MEDIATEK
Connectivity Chip Design

snug

PowerReplay :
Automated Name Mapping for DV Checker using PowerReplay

MediaTek : Chia-Cheng Wu
Synopsys : Marty Huang, Lisa Hsieh

Agenda

- Introduction
- Design verification by checker
- Pain Points
- New Method introduction : PowerReplay
- PowerReplay Flow
- Enhancements from PowerReplay
- Conclusion
- Future work

Introduction

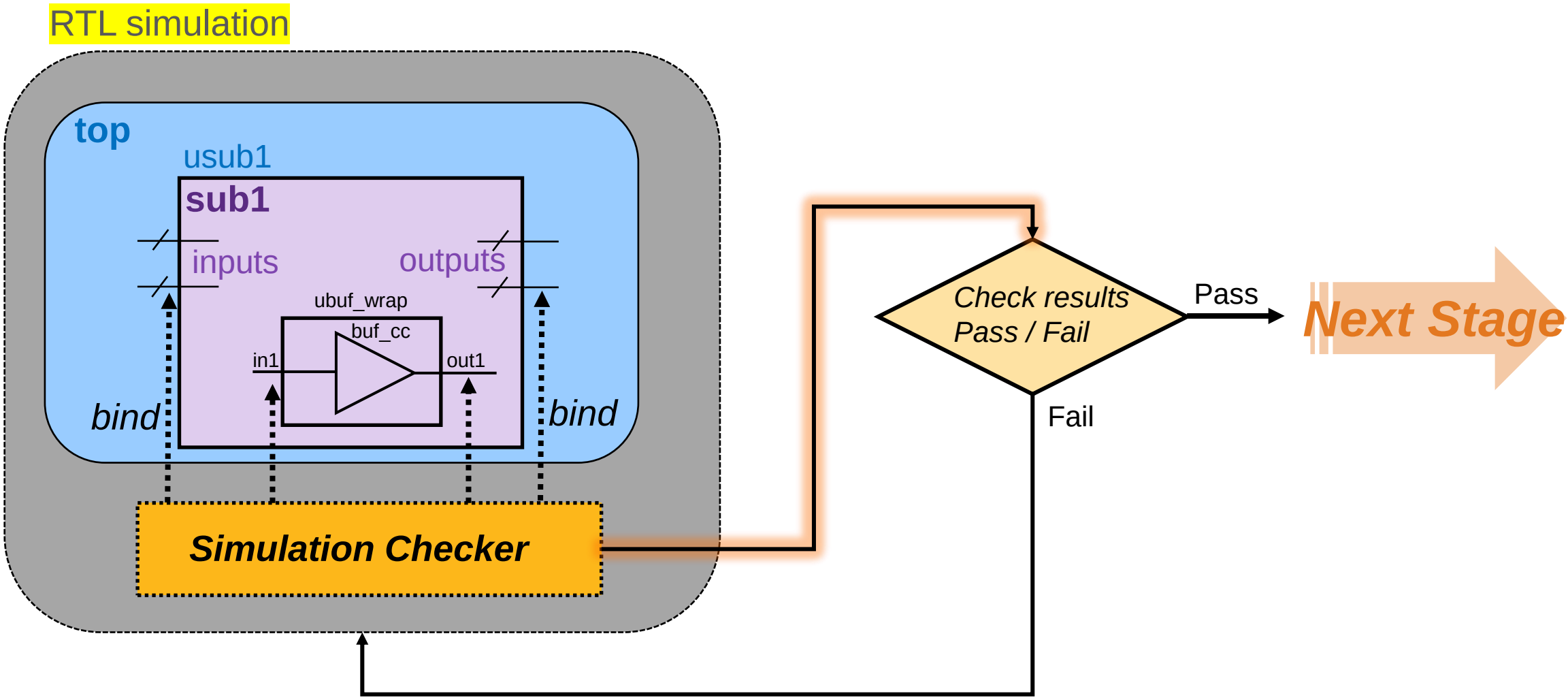
❑Background

- After RTL coding and synthesis to gate level, designer want to check both designs' equivalence and meet the design expectations.
- The design checkers need the names of instances in both RTL and gate level.
- Pain Point: It takes time for designer to search for each instances' name from both RTL and gate level. Especially the names are not exactly same in both structure.

