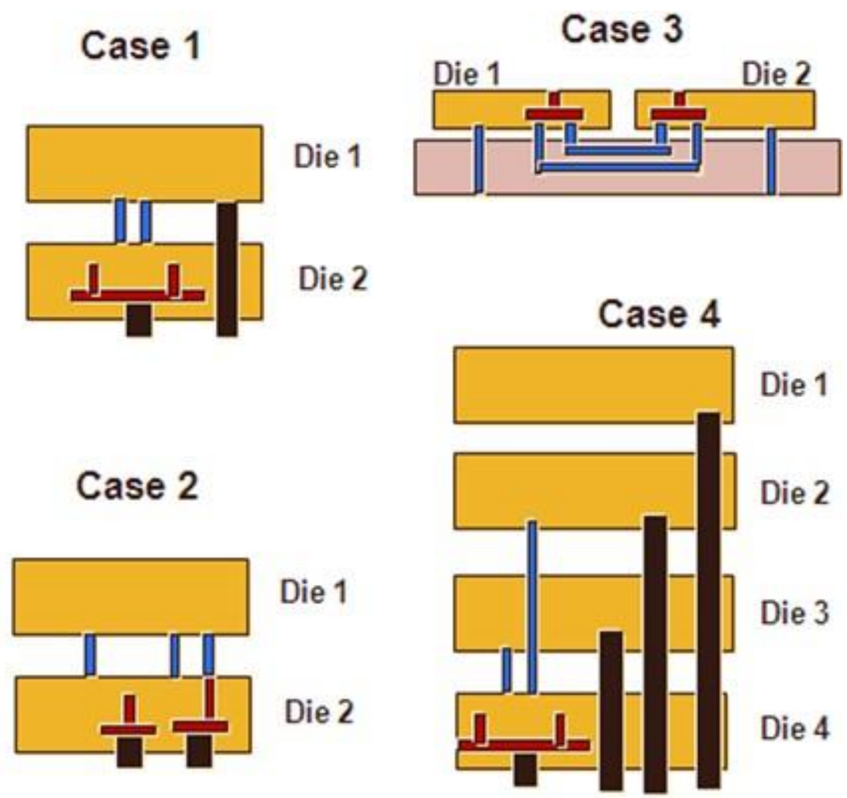
Introduction

3D die stacking using Through Silicon Vias (TSVs) is an emerging technology with considerable promise in reducing the area, performance and power limitations of transmitting signals between multiple dies. It provides the flexibility of connecting chips performing different functions (memory, processor, power management) and fabricated using different processes inside the same package in a significantly closer form than what is possible in other multi-die packages like MCMs and 3D ICs that do not use TSVs.

But challenges increase in the PDN network design and in ensuring power integrity. Power Noise to the top die comes from the switching within the die as well as from the noise from bottom die. Both the die's PDN are not mutually exclusive. Count, design and placement of TSV's play a major role in ensuring power integrity.

Supported Configurations

RedHawk supports several TSV based IC configurations as shown in the figure.

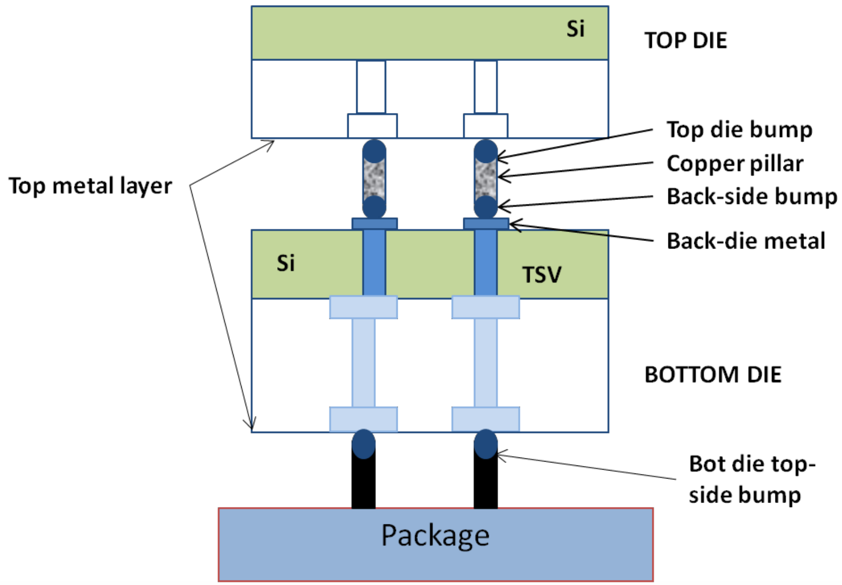


Analysis Methods

- RedHawk can do stacked IC analysis using 2 ways :
  - **Concurrent Analysis :** If the design database is available for both the dies , then concurrent analysis [both dies together] can be done in a single run.
  - **Model Based Approach :** It is quite possible that the database may not be available for one or more of the dies, as they may come from different vendors. Hence vendors can create a Electrical Model of the die called CPM(Chip Power Model) and give it to other customers. Customers can use mixed approach of CPM + die or CPM + CPM to perform IR drop analysis.

Data Setup For Concurrent Analysis

- The data setup for concurrent analysis is only slightly different from normal RedHawk data preparation.
  - RedHawk needs the data for multiple dies in separate GSR's (Global System Requirements file)
  - For e.g. , for the sample 3D IC configuration shown below, two GSR files: **top_die.GSR** and **bottom_die.GSR** need to be created for importing the data in RedHawk



Creating Bottom DIE GSR

- The tech file needs some additional information for the bottom die (Explained in detail in next page)

```
TECH_FILE <path_to_bottom_die_tech_file>
```

- The package gets connected to bottom die. Hence the bottom die GSR must have the bump locations file under PAD_FILES

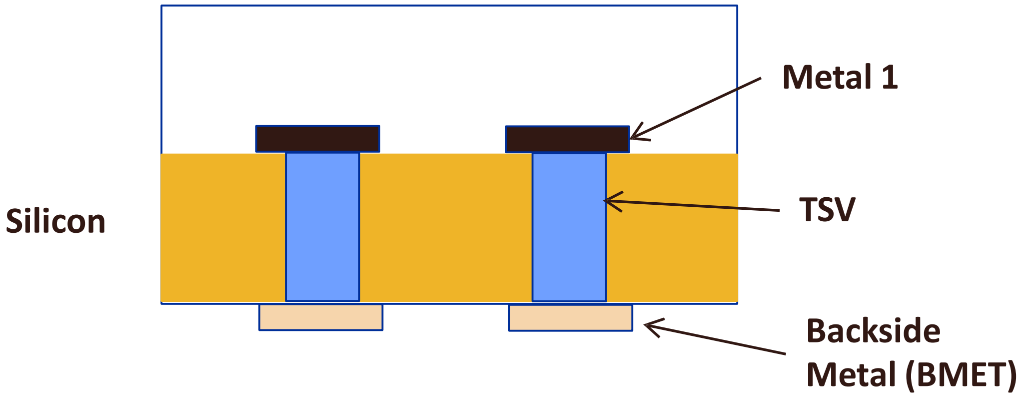
```
PAD_FILES {
<path_to_file>
}
```

Tech File Information

- TECH file requirements :
  - RedHawk models TSVs as vias in the TECH_FILE. Hence the bottom die which contains the TSV's need EM, RLC, area, etc related information in its tech file (e.g. shown below).
  - All the other metal layers below the substrate which connect to the top metal layers of the top die also need to be modeled appropriately in tech file

```
via TSV {
     em 0.072
     Area 0.003600
     UpperLayer met1
     LowerLayer BMET
     Resistance 0.8
     Capacitance 50
     Inductance 0.130
     }
```

```
metal BMET {
  T 125
  Coeff_RT1 2.950e-03
  MINWIDTH 0.08
............
}
```



