XAPP689 (v1.2) October 30, 2007

Managing Ground Bounce in Large FPGAs

Author: Tony Duong

Summary

Ground bounce must be controlled to ensure proper operation of high-performance FPGA devices. Particular attention must be applied to minimizing board-level inductance during PCB layout. This document describes calculations to help to ensure that a design meets input undershoot and logic-Low voltage requirements for devices receiving signals from an FPGA. This application note is for the Virtex™-II Pro, Virtex-II, Virtex-E, Spartan™-II, Spartan-IIE, Spartan-3, Spartan-3E, Spartan-3A, Spartan-3AN, and Spartan-3A DSP FPGA families.

Introduction

IC manufacturers have traditionally specified simultaneous switching output (SSO) guidelines for each driver type, based on the number of drivers allowed per power/ground pair. For LVTTL-type drivers, ground bounce can be as much as 800 mV before tripping the input-low threshold. Such a level for ground bounce might no longer be acceptable today.

What Changed?

A number of recent developments warrant revisiting published SSO guidelines. While lead inductance has been whittled down to less than 100 pH per ground pin in flip-chip packages, board-level inductance has more than doubled to a 2–3 nH range in recent years, due to miniaturized vias, smaller trace widths, and increased board thickness. As a result, substantially more ground bounce voltage now develops across the PCB, and this is not fully accounted for in published SSO guidelines. Actual ground bounce voltage on PC boards that use 8 mil vias on 125 mil FR4 material could far exceed the Xilinx ground bounce expectation of 600 mV.

For ease of PCB routing and debugging, there is a tendency to cluster wide buses at one edge of the chip. This causes uneven current distribution and ground bounce within the package. As supply voltage decreases, components can tolerate even less noise. Some new devices tolerate as little as 300 mV undershoot at the input. Also, as load capacitance increases, ground bounce deteriorates. Altogether, these design issues now require more critical handling of ground bounce than ever before.

Package-Level SSO

A key point to note is that ground bounce scales linearly. This makes it possible to determine Weighted Average SSO (WASSO) at the package level for entire designs using drivers of different types, current strengths, and slew rates. WASSO is computed first on a per I/O bank basis, then it is calculated for two adjacent banks, and is calculated again across all banks to determine the effective WASSO for the entire package.

One criterion for package-level SSO ensures that the number of simultaneously switching outputs does not exceed the per-bank limit. Another criterion ensures even distribution of fast/strong drivers across the package. A final criterion ensures that the chip does not generate excessive ground bounce. A quantity called SSO allowance is used in all three constraints. It takes into account design-specific parameters, such as board-level inductance, input logic-low threshold, input undershoot voltage, and output load capacitance.

© 2003–2007 Xilinx, Inc. All rights reserved. XILINX, the Xilinx logo, and other designated brands included herein are trademarks of Xilinx, Inc. All other trademarks are the property of their respective owners.

The following rules apply at the package level:

- WASSO for any bank cannot exceed 1.15 times the SSO allowance. If this (product) number is greater than 100%, it is capped at 100%.
- Average WASSO of two adjacent banks cannot exceed 105% of the SSO allowance. If this (product) number is greater than 100%, it is capped at 100%.
- Package WASSO cannot exceed the SSO allowance.

To allow flexibility in pin-out assignment, bank and two-bank WASSO limits are slightly higher than the SSO allowance. Whenever possible, distribute drivers evenly throughout the package to make WASSO utilization approximately the same in each bank. This minimizes the difference in WASSO utilization from one bank to the next.

SSO Allowance

The SSO allowance is a number ranging from 0 to 100% and is a product of three scaling factors.

The *first scaling factor* is determined by dividing Xilinx-assumed inductance by the user's PCB total inductance. This factor is calculated from board geometries (via diameter, board thickness, breakout trace width and length.)

The *second scaling factor* is determined by dividing the lesser of the input undershoot voltage and input logic low threshold voltage by the maximum ground bounce voltage (600 mV) used by Xilinx to establish SSO guidelines.

The *third scaling factor* is determined by dividing the Xilinx ground bounce of 600 mV for a 15 pf load by ground bounce that is extrapolated for loads greater than 15 pf. Xilinx output drivers produce an additional 9 mV of ground bounce for each additional 1 pf of load capacitance.

