

EDA Tech Forum

Volume 2 Issue 2 June 2005

TESTED COMPONENT TO SYSTEM

DESIGN TO SILICON

ESL / SYSTEM C

VERIFIED RTL TO GATES

DIGITAL / ANALOG IMPLEMENTATION

INSIDE

- > Making Network-on-Chip happen
- > When Classical Scaling attacks
- > The SystemVerilog revolution
- > Making DFM user-friendly
- > Highlights at DAC 2005

**First Anniversary
Issue!**

contents

< COMMENTARY >

5

Strategy begets solutions

We do not have all the answers to today's technology challenges but we do know how to get them.

6

Destination network-on-chip

A University of British Columbia team outlines some of its research results on the progression from SoC to NoC.

8

The whole picture

SiGe pioneer and DAC keynoter Bernard Meyerson says Moore's Law has been misunderstood and that a new, holistic approach to design is needed to overcome the discontinuity.

10

SystemVerilog is changing everything

Accellera secretary Karen Bartleson outlines the benefits of the first hardware design and verification language.

12

Innovate in volume

And it isn't easy, even for Texas Instruments, as Hans Stork, its CTO, explains.

42

I, User

The ultimate wish-list

The clearest indicator of what engineers want from EDA is the annual DAC line-up. What does this month's conference highlight as 2005's priorities.

< TECH FORUM >

14

ESL/System C

Micro-architecture exploration for deep submicron design

Alcatel Space's Louis Baguena describes how synthesizing directly from C cut design cycle times.

18

Verified RTL to gates

Formal verification poised for rapid growth

Pranav Ashar and Rich Faris of Real Intent explain how static assertion-based verification is entering the mainstream.

22

Digital/analog implementation

Improving team productivity with efficient data management for SoC designs

Scott Humphreys of RF Micro Devices and Srinath Anantharaman of ClioSoft outline how you can use DM to smooth the design flow.

28

Design to silicon

Achieving better DFM: EDA tools pave the way to improved yield

Mentor Graphics' Joseph Sawicki sets out how EDA tools are aligning the traditional world of design rules with design for manufacture.

34

Design to silicon

A methodology of integrated post tape-out flow for fast design to mask TAT

SMIC's Chi-Yuan Hung explains how the foundry secured a 5X improvement in turn-around-time.

38

Tested component to system

Power analysis for efficient power planning

Mouzam Kahn of Altera highlights the importance of power analysis in PCB design.

eight



ten



twelve



fourteen



forty two



EDA Tech Forum
Volume 2, Issue 2
June 2005

Editor in Chief
Paul Dempsey
t +1 703 536 1609
edatechforum@cox.net

Publisher
Bill Reis
breis@cmp.com

Publication Manager
Angela Duarte
aduarte@cmp.com

Ad Sales
Richard Carlisle
t +1 541 318 0616
rcarlisl@cmp.com

Art Direction and Design
Creative Services East
Diane Schaffer
dschaffe@cmp.com

Production
Bob Steigleider
rsteigle@cmp.com

Circulation
Shannon Aronson
saronson@cmp.com

VP Sales, Electronics Group
Stuart Bowen
sbowen@cmp.com

EDA Tech Forum is a trademark of Mentor Graphics Corporation, and is owned and published by Mentor Graphics. Rights in contributed works remain the copyright of the respective authors. Rights in the compilation are the copyright of Mentor Graphics Corporation. Publication of information about third party products and services does not constitute Mentor Graphics' approval, opinion, warranty, or endorsement thereof. Author's opinions are their own and may not reflect the opinion of Mentor Graphics.

Improving team productivity data management for SoC de

Scott Humphreys, *RF Micro Devices* and Srinath Anantharaman,



Scott Humphreys is an RFMD Fellow at RF Micro Devices in the company's digital cellular product line. He has a BSEE from Virginia Tech, Blacksburg, an MSEE from the National Technological University, Ft. Collins, and holds 24 patents.



Srinath Anantharaman is founder and CEO of ClioSoft and has over 20 years of experience in the electronic design automation industry. He is a graduate of the Indian Institute of Technology, Kanpur, and Washington University in St. Louis.