Creating top DIE GSR

- A “TSV” file indicates how this “top die” (which references this TSV file) connects to the other die defined in the TSV file.
  - The TSV file is an input given under PAD_FILES in top DIE GSR.
  - It describes the method by which the bottom die gets connected to top die and the location of copper pillars.
  - Example of a TSV file shown in the next slide.

```
PAD_FILES {
top_die.tsv # the TSV file indicates how this die connects
   to other dies
}
```

Creating a TSV File

DIE bottom_die

ORIENTATION FS

PLACE 1080 1000

LAYER AP

CONNECT BMET

RESISTANCE 0.01

CAPACITANCE 0.8

INDUCTANCE 11

CP bump1 1001 1332 1021 1352

CP bump2 6545 2295 6565 2315

Name of the die to which this “top die” connects

Refer to ***next few slides*** how to decide these settings

The copper pillar connects the layer defined by “LAYER” (in top die) to the layer defined by “CONNECT” (in bottom die)

R,L,C of Copper Pillars, default values if not defined individually

Location of copper pillar landing (coordinates refer to bottom die origin, lower left)

Defining Top DIE Orientation in TSV file

In a RedHawk run, the bottom die is the reference for the coordinate system and orientation of the top die. Top die can be treated as an instance of the bottom die and then decide on its orientation with respect to bottom die like any other instance in the DEF of the bottom die.

Next few slides show some basic points to keep in mind while preparing .tsv file for top die **with the help of an example**, and below are the description for some orientation and connectivity related keywords in the .tsv file:

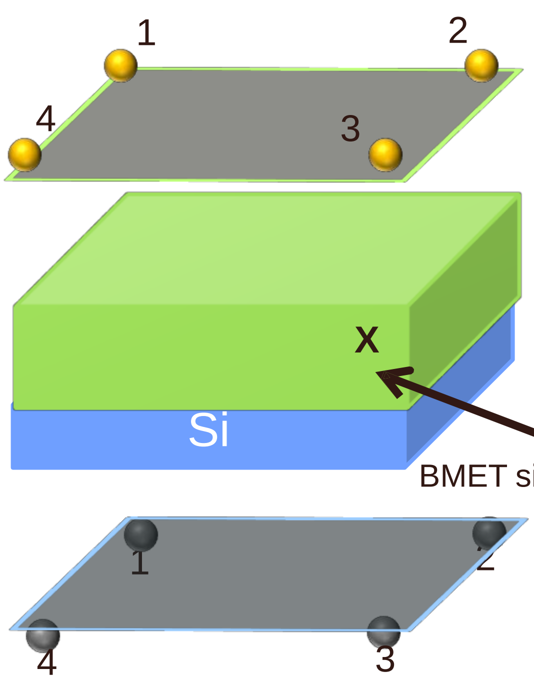
Section of .tsv file

DIE <**bot_die_ptr**>	### die to which the die specifying the .tsv is being attached ### should match "import GSR <bot_GSRfile> -die <**bot_die_ptr**>" ### in RH command file i.e. the name assigned to the bottom die
ORIENTATION FS	### refer to next slides
PLACE x y	### offset of top die origin w.r.t. bottom die
LAYER <layer_top>	### layer of the top die connected to bottom die
CONNECT <layer_bot>	### layer of the bottom die to which top die connects

Definitions: Bump Placements on Bottom and Top Dies

Assuming equal number of power/ground bumps on both dies (shown), say both dies appear as below in **‘N’** orientation:

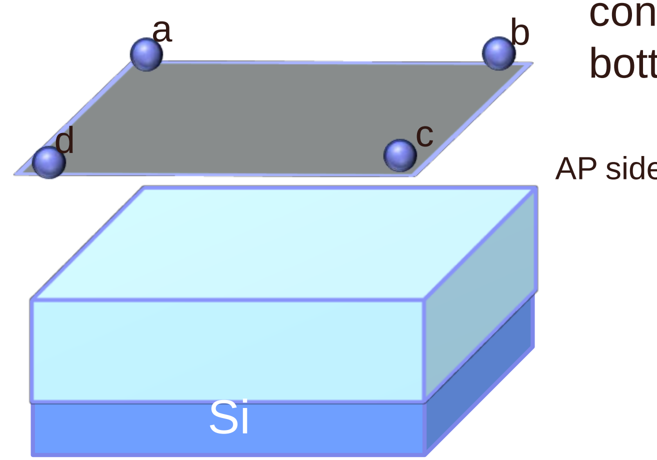
Bottom die



AP layer (top most metal)

BMET layer (connects to MET1 using TSV)