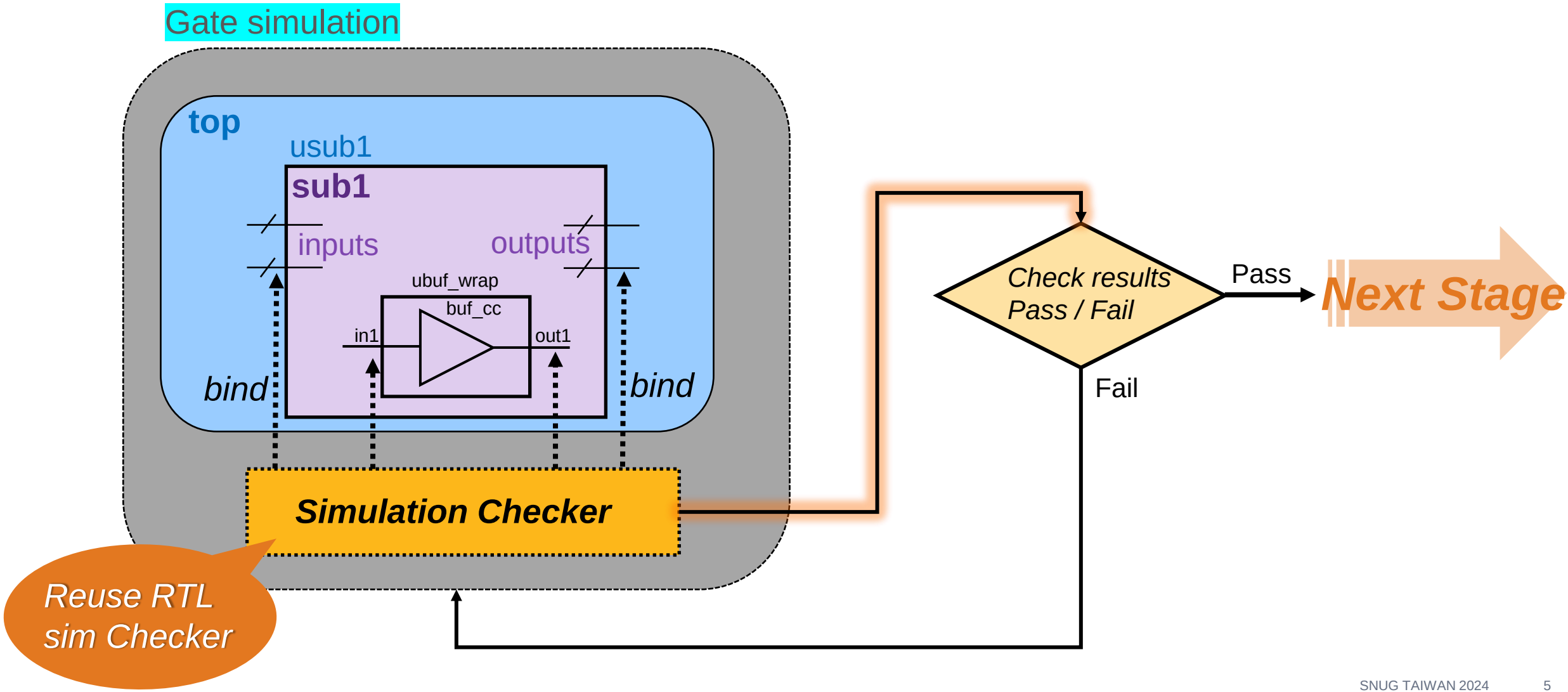
❑VCS Design Checker

- Design Checker is a static code purification tool in VCS
- It helps to identify common and subtle coding mistakes
- The tool contains approximately 500+ rules that perform semantic analysis and check for coding styles that may cause problems in the simulation and synthesis flow

The Design Verification By Checker (RTL stage)