The efficient management and synchronization of design data is now a necessity and no longer something that is a 'nice-to-have'. Design complexity continues to increase with one result being that development spreads across several groups (often in different places). A project may also have several thousands design files generated by various tools from different vendors at multiple stages.

As SoC design is a team effort, some of the biggest productivity gains can come from defining processes and deploying tools that allow the team to communicate and collaborate efficiently. Moreover, effective data management (DM) not only helps define productivity but also the quality of the end product.

Streamlining the data flow

As projects evolve, more engineers in different time zones using various tools tend to be added. Ad hoc processes evolve as this dispersed cadre tackles its need to constantly communicate and collaborate. However, while informal processes may solve the immediate problem they are drafted to address, they come under serious strain or fall down as the project grows and its priorities and requirements change. Typical hazards are:

- Delays in time-to-market.
- Design errors.

Today's designs generate huge volumes of data. Managing this flow of information is a vital task that needs to be planned out before a design project begins. The authors share their experiences of planning this process with specific reference to AMS projects, although the lessons they have learned are more generally applicable.

- Decreased predictability, control and visibility.
- Logistical nightmares for CAD support.

Project managers need processes, tools and infrastructure that meet DM requirements throughout the lifespan of the work.

You can think of a DM infrastructure as a 'data bus' that allows each engineer to easily plug-and-work. For this to happen:

1. Engineers' access to this bus must be efficient. For multiple sites, the network link between them must have sufficient bandwidth, high

reliability and availability and the DM tool must support effective multi-site collaboration.

2. EDA and other software tools used should be integrated (or easy to integrate) with the DM system. Engineers should ideally perform DM operations within their design tools, and working at the appropriate abstraction level.

3. The processes and protocols that interact with the data bus should be simple, well-defined and well-documented. The process should meet the project's needs and be easy for engineers to learn and use everyday without undue burden.

Version control – data

As a rule of thumb, manage and place under version control any

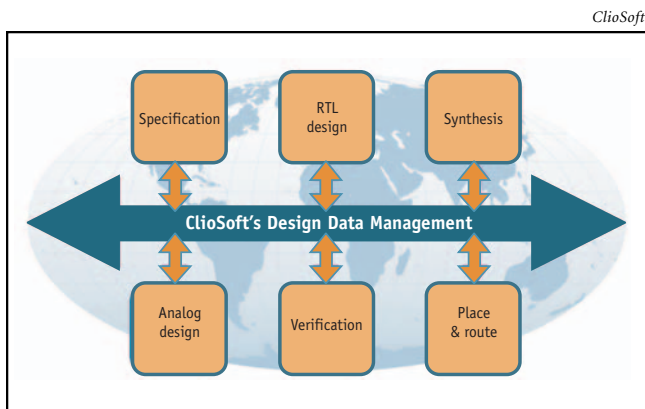


FIGURE 1 Streamline the SoC design Flow

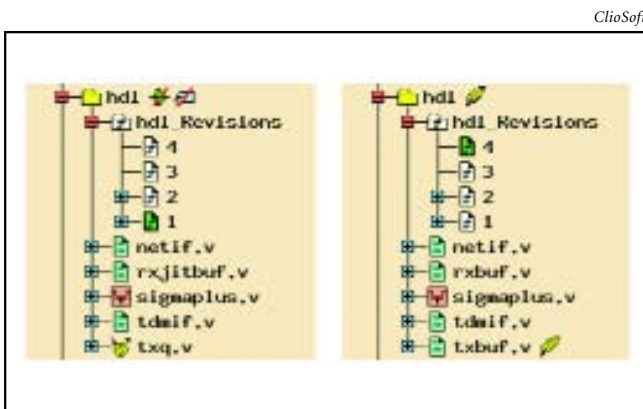


FIGURE 2 Two versions of a directory after files have been deleted, added, and renamed

with efficient designs

ClioSoft

data that users create. Examples include:

- Specifications and documentation.
- Design data files from RTL through GDSII.
- Makefiles and automation scripts.
- Stimulus vectors.
- Environment settings.

