Are Your PLDs Metastable?

This application note provides a detailed description of the metastable behavior in PLDs from both circuit and statistical viewpoints. Additionally, the information on the metastable characteristics of Cypress PLDs presented here can help you achieve any desired degree of reliability.

Metastable is a Greek word meaning “in between.” Metastability is an undesirable output condition of digital logic storage elements caused by marginal triggering. This marginal triggering is usually caused by violating the storage elements’ minimum set-up and hold times.

In most logic families, metastability is seen as a voltage level in the area between a logic HIGH and a logic LOW. Although systems have been designed that did not account for metastability, its effects have taken their toll on many of those systems.

In most digital systems, marginal triggering of storage elements does not occur. These systems are designed as synchronous systems that meet or exceed their components’ worst-case specifications. Totally synchronous design is not possible for systems that impose no fixed relationship between input signals and the local system clock. This includes systems with asynchronous bus arbitration, telecommunications equipment, and most I/O interfaces. For these systems to function properly, it is necessary to synchronize the incoming asynchronous signals with the local system clock before using them.

Figure 1 shows a simple synchronizer, whose asynchronous input comes from outside the local system. The synchronizer operates with a system clock that is synchronous to the local system’s operation. On each rising edge of this system clock, the synchronizer attempts to capture the state of the asynchronous input. *Figure 2* shows the expected result. Most of the time, this synchronizer performs as desired.

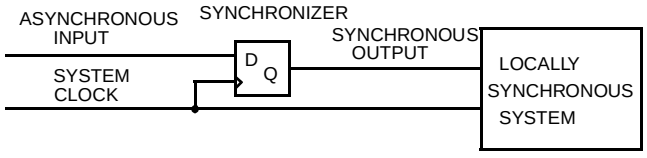


Figure 1. Simple Synchronizer

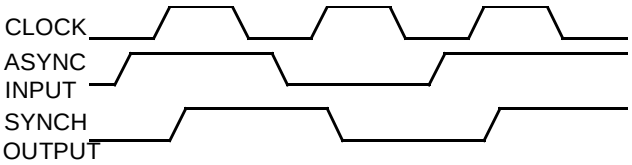


Figure 2. Expected Synchronizer Output

Digital systems are supposed to function properly all the time, however. But because there is no direct relationship between the asynchronous input and the system clock, at some point the two signals will both be in transition at very nearly the same instant. *Figure 3* shows some of the synchronizer’s possible metastable outputs when this input condition occurs. These types of outputs would not occur if the synchronizer made a decision one way or the other in its specified clock-to-output time. A flip-flop, when not properly triggered, might not make a decision in this time. When improperly triggered into a metastable state, the output might later transition to a HIGH or a LOW or might oscillate.

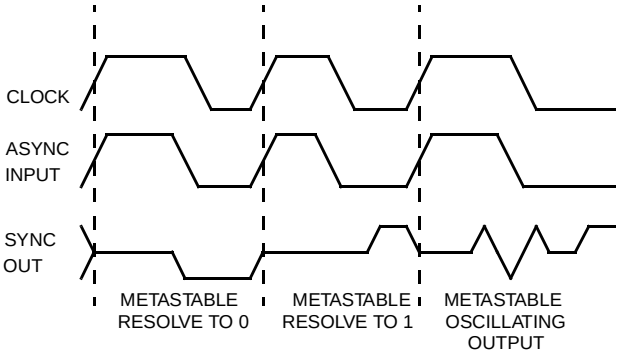


Figure 3. Possible Metastable States of Synchronizer

When other components in the local system sample the synchronizer’s metastable output, they might also become metastable. A potentially worse problem can occur if two or more components sample the metastable signal and yield different results. This situation can easily corrupt data or cause a system failure.

Such system failures are not a new problem. In 1952, Lubkin (Reference 1) stated that system designers, including the designers of the ENIAC, knew about metastability. The accepted solution at that time was to concatenate an additional flip-flop after the original synchronizer stage (*Figure 4*). This added flip-flop does not totally remove the problem but does improve reliability. This same solution is still in wide use today.

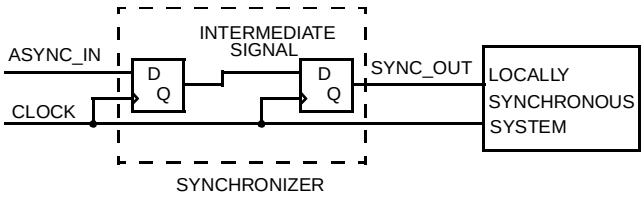


Figure 4. Two-Stage Synchronizer

Recovery from metastability is probabilistic. In the improved synchronizer, the first flip-flop's output might still be in a metastable state at the end of the sample clock period. Because the flip-flops are sequential, the probability of propagating a metastable condition from the second flip-flop stage is the square of the probability of the first flip-flop remaining metastable for its sample clock period. This type of synchronizer does have the drawback of adding one clock cycle of latency, which might be unacceptable in some systems.

As system speeds increase and as more systems utilize inputs from asynchronous external sources, metastability-induced failures become an increasingly significant portion of the total possible system failures. So far, no known method totally eliminates the possibility of metastability. However, while you cannot eliminate metastability, you can employ design techniques that make its probability relatively small compared with other failure modes.

Explanation of Metastability

In a flip-flop, a metastable output is undefined or oscillates between HIGH and LOW for an indefinite time due to marginal triggering of the circuit. This anomalous flip-flop behavior results when data inputs violate the specified set-up and hold times with respect to the clock.

In the case of a D-type flip-flop, the data must be stable at the device's D input before the clock edge by a time known as the set-up time, t_s. This data must remain stable after the clock edge by a time known as the hold time, t_h (*Figure 5*). The data signal must satisfy both the set-up and hold times to ensure that the storage device (register, flip-flop, latch) stores valid data and to ensure that the outputs present valid data after a maximum specified clock-to-output delay t_{co_max}. As used in this application note, t_{co_max} refers to the interval from the clock's rising edge to the time the data is valid on the outputs. In most cases, t_{co_max} refers to the maximum t_{co} specified by a data sheet, as opposed to the average or typical t_{co} value.

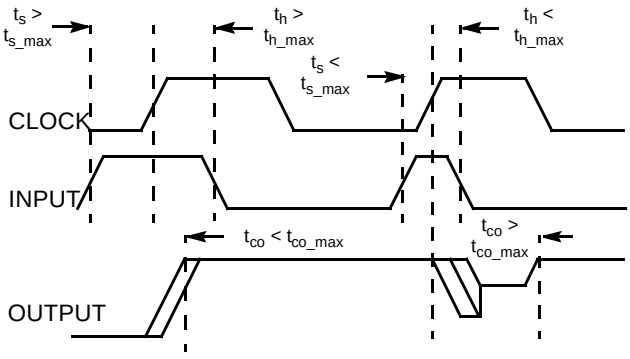


Figure 5. Triggering Modes of a Simple Flip-Flop

If the data violates either the set-up or hold specifications, the flip-flop output might go to an anomalous state for a time greater than t_{co_max} (*Figure 5*). The additional time it takes the outputs to reach a valid level can range from a few hundred picoseconds to tens of microseconds. The amount of additional time beyond t_{co_max} required for the outputs to reach a valid logic level is known as the metastable walk-out time. This walk-out time, while statistically predictable, is not deterministic.

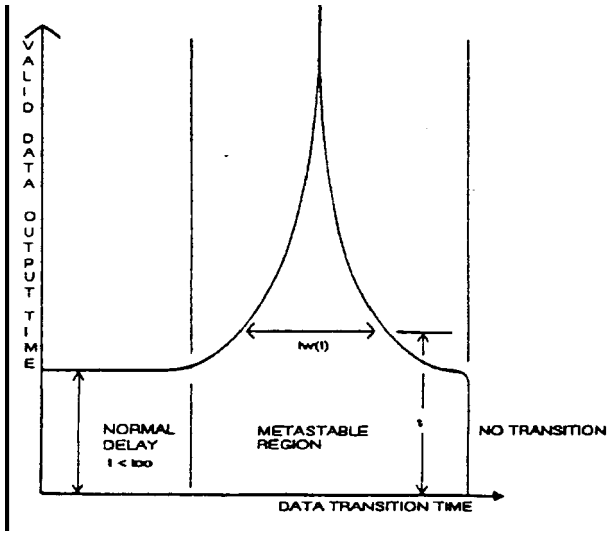


Figure 6. Output Propagation vs. Data Transition

Figure 6, from Reference 2, shows the variation in output delay with data input time. The left portion of the graph shows that when the data meets the required set-up time, the device has valid output after a predictable delay, which equals t_{co}. The middle portion of the graph indicates the metastable region. If the data transitions in this region, valid output is delayed beyond t_{co_max}. The closer the input transitions to the center of the metastable region, violating the device's triggering requirements, the longer the propagation delay. If the data transitions after the metastable region, the device does not recognize the input at that clock edge, and no transition occurs at the output. As given in Reference, you can predict the region t_w, where data transitions cause a propagation delay longer than t, from the formula:

$$t_w = t_{co} e^{\frac{-(t - t_{co})}{\tau}} \qquad \text{Eq. 1}$$

where t depends on device-specific characteristics such as transistor dimensions and the flip-flop's gain-bandwidth product.

Figure 7 shows another way of looking at metastability. A flip-flop, like any other bistable device, has two minimum-potential energy levels, separated by a maximum-energy potential. A bistable system has stability at either of the two minimum-energy points. The system can also have temporary stability—metastability—at the energy maximum. If nothing pushes the system from the maximum-energy point, the system remains at this point indefinitely.

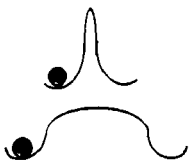


Figure 7. Triggering Modes of a Simple Flip-Flop

A hill with valleys on either side is another bistable system. A ball placed on top of the hill tends to roll toward one of the minimum-energy levels. If left undisturbed at the top, the ball can remain there for an indeterminate amount of time. As this figure indicates, the characteristics of the top of the hill as well as natural factors affect how long the ball stays there. The steepness of the hill is analogous to the gain-bandwidth product of the flip-flop's input stage.

Causes of Metastability

Systems with separate entities, each running at different clock rates, are called globally asynchronous systems (Reference 4). The entities might include keyboards, communication devices, disk drives, and processors. A system containing such entities is asynchronous because signals between two or more entities do not share a fixed relationship.

Metastability can occur between two concurrently operating digital systems that lack a common time reference. For example, in a multiprocessing system, it is possible that a request for data from one system can occur at nearly the exact moment that this signal is sampled by another part of the system. In this case, the request might be undefined if it does not obey the set-up and hold time of the requested system.

When globally asynchronous systems communicate with each other, their signals must be synchronized. Arbitration must occur when two or more requests for a shared resource are received from asynchronous systems. An arbiter decides which of two events should be serviced first. A synchronizer, which is a type of arbiter with a clock as one of the arbited signals, must make its decision within a fixed amount of time. A device can synchronize an input signal from an external, asynchronous device in cases such as a keyboard input, an external interrupt, or a communication request.

