Summary of Understanding Metastability in FPGAs

We can say that metastability is an instability that occurs during the transfer of a signal between asynchronous clocks. MTBF helps assess metastability risks. This paper explains its calculation and ways to improve it, enhancing system reliability through design optimizations.

What Is Metastability ?

- Metastability problems commonly occur when a signal is transferred between circuitry in unrelated or asynchronous clock domains
- Certain signal timing requirements must be applied to ensure that data is correctly captured and output along with the signal. Failure to meet these requirements can result in metastability.
- **Note:** At higher clock frequencies, setup and hold time requirements become more critical.
- The likelihood that a register enters a metastable state and the time required to return to a stable state vary depending on the process technology used and the operating conditions. In most cases, registers quickly return to a stable defined state.
-

Figure 1. Metastability Illustrated as a Ball Dropped on a Hill

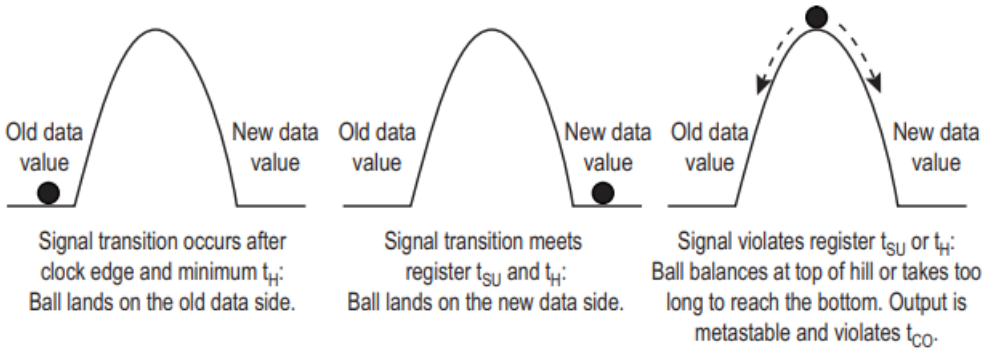
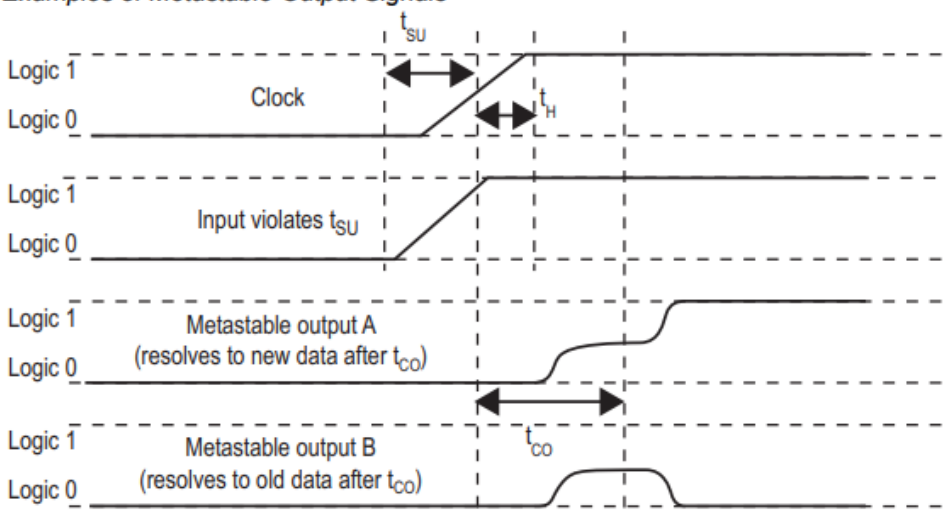


Figure 2. Examples of Metastable Output Signals



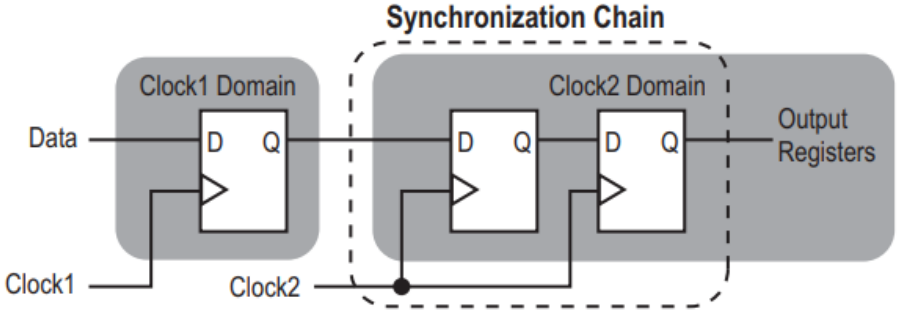
- The section where the output remains in an undefined state before settling to logic 0 or logic 1 can be considered the metastable state.

- **Metastability leads to design failures only if the instability period continues until the next register's data sampling time. However, if the metastability period resolves within the system's timing tolerances, no issue occurs. This interpretation would be correct ?**

Synchronization Registers

A signal that transfers between asynchronous clock domains must be synchronized by the first register before it can be used. To minimize metastability-related failures in asynchronous signal transfers, designers use a series of registers (synchronization register chain) in the destination clock domain to resynchronize the signal. This chain provides extra time for a metastable signal to resolve to a known value. The synchronization register chain consists of a series of registers that are clocked by the same clock, with the first register driven asynchronously and each register only faning out to the next one. The length of the chain is the number of registers that meet these requirements.