It is not as important to track every change to derived files such as simulation output or synthesis DB files. These are often very large and can be generated if the tools and source files are available. However, on some critical occasions, you may want to archive these derived objects in the DM system. Then, it is best to place them in a separate directory so they do not interfere with the normal tool runs that output the objects.

Version control – directories

Designers will modify source files during a project, but will also add files or directories, delete or rename files, and sometimes reorganize the entire project structure. It is thus equally important to version control changes made to the directory structure so that:

- Designers can change the structure in a way that the changes propagate seamlessly to other team members on demand.
- If there is a need to restore a previous time point or milestone, the directory structure will also be reproduced exactly and all scripts and automation will work as before.

Managing workareas

The user's workarea should be distinct and separate from the project repository where revisions of all files and directories are maintained. The project repository file system should be protected

with access granted only to the project administrator.

Some design teams use a common workarea where all or many engineers work. These may be suitable for small close-knit design teams but present significant challenges for larger ones. Data may be accidentally overwritten or deleted and the approach does not provide a stable environment to run system level tasks.

It is much better for each engineer to have a dedicated workarea. They can check out design files to work on and make necessary changes without affecting others, and then check the files back into the project repository when the work is completed.

Physical copies or symbolic links

Hardware design data files may vary in type and can be large. Although disk space is relatively inexpensive, managing heterogeneous data files and moving large files around can still create difficulties. It is therefore important to decide whether a workarea should contain physical copies of files or symbolic links.

Engineers working on a design's front end edit Verilog, VHDL, or C files. These are usually small text files, so a workarea with physical copies of them is most appropriate.

For large libraries with analog schematics or layout views, the better approach is to create a workarea where all directories are physical but the files are symbolic links to a shared area. When a file is checked out for editing, the link is broken and a physical copy is placed in the user's workarea. Engineers share the files except for those checked out for editing.

Smart symbolic links

ClioSoft's Smart Cache server provides the best features of both physical files and symbolic links. An engineer can create a workarea with symbolic links to a cache maintained by the server. The cache server maintains a copy of all revisions of files being used by all the workareas, and tracks how many symbolic links point to each version. Each engineer can create a workarea by specifying a rule to select revisions (called a revision search order). A revision is purged from the cache only when no links point to it.

Workareas with smart symbolic links have several advantages:

- The workarea is stable until the engineer chooses to update it.

>>> *continued on next page*

ClioSoft

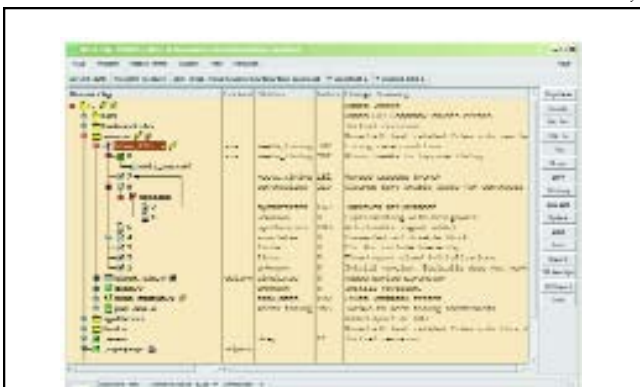


FIGURE 3 Collaboration icons always keep you aware of project status

ClioSoft

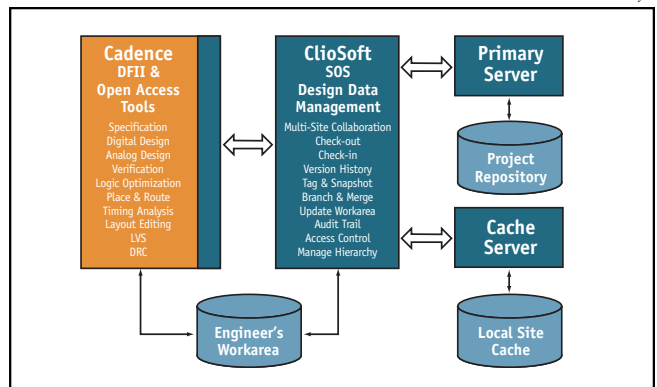


FIGURE 4 Overview of DM integration with Cadence IC Design Platform



FIGURE 5 ClioSoft's DM available directly from Mentor's ICstudio

- The use of disk space is optimized.
- Demand on network bandwidth for multi-site projects is minimized.
- Communication and collaboration are straightforward.