- First Scaling Factor = 1.0 nH/1.1 nH = 0.909
- Second Scaling Factor = 550 mV/600 mV = 0.917
- Third Scaling Factor = 600 mV/(((22 pF – 15 pF) × 9 mV/pF) + 600 mV)
  = 600 mV/663 mV = 0.905
- SSO Allowance = (0.909 × 0.917 × 0.905) × 100% = 75.4%.

Calculating Package SSO

A sample calculation of package SSO for a Virtex™-II XC2V6000 device in an FF1152 package follows.

For the number of SSOs allowed per power/ground pair and the number of power/ground pairs per bank, refer to SSO guidelines provided in these documents available on www.xilinx.com:

- Virtex-II Pro Platform FPGA User Guide
- Virtex-II Platform FPGA User Guide
- Virtex-E 1.8V FPGA Product Specification (Module 2)
- Spartan-3A DSP FPGA Family Data Sheet
- Spartan-3A FPGA Family Data Sheet
- Spartan-3AN FPGA Family Data Sheet
- Spartan-3E FPGA Family Data Sheet
- Spartan-3 FPGA Family Data Sheet
- Spartan-IIE FPGA Data Sheet
- Spartan-II FPGA Data Sheet
- Device Package User Guide

Step 0: Calculate the package SSO allowance.

- (1.0 nH/1.1 nH) × (550 mV/600 mV) × (600 mV/(((22 pF – 15 pF) × 9 mV/pF) + 600 mV)) = 75.4%

Step 1: Calculate the WASSO utilization for each individual bank, starting with bank 0, and ensuring each bank's utilization does not exceed 115% of the SSO allowance.

Assume a design uses 36 - 24 mA Fast drivers, 30 - 12 mA Fast drivers, and 22 - 8 mA Slow drivers.

- Percent contribution by 24 mA Fast drivers
  = 36 drivers used/(5 - 24 mA Fast drivers allowed per V_{CC}/GND pair × 13 pairs in I/O Bank 0)
  = 36/65 = 55.4%
- Percent contribution by 12 mA Fast drivers
  = 30 drivers used/(10 - 12 mA Fast drivers allowed per V_{CC}/GND pair × 13 pairs in I/O Bank 0)
  = 30/130 = 23.1%
- Percent contribution by 8 mA Slow drivers
  = 22 drivers used/(22 - 8 mA Slow drivers allowed per V_{CC}/GND pair × 13 pairs in I/O Bank 0)
  = 22/286 = 7.7%
- WASSO utilization for Bank 0 = 55.4% + 23.1% + 7.7% = 86.2%
- The WASSO calculation for banks 1 through 7 is the same as for Bank 0. For this example, assume WASSOs of 45%, 50%, 60%, 60%, 35%, 40%, and 15% are obtained. Ensure the WASSO for each bank does not exceed 115% of the SSO allowance. Since WASSO for bank 0 is 86.2%, is less than 86.7% (calculated from 75.4% × 1.15), this percentage is acceptable. If the Bank WASSO exceeds 115% of the SSO allowance, apply ground bounce reduction techniques to the bank in question.

Step 2: Calculate the WASSO utilization for two adjacent banks, and ensure it does not exceed 105% of the SSO allowance.

- Two-Bank WASSO utilization between Banks 0 and 1
  = ((WASSO of Bank 0) × (number of power/GND pairs in bank 0) + (WASSO of Bank 1) × (number of power/GND pairs in bank 1))/(total number of power/GND pairs in banks 0 and 1)
  = (86.2% × 13 power/GND pairs + 45% × 13 power/GND pairs)/(13 + 13 power/GND pairs)
  = 65.6%
  Since 65.6% is less than 79.2% (from 75.4% × 1.05), this percentage is acceptable.
- For packages with the same number of power/GND pairs per bank, these calculations can be simplified to (WASSO of the left bank + WASSO of the right bank)/2.
- If WASSO utilization of two adjacent banks exceeds 105% of the SSO allowance, apply ground bounce reduction techniques to the two banks in question. It is desirable to maintain a similar level of WASSO utilization between any two adjacent banks.

Step 3: Calculate the package WASSO and ensure it does not exceed the SSO allowance.