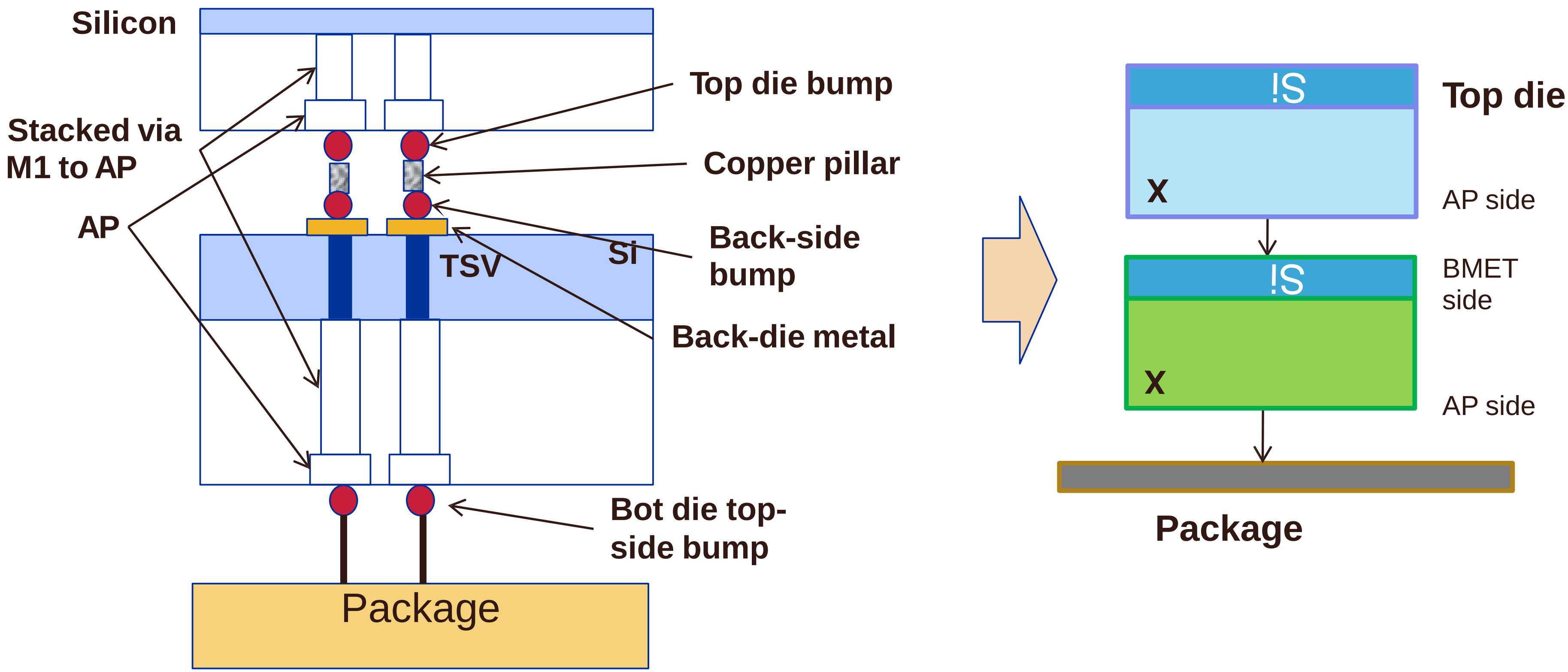
Top die



AP layer, connecting to bottom die

Symbol X to show the orientations: ‘X’ on the front right top corner means die is in ‘N’ orientation (followed in next few slides)

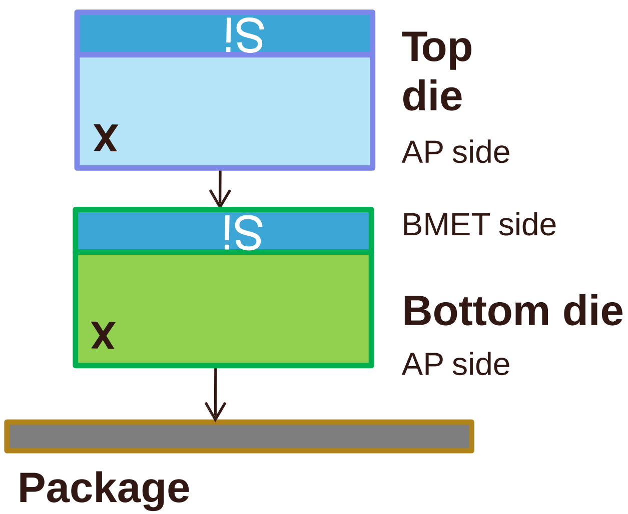
Example Configuration #1
Two Chips Facing Same Direction

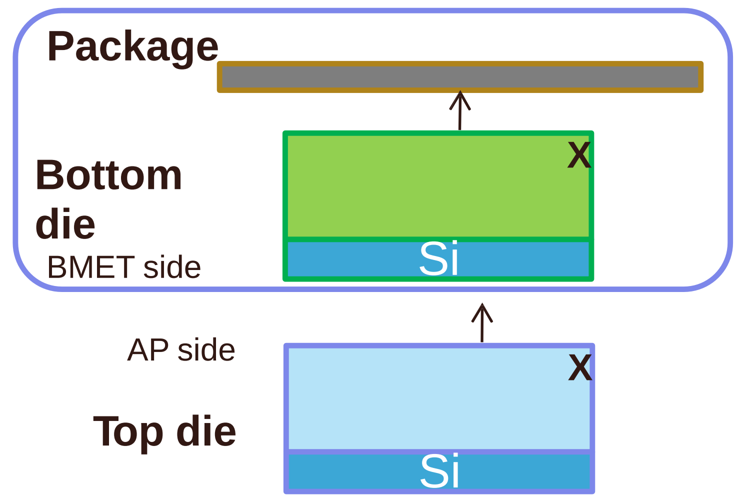


Example Configuration #1
Two Chips Facing Same Direction (contd.)

So when the bottom and top die both are flipped chip designs, connected face-to-back, then **ports 1,2,3,4** for bmet layer of bottom die connect to **ports a,b,c,d** of top die AP layer **respectively**

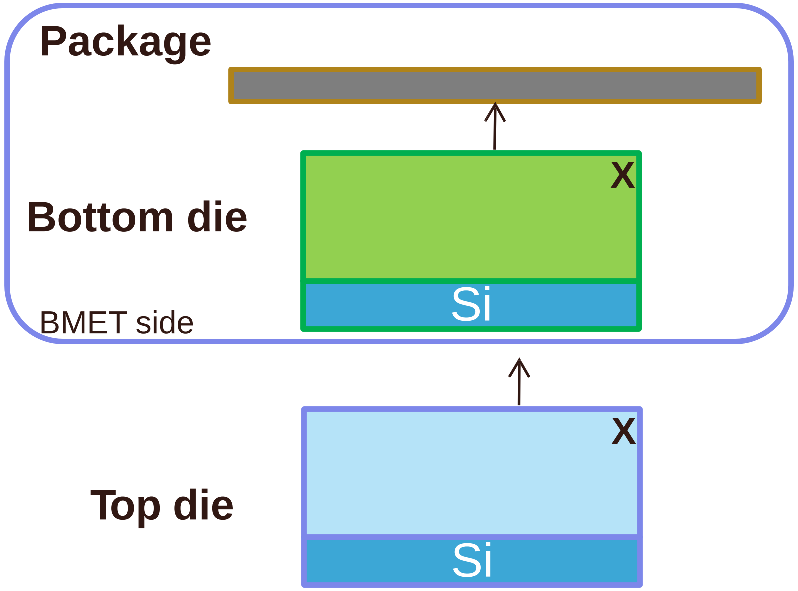
RedHawk sees the design such that the silicon of the bottom die faces the bottom, so the configuration needs to be visualized as follows for the die-to-die connection.





Example Configuration #1
Two Chips Facing Same Direction (contd.)

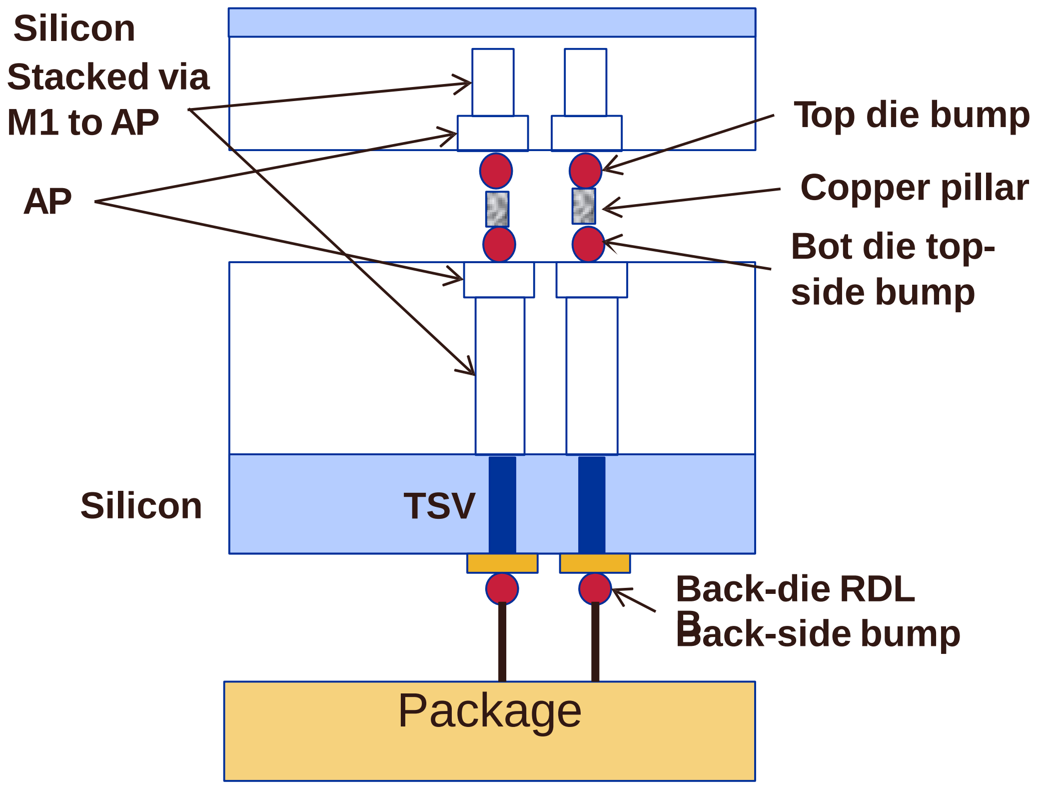
RedHawk sees the design such that the silicon of the bottom die faces the bottom, so the configuration needs to be visualized as follows for the die-to-die connection.