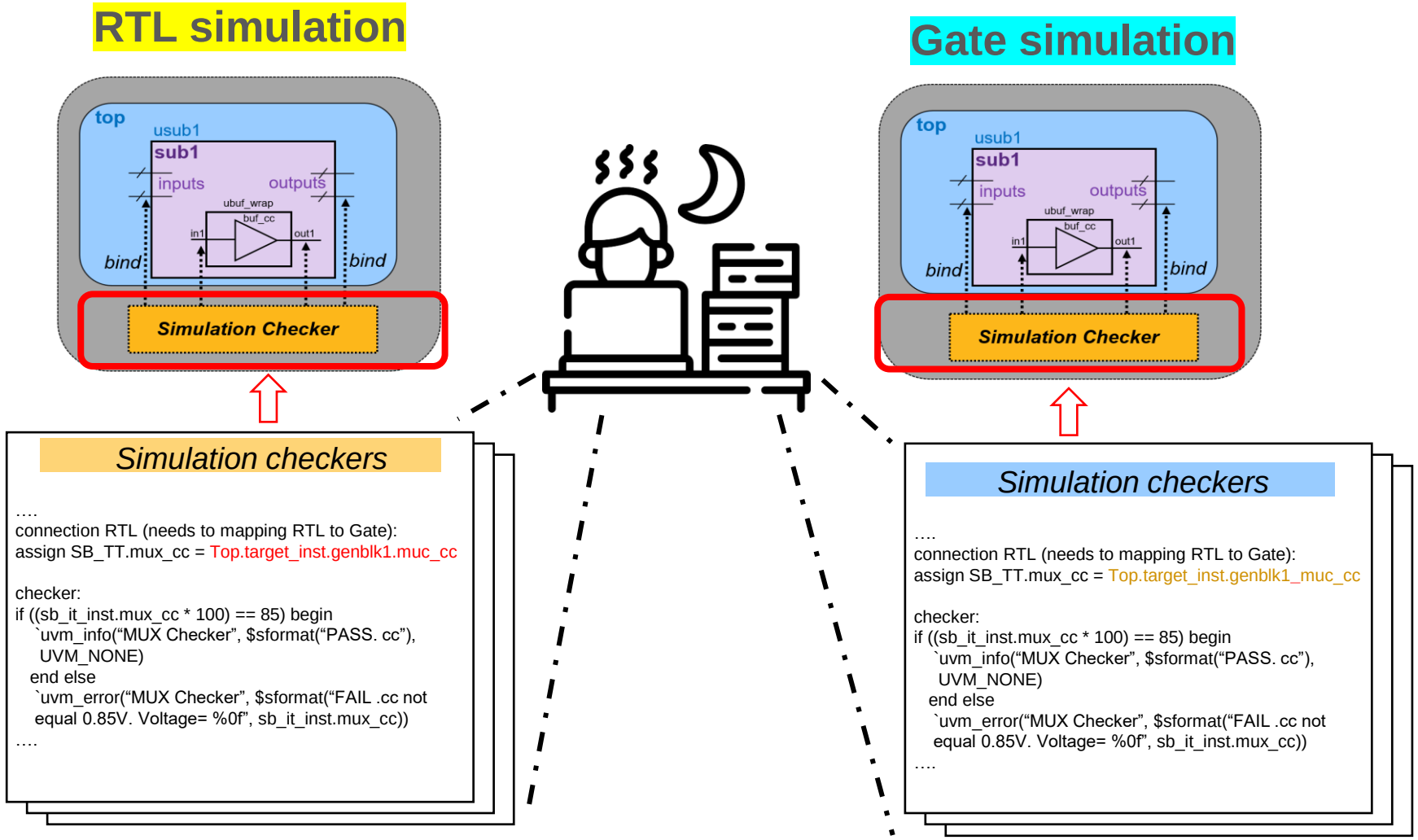


The Design Verification By Checker (Gate Level)



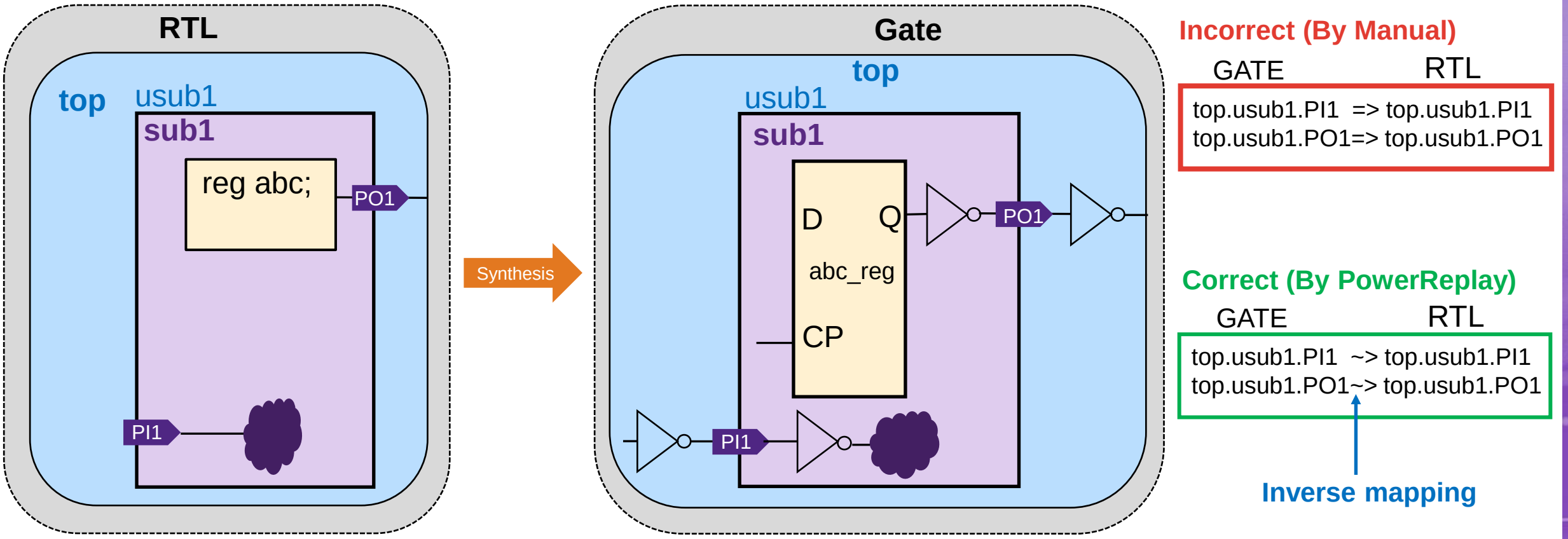
Pain Point 1 -- Occupied Human Resource

- Modifying checkers from RTL to Gate by manual takes time and engineer efforts.
- Especially the names are not exactly same in both stage.



Pain Point 2 -- Correctness

- Some signals the RTL and Gate have the same name, but function is different as below scenarios, modify by manual may be incorrect.
- Using PowerRelay can avoid these kinds of polarity issues.