Figure 3. Sample Synchronization Register Chain



The fact that asynchronous input signals or signals transferred between unrelated clock domains can transition at any point relative to the capturing register's clock edge limits the designer's knowledge until the data transitions. The designer can address this uncertainty and metastability by using FIFO logic. Altera offers a DCFIFO megafunction for this operation, which provides various levels of latency and metastability protection for control signals. Synchronization registers are used to ensure that metastability has enough settling time by determining when data transfer can occur between clock domains through control signals. The design will function correctly as long as each signal reaches a stable value before being used.

Calculating Metastability MTBF

MTBF (Mean Time Between Failures) estimates the average time between instances when metastability could cause a design failure. A high MTBF is more beneficial for systems. A high MTBF means fewer metastable events.

$$MTBF = \frac{e^{t_{MET}/C_2}}{C_1 \cdot f_{CLK} \cdot f_{DATA}}$$

The overall design MTBF can be determined by the MTBF of each synchronizer chain in the design. The failure rate for a synchronizer is calculated as 1/MTBF, and the failure rate for the entire design is calculated by adding the failure rates of each synchronizer chain.

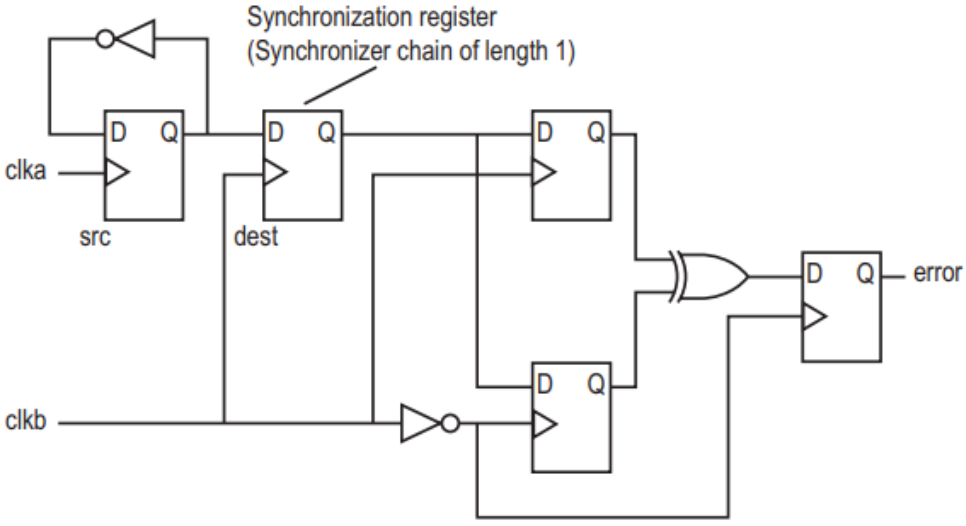
$$failure\_rate_{design} = \frac{1}{MTBF_{design}} = \sum_{i=1}^{number\ of\ chains} \frac{1}{MTBF_i}$$

The design metastability MTBF is calculated as 1/failure_ratedesign. Designers using Altera® FPGAs don't need to perform manual calculations, as Altera's Quartus® II software includes metastability parameters and reports the MTBF for synchronization chains, along with the overall design metastability MTBF.

Characterizing Metastability Constants

In summary, determining metastability constants for FPGAs is challenging because MTBFs span years, making real-time measurements impractical. Altera uses a specially designed test circuit to characterize these constants.

Figure 4. Test Circuit Structure for Metastability Characterization



In this design, clka and clkb are two unrelated clock signals, and the data input to the synchronizer toggles every clock cycle (high fDATA). The synchronizer has a length of

1 because a single synchronizing register sends data to two destination registers. If the signal becomes metastable without resolving at the next clock edge, the circuit detects that the sampled signals are different and generates an error signal. This circuit detects a large portion of the metastability events that occur during the half-clock cycle. This is the general logic.

Improving Metastability MTBF

Metastability can be improved by optimizing the device's C2 constant or increasing the tMET in synchronization registers. As FPGAs transition from 180-nm to 90-nm process geometries, increased transistor speed typically improves metastability MTBF. However, if the FPGA manufacturer does not design the circuit to improve metastability robustness, it can worsen. Altera optimizes the FPGA architecture for metastability MTBF improvement by analyzing metastability, with recent improvements in the 40-nm Stratix® IV FPGA architecture lowering the MTBF C2 constant and increasing robustness.

Design Optimizations

An increase in the tMET value leads to a significant increase in MTBF. The overall MTBF of the design is determined by the synchronizer chain with the lowest MTBF. For example, even if all chains in a design have high MTBF, if one chain has a very low MTBF, the overall design MTBF will be close to that chain's MTBF. In other words, the weakest chain determines the failure rate of the design. A poorly designed or implemented synchronization chain affects the overall metastability MTBF of the design. Therefore, it is important to perform metastability analysis for all asynchronous signals and clock domain transfers. To improve MTBF, designers can add extra register stages to synchronization register chains. This increases the tMET value. Altera recommends using three registers for better metastability protection. Additionally, using the Altera FIFO megafunction can increase metastability protection and latency.

Conclusion

Metastability can occur during signal transfer between asynchronous clock domains. FPGA designers can improve metastability MTBF by increasing timing slack in synchronization registers. Altera characterizes MTBF parameters for its FPGAs and helps optimize design MTBF with Quartus II software.