Best Design Practices for HardCopy Devices

AN-658-1.1 Application Note

This application note describes several good design practices that are commonly missed during the FPGA design phase when migrating a design to a HardCopy® ASIC device. The information in this application note will help you avoid these common design pitfalls and ensure that your system continues to perform well with a HardCopy device.

Customers often use the FPGA flow to first verify their design on an FPGA before migrating the design to a HardCopy ASIC. Even if good design practices are not followed, the FPGA prototype system might still pass all the testing. However, you could see an unexpected outcome after replacing the FPGA device with the HardCopy ASIC device. This situation usually results from subtle architectural differences between HardCopy and FPGA devices that were not considered during the design phase. Employing the design practices suggested in this application note could help you avoid an expensive respin of the HardCopy ASIC device.

By understanding some of the architectural features and how they affect the outcomes of the HardCopy and FPGA devices, you can expect your system to work just like the FPGA when you replace it with the HardCopy device.

This application note describes the following design considerations when migrating from an FPGA to a HardCopy III or HardCopy IV ASIC:

The "Design Checklist" on page 15 summarizes the checklist items presented in this document. You can use this checklist to ensure that you have reviewed all the guidelines before migrating your design to a HardCopy device.

This application note assumes you have a basic understanding of the above topics. For more information, refer to the *HardCopy III Device Handbook* and the *HardCopy IV Device Handbook*.

101 Innovation Drive
San Jose, CA 95134
www.altera.com

Power Sequence Design Considerations

This section describes the power-on reset (POR) requirements and key considerations for migration to a HardCopy device.

This section does not discuss hot-socketing. For more details, refer to the hot-socketing chapters in the *Stratix III*, *Stratix IV*, *HardCopy III*, and *HardCopy IV* device handbooks.

Power Supply Ramp-Up Behavior

☐ 1. All power supplies should ramp up within the specified t_{RAMP} time in a monotonic manner (without a plateau) to avoid a brown-out condition.

As shown in Figure 1, the t_{RAMP} time is the time the power supply takes to ramp up from 0 to its full voltage level and is bounded by the minimum and maximum values.

For the recommended t_{RAMP} operating conditions, refer to the datasheets in the *Stratix III*, *Stratix IV*, *HardCopy III*, and *HardCopy IV* device handbooks.

If a power supply is not ramped up monotonically, the power supply could fall below the upper trip point of the POR, causing the device to reset. This event is called a brown-out condition.

Figure 1. Ramp-Up Behavior of a Power Supply

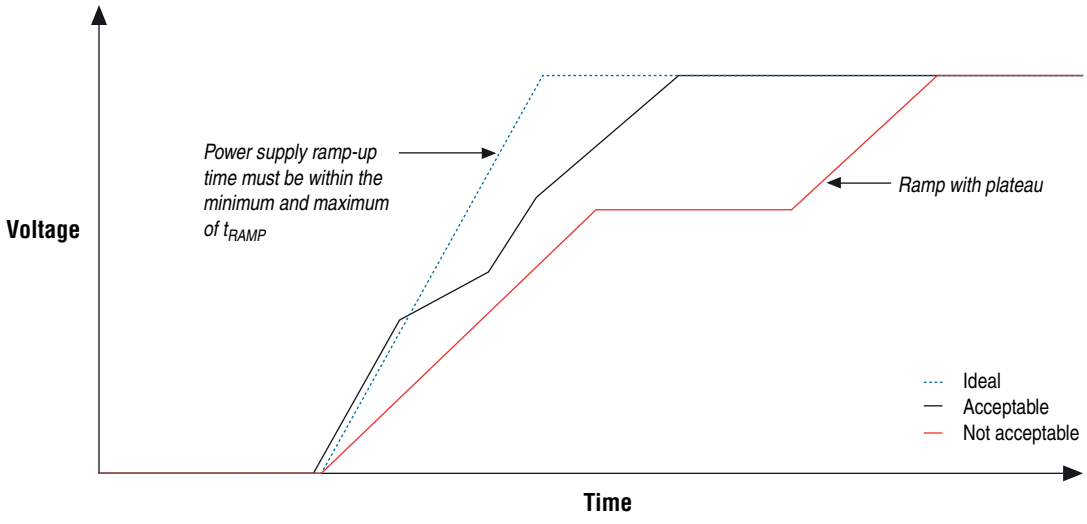
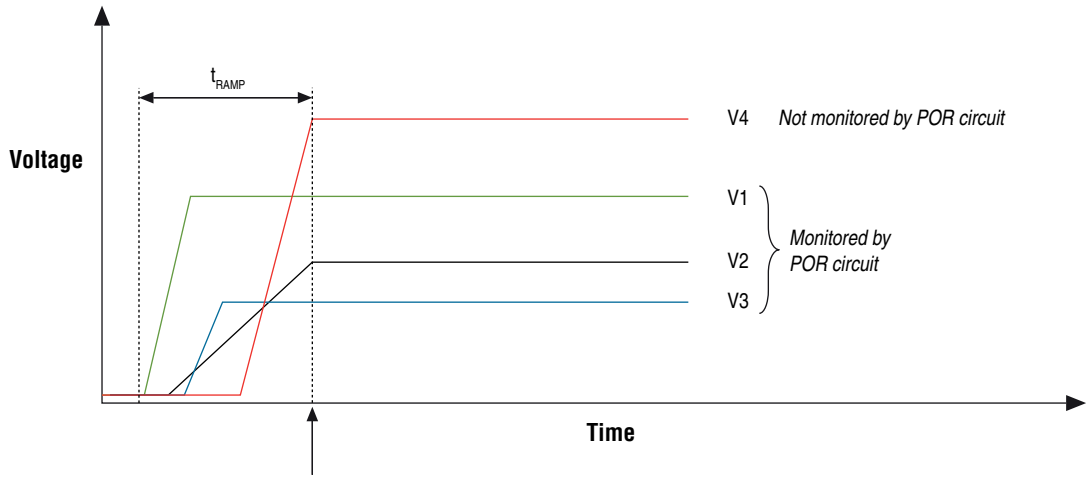


