EMA Now Offers OrCAD/Aldec FPGA Design Bundles

EMA Design Automation recently announced the availability of OrCAD/Aldec FPGA design and simulation bundles, which provide a complete flow for PCB and FPGA design. "The OrCAD® product line provides a powerful set of tools for PCB design. With the recent addition of the Cadence OrCAD FPGA System Planner, users now have a solution to help integrate the FPGA onto their boards.

Also in this issue:

…and Much More!
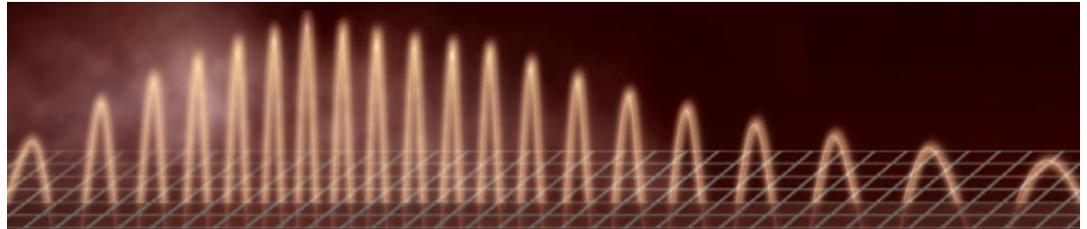
Creating a Toleranced Transfer Function in PSpice A/D

By Andrew G. Bell, Senior Staff Engineer, ITT Geospatial Systems

The use of toleranced models for worst-case analysis (WCA) is a common practice to determine whether a design will meet the specified design requirements. Typically, parts will vary in value due to how they were manufactured, how they are used, and environmental conditions. Simulation circuits that include these toleranced models can be used in tandem with the Monte Carlo feature in Cadence® PSpice® Advanced Analysis to determine the worst case performance of a circuit. But the question arises: how do you "tolerance" a transfer function?
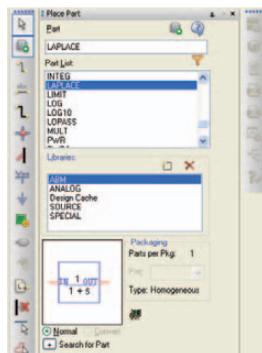
Figure 1: PSPICE LAPLACE Behavioral Model

Figure 2: Properties of LAPLACE Model

First, the PSpice A/D behavioral transfer function model "LAPLACE" can be found in the ABM library (and is shown in Figure 1). Of course, higher order transfer functions can be implemented by modification of the "DENOM" and "NUM" model properties (shown in Figure 2). For example, to implement the transfer function

$$\frac{Y(s)}{X(s)} = \frac{250}{s^2 + 10s + 10}$$

the numerator would be set equal to "250" and the denominator would be set equal to "s**2 + 10*s+100" (note that the s2 term syntax is s2 = s**2 and not s^2).

One of the limitations with the "LAPLACE" transfer function model is that the model cannot be toleranced. A toleranced model would allow the designer to evaluate variations in DC gain and pole-zero locations. To build a toleranced transfer function model, the transfer function must be translated into models which can be toleranced, like resistors. The technique I propose for this is to convert the transfer function into a simple mathematical simulation diagram where gain blocks and integrators are used to "solve the differential equation" (many years ago, analog computers were used to solve differential equations in much the same way, using Operational Amplifiers which could be programmed as gain blocks, integrators, and summers). From there, the gain

EMA Currents

EMA Adds Future Electronics to Component Data Management Solution

EMA recently announced the addition of Future Electronics to the list of component distributors included in the new 4.0 version of Component Information Portal™ (CIP), EMA's component data management solution. CIP provides design teams an automated method to manage their electronic part data. It is tightly integrated with Cadence® OrCAD® Capture CIS, providing access to component distributor data directly within the OrCAD design environment while improving both data accuracy and operating efficiency.

"Since its initial release in July of 2008, customer adoption of CIP continues to increase because customers see real value in an out-of-the-box data management solution," said Manny Marcano, president and CEO of EMA Design Automation. "With CIP, engineers are able to benefit from all the advantages that come with a centralized part library without having to become bogged down with designing and implementing a component database from scratch."

CIP 4.0 adds the ability to search parametric data from the Future Electronics part catalog to its existing list of supported distributors, Digi-Key®, Newark®, and Premier Farnell®. Users have online access to these part distributor databases, allowing them to easily find available parts that meet their performance and financial requirements directly from their engineering desktop. Once the desired part is found,

blocks can be replaced with a combination of a voltage controlled current source (VCCS), a 1 ohm parallel resistor, and a voltage controlled voltage source (vcvs). The resistor is then toleranced to match the amount of variability needed for each gain block.

