XILINX®

Chapter 7

FPGA Configuration

This chapter discusses how to configure the Xilinx® 7 series FPGA so that the device can link up and be recognized by the system. This information is provided for the user to choose the correct FPGA configuration method for the system and verify that it works as expected.

This chapter discusses how specific requirements of the *PCI Express Base Specification* and *PCI Express Card Electromechanical Specification* apply to FPGA configuration. Where appropriate, Xilinx recommends that the user read the actual specifications for detailed information. See Tandem PROM, page 204 for more information on meeting configuration requirements after reading this section.

This chapter is divided into four sections:

- Configuration Terminology. Defines terms used in this chapter.
- Configuration Access Time. Several specification items govern when an Endpoint device needs to be ready to receive configuration accesses from the host (Root Complex).
- Board Power in Real-World Systems. Understanding real-world system constraints related to board power and how they affect the specification requirements.
- Recommendations. Describes methods for FPGA configuration and includes sample problem analysis for FPGA configuration timing issues.

Configuration Terminology

In this chapter, these terms are used to differentiate between FPGA configuration and configuration of the PCI Express® device:

- Configuration of the FPGA. *FPGA configuration* is used.
- Configuration of the PCI Express device. After the link is active, *configuration* is used.

Configuration Access Time

In standard systems for PCI Express, when the system is powered up, configuration software running on the processor starts scanning the PCI Express bus to discover the machine topology.

The process of scanning the PCI Express hierarchy to determine its topology is referred to as the *enumeration process*. The root complex accomplishes this by initiating configuration transactions to devices as it traverses and determines the topology.

All PCI Express devices are expected to have established the link with their link partner and be ready to accept configuration requests during the enumeration process. As a result, there are requirements as to when a device needs to be ready to accept configuration requests after power up; if the requirements are not met, this occurs:

- If a device is not ready and does not respond to configuration requests, the root complex does not discover it and treats it as non-existent.
- The operating system does not report the device's existence and the user's application is not able to communicate with the device.

Choosing the appropriate FPGA configuration method is key to ensuring the device is able to communicate with the system in time to achieve link up and respond to the configuration accesses.

Configuration Access Specification Requirements

Two PCI Express specification items are relevant to configuration access:

1. Section 6.6 of *PCI Express Base Specification*, rev 1.1 states "A system must guarantee that all components intended to be software visible at boot time are ready to receive Configuration Requests within 100 ms of the end of Fundamental Reset at the Root Complex." For detailed information about how this is accomplished, see the specification; it is beyond the scope of this discussion.

   Xilinx compliance to this specification is validated by the PCI Express-CV tests. The PCI Special Interest Group (PCI-SIG) provides the PCI Express Configuration Test Software to verify the device meets the requirement of being able to receive configuration accesses within 100 ms of the end of the fundamental reset. The software, available to any member of the PCI-SIG, generates several resets using the in-band reset mechanism and PERST# toggling to validate robustness and compliance to the specification.

2. Section 6.6 of *PCI Express Base Specification v1.1* defines three parameters necessary "where power and PERST# are supplied." The parameter T_{PVPERL} applies to FPGA configuration timing and is defined as:

   T_{PVPERL} - PERST# must remain active at least this long after power becomes valid.

   The *PCI Express Base Specification* does not give a specific value for T_{PVPERL} – only its meaning is defined. The most common form factor used by designers with the Integrated Block core is an ATX-based form factor. The *PCI Express Card Electromechanical Specification* focuses on requirements for ATX-based form factors. This applies to most designs targeted to standard desktop or server type motherboards. Figure 7-1 shows the relationship between Power Stable and PERST#.

