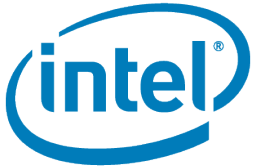
2017 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION UNITED STATES

Comprehensive and Automated Static Tool Based Strategies for the Detection and Resolution of Reset Domain Crossings

Yossi Mirsky (joseph.mirsky@intel.com)

Intel Corporation

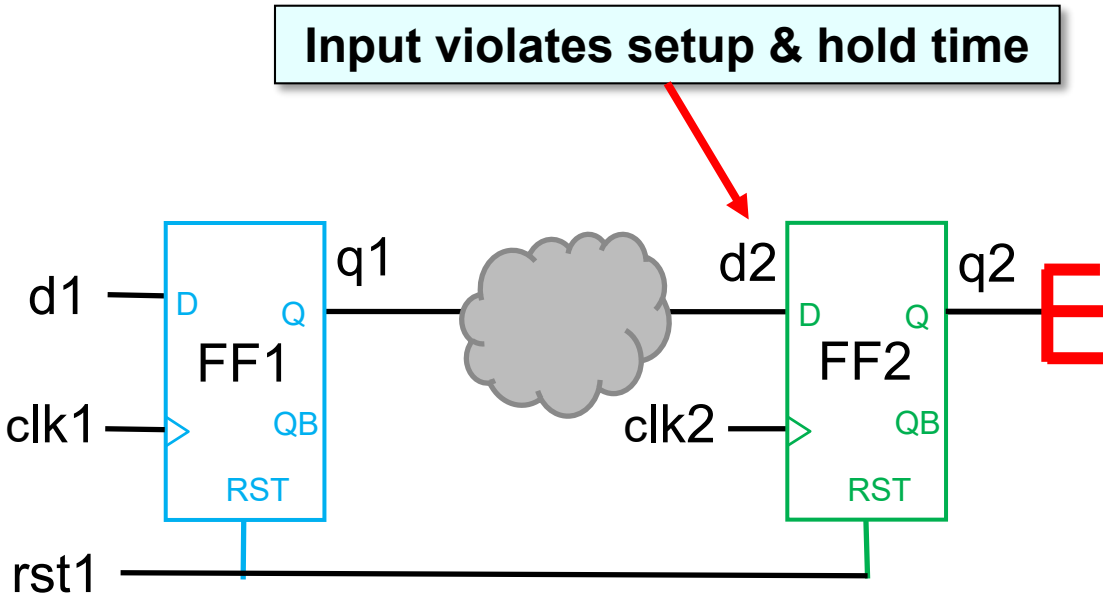


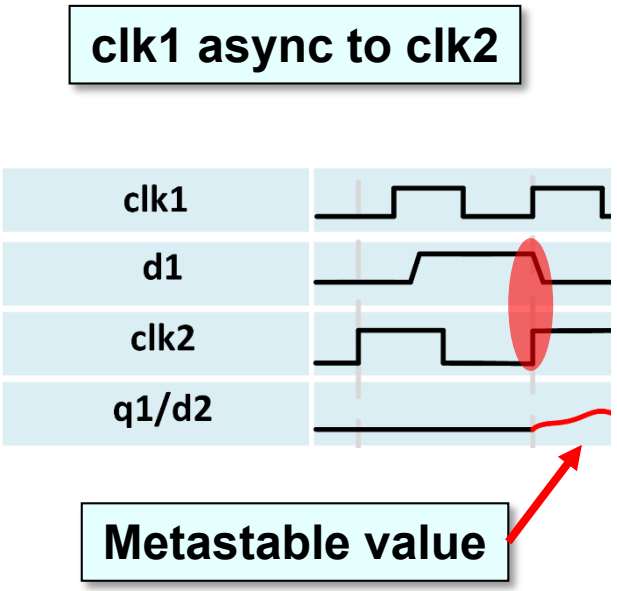
2017 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION UNITED STATES

Agenda

- Introduction to Reset Domain Crossings
- Examples of RDC Schemes
- Mitigating and Avoiding RDC Issues
- Setup and Running RDC Static Tools
- Real World Results