Incorrect (By Manual)

GATE	RTL
top.usub1.PI1	=> top.usub1.PI1
top.usub1.PO1	=> top.usub1.PO1

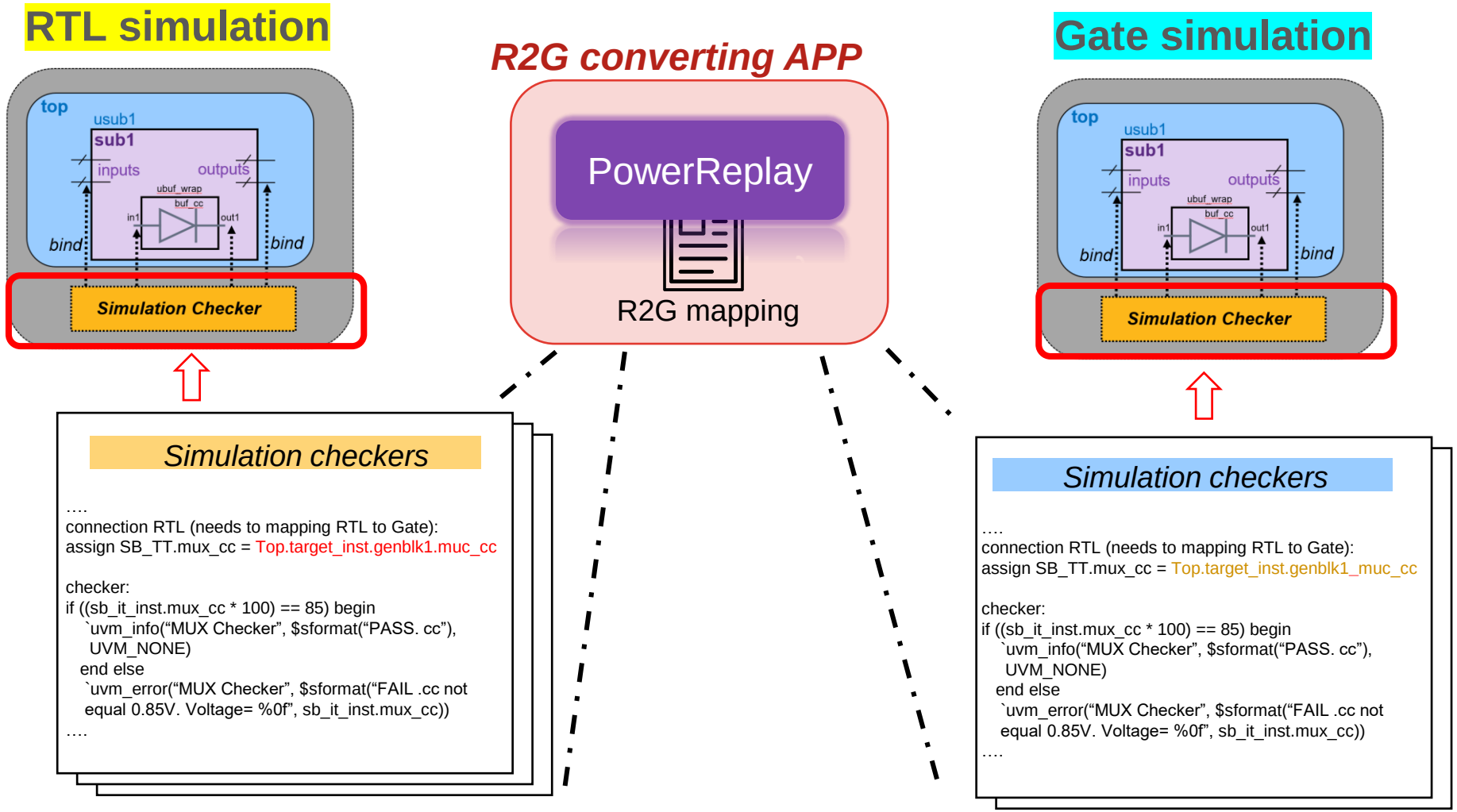
Correct (By PowerReplay)

