Altera XML PSDF Delay File
Detailed Design

Version 1.2

May 27, 2005

By

Altera Corporation

Table of Contents

1. Overview

The Altera® XML PSDF (Physical Synthesis Data Format) Delay File is a communications standard for describing cell-to-cell delay values to EDA tool vendors in a concise, accurate and digestible format. This document is intended for external use. This document describes in the format of the Altera XML PSDF Delay document and fills in any gaps in the format the XML Schema does not explicitly cover. It is intended as a guide for both the future maintenance of the document specification and for the third party programmer attempting use the information contained in an XML file that meets the Altera XML PSDF Delay File specification.

XML format is a flexible, platform-independent method for sharing information. The Altera XML PSDF Delay File further constrains this format to provide a flexible and understandable method for sharing Altera architecture delay information. The standard is set forth in an XML Schema and files validated against this schema are said to be valid Altera XML PSDF Delay Files. A myriad of XML parsers are available for most languages to allow quick and easy digestion of the information in the Altera XML PSDF Delay File.

2. Executive Summary

This document is to be used in conjunction with the document *Altera_XML_Architecture_Description_Detailed_Design.doc* and *Altera_XML_Point_To_Point_Delay_File_Detailed_Design.doc*. This document describes the delay values from cell to cell after the place and route flow in the Quartus® II software. It applies to the Quartus II 2.2 release and beyond.

This document has the following purposes:

- Describe the file formats used to describe cell-to-cell delays in Altera devices
- Explain each part of the file
- Give various XML links

This file does not:

- Give detailed architecture information like block information and layout
- Give detailed intra-cell delay information
- Give an overview of why this information is relevant

After reading this file and referenced material you should be able to do the following:

- Understand the raw format of an Altera XML PSDF Delay File

3. Glossary

XML Document – A file based on the W3C's eXtensible Markup Language format. All XML documents described in this detailed design document conform to the W3C XML 1.0 standard. For more information on XML see: http://www.xml.com/axml/testaxml.htm

XML Schema – Templates for XML documents; they allow you to specify valid arrangements of elements and attributes within an XML document and impose restrictions on their contents. An XML document that is valid according to an XML schema does not break any of the rules laid down in the XML schema.

Element – An XML container formed with a start tag (<START_TAG>) and an end tag (</END_TAG>). The start tag can contain some number of attributes (<START_TAG att1=val1 att2=val2>) according the schema for the document. Data that falls between the start and end tags, be it additional elements or text, is said to contained by the element. This allows for a parent-child tree structure in XML data to be formed. If an element contains no children the start/end tag notation can be abbreviated in to a form known as an empty tag (<START_TAG att1=val1 att2=val2 />).

Block – The elements upon which a chip is built. They are hierarchical representations of basic pieces of the architecture. For example: a logic element block, or LE; a LAB block, which contains a number of LE blocks.

Architecture – A top-level container for devices. The architecture describes common blocks that are then instantiated in devices of various sizes and configurations.

Device – A group of blocks instantiated in a particular configuration.

4. The Altera XML PSDF Delay Schema

The Altera XML PSDF Delay schema describes the format for any XML-based inter-cell delay description. A tree-view of this schema is seen in Figure 4-1. For more information on each specific element in the diagram please see Section 5.