Avoid email overload

Clear communication is particularly important when teams across different time zones are collaborating on the same project and communication tends to be asynchronous.

Email is frequently seen as a universal panacea for communication needs. For instance, triggers are often set up to send emails when files are checked in. But, in any real project, such measures can lead to email overload, hampering productivity.

The team can instead set up systems that provide visibility on demand. Automated scripts and reports provide status and progress metrics when needed. Graphical displays from DM tools, such as ClioSoft's SOS, keep you continuously informed of status.

A simple but well-defined methodology

When sharing thousands of files, each engineer must get the right sets and versions. In an SoC design with multiple stages and groups it is impractical for everyone to be always accessing the latest file versions. Such a methodology will be very restrictive and discourage users from checking in changes for long periods of time. Instead a well defined but simple version tagging and promotion methodology is required.

One option is to define a tag for each stage of the design. So, let us say there are three stages in a design flow – design, verification, and layout, and from there define three tags – design_done, verify_done, and layout_done.

The design group will tag a consistent set of revisions of design files to 'design_done'. The verification engineers will use the 'design_done' tag to create workareas for verification while the design engineers can continue development. Revisions that successfully pass verification are tagged 'verify_done'. Engineers further along the chain can then use this tag to create a workarea.

Once the rules of engagement are understood and followed then data can be shared without confusion and error. *But, keep the rules as simple as possible otherwise they will either not be followed correctly or will add undue overhead.* If more complex procedures are required, scripts can be written to simplify day-to-day DM tasks.

In ClioSoft's SOS, pre- and post-event triggers can be set up by the project administrator on DM commands to streamline the design process. Thus, a pre-event trigger on the check-in command can run a lint check on Verilog files before checking them in. This can prevent syntax errors being propagated.

Collaborating across sites and time zones

The first thing to do when you have development spread across sites is to make sure that you have a reliable and secure network with sufficient bandwidth. Then, deploy a DM system that is optimized for multi-site collaboration.

Finally keep it real – realtime that is. Deploy tools such as instant messengers and VNC or services such as WebEx and encourage synchronous interaction between the sites. This will help catch errors and misunderstandings early on and build team spirit. SOS has a built in chat feature that allows engineers in the project to interact in realtime without maintaining 'buddy lists'.

Working with EDA tools

Electronic design data tends to be very complex. The relationship between the design abstractions the engineer works with and the physical files is one-to-many and can change all the time.

A designer using Mentor Graphics' ICstudio or Cadence Design System's Virtuoso Custom IC platform thinks in terms of libraries, cells and views. Each cell-view consists of several files. Some files are backup files or derived files and should not be version controlled. Sometimes the set of files that represent a design unit will change between revisions.

Only the EDA tool knows the relationship between the logical design units and the physical files. If engineers try to version control data at the file level, then they run a high risk of missing some required files or placing run files under version control preventing others team members from, say, running simulations.

To deal with this issue, EDA tools must be closely integrated with the DM system. The integration should be aware of and manage the sets of files that are part of a design unit. Users should be able to perform DM operations such as check-out and check-in at the abstraction level they are familiar with, and the integration should translate that to the physical files that must be operated on by the DM system.

Additionally, engineers are used to working in their EDA environment. An analog designer may be editing schematics and running simulations. A place and route expert may be editing the layout. Going to a separate DM tool for version management adds a

>>> continued on page 26

significant hurdle to deployment. By contrast, making DM commands directly available in the EDA tool environment will enhance usability and ensure the correct use of your DM methodology.

The EDA tool is aware of when a user may want to edit a design unit and can prompt the user to check-out the design unit just-in-time. Also, since the EDA tool is design-aware, DM operations can be performed on an entire design hierarchy.

Introducing DM to the design team

Once you decide to introduce a DM system, work out your requirements and methodology prior to any implementation. A small team representing the different disciplines involved can work this out for you. For a mixed-signal IC design group, this team may include a digital design engineer, an analog design engineer, a CAD engineer and an IT/network administrator.