Care must be taken when two locally synchronous systems communicate in a globally asynchronous environment. A synchronization failure occurs when one system samples a flip-flop in the other system that has an undefined or oscillating output. This event can distribute non-binary signals through a binary system (Reference 5).

In synchronizers, the circuit must decide the state of the data input at the clock input's rising edge. If these two signals arrive at the same time, the circuit can produce an output based on either decision, but must decide one way or the other within a fixed amount of time.

Attacking Metastability

The design of synchronous systems is much different than the design of globally asynchronous systems. The design of a synchronous digital system is based on known maximum propagation delays of flip-flops and logical gates. Asynchronous systems by definition have no fixed relationship with each other, and therefore, any propagation delay from one locally synchronous system to the next has no physical meaning.

Two different methods are available to produce locally synchronous systems from globally asynchronous systems. The first method involves creating self-timed systems. In a self-timed system, the entity that performs a task also emits a signal that indicates the task's completion. This handshaking signal allows the use of the results when they are ready instead of waiting for the worst-case delay. Such handshaking signals allow communications between locally synchronous systems.

The advantage of the self-timed method is that it permits machines to run at the average speed instead of the worst-case speed. The disadvantages are that a self-timed system must have extra circuitry to compute its own completion signals and extra circuitry to check for the completion of any tasks assigned to external entities.

Petri Nets, data flow machines, and self-timed modules all use the self-timed method of communication among locally synchronous systems. Self-timed structures do not completely eliminate metastability, however, because they can include arbiters that can be metastable. Most systems do not include self-timed interfaces due to the additional circuitry and complexity.

The second method of producing locally synchronous systems from globally asynchronous systems is the simple synchronizer. This is the most common way of communicating between asynchronous objects. The metastability errors that might arise from these systems must be made to play an insignificant role when compared with other causes of system failure.

Many metastability solutions involve special circuits (References 6 and 7). Some of these solutions do not reduce metastability at all (References 13 and 8). Others, however, do reduce metastability errors by pushing the occurrence of metastability to a place where sufficient time is available for resolving the error. Most of these circuits are system dependent and do not offer a universal solution to metastability errors.

The easiest and the most widely used solution is to give the synchronizing circuit enough time to both synchronize the signal and resolve any possible metastable event before other parts of the system sample the synchronized output. This solution requires knowledge of the metastable characteristics of the device performing the synchronization.

Many semiconductor companies have developed circuits such as arbiters, flip-flops, and latches that are specifically designed to reduce the occurrence of metastability. Although these parts might have good metastability characteristics, they have very limited application. The circuits can only function as flip-flops or arbiters and do not have the flexibility of PLDs. Cypress Semiconductor has designed the flip-flops in the company's PLDs to be metastability hardened. This allows you to use Cypress PLDs in a wide range of systems requiring synchronization.

Circuit Analysis of Metastability

Many authors have written papers detailing the analysis of metastability from a circuit standpoint (References 5, 7, 8, 9, 10, 11, and 12). In Reference 11, for example, Kacprzak presents a detailed analysis of an RS flip-flop's metastable operation. He states that a flip-flop has two stages of metastable operation (*Figure 8*).

During the initialization phase, the Q and $\overline{Q}$ outputs move simultaneously from their existing levels to the metastable voltage V_m, which is the voltage at which $V_q = \overline{V}_q$.

The second or resolving phase occurs when the outputs once again drift toward stable voltages. Once a flip-flop has entered a metastable state, the device can stay there for an indeterminate length of time. The probability that the flip-flop will stay

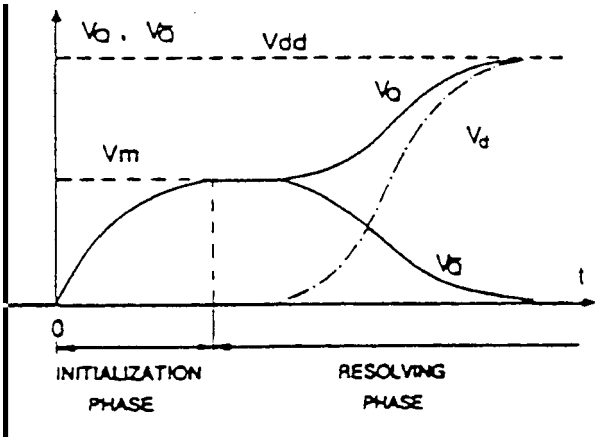


Figure 8. Two Phases of Metastability

metastable for an unusually long period of time is zero, however, due to factors such as noise, temperature imbalance within the chip, transistor differences, and variance in input timing. During the second phase of metastability, for very small deviations around the metastable voltage, V_m, the flip-flop behaves like two cross-coupled linear amplifier stages that gain $V_d = V_q - \overline{V}_q$. When the gain of the cross-coupled loop exceeds unity, the differential voltage increases exponentially with time.

The length of time the flip-flop takes to resolve cannot be exactly determined. The probability that the flip-flop will resolve within a specific length of time, however, can be predicted. This probability depends on the electrical parameters of the flip-flop acting as a linear amplifier around the metastability voltage. The solution (Reference 11) to the differential voltage $V_d(t)$ driving the resolving phase is given by

$$V_d(t) = V_d(t_0)e^{\frac{(t-t_0)}{\tau}} \qquad \text{Eq. 2}$$

where t depends directly on the amplifier gain and capacitance, and where $V_d(t_0)$ represents the differential voltage at some time t_0. You can use this equation to determine the length of time that the output voltage will take to drift from the metastable voltage V_m to a specified voltage difference V_d.

Horstmann (Reference 5) states that a flip-flop, like any other system with two stable states, can be described by an energy function with two local energy minima where P(x) = 0 (*Figure 9*). Any bistable system has at least one metastable state, which is an unstable energy level within the system and represents the local maximum of the energy function. The system's gradient can be represented by a force, F(x), that is zero at stable and metastable states (inflection points of the energy function).

Figure 10 shows a simplified first-order model of an RS flip-flop used to predict and visualize metastability. A flip-flop energy transfer curve (*Figure 11*) shows the relationship between the two outputs. The two stable states are local energy minima of the system. The metastable state, M, is a local energy maximum and represents an unstable state with loop gain near M that is greater than one.

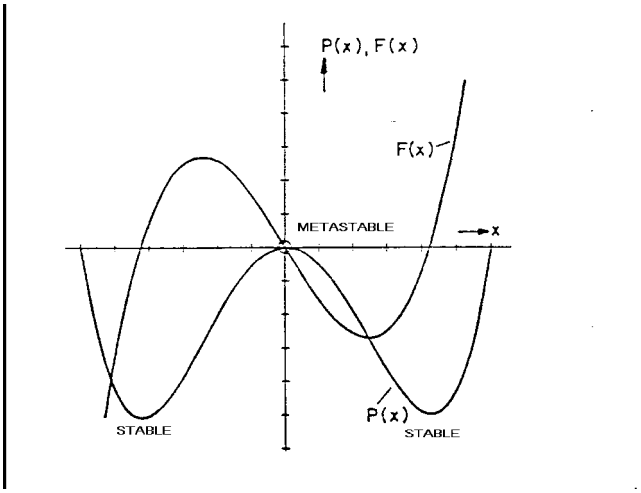


Figure 9. Energy/Force Function of a Bistable System

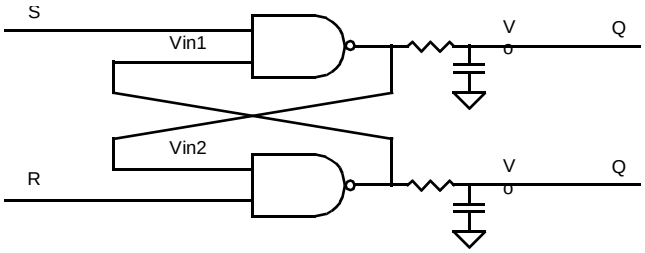


Figure 10. First-Order Flip-Flop

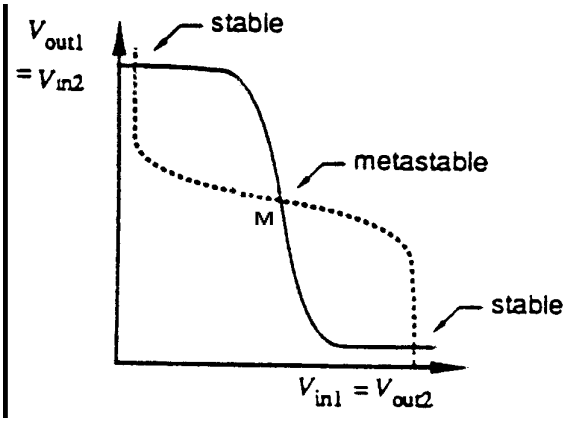


Figure 11. Energy Transfer Diagram of Simple RS Flip-Flop

Figure 12 shows the trigger line for the first-order approximation of the flip-flop. The dashed line RS represents the device's normal trigger line, which does not follow the transfer curve because, during triggering, the feedback loop has not been established. If at varying points along the trigger line the feedback loop is re-established, the nodes of the device follow the curves that lead to the line $S_0 - S_1$. Once on this line, the circuit exponentially drifts toward stability at either S_0 or S_1, depending on which side of the line Q = $\overline{Q}$ the feedback loop

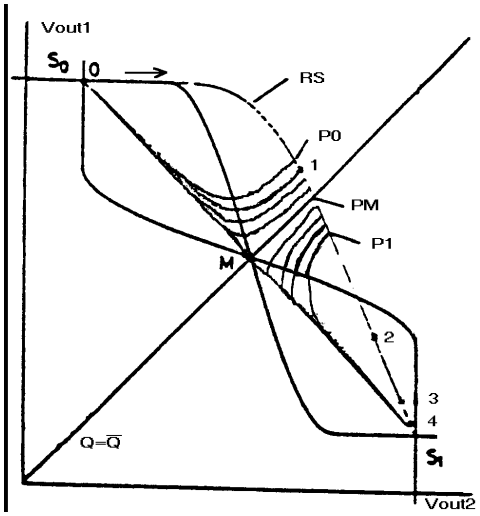


Figure 12. Energy Transfer Curves showing Trigger Paths

was re-established. The curves are solutions to the first-order model circuit equations for the device shown in *Figure 10.*

When the feedback loop is restored near the line Q = $\overline{Q}$, the system moves toward the unstable state M and can take an indefinite amount of time to exit from this metastable state. You can see this from the graph by noticing that S_0 and S_1 are equally likely solutions for system stability from M. Once the feedback loop is re-established, the system exponentially decays toward M and then exponentially grows toward S_0 or S_1.

Figure 13 shows the system's possible trigger events using the implied time scale of the state-space curves. The solution of these simplified first-order equations indicates that the fastest metastable resolution time occurs when the circuit's gain-bandwidth product is maximized.