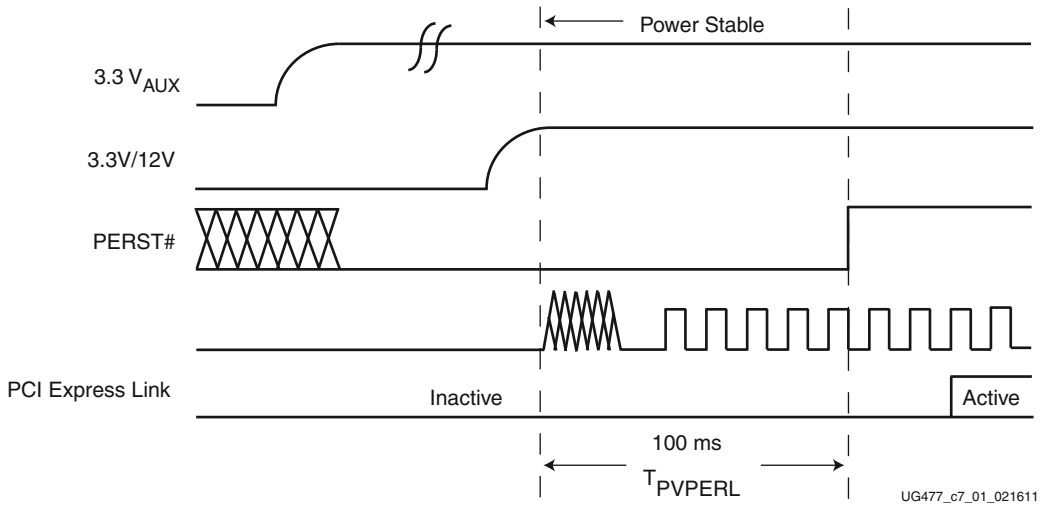

Figure 7-1: **Power Up**

Section 2.6.2 of the *PCI Express Card Electromechanical Specification, v1.1* defines T_{PVPREL} as a minimum of 100 ms, indicating that from the time power is stable the system reset is asserted for at least 100 ms (as shown in Table 7-1).

Table 7-1: **T_{PVPERL} Specification**

Symbol	Parameter	Min	Max	Units
T_{PVPERL}	Power stable to PERST# inactive	100		ms

From Figure 7-1 and Table 7-1, it is possible to obtain a simple equation to define the FPGA configuration time as follows:

$$\text{FPGA Configuration Time} \leq T_{PWRVLD} + T_{PVPERL} \qquad \textit{Equation 7-1}$$

Given that T_{PVPERL} is defined as 100 ms minimum, this becomes:

$$\text{FPGA Configuration Time} \leq T_{PWRVLD} + 100\ \text{ms} \qquad \textit{Equation 7-2}$$

Note: Although T_{PWRVLD} is included in Equation 7-2, it has yet to be defined in this discussion because it depends on the type of system in use. The Board Power in Real-World Systems section defines T_{PWRVLD} for both ATX-based and non ATX-based systems.

FPGA configuration time is only relevant at cold boot; subsequent warm or hot resets do not cause reconfiguration of the FPGA. If the design appears to be having issues due to FPGA configuration, the user should issue a warm reset as a simple test, which resets the system, including the PCI Express link, but keeps the board powered. If the issue does not appear, the issue could be FPGA configuration time related.

Board Power in Real-World Systems

Several boards are used in PCI Express systems. The *ATX Power Supply Design* specification, endorsed by Intel, is used as a guideline and for this reason followed in the majority of mother boards and 100% of the time if it is an Intel-based motherboard. The relationship between power rails and power valid signaling is described in the ATX 12V Power Supply Design Guide. Figure 7-2, redrawn here and simplified to show the information relevant to FPGA configuration, is based on the information and diagram found in section 3.3 of the *ATX 12V Power Supply Design Guide*. For the entire diagram and definition of all parameters, see the *ATX 12V Power Supply Design Guide*.

Figure 7-2 shows that power stable indication from Figure 7-1 for the PCI Express system is indicated by the assertion of PWR_OK. PWR_OK is asserted High after some delay when the power supply has reached 95% of nominal.

