GFIR-55: WP272 Summary

Think Local, Not Global

While global reset is often considered essential in FPGA designs, it is actually recommended to avoid it and adopt localized reset instead.

(*“So, some may be surprised to learn that applying a global reset to your FPGA designs is not a very good idea and should be avoided.”* ref: Page 1)

Global Reset Timing Issues:

- Global reset signals usually originate from slow and unpredictable sources.
  (*"What are the typical drivers of a global reset signal?*
  • Press switch: Definitely slow and very undefined timing.
  • Power supply status output: Active for a long period until supply is stable.
  • Microprocessor: Pulse tends to be long." ref: Page 2)

- As clock frequencies increase, the de-assertion of the reset becomes a timing-critical event.

  (*"In Figure 1, a reset signal is de-asserted at some time between clock edges. The signal then propagates to the various flip-flops. At each flip-flop, the signal should be de-asserted a “set-up period” before the active clock edge. It is obvious that as the clock rate goes up, the time available to distribute the reset signal goes down.”* ref: Page 2)

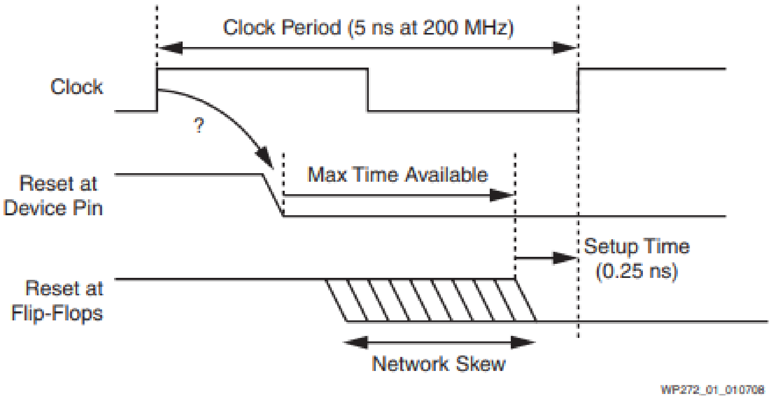

Figure 1: **Reset Timing Diagram - Asserted Between Clock Edges**

- Due to high fan-out networks, it is challenging to meet timing requirements for all flip-flops, potentially causing metastability issues.

  (*"Flip-flops at B are difficult to define and may even lead to metastability." “With increasing clock rates and the potential distribution skew associated with large devices, it becomes almost inevitable that not all flip-flops are released in preparation for the same clock edge (Figure 3).”* ref: Page 3))

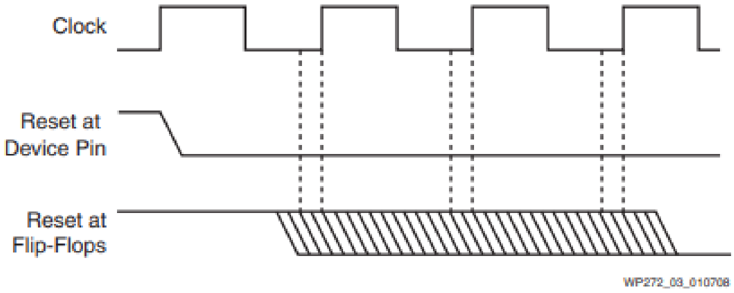

Figure 3: **Reset Timing Diagram - High Clock Rate**

When Does It Matter?

