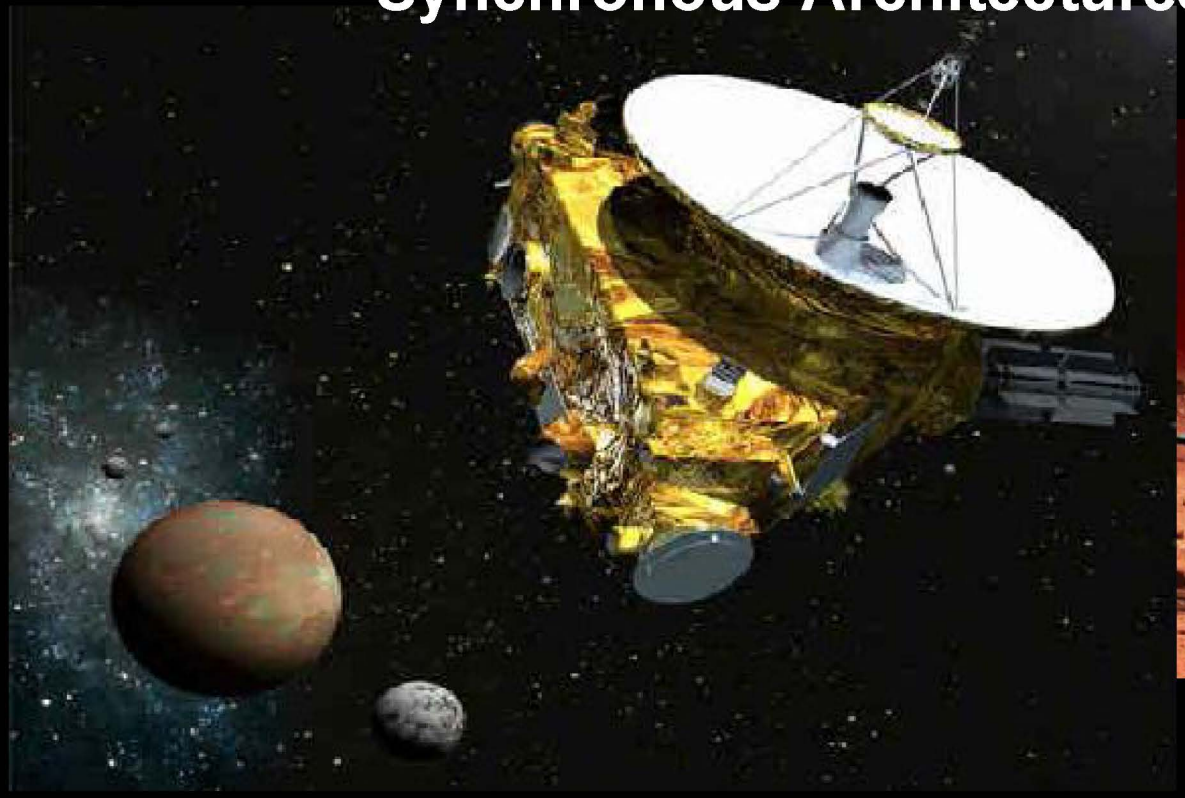
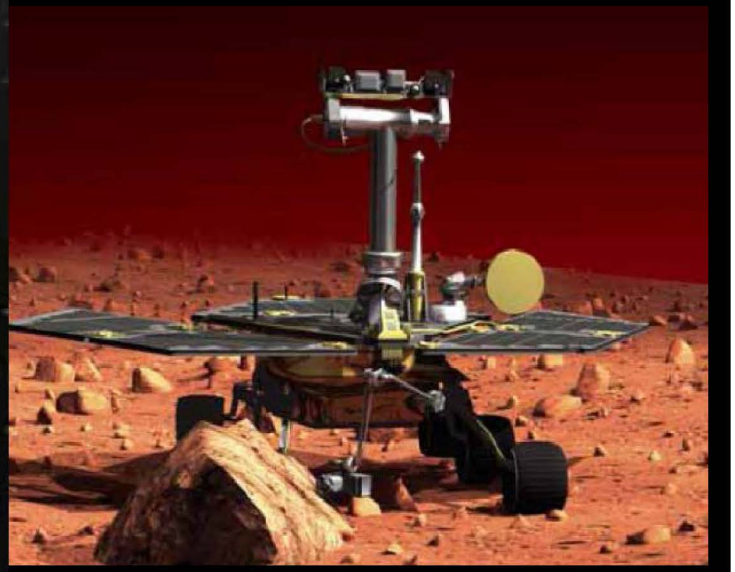
The Effects of Race Conditions when Implementing Single-Source Redundant Clock Trees in Triple Modular Redundant Synchronous Architectures





To be presented by Melanie D. Berg at Radiation Effects on Components and Systems (RADECS) Conference, Bremen, Germany, September 19-23, 2016

Acronyms

- Clock cycle time (T_{clk})
- Combinatorial logic (CL)
- Data-path hold time requirement (T_{HOLD})
- Design under analysis (DUA)
- Delay of combinational logic delay (T_{comb})
- Delay of data output of DFF ($T_{clk \rightarrow q}$)
- Device under test (DUT)
- DFF setup time (T_{setup}).
- DFF hold time ($T_{DataStable}$)
- Distributed triple modular redundancy (DTMR)
- Edge-triggered flip-flops (DFFs)
- Field programmable gate array (FPGA)
- Global triple modular redundancy (GTMR)
- Hardware description language (HDL)
- Input – output (I/O)
- Linear energy transfer (LET)
- Mean time to failure (MTTF)
- Mitigation window (MW)
- Multiple bit upset (MBU)
- Radiation Effects and Analysis Group (REAG)
- Single Error Correct Double Error Detect Single event functional interrupt (SEFI)
- Single event effects (SEEs)
- Single event transient (SET)
- Single event upset (SEU)
- Single event upset cross-section (σ_{SEU})
- Static random access memory (SRAM)
- Static timing analysis (STA)
- Triple modular redundancy (TMR)

Problem Statement

- Triple modular redundancy (TMR) can be implemented in a variety of topologies.
- This presentation focuses on the trade-offs between implementing TMR with:
  - Multiple clock domains (Three clocks… one per TMR domain):
    i.e., global TMR (GTMR) and
  - A single clock shared across the three TMR domains:
    i.e., distributed TMR (DTMR).
- For many organizations, GTMR is the mitigation strategy of choice because of its redundant clock topology.
- However, as FPGA devices and designs become larger and more complex, clock-skew between separate domains is increasing and becoming impossible to control.
- Unfortunately, mismanaged clock-skew can cause timing violations or circuit race conditions in synchronous designs.