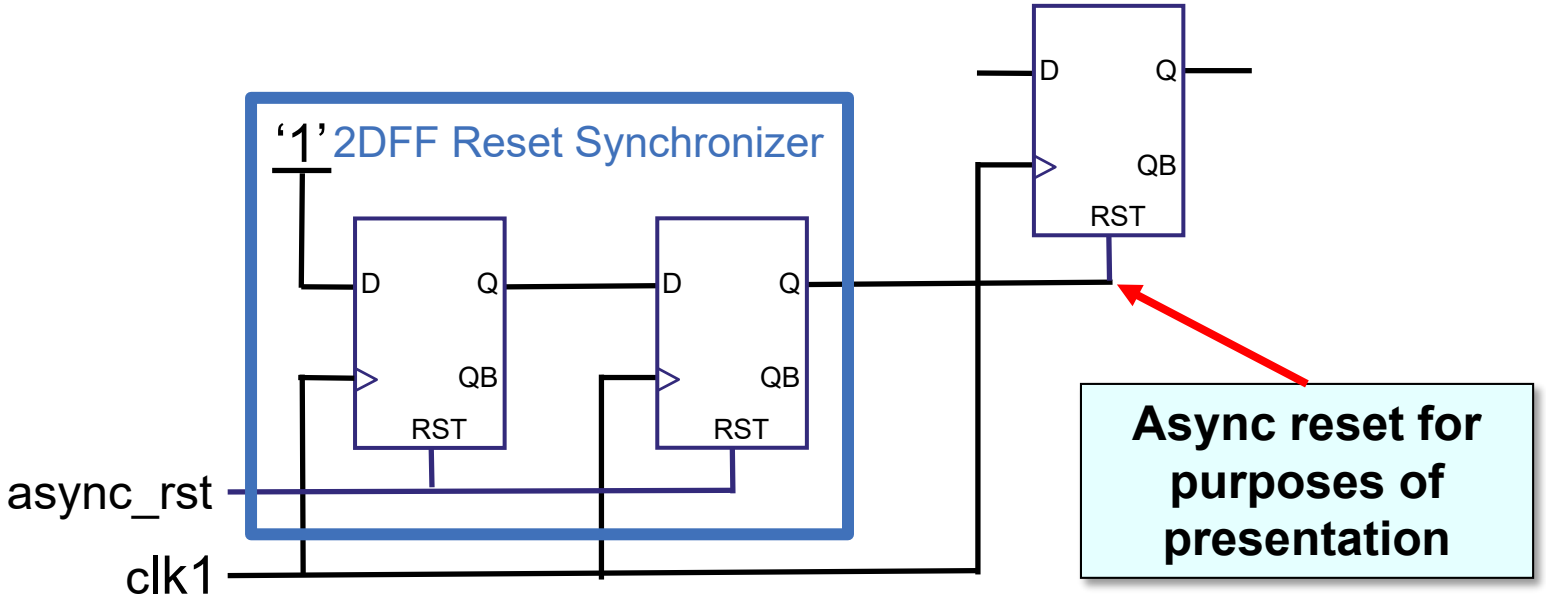
Clock Domain Crossings





- Metastability is a major concern specifically when FF2 has a fanout > 1.
- Metastable state may propagate along fanout and settle to a random (and functionally illegal) combination of “1”s and “0”s.

Asynchronous Resets

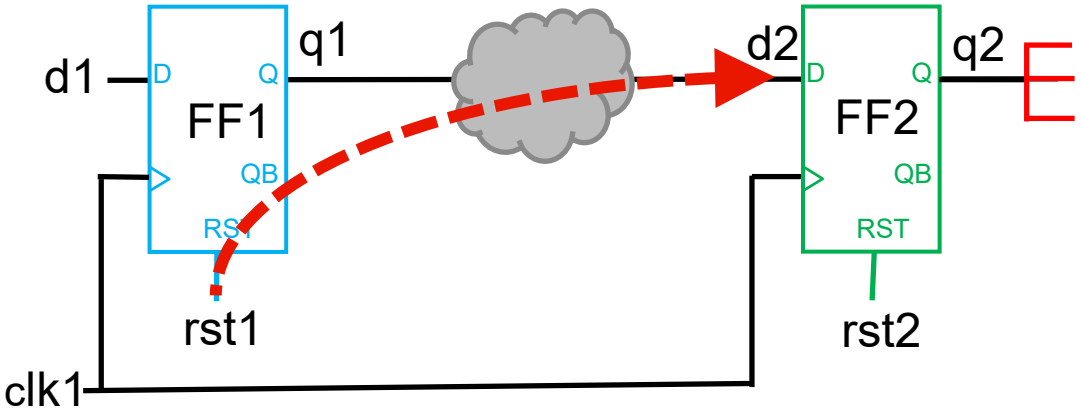


- Async resets are asserted asynchronously (timing, area, power considerations), but de-asserted **synchronously**!
- Failure to de-assert synchronously may induce metastability into the receive reset domain if de-assert close to clock edge.