- **Non-Critical Scenarios:** In pipelined data paths. (*"When there is a data flow through a pipelined process, it really doesn't matter when the master reset is released." ref: Page 4*)

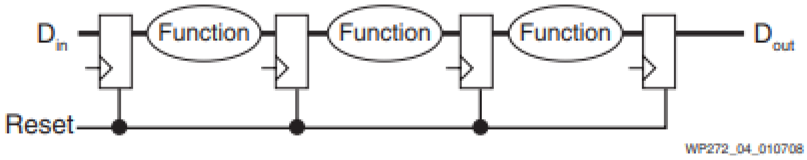

Figure 4: **Reset for a Pipeline**

- **Critical Scenarios:** In circuits with feedback, such as a one-hot state machine, improper reset release timing can cause permanent malfunction or illegal states. (*"In this example of a simple one-hot state machine, there is a clear potential for failure."* ref: Page 4)

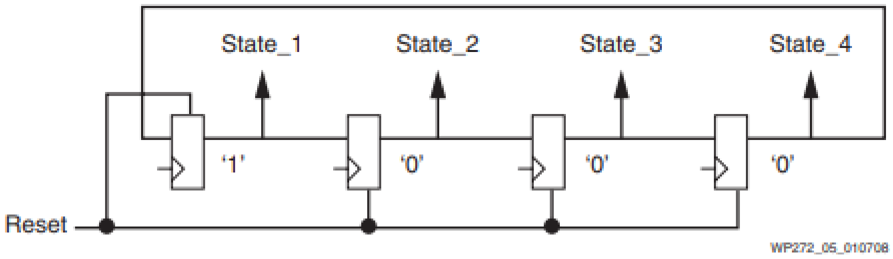

Figure 5: **Reset for a One-Hot State Machine**

Built-in Initialization:

- FPGAs which has RAM automatically initialize all flip-flops and RAM cells during configuration. (*"When a Xilinx FPGA is configured or reconfigured, every cell is initialized. This is the ultimate in master reset..."* ref: Page 5)

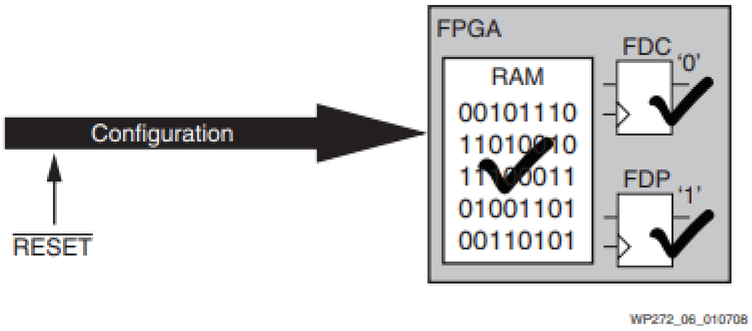

Figure 6: **FPGA Configuration**

Recommended Local Reset Strategy:

- A simple shift register chain can generate a localized reset signal that is released synchronously. (*"The circuit in Figure 7 shows a potentially useful mechanism to control a localized reset network... the shift register chain begins to fill with 0s each clock cycle."* ref: Page 5)

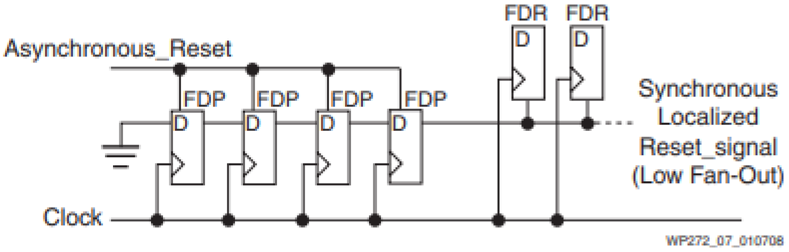

Figure 7: **Localized Reset**

The Hidden Cost of Global Reset:

- Consumes significant routing and logic resources.
  (*"The cost can be significant:*
  • Routing resources of the device are used...
  • Logic resources of the device are used..." ref: Page 6)

- Global reset increases development time and costs. (*"Inserting a global reset will impact development time and final product costs even if time and cost cannot be easily quantified."* ref: Page 7)

- This leads to longer place & route times.

Conclusion:

A global reset network is unnecessary in Xilinx FPGA designs. Instead:

- Identify and reset only the essential circuits with localized networks.
- Avoid redundant resets.
- Rely on the FPGA's built-in initialization, which is more reliable and comprehensive than a traditional global reset.