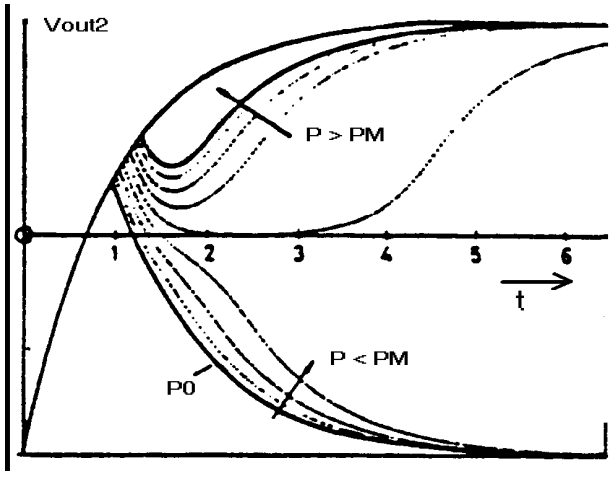


Figure 13. Time Scale Showing Trigger Paths

Flannagan (Reference 12), in an attempt to maximize the gain-bandwidth product, solves simplified flip-flop equations to determine the phase trajectory near the metastable point. His results, which are supported by other authors, indicate that p and n devices with equal geometries produce the optimal gain-bandwidth product for metastable event resolution.

Statistical Analysis of Metastability

To begin the analysis of metastability, assume that the flip-flop's probability of resolving its metastable state does not depend on its previous metastable state. In other words, the metastable device has no memory of how long it has been in a metastable region. The analysis of metastability also assumes that the flip-flop's probability of resolving its metastable state in a given time interval does not depend on the metastable resolution in another disjoint time interval. The probability that a metastable event will resolve in a given interval (0,t) is only proportional to the length of the interval.

These assumptions yield an exponential distribution that describes the probability that the flip-flop resolves its metastability at a time t. The exponential distribution has the form

$$f(x) = \mu e^{-\mu t} \qquad \text{Eq. 3}$$

where m is the expected value of metastability resolution per unit time (settling rate).

Using this equation and given that the flip-flop was metastable at time t = 0, the probability of a metastable event lasting a time t or longer is

$$P(met_t | met_{t=0}) = \int_t^{\infty} \mu e^{-\mu t} dT = e^{-\mu t} \qquad \text{Eq. 4}$$

The next part of the analysis involves the probability that the flip-flop is metastable at time t = 0. This part of the analysis assumes that the probability that the data transitions in a given time interval depends only on the length of the interval. A Poisson process with rate f_d describes the probability of the data transitioning at a time t:

$$p(x) = \frac{e^{-f_d t}(f_d t)^x}{x!} \qquad \text{Eq. 5}$$

where *x* is the number of transitions.

If a data transition within a bounded time interval, W, of the clock edge causes a metastable condition, the expected number of transitions of this Poisson process with rate f_d in time interval W is

$$E(X) = \sum_{x=0}^{\infty} \frac{x e^{-f_d W}(f_d W)^x}{x!} = f_d W \qquad \text{Eq. 6}$$

Because this expected number of transitions is the same as the probability that the flip-flop is metastable at t = 0, the equation for the probability at t = 0 is

$$P(met_{t=0}) = f_d W \qquad \text{Eq. 7}$$

Using *Equations 5* and 7, the probability that a given clock cycle results in metastability that lasts at most a time t is

$$P(met_t) = P(met_t | met_{t=0}) P(met_{t=0}) = f_d W e^{-\mu t} \qquad \text{Eq. 8}$$

Substituting $\frac{1}{t_{sw}}$ for μ allows this variable to be expressed as a settling time constant of the flip-flop. Further, a synchronization failure for a given clock cycle exists whenever a metastable event lasts a specified time (t_r) or longer. Using these two substitutions, the probability that the flip-flop is metastable in a given clock cycle is

$$P(fail_{1\ clock}) = f_d W e^{\frac{-t_r}{t_{sw}}} \quad \text{Eq. 9}$$

Because the data transitions are independent, the number of failures in n clock cycles has a binomial distribution with an expected number of failures:

$$E(fail_{n\ cycles}) = nP(fail_{1\ cycle}) \quad \text{Eq. 10}$$

Assuming a sample clock frequency, f_c, that represents the number of clock cycles, n, per unit time, the expected number of failures per unit time is

$$E(fail_{unit\ time}) = f_c f_d W e^{\frac{-t_r}{t_{sw}}} \quad \text{Eq. 11}$$

Assuming that all data transitions are independent and that the clock has a fixed period, the mean time between failures (MTBF) is

$$MTBF = \frac{1}{E(fail_{unit\ time})} = \frac{e^{\frac{t_r}{t_{sw}}}}{f_c f_d W} \quad \text{Eq. 12}$$

where MTBF is a measure of how often, on the average, a metastable event lasts a time t_r or longer.

Metastability Data

The strong resemblance between *Equation 12* and *Equation* 2 is based on the predictions of the first-order circuit analysis of an RS flip-flop. In fact, the metastability resolving time constant, t_{sw}, is directly related to the variable t, which is based on the flip-flop's gain-bandwidth product.

The device-dependent variable W depends mostly on the window of time within which the combination of the input and clock generate a metastable condition. This parameter also depends on process, temperature, and voltage levels. The MTBF equation is usually plotted with t_r (the resolving time allowed for metastable events) on the X axis and the natural log of the MTBF plotted on the Y axis (see the appendix in this note). Because the metastability equation is plotted on a semi-log scale, the graph of t_r vs ln(MTBF) is a line described by the equation

$$\ln(MTBF) = \frac{t_r}{t_{sw}} - \ln(f_c f_d W) \quad \text{Eq. 13}$$

Graphically, the parameter t_{sw} is 1/slope of the line on this graph. The equation for t_{sw} from the graph is

$$t_{sw} = \frac{t_{r1} - t_{r2}}{\ln(MTBF_1) - \ln(\mathrm{MTBF}_2)} \quad \text{Eq. 14}$$

To determine how often, on the average, a given synchronizer in a system will go metastable (MTBF), you must know the two device-specific parameters W and t_{sw}, which should be available from the manufacturer. *Table 1*, discussed later in this note, lists these values for Cypress PLDs. Additional values you need are the average frequency of both the system data and the synchronizer clock and the amount of time after the synchronizer's maximum clock-to-Q time that is allowed to resolve metastable events.

For example, consider the method for determining the MTBF for a Cypress PALC22V10 registered PLD used as a synchronizer in a system with the following characteristics:

W = 0.125 ps

t_{sw} = 190 ps

f_c = system clock frequency = 25 MHz

f_d = average asynchronous data frequency = 10 MHz

In addition to these values, the PLD's maximum operating frequency, f_{max}, is taken directly from the data sheet. The frequency is specified as the internal feedback maximum operating frequency. It is calculated as

$$f_{max} = \frac{1}{t_{cf} + t_s} = 41.6MHz$$

where t_{cf} is the clock-to-feedback time. If the data sheet does not specify t_{cf}, you can use t_{co} as t_{cf}'s upper bound.

Using f_{max}, you calculate the amount of time that a metastable event is allowed to resolve, t_r, with

$$t_r = \frac{1}{f_c} - \frac{1}{f_{max}} = \frac{1}{25MHz} - \frac{1}{41.6MHz} = 16ns$$

Now you enter these values into the MTBF equation, making sure to keep all units in seconds:

$$MTBF = \frac{e^{\frac{t_r}{t_{sw}}}}{f_c f_d W}$$

$$= \frac{e^{\frac{16 \times 10^{-9} s}{190 \times 10^{-12} s}}}{25 \times 10^6 s^{-1} \times 20 \times 10^6 s^{-1} \times 0.125 \times 10^{-12} s}$$

$$= 59.7 \times 10^{33} s$$

$$= 1.89 \times 10^{27}\ years = Almost\ forever$$

If the operating frequency of the system, f_c, is simply changed to 33.3 MHz,

$$MTBF = \frac{e^{\frac{6 \times 10^{-9} s}{190 \times 10^{-12} s}}}{33.3 \times 10^6 s^{-1} \times 20 \times 10^6 s^{-1} \times 0.125 \times 10^{-12} s}$$

$$= 623 \times 10^{27}\ s$$

the system fails, on the average, about every 19,700 years—still beyond the system's normal lifetime.

And if f_c is changed to f_{max} (41.6 MHz),

$$MTBF = \frac{e^{\frac{0 \times 10^{-9} s}{190 \times 10^{-12} s}}}{41.6 \times 10^{6} s^{-1} \times 20 \times 10^{6} s^{-1} \times 0.125 \times 10^{-12} s}$$

the system fails, on the average, every 9.62 ms.

A 16-ns difference in resolve time, t_r, results in almost 36 orders of magnitude difference in MTBF. Obviously, accurate data is needed to design a system with a high degree of reliability without being overly cautious.

Characterization of Metastability

Many authors (References 6, 8, 9, 10, 11, and 12) have performed numerous experiments on circuits to predict the likelihood of device metastability. These researchers have used several testing theories and apparatus that can be classified into three basic types (Reference 14).

Intermediate voltage sensors constitute the first type. Two voltage comparators determine whether the output voltage, Q, lies between two given voltages. The fixture produces an error output if Q has a level that is neither HIGH nor LOW, hence metastable. *Figure 14* shows an intermediate voltage sensor.

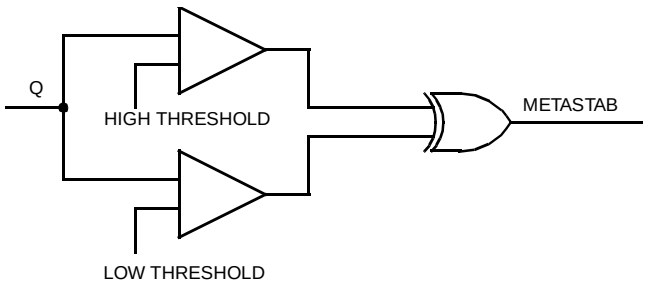


Figure 14. Intermediate Voltage Sensor

The second type of apparatus uses an output proximity sensor to determine if the Q and $\overline{Q}$ outputs have approximately the same voltages, which would indicate that the device is metastable. *Figure 15* shows an output proximity sensor.

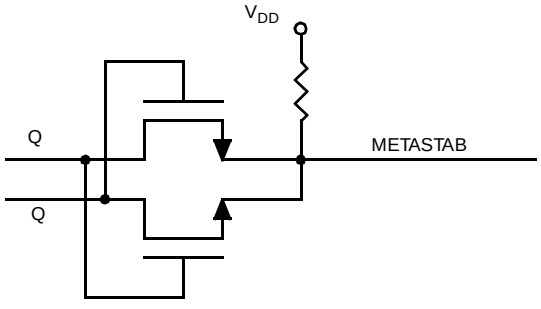


Figure 15. Output Proximity

The last type of apparatus uses a late-transition sensor to test for metastability. Note that if one or more gates separate the sensor from the metastable signal, the metastability might not be detected. The test circuitry must infer the occurrence of metastability by some other means. *Figure 16* shows an example of a late-transition sensor. The sample input is detected at time t_1, then at a later time t_2. If these two signals disagree, the device under test was metastable at t_1.