Figure 2 shows that all the essential power supplies must be fully ramped up before the device enters user mode.

Figure 2. Diagram Showing Power-Up Event of Some Power Supplies



In both FPGA and HardCopy devices, the POR circuit monitors the power supply voltage levels that keep the I/O pins tri-stated and the system in reset until the device is in user mode. While designing your power management, remember that the POR circuit does not monitor all of the power supplies. Check the device errata sheets for any change to the power sequence requirements.

For the list of power supplies monitored by the POR circuit in a particular device family, refer to the *Stratix III*, *Stratix IV*, *HardCopy III*, and *HardCopy IV* device handbooks.

When the last power supply monitored by the POR circuit has reached its trip level, the POR circuit becomes active and sends out an internal POR reset signal in the chip. The duration of this reset is determined by the PORSEL pin setting.

POR Delay

The PORSEL pin selects a long or short POR delay duration. For any PORSEL setting, the POR delay of a device is a range with the minimum and maximum values specified in the device datasheet.

☐ 2. When you migrate from an FPGA device to a HardCopy ASIC, make sure your system power sequence management takes into consideration the minimum and maximum delay boundary values so that all devices on the system board are ready for handshaking activities.

For example, in applications that use PCI Express® (PCIe®), select the shorter POR delay duration so the device is ready for the PCIe activities. Refer to the device datasheet for the specifications.

Configuration Time

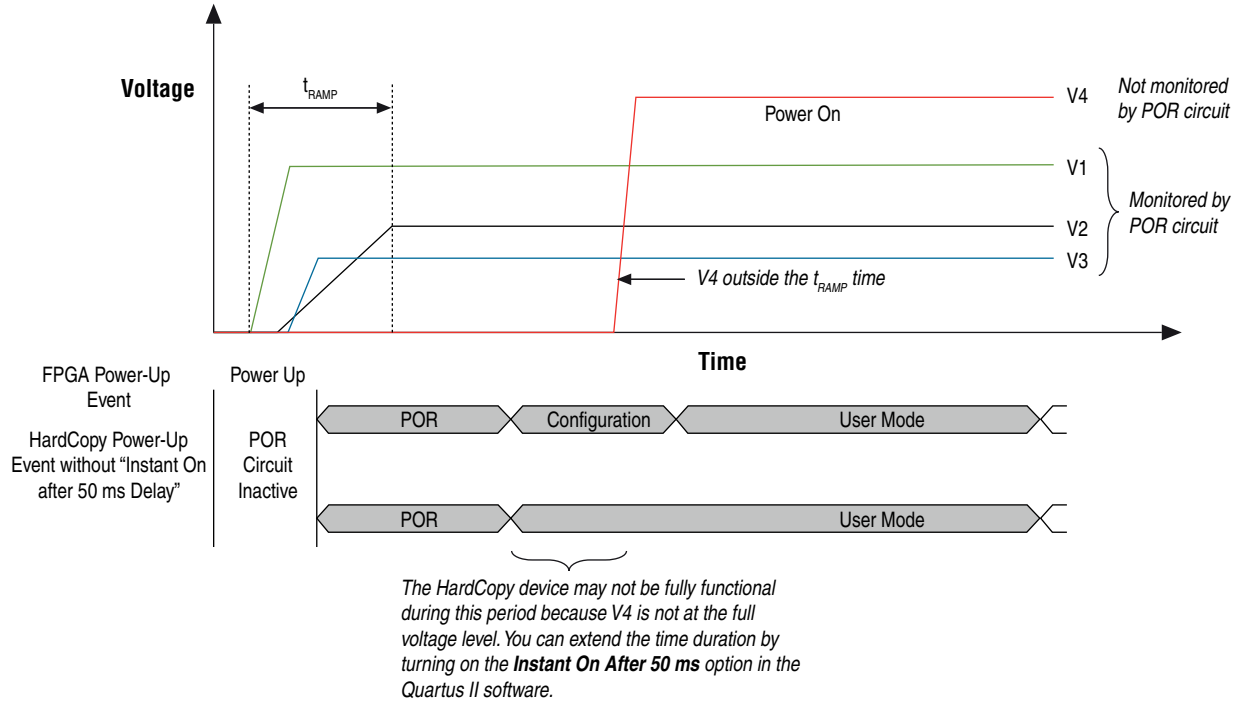
For FPGAs, an additional period of time is required for configuration after the POR is released and before entering user mode. HardCopy devices, however, do not require configuration. Depending on the power-up mode selected, the device can almost instantly enter user mode after power up. Therefore, a HardCopy device enters user mode sooner than an FPGA after the POR event.