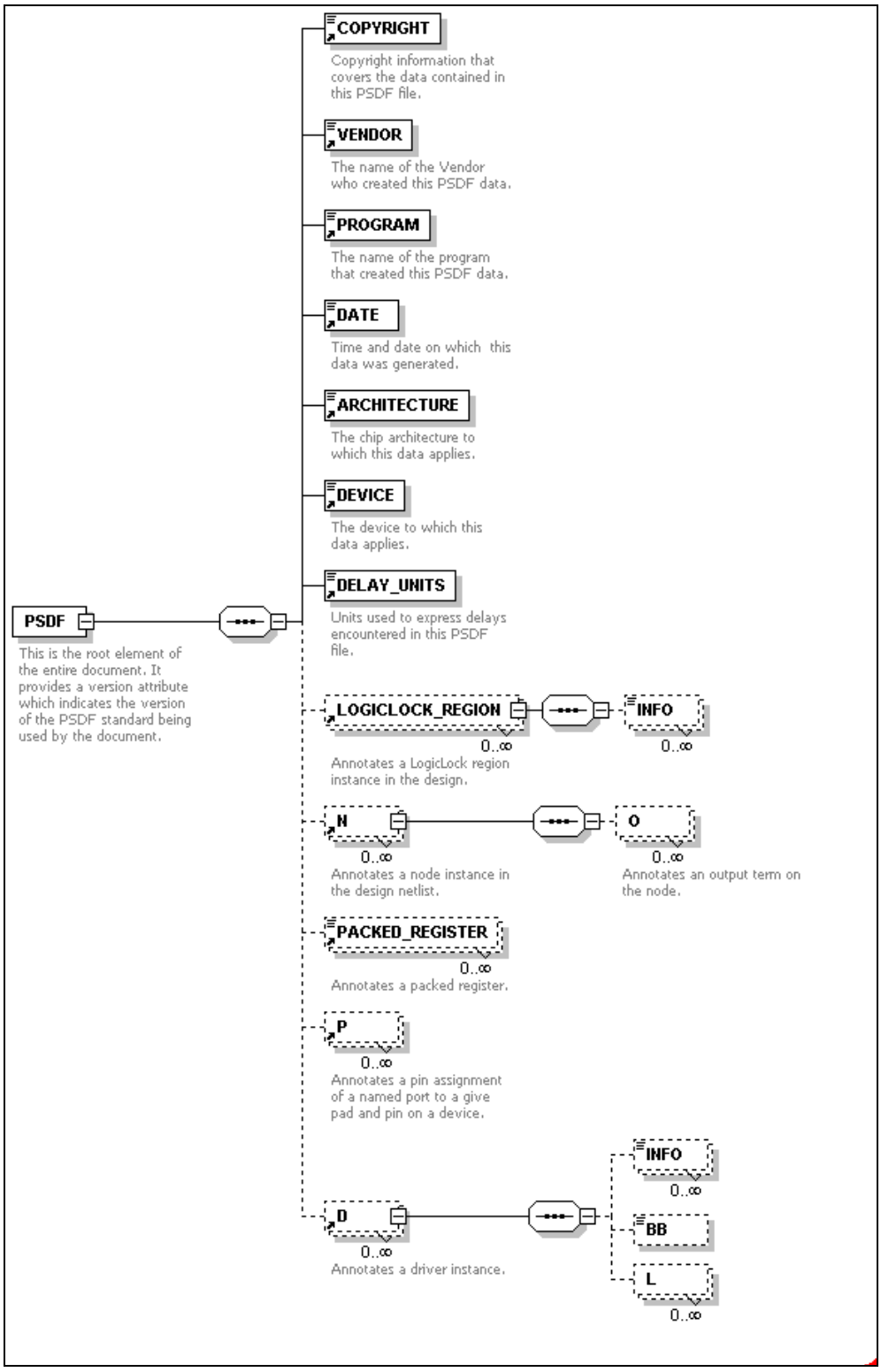

Figure 4-1: Tree View of the Altera XML PSDF Delay Schema

The top-level container, or root element, is the <PSDF> which is a container element, used to hold information about a specific design compilation for a specific device.

The dashed lines/boxes in **Error! Reference source not found.** represent optional elements. The number of times an element can occur is listed directly below the element boxes. Order is important in XML. For example, you can have from one to an infinite number of <N> elements inside the <PSDF> element, but they must all occur before any <P> element in the <PSDF> element. Once you see an <P> element, based on this schema, you can assume you will never see an <N> element again for this <PSDF>.

Solid line/boxes represent elements that must occur at least once. According to **Error! Reference source not found.** an <PSDF> element must have at least one <VENDOR> element. The <D> elements however can have any number of the child elements shown, including zero.

5. Detailed Altera XML PSDF Delay File Element Information

This section has detailed information for every possible element that can exist, according to the schema, in an Altera XML PSDF Delay File. It can be used as a reference when creating as well as parsing an XML file that is validated against this schema. A list of elements and their attributes can be found in Table 1. Detailed descriptions of each element can be found in the subsections below or by clicking on the Element tag in the table below.