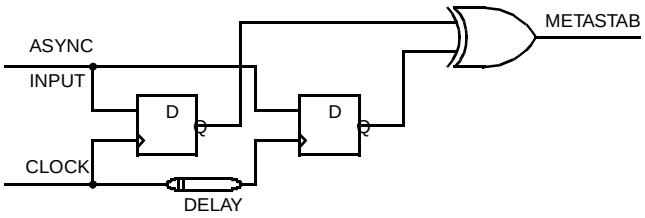


Figure 16. Late-Transition Sensor

Information from Manufacturers

Many semiconductor companies provide metastability data on their parts. However, most companies do not present the data in a format the engineer can use. They either present inconclusive and incomplete data or they assume the engineer can use the data without further explanation. Few companies compare their devices with similar devices.

PLD manufacturers provide little data largely because of a fear that telling the design community that devices can fail in synchronizing applications will cause designers to use a competitor's parts. The truth is that no company can provide a device that is guaranteed never to become metastable when used as a synchronizer. At a given operating frequency, with a given asynchronous input, and given enough time, the device becomes metastable.

Cypress provides you with data you can use to build a system to any given level of reliability when using Cypress PLDs. Cypress has performed numerous tests and collected extensive data on Cypress PLDs, as well as PLDs from other companies. This data gives you a perspective of the parts that are best suited for a specific application. Specific data on the metastability characteristics of Cypress PLDs is found in this application note in the Test Results section.

The Test Circuit

Cypress uses a test that falls into the category of the late-transition detection. Directly measuring the outputs of the flip-flop in a PLD are impossible due to the additional circuitry that lies between the flip-flop and the outside world. The metastability detection circuitry must, instead, infer the flip-flop's state.

Figure 17 shows the metastability test circuit implemented in each test PLD. This circuit allows the PLD under test to effectively test itself. The device under test will both produce and record metastable conditions.

Figure 18 is a state diagram showing the operation of the device. During normal operation, the two flip-flops' outputs (F_1, F_2) transition between states S_1 and S_2, depending on the synchronizer's state. During normal operation, the Exclusive-OR on these outputs produces a HIGH. This indicates either that metastability has not occurred within the device or that metastability that has occurred has resolved before the next clock cycle.

If a metastable event cannot resolve before the next clock cycle, the state machine move to states S_3 or S_4. In this case,

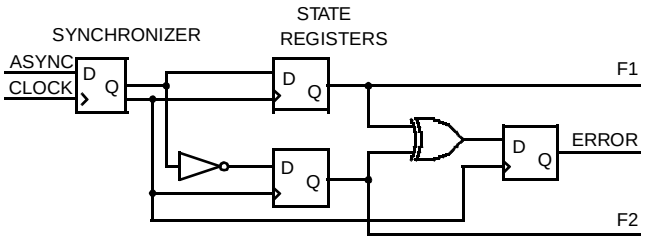


Figure 17. Metastability Test Circuit

the state flip-flops have interpreted the signal from the synchronization register differently; exclusive-ORing this signal produces a LOW at the device's output, indicating that unresolved metastability has occurred.

This test circuit does not catch all metastable events. Specifically, it does not record metastable events that resolve before the next clock cycle. But metastability causes an error only when it has not resolved by the time the signal is needed. The Cypress tests thus reveal the information designers need to know: how often metastability creates an error in the system.

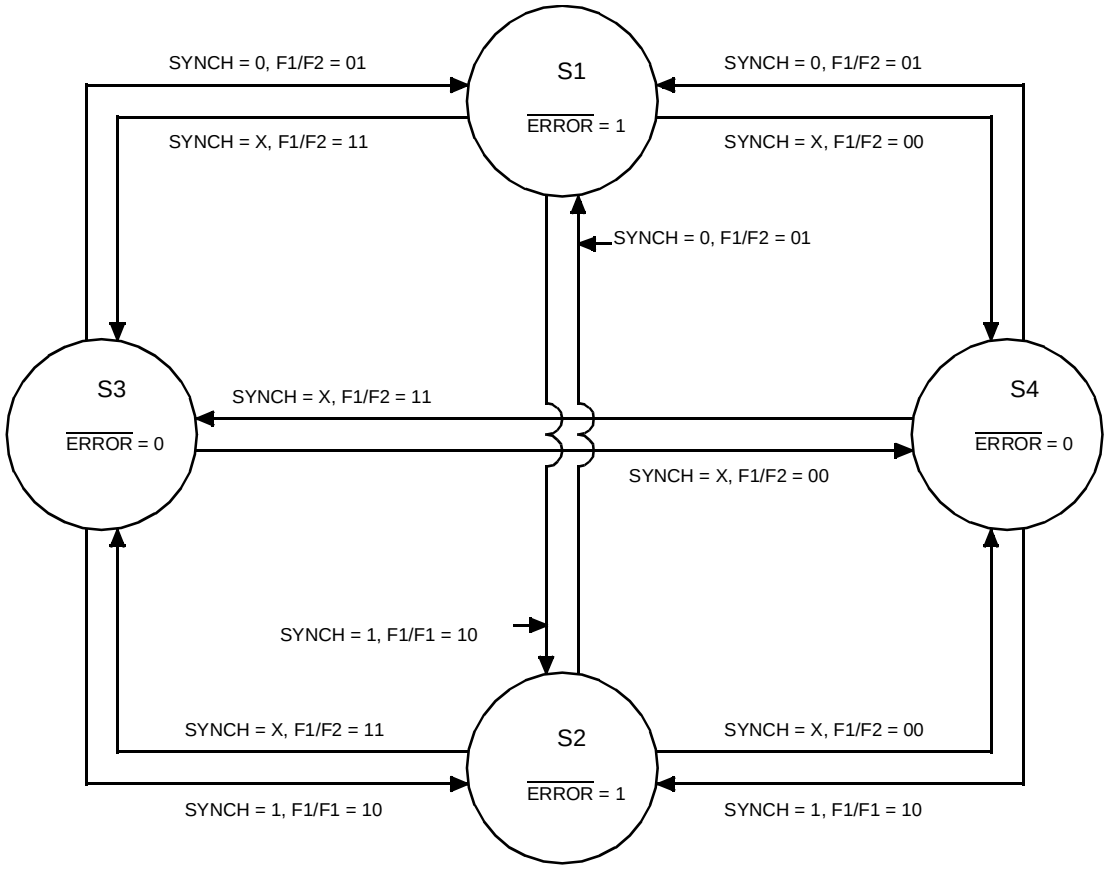


Figure 18. Metastability Testing State Diagram

The test circuit also includes the ability to check the maximum operating frequency of the device under test (*Figure 19*). At each clock edge, the first register's output toggles. When the device reaches its maximum operating frequency, the PLD array cannot resolve the changing signal fast enough to produce a valid output. At this speed, one register might resolve the signal correctly and one might not, or both might produce invalid signal resolutions. In any case, when Exclusive-ORing the state T_1/T_2 of the two maximum-frequency testing registers results in anything other than a HIGH, the part's maximum operating frequency is exceeded.

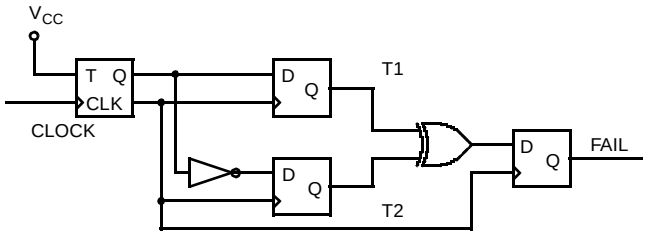


Figure 19. Maximum Operating Frequency Test

The Test Board

A four-layer printed circuit board with two signal planes, a ground plane, and a power plane is used to perform the metastability measurements. Using this four-layer board gives a quiet testing environment with reliable, repeatable results. *Figure 20* shows a block diagram of the test board, with the

complete schematic shown in *Figure 21*. The device under test (DUT) is decoupled with 0.01-mF and 100-pF capacitors. The test circuit is designed to fit all industry-standard and Cypress-proprietary PLDs. The socket allows DUT pins 1, 2, and 4 to serve as clock pins. Pin 3 is the device's asynchronous input. The ERROR condition is located on pin 27 of a 28-pin device, and the FAIL condition is on pin 20. Two additional outputs, F_1 and F_2, monitor the state of the metastability test circuit flip-flops.