☐ 3. You must take this time difference in the power-up sequence into consideration during the design phase. All the power supplies must be at full rails before entering user mode in the HardCopy device. If additional delay time is needed, you can implement the **Instant On After 50 ms Mode** option.

Figure 3 shows the different power-up events for the FPGA and HardCopy devices. V1, V2, and V3 represent hypothetical power supplies that are monitored by the POR circuit, while V4 is not monitored. Refer to the device datasheet in the appropriate handbook for the list of power supplies that are monitored by the POR circuit.

In Figure 3, V4 is being powered during the FPGA configuration period, but reaches its full voltage level before the end of configuration. However, when the FPGA is replaced with the HardCopy device, the HardCopy device will enter into user mode before V4 reaches its full voltage level and may produce erroneous functional results.

Figure 3. Conceptual Timing Diagram Showing Power-Up Events



One example of a power-up issue in a HardCopy device is when V4 is the V_{CCIO} supply and is used for on-chip termination (OCT) calibration. This issue does not appear in the FPGA because the V_{CCIO} is able to ramp up to its full rail during the configuration period before the FPGA enters user mode.

However, the HardCopy device may encounter a power-up issue because the HardCopy device enters user mode before the V_{CCIO} reaches its full rail and the I/Os are not ready for calibration.

To avoid this situation, all power supplies should reach their full rails within the t_{RAMP} duration prior to the end of POR, as specified in the datasheets.

If your HardCopy device has an issue due to a delay in the power supply ramping up slower than required and you are unable to change the power sequence, try pulsing the `nCONFIG` pin to reset the HardCopy device. Pulsing the `nCONFIG` pin is similar to resetting the HardCopy device while all the power supplies are at their full voltage rails.

I/O Contentions

Not meeting power supply requirements can result in I/O contentions. This situation occurs when certain I/Os, such as bidirectional I/Os, are not ready to drive or receive electrical signals because their power supplies are not ready.

TriMatrix Memories

This section describes the TriMatrix embedded memory blocks (MLAB, M9K, and M144K).

Read Address Register Timing Requirements

☐ 4. For a valid read or write operation to the M9K and M144K memories, the memory block address registers must meet the setup and hold time requirements to avoid corrupting the memory contents.

If the timing of the address register to an M9K or M144K is violated, the address register output can become metastable and momentarily stay at an unknown state between logic 1 and 0. This condition could trigger the address decoder inside the M9K or M144K to turn on two memory locations, resulting in charge sharing and corrupting the memory contents.

This condition applies to both read and write operations. Although the MLAB has a different architecture that prevents this from happening, take precautions to avoid any timing violations. Use successive synchronization flipflop registers to minimize metastability issues if you are in a clock domain crossing situation.

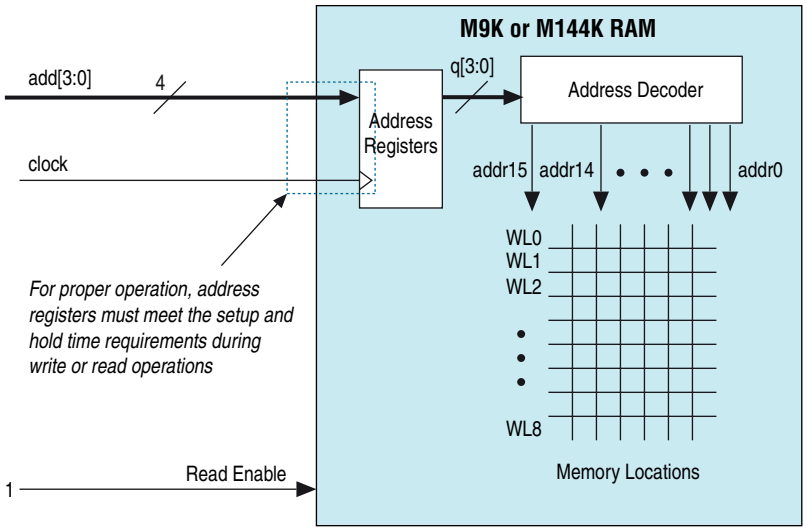
A Read Example

Figure 4 shows how a read operation with a timing violation in the read address register corrupts the memory content.