To illustrate the proposed approach, consider the following simple example. Suppose you want to tolerance the following transfer function:

$$\frac{Y(s)}{X(s)} = \frac{1570}{s + 628}$$

First, the transfer function must be translated into a simple mathematical form by multiplying the numerator and denominator by the inverse of the highest order of the "s" term multiplied by U(s)/U(s):

$$\frac{Y(s)}{X(s)} = \left[\frac{1570}{s+628}\right]\left[\frac{\frac{1}{s}}{\frac{1}{s}}\right] = \frac{\left(\frac{1570}{s}\right)}{\left(1 + \frac{628}{s}\right)} = \frac{Y(s)\,U(s)}{X(s)\,U(s)}$$

Next, split the numerator and denominator into two transfer functions. The numerator transfer function, Y(s), can be rearranged as

$$\frac{Y(s)}{U(s)} = \left(\frac{1570}{s}\right) = 1570 * \left[\frac{1}{s}\right] \Rightarrow Y(s) = 1570 * \left[\frac{1}{s}\right] * U(s)$$

and the denominator transfer function, U(s), can be rearranged as

$$\frac{U(s)}{X(s)} = \frac{1}{\left(1 + \frac{628}{s}\right)} \Rightarrow U(s) = X(s) - 628 * \left[\frac{1}{s}\right] * U(s)$$

A simulation diagram, shown in Figure 3, can then be directly constructed from the two transfer functions using gain blocks, a summer, and an integrator. All these models can be found in the PSpice A/D ABM library. A PSpice A/D version of the simulation diagram, shown in Figure 4, can also be constructed.

Figure 3: Simulation Diagram

Figure 4: PSpice Simulation Diagram

To tolerance the transfer function, each gain block will be translated into a VCCS, a resistor, and a

VCVS. Then, the resistor value will be set equal to the desired gain and the resistor tolerance will be set equal to the tolerance for the gain. To independently tolerance the pole location and the DC gain of the transfer function, they must be split into two gain blocks of 628 rads/sec [R4] and 2.5 (V/V) [R5]. The R4 resistor value and tolerance establishes the value of the pole location and its variation. In this case, the pole location is 100 Hz, or 628 rads/sec. The R5 resistor value and tolerance establishes the DC gain of the transfer function, or 2.5V/V.

Figure 5: Fully Toleranced PSPICE Simulation Diagram

The pole location (determined by R4) has a ±10% tolerance, while the DC gain (determined by R5) has a ±5% tolerance. A 250 run Monte Carlo simulation (using PSpice Advanced Analysis), which measures the DC gain [MAX(V(VOUT3)/V(Vin))] and 3dB frequency [Cutoff_Lowpass_3dB(V(VOUT3)/V(VIN))], can be used to verify that the toleranced transfer function produces the desired variations. As shown in both Figures 6 and 7, the variations are close to the expected values:

n samples	= 250	minimum	= 89.7727	maximum	= 109.668
n divisions	= 10	10th %ile	= 91.8983	3*sigma	= 16.9473
mean	= 99.8145	median	= 99.7239		
sigma	= 5.6491	90th %ile	= 107.923		

Figure 6: Pole Variation Histogram

n samples	= 250	minimum	= 2.37578	maximum	= 2.62268
n divisions	= 10	10th %ile	= 2.40577	3*sigma	= 0.206353
mean	= 2.49943	median	= 2.49952		
sigma	= 0.0687844	90th %ile	= 2.59363		

Figure 7: DC Gain Variation Histogram

This was a simple example of how to tolerance a transfer function. This technique can easily be duplicated for higher order and more complicated transfer functions, with the added advantage that both PSpice A/D frequency and time domain simulations can be analyzed.

For more information e-mail info@ema-eda.com. ●

Adding Assertions to your FPGA Design Process

By Chris Banton, Product Marketing Manager, EMA Design Automation

Assertions are quickly becoming a standard part of the FPGA design and verification process, making the ability to create and use assertions a required skill-set for FPGA designers. But what exactly are assertions, and how can utilizing them benefit yourself and your company?

Our application engineers and account managers receive questions on assertions on an almost daily basis. Below are answers to the most commonly asked questions on the topic:

What are Assertions?

