

CDC HANDSHAKE ADVANCED SIGNOFF METHODOLOGY

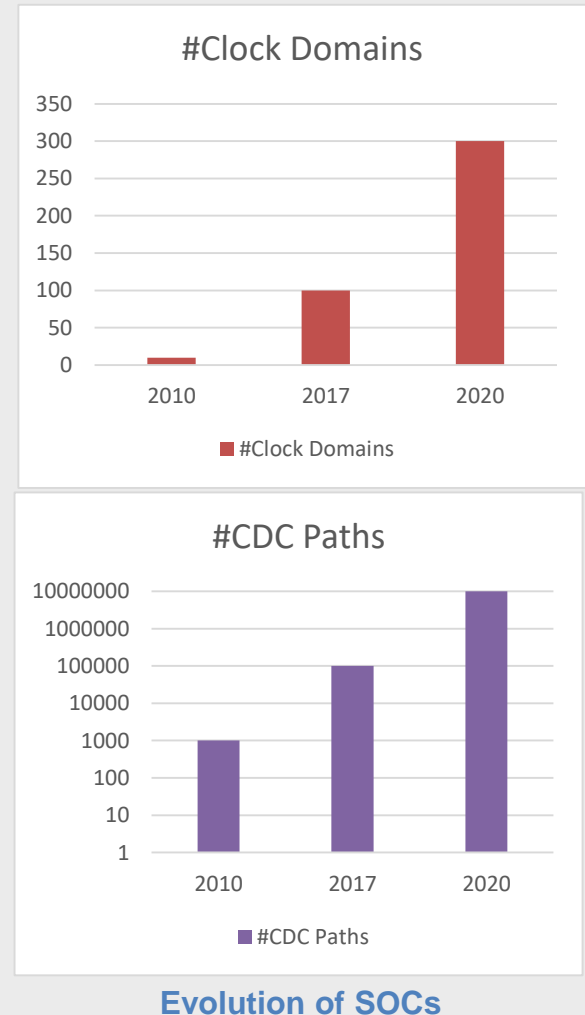


LOUIS CARDILLO
PRAKASH MUKRE
RAMAKRISHNA RAYAPROLU
VIVEK GUPTA
VIKAS SACHDEVA



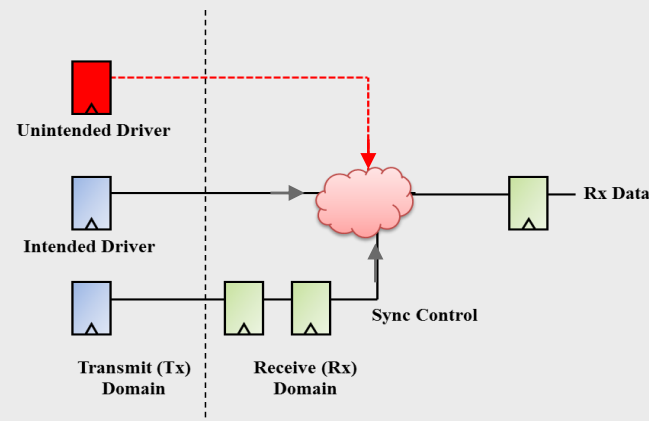
Motivation

- SOCs are extremely complex
 - Numerous (> 100!) asynchronous clock domains
 - *Millions* of clock-domain crossing paths
- Handshake Mechanisms (CDC Interfaces) are common design patterns for faithful data transfer across domains
 - Typical SOC contains *thousands* of such interfaces
- *Every single* interface **must be** reviewed for sign-off
 - Reviewing each interface is a colossal sign-off effort
 - Not reviewing all may cause CDC issues to be missed
- All the effort notwithstanding, today's interface review process is not entirely flawless
 - CDC tools miss bugs