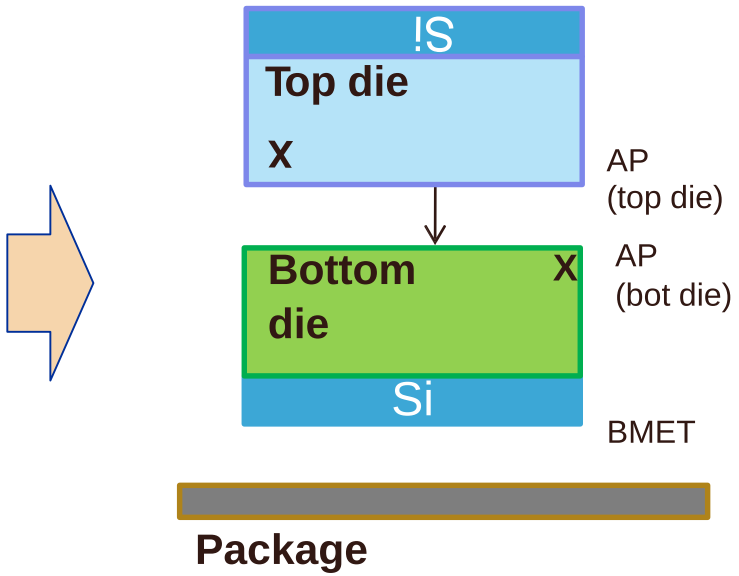


TSV settings needed:

ORIENTATION	N
LAYER	AP
CONNECT	BMET

Example Configuration #2
Silicon of Two Chips Away From Each Other

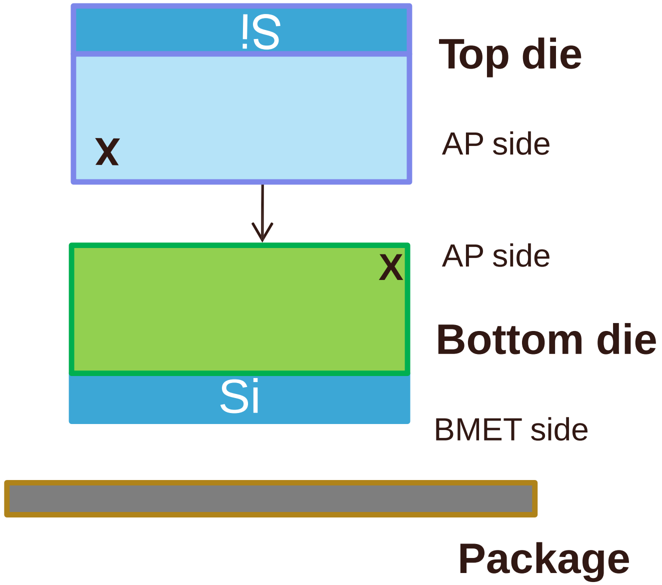




Example Configuration #2
Silicon of Two Chips Away From Each Other

When the bottom die is not flipped chip but top die is, then **ports 1,2,3,4** for AP layer of bottom die connect to **ports b,a,d,c** of top die AP layer **respectively**
(refer to the views shown earlier)

RedHawk always sees the design such that the silicon of the bottom die (reference die) is facing the bottom **So top die needs to be flipped and connection of its AP layer to AP layer of bottom die** is to be specified so that correct connections are made



TSV settings needed:

ORIENTATION	FN
LAYER	AP
CONNECT	AP

RH Command File

#Setting up design :

```
import GSR top_die.GSR -die topdie
import GSR bottom_die.GSR -die botdie
setup design
```

The "-die" option associates a "name" with each of the die's being imported using the GSR.
These names are not keywords.
These are used to refer to the dies at other places in the file and also in .tsv file

#Performing Power Calculation and extraction:

```
perform pwrcalc
perform extraction -power -ground -c
```

#Static/Dynamic Analysis :

```
perform analysis -static -die topdie
perform analysis -vectorless -die topdie
                 or
perform analysis -static
perform analysis -vectorless
```