Race conditions from clock-skew weaken mitigation and can cause system malfunction!

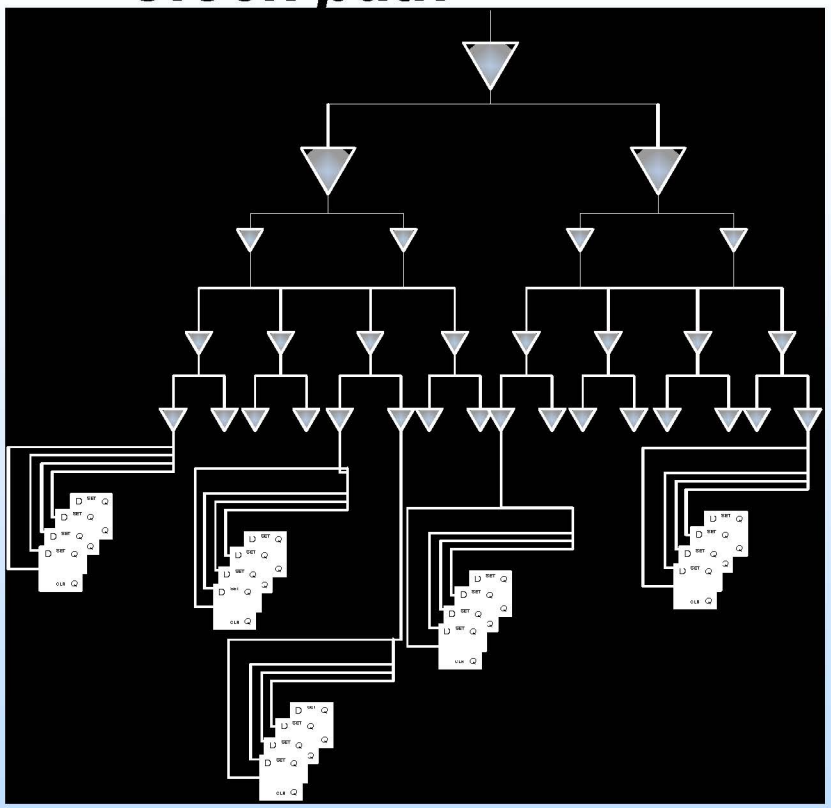
Abstract

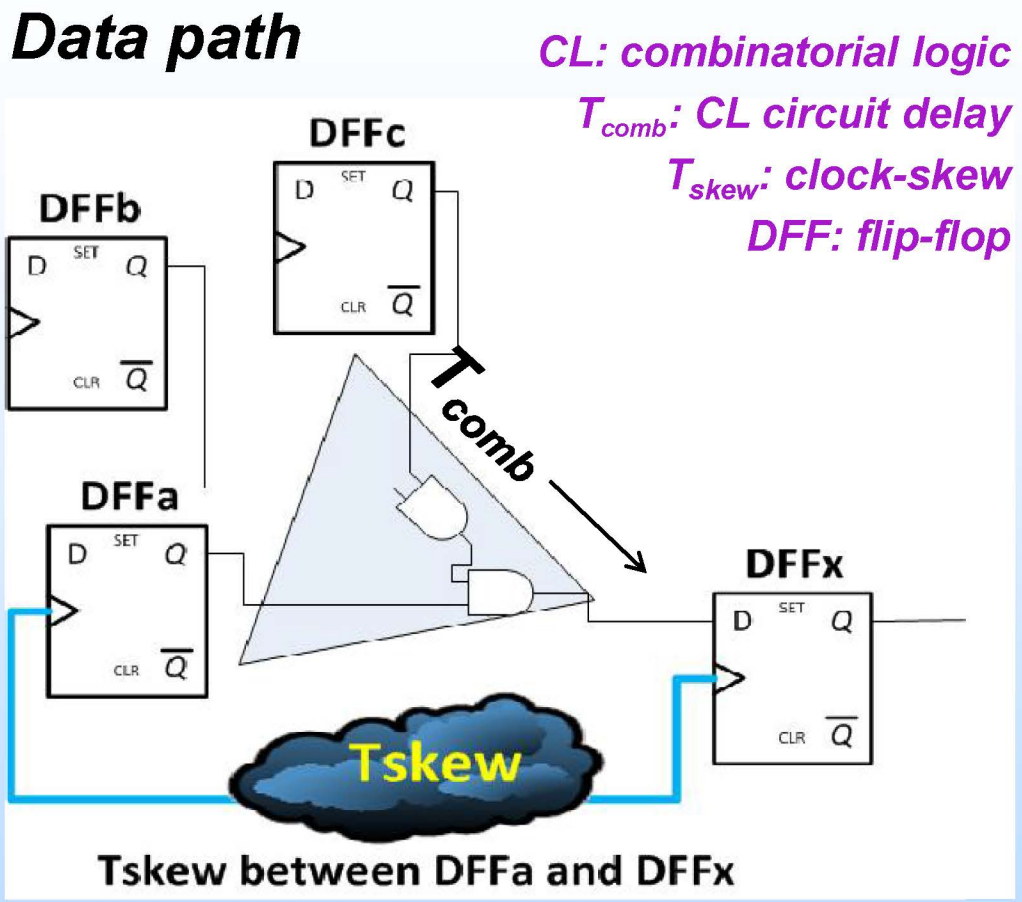
We present the challenges that arise when using redundant clock domains due to their clock-skew.

Heavy-ion radiation data show that a singular clock domain (DTMR) provides an improved TMR methodology for SRAM-based FPGAs over redundant clocks.

Clock-skew

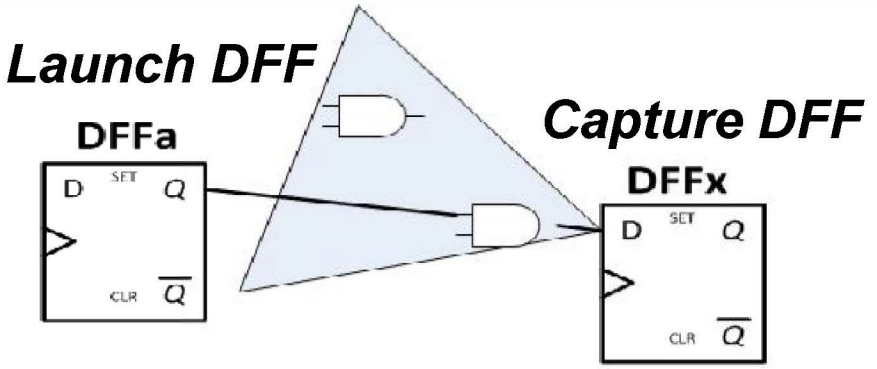
Clock-skew within One Clock Domain





The difference in time for a clock edge's arrival at one DFF with respect to its arrival at another DFF is defined as clock-skew (T_{skew}).