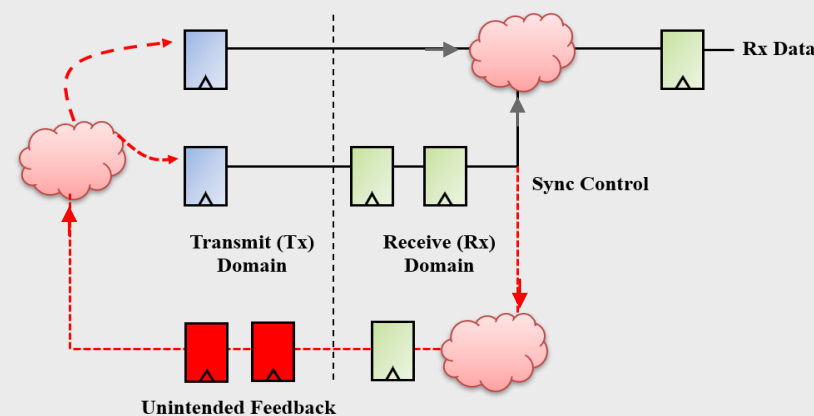


Structural Signoff - Current Methodology Limitations

- Imperfect Sign-Off In State-of-Art Tools
- Most of current flows rely on Structural Signoff
- **Structural Sign-off**
 - Based on Control/Qualifier signal associated with driver data-signal and feedback
- Flows can miss CDC errors in complex handshake scenarios
 - Signals might get classified *incorrectly* as safe purely based upon structural principles
 - Unintended feedback associations can be made for closed loop CDC transfers



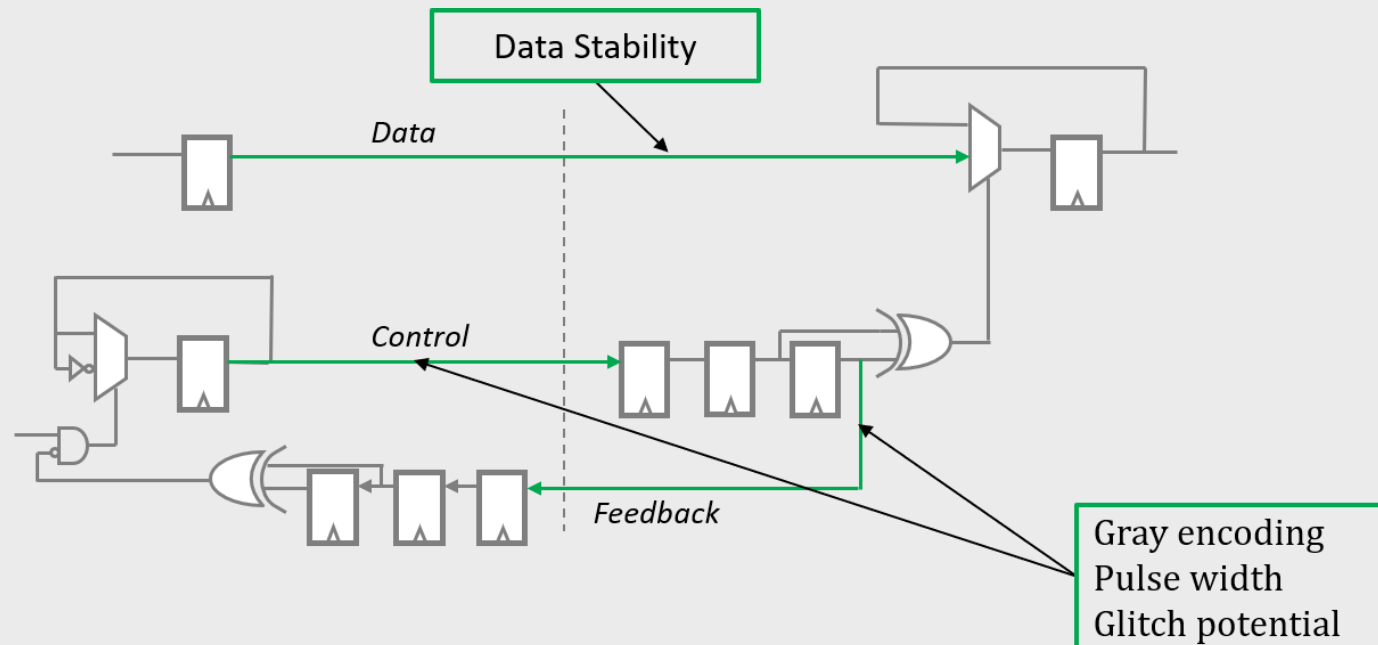
Unintended driver can be classified as safe purely based upon structural principles



