Multiple clock domain SoCs: Addressing structural defects

Tejas Dave, Amit Jain, Divyanshu Jain, - October 22, 2014

In the previous blog (Synchronization techniques for multi-clock domain SoCs & FPGAs), we studied different types of synchronization techniques to synchronize signals from one clock domain to another. Even when proper synchronization techniques are used in a multi-clock design, they are not sufficient to avoid all possible problems due to structural defects. Hence extra care must be taken by designers while applying these synchronization techniques in real designs. Some of the structural problems that may cause functional errors in multi-clock-based systems follow.

Convergence in the crossover path

Input to any synchronizer must be output of a flip flop operating in source clock domain. As shown in Figure 1, if combinational output is directly fed to any synchronizer it may cause undesirable effects.

![Figure 1 Convergence in crossover path](image-url)
As shown in Figure 2, if combinational logic output is directly connected to synchronizer input, a glitch (shown by red circle) produced by combinational logic may be synchronized in clock “B” domain and it will result in unexpected behavior at destination side logic.

In order to resolve this issue, combinational output must be registered before connecting it to synchronizer input as shown in Figure 3 and corresponding timing diagram is also shown in Figure 4.
Divergence in crossover path

As shown in Figure 5, same signal must not be synchronized in same clock domain more than once. By virtue of metastability, outputs of both synchronizers can consequence in different stable value though both are synchronizing same signal in same clock domain.

As shown in Figure 6, signal “A1-q” is synchronized more than once (B1-C1 & B2-C2) in the same clock “B” domain. But it is not always guaranteed that output of these two synchronizers will change at same time in destination clock domain. One path may take one extra cycle for synchronization due to metastability problem as shown in timing diagram of Figure 6. Due to this, signals “C1-q” and “C2-q” might not be aligned and there is a gap of one clock cycle between them. It may cause problem in destination side logic if there is any relationship between these two signals. Better and safer approach is to synchronize the signal only once in destination domain and distribute it in destination side logic as shown in Figure 7 and corresponding timing diagram is shown in Figure 8.
Figure 6 Timing for divergence in crossover path

Figure 7 Solution for divergence in crossover path

Figure 8 Timing for divergence solution circuit

Reconvergence
Reconvergence

Two different signals generating from same source clock domain or different clock domain are synchronized to same destination clock domain using separate synchronizers and then used in common combinational logic is called as re-convergence of signals. Re-convergence of more than one signal can also cause undesired effects.

As shown in Figure 9, both signals “Sig_1” and “Sig_2” are synchronized in same clock “B” domain using separate pulse synchronizer (either toggle or handshake) and outputs of both pulse synchronizers are used in same combinational logic (AND gate). In this case, both signals “Sig_1” and “Sig_2” are asserted simultaneously for one source clock period and it is expected that both pulse synchronizers will always output pulse at same time in destination clock domain and output of AND gate will be asserted consequently. But this is not always true and cannot be guaranteed.
As shown in timing diagram of Figure 10, there are total three possible scenarios in this case. Because one path may take one extra clock cycle for synchronization compared to other path due to metastability problem. Hence there are possibility during device life cycle that AND gate output may not be asserted even if “Sig_1” and “Sig_2” signals are asserted at same time in source clock domain and it may cause problem in destination side logic. How frequently it will happen? It is not predictable.

In order to avoid this problem, it is better to perform operation on “Sig_1” and “Sig_2” in source clock domain and synchronize the final output from clock “A” domain to clock “B” domain as shown in Figure 11 and corresponding timing diagram is shown in Figure 12.

Note: In Figure 11, there is no need to register AND gate output before giving it to pulse synchronizer because pulse synchronizer is going to latch this signal in clock “A” (source) domain before synchronize it in clock “B” (destination) domain. Please refer to Toggle/Handshake pulse synchronizer circuit in part1 article.
Figure 12 Timing for Re-convergence solution circuit

The authors are engineers at eInfochips.

Also see:

- Synchronizer techniques for multi-clock domain SoCs & FPGAs