WP275 - Get Your Priorities Right: Make Your Design Up to 50% Smaller**

- **Single-Level Logic** – Keeping combinational logic compact, designers can achieve higher efficiency and reduce timing delays.

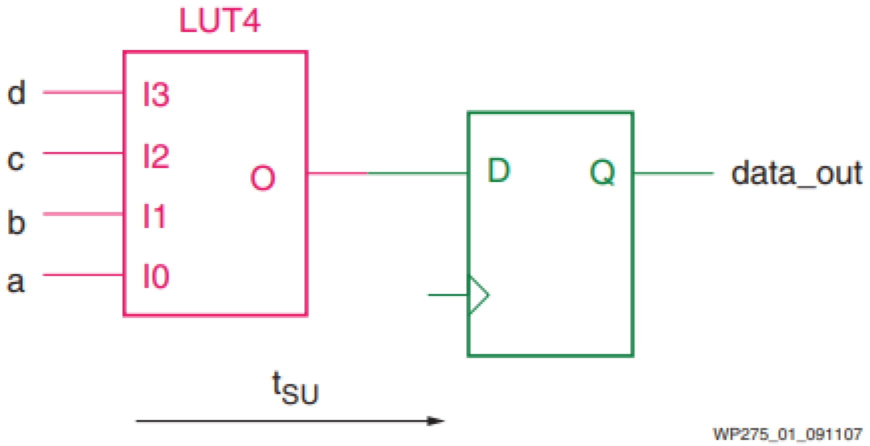

Figure 1: **Simple Logic Using a LUT and a Flip-Flop**

- **Two-Level Logic** – When a logic function requires more than four inputs, the synthesis tool must distribute it across multiple LUTs, leading to increased resource consumption, added propagation delay, and potential timing challenges.

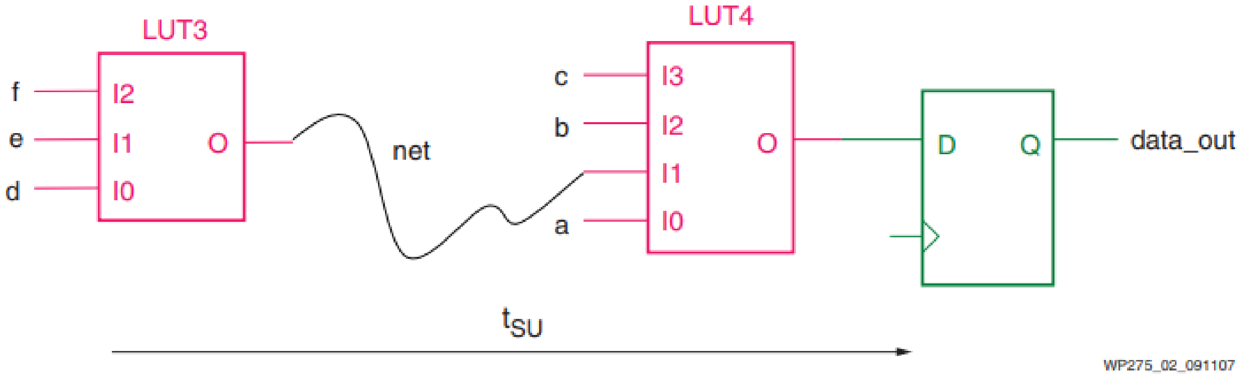

Figure 2: **Two-Level Logic**

- **Adding a Reset**: Unnecessary resets not only waste resources but also contribute to timing inefficiencies.
- **Adding More Controls**: Proper utilization of dedicated flip-flop control inputs, such as clock enables and synchronous set/reset signals, helps optimize logic placement, reducing the

need for additional gates and interconnects.