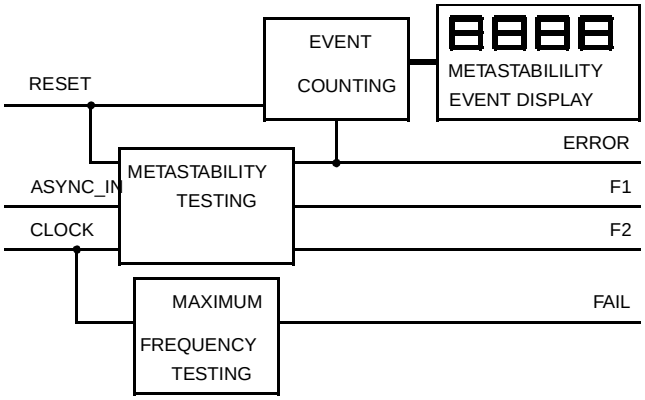


Figure 20. Metastability Test Board Block Diagram

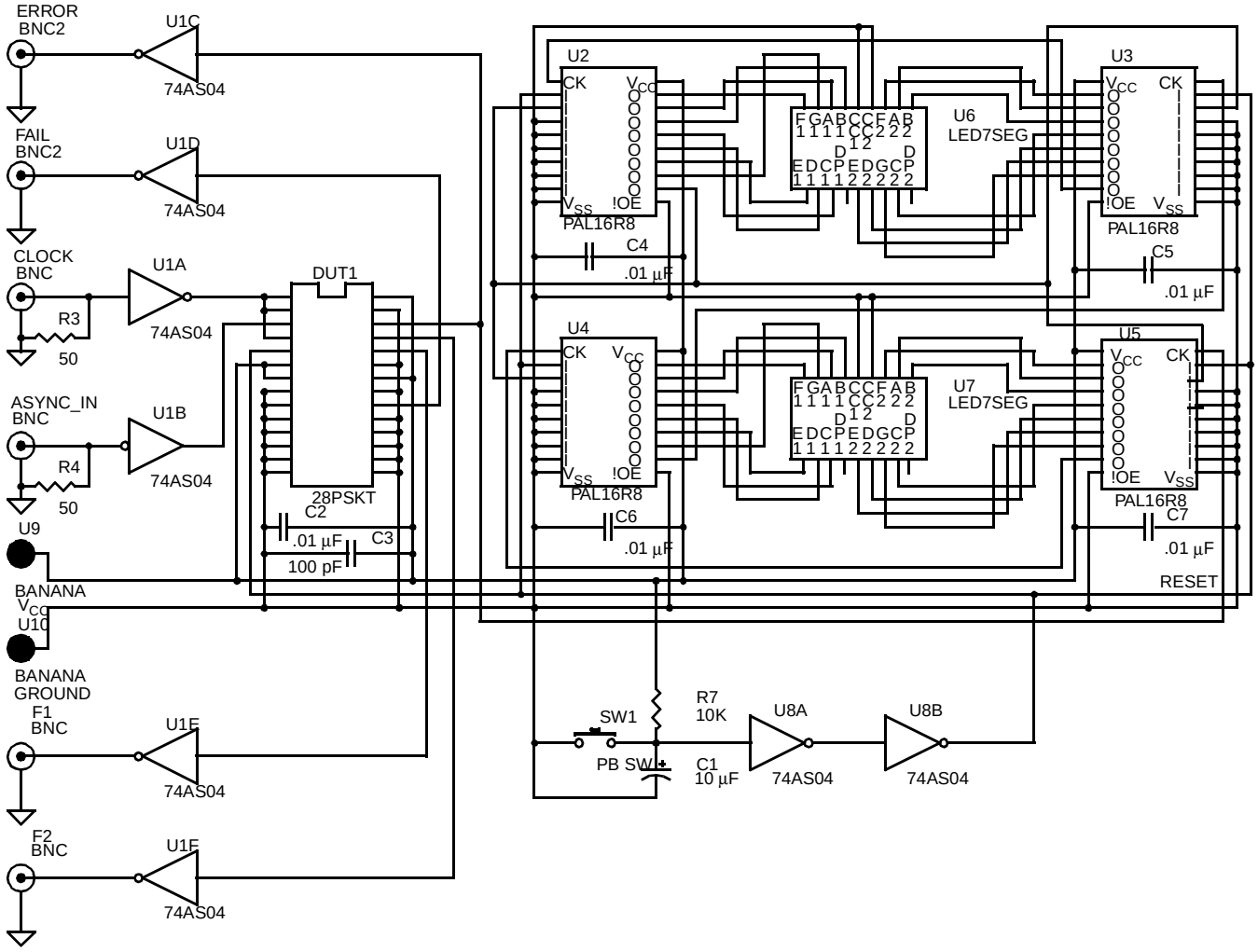


Figure 21. Metastability Test Board Schematic

All inputs and outputs connect with BNC connectors located around the board. The clock line, which is terminated with a 50Ω resistor to match the coax input impedance, is buffered with a 74AS04 and isolated from other signals by a ground trace. The input line is also terminated with a 50Ω resistor and buffered with a 74AS04. Four PLDs drive a four-digit LED display that counts metastability occurrences.

After going LOW in response to a metastable event, the $\overline{\text{ERROR}}$ signal automatically transitions HIGH again at the next system clock. This LOW-to-HIGH pulse produces a clock to the input of the first PLD, which in turn increments the display of metastable events. When a digit reaches 9, the next occurrence of metastability generates a cascade signal to the next higher digit.

In this way, the test board can record a maximum of 9,999 metastable events. If a metastable event is received at 9,999, all LEDs switch to E, indicating that an overflow condition occurred. A reset button resets all counters and initializes the DUT.

Test Set-Up

Figure 22 shows a block diagram of the test set-up used for metastability testing. Two independent pulse generators (Hewlett-Packard 8082As) produce the CLOCK and the ASYNC_IN signal to the test board. A Tektronix DAS9200 logic analyzer records metastable events. A 2465 CTS digital oscilloscope with frequency counter accurately determines the DUT's maximum operating frequency and the ASYNC_IN and CLOCK frequencies.

Test Procedure

Cypress has tested all its 20-, 24-, and 28-pin PLDs. The fastest speed grades of each device type were tested because these devices have the best metastable resolution time and thus make the best synchronizers. Several parts from each device type were tested to ensure an average metastability characteristic for that product. Where possible, parts from different date codes were selected to eliminate variations among different wafer lots.

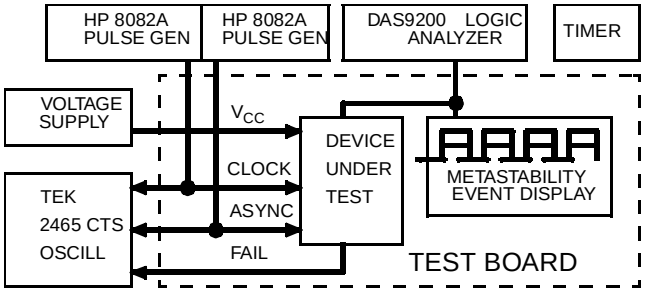


Figure 22. Metastability Test Set-Up

Testing for a specific device starts by creating the high-level description written in VHDL to be used with the *Warp2* VHDL Compiler. *Figure 23* lists the behavioral description used for generating a JEDEC file. All devices were programmed using JEDEC files generated by *Warp2*.

```
package test is
  component metastability port (
    clock, async_in, reset : in bit;
      fail, perror, f1, f2 : out bit);
  end component;
end test;

entity metastability is port (
  clock, async_in, reset : in bit;
    fail, perror, f1, f2 : out bit);
end metastability;

use work.bv_math.all

architecture fsm of metastability is

signal  sync  : bit;
signal  tsync : bit;
signal  t1,t2 : bit;
signal  f1_tmp, f2_tmp : bit;
signal  error_tmp : bit;
signal  fail_tmp : bit;

begin
  proc1: process begin

    wait until clock = '1';
    sync <= async_in;

  end process;
```

Figure 23. *Warp2* VDHL Behavorial Description for Metastability Testing

```
  proc2: process begin

    wait until clock = '1';
    f1_tmp <= sync;
    f2_tmp <= inv(sync);

  end process;

  proc3: process begin

    wait until clock = '1';
    error_tmp <= ((((inv(reset) and f1_tmp) and inv(f2_tmp))
      or ((inv(reset) and inv(f1_tmp)) and f2_tmp))
      or (reset and inv(error_tmp)));

  end process;

  proc4: process begin

    wait until clock = '1';
    if (tsync = '1') then
      tsync <= '0';
    else
      tsync <= '1';
    end if;

  end process;

  proc5: process begin

    wait until clock = '1';
    t1 <= tsync;
    t2 <= inv(tsync);

  end process;

  proc6: process begin

    wait until clock = '1' ;
    fail_tmp <= (t1 xor t2);

  end process;

fail <= inv(fail_tmp);
perror <= inv(error_tmp);
f1 <= inv(f1_tmp);
f2 <= inv(f2_tmp);

end fsm;
```

Figure 23. *Warp2* VDHL Behavorial Description for Metastability Testing (continued)

Each part is programmed, then tested for its maximum operating frequency, f_{max}. By attaching the FAIL output to the oscilloscope and observing the clock frequency at which the device started to malfunction (FAIL going LOW periodically), the maximum operating frequency for that part is determined. f_{max} indicates the maximum rate at which metastability measurements can be taken with accurate results. Above this frequency, metastable events are indistinguishable from errors caused by exceeding f_{max}.

To determine each device's metastability characteristics, measurements are taken of the number of metastable events that occurred in a given time interval for several different clock and data frequencies.

Equation 13 can be used to describe the graph of the metastability characteristics of the device:

$$\ln(MTBF) = \frac{t_r}{t_{sw}} - \ln(f_c f_c W)$$

The slope of the line, t_{sw}, can be determined only by forcing the Y intercept of the graph ($\ln(f_c f_d W)$) to a constant value when using *Equation 14*:

$$t_{sw} = \frac{t_{r1} - t_{r2}}{\ln(MTBF_1) - \ln(MTBF_2)}$$

Note that t_{sw} is a constant, device-specific parameter.

Because W is also a constant, device-specific parameter, it is only necessary to hold the product $f_c f_d$ constant to make $\ln(f_c f_d W)$ constant. The independent variable t_r is varied by changing f_c to produce changes in the dependent variable ln(MTBF). Decreasing the frequency f_c from its f_{max} value increases the metastable resolution time, t_r, and decreases the probability that a metastable event will last longer than t_r.

As f_c is decreased below a certain limit, the MTBF becomes too large to measure accurately. A metastable event occurring every minute is chosen as the upper limit for MTBF measurements. The range of clock rates for metastability testing is then between f_{max} and the metastable-event-per-minute clock rate. Between these two rates, a selected frequency constant ($f_c f_d$) ensures that no point in this range has a clock frequency less than twice the data frequency. This is because a data signal that transitions more than once per clock period cannot be effectively sampled.

After determining this constant, data is taken from several test points within the test range by varying f_c and f_d. The data at each test point is averaged among all test devices, and the equation for the line through these points is determined using a linear regression analysis. The correlation between the line and the data points verifies that the metastability equation accurately describes the test data. From the calculated results, the constants W and t_{sw} are extracted.

Test Results

Table 1 and the Appendix list the results of the metastability analysis of Cypress PLDs. *Table 1* also lists the maximum data book operating frequency, f_{max}; the metastability equation constants, W and t_{sw}; the metastability resolve time, t_r, required for a 10-year MTBF; and the process for that part.

You can use this data to determine the maximum metastability resolve time (t_r) that you must use in a system to yield a given degree of reliability. The graphs and constants (W and t_{sw}) can be used with any speed grade of the device, but it is suggested that the fastest speed grade of the specific PLD be used for optimum synchronizer performance. These graphs indicate the time (t_r) and the device's minimum clock period that must be used to produce a desired degree of reliability.

For example, to determine the operating parameters of the Cypress PALC22V10-20 from *Table 1* when using the device as a synchronizer, determine the desired MTBF. With a 10-yr (315×10^6s) MTBF, for instance, a synchronization failure will occur once every 10 years on the average. The maximum operating frequency (f_{max}) from the PALC22V10's data sheet is 41.6 MHz. From this information, you can determine the minimum time (t_r) beyond the device's minimum operating period that must be added for metastability resolution:

$$MTBF = \frac{e^{\frac{t_r}{t_{sw}}}}{f_c f_d W}$$

$$t_r = t_{sw}(\ln(MTBF) + \ln(f_c f_d W))$$

$$t_r = (0.190 \times 10^{-9}s)[\ln(315 \times 10^6 s) + \ln(41.6 \times 10^6 \times 41.6 \times 10^6 \times 0.125 \times 10^{-12})]$$

$$= 4.73 \text{ ns}$$

This analysis assumes that the clock, f_c, operates at f_{max} (41.6 MHz) and that the average asynchronous data frequency is no more than half the clock frequency. The latter condition ensures effective data sampling by the synchronizer. f_d, as explained in the Statistical Analysis of Metastability section represents the rate at which the data changes state. f_d is twice the average frequency of the asynchronous data input because, during any given asynchronous data period, the asynchronous data changes state twice: once from LOW to HIGH and again from HIGH to LOW. Because either of these state changes can cause a metastable event, f_d must be set to twice the average asynchronous data frequency when determining the worst-case MTBF.