Table 1: Element and attributes found in an Altera XML PSDF Delay File

Element	Attribute	Type	Use	Description
<ARCHITECTURE>	*no attributes*			
<BB>	*no attributes*			
<COPYRIGHT>	*no attributes*			
<D>	id	integer	required	The integer node ID of this driver.
	pin	string	required	The name of the pin to which this driver is mapped.
	location	string	required	The string representation of this driver's location in the device.
	ipvector	string	optional	A vector that represents any LUT input rotation that occurred during placement or routing. For more information on the format of this vector see section 5.4.
<DATE>	*no attributes*			
<DELAY_UNITS>	*no attributes*			
<DEVICE>	*no attributes*			
<INFO>	name	string	required	The string key for this information.

<L>	id	integer	required	The integer node ID of this load.
	pin	string	required	The name of the pin to which this load is mapped.
	delay	integer	required	The amount of delay from the driver to this load.
	slack	float	optional	The slack ratio of the path from the driver to this load. Available only when there are timing constraints on the design.
<LOGICLOCK_REGION>	Id	integer	required	The integer ID of this region.
<N>	id	integer	required	The integer ID of this node instance.
	name	string	required	The string name of this node instance.
	type	string	required	The type of block this node represents on the device.
<O>	id	integer	required	The integer ID of the parent node instance.
	name	string	required	The string name of this node instance.
	type	string	required	The type of block this node represents on the device.
	pin	string	required	The pin name on the wysiwyg parent this oterm instances map on to.
<P>	name	string	required	The name of the I/O node.
	pin	string	required	The name of the pin associated with this I/O node.
	location	string	required	The string representation of the I/O node's location.
<PACKED_REGISTER>	name	string	required	The name of the packed register node.
<PROGRAM>	version	string	required	The version string of the program.
<PSDF>	version	string	required	The version of the PSDF information generator.
<VENDOR>	*no attributes*			

5.1 <ARCHITECTURE>

The <ARCHITECTURE> element holds a string that represents the device architecture used by the design.

5.2 <BB>

The <BB> element is used to indicate that a driver is inside a black box and gives the block box name.

5.3 <COPYRIGHT>

Legal permission and usage statements for the information contained in the XML PSDF Delay file.

5.4 <D>

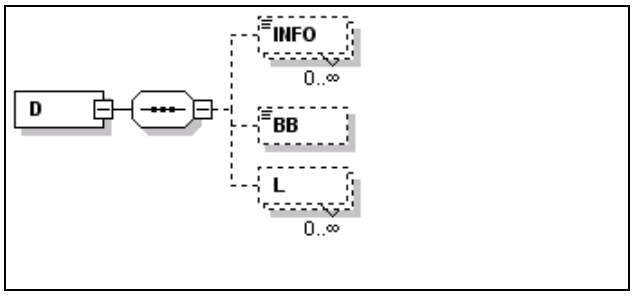

Figure 5-1: The <D> element and its children

A driver instance. The <D> element attributes provide information about the driver, including the ID number of the node this driver belongs to, the name of the pin to which this driver is associated, the location of the driver on the device and a vector that represents any LUT input rotation that this driver underwent during the place and route flow.

The 4-element ipvector attribute that represents any LUT input rotation that occurred during place and route has the regular expression format: {\d,\d,\d,\d}. An example of this vector would be {2,4,1,3} which indicates that LUT inputs D1, D2, D3 and D4 have been remapped {D2, D4, D1, D4} respectively. If no rotation has occurred the vector has zeros to indicate this: {0,0,0,0}.

A <D> element may possibly contain several <INFO> elements that provide additional information about the driver. If the <D> element represents a driver that is part of a black box the <BB> element will appear as a child of the <D> element with the name of the black box to which this driver belongs. Finally all loads this driver drives are written as <L> elements.

5.5 <DATE>

The <DATE> element contains the time and date when the XML PSDF document was generated. The format is ISO 8061 extended format. An example would be:

<DATE>1978-01-12T11:00:00</DATE>

This represents the date and time: January 12th, 1978, 11:00:00am.

5.6 <DELAY_UNITS>

The unit of time for all delay values in the file.

5.7 <DEVICE>

The device used by the design.

5.8 <INFO>

The <INFO> element allows for simple (name, value) paired information to be expressed. It provides additional information about its parent or containing element that is possibly non-essential, but perhaps useful. An example of a common <INFO> element for the <D> element is:

<D>
<INFO name="mode">arithmetic</INFO>
</D>

This <INFO> element tells us that the parent <D> element is in a 'mode' called 'arithmetic'.