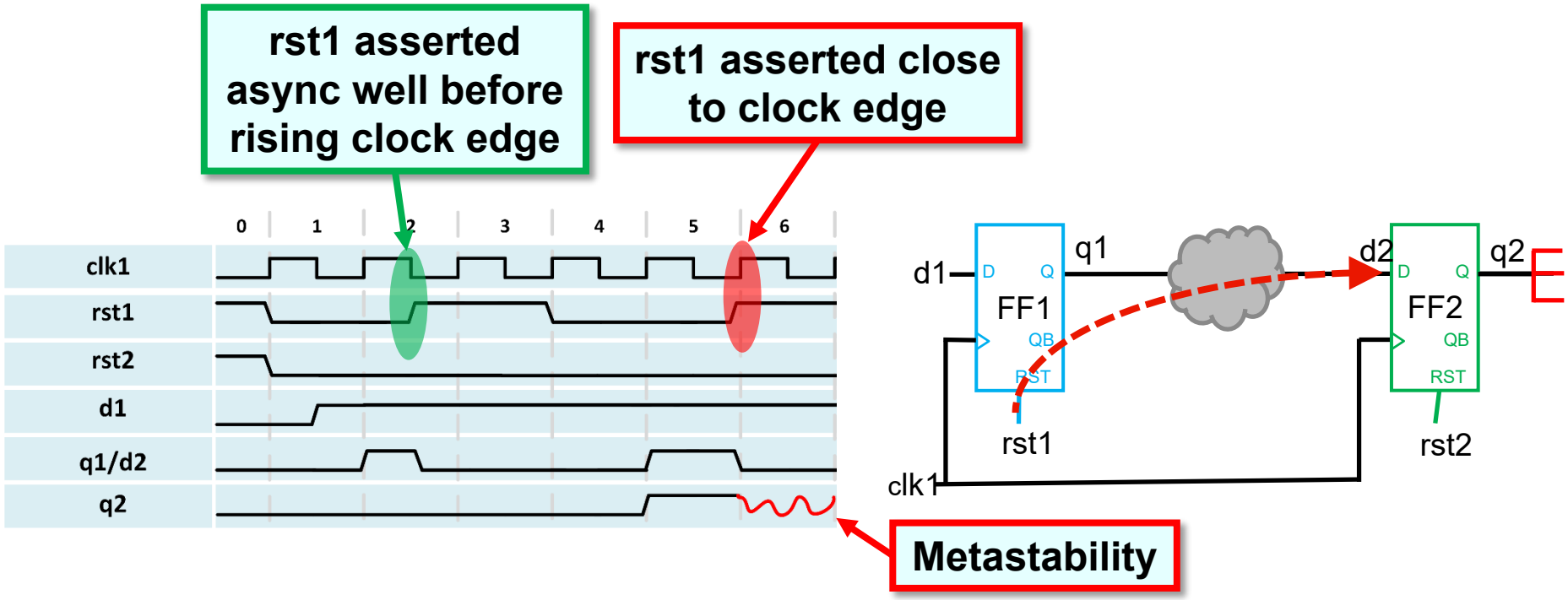
Reset Domain Crossings

- Formerly- chips used a small number of resets.
- Today- designs employ multiple resets (power domains, IPs, SW, FW, etc.)
- Similar to CDCs, design can be divided into reset domains.
- Assertion of rst1 causes async change of value on d2 regardless of clock edge, potentially violating setup & hold.

RDC Conditions
- clk1, clk2: sync or async.
- rst1 is async
- rst2 sync or async
- q1 **not** at reset value (‘1’).
- rst2 is de-asserted.



Reset Domain Crossings



- Timing path from rst1 to d2 not considered in static timing analysis.
- Front end design bug, not backend's responsibility.

Why Worry About RDC?

RDCs cause chip failures which require silicon respins!

- While CDC checks standard in the industry, little awareness of the critical importance of RDC issues.
- RDC issues incredibly difficult to catch, diagnose or reproduce in the silicon, may only be found by customer.
- Risk grows exponentially as design sizes, complexity and number of resets increases.
- Proper manual reviews of a design practically impossible.
- Simulation x-injection, formal assertions, etc. are neither scalable nor robust.

RDC Versus CDC

RDC similar to CDC.

- Async data change transmitted to receive domain.
- May violate of setup & hold time and cause metastability.
- We relate to each async reset as its own reset domain.
- Required to verify all crossings between reset domains.
- Past designs had 1 reset/clock domain, checks not needed.

RDC different than CDC.

- RDC issues can occur even in the same clock domain.
- Not flagged by standard CDC tools.
- FFs can be affected by multiple resets simultaneously.
- New categories of constraints and synchronizers to resolve.
- CDC checks are already standard.

2017 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION UNITED STATES

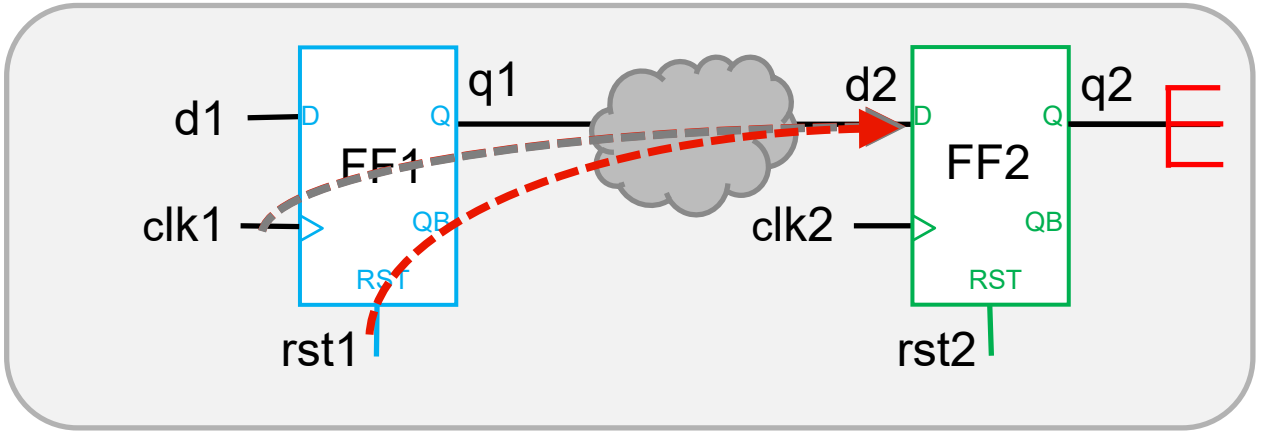
Agenda

- Introduction to Reset Domain Crossings
- Examples of RDC Schemes
- Mitigating and Avoiding RDC Issues
- Setup and Running RDC Static Tools
- Real World Results

