

PARAGON: A SOFTWARE ARCHITECTURE AND IMPLEMENTATION FOR MIXED-SIGNAL MODEL DESIGN



Matt Francis, afranci@uark.edu
 Faculty Advisor: Alan Mantooth

ABSTRACT The intent of the research undergone is the design and implementation of a software architecture for a graphical environment for the creation, testing and validation of HDL-based, mixed-signal models.

BACKGROUND The popularity of integrating digital and analog components (mixed-signal electronics) on the same chip (System on a Chip, SOC) in recent years has revealed the difficulty associated with simulation of such circuits. The current trend involves creating “models” for each chip through use of any one of several Hardware Description Languages (HDLs), previously used for the simulation of digital circuits. While this procedure has been successful, it has several drawbacks, including a lack of portability and a requirement of a programming mindset not necessarily desired by analog circuit designers.

WORK DONE The architecture designed consists of three essential components: A Database to store a non-HDL specific form of the model, “Editors” to facilitate model entry and modification, and various “Tools” to validate design assumptions. Each interface (editor or tool) is based on shared “Technology Managers” which provide the base functionality for each interface and control interface communication.

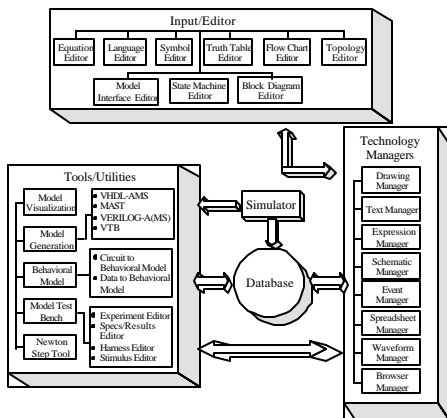


Figure 1. The Paragon Architecture

All of these components are used through a main “Composer” interface, where the model is entered and validated. When the model is deemed functional, the database is written and models in various HDLs (VHDL-AMS, Mast, etc) can be produced. A scripting language with extensions for Graphical User Interfaces (GUIs) was chosen to aid in rapid development of all user interfaces, and allows the environment to easily execute on various computing platforms. The database format is the XML (eXtensible Markup Language) format, chosen for its ease of portability.

RESULTS The resulting tool has proven quite successful in generating models. Ease of use has been extraordinary and generated code performs well in several simulation environments. A public release of the tool is pending shortly.

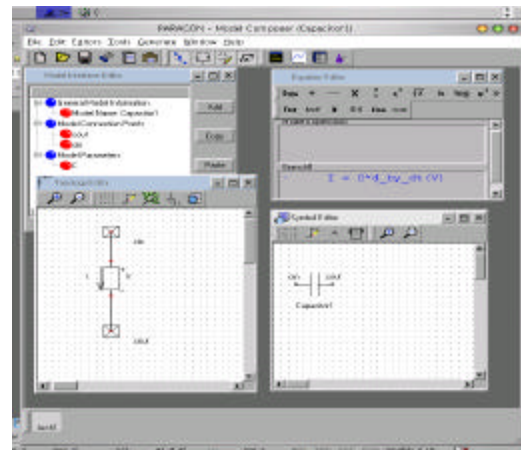


Figure 2. Paragon Application

FUTURE WORK The next challenge facing the Paragon project is the addition of digital design tools to the environment. This work is currently in its infancy, but hopes to provide such useful tools as a finite mixed-signal state machine editor and other enhancements which will make paragon a truly “mixed-signal” environment.