In this example, the RAM has a 4-bit address bus. A read operation is performed at address location 4'b1000 of a memory block. Assume that only the `add[3]` register (the MSB) of its 4-bit address line does not meet the setup or hold times and enters a metastable condition. The other addresses `add[2:0]` meet the timing requirements.

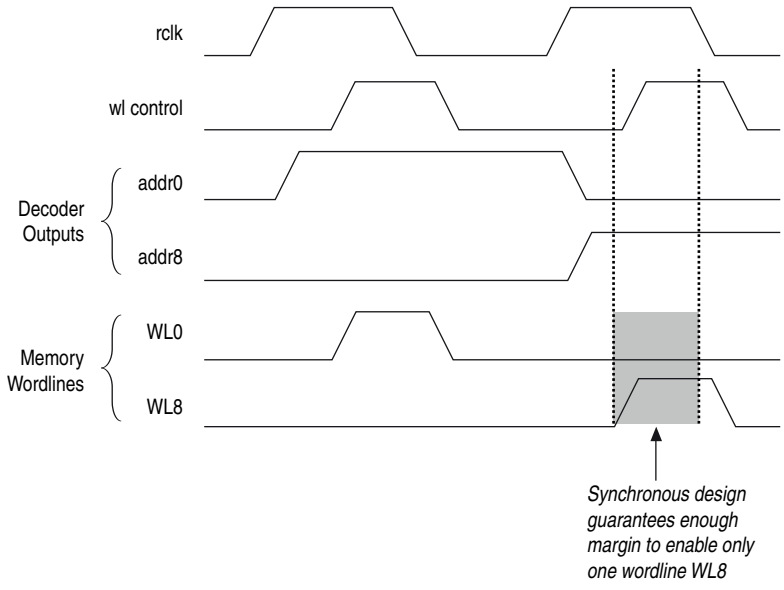
Assume further that the read-enable signal is tied active high so the address signals are not gated to the address decoder. With a memory address that is 4-bits wide, the decoder can address 16 memory locations from wordlines `WL0` to `WL15` (Figure 4).

Figure 4. Simplified Diagram of M9K or M144K Read Example



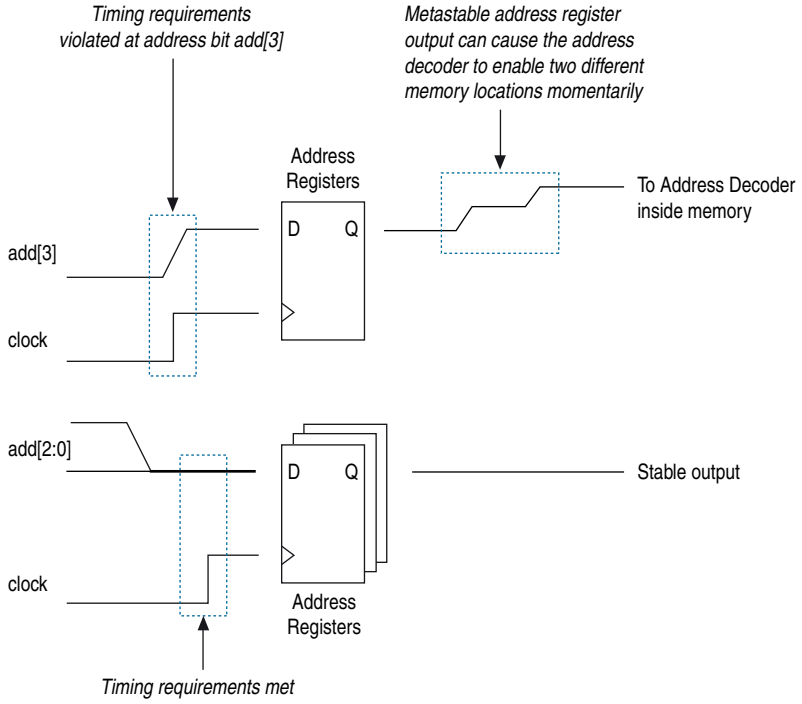
In a normal operation with all the timings met, the address decoder selects only wordline `WL8` because the decoder receives the value 4'b1000 at its input with `add[3]` correctly detected as a stable **1**, as shown in Figure 5.

Figure 5. Proper Read Operation without Timing Violation



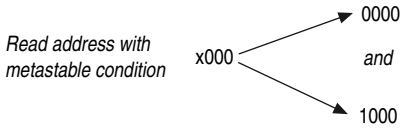
Now assume the address register `add[3]` did not meet the timing requirements. The output then enters a metastable condition. In this condition, the output can be at some undetermined voltage level between logic **1** and logic **0** (Figure 6).