Synchronous Data Capture



DFF: flip flop

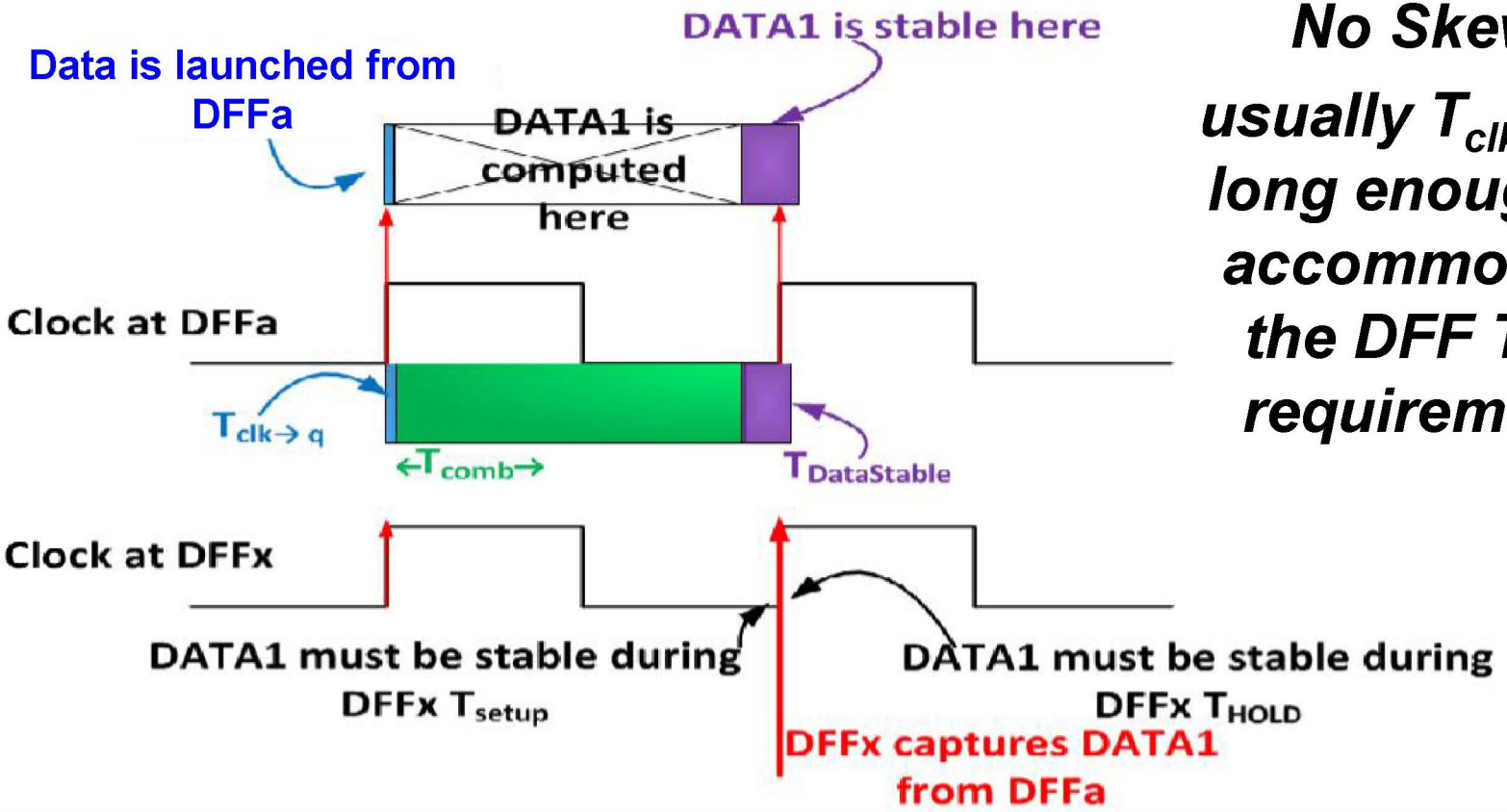
T_{comb}: combinational logic delay.

$T_{clk \rightarrow q}$: delay of data output from DFF.

$T_{DataStable}$: Data-path hold time requirement.

T_{setup}: DFF setup time.

T_{HOLD}: DFF hold time



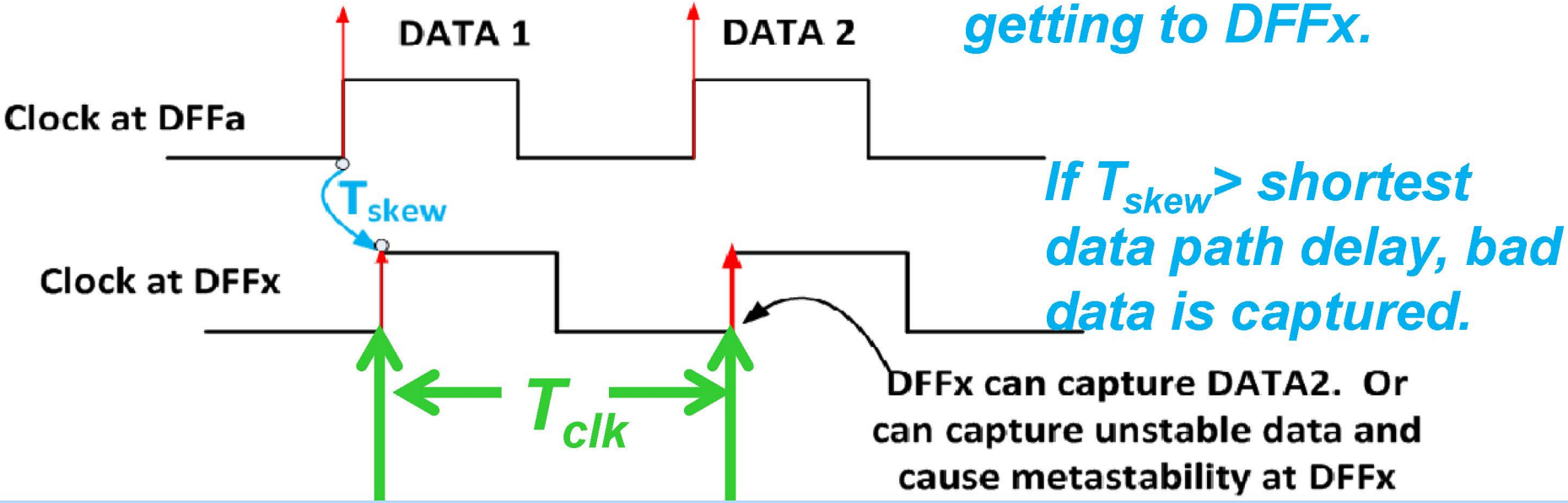
No Skew:

usually $T_{clk \rightarrow q}$ is long enough to accommodate the DFF T_{hold} requirement.