- Package WASSO utilization
  = ((WASSO of bank 0 × number of power/GND pairs in bank 0) + (WASSO of bank 1 × number of power/GND pairs in bank 1) + … + (WASSO of bank n × number of power/GND pairs in bank n)) / (total number of power/GND pairs in banks 0 through n)
- For this example, Package WASSO Utilization =
  ((97 × 13) + (45 × 13) + (50 × 13) + (60 × 13) + (60 × 13) + (35 × 13) + (40 × 13) + (15 × 13))% / (13 + 13 + 13 + 13 + 13 + 13 + 13 + 13)
  = 50.3%
  Since 50.3% is less than 75.4%, this percentage is acceptable.
- For packages with the same number of power/GND pairs per bank, the calculation above simplifies down to (Sum of WASSO from all banks)/(number of banks available in the package).
- If package WASSO utilization exceeds the SSO allowance, apply ground bounce reduction techniques to reduce WASSO.

WASSO Calculator

A Microsoft Excel-based spreadsheet entitled "WASSO Calculator" is provided to automate these calculations. The WASSO calculator uses PCB geometry, such as board thickness, via diameter, and breakout trace width and length, to determine board inductance. It determines the smallest undershoot and logic-low threshold voltage among all input devices, calculates the average output capacitance, and determines the SSO allowance by taking into account all of the board-level design parameters mentioned in this document. In addition, the WASSO calculator performs checks to ensuring the overall design does not exceed the SSO allowance. This calculator can be downloaded from the following web site:

http://www.xilinx.com/bvdocs/appnotes/xapp689.zip

Differentiating Ground Bounce From Other Signal Integrity Issues

The issues discussed and the calculations presented in this application note apply specifically to managing the amount of ground bounce generated by the FPGA device alone. The intent of the WASSO method is to limit the amount of ground bounce present immediately at the output of the FPGA and not corrupt the operation of the device(s) the FPGA drives. WASSO assumes proper PCB decoupling along with proper termination of signals between the FPGA and external components.

Use following debug steps to determine whether the FPGA is producing too much ground bounce or if there are other signal-integrity related issues:

Step 1. Configure the FPGA for Spyhole0 (output logic Low) and Spyhole1 (output logic High) outputs using strong/fast drivers (typically, LVCMOS 24mA Fast) in the I/O bank where ground bounce problem is suspected. If possible, leave pins physically adjacent to Spyhole outputs undriven to minimize the effect of pin-to-pin coupling and crosstalk.

Step 2. Make measurements of Spyhole0 to PCB ground (ground bounce) and Spyhole1 to PCB ground (V_{CC} bounce) at the output driver right on the back-side of the FPGA. If necessary, remove the solder mask on the back-side via. Make sure to connect the oscilloscope probe's ground to a ground via near the Spyhole outputs through a very short ground clip - typically less than 0.25 inches in length.

Step 3. Repeat the measurement at the input device (at the far end of the trace, away from the FPGA) and compare the two sets of measurements. The possible scenarios are as follows:

Identify Improper Termination of Signals

To diagnose this issue, a spyhole must be implemented on a signal trace connecting the output driver of the FPGA to the input device.

A. If the amount of noise V_{PP} (peak-to-peak voltage) on Spyhole0 gets worse at the far end (typically1.5x to 2x), there is a reflection problem, not a ground bounce problem. The traces in question need to be terminated properly.

B. If the amount of noise V_{PP} on Spyhole1 gets worse at the far end (typically 1.5x to 2x), there is a reflection problem, not a V_{CC} bounce problem. The traces in question need to be terminated properly.

Identify PDS Design Issues

C. If the amount of noise V_{PP} on Spyhole1 is much greater than the amount of noise on Spyhole0 (both measured at the back-side of the FPGA), do the following:

   a. Double-check the scope measurement technique. Make sure the scope probe has a very short ground clip and is properly connected to a ground via near the output driver being measured. A long ground wire (typically two inches or longer) will yield erroneously high readings

   b. Either the oscilloscope itself and/or the scope probe must be properly terminated, typically with 50Ω.

c. Use the clock used to toggle outputs as a scope trigger to synchronize the V_{CC} bounce measurement.

d. Verify that all inputs to the FPGA must not exceed the overshoot specification of the device. Excessive input overshoot will cause more V_{CC} bounce at the output of the FPGA.

e. Refer to XAPP623. This application note covers Power Distribution Systems. Design a different decoupling capacitor network, one that is more effective at suppressing digital logic switching noise at the operating frequency of the circuit. On a properly decoupled PCB, the peak-to-peak noise on Spyhole1 should roughly equal the peak-to-peak noise on Spyhole0 when measured at the operating frequency (as triggered by the clock driving the output).