Due to the real-world uncertainty in factors such as trace delays and the skew in clock generators, 5 ns is used instead of 4.73 ns for t_r. The synchronizer's maximum operating frequency, f_c, in this system is then

$$f_c = \frac{1}{t_s + t_{cf} + t_r} = \frac{1}{10ns + 12ns + 5ns} = 37.0MHz$$

The effective MTBF using these new values for t_r and f_c is

$$MTBF = \frac{e^{\frac{5 \times 10^{-9}s}{0.190 \times 10^{-9}s}}}{37.0 \times 10^6 s^{-1} \times 37.0 \times 10^6 s^{-1} \times 0.125 \times 10^{-12}s}$$

$$= 1.57 \times 10^9 = 49.7 \text{ years}$$

Table 1.Metastability Characteristics of Cypress PLDs.

Device	f_{max} (MHz)	W (s)	t_{sw} (s)	t_r for 10-yr MTBF (ns)
PALC16R8-25	28.5	9.503E–12	0.515E–9	14.68
PLDC20G10-20	41.6	3.730E–12	0.173E–9	4.91
PALC20RA10-15	33.3	2.860E–12	0.216E–9	5.87
PALCE22V10-7	100	32.35E–12	0.347E–9	10.56
PALC22V10B-15	50.0	55.76E–12	0.261E–9	8.19
PALC22V10-20	41.6	0.125E–12	0.190E–9	4.73
CY7C331-20	31.2	0.298E–9	0.184E–9	5.91
CY7C335-100	58.8	0.288E–12	0.189E–9	4.95
CY7C344-20	41.6	0.966E–9	0.223E–9	7.55

Another example focuses on the CY7C330-50 used as a synchronizer in a system whose output registers are clocked at an f_c of 35.7 MHz, and the data has an average frequency of 10 MHz. The MTBF for this device used as a synchronizer is calculated by first determining the metastable resolution time, t_r, allowed for synchronization. The maximum operating frequency of the part is specified in Cypress's *Data Book* as

$$f_{max} = \frac{1}{t_{co} + t_{is}}$$

where t_{co} in this case specifies the clock-to-feedback delay, and t_S specifies the set-up time of the output registers. t_r is calculated with the equation:

$$t_r = \frac{1}{f_c} - \frac{1}{f_{max}} = \frac{1}{35.7MHz} - \frac{1}{50.0MHz} = 8 \text{ ns}$$

With this result, the MTBF is

$$MTBF = \frac{e^{\frac{8\times10^{-9}s}{0.290\times10^{-9}s}}}{35.7\times10^{6}s^{-1}\times20.0\times10^{6}s^{-1}\times1.02\times10^{-12}s}$$

$$= 1.31\times10^{9} = 41.6 \textit{ years}$$

This equation uses the same values for W and t_{sw} with this 50-MHz device as with the 66-MHz device listed in *Table 1*. As stated previously, the constants listed in*Table 1* are valid for all speed grades of a specific device. Also note that the 10-MHz average data frequency is doubled to produce the frequency of data transitions, f_d.

The last example illustrates how to use a Cypress PALC22V10C-10 as a synchronizer. For a 10-year MTBF, assuming the maximum f_c from Cypress's *Data Book* and f_d, the required t_r is

$$t_r = (0.547\times10^{-9}s)[\ln(315\times10^{6}s) + \ln(90.9\times10^{6}\times90.9\times10^{6}\times8.08\times10^{-15})]$$

$$= 13.0 \textit{ ns}$$

Using this result, the synchronizer's maximum operating frequency is reduced from 90.9 MHz to

$$f_c = \frac{1}{\frac{1}{f_{max}} + t_r} = \frac{1}{\frac{1}{90.9MHz} + 13.0ns} = 41.6MHz$$

Two-Stage Synchronization

As explained earlier, you can use a second register in series to perform two-stage synchronization (*Figure 4*). This is accomplished by feeding the output of the first synchronization register to the input of the second synchronization register. In PLDs, this method is common because the first synchronization stage can synchronize the asynchronous input signal, and the second synchronization stage can perform a Boolean function on a combination of the input and output signals. Boolean functions can be performed at either stage; the metastability characteristics listed in *Table 1* apply to PLD registers' asynchronous inputs that are used directly as well as asynchronous inputs used as a Boolean combination of existing inputs and outputs.

When implementing a two-stage synchronizer in a PLD, the probability that a synchronizer is metastable after the second stage of synchronization is the square of the probability that a synchronizer is metastable after the first stage of synchronization. The MTBF equation is

$$MTBF = \left(\frac{e^{\frac{t_r}{t_{sw}}}}{f_c f_d W}\right)^2$$

From this result, the equation for t_r becomes

$$t_r = \frac{t_{sw}(\ln(MTBF) + 2\times\ln(f_c f_d W))}{2}$$

Using this result for a two-stage synchronizer in a Cypress PALC22V10C, the t_r for a 10-year MTBF is reduced from 13.0 ns to

$$t_r = (0.5)(0.547\times10^{-9}s)[\ln(315\times10^{6}s) + \ln(90.9\times10^{6}\times90.9\times10^{6}\times8.08\times10^{-15})]$$

$$= 7.65 \textit{ ns}$$

The maximum f_c increases from 41.6 MHz to

$$f_c = \frac{1}{\frac{1}{f_{max}} + t_r} = \frac{1}{\frac{1}{90.9MHz} + 7.65ns} = 53.6MHz$$

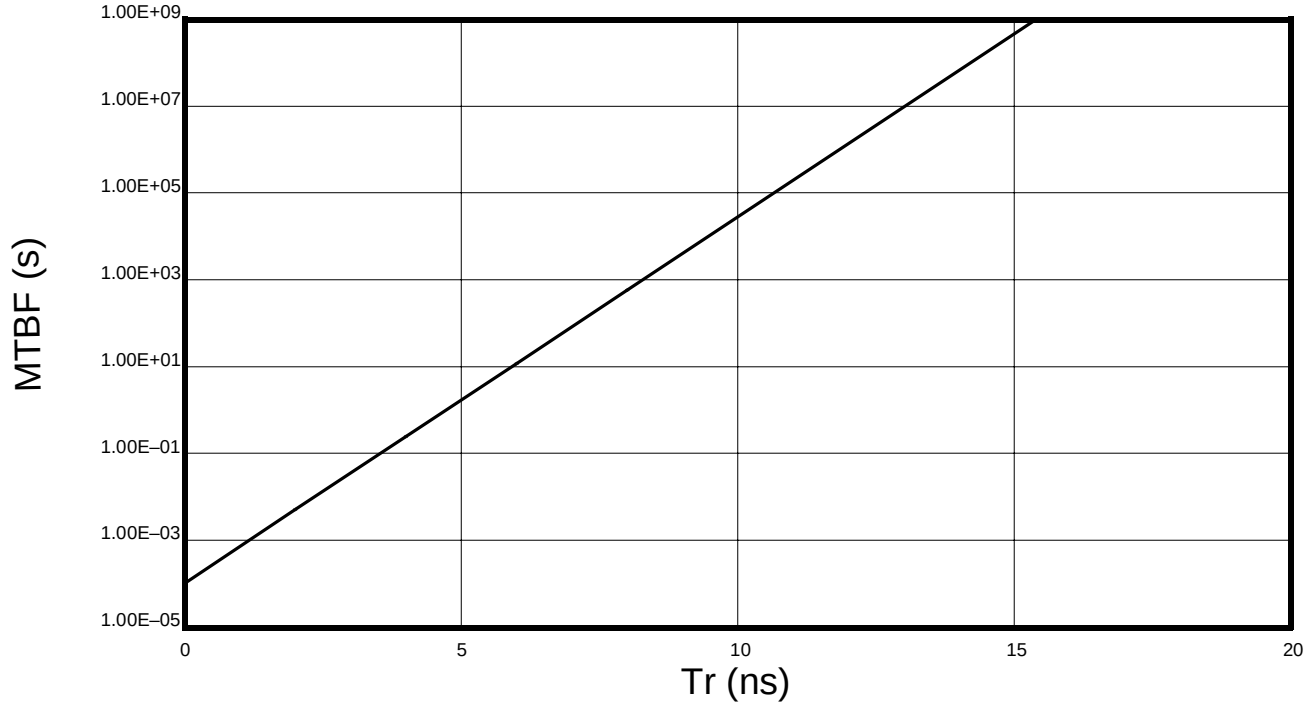
This example shows that if the cycle of latency caused by the additional synchronization stage is acceptable, you can dramatically increase the synchronizer's maximum operating frequency.