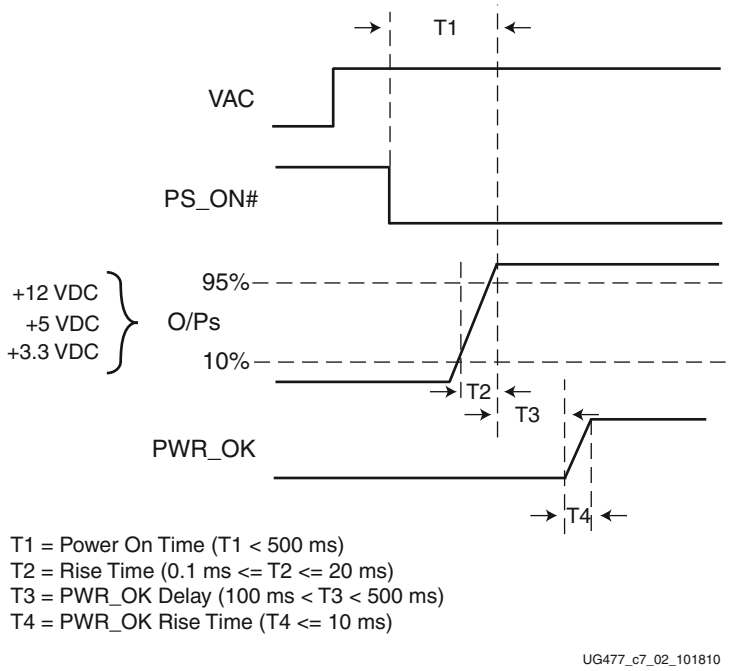

Figure 7-2: **ATX Power Supply**

Figure 7-2 shows that power is valid before PWR_OK is asserted High. This is represented by T3 and is the PWR_OK delay. The *ATX 12V Power Supply Design Guide* defines PWR_OK as 100 ms < T3 < 500 ms, indicating that from the point at which the power level reaches 95% of nominal, there is a minimum of at least 100 ms but no more than 500 ms of delay before PWR_OK is asserted. Remember, according to the *PCI Express Card Electromechanical Specification*, the PERST# is guaranteed to be asserted a minimum of 100 ms from when power is stable indicated in an ATX system by the assertion of PWR_OK.

Again, the FPGA configuration time equation is:

$$\text{FPGA Configuration Time} \leq T_{PWRVLD} + 100\ \text{ms} \qquad \textit{Equation 7-3}$$

T_{PWRVLD} is defined as PWR_OK delay period; that is, T_{PWRVLD} represents the amount of time that power is valid in the system before PWR_OK is asserted. This time can be added to the amount of time the FPGA has to configure. The minimum values of T2 and T4 are negligible and considered zero for purposes of these calculations. For ATX-based

motherboards, which represent the majority of real-world motherboards in use, T_{PWRVLD} can be defined as:

$$100\ ms \leq T_{PWRVLD} \leq 500\ ms \quad \textit{Equation 7-4}$$

This provides these requirements for FPGA configuration time in both ATX and non-ATX-based motherboards:

- FPGA Configuration Time ≤ 200 ms (for ATX based motherboard)
- FPGA Configuration Time ≤ 100 ms (for non-ATX based motherboard)

The second equation for the non-ATX based motherboards assumes a T_{PWRVLD} value of 0 ms because it is not defined in this context. Designers with non-ATX based motherboards should evaluate their own power supply design to obtain a value for T_{PWRVLD}.

This chapter assumes that the FPGA power (V_{CCINT}) is stable before or at the same time that PWR_OK is asserted. If this is not the case, then additional time must be subtracted from the available time for FPGA configuration. Xilinx recommends to avoid designing add-in cards with staggered voltage regulators with long delays.

Hot Plug Systems

Hot Plug systems generally employ the use of a Hot-Plug Power Controller located on the system motherboard. Many discrete Hot-Plug Power Controllers extend T_{PVPERL} beyond the minimum 100 ms. Add-in card designers should consult the Hot-Plug Power Controller data sheet to determine the value of T_{PVPERL}. If the Hot-Plug Power Controller is unknown, then a T_{PVPERL} value of 100 ms should be assumed.