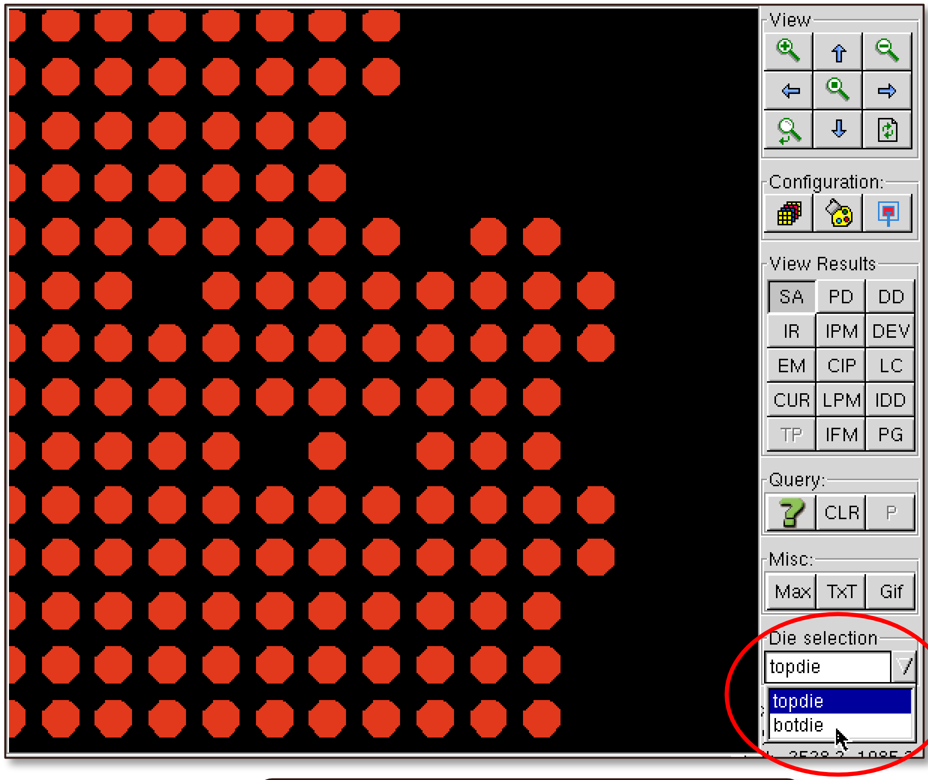
For performing concurrent analysis

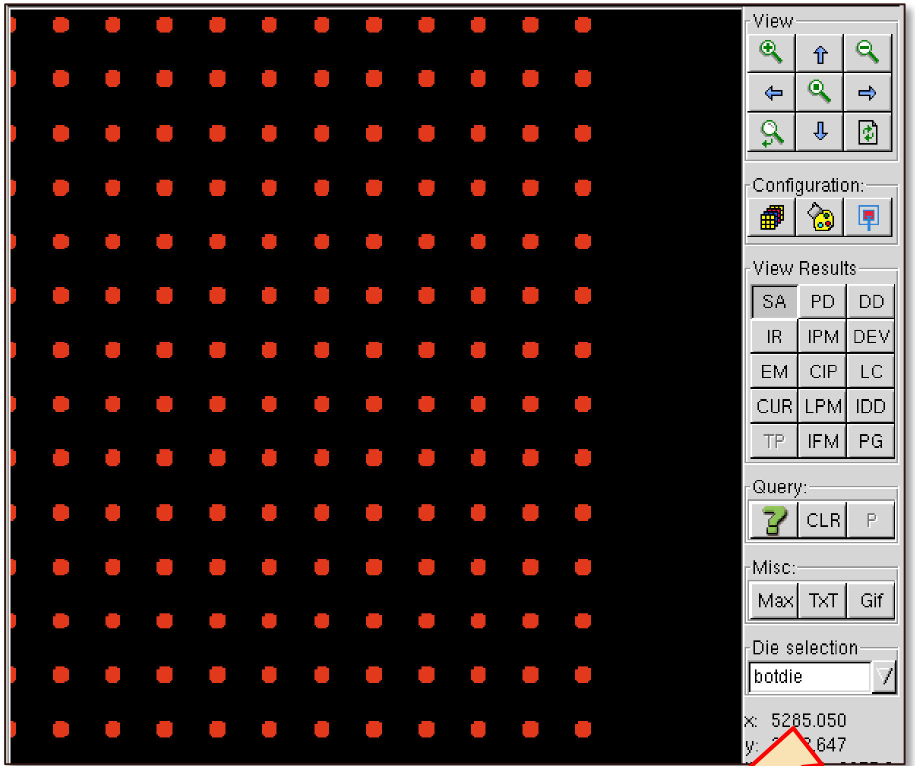
```
perform analysis -static -die botdie
perform analysis -vectorless -die botdie
```

For performing just bottom die analysis

GUI Usage for Multiple Dies

- Use “die selection” option in GUI to switch between dies

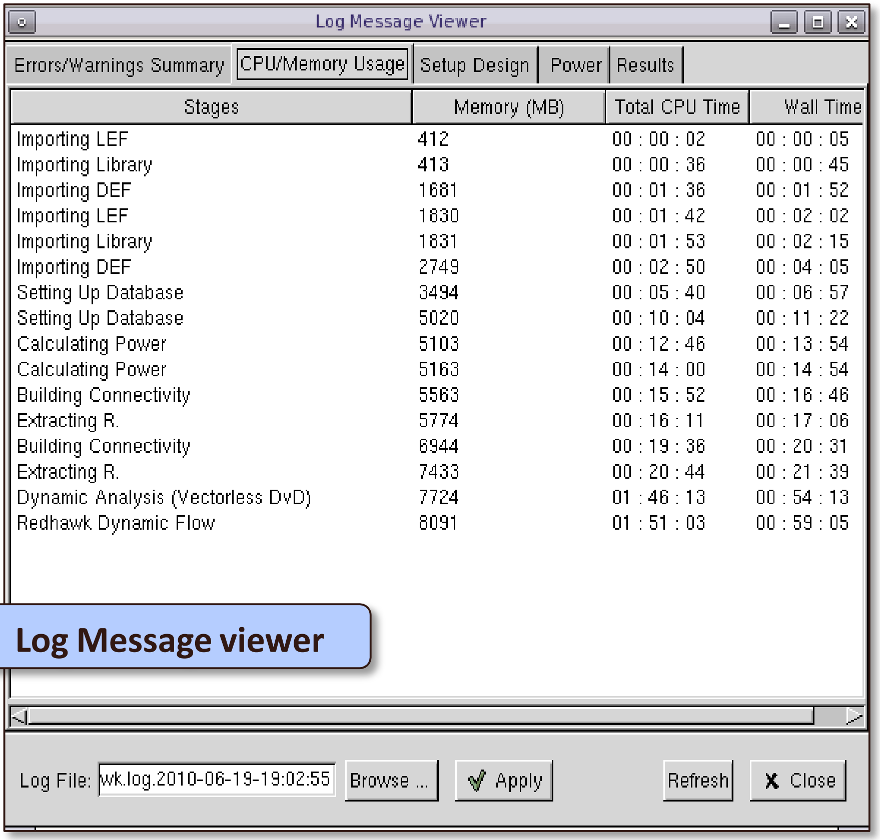




Switching from ‘top die’ view to ‘bottom die’ view in GUI

Log message and file structure

- The log message viewer shows steps for both dies and adsRpt directory also contains “botdie” and “topdie”, each of which contain usual redhawk output files for respective dies:



Log Message Viewer

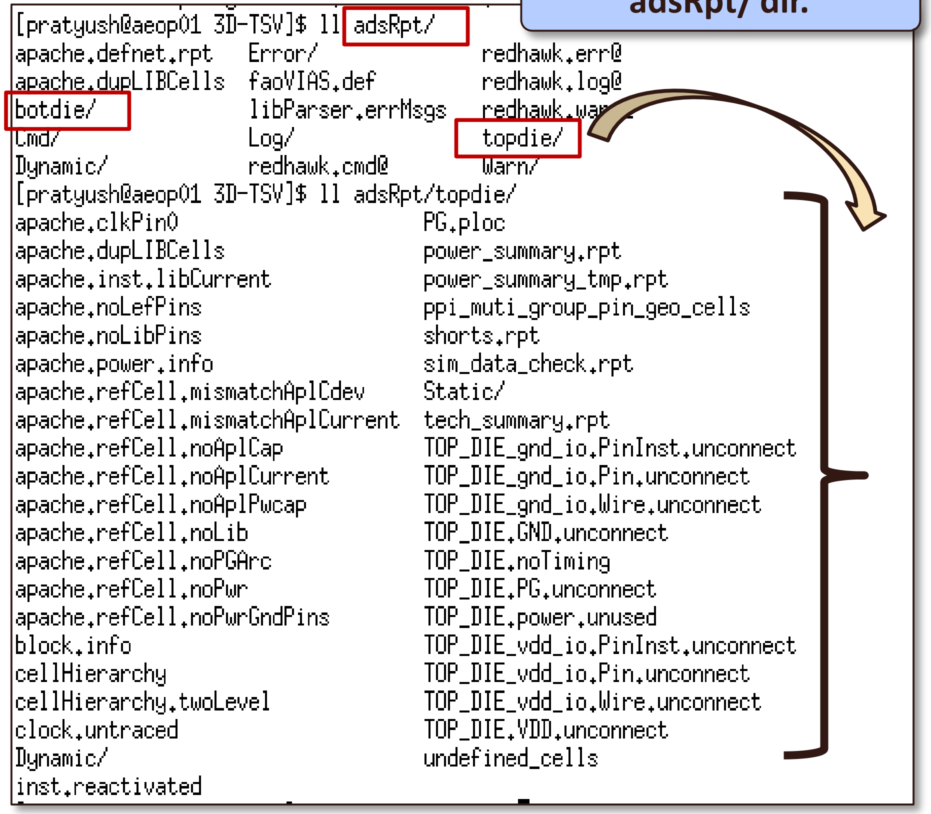
Errors/Warnings Summary | CPU/Memory Usage | Setup Design | Power | Results