GATE	RTL
top.usub1.PI1	~> top.usub1.PI1
top.usub1.PO1	~> top.usub1.PO1

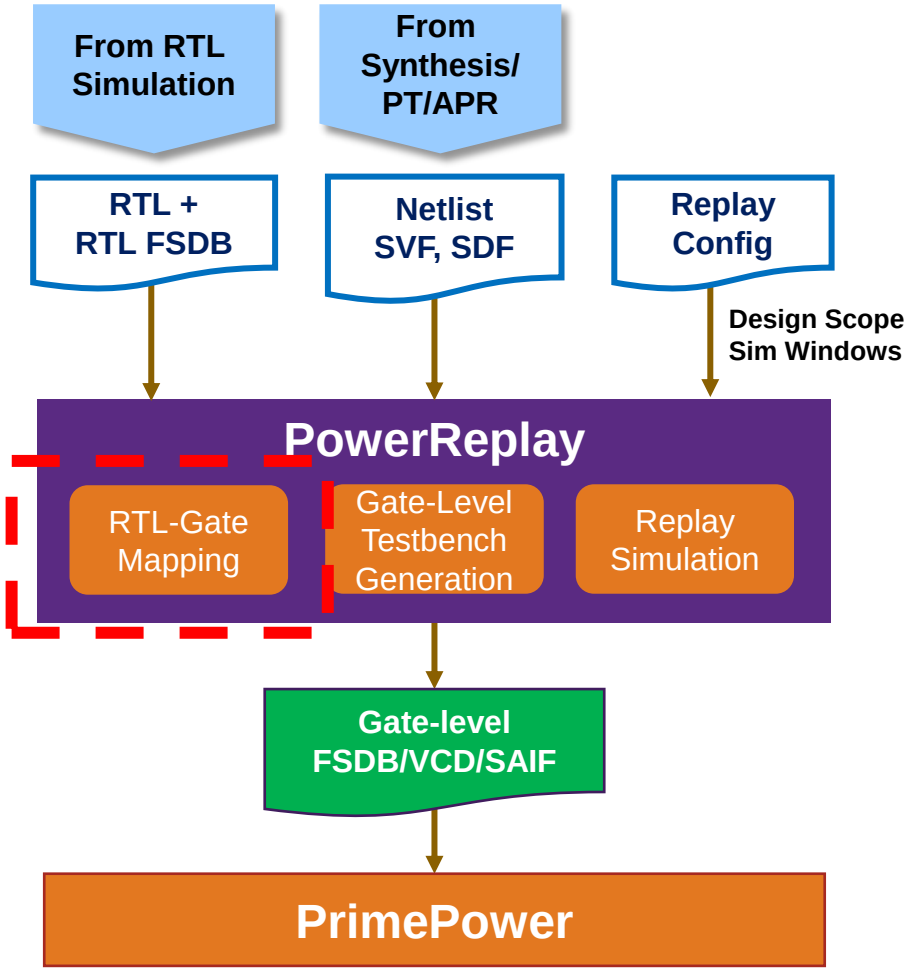
Inverse mapping

New Method

- **PowerReplay** provides an application of generating mapping file with G2R or R2G.
- Within mapping file generated, users can convert simulation checker's scripts efficiently.



PowerReplay Flow



1. RTL-Gate Mapping

- Use RTL KDB/RTL FSDB and netlist designs
- Leverage SVF from SNPS Synthesis Tools
- **Generate G2R mapping**

2. Replay Simulation

- Use RTL FSDB
- Specify sim. windows & design scope
- Auto-generate Gate-level TB
- Leverage SDF from PrimeTime
- Leverage Q pin delay from PrimeTime
- Generate Gate-level activity data (FSDB/VCD/SAIF)

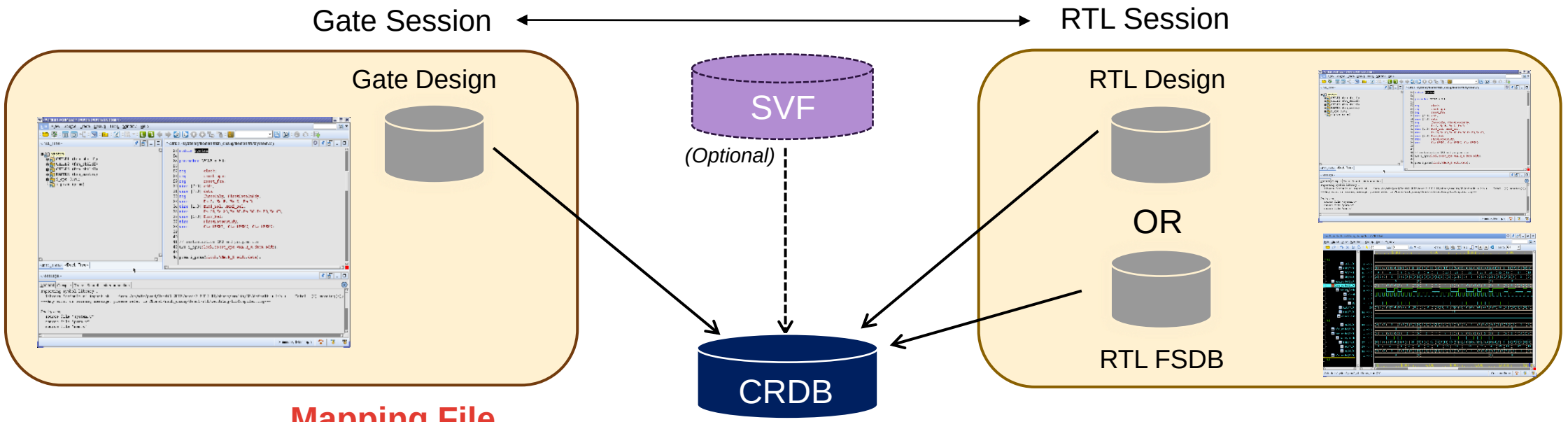
3. Power Analysis

- Feed FSDB/VCD/SAIF to PrimePower
- Compute power metrics
- Feed FSDB to IR-drop tools for analyzing and fixing IR-drop

Synopsys Confidential Information

PowerReplay Mapping Technology

- *RTL-Gate Mapping*



Mapping File

TOP.u_dut.u_inst1.cen_reg_1_.Q	=>	TOP.u_dut.u_inst1.cen[1]
TOP.u_dut.u_inst1.cen_reg_2_.Q	=>	TOP.u_dut.u_inst1.cen[2]
TOP.u_dut.u_inst1.cen_reg_3_.Q	~>	TOP.u_dut.u_inst1.cen[3]

Gate Signals RTL Signals

Automatic mapping of signals between RTL and Gate level design

Performance improvement

- The new method has applied to two cases.
- According to the time calculation as table below, the improvements have saved **83%** and **79%** of time.