Missing feedback error can be missed because of unintended feedback

Formal Signoff - Methodology Challenges

- To strengthen structural signoff, formal signoff is used
- **Formal Sign-off**
 - Based on sequential proof that control path qualification is always safe
- Formal sign-off removes limitations of structural sign-off
- Involves huge effort for 100% signoff



Advanced CDC Handshake Sign-Off

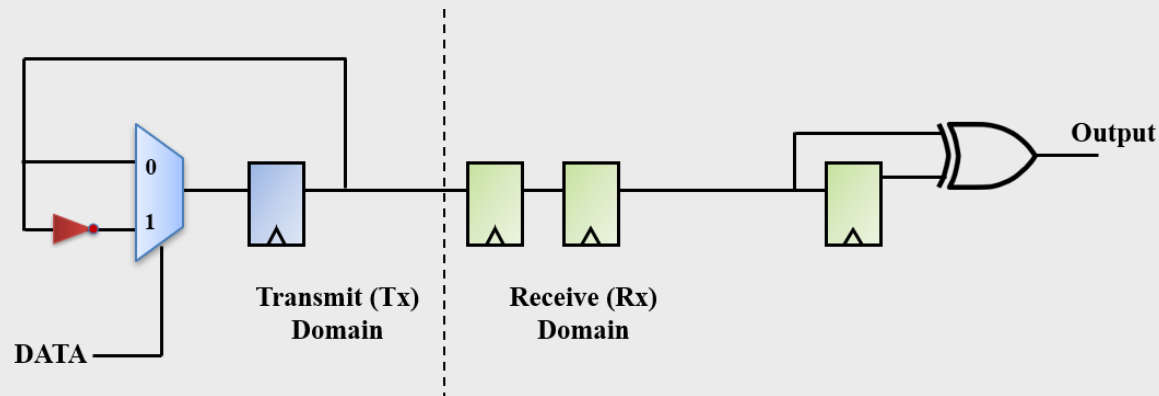
New Methodology Requirements

- Should supplement the current structural sign-off flow
- Should be conservative enough not to hide any potential CDC issues
- Able to guarantee 100% sign-off
- Intelligent enough to not increase sign-off effort
- Portable across multiple designs
- Ease of implementation and integration

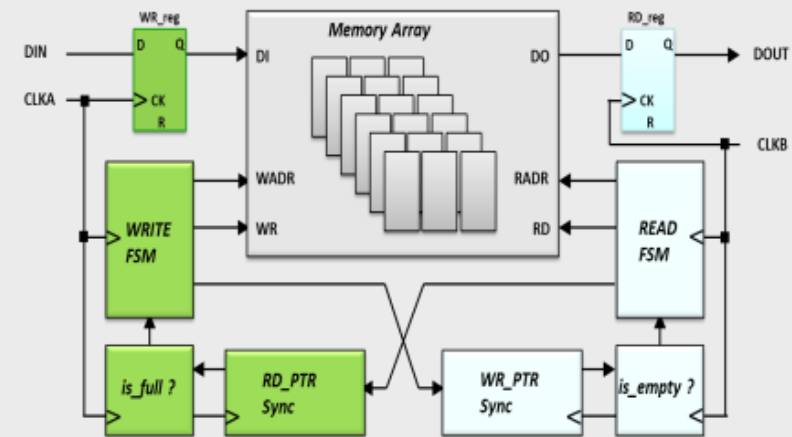
CDC Handshake Sign-Off – *New Methodology*

Interface Library Creation

- For library creation most-widely used schemes were identified
 - Pulse-synchronizer based handshake
 - FIFO-based handshake
- Flow scripts created
 - Detect presence of these components
 - Automatically plug-in create association commands for Meridian CDC
 - create_association is tool specific command to define a pre defined interface