Stages	Memory (MB)	Total CPU Time	Wall Time
Importing LEF	412	00 : 00 : 02	00 : 00 : 05
Importing Library	413	00 : 00 : 36	00 : 00 : 45
Importing DEF	1681	00 : 01 : 36	00 : 01 : 52
Importing LEF	1830	00 : 01 : 42	00 : 02 : 02
Importing Library	1831	00 : 01 : 53	00 : 02 : 15
Importing DEF	2749	00 : 02 : 50	00 : 04 : 05
Setting Up Database	3494	00 : 05 : 40	00 : 06 : 57
Setting Up Database	5020	00 : 10 : 04	00 : 11 : 22
Calculating Power	5103	00 : 12 : 46	00 : 13 : 54
Calculating Power	5163	00 : 14 : 00	00 : 14 : 54
Building Connectivity	5563	00 : 15 : 52	00 : 16 : 46
Extracting R.	5774	00 : 16 : 11	00 : 17 : 06
Building Connectivity	6944	00 : 19 : 36	00 : 20 : 31
Extracting R.	7433	00 : 20 : 44	00 : 21 : 39
Dynamic Analysis (Vectorless DvD)	7724	01 : 46 : 13	00 : 54 : 13
Redhawk Dynamic Flow	8091	01 : 51 : 03	00 : 59 : 05

Log File: wk.log.2010-06-19-19:02:55 Browse ... Apply Refresh Close

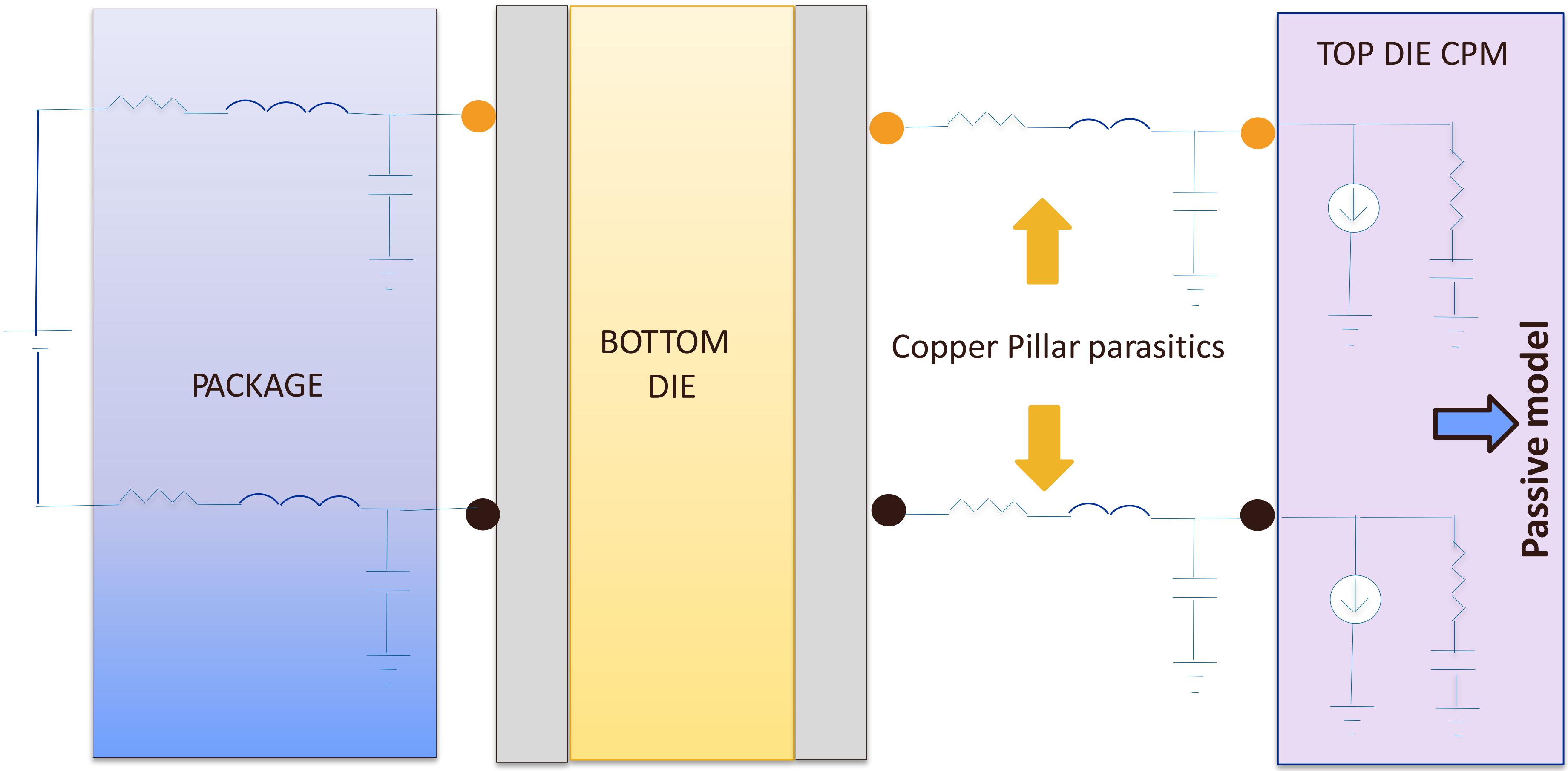
Log Message viewer

adsRpt/ dir.



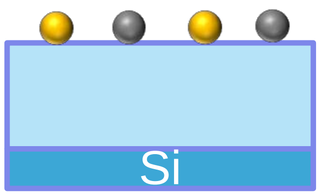
Model Based Approach

User can create a CPM for one of the dies and hook it to the other die and do a CPM + RH analysis. Example setup shown below.



Setting up the flow contd.

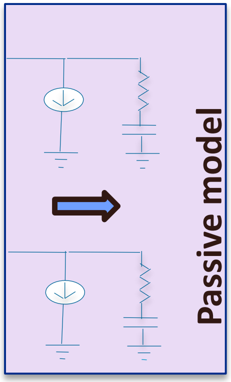
- User can create a CPM for one of the dies(say top die) and hook it to the other die and do a CPM + RH analysis
  - Create some plocs on the top most layer of the top die where connections come from the bottom die




```
vdd_1   1010 2021 MET8 VDD
vdd_2   100 1902 MET8 VDD
.............................................
vss_1   100.2 202.34 MET8 VSS
vss_2   190.2 54.34 MET8 VSS
```

  - Create CPM for the top die using the perform power model command. E.g. :

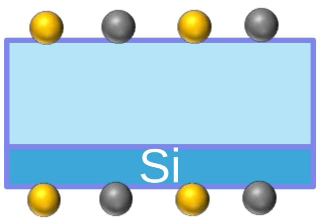
```
perform powermodel -wirebond
```



Setting up the flow contd.