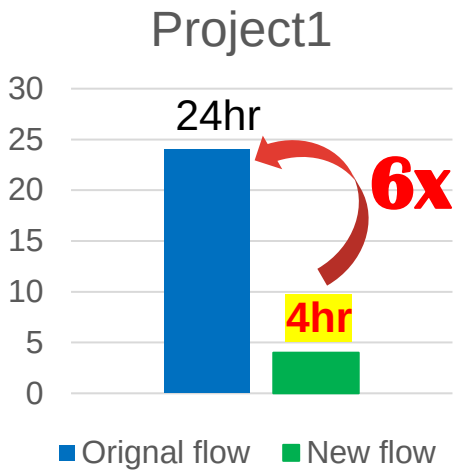
Run time original flow vs. new flow

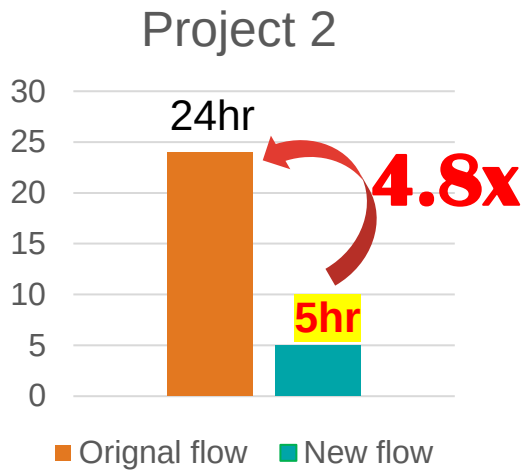
Project	Original Flow (manually map)	New Flow (auto-flow)	Saving time (%)
Project1	24 hrs	4 hrs	83%
Project2	24 hrs	5 hrs	79%

R2G converting APP

PowerReplay

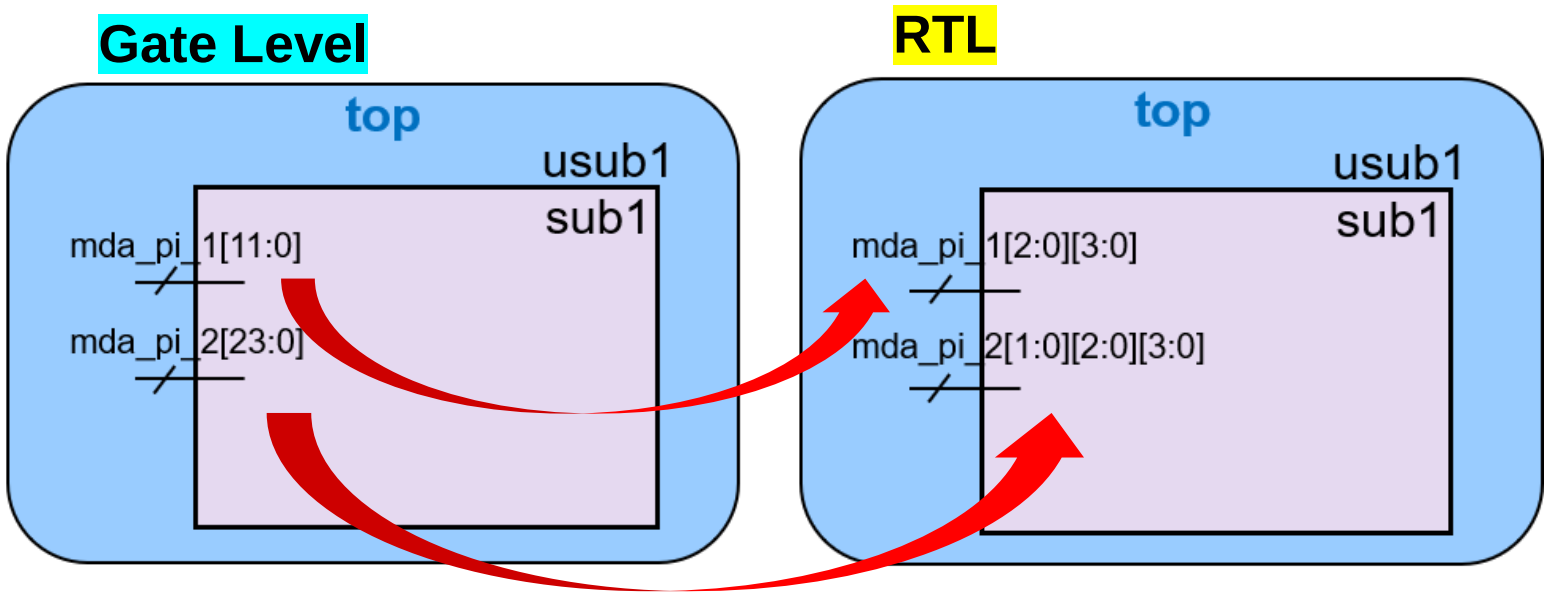
R2G mapping





PowerReplay Enhancement (1)

- **Multi-Dimension Array Primary Input mapping**
- Within Gate level named in 1-D, RTL ports named in multi-dimension. PowerReplay is able to detect the connection between these two structures.