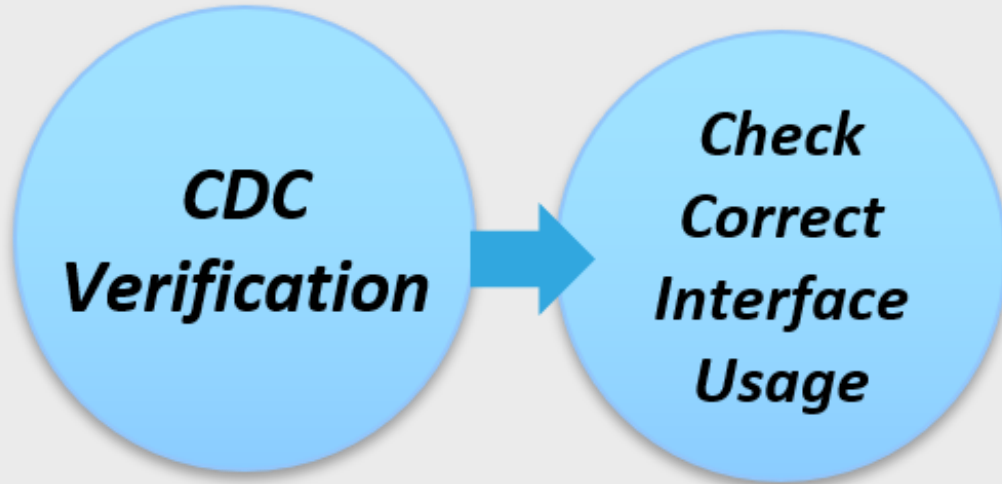
Pulse synchronizer based handshake



FIFO Handshake

CDC Handshake Sign-Off – *New Methodology*

Interface Verification

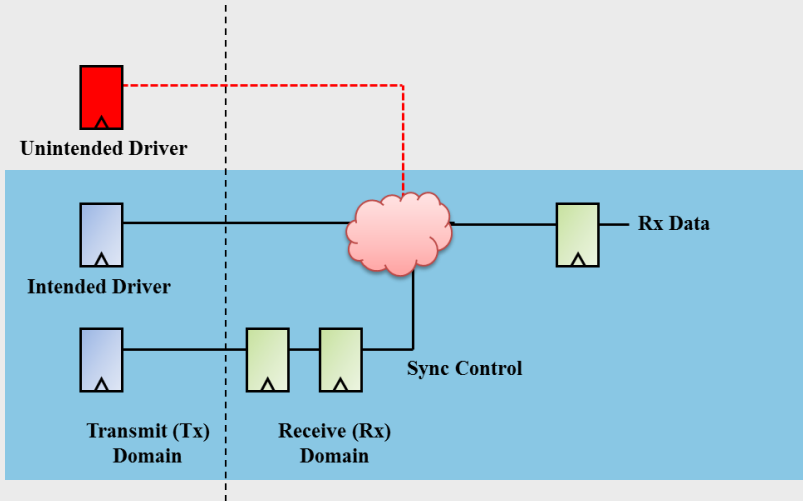


- **create_association commands are auto applied in the flow when library interface is instantiated**
- **Thorough structural analysis on the create_association constraint done even though library interfaces were pre-verified**

