# Metastability in MACH Devices

February 2002 Application Note AN8060

## Introduction

A significant number of digital systems must deal with inputs not synchronized to their own internal clocks. These asynchronous signals can arise from any of the various asynchronous protocols which are often used in bus designs, they can result from sharing signals from systems with different clocks, or they may be the response of a system user who is not synchronized with the system. The result can be metastability, a problem that can plague unwary designers. It is not a newly discovered phenomenon, but is normally dealt with somewhat qualitatively, and, unfortunately, is usually ignored as much as possible.

## Causes Of Metastability

The flip-flop setup time is the parameter that is most often at the root of metastability. The setup time is a requirement that data be made available at the input to the flip-flop before the clock signal arrives. The data must not only be there, but must also be stable. In a PLD, the use of an array for the data adds to the setup time. The data passes through the array on its way to the flip-flop (Figure 1). The clock signal, on the other hand, goes directly from the clock pin to the flip-flop. Its path is much shorter than the data path. The setup time is a requirement that the data signal must be given more time to get to the flip-flop before the clock signal. If the published setup time is satisfied, the data arrives at the flip-flop well before the clock, and the output to the flip-flop will change as desired (Figure 2). If the setup time is violated, no guarantee can be made about what the output will do. The output could be normal, since the published setup time is a worst-case number. However, if the timing between the clock and data is just right, the output will be unstable for some time before it settles into a state. Neither the time the output remains unstable nor the final state is predictable (Figure 3). This condition is metastability.

***Figure 1. Clock and Data Paths in a PLD***

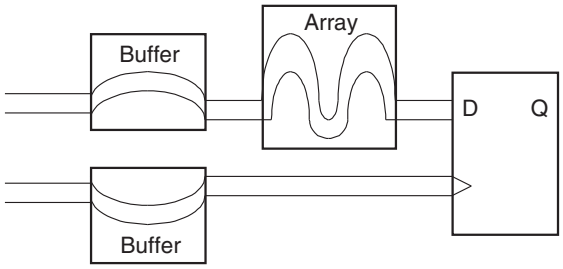
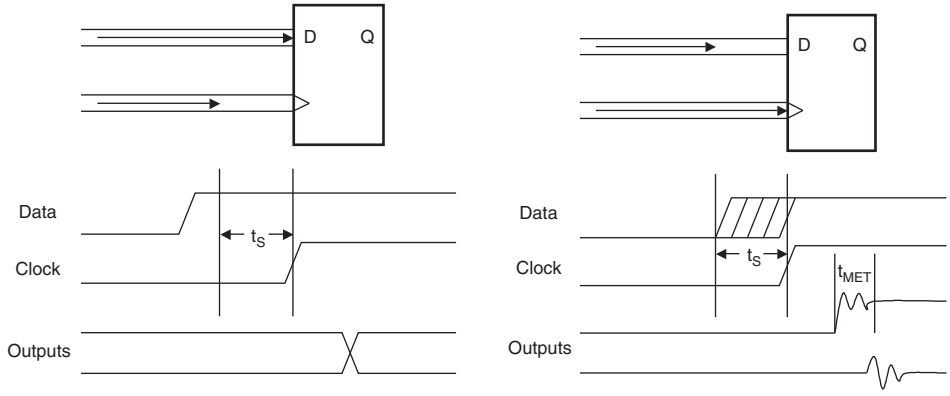



***Figure 2. Output Responses***




**Output Response when Setup Time is Satisfied**

**Possible Output Response when Setup Time is Violated**

## Calculating Metastability

The probability of a metastable event occurring in a design can be calculated and is based on five parameters. The probability is given in terms of a Mean Time Between Failure (MTBF) of two metastable events happening. The formula used to calculate the MTBF is:

$$MTBF = \frac{e^{C2 \times t_{MET}}}{C1 \times f_{CLOCK} \times f_{DATA}}$$

C1 and C2 are parameters that are specific to the process technology and architecture and will be similar for different devices in the same architecture and technology. They are determined by taking empirical data. Values for C1 and C2 for the MACH 4 and MACH 5 families of devices are given in Table 1.