Examples of RDC Schemes

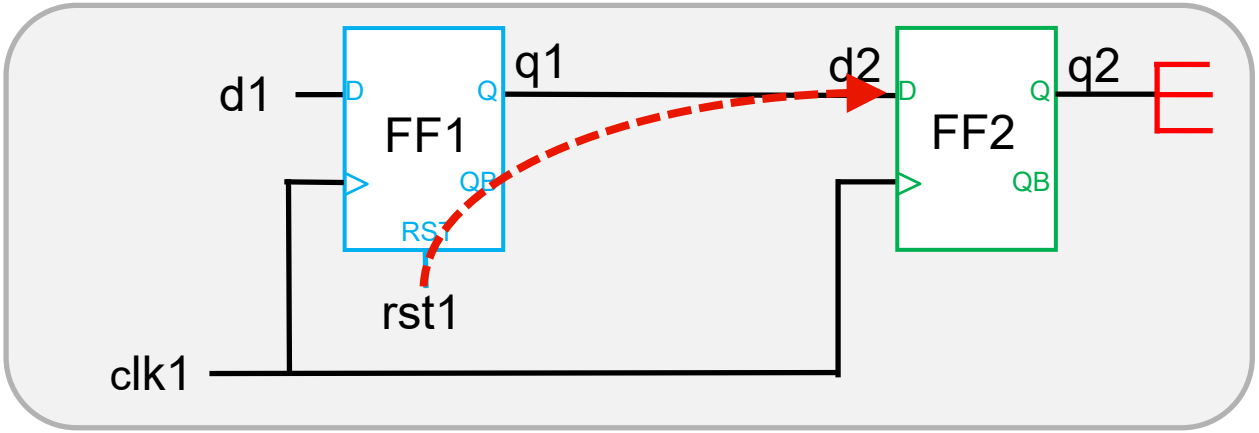
2 Clock Domains

Might have been waived as CDC issue!



No Receive Reset Domain

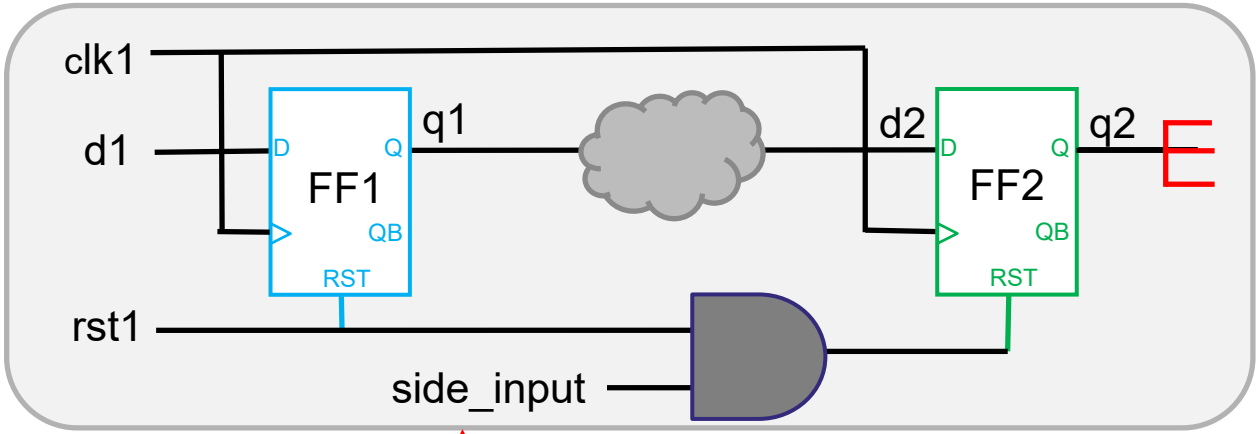
Path still exists from rst1 that violates setup & hold!