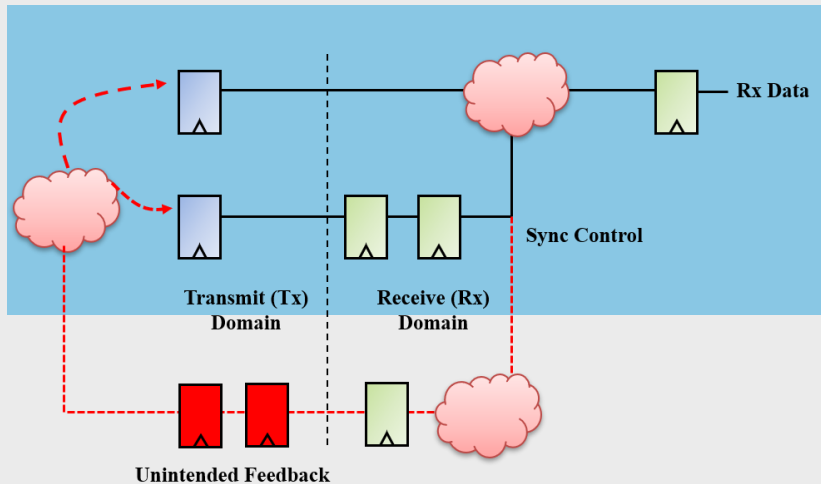
- Flow auto-applies create_association commands for these known structures in Meridian CDC
- Assumptions:
 - Detailed interface verification is done at interface library level
 - Top level only needs to verify structural instantiation
- Meridian CDC verifies create_association:
 - All handshake (TX, RX, Feedback) components
 - No issues in structural associations of components
 - Any unintended drivers or receivers in the interface
 - Any unintended parts, synchronizers which differ from library specification

CDC Handshake Sign-Off – *New Methodology*

Interface Review



Intended interface region as specified by library



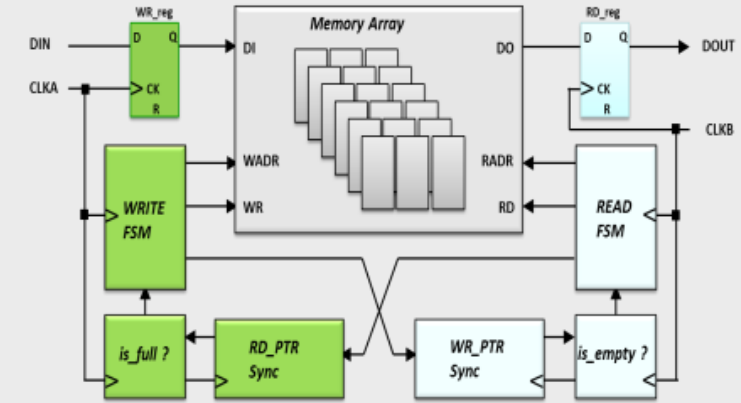
Intended interface region as specified by library

- After tool's analysis
 - Unintended drivers are classified as Errors in handshake
 - Unintended feedback or unintended flops are also classified as Errors
- Users need to review the Errors and
 - Fix the design interface
 - Add appropriate constraints in the tool if unintended parts are deemed to be safe

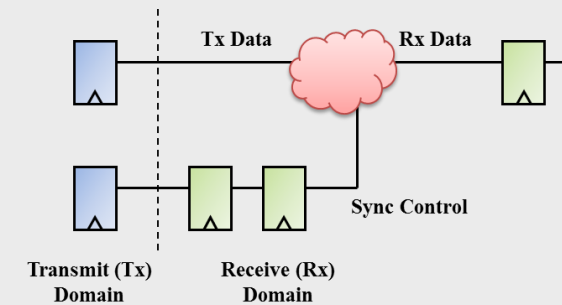
CDC Handshake Sign-Off – *New Methodology*

Interface Review

- After tool has completed analysis
 - Interfaces instantiated as per library specification and are accepted by the tool are classified as INFO
 - No further action needed from user
- Hand crafted or IP-specific interfaces are classified as Warning
 - Users need to run additional checks (formal CDC, simulation etc.) on these
 - After verification, users should add create_association commands so they become part of IP-specific local interface library