Recommendations

For minimum FPGA configuration time, Xilinx recommends the BPI configuration mode with a parallel NOR flash, which supports high-speed synchronous read operation. In addition, an external clock source can be supplied to the external master configuration clock (EMCCLK) pin to ensure a consistent configuration clock frequency for all conditions. See UG470, *7 Series FPGAs Configuration User Guide* [Ref 2], for descriptions of the BPI configuration mode and EMCCLK pin. This section discusses these recommendations and includes sample analysis of potential issues that might arise during FPGA configuration.

FPGA Configuration Times for 7 Series Devices

During power up, the FPGA configuration sequence is performed in four steps:

1. Wait for power on reset (POR) for all voltages (V_{CCINT}, V_{CCAUX}, and VCCO_0) in the FPGA to trip, referred to as POR Trip Time.
2. Wait for completion (deassertion) of INIT_B to allow the FPGA to initialize before accepting a bitstream transfer.

   Note: As a general rule, steps 1 and 2 require ≤ 50 ms
3. Wait for assertion of DONE, the actual time required for a bitstream to transfer depends on:
   - Bitstream size
   - Clock (CCLK) frequency
   - Transfer mode (and data bus width) from the flash device

- SPI = Serial Peripheral Interface (x1, x2, or x4)
- BPI = Byte Peripheral Interface (x8 or x16)

Bitstream transfer time can be estimated using this equation.

Bitstream transfer time = (bitstream size in bits)/(CCLK frequency)/ (data bus width in bits) **Equation 7-5**

For detailed information about the configuration process, see the *7 Series FPGAs Configuration User Guide* (UG470).

Sample Problem Analysis

This section presents data from an ASUS PL5 system to demonstrate the relationships between Power Valid, FPGA Configuration, and PERST#. Figure 7-3 shows a case where the Endpoint failed to be recognized due to a FPGA configuration time issue. Figure 7-4 shows a successful FPGA configuration with the Endpoint being recognized by the system.

Failed FPGA Recognition

Figure 7-3 illustrates an example of a cold boot where the host failed to recognize the Xilinx FPGA. Although a second PERST# pulse assists in allowing more time for the FPGA to configure, the slowness of the FPGA configuration clock (2 MHz) causes configuration to complete well after this second deassertion. During this time, the system enumerated the bus and did not recognize the FPGA.