References

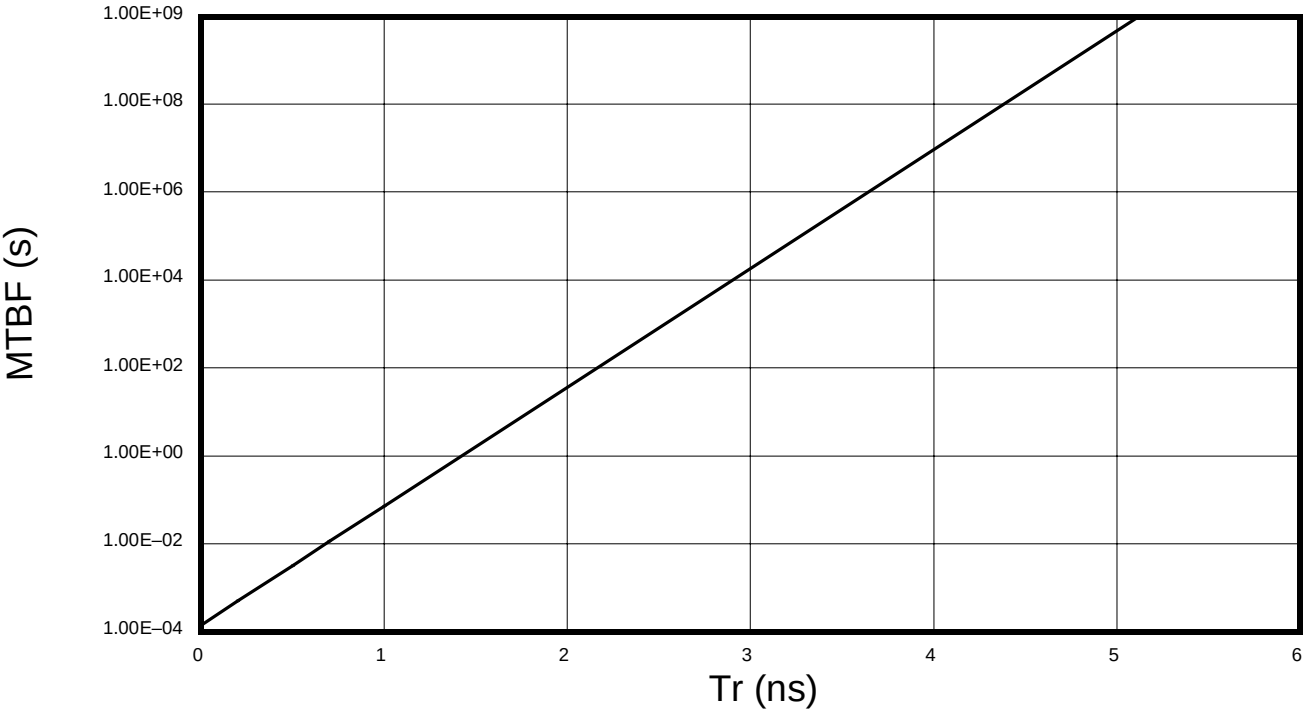
1. Lubkin, S., (Electronic Computer Corp.), “Asynchronous Signals in Digital Computers,” *Mathematical Tables and Other Aids to Computation*, Vol. 6, No. 40, October 1952, pp. 238-241.
2. Nootbaar, Keith, (Applied Microcircuits Corp.), “Design, Testing, and Application of a Metastable-Hardened Flip-Flop,” WESCON 87 (San Francisco, CA, Nov. 17-19, 1987), Electronic Conventions Management, Los Angeles, CA 90045.
3. Stoll, Peter A., “How to Avoid Synchronization Problems,” *VLSI Design*, November/December 1982, pp. 56-59.
4. Chapiro, Daniel M., *Globally-Asynchronous Locally-Synchronous Systems*, Stanford University, Department of Computer Science Report No. STAN-CS-84-1026, October 1984.
5. Horstmann, Jens U., Eichel, Hans W., Coates, Robert L., “Metastability Behavior of CMOS ASCI Flip-Flops in Theory and Test,” *IEEE Journal of Solid-State Circuits*, Vol. 24, No. 1, February 1989, pp. 146-157.
6. Wormald, E.G., “A Note on Synchronizer or Interlock Maloperation,” *Professional Program Session Record 16*, WESCON 87, November 17-19, 1987, Electronic Conventions Management, Los Angeles, CA 90045.
7. Pechouchek, Miroslav, “Anomalous Response Times of Input Synchronizers,” *IEEE Trans. Computers*, Vol. C-25, No. 2, February 1976, pp. 133-139.
8. Chaney, T. J., “Comments on ‘A Note on Synchronizer or Interlock Maloperation,” *IEEE Trans. Computing*, Vol. C-28, No. 10, Oct. 1979, pp. 802-804.
9. Couranz, George R., Wann, Donald F., “Theoretical and Experimental Behavior of Synchronizers Operating in the Metastable Region,” *IEEE Trans. Computers*, Vol. C-24, No. 6, June 1975, pp. 604-616.
10. Veendrick, Harry J.M., “The Behavior of Flip-Flop Used as Synchronizers and Prediction of Their Failure Rate,” *IEEE Journal of Solid-State Circuits*, Vol. SC-15, No. 2., April 1980, pp. 169-176.
11. Kacprzak, Tomasz, Albicki, Alexander, “Analysis of Metastable Operation in RS CMOS Flip-Flops,” *IEEE Journal of Solid-State Circuits*, Vol. SC-22, No. 1, February 1987, pp. 57-64.
12. Flannagan, Stephen T., “Synchronization Reliability in CMOS Technology,” *IEEE Journal of Solid-State Circuits*, Vol. SC-20, No. 4, Aug 1985, pp. 880-882.
13. Wakerly, John F., *A Designers Guide to Synchronizers and Metastability*, Center for Reliable Computing Technical Report, CSL TN #88-341, February, 1988 Computer Systems Laboratory, Departments of Electrical Engineering and Computer Science, Stanford University, Stanford, CA.
14. Freeman, Gregory G., Liu, Dick L., Wooley, Bruce, and McClusky, Edward J., *Two CMOS Metastability Sensors*, CSL TN# 86-293, June 1986, Computer Systems Laboratory, Electrical Engineering and Computer Science Departments, Stanford University, Stanford, CA.
15. Rubin, Kim, “Metastability Testing in PALs,” WESCON 87 (San Francisco, CA, Nov. 17-19, 1987), Electronic Conventions Management, Los Angeles, CA 90045.

Appendix A. Metastability Graphs of Cypress Devices

Cypress PALC16R8-25

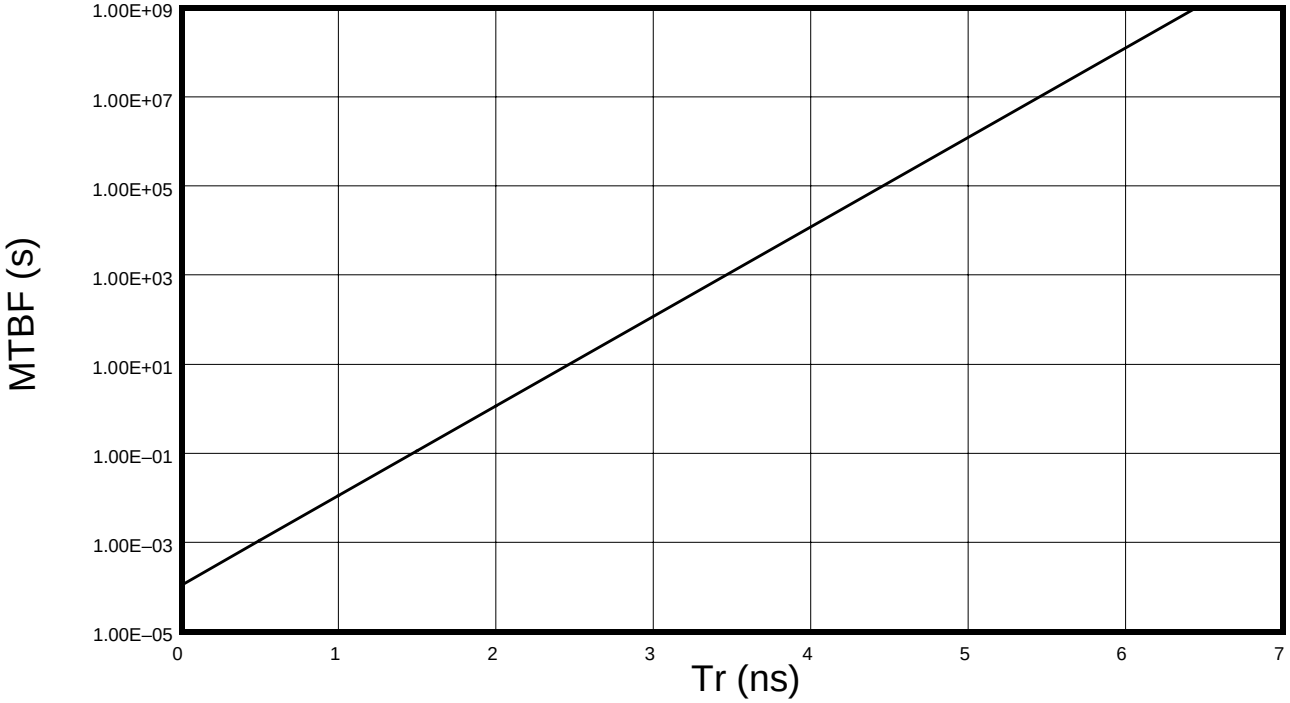


Cypress PLDC20G10-20

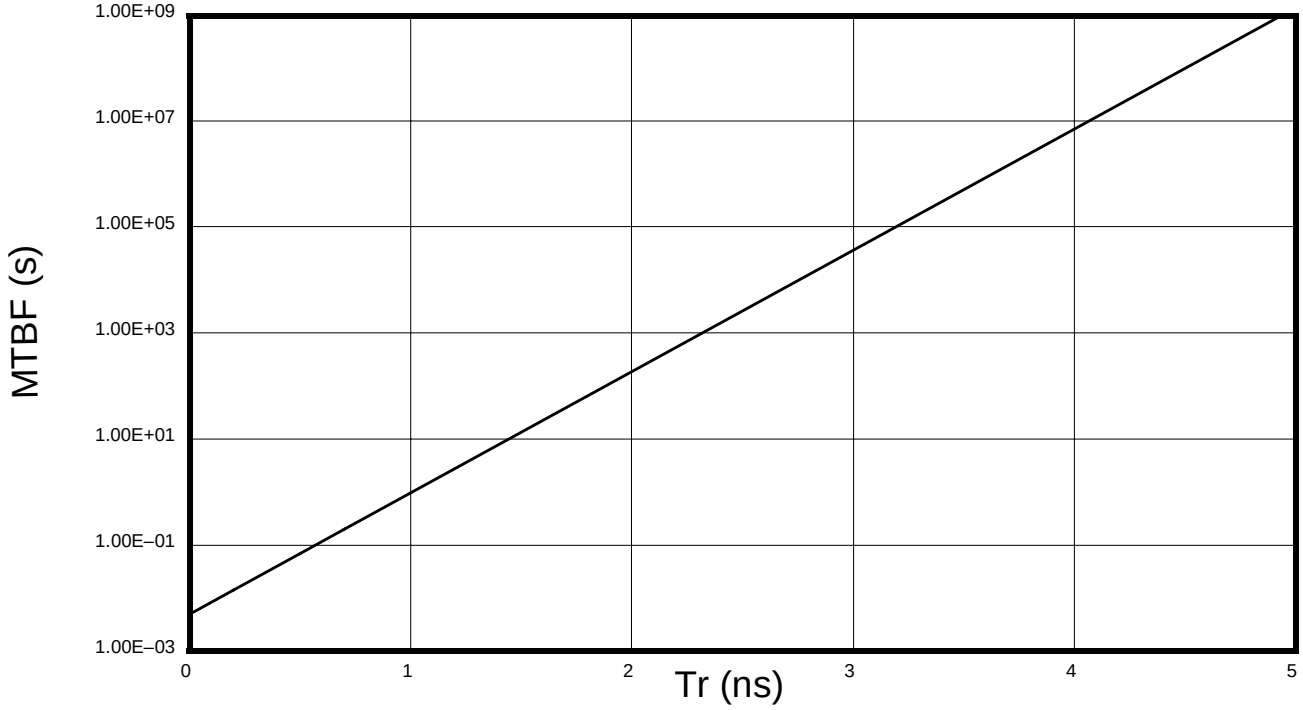


Appendix A. Metastability Graphs of Cypress Devices (continued)