Figure 6. Read Operation with Address Value 4'b1000



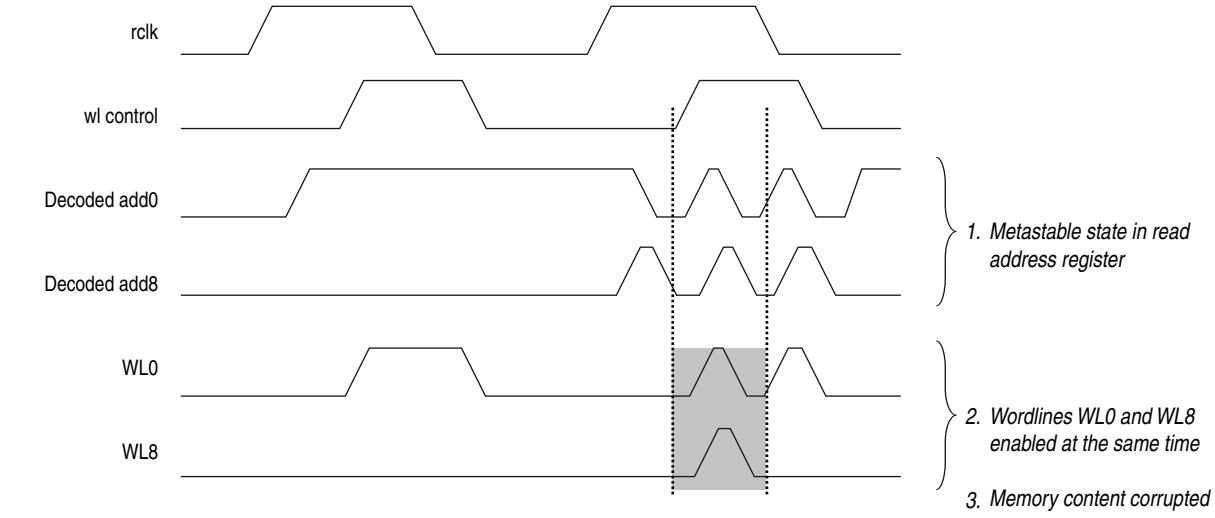
In this condition, the decoder may interpret the output of the whole address register to be momentarily between 4'b0000 and 4'b1000 and selects both wordlines `WL0` and `WL8` (Figure 7).

Figure 7. Metastable Condition



When this happens, charge-sharing can occur between the registers (Figure 8), causing an incorrect value to be read out and the memory content corrupted.

Figure 8. Improper Read Operation Due to Timing Violation in MSB Address Bit

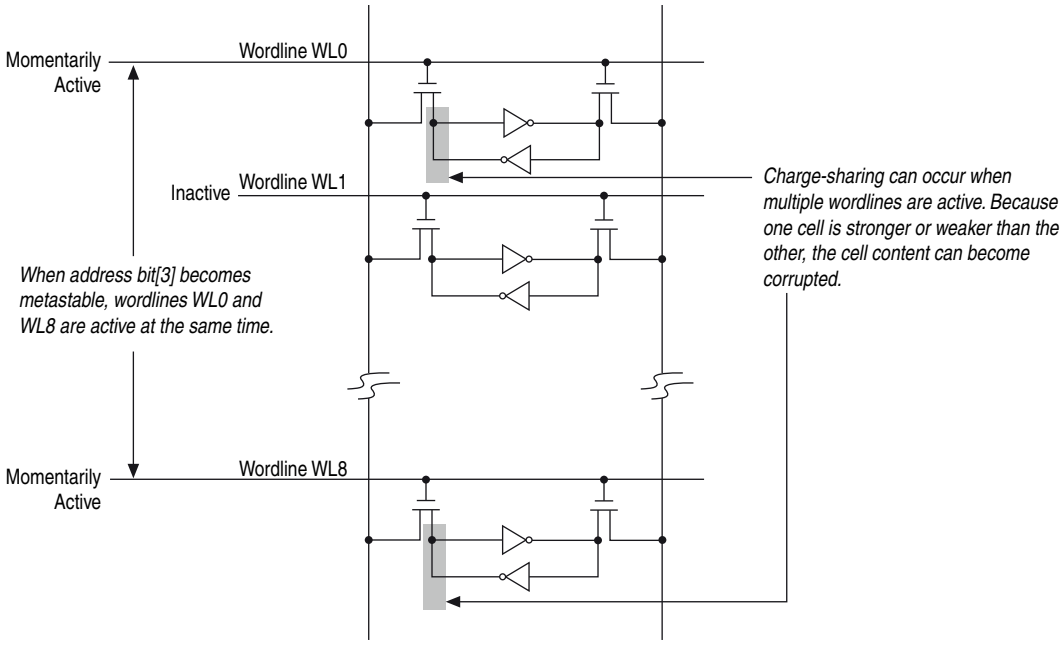


Charge-Sharing

The M4K and M144K RAM contents are determined by the amount of charge stored in them as logic **1** or **0**. When two memory locations are momentarily active, charge-sharing can occur if both cells hold different logic levels and have different hold strength, which can lead to the content corruption.