***Table 1. Metastability Constants***

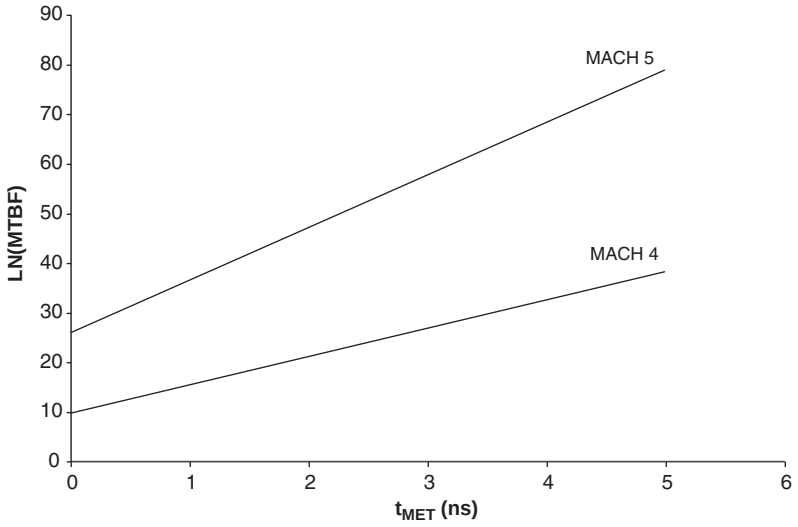
| Family | C1 | C2 |
|---|---|---|
| MACH 4 | $1.01139 \times 10^{-16}$ | $2.391 \times 10^{10}$ |
| MACH 5 | $5.417 \times 10^{-25}$ | $1.07 \times 10^{10}$ |

The time it takes for a register to stabilize out to a known value once a metastable event has occurred is referred to as $t_{MET}$. The register does not necessarily stabilize to the correct value, but it does stabilize to a logic '1' or logic '0.' This is shown in Figure 3.

The clock and data frequencies also affect the probability that a metastable event will occur. In an asynchronous design, these two frequencies are not always known and may have to be estimated.

Using the formula listed above, the logarithmic curves for MTBF can be calculated. Figure 3 shows the curves using $f_{DATA}$ = 1MHz and $f_{CLOCK}$ = 10MHz.

***Figure 3. Metastability Curves for MACH 4 and MACH 5***




To calculate the stabilization time based on the MTBF:

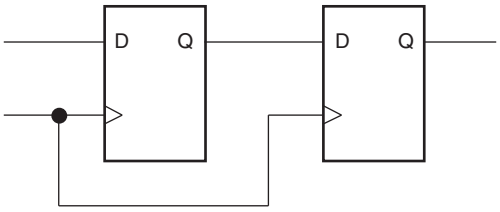
$$t_{MET} = \frac{\ln(MTBF \times f_{CLOCK} \times f_{DATA} \times C1)}{C2}$$

All values should be calculated in terms of seconds and Hertz.

## Solutions For Metastability

The most common solution for metastability is to synchronize the inputs with an extra flip-flop (Figure 4). Ideally, if the first flip-flop goes metastable, the delay between clock pulses will allow the ringing to subside before clocking into the next flip-flop. This improves the chances of having good data in the second flip-flop.

***Figure 4. Dual Synchronizer***




Note that each extra stage of flip-flop means an extra clock delay for the data must be absorbed by the system. This method is not foolproof. The possibility of metastability is reduced, but not eliminated. A flip-flop can go metastable if the preceding stage does not recover quickly enough.

The best way to avoid metastability is to avoid asynchronous clocking when possible. Many applications, such as bus arbitration schemes, use asynchronous clocking because it provides the only convenient way to store data. Unfortunately, this requires a system that is inherently asynchronous and adds some synchronizing elements in the middle.

## Summary

Metastability can occur in a number of asynchronous systems, usually due to the inability to guarantee that the setup time of the flip-flops will be satisfied. In standard synchronous systems where the setup time (and all other timing requirements) is specifically designed in, metastability will never be a problem.

In some situations, metastability is caused by the need to interface systems with different clocks. In this case, it will never be possible to completely eliminate the possibility of metastability. Instead, the designer must take steps to reduce the probability of system failure due to metastability.

## Technical Support Assistance

Hotline: 1-800-LATTICE (North America)
+1-408-826-6002 (Outside North America)
e-mail: techsupport@latticesemi.com
Internet: www.latticesemi.com