Cypress PALC20RA10-15

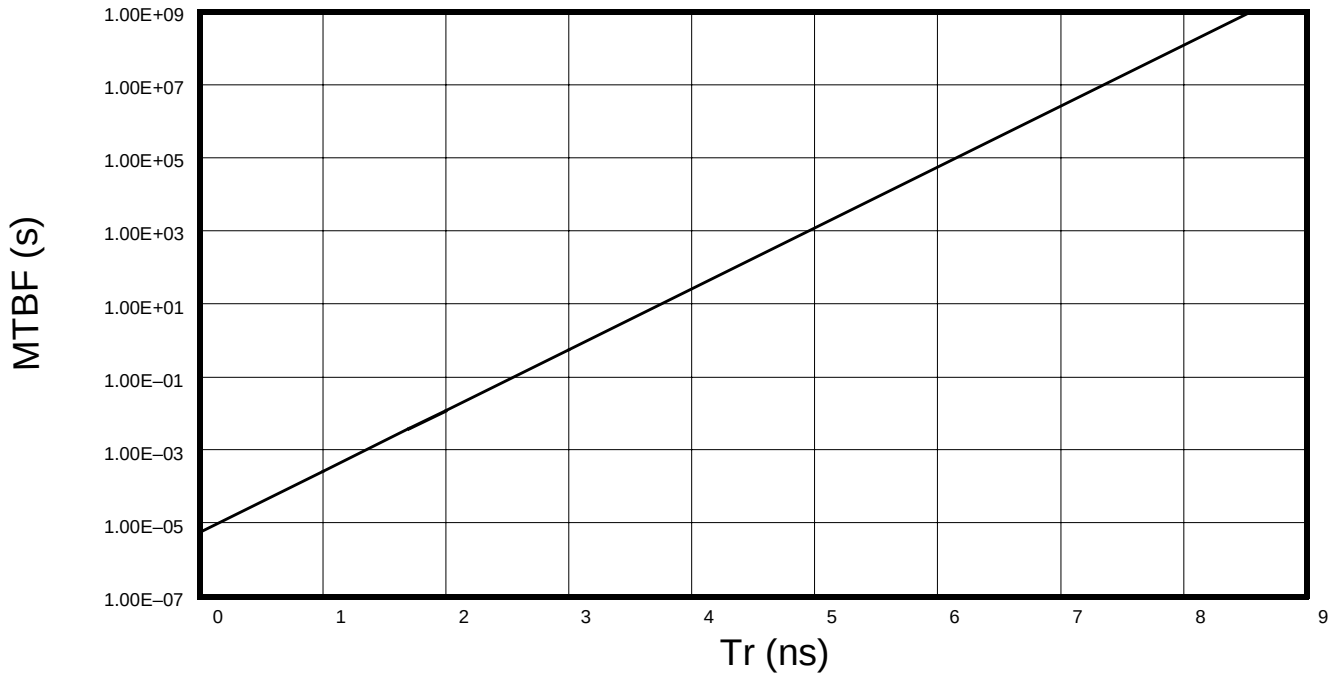


Cypress PALC22V10-20

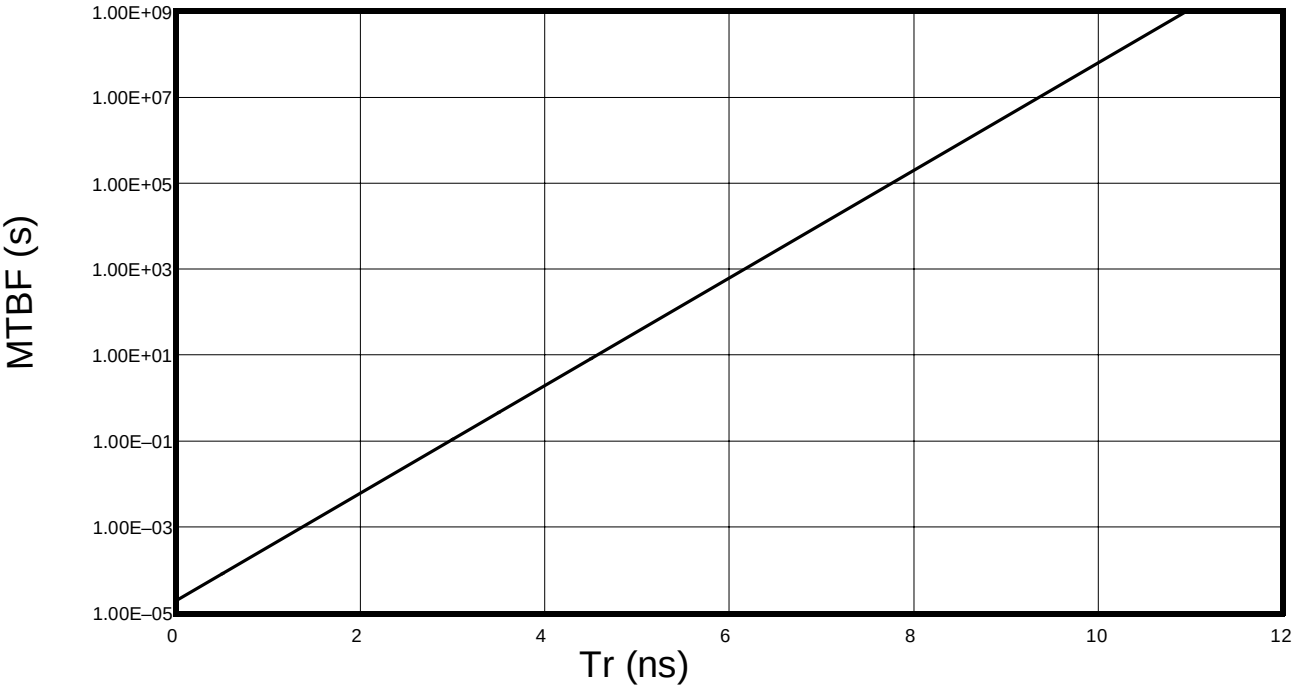


Appendix A. Metastability Graphs of Cypress Devices (continued)

Cypress PALC22V10B-15

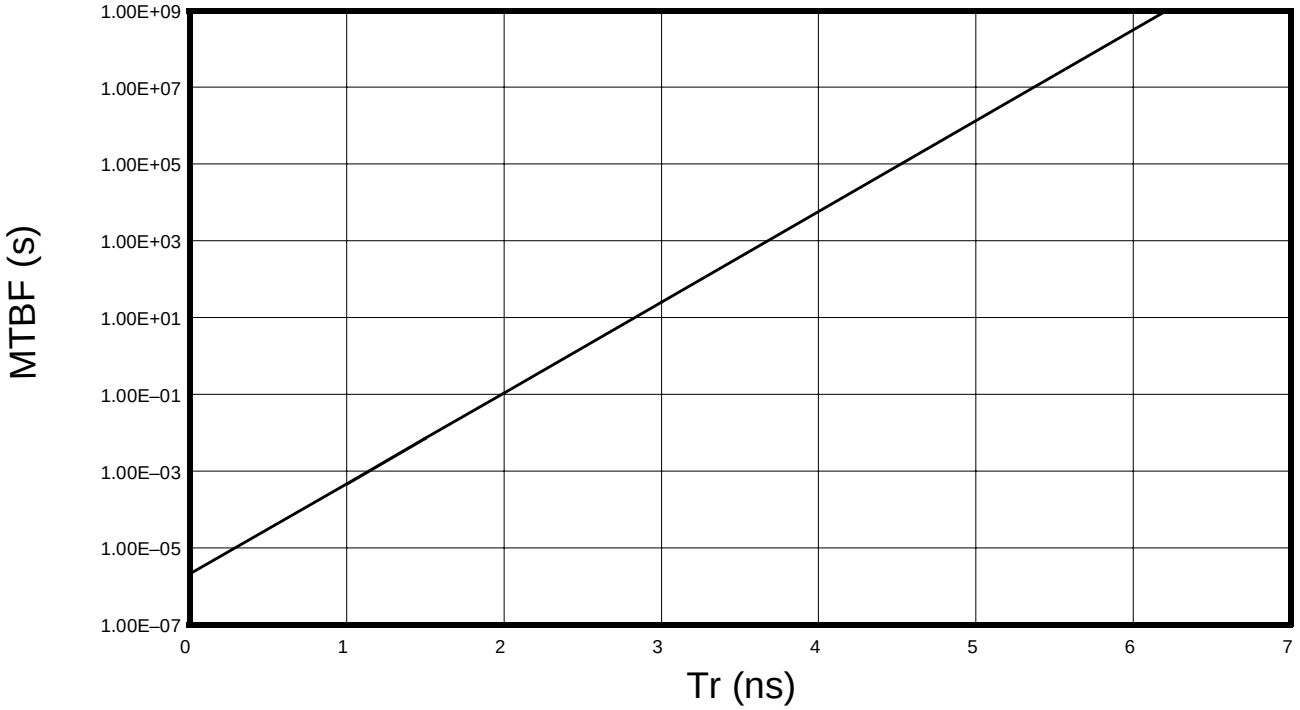


Cypress PALC22V10D-7

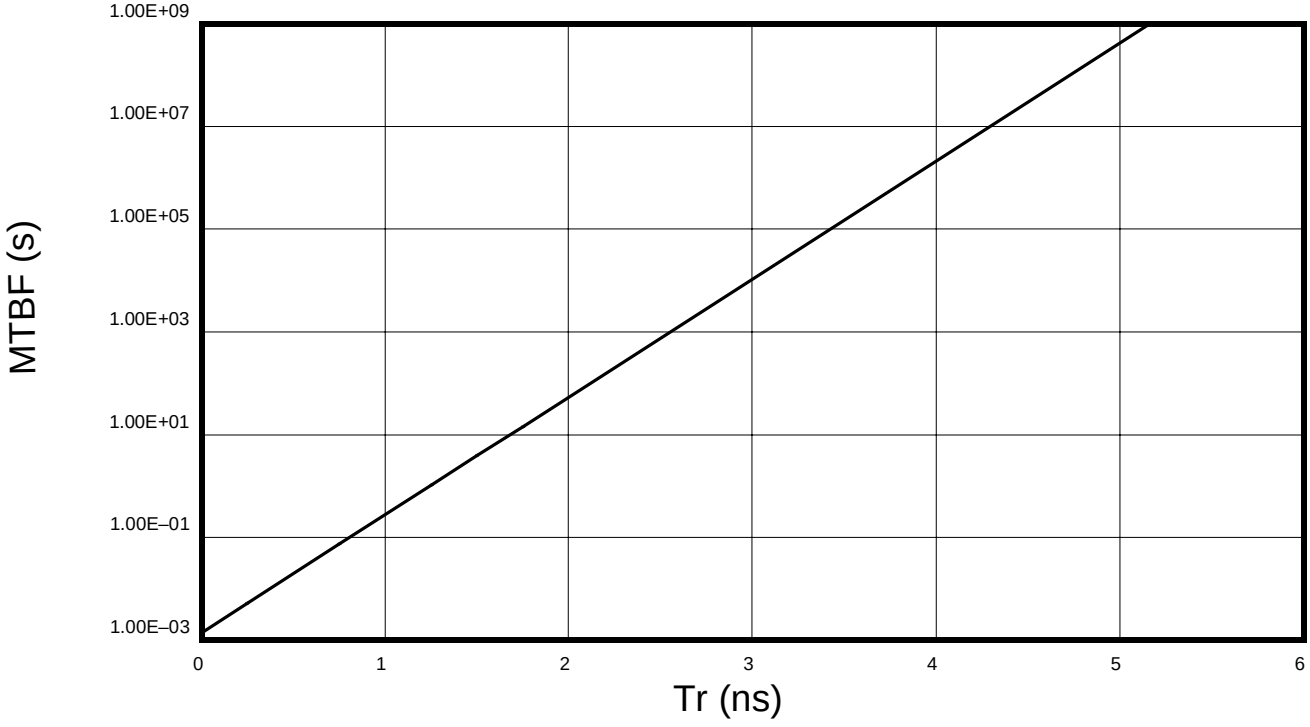


Appendix A. Metastability Graphs of Cypress Devices (continued)

Cypress CY7C331-20



Cypress CY7C335-100



Appendix A. Metastability Graphs of Cypress Devices (continued)