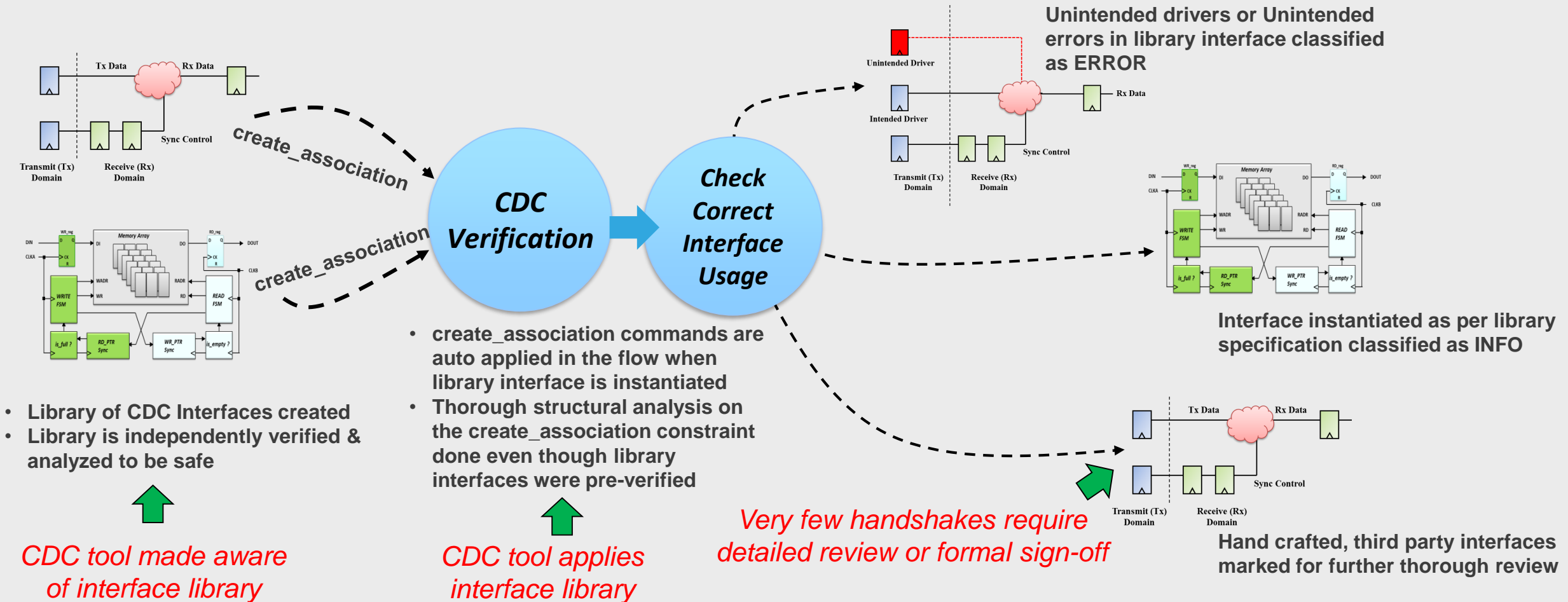
Interface instantiated as per library specification
classified as INFO



Handcrafted, third party interfaces
marked for further thorough review

CDC Handshake Sign-Off – *New Methodology*

Complete Flow



Results

Units with good ($\geq 50\%$) coverage: 25/55				
Design	Total # of Interfaces	#of library interfaces	#of non-library interfaces	%ge coverage
IP1	11	11	0	100
IP2	14	14	0	100
IP3	9	8	1	88.8
IP4	347	296	51	85.3
IP5	735	609	126	82.8
IP6	12	9	3	75.0
IP7	23	17	6	73.9
IP8	173	120	53	69.3
Future work: Sample Units for which coverage can be improved				
IP9	127	24	103	18.9
IP10	371	59	312	15.9
IP11	91	6	85	6.6
IP12	38	1	37	2.6
IP13	88	2	86	2.2
IP14	16		16	0.00

- Initial results showed
 - $> 85\%$ coverage on smaller IPs
 - $> 50\%$ coverage bigger IPs
- Some IPs have $< 50\%$ coverage
- Future work
 - Add more library components to improve the coverage
 - Analyze if some IPs are exceptions to interface principles

Summary

- Handshake mechanism (Interfaces) sign-off is one of the most important aspects of CDC verification.
- Structural Sign-off in state-of-art tools is imperfect and can generate false positives
- Formal sign-off requires huge sign-off effort
- New Interfaces sign-off methodology provides scale and accuracy
 - Is conservative to avoid any potential CDC issues
 - Able to guarantee 100% sign-off
 - Intelligent enough to not increase sign-off effort