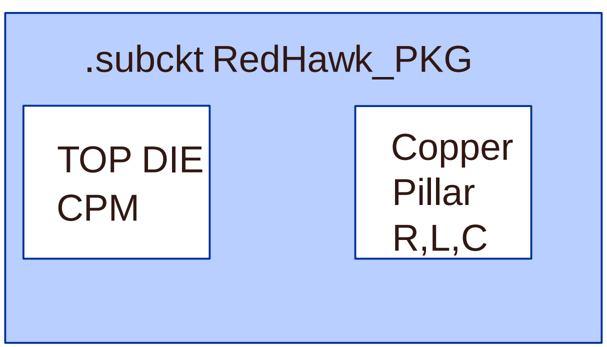
- Create Redhawk setup for bottom die stand alone

- Create plocs on the topmost as well as bottommost layer of the bottom die




```
top_vdd_1   1090 1521 MET8 VDD
bot_vdd_1   120 1902 BMET1 VDD
………………………………………………………………..
top_vss_1   170.2 2802.34 MET8 VSS
bot_vss_1   180.2 894.34 BMET1 VSS
```

- Edit package file for the bottom die to include CPM and copper pillar parasitics




```
.subckt RedHawk_PKG top_vdd1 bot_vdd1 top_vss_1 bot_vss_1
 ……………………………………………………………………………………………………………..

include 'top_die_cpm.sp'

XadsPowerModel topdie_CPM_VDD_PORT1 topdie_CPM_VSS_PORT1
adsPowerModel

R_connection_VDD top_vdd1 tmp1 10e-03
L_connection_VDD tmp1 topdie_CPM_VDD_PORT1 120e-12
C_connection_VDD topdie_CPM_VDD_PORT1 0 2e-12
……………………………………………………………………………………
```

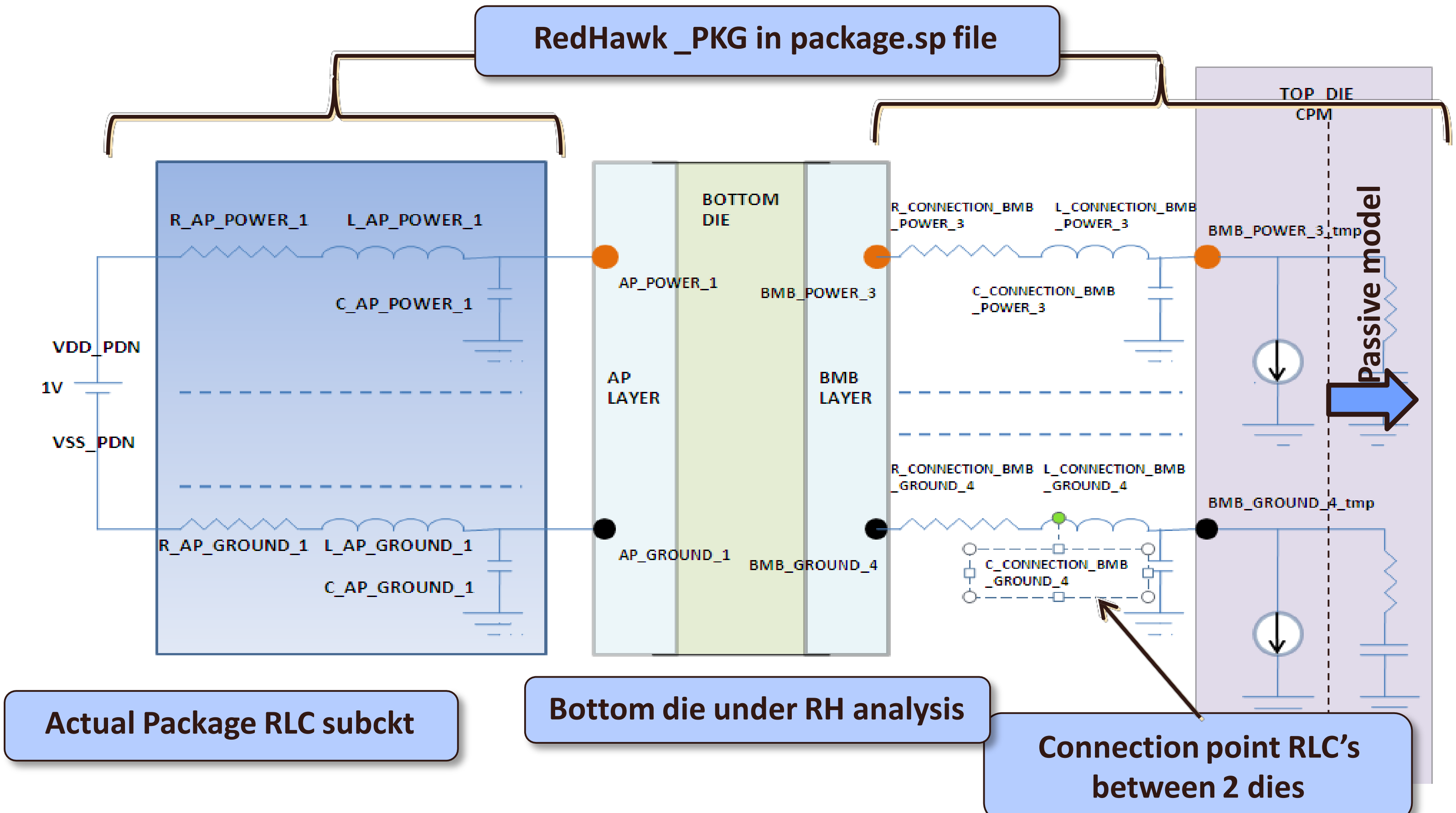
Setting up the flow contd.

- Hook up the CPM within the package
  - Note that it is important to take care of the top die placement and orientation with respect to bottom die while hooking up the CPM to the bottom die

- Run RedHawk
  - The GSR and command file remain the same as for bottom die stand alone
  - Include the package in the GSR

Example: Top die CPM hookup

Package gets connected to the bottom die. Package model contains the top die CPM inside it as explained in earlier slides.



Example: Top die CPM included inside the package

Package subckt needs to be modified so that it contains the additional ports from CPM and invokes CPM model from within.

package.sp file

```
.subckt REDHAWK_PKG BMB_GROUND_4 BMB_GROUND_3 BMB_GROUND_2 BMB_GROUND_1
2 BMB_GROUND_11 BMB_GROUND_10 BMB_GROUND_9 BMB_GROUND_16 BMB_GROUND_15
MB_POWER_6 BMB_POWER_5 BMB_POWER_4 BMB_POWER_9 BMB_POWER_8 BMB_POWER_7
_GROUND_3 AP_GROUND_4 AP_GROUND_5 AP_GROUND_6 AP_GROUND_7 AP_GROUND_8 A
_GROUND_14 AP_GROUND_15 AP_GROUND_16 AP_POWER_1 AP_POWER_2 AP_POWER_3 A
AP_POWER_10 AP_POWER_11 AP_POWER_12


.include 'multiport.sp'

XadsPowerModel BMB_GROUND_14_tmp BMB_GROUND_15_tmp BMB_GROUND_16_tmp BM
tmp BMB_GROUND_5_tmp BMB_GROUND_6_tmp BMB_GROUND_7_tmp BMB_GROUND_8_tmp
tmp BMB_POWER_10_tmp BMB_POWER_11_tmp BMB_POWER_12_tmp BMB_POWER_7_tmp
MB_POWER_6_tmp BMB_POWER_1_tmp BMB_POWER_2_tmp BMB_POWER_3_tmp BMB_GROU

*** DIE connection point RLC ****

R_connection_BMB_GROUND_4 BMB_GROUND_4 tmp1 0
L_connection_BMB_GROUND_4 tmp1 BMB_GROUND_4_tmp 0
C_connection_BMB_GROUND_4 BMB_GROUND_4_tmp 0  0

R_connection_BMB_GROUND_3 BMB_GROUND_3 tmp2 0
L_connection_BMB_GROUND_3 tmp2 BMB_GROUND_3_tmp 0
C_connection_BMB_GROUND_3 BMB_GROUND_3_tmp 0  0
```

