



1Team®-Genesis IO

Automated I/O Generation



Today's complex SoC's include more peripheral interfaces in the core than can be accessed at one time. This requires a complex, yet flexible muxing strategy that allows configuration pins or software running on the system to control which interfaces are accessible at any given time. Consequently, I/O subsystem designers spend a lot of time designing these complex I/O subsystems and then adapting them to inevitable specification changes during the course of a design. 1Team®-Genesis IO provides a platform to address these issues and seamlessly integrate the I/O fabric with the rest of the SoC.

The Problem

High-functionality SoCs such as cell-phone systems and catalog components targeted to multiple sockets share a common problem – the need to flexibly adjust connectivity between package pins and core IP signals through software control and/or configuration pin settings. This demands complex multiplexing between the core and I/O pads, logic to control mux selection as well as control logic for pull control, power control and many other signals. For such cases, it is not uncommon for the I/O logic RTL to run into tens of thousands of lines. This complexity is further compounded by boundary scan, power and voltage-domain logic and the need to partition I/O logic for optimal physical layout. When planning I/O subsystem design, the designer must consider:

- Which pad cell should be used for each pad and how these will map both to bond-pads on the die and to balls on the BGA package
- The pin-muxing strategy between the buffers and core interface pins
- Which signal groups need not be simultaneously accessible and ensure that "like" signal types are grouped, so that clock signals are not grouped with non-clock signals
- Whether or not to insert boundary scan into these paths. While some design flows use existing ATPG tools to accomplish boundary-scan insertion, that approach will not necessarily suit all design flows
- Strategies to hook up all the many additional control signals required by I/O-buffer cells and boundary scan cells. Some of these signals may be direct connections to the core, some may also need to be muxed. Strategies to partition the generated logic for power management and for optimal physical design must also be considered

When a package can have 400 pins and core interfaces may define well in excess of 1000 pins, automation clearly becomes essential. The objective of this automation is to capture the I/O specification in as reduced a form as possible, consistent with maintaining flexibility, and to be able to produce new netlists, BGA diagrams and other required outputs at the push of a button. This greatly simplifies updates to the I/O strategy, from early definition all the way through to late-stage tweaks.

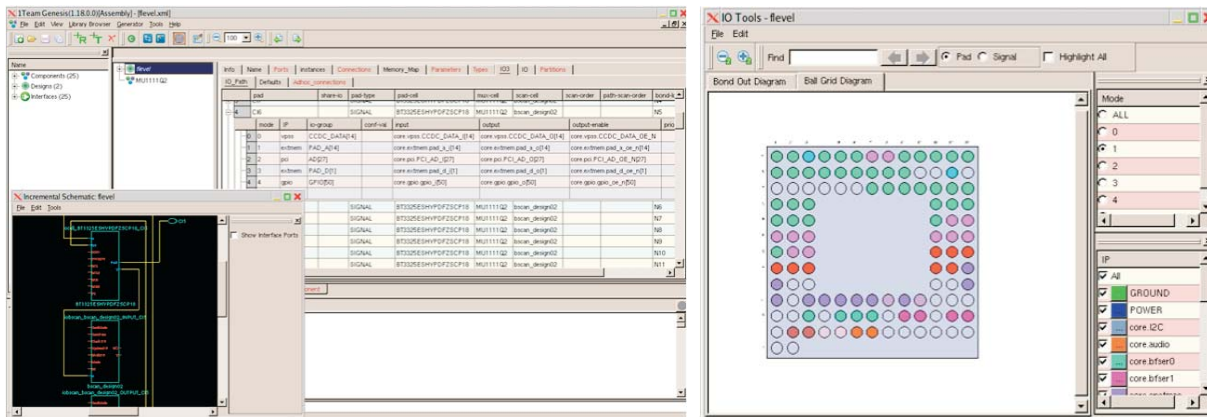
The Atrenta Solution – 1Team-Genesis IO

- Easily captures and allows review of the I/O specification through interactive entry or tcl commands; Complete SoC I/O layer specification for I/O-cell buffers, configuration control, pin multiplexing, boundary scan and package definition parameters
- Provides per-path definable mux-cells and muxing priority to manage timing-critical paths – supports input muxing, input deselect muxing and output muxing in both function and test paths

ATRENTA®, **Right** from the Start!

The Atrenta Solution – 1Team-Genesis IO (Contd.)

- Automatically generates select logic driven by configuration pins and/or core signals
 - Supports user-defined sets of muxed control signals (pull-enable for example) and non-muxed signals (pull-up/down for example)
- Generates bond pad and ball-grid diagrams to enable visualization of pad and ball layout
- Optionally inserts boundary scan cells and stitches the scan chain, under user control. Supports multiple cores for flexibility in combining subsystems under a common I/O fabric
- Provides user-guided netlist partitioning to control partitioning of subsystems within the generated logic
- Supports analog and differential paths
- Unique incremental schematic features enable review and debug of the logic cone associated with a single device pin
- Automatically generates I/O fabric RTL in either Verilog or VHDL
- Minimizes opportunities for human error through real-time consistency checking
- Automatically generates assertions and a testbench to validate the generated RTL
- Enables cell library model definition in Tcl or its import from RTL or IP-XACT and augmentation with pin-type information through the GUI or Tcl



The Atrenta Difference

- Proven I/O fabric generation technology that can handle a wide range of I/O architectures, yet reduces the capture task to a common table
- Complete platform for specifying I/O architecture including functional, test and analog paths, pin multiplexing and validation
- Accelerates turn-around time for late changes to the I/O architecture
- Accelerates netlist generation and documentation interfaces as a result of tight integration with the 1Team-Genesis Assembly and Registers solutions



CONTACT:
1.408.453.3333
 moreinfo@atrenta.com
 www.atrenta.com

Atrenta is the leading provider of Early Design Closure[®] solutions that radically improve the efficiency of integrated circuit design. Using Atrenta's comprehensive tool suite for architectural chip assembly and RTL analysis, customers can create robust and correct designs rapidly, preventing expensive and tedious iterations during the implementation phase. With over 150 customers worldwide, including the world's top 10 semiconductor companies, Atrenta provides the most comprehensive solution in the industry for Early Design Closure. Atrenta, Right from the Start!

© Copyright 2009 Atrenta Inc. All rights reserved. Atrenta, the Atrenta logo, SpyGlass, 1Team, and Early Design Closure are registered trademarks of Atrenta Inc. All others are the property of their respective holders.