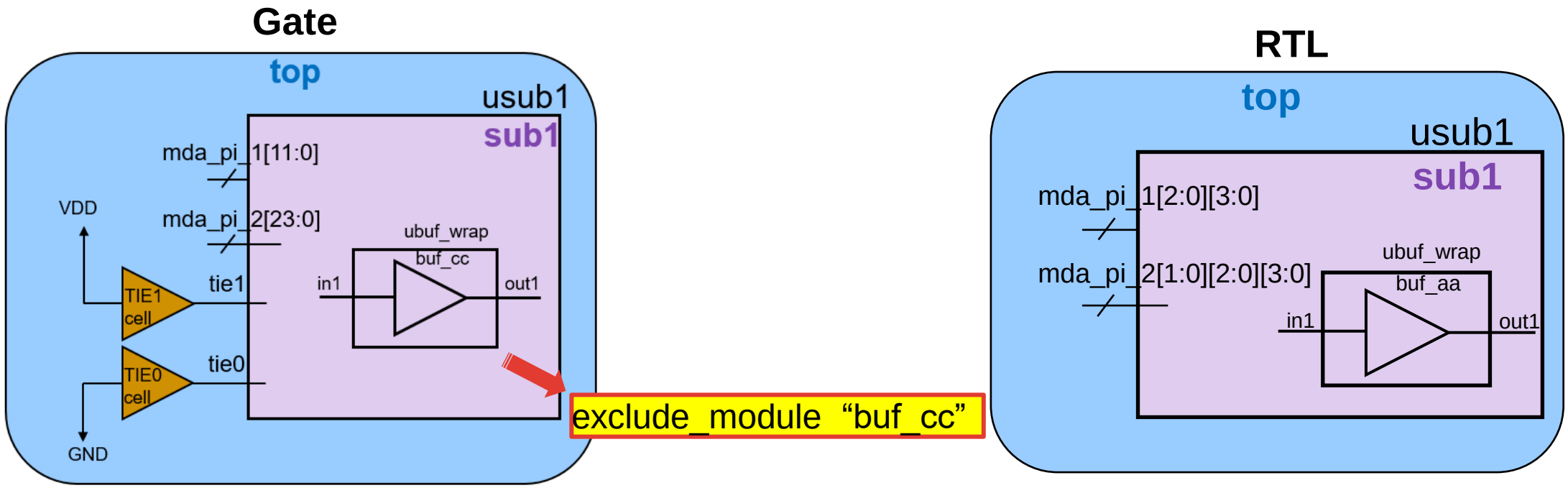


Expected Mapping

Gate		RTL
top.usub1.mda_pi_1[0]	=>	top.usub1.mda_pi_1[0][0]
top.usub1.mda_pi_1[1]	=>	top.usub1.mda_pi_1[0][1]
top.usub1.mda_pi_1[2]	=>	top.usub1.mda_pi_1[0][2]
top.usub1.mda_pi_1[3]	=>	top.usub1.mda_pi_1[0][3]
top.usub1.mda_pi_1[4]	=>	top.usub1.mda_pi_1[1][0]
top.usub1.mda_pi_1[5]	=>	top.usub1.mda_pi_1[1][1]
top.usub1.mda_pi_1[6]	=>	top.usub1.mda_pi_1[1][2]
top.usub1.mda_pi_1[7]	=>	top.usub1.mda_pi_1[1][3]
top.usub1.mda_pi_1[8]	=>	top.usub1.mda_pi_1[2][0]
top.usub1.mda_pi_1[9]	=>	top.usub1.mda_pi_1[2][1]
top.usub1.mda_pi_1[10]	=>	top.usub1.mda_pi_1[2][2]
top.usub1.mda_pi_1[11]	=>	top.usub1.mda_pi_1[2][3]
top.usub1.mda_pi_2[0]	=>	top.usub1.mda_pi_2[0][0][0]
top.usub1.mda_pi_2[1]	=>	top.usub1.mda_pi_2[0][0][1]
top.usub1.mda_pi_2[2]	=>	top.usub1.mda_pi_2[0][0][2]
top.usub1.mda_pi_2[3]	=>	top.usub1.mda_pi_2[0][0][3]
top.usub1.mda_pi_2[4]	=>	top.usub1.mda_pi_2[0][1][0]
top.usub1.mda_pi_2[5]	=>	top.usub1.mda_pi_2[0][1][1]
top.usub1.mda_pi_2[6]	=>	top.usub1.mda_pi_2[0][1][2]
top.usub1.mda_pi_2[7]	=>	top.usub1.mda_pi_2[0][1][3]
top.usub1.mda_pi_2[8]	=>	top.usub1.mda_pi_2[0][2][0]
top.usub1.mda_pi_2[9]	=>	top.usub1.mda_pi_2[0][2][1]
top.usub1.mda_pi_2[10]	=>	top.usub1.mda_pi_2[1][2][2]
top.usub1.mda_pi_2[11]	=>	top.usub1.mda_pi_2[1][2][3]
top.usub1.mda_pi_2[12]	=>	top.usub1.mda_pi_2[1][0][0]
top.usub1.mda_pi_2[13]	=>	top.usub1.mda_pi_2[1][0][1]
top.usub1.mda_pi_2[14]	=>	top.usub1.mda_pi_2[1][0][2]
top.usub1.mda_pi_2[15]	=>	top.usub1.mda_pi_2[1][0][3]
top.usub1.mda_pi_2[16]	=>	top.usub1.mda_pi_2[1][1][0]
top.usub1.mda_pi_2[17]	=>	top.usub1.mda_pi_2[1][1][1]
top.usub1.mda_pi_2[18]	=>	top.usub1.mda_pi_2[1][1][2]
top.usub1.mda_pi_2[19]	=>	top.usub1.mda_pi_2[1][1][3]
top.usub1.mda_pi_2[20]	=>	top.usub1.mda_pi_2[1][2][0]
top.usub1.mda_pi_2[21]	=>	top.usub1.mda_pi_2[1][2][1]
top.usub1.mda_pi_2[22]	=>	top.usub1.mda_pi_2[1][2][2]
top.usub1.mda_pi_2[23]	=>	top.usub1.mda_pi_2[1][2][3]