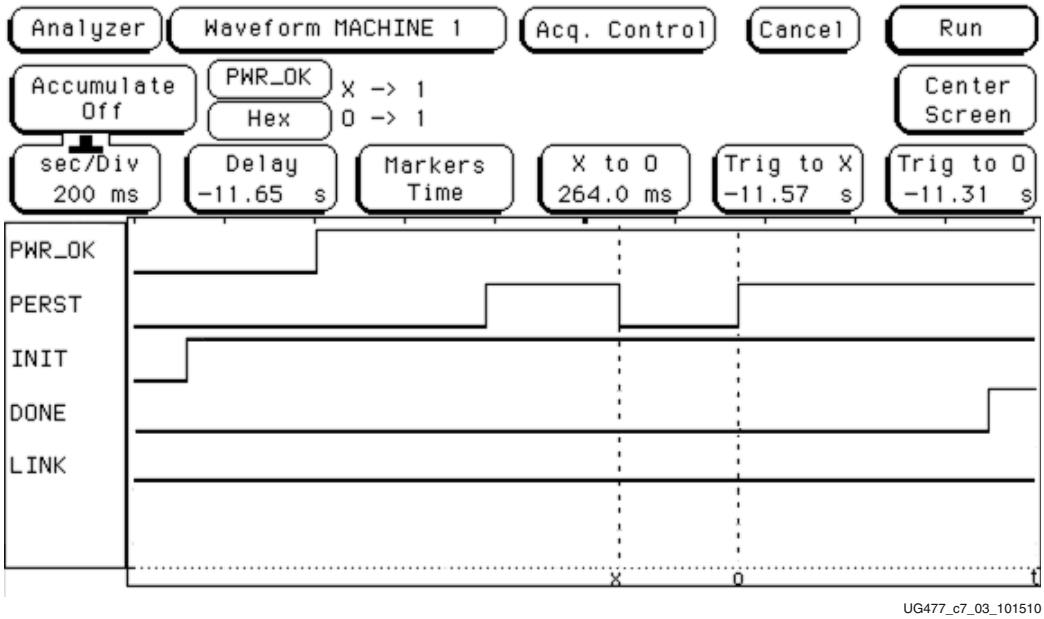

Figure 7-3: **Host Fails to Recognize FPGA Due to Slow Configuration Time**

Successful FPGA Recognition

Figure 7-4 illustrates a successful cold boot test on the same system. In this test, the CCLK was running at 50 MHz, allowing the FPGA to configure in time to be enumerated and recognized. The figure shows that the FPGA began initialization approximately 250 ms before PWR_OK. DONE going High shows that the FPGA was configured even before PWR_OK was asserted.

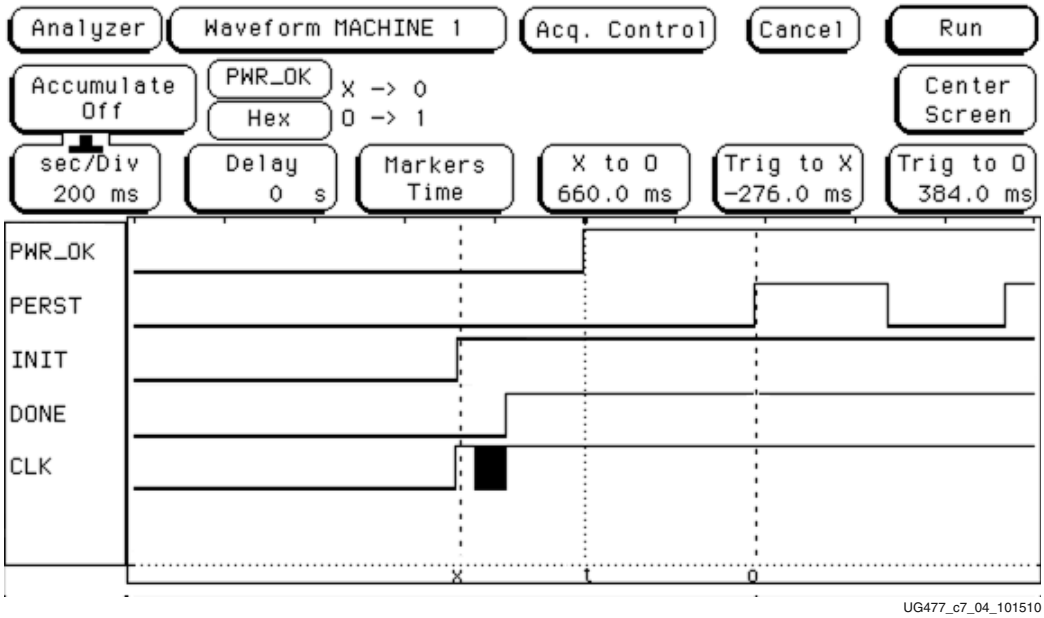

Figure 7-4: **Host Successfully Recognizes FPGA**

Workarounds for Closed Systems

For failing FPGA configuration combinations, designers might be able to work around the issue in closed systems or systems where they can guarantee behavior. These options are not recommended for products where the targeted end system is unknown.

1. Check if the motherboard and BIOS generate multiple PERST# pulses at start-up. This can be determined by capturing the signal on the board using an oscilloscope. This is similar to what is shown in Figure 7-3. If multiple PERST# pulses are generated, this typically adds extra time for FPGA configuration.

   Define $T_{PERSTPERIOD}$ as the total sum of the pulse width of PERST# and deassertion period before the next PERST# pulse arrives. Because the FPGA is not power cycled or reconfigured with additional PERST# assertions, the $T_{PERSTPERIOD}$ number can be added to the FPGA configuration equation.

   $$\text{FPGA Configuration Time} \leq T_{PWRVLD} + T_{PERSTPERIOD} + 100\ \text{ms} \qquad \textit{Equation 7-6}$$

2. In closed systems, it might be possible to create scripts to force the system to perform a warm reset after the FPGA is configured, after the initial power up sequence. This resets the system along with the PCI Express subsystem allowing the device to be recognized by the system.