Comprehensive, and Automated IP and SoC Connectivity Verification Signoff using Formal DV Technique

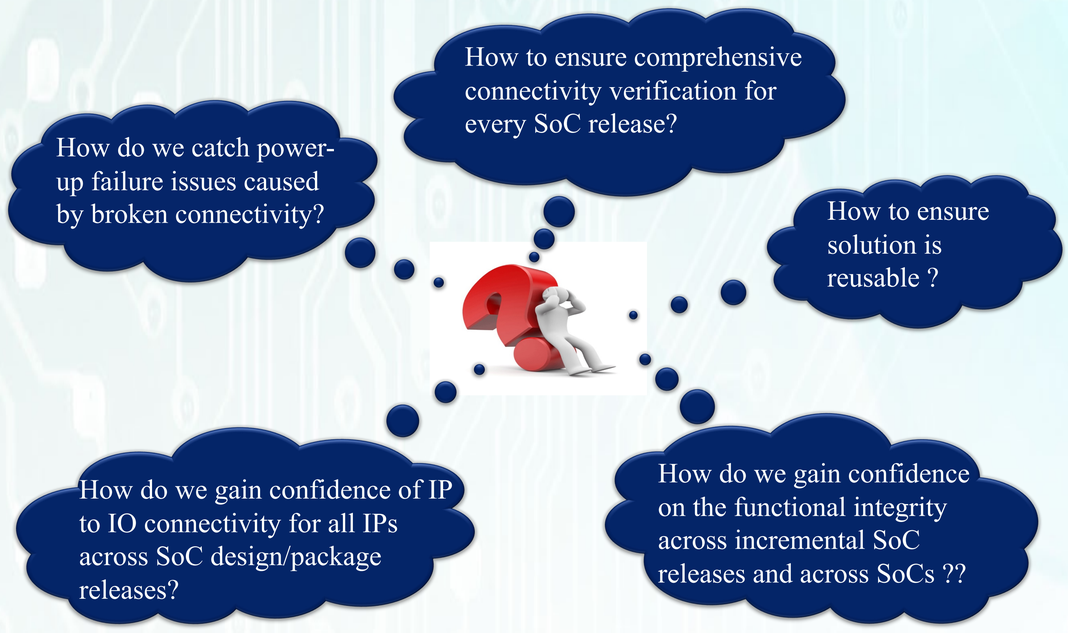
Abhinav Parashar, Parthasarathy Ramesh, Harish Maruthiyodan, Prasanth Kumar Narava , Ayush Shishir Jodh,