5.9 <L>

The <L> element represents a load being driven by a <D> element. Each <L> element provides useful information about the load including: the ID of the node to which this load belongs, the name of the pin on the node to which this load is mapped, the delay from the <D> element to the load (the units of the delay are defined by the <DELAY_UNITS> element), and the slack of this delay path (available only when there are timing requirements on the design). For more information on obtaining slack numbers in the PSDF see Section 6.1.

5.10 <LOGICLOCK_REGION>

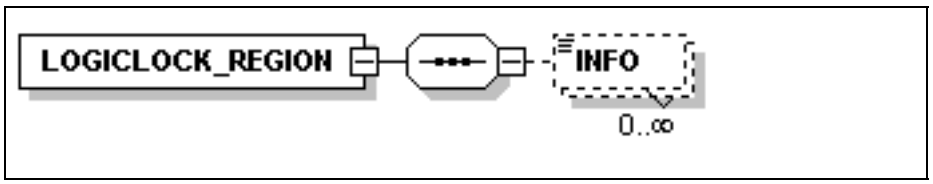

Figure 5-2: The <LOGICLOCK_REGION> element and its children

The <LOGICLOCK_ELEMENT> element provides information for a single LogicLock region in the design. The information provided, in the form of key/value pair <INFO> elements, is limited to

information that may have been changed by the fitting process. For example the width and height of the region, the origin of the region and the state of the region are reported.

5.11 <N>

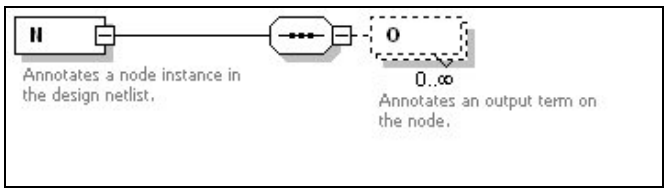

Figure 5-3: The <N> element and its children

The <N> element represents a node instance in the design. The element's attributes provide a unique node ID number for the instance, the name of the node and the type of the node. The node ID number is referenced by other elements in the file to indicate that there are ports on this node.

5.12 <O>

The <O> element represents an output term instance in the design. The element associates possible output terms with each instance in the design. The integer ID refers to the instance that the output term is a child of. The statement also provides the name of the entity for the output term, but the main purpose is to identify possible buried instances in the design as a result of register packing.

5.13 <P>

The <P> element represents a port instance in the design. The element's attributes provide a name for the port, a pin name to which this port is bound and a string that represents the physical location of the port using the Cartesian co-ordinate system for the device as described in the Altera_XML_Architecture_Description_File_Detailed_Design.doc document.

5.14 <PACKED_REGISTER>

The <PACKED_REGISTER> element describes a register packed into an I/O (or <P> element). The id attribute provides the ID of the packed register and the element contains text that represents the name of the <P> element into which this register has been packed. For example:

```
<P name="gclk" pin="R_1" location="LC_X0_Y0_N0"/>
…
<PACKED_REGISTER id="reg001">gclk</PACKED_REGISTER>
```

This would indicate that the register named 'reg001' had been packed with the I/O named 'gclk' at location (0,0,0).

5.15 <PROGRAM>

The <PROGRAM> element provides information about the software that generated the PSDF file being viewed. Including the name of the software and a string representing the version of the software.

5.16 <PSDF>

This is the root element of the entire document. It provides a *version* attribute which indicates the version of the PSDF standard being used by the document.

5.17 <VENDOR>

The <VENDOR> element contains a string that is the name of the vendor who created the PSDF file.

6. Generating an XML PSDF File with the Quartus II 2.2 Software

To generate an XML PSDF file for a design during compilation take the following steps:

1. From within the GUI select **Project -> Settings -> EDA Tool Settings…** from the toolbar. You will see a pop-up window like Figure 6-1. For the **EDA Resynthesis Tool type** select ‘Amplify’ as the **Tool name** and click the **OK** button.
2. Recompile your design and the XML PSDF file we be created at the end of the place and route in flow as <design directory>/resynthesis/amplify/<design name>-psdf.xml.