Positive Skew and Data Capture

- Large T_{skew}: DFF_x will capture the wrong data (cycle ahead).
- Small T_{skew}: DFF_x capture can be in the DFF T_{hold} window…data is unstable (metastability).
- Changing the clock cycle time (T_{clk}) will not fix T_{skew}.
- Longer data path delays that make incoming data stable at the capture DFF helps to accommodate skew.

Not shown: Data 1 and DATA 2 will be delayed getting to DFFx.

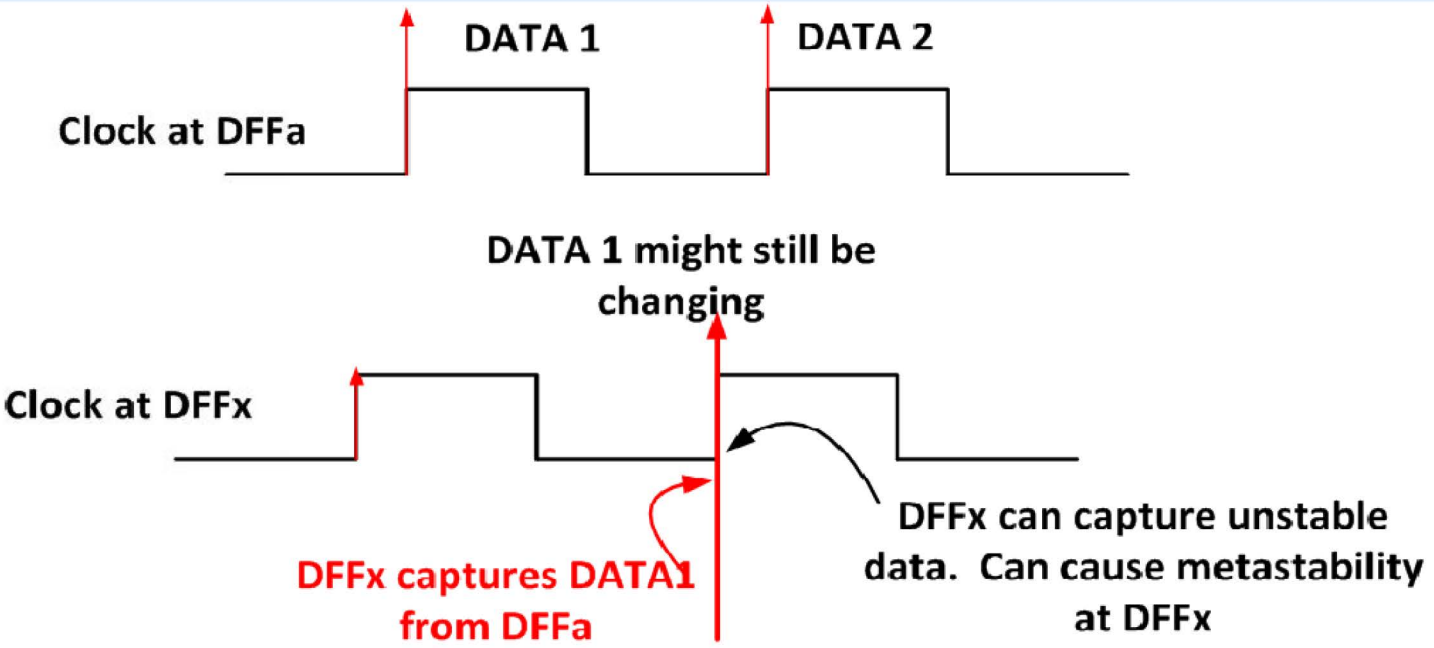


If T_{skew} > shortest data path delay, bad data is captured.

Negative Skew and Data Capture

In a system with negative skew, there is the possibility that data can be captured during it's computation time.

- **T_{setup} is violated.**
- **This can cause metastability.**
- **Data is invalid.**

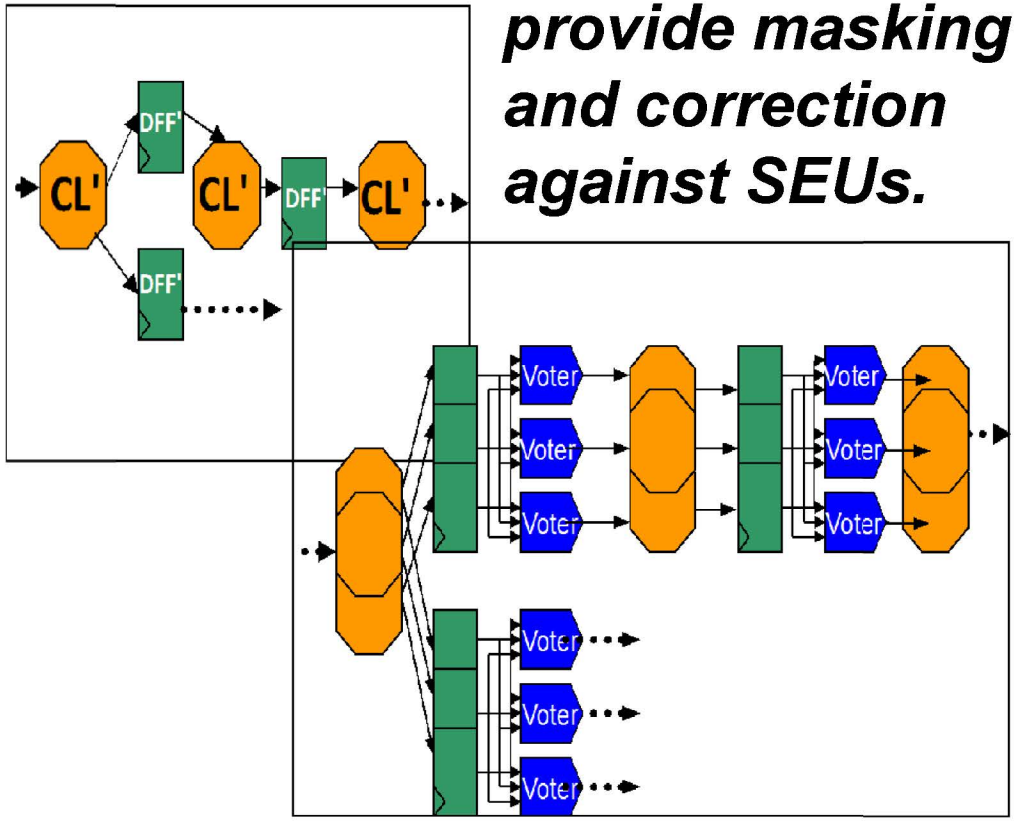


Triple Modular Redundancy (TMR)