In Figure 9, the wordlines `WL0` and `WL8` inside the memory block are both made active due to the metastable output from the address register's flipflops. When this situation occurs, the contents in the wordlines can be corrupted due to charge-sharing between these active memory cells. Therefore, a read operation can effectively corrupt the memory contents permanently unless the proper data is written again.

Figure 9. Charge-Sharing Caused by Timing Violation During a Read Operation



Design Considerations

☐ 5. A good design practice is to make sure the read-enable signal is not valid during the address change. A better way is to use the Address Clock Enable feature. This feature sets the `addressstall` signal to 1 to hold the present value loaded on the address register during a clock edge. The address register then avoids going into a metastable state if the address inputs are unresolved at the time of a clock edge arrival.

You can check the Quartus® II timing report for any violation. If you declare a false path on any of the memory address paths, ensure that the setup and hold times are stable during the read cycle.

Whenever M9K or M144K memory blocks are used in your design, the Quartus II software generates the following message during compilation, whether timing is violated or not, to remind you to meet all the timings:

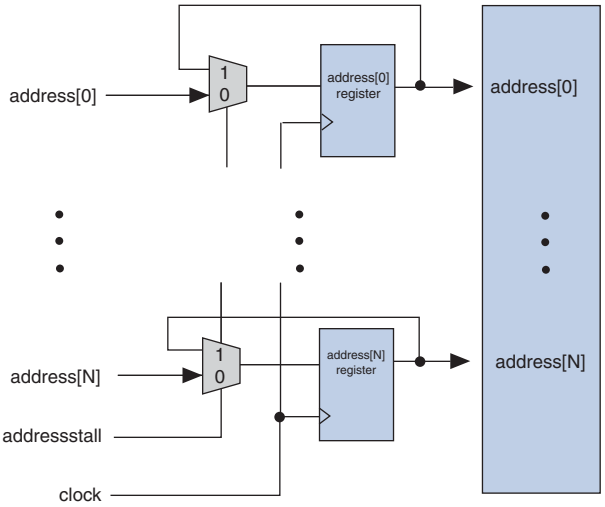
> Info: Design uses memory blocks. Violating setup or hold times of memory block address registers for **either read or write** operations could cause memory contents to be corrupted. Make sure that all memory block address registers meet the setup and hold time requirements.

If possible, convert all the asynchronous paths on the RAM address registers to be fully synchronous in your RTL logic.

Carefully analyze false paths to make sure they are truly "don't care" static signals. These signals must not change value and cause an unresolved signal to be seen on the RAM address register during the valid read operation.

In cases where there are multicycle paths to the RAM address registers, use the Address Clock Enable feature (Figure 10).

Figure 10. Address Clock Enable



For more information about the address stall, refer to the *TriMatrix Embedded Memory Blocks* chapters in the *Stratix III* or *Stratix IV* handbooks.

Power-Up Initial Values

You can initialize values for the memories in an FPGA during configuration using the values stored in the **.mif** file. However, HardCopy internal memories are powered up uninitialized with unknown values.

☐ 6. You should first perform a write operation with known values or automatically read from a ROM into these memories before executing any read operation.

Because the initial memories' power-up values are unknown, incorporate this condition in your functional simulation to verify your HardCopy functionality using initial unknown values.

Read-During-Write Output Data

For the TriMatrix memory blocks (MLAB, M9K, and M144K), depending on the types of mixed-port read-during-write mode (reading and writing at the same memory location), there can be three output choices for the RAM outputs: "new data," "old data," and "don't care."

For the "new data" and "old data" modes, a read-during-write operation causes the RAM output to reflect the new data or old data, respectively.

If you design your own parity checker on this data outside the memory block, the parity error will not be flagged under "new data" and "old data" modes if the data is not corrupted, ensuring proper parity.

However, if you choose the "don't care" mode, your custom parity checker may occasionally flag parity errors because the data coming out is not the complete new or previous data, but possibly a random mix of old and new data with parity violations.

☐ 7. Make sure that your design takes this situation into consideration. The system parity error checker takes into consideration when the "don't care" option is selected.