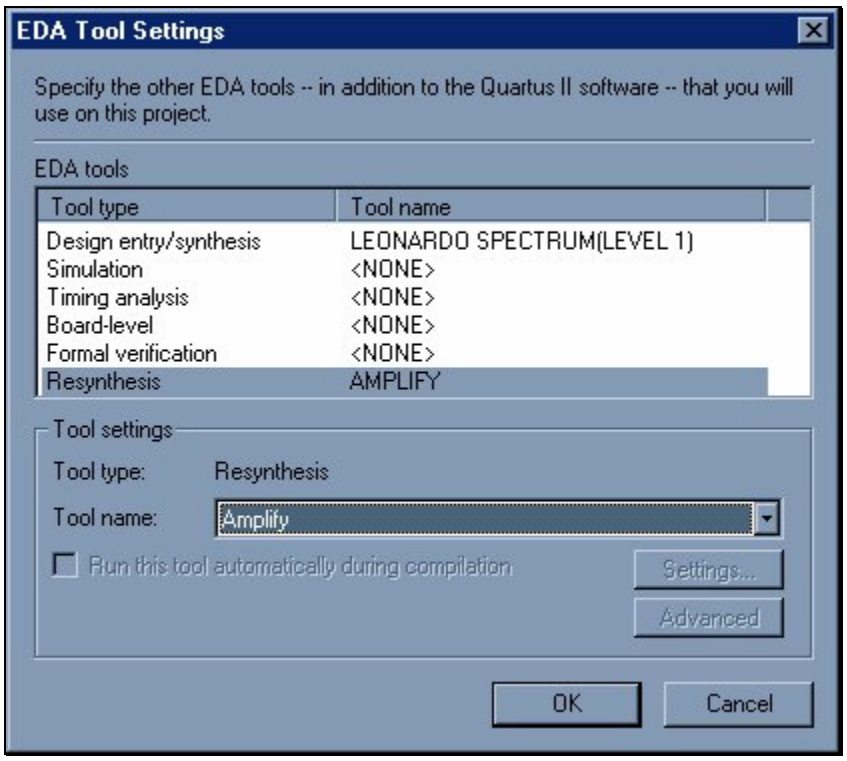

Figure 6-1: The Quartus II 2.2 EDA Tool Settings window

6.1 Writing slack information in the PSDF file

Slack information for each [Driver, Load] pair can be included in the PSDF using either a ini variable or a PSF setting. This information is meant for debugging purposes only and is provided to help ease the process of correlating timing information in Quartus and Third Party tools. To turn on slack information you may either add the following to your Quartus ini file:

```
psyn_write_slacks_to_psdf=on
```

Or use the Quartus Tcl interface to turn on resynthesis extra debugging information in the PSF project setting file for given project. Add the following to your project Tcl script to accomplish this:

```
project add_assignment “” “eda_design_synthesis” “” “” \
“RESYNTHESIS_EXTRA_DEBUGGING_INFORMATION” “ON”
```

Sack numbers will be printed for each load and represent the worst case slack between the load and its driver. Where slack is not available and incredibly large integer will be written instead of some sensible value.

7. Additional References

http://www.w3.org/XML/ -- The World Wide Web Consortium is in charge of the XML standard. This is a good place to start for general standard information on XML, XSLT, XML Schema, DOM and other XML-related standards.

http://www.xml.com/ -- The O'Reilly Group's XML resource site. Lots of tutorials and examples are available under the 'Programming' section. There is also a great annotated version of the XML 1.0 specification, complete with in-line notes and pointers, at http://www.xml.com/axml/testaxml.com.

http://xml.apache.org/ -- Home of the Apache Group's Xerces XML parser. Their goal is "to provide commercial-quality standards-based XML solutions" within the open source framework. In addition to Xerces for C++, Java, Perl and COM you will Xalan, an XSLT stylesheet processor for transforming XML in Java and C++.

CodeNotes for XML, Edited by Gregory Brill – An excellent and invaluable reference for any developer new to XML.

The **Altera XML Architecture Description File Detailed Design document** (Altera_XML_Architecture_Description_File_Detailed_Design.doc) describes in detail how blocks are assembled and built in to devices. It is a worth at least glancing at if you are trying to understand the fine details of intra-block delays described in this document. It also contains a detailed description of the Cartesian co-ordinate system used for describing node locations.

The **Altera XML Point to Point Delay File Detailed Design document** (Altera_XML_Point_To_Point_Delay_File_Detailed_Design.doc) describes in detail the format used to express intra-cell delays for each Altera device. It is a necessary read for anyone wishing to construct complete register-to-register delay paths.