Examples of RDC Schemes

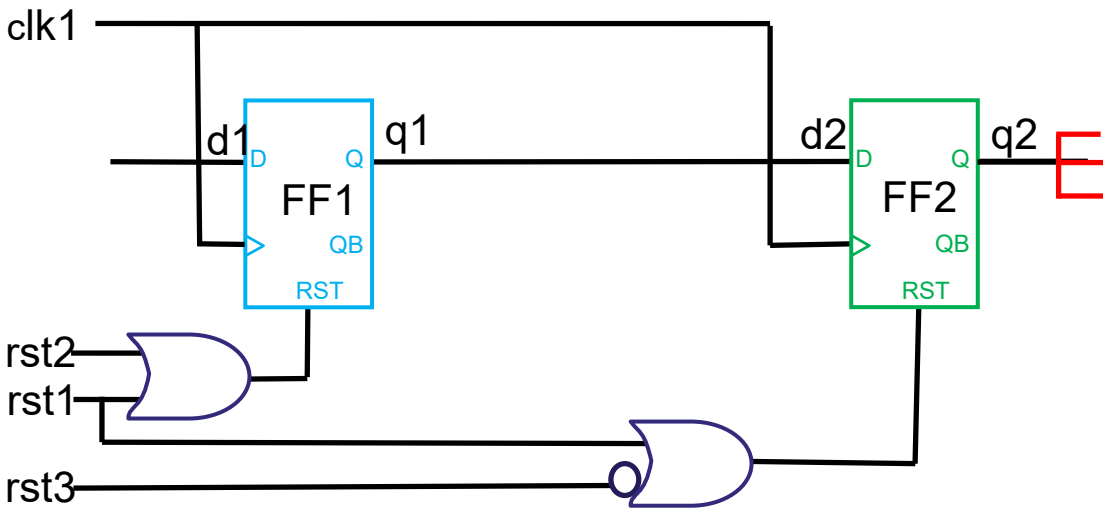
Side Input

RDC possible even in the same reset domain.



Signal may be asynchronous to clk1, many CDC tools will not catch this.

RDCs From Multiple Resets



FF reset when RST='1'

Source	Receive	RDC?
rst1	rst1	No
rst1	rst3	No
rst2 ('0'→'1')	rst1 ('0')	Yes
rst2 ('0'→'1')	rst3 ('1')	Yes

- Unlike CDCs, FF's can be affected simultaneously by multiple dependent resets/scenarios.
- Reset domain definitions not always clear cut or well defined.
- 100% coverage of such complex schemes cannot be met by simulation or formal approaches; static has advantage.

2017 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION UNITED STATES

Agenda

- Introduction to Reset Domain Crossings
- Examples of RDC Schemes
- Mitigating and Avoiding RDC Issues
- Setup and Running RDC Static Tools
- Real World Results

Avoiding & Mitigating RDC

Strategies in order of preference.

Avoid

1. Don’t use async resets (generally not realistic ☺).
2. Architect design to align reset domains with functional logic to prevent/limit communication between reset domains.

Mitigate

3. Define resets in the same domain.
4. Define reset assertion order.
5. Add synchronizers to RDC paths.
6. Implement qualifier schemes.
7. Waivers.

3. Same Reset Domain

3 reset relationships can be defined as same “reset domain”:

1. 2 or more resets always asserted together, i.e. from the top, global reset, etc. (share polarity, reset value, a/sync, etc.).
2. Assertion of rst1 will always trigger **immediate** assertion rst2
   Assertion of rst2 will always trigger **immediate** assertion rst1.
3. Assertion of rst1 will always trigger **delayed*** assertion rst2
   Assertion of rst2 will always trigger **delayed*** assertion rst1.

***Assuming delay is minimal enough. Designer must determine that the interim values in receive reset domain are functionally irrelevant.**

4. Reset Order

- Often resets exhibit an assertion relationship or dependency.
- For example: shallow global resets such as power_on_good may always trigger deeper, local resets, but not the opposite.
- This “reset order” can be captured for the tool as constraints:
  - Assuming assertion of rst1 will always trigger immediate or delayed assertion rst2 (but not vice versa).
  - Tool will not report RDC violations for direction rst1→ rst2, but will report rst2 → rst1.

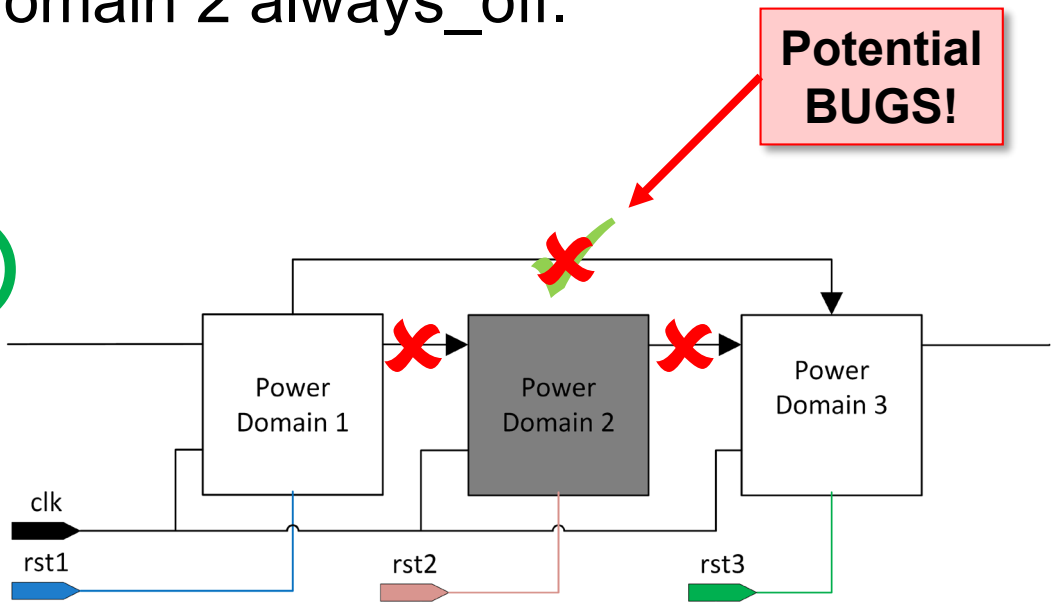
```
#Constraints
Reset_order rst1 to rst2
Reset_order rst2 to rst3
#Tool will conclude itself
Reset_order rst1 to rst3
```

Potentially usefully, but can be dangerous

4. Reset Filter

- Similar to reset order, can filter out RDC violations from one reset domain to another but without tool making assumptions.
- Example: assume power domain 2 always_off.