Identify Ground Bounce Issues

D. If the Spyhole0 (ground bounce) measurement on the back-side of the FPGA exceeds the number computed in cell B16 of the WASSO calculator, then do the following:

a. Double-check the scope measurement technique. Make sure the scope probe has a very short ground clip and is properly connected to a ground via near the output driver being measured. A long ground wire (typically two inches or longer) will yield erroneously high readings

b. Either the oscilloscope itself and/or the scope probe must be properly terminated, typically with 50Ω.

c. Use the clock used to toggle outputs as a scope trigger to synchronize the ground bounce measurement.

d. Verify that all inputs to the FPGA must not exceed the undershoot specification of the device. Excessive input undershoot will cause more ground bounce at the output of the FPGA.

e. Verify entry in the WASSO Calculator for the following items:

- PCB parameters in cell B2 through B7
- the number of power/GND pairs per bank (row 30)
- the number of drivers (SSOs) allowed per power/GND pair for each driver type (rows 33, 36, 39, 42, and 45)
- the number of drivers used for each driver type (rows 32, 35, 38, 41, and 44)
- the number of drivers with similar output load profile (row 20)
- the number of loads per driver with similar loads (row 21)
- value of input capacitance per load (row 22)

Step 4. If steps 1 through 3 (particularly 3D) do not resolve the issue, and the input device does not operate correctly as a result of excessive ground bounce presented at Spyhole0 when measured on the back-side of the FPGA, open a WebCase with Xilinx Technical Support at http://www.xilinx.com/support/clearexpress/websupport.htm.

Tips on Effectively Managing Ground Bounce

Ground bounce must be managed to ensure proper operation of high-performance devices. Particular attention must be applied to minimizing board-level inductance during PCB layout. This is achieved by minimizing board thickness and break-out trace length while maximizing via diameter and break-out trace width. During the FPGA design phase, plan FPGA pin-out assignments to spread out fast/strong drivers across multiple I/O banks (all if possible), with the goal of achieving a similar WASSO utilization level across all banks. When possible, avoid using devices with low input undershoot threshold and/or high input capacitance loading. Also, choose FPGA devices and packages with high power/GND pair to I/O pin ratios.

Conclusion

The "WASSO Calculator" (Excel spreadsheet) is provided to ensure designs meet the input undershoot and logic-low voltage requirements of devices receiving signals from FPGAs. To control ground bounce, manage PCB design parameters and FPGA pin-out assignment, and choose driver strength and slew-rate appropriately.

Revision History

The following table shows the revision history for this document.

Date	Version	Revision
09/16/03	1.0	Initial Xilinx release.
12/08/04	1.1	New sections, formulas, and examples added.
10/30/07	1.2	Expanded to include Spartan-3A generation devices.

Notice of Disclaimer

Xilinx is disclosing this Application Note to you "AS-IS" with no warranty of any kind. This Application Note is one possible implementation of this feature, application, or standard, and is subject to change without further notice from Xilinx. You are responsible for obtaining any rights you may require in connection with your use or implementation of this Application Note. XILINX MAKES NO REPRESENTATIONS OR WARRANTIES, WHETHER EXPRESS OR IMPLIED, STATUTORY OR OTHERWISE, INCLUDING, WITHOUT LIMITATION, IMPLIED WARRANTIES OF MERCHANTABILITY, NONINFRINGEMENT, OR FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT WILL XILINX BE LIABLE FOR ANY LOSS OF DATA, LOST PROFITS, OR FOR ANY SPECIAL, INCIDENTAL, CONSEQUENTIAL, OR INDIRECT DAMAGES ARISING FROM YOUR USE OF THIS APPLICATION NOTE.