Ports added to subckt to hook CPM vdd/vss

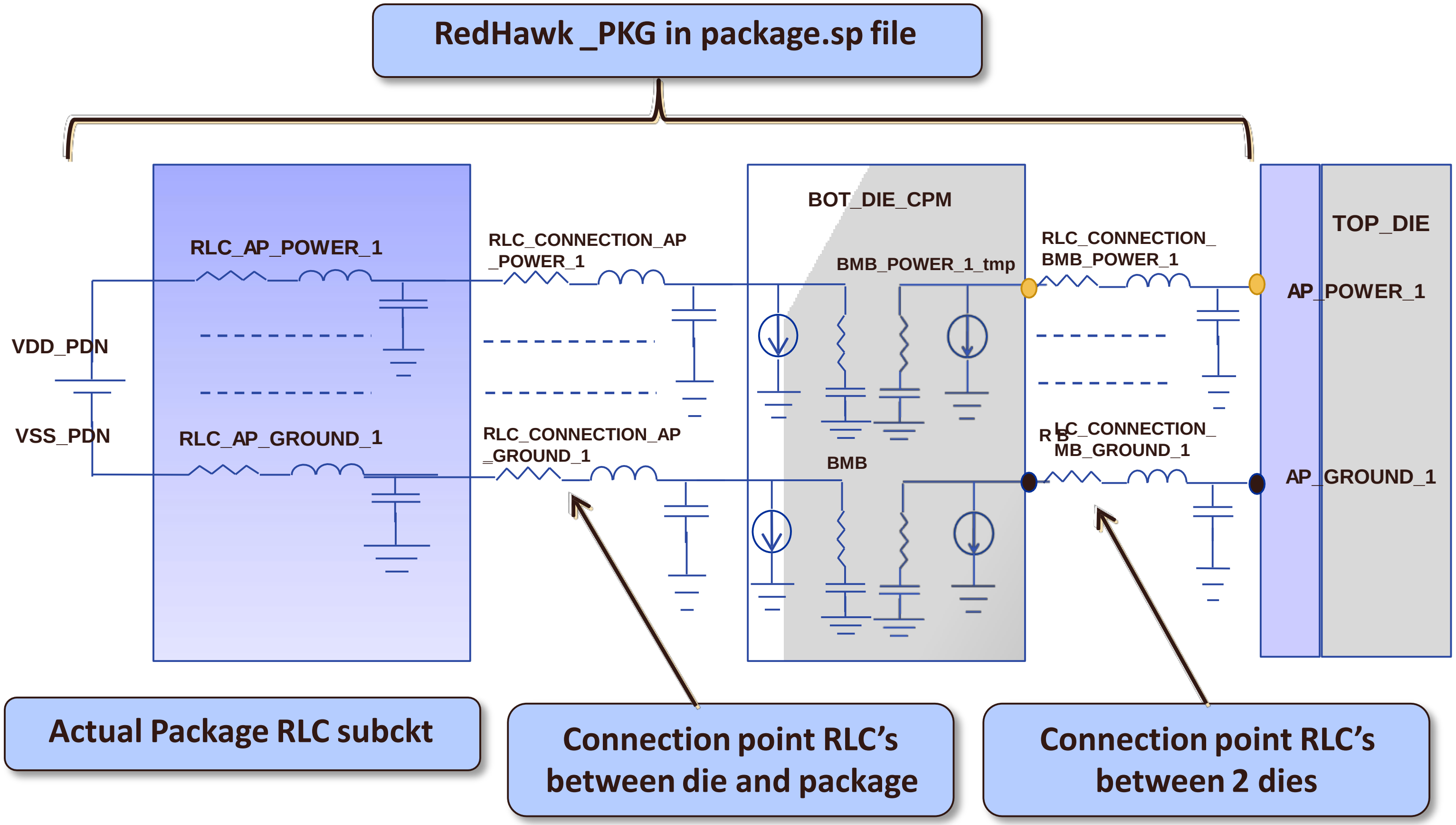
Top die CPM model being invoked inside the package.sp

Any connection point , copper piller RLC's between the dies to be included here

In this E.g. BMB ports are on bottom die hook to the top die

Example: Bottom die CPM hookup

Top die is under redhawk analysis, package model contains the bottom die CPM inside it hooked up to the package as depicted below:



Example: Bottom die CPM included inside the package

Package subckt needs to be modified so that it contains the additional ports from CPM and invokes CPM model from within.

package.sp file

```
.subckt REDHAWK_PKG BMB_GROUND_4 BMB_GROUND_3 BMB_GROUND_2 BMB_GROUND_1
ROUND_12 BMB_GROUND_11 BMB_GROUND_10 BMB_GROUND_9 BMB_GROUND_16 BMB_GROU
 BMB_POWER_1 BMB_POWER_6 BMB_POWER_5 BMB_POWER_4 BMB_POWER_9 BMB_POWER_8


.include 't1_cpm.spi'

XadsPowerModel AP_POWER_1 AP_POWER_2 AP_POWER_3 AP_POWER_4 AP_POWER_5 AP
POWER_11 AP_POWER_12 BMB_POWER_1_tmp BMB_POWER_2_tmp BMB_POWER_3_tmp BMB
tmp BMB_POWER_8_tmp BMB_POWER_8_tmp BMB_POWER_10_tmp BMB_POWER_11_tmp BM
OUND_4 AP_GROUND_5 AP_GROUND_6 AP_GROUND_7 AP_GROUND_8 AP_GROUND_9 AP_GR
14 AP_GROUND_15 BMB_GROUND_1_tmp BMB_GROUND_2_tmp BMB_GROUND_3_tmp BMB_G
7_tmp BMB_GROUND_8_tmp BMB_GROUND_9_tmp BMB_GROUND_10_tmp BMB_GROUND_11_
 BMB_GROUND_15_tmp BMB_GROUND_16_tmp  AP_GROUND_16 adsPowerModel


*** DIE connection point RLC ****

R_connection_BMB_GROUND_4 BMB_GROUND_4 tmp1 0
L_connection_BMB_GROUND_4 tmp1 BMB_GROUND_4_tmp 0
C_connection_BMB_GROUND_4 BMB_GROUND_4_tmp 0  0

R_connection_BMB_GROUND_3 BMB_GROUND_3 tmp2 0
L_connection_BMB_GROUND_3 tmp2 BMB_GROUND_3_tmp 0
C_connection_BMB_GROUND_3 BMB_GROUND_3_tmp 0  0
```

Ports from bottom die CPM connecting to top die

Bottom die CPM model being invoked inside the package.sp

Any connection point , copper piller RLC's between the dies to be included here

In this E.g. BMB ports are on bottom die hook to the top die