Ruchi Shankar

60 DESIGN AUTOMATION CONFERENCE
FROM CHIPS TO SYSTEMS — LEARN TODAY, CREATE TOMORROW

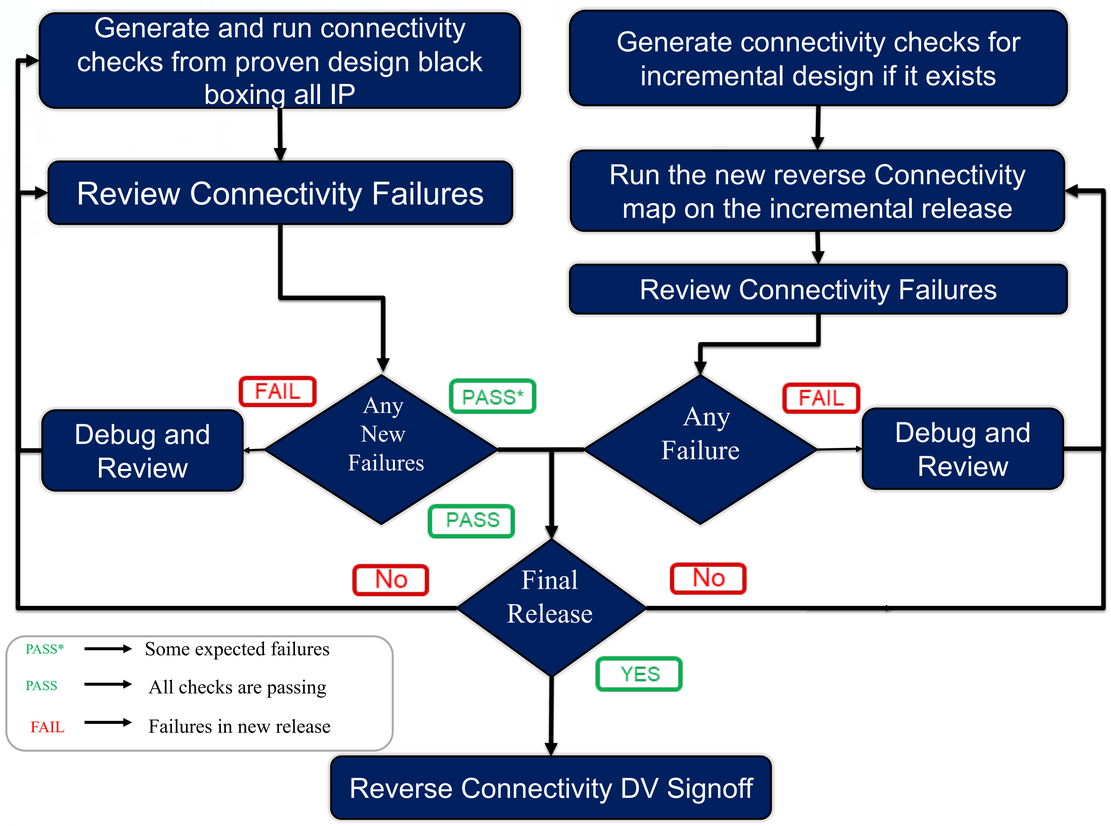
1. Motivation & Problem Statement

The motivation is to come-up with a methodology to prove Connectivity Correctness comprehensively across SoC releases and across SoC designs in a platform (with minimal changes)



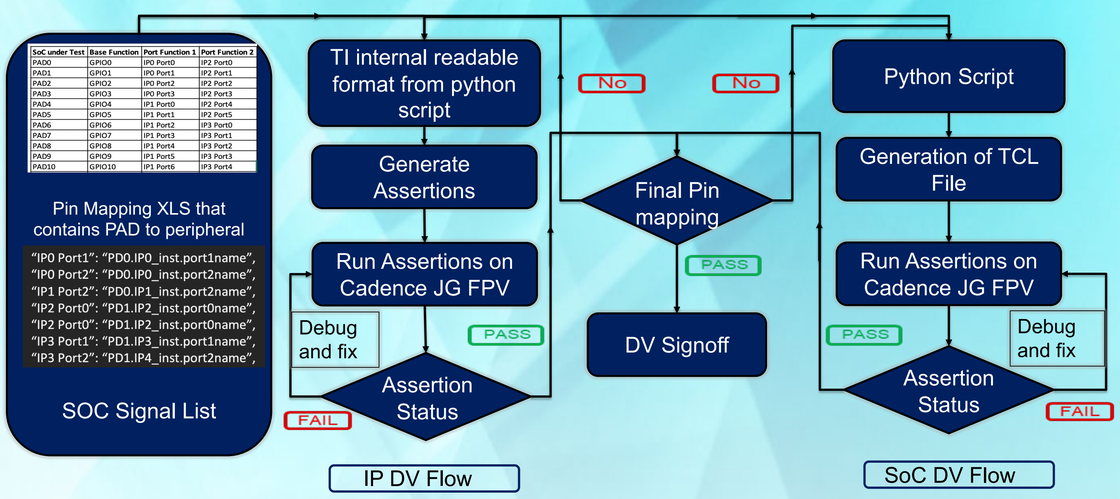
- SoC and IP Design **Cycle time reduction results** in **cost savings, early samples to customers**
- It becomes **incumbent to verify Connectivity** within a **SoC** and **across SoCs spins** with a reduced cycle time
- It is important to look for a **re-usable, automated, and scalable solution** to tackle connectivity verification
- This gives rise to **two verification methodologies** that expedites Connectivity Verification - **Reverse Connectivity & IO PinMuxing DV**