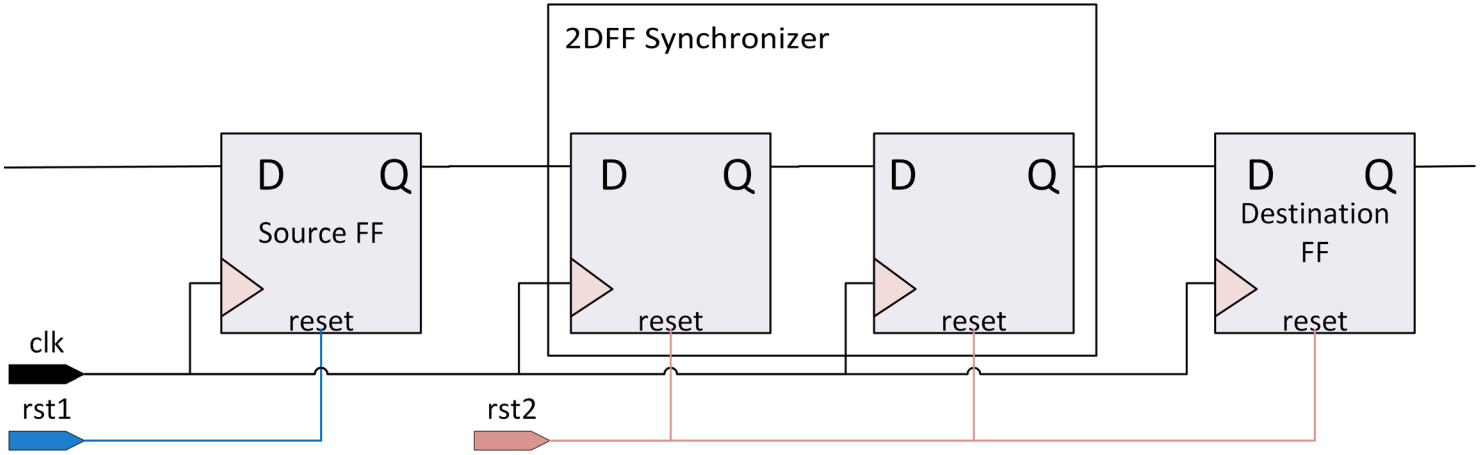
```
#Constraints
Reset_filter rst1 to rst2
Reset_filter rst2 to rst3
```




```
#Constraints
Reset_order rst1 to rst2
Reset_order rst2 to rst3
#Tool will conclude itself
Reset_order rst1 to rst3
```

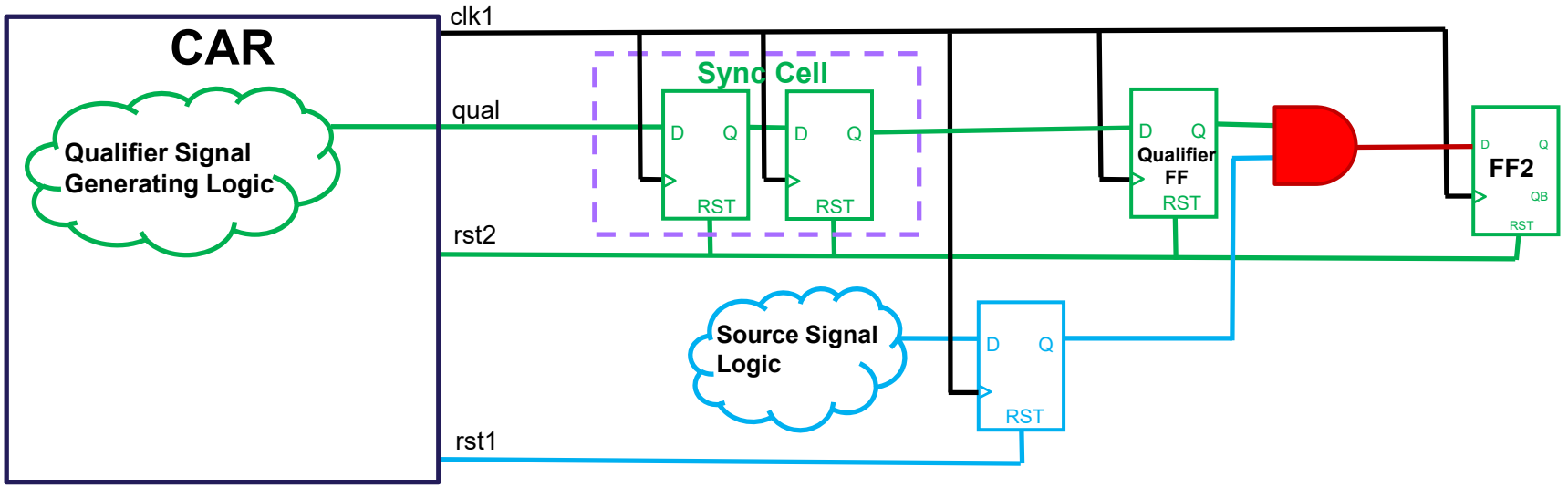
5. Synchronizers

- Just like CDCs, RDCs cause silicon bugs through asynchronous data changes that violate setup & hold.
- Standard CDC sync schemes ensure data values stabilize before propagating to the receive domain's functional logic.
- Tools detect synchronized RDC paths, won't report violations.