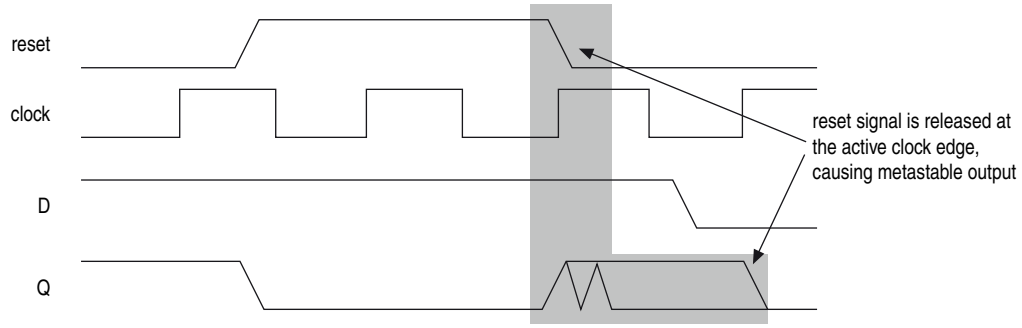
Due to routing differences of the internal memory blocks between FPGA and HardCopy devices, you may consistently read "new data" or "old data" in the FPGA while in "don't care" mode and not see any parity errors. However, in the HardCopy device you may read random data, causing the parity error flag to occur.

Asynchronous Reset Structure

In an asynchronous reset, the assertion is not much of a concern, but the deassertion, also called reset removal, can be a problem. Asynchronous assertion takes effect immediately, but when the asynchronous reset is released at or near the clock edge of the flipflop, the output may go metastable and the reset state may not be effective. A metastable condition at the register output can cause an unexpected reset sequence to occur within your design. If you have an asynchronous reset at the top level of your design, you must consider the metastability behavior when this signal is propagated through your design, as shown in Figure 11.

☐ 8. Check whether your design can tolerate or ignore this metastable output.

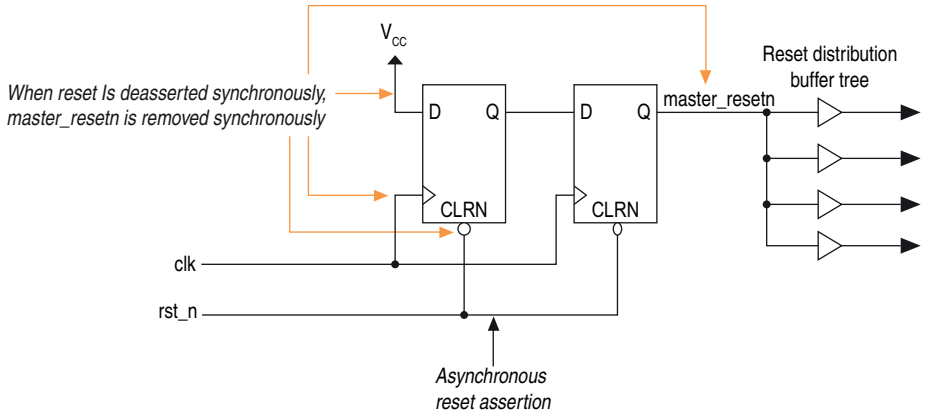
Figure 11. Asynchronous Reset Released Near the Clock Edge



☐ 9. You can implement a resynchronization register to synchronize the asynchronous reset to remove any metastability that might be caused by the reset signal being deasserted too close to the clock edge.

Figure 12 shows an example of synchronizing the asynchronous reset.

Figure 12. Asynchronous Reset Assertion and Synchronous Reset Removal at the Top Level Design Structure



In an asynchronous reset system, design each clock domain block to have its own local reset recovery or removal circuit.

Other considerations when designing the reset structure include the following:

☐ 10. For debug purposes, your design should allow the reset of PLLs and PLL reconfiguration blocks to be separate from other logic blocks in the design. This practice ensures that you see the check functional performance between your FPGA and HardCopy devices.

☐ 11. Ensure that all cases in a state-machine are dealt with. Due to timing differences, the FPGA may not enter a dead state in a state-machine, while a dead state may be encountered in a HardCopy device. You should also use explicit reset for all sequential elements.

☐ 12. Use a synchronous reset in your system.

Power Supply Compatibility

☐ 13. If the VCCL is 1.1 V in a Stratix III device and you want to migrate the design to a HardCopy III device, you must ensure the board voltage regulator can switch to a 0.9 V rail for the HardCopy VCCL power supply.

This requirement is necessary because the HardCopy III device is manufactured with a 40 nm process and uses 0.9 V for V_{CCL}, while the Stratix III device uses a 60 nm process.

Package and Thermal Considerations

- ☐ 14.If you are migrating from a lid-package FPGA to a lidless-package HardCopy device, consider carefully the thermal heat dissipation, even though the HardCopy device might have lower overall power consumption.

A HardCopy die can still experience a higher temperature than an FPGA, because the HardCopy die can have a higher thermal density due to its smaller die size and lidless package. A lidless package has a higher thermal resistance; therefore, it is harder for the die to dissipate its junction heat to the ambient. The lid of an FPGA die package acts as a heat sink and therefore dissipates thermal heat more efficiently.

Perform thermal simulation on the HardCopy device to ensure that you have sufficient thermal transfer.

