Using Manual Power Down Mode With SpartanXL™ FPGAs

XAPP124 March 22, 1999 (Version 1.1)

Application Note By: Kim Goldblatt and Ashok Chotai

Summary

SpartanXL FPGAs come equipped with a Power Down mode that permits an exceptionally low level of power consumption (I_{CCO} = 100 µA typical), making the family ideal for portable battery-powered applications. This application note provides all the information the designer needs to use Power Down mode effectively, including descriptions of the mode's common applications, internal functioning and electrical characteristics.

Xilinx Families

SpartanXL Only

Introduction

The SpartanXL family is equipped with a feature known as Power Down mode, that significantly reduces the FPGA's quiescent supply current requirements. The mode becomes active once a low-level pulse is applied to the $\overline{\text{PWRDWN}}$ pin. This makes the family ideal for portable battery-powered applications as well as for designs with tight thermal budgets. Powered by a 3.3V supply, the SpartanXL family can draw up to a maximum quiescent supply current of I_{CCO} = 5 mA under normal operating conditions. Once in Power Down mode, the I_{CCO} is 100 µA typical. This is substantially lower than the quiescent supply current requirements for other manufacturers' FPGAs. For example, Altera's FLEX 10KA has an I_{CCO} specification of 300 µA typical, whereas the company's FLEX 6KA has an I_{CCO} specification of 500 µA typical.

For any given design, Power Down mode provides a reduction in I_{CCO} down to the minimum levels possible with SpartanXL FPGAs.

Low Power Applications

The low quiescent supply current possible with Power Down mode is especially useful for products that must use power as efficiently as possible. This applies to battery-powered equipment such as notebook computers, digital cameras, camcorders and cellular telephones. Satellites, which must make the best possible use of the limited power from solar panels, serve as yet another example.

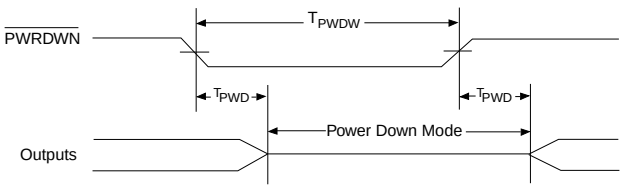
Since the logic within the FPGA is disabled during Power Down mode, the SpartanXL family is particularly well suited to power-sensitive applications that are called into use intermittently. In such cases, the SpartanXL device saves significant quiescent power while internal functions are temporarily suspended.

Power Down Modes

There are two kinds of power down modes that can be used to conserve power in SpartanXL FPGAs. They are: manual mode and the automatic mode. Both of them provide low quiescent (standby) power while retaining the bit map (i.e., configuration data) with which it had been configured. In the manual mode, the device is fully inactive, the register data is lost, and the activation and de-activation of the power down mode is controlled by the $\overline{\text{PWRDWN}}$ pin. In the automatic mode, the device is active, the register data is maintained and the power is user controlled by selectively controlling the features of a design which consume relatively a large amount of power. This application note discusses the manual power down mode. For automatic power down mode, refer to the application note *XAPP125, Conserving Power With Auto Power Down Mode In SpartanXL™ FPGAs* at http://www.xilinx.com/xapp/xapp125.pdf on the Xilinx web site.

How Manual Power Down Mode Works

Power Down mode is initiated by applying a low-level pulse to the $\overline{\text{PWRDWN}}$ input of the SpartanXL device. The timing is shown in Figure 1. The pulse needs to be a minimum of 50 ns wide. Both entering the mode (on the falling edge) and exiting the mode (on the rising edge) are accomplished



Description	Symbol	Min	Max
Power Down Time	T_{PWD}	-	50ns
Power Down Pulse Width	T_{PWDW}	50ns	-

xap124_1

Figure 1: $\overline{\text{PWRDWN}}$ Pulse Timing

asynchronously. From the point at which $\overline{\text{PWRDWN}}$ is taken low, it takes 50 ns (max.) for the device to fully enter Power Down mode. The device takes a number of steps to conserve power which are described below in the order that they occur:

- All inputs (including M0, M1, DONE, CCLK and TDO) except $\overline{\text{PWRDWN}}$ are disconnected from their sources. Internal to the device, the input signals are pulled to GND, eliminating the power associated with any dynamic activity on the input pins.
- All pull-up and pull-down resistors on all I/Os (except $\overline{\text{PWRDWN}}$) are disabled, saving the current that would otherwise flow through them. Floating inputs do not cause power consumption while $\overline{\text{PWRDWN}}$ is Low.
- The Global Set-Reset or GSR net is activated, clearing all registers in the device. This reset state is held as long as the device remains in Power Down mode. As a result, not only is internally generated dynamic power reduced to zero, but any glitches that might occur while entering or exiting Power Down mode are also avoided.
- The Global Three State or GTS net is activated, putting the device outputs into a high-impedance state. This action saves the current used to drive the load that is connected to the outputs.

The user has the option of activating either the GTS net or the GSR net earlier, provided the STARTUP block has been instantiated in the design and the GTS and GSR enable lines have been assigned to I/Os.

The device remains in a DC state, drawing minimal power, until $\overline{\text{PWRDWN}}$ goes High, at which point it returns to full operation over a period of 50 ns (max). During this period, the inputs go active first, then GSR is released and finally the outputs are enabled. The device is now ready to recommence logical operation. The user has the option of keeping either the GSR net or the GTS net active after exiting Power Down mode. This practice can be used to delay the release of the GSR net beyond the enabling of the outputs.

Throughout Power Down mode, the device retains the bit map (i.e., configuration data) with which it had been configured, but loses all the data stored in the device. The combinatorial logic is fully functional, but the all the registered logic is reset. Once having exited the mode, logical operation recommences from the reset state.