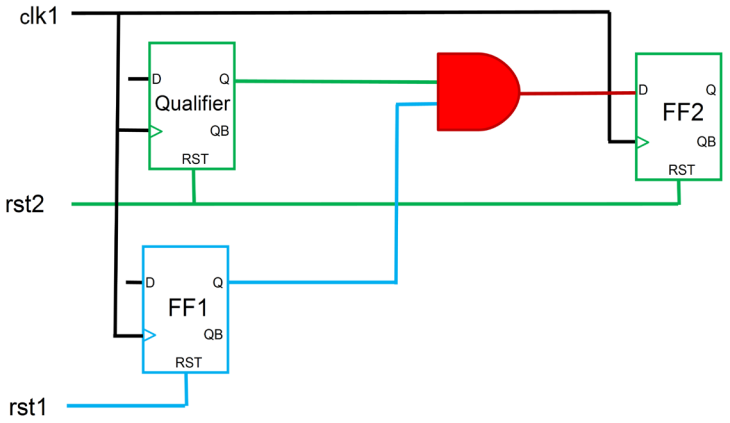
6. Qualifiers

- "Pre-indication" signals indicating an imminent reset assertion can be created or may already exist in the design.
- Used to "qualify" or gate RDC paths between 2 reset domains.
- Define qualifier signal through constraints, tool won't report affected RDCs.

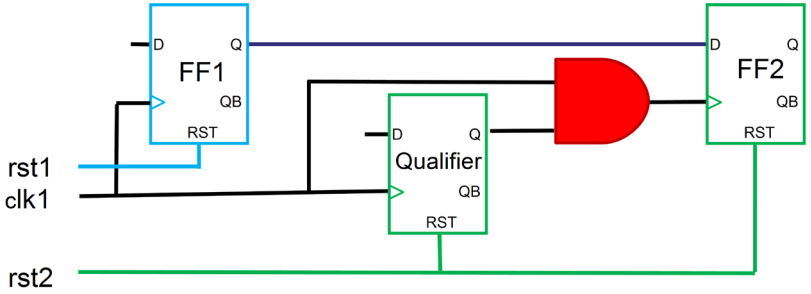


6. 3 Qualifiers Schemes

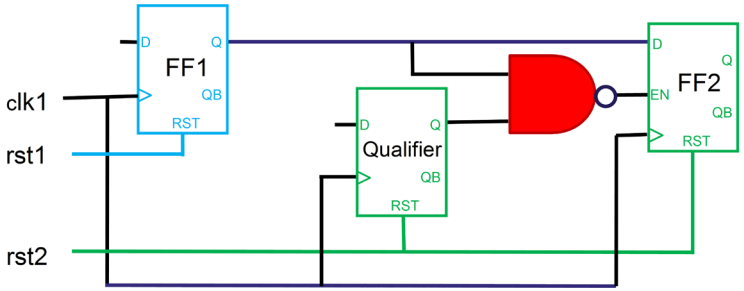
Data Qualifier



Clock Qualifier



Enable Qualifier



Assuming qualifier active low.

- Qualifier must arrive before reset assertion.
- Qualifier must be sampled in reset & clock receive domain.
- Separate AND gate must be implemented for every RDC.
- Possible to use OR gate qualifier with active high.

Caution! Tools accept qualifier definitions “as-is”!