Protection against single event upsets (SEUs)

DTMR and GTMR Topologies

- With DTMR and GTMR all circuits are triplicated; creating three TMR domains.
- Voters are placed after the internal flip-flops (DFFs).
- DTMR: only one clock per TMR domain.
- GTMR: Three separate clocks per TMR domain.



Internal Voters provide masking and correction against SEUs.

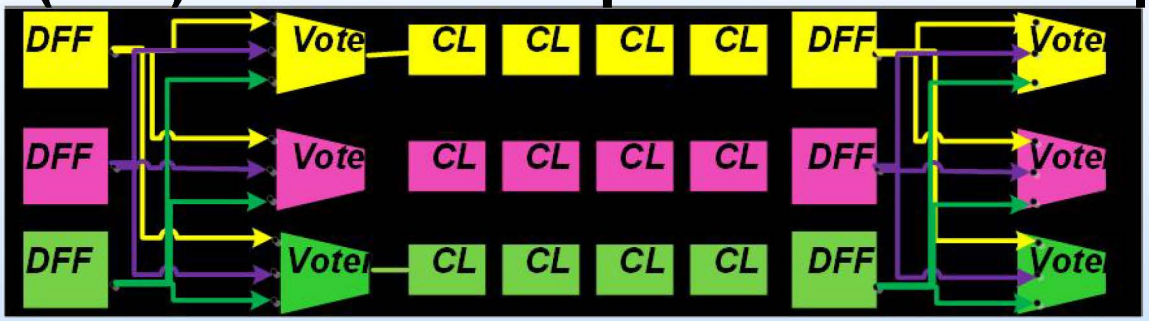
GTMR violates synchronous design protocol because of sharing data across clock domains without synchronization.

TMR Mitigation Window Definition

DFF to DFF data-path

DTMR and GTMR conversion of DFF to DFF data-path.

Mitigation Window (MW) is DFF-voter pair to DFF-voter pair.



In the absence of SEUs:

With GTMR, there is a possibility of having broken MWs because of T_{skew}. There are no broken MWs with DTMR.

With the occurrence of SEUs:

The broken GTMR MWs have weakened mitigation (masking and correction cannot be guaranteed).

Challenges of GTMR System Implementation

System Implementation: Sources of Clock-skew

- **Board Level:**
  - One board clock source (oscillator): routes from board clock source must be the same length to FPGA clock inputs.
  - Three board clock sources: Don't!
- **Internal to the FPGA:**
  - Clock pin to clock tree routing differences,
  - Skew within a single clock tree, and
  - GTMR has additional skew from use of different clock trees.

GTMR Clock-skew Management

- Board level and routing clock-skew can be managed.
- However, clock-skew within a single tree and between different trees is based on the manufacturer's product and can be difficult or impossible to control.
- Although a concern, clock-skew was less of a problem in smaller Xilinx FPGA devices (e.g., Virtex 5 or smaller).
- Clock-skew is more of a challenge in the larger family of Xilinx devices (e.g., Xilinx-7 series and above).
  - More skew within one clock tree (especially as distance between DFFs increase).
  - More skew between separate clock trees.
  - Faster routes and combinatorial logic… data changes quicker. $T_{skew} > T_{comb}$… *Race condition*

Detection of T_{skew} with GTMR

- GTMR T_{skew} is difficult to detect due to the following:
  - Many static timing analysis (STA) tools do not accurately report hold time violations across clock domains.
    - Hence the user might not understand the full extent of T_{skew}.
  - T_{skew} is temperature and voltage dependent; and will vary.
    - Hence, a design can work during ground testing yet have failures during operation in its target environment.
  - In the presence of clock-skew, usually two out of three of the domains are in sync.
    - Hence the design will appear to operate normally.
- Due to state space explosion, fault injection and simulation will not provide sufficient coverage.

T_{skew} System Effects

- Significantly large T_{skew}: can cause one domain to always be out of sync with the other two domains. Easiest skew to manage and detect.
- Marginal T_{skew}: can cause metastable circuits.
- Variable T_{skew}: can cause pockets of T_{skew} such that some portions of the circuit contain:
  - Positive T_{skew},
  - Negative T_{skew}, and
  - Marginal T_{skew}.
- With multiple clock domains, when overall T_{skew} decreases, (e.g., via board level and routing management) pockets of variable T_{skew} start to exist.
- This is more prominent in large FPGA devices such as the Xilinx 7-series.