The team needs to address the issues outlined above and develop a real working environment to test out the system. Along the way, several issues and questions can arise, which are not obvious at the outset.

The team may find that the directory structure and the tool and environment setup scripts will need to be modified to fit well with the new methodology. Here are other issues that frequently arise.

- Environment settings that resided in .rc files in users' home directories need to be incorporated into the revision-controlled setup files for the project.
- If users can customize part of their working environment, the environment settings that must be kept constant for all users need to be separated from those that may be altered. This may require the addition of non-revision controlled, user-editable 'include' files to the user workareas, to be read by include statements within the revision-controlled setup files.
- Simulation speed is often enhanced by running simulations on the user's local disk or submitting jobs to a remote machine or grid. To benefit from these techniques, it must be possible to cleanly separate the files needed for a particular simulation 'job' from the revision-controlled simulation source and control files.
- Clear check-in and tagging procedures must be defined, and their implications must be understood. For example, is it acceptable to check-in files/cells that do not compile or simulate properly in order to facilitate sharing of work-in-progress at different sites? If so, then users will rely on proper tagging to ensure that they obtain 'working' versions of files and cells.
- As release of a design approaches, the lead designers must have a well defined method of capturing the data files for the release. Should the file set be copied to a non-revision-controlled directory for separate back-up and submission to the vendor? At what point in the release flow should that occur?

Meanwhile, your CAD and IT groups need to be involved in identifying hardware and network requirements for the DM system and integrating it into monitoring activities and backup systems.

The CAD group must recognize its responsibility to help maintain the DM server, handle day-to-day user requests and interact with the DM vendor's support engineers.

ClioSoft

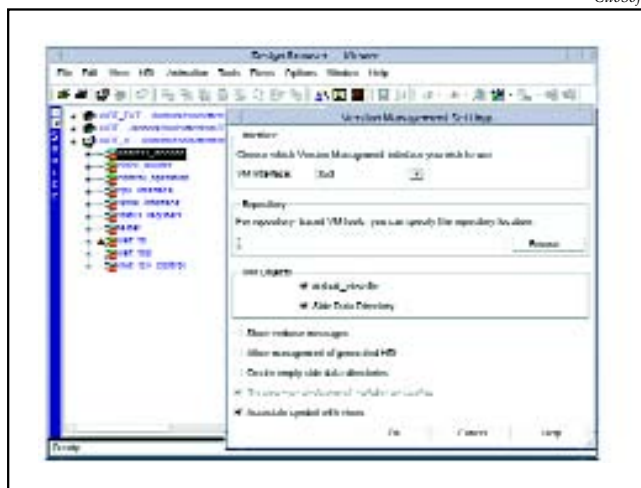


FIGURE 6 ClionSoft's DM available directly from Mentor's HDL Designer

Finally, the team leaders must recognize that many engineers have no experience of revision control and DM systems and concepts.

The CAD engineers and lead designers should provide clear documentation and possibly training. Design engineers may feel that the DM system imposes additional tasks or burdens, so it is important to highlight similarities and differences between ad hoc and DM strategies.

Often, a DM system clarifies tasks that have been handled by error-prone, ad hoc methods before, such as copying design data from another user for use in a simulation (e.g. is this really the same data that's going on the chip?). A team can usually be won over by demonstrating how the DM methodology enables everyone to be working with the 'real', verified project data, rather than having to maintain individual, probably out-of-date, copies.

Conclusion

Having a simple well-defined design DM methodology with the right infrastructure and tools seamlessly integrated with your EDA tools can help make your team more effective and productive. It will reduce communication errors, directly helping to improve quality, meet schedules, and significantly reduce the chance of requiring expensive re-spins.

RF Micro Devices
7628 Thorndike Road
Greensboro, NC 27409-9421
USA
T: (1) 336 664-1233
F: (1) 336 931-7454
www.rfmd.com

ClioSoft
43551 Mission Blvd, Suite 107
Fremont, CA 94539
USA
T: (1) 510 440-8424
F: (1) 510 440-8852
www.cliosoft.com