2. Reverse Connectivity solution using Formal Verification technique for verification at SoC level(1/3)

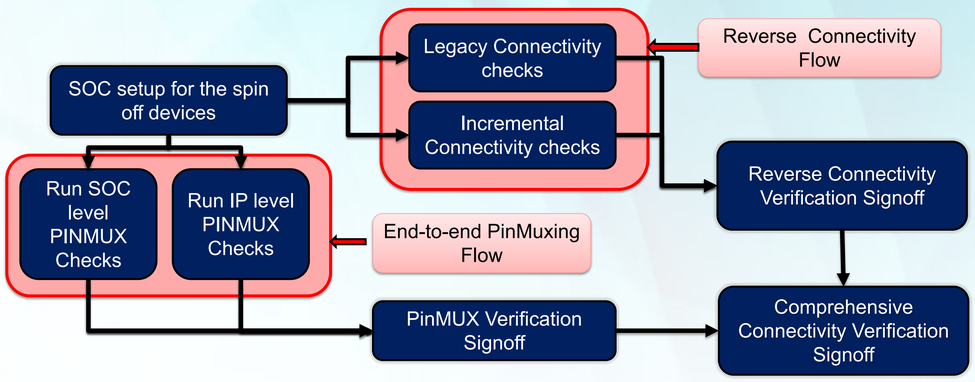


- ✓This flow ensures that connectivity of **feature set from Proven Silicon Design** and that of SoC2 is **unaffected with incremental updates** .
- ✓The flow is achieved using the JasperGold Connectivity app, we **Generate the Connectivity Checks** on the SoC taking each hierarchy as target point until the last black-boxed IP hierarchy
- ✓The **Legacy** run **summary should remain same** throughout incremental releases once SoC design is stable. The Flow is run till the **RTL freeze** happens

3. PinMuxing DV solution using Formal Verification technique for IO Controller verification at IP and SoC level(2/3)

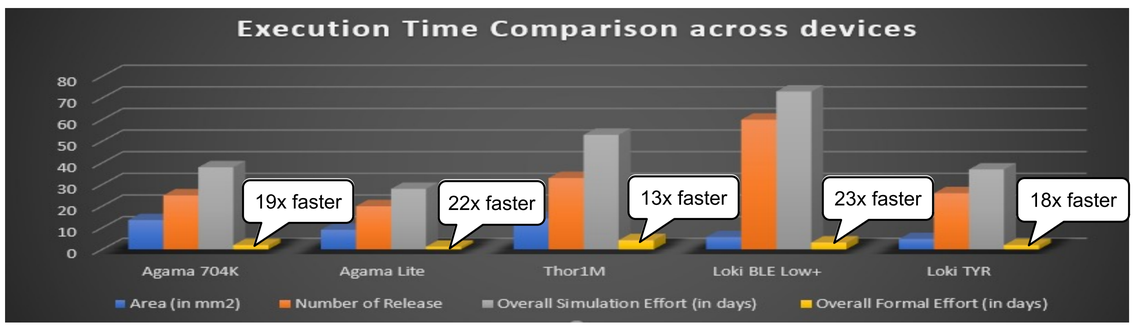


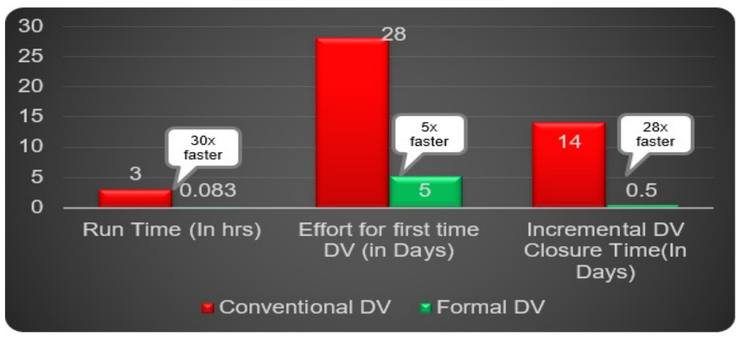
4. Comprehensive Connectivity Verification Flow(3/3)



- PinMuxing DV and Reverse Connectivity flows incorporated into a **single Comprehensive Connectivity Verification Methodology**
- First Perform the **IP level PinMUX checks** followed by **SOC level PinMUX checks** Debug for failures if any, and if absent **signoff PinMux Verification**
- **Generate** the **Legacy** and **incremental** connectivity verification setup. Debug failures for the Reverse Connectivity flow, if absent **signoff Connectivity Verification**

5. Evidence and Results





A very significant improvement can be seen in results in both the approaches thus reducing the overall required DV time

7. Conclusions

- **Efficiency:** **Improves the cycle time** of the overall connectivity verification
- **Quality:** **No compromise in quality** with a formal based verification approach
- **Impact on Silicon:** **(No connectivity issues seen in Silicon Bring up for 4 devices across 2 platforms)**
- **In Summary:**
  - Able to **prove & verify PinMuxing** successfully on **4 devices** on **two platforms**, with the strategy planned to be adopted for all other ongoing (**4 SoCs**)/future devices in the platform
  - Able to **achieve 30x faster verification closure time** for **Comprehensive Connectivity verification**, in terms of effort incurred as well as run time incurred

Acknowledgements

TEXAS INSTRUMENTS