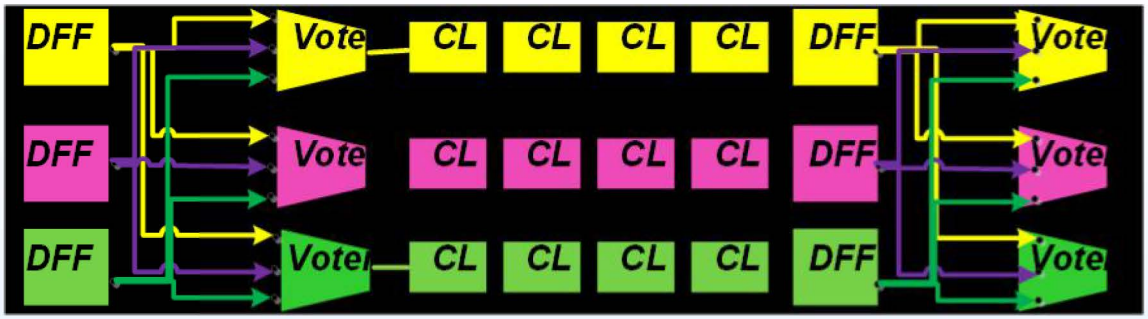
Accelerated Heavy Ion Testing

Accelerated Radiation Testing

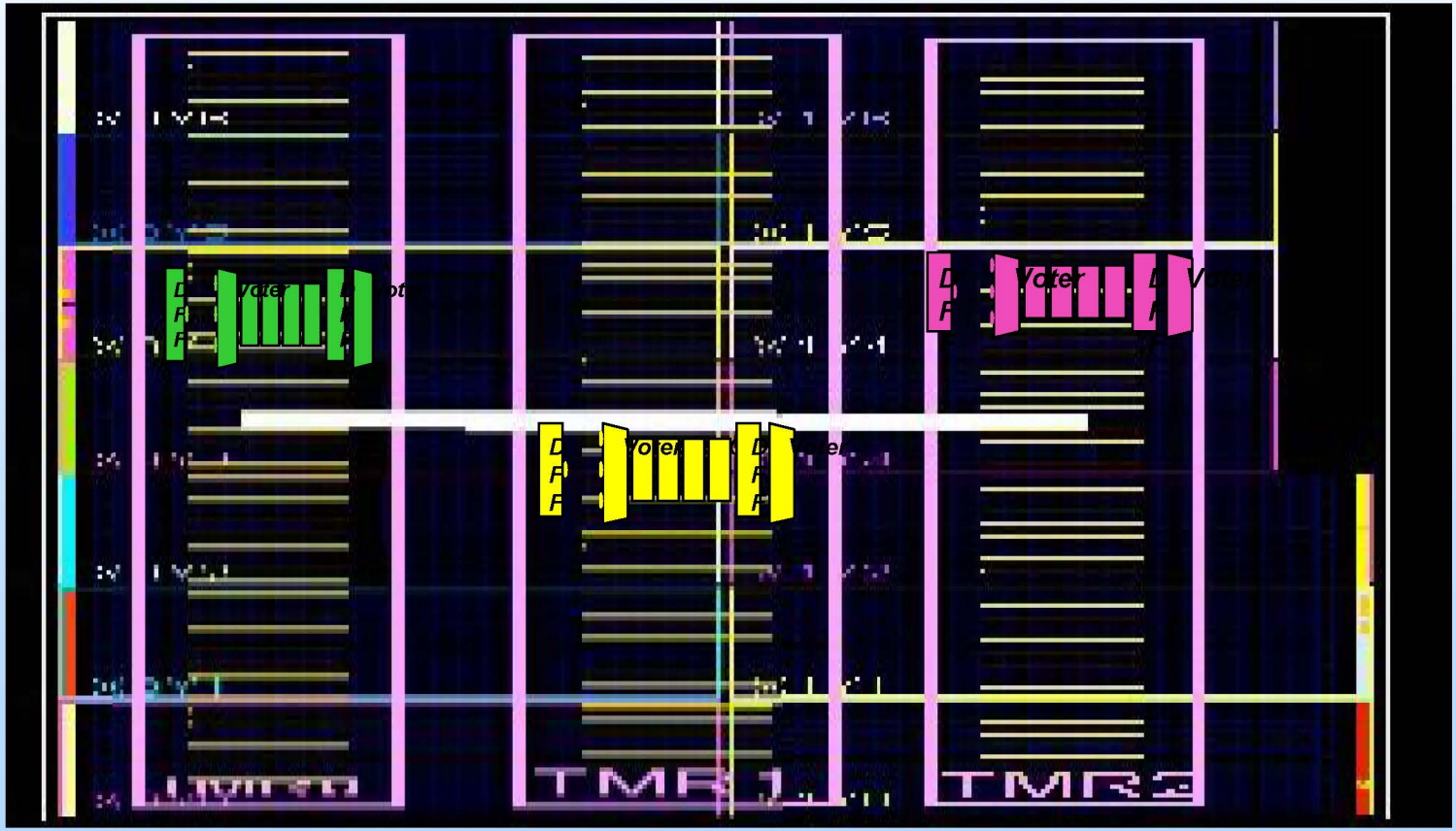
- Device under test (DUT): Xilinx Kintex-7 FPGA (XC7K325T).
- The base design (DUA) was the counter-array created by NASA Electronics Parts and Packaging (NEPP) Program.
- There were three versions of the counter-array DUA; based on the inserted TMR scheme:
  - No TMR,
  - GTMR, and
  - DTMR.
- The TMR DUAs were physically partitioned across TMR domains in order to reduce shared resources (single points of failure).

SRAM Based FPGAs… two types of SEUs of major concern: Configuration bit SEUs and Global routes.

TMR Mitigation Window and Partitioning



SEUs that occur in one TMR domain within a MW are expected to be mitigated.

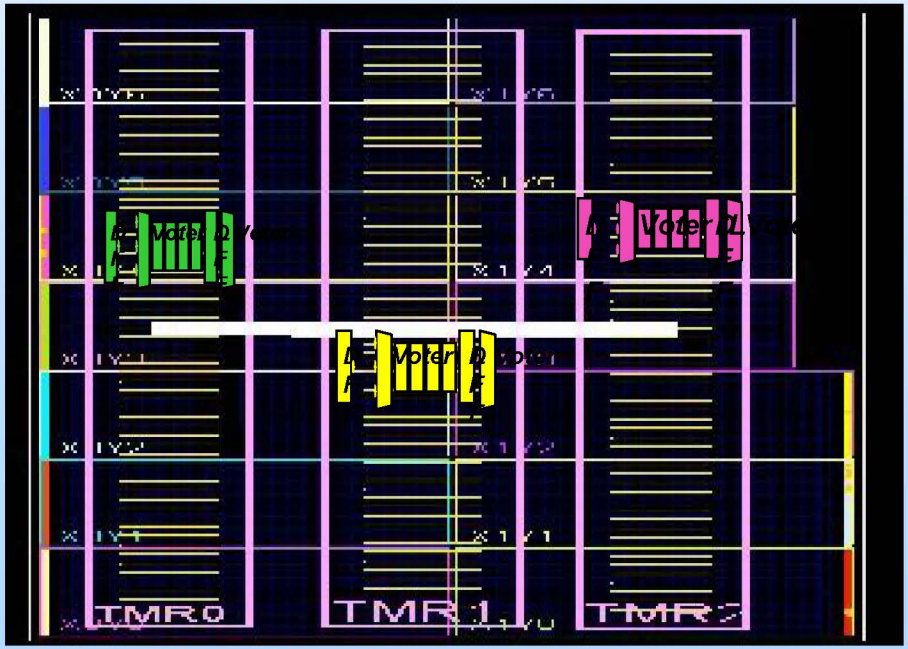


Analyzing LET Values less than 2.0MeVcm^2/mg and Configuration SEUs

- SEUs in this LET range generally occur in configuration bits.
- However, there is a very small number of configuration bit SEUs.
- A configuration bit SEU is expected to be mitigated if the MW is not broken and is partitioned correctly.