PowerReplay Enhancement (2)

- Originally, mapping file mismatched in buffer pins
- Enhancement: exclude module map for buffer wrapper



Original Mapping (Unexpected):

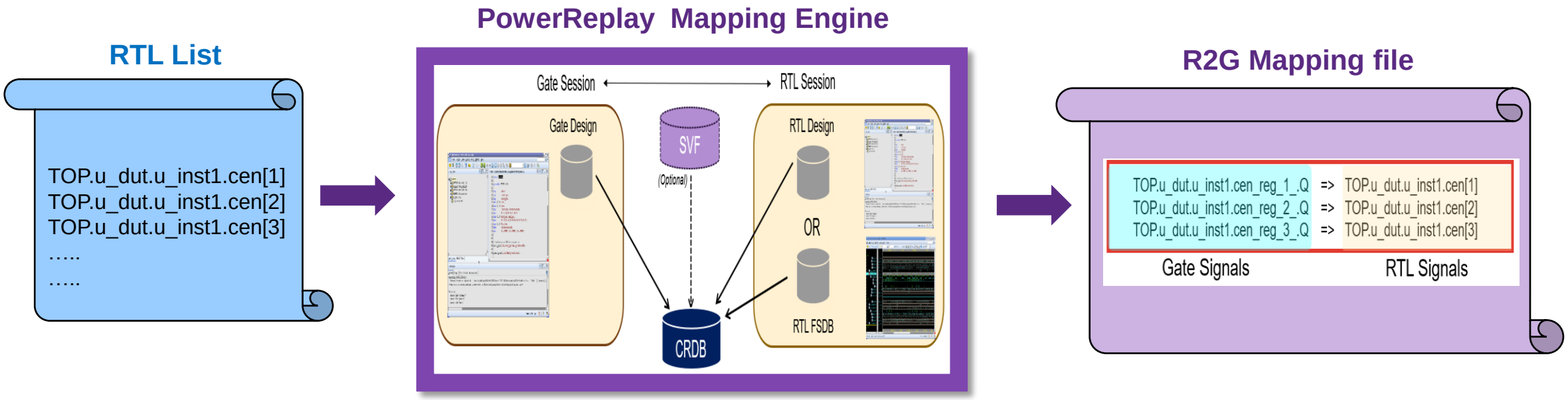
top.usub1.ubuf_wrap.in1 => top.usub1.ubuf_wrap.in1
top.usub1.ubuf_wrap.out1 => top.usub1.ubuf_wrap.in1

New enhancement (Expected Mapping) :

top.usub1.ubuf_wrap.in1 => top.usub1.ubuf_wrap.in1
top.usub1.ubuf_wrap.out1 => top.usub1.ubuf_wrap.out1

PowerReplay Enhancement (3) -- On-going

- Within RTL list, PowerReplay mapping engine can search for correspond points from gate automatically.
- This enhancement can save run time compared to generating the mapping file with whole design .



Conclusions

- Utilizing the PowerReplay Mapping Engine in conjunction with the MTK R2G converting App can reduce the Gate DV sim preparation time by approximately 5 ~ 6 times

- The Gate DV sim preparation time can be further reduced once the mapping by RTL list feature is fully developed.

- Adopting this flow not only reduces the Gate DV sim preparation time and engineer effort but it also helps avoid potential correctness issues

Future Works

- The runtime is too long for mapping a specific target
  - The runtime is equivalent to processing the entire design because the tool maps all essential points internally, regardless of whether the target scope is the top-level or a sub-block.

- The tool is currently unable to properly map the wire declaration points in the RTL
  - Sometimes the DV engineers do not insert checker on module boundaries or registers. Instead, they insert checkers at wire declaration points, which PowerReplay can’t map them.


THANK YOU
Our Technology, Your Innovation™