2017 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION UNITED STATES

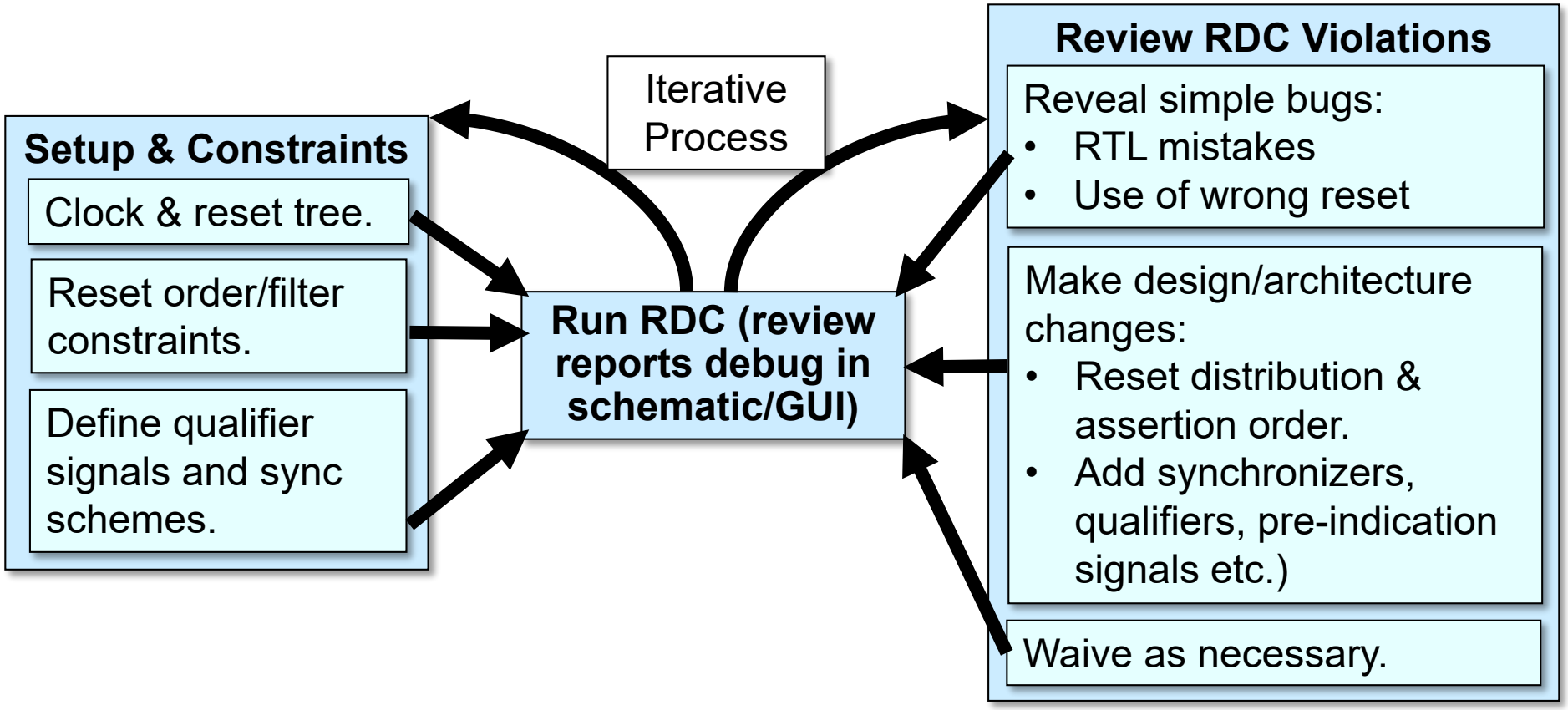
Agenda

- Introduction to Reset Domain Crossings
- Examples of RDC Schemes
- Mitigating and Avoiding RDC Issues
- Setup and Running RDC Static Tools
- Real World Results

Setup & Running RDC Static Tools

- Industry standard CDC/LINT tools can now flag RDC issues.
- Capable of running as a static check, flat on large designs.
- Most effective full chip, as most RDCs between blocks.
- 2 methods for adding RDC to flow:
  1. If already using platform CDC/LINT tool that supports RDC, just add reset definitions and relevant RDC rules.
  2. Add new RDC tool in parallel to flow (minimal setup).
     - Filelist, clock/reset domain info, & functional mode constraints can be quickly adapted from pre-existing CDC/LINT tool flow.
     - May need to slightly massage clock/reset tree naming.
     - Shouldn't be necessary to convert/translate CDC waivers, stable signal declarations, etc.

Tool Flow


Iterative Process
Setup & Constraints
Clock & reset tree.
Reset order/filter constraints.
Define qualifier signals and sync schemes.
Run RDC (review reports debug in schematic/GUI)
Review RDC Violations
Reveal simple bugs:
• RTL mistakes
• Use of wrong reset
Make design/architecture changes:
• Reset distribution & assertion order.
• Add synchronizers, qualifiers, pre-indication signals etc.)
Waive as necessary.

2017 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION UNITED STATES

Agenda

- Introduction to Reset Domain Crossings
- Examples of RDC Schemes
- Mitigating and Avoiding RDC Issues
- Setup and Running RDC Static Tools
- Real World Results

Real World Results

- Several RDC bugs found in silicon triggered search for better solution as manual methods (i.e. GREP) were not effective.
- Our internal POR static CDC/LINT tools don't support RDC so we added a 2nd tool in parallel.
- First RDC static tool trial in mid 2014 on large network chip close to tape out & mature (clean LINT/CDC/simulation/etc.).
  - ~2.5 Million FFs, ~52 Million NAND Gates.
  - 13 Clock Domains
  - 8 Reset Domains
- Found several serious RDC bugs that required RTL/design fixes and designers fixed many issues "on the fly".

Real World Results

- Users found the RDC flow robust, accurate, friendly to debug, & low noise when employing the strategies outlined above.
- Expect initially a large number of **real** RDC violations.
- During our initial RDC trial a smaller parallel project declined to adopt RDC flow due to time constraints, and an easily detectable RDC bug was found in their silicon.
- RDC checks have now become POR for our division.
- Since the trial, RDC tools continue to improve with many enhancements such as new qualifier constraints and formal capabilities.
- RDC checks continue to gain recognition towards becoming a standard part of verification flow in Intel and the industry.

2017 DESIGN AND VERIFICATION™ DVCON CONFERENCE AND EXHIBITION UNITED STATES

Questions

Backup: Waivers

- Similar to standard static CDC and LINT flows, can waive.
- Last resort after other methods failed and proper review.
- Certain RDC tools provide dynamic TCL capabilities to generate waivers based on the fan-in or fan-out of the RDC.