DTMR:

- MWs can mitigate configuration bit SEUs.



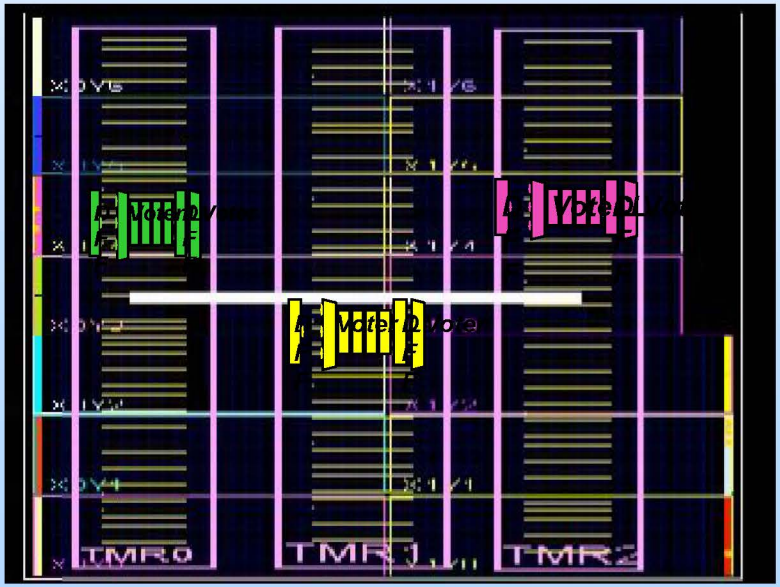
GTMR:

- If a portion of MWs are broken due to skew, with low LETs, there is a low probability of reaching a broken MW. Hence a low probability of causing system failure.
- However, if all MWs are broken due to skew, most configuration bit SEUs (those that control used design structures), will cause system failure.

Analyzing LET Values greater than 2.0MeVcm^2/mg and Configuration SEUs

GTMR:

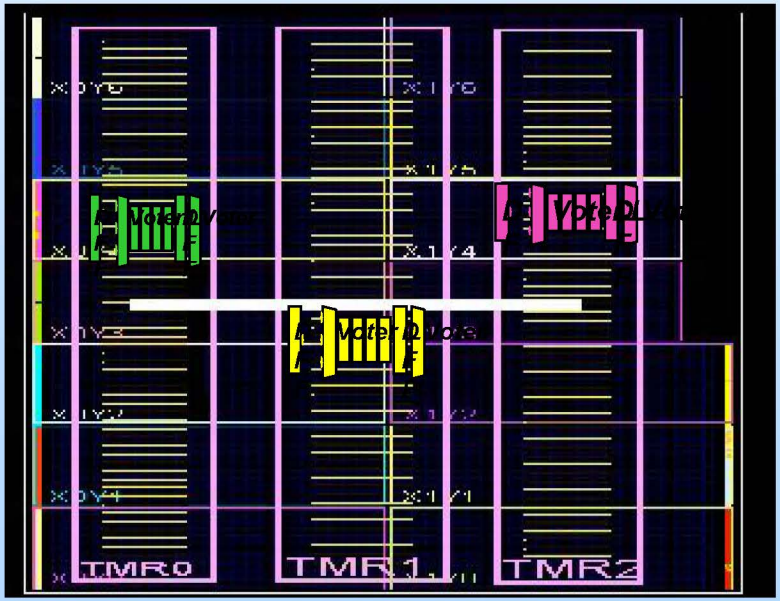
- As LET increases, there is an increase in configuration bit SEUs:
  - Can cause malfunction, if the configuration bit controls a broken MW.
  - As the number of configuration bit SEUs increases, the probability of reaching a broken MW also increases.

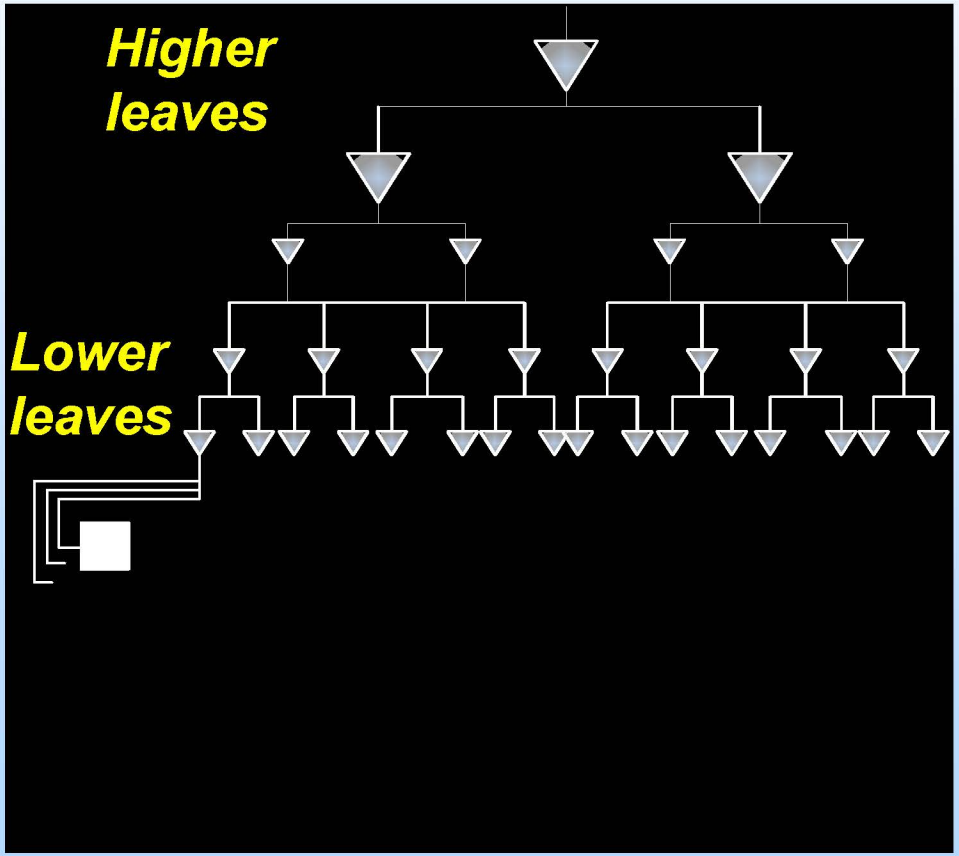


Because of partitioning, there is a low probability of having shared resources across TMR domains.