Consider the physical dimension of the heat sink if you are performing a non-socket replacement flow that targets a different package outline dimension.

Power

You can typically expect to see a reduction in both dynamic and static power when migrating from an FPGA to a HardCopy ASIC device.

- ☐ 15.Use the Early Power Estimator (EPE) to determine a conservative approximation of the power consumption in the HardCopy device to ensure it is within the power management. HardCopy I/O power consumption is the same as that of the FPGA.

Timing

Before migrating your design to a HardCopy device, do the following:

- ☐ 16.Close the I/O timings in your Quartus II HardCopy compilation.
- ☐ 17.Make sure that there is no core setup violation.

A small hold time violation is acceptable, but it is subject to review by the HardCopy Design Center to see if this timing violation can be closed in the back end for migration. Reset recovery and removal timings must be without timing violations.

External PLL Feedback Clock

If you are using the external feedback compensation mode for the PLL, treat the external clock feedback signal as any other clock signal.

- ☐ 18.Ensure that this signal is routed as a high frequency trace with proper termination and guarded from high voltage switching noise coupling.

Reference Voltages

☐ 19.Ensure that the power supply for the VREF pin is robust and stable.

Any variation on the power supply translates into noise for any circuit that depends on it. This noise can manifest itself as jitter, voltage threshold variation, or reduce the timing margin inside the device.

Temperature Sensing Diode

Stratix IV and HardCopy IV devices offer a temperature sensing diode (TSD) that self-monitors the device junction temperature and can be used with external circuitry. The internal TSD is a very sensitive circuit that can be influenced by noise coupled with other traces on the board, switching I/Os, and possibly within the device package itself, depending on your device usage.

☐ 20.Altera recommends taking temperature readings during periods of device inactivity if you are using the TSD with the internal or external ADC.

You should continue taking temperature readings for the HardCopy device during periods of inactivity even if the FPGA prototype shows accurate temperature readings with device activity happening. For board connection guidelines for the TSD external pin connections for both FPGA and HardCopy designs, refer to the *Power Management in Stratix IV Devices* chapter in the *Stratix IV Device Handbook*.

Document Revision History

Table 1 lists the revision history for this application note.

Table 1. Document Revision History

Date	Version	Changes
July 2012	1.1	Added check list.
April 2012	1.0	Initial release.

Design Checklist

This checklist provides a summary of the guidelines described in this application note. Use the checklist to verify that you have followed the guidelines for each stage of your design.

Done	N/A	
☐	☐	1. All power supplies should ramp up within the specified t_{RAMP} time in a monotonic manner (without a plateau) to avoid a brown-out condition.
☐	☐	2. Make sure your system power sequence management takes into consideration the minimum and maximum delay boundary values.
☐	☐	3. All power supplies must be at full rails before entering user mode in the HardCopy device.
☐	☐	4. For a valid read or write operation to the M9K and M144K memories, the memory block address registers must meet the setup and hold time requirements to avoid corrupting the memory contents.
☐	☐	5. Use the Address Clock Enable feature to make sure the read-enable signal is not valid during the address change.
☐	☐	6. Perform a write operation with known values or automatically read from a ROM into the HardCopy internal memories before executing any read operation.
☐	☐	7. Make sure that your design takes the different output choices into consideration. The system parity error checker takes into consideration when the “don’t care” option is selected.
☐	☐	8. Check whether your design can tolerate or ignore a metastable output.
☐	☐	9. Implement a resynchronization register to synchronize the asynchronous reset.
☐	☐	10.For debug purposes, your design should allow the reset of PLLs and PLL reconfiguration blocks to be separate from other logic blocks in the design.
☐	☐	11.Ensure that all cases in a state-machine are dealt with. Use explicit reset for all sequential elements.
☐	☐	12.Use a synchronous reset in your system.
☐	☐	13.If the VCCL is 1.1 V in a Stratix III device and you want to migrate the design to a HardCopy III device, you must ensure the board voltage regulator can switch to a 0.9 V rail for the HardCopy VCCL power supply.
☐	☐	14.If you are migrating from a lid-package FPGA to a lidless-package HardCopy device, consider carefully the thermal heat dissipation, even though the HardCopy device might have lower overall power consumption.
☐	☐	15.Use the Early Power Estimator (EPE) to determine a conservative approximation of the power consumption in the HardCopy device to ensure it is within the power management. HardCopy I/O power consumption is the same as that of the FPGA.
☐	☐	16.Before migrating your design to a HardCopy device, close the I/O timings in your Quartus II HardCopy compilation.
☐	☐	17.Make sure that there is no core setup violation.
☐	☐	18.Ensure the external clock feedback signal is routed as a high frequency trace with proper termination and guarded from high voltage switching noise coupling.
☐	☐	19.Ensure that the power supply for the VREF pin is robust and stable.
☐	☐	20.Altera recommends taking temperature readings during periods of device inactivity if you are using the TSD with the internal or external ADC.