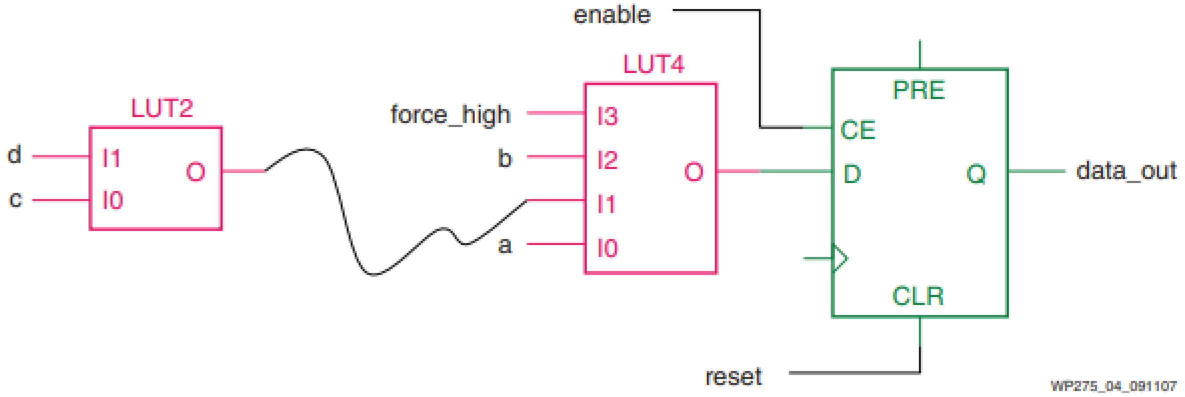

Figure 4: **Additional Controls**

- **Do Not Mix Asynchronous and Synchronous Resets**: Combining asynchronous and synchronous reset structures forces the synthesis tool to implement inefficient logic. It may leads to increased area usage and low timing performance.

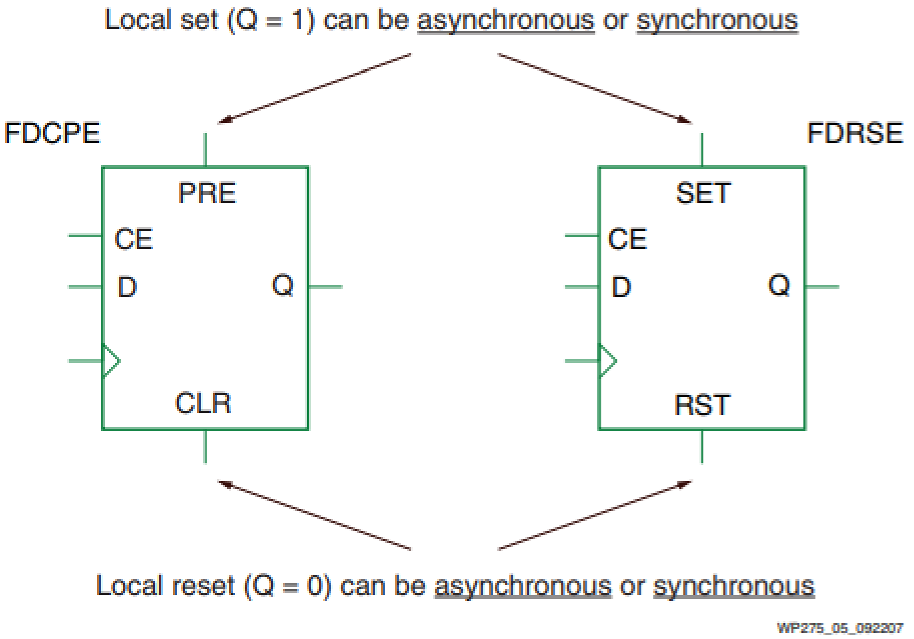
Local set (Q = 1) can be asynchronous or synchronous
FDCPE
FDRSE
PRE
CE
D
Q
CLR
SET
CE
D
Q
RST
Local reset (Q = 0) can be asynchronous or synchronous
WP275_05_092207

Figure 5: **Asynchronous versus Synchronous Controls**

- **Synchronous Design**: Preferring synchronous resets over asynchronous ones for predictable behavior and better performance. Asynchronous resets can introduce metastability issues, while synchronous resets integrate easily into the clocked logic structure.
- **Get Your Priorities Right**: To achieve optimal synthesis results, control signal priorities must align with the FPGA flip-flops. Priorities may lead to unnecessary logic complexity and inefficient resource utilization.
- **The Challenge**: Reviewing HDL code for improper reset usage and unprioritized control structures can significantly reduce design size and improve performance.