Analyzing Clock SETs

- It is rare for clock SETs to occur at lower range LETs.
- **DTMR:** Lower leaf clock SETs will only affect a small number of DFFs… locally placed connections. Expected to be mitigated.
- **DTMR:** Higher leaf clock SETs can affect a large number of DFFs. Hence can cross TMR domains and break MWs.
- **GTMR:** Clock SETs can affect multiple MWs. Hence there is a higher probability of reaching a broken MW.



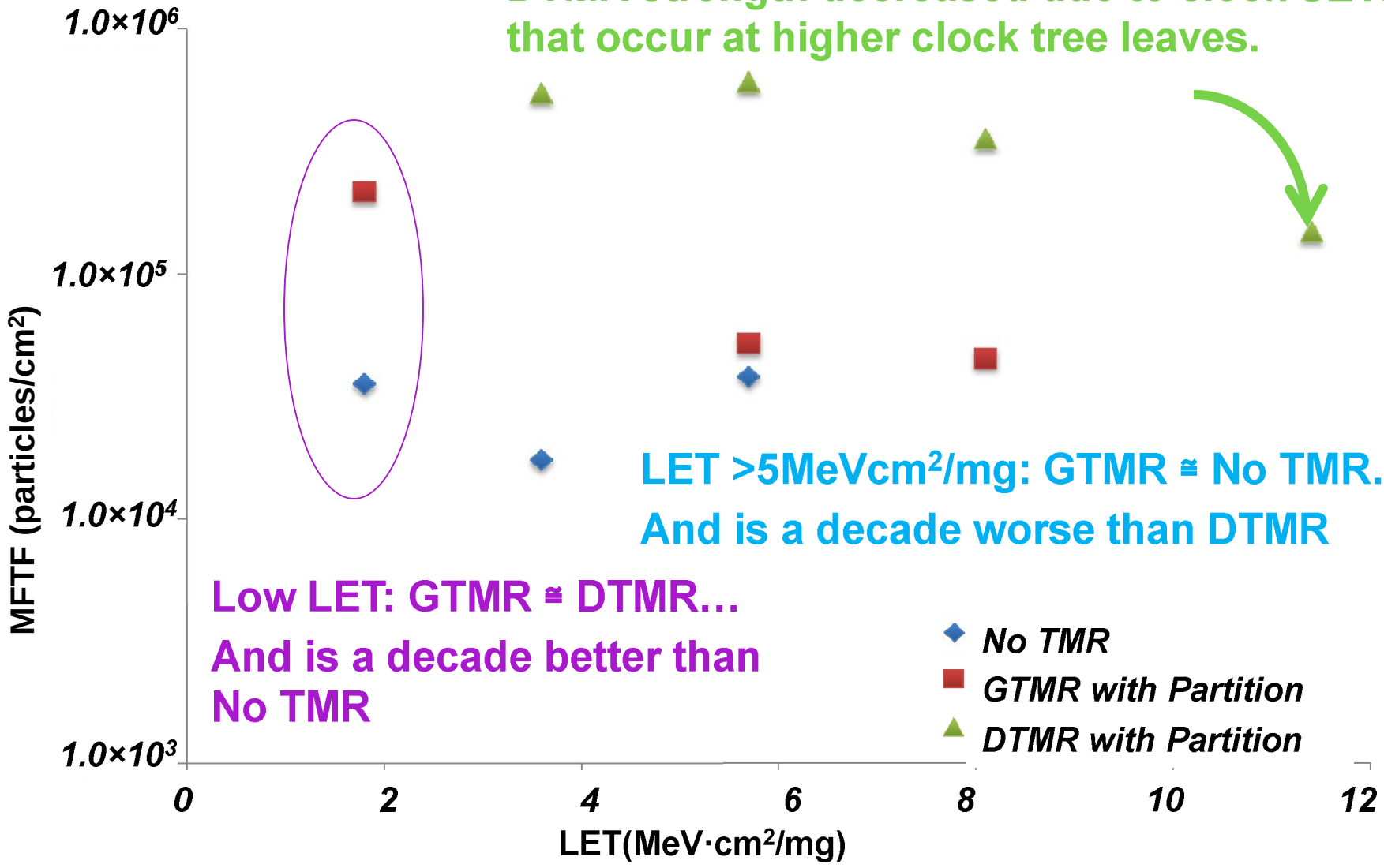


Heavy-Ion Results

LET: linear energy transfer

MFTF: Mean failure to fluence

DTMR strength decreased due to clock SETs that occur at higher clock tree leaves.



LET >5MeVcm^2/mg: GTMR ≅ No TMR…

And is a decade worse than DTMR

Low LET: GTMR ≅ DTMR…

And is a decade better than No TMR

GTMR Can Contain Pockets of Clock-skew

- If results were due to large T_{skew}, one GTMR clock domain would always be out of sync; and GTMR would always have results similar to No TMR.
- Because GTMR SEU data are near DTMR at low LET and approach No TMR as LET increases, suggests that the failures are mostly due to scattered pockets of clock-skew.
- Remember …MBUs are not considered the differentiating factor between DTMR and GTMR because both systems are partitioned in the same manner.

Conclusion

- Theoretically, GTMR should be the strongest TMR mitigation scheme.
- For this reason, it has been suggested as the TMR strategy of choice for SRAM-based FPGAs.
- However, the uncontrollable clock-skew between GTMR clock domains can cause race conditions that inevitably weaken GTMR mitigation.
- For small (less complex) designs implemented in FPGAs that contain clock trees with minimal T_{skew}, GTMR can be realizable.
- As device and design area increase, e.g., modern devices such as the Xilinx Kintex-7, GTMR clock-skew also increases.
- Some race conditions can be uncontrollable and unrecognizable by manufacturer-supplied design tools.
- Consequently, Kintex-7 GTMR versus DTMR heavy-ion data show that GTMR is an ineffective and unreliable mitigation solution.
- In conclusion, we suggest that DTMR is a more applicable TMR strategy for larger commercial SRAM-based FPGA devices.

Acknowledgements

- *Some of this work has been sponsored by the NASA Electronic Parts and Packaging (NEPP) Program and the Defense Threat Reduction Agency (DTRA).*
- *Thanks is given to the NASA Goddard Radiation Effects and Analysis Group (REAG) for their technical assistance and support. REAG is led by Kenneth LaBel and Jonathan Pellish.*

Contact Information:

Melanie Berg: NASA Goddard REAG FPGA Principal Investigator:

Melanie.D.Berg@NASA.GOV