Power Down Mode in Practice

In order to use Power Down mode effectively, it is important to observe the guidelines described in this section.

The design must be carefully checked to avoid any possible contention that might result during Power Down mode. $\overline{\text{PWRDWN}}$ disconnects all the devices' internal input nodes from the pins on the package and pulls them Low. Contention can occur in cases where one node has the effect of driving an internal line High when another node causes the same line to be drive Low. For example, Figure 2 shows

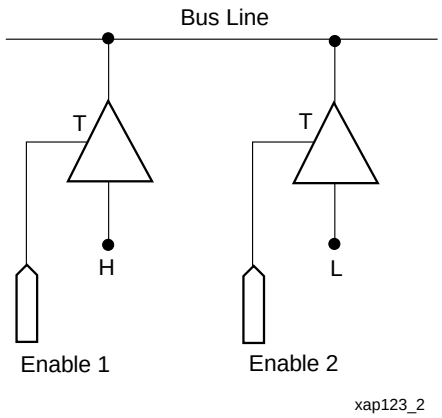


Figure 2: $\overline{\text{PWRDWN}}$ Mode Contention

two three-state buffers (BUFT) that drive a common bus line. The input to the first buffer is HIGH whereas the input to the second buffer is LOW. During normal operation, only one of the two enable inputs is permitted to go Low at any time, so only one buffer can drive the line. Once in Power Down mode, however, both Enable 1 and Enable 2 will go Low at the same time, causing contention on the bus line.

To achieve the minimum possible quiescent I_{CCO} level (100 µA) during the Power Down period, it is necessary to turn OFF the “5V tolerant I/Os (SpartanXL only)” option when implementing the design. The option can be found in the Configuration Options template, which is a part of the Xilinx development software. Selecting 5V tolerance (the default), enables a circuit that draws additional current to protect the input buffers from large signals.

The 3.3V signal standards compatible with the SpartanXL inputs, LVTTL, LVCMOS and PCI3V, all allow for minimum quiescent I_{CC} in Power Down mode, since none of these require 5V tolerance. However, using the 5V signal standards TTL and PCI5V, both of which require 5V tolerance, will result in a higher level of during the Power Down period.

For versions of the Xilinx development software 1.5i or later, the internal pull-up resistor associated with the $\overline{\text{PWRDWN}}$ pin is enabled by default during implementation. To achieve the minimum possible supply current in Power Down mode, disable this resistor. The option can be found in the Configuration Options template, which is a part of the Xilinx development software.

When coming out of Power Down mode, care must be taken to avoid glitches from occurring in the case that the SpartanXL device's outputs go active while the inputs of the next circuit stage are receptive to signal transitions (Figure 3). In this case, it is possible for a SpartanXL output, as it goes from a high impedance state to a High or Low logic level, to cross the switching threshold of the inputs to the next stage, thereby causing a “false” transition. In cases where it is possible to disable the inputs to the next stage, the problem can easily be avoided: simply disable

the inputs for a period of t_{PWDW} following the rising transition of $\overline{PWRDWN}$.

If the inputs of the next stage cannot be disabled, then global reset for that stage can be used to make sure there is no false transition.

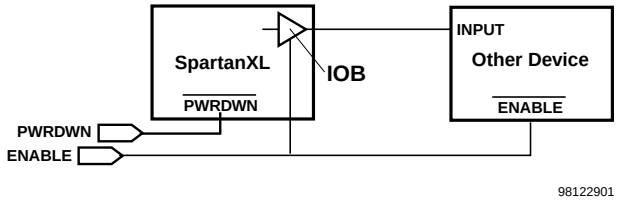


Figure 3: Glitch Scenario

State Recovery

When the device comes out of Power Down mode, it will be in a reset state. While most designs can recommence operation directly from the reset state, some designs need to continue functioning from the “last” state (i.e., the state that existed just prior to power down). This is particularly true for sparse-input synchronous state machines. In such cases, the design must provide a way to restore the “last” state.

In the general case, the last state differs from one instance of power down to the next. Here, recovery may simply be a matter of providing the shortest sequence of input patterns to the device that is necessary to guide the machine back to the last state. An alternative solution uses a “state bus” (or a serial path) that is multiplexed to the inputs and outputs of those registers whose bit values define the state of the machine. The bus captures the state of the registers just before entering Power Down mode. The bit values are stored outside of the device during the Power Down period. Once the device comes out of Power Down mode (t_{PWDW} after pulling $\overline{PWRDWN}$ High), the bit values are clocked back into their corresponding registers.

The “serial path” version employs a design similar to that used for JTAG boundary scan. (See *Figure 2* in *XAPP17, Boundary Scan in XC4000 and XC5000 Series Devices* for a schematic of the boundary scan logic that is built into the IOBs of SpartanXL devices.) Provided that a sufficient number of IOBs can be dedicated to the process of state recovery, the built-in boundary scan logic can be used as the serial path that stores and restores the register bit values. Implementing this capability becomes even easier when the state machine employs registers in IOBs, since the boundary scan logic has direct access to these registers. If an insufficient number of IOBs are available, the serial path can be created using other available logic resources in the device (i.e., CLBs).

If the last state happens to be the same every time Power Down mode is initiated, it may be possible to use a reset state that is equivalent to the last state. This is accomplished by selecting either a set or a reset independently for each register, according to whichever GSR option is needed to replicate the last state.

Table 1: Revisions

Version	Description
3/22/99	Rev. 1.1 Modified the title to “Using Manual Power Down Mode in SpartanXL FPGAs”. Added the section “Power Down Modes”.