Before we answer this question, we have to introduce design properties first, as assertions are a subset of design properties. Design properties are the formal way of specifying the desired or undesired behavior of the design. This behavior can be specified in many different ways, but there are two methods that are used the most frequently: the assert statement and the cover statement. If you are interested in using assert statements (assertions), then you typically will also want to employ cover statements as well.

Assert statements allow you to embed monitors in your design to verify that bad behavior never happens. If unanticipated or bad behavior does occur, the monitor will be flagged, allowing you to quickly locate the point of failure. Assertions can be very beneficial in your day-to-day design work by allowing you to catch bugs much faster and deliver the completed design much earlier. The larger the design you are working on, the more visible the benefits of using assertions will be.

Cover statements, on the other hand, allow you to check that good behavior is exercised during verification. Coverage improves your design quality by making the verification process more thorough. With coverage, the verification engineer is informed if certain behaviors of your design are not tested. It is better to discover this during verification then after manufacturing or delivering the design to your customer.

> Assertions can be very beneficial in your day-to-day design work by allowing you to catch bugs much faster and deliver the completed design much earlier.

These properties provide a way to speed up verification through faster and more accurate bug identification, while ensuring all aspects of the design have been exercised appropriately. Remember that design verification really means the point at which you can declare the design "ready for production". This does not mean using asserts and covers will lead to less simulations (although it probably will). What it does do is allow you to get to that "ready for production" point faster and with greater confidence that you really have fully exercised the design (which is the real goal).

Why Should I Bother With Assertions?

A major reason to begin using properties and assertions, even when considering your current project list and workload, is the fact that your design specification is already full of design properties; they are just expressed in plain English. Adding assert and cover statements is a way of embedding those design properties from your specification in the actual design code in a format that your verification tools can understand. Since your verification tool can now interpret your design specifications, it can now provide you direct data as to whether you are meeting those high level design requirements. As a result, your verification tools will be doing the hard work for you. They will alert you when some properties do not hold and tell you if your testbench tested all functional aspects of your design.

How do Assertions Work?

Many designers want to know how assertions and coverage work before learning how to use them. Imagine adding a time dimension to the traditional Boolean logic: instead of static values, it would have to operate on paths that represent sequences of states of your design. Such extension of Boolean logic exists and is called temporal logic. Properties, assertions, and coverage implement principles of temporal logic. Using this temporal logic, your simulator will then build internal state

EMA Currents

However, they are still lacking the ability to design and verify the FPGA itself," said Manny Marcano, president and CEO of EMA Design Automation. "By including Aldec Active-HDL™ in these bundles, we have filled that gap, providing multi-language design entry and simulation for FPGA logic."

The increased functionality and performance available within today's FPGAs have resulted in a growing use of programmable logic across all areas of electronic design. While this has led more design teams to select an FPGA based implementation, this additional functionality presents a rising design and verification challenge. The OrCAD/Aldec FPGA Bundles provide the tools needed to handle the increasing complexity of FPGA development. Going beyond what typical vendor supplied simulators can provide, these bundles deliver a scalable vendor independent platform to accommodate even the most complex mixed language FPGA verification tasks. In addition, the Aldec simulator provides a significant speed increase over vendor supplied offerings and contains no limitations on design size. "Bundling OrCAD and Aldec solutions together provide a powerful FPGA solution to the mainstream market at very competitive price points," said David Rinehart, vice president at Aldec.

The OrCAD/Aldec FPGA Bundles add FPGA design entry and simulation to OrCAD products at an incremental cost starting at $1,756. For more information, please visit www.ema-eda.com/currents/orcadaldec, call 877-362-3321, or e-mail info@ema-eda.com.

machines, implementing behaviors described by the properties you have defined and comparing the state of those machines against the state of your design to evaluate assertions and coverage.

What Languages can be used to Express Properties, Assertions, and Coverage?

Another concern for designers considering the use of assertions is understanding what language assertions can be used in. The dedicated language to express properties, assertions, and coverage for property-based verification is Property Specification Language (PSL). These three items are also a part of verification languages, such as System Verilog, OpenVera, or e.

Which Assertion Language Should a User Choose?

Any of the above languages can be used; it all boils down to personal preference. However, those designing in VHDL typically will find PSL the most comfortable fit (a PSL subset is also included in the new version VHDL IEEE Std 1076-2008), while Verilog users will gravitate toward System Verilog (as it's a natural extension of Verilog). OpenVera and e tend to be used more in Verilog environments as well.

Will I be able to Synthesize My Sources After Adding Assertions?

Some are hesitant to learn about assertions because they think they will not be able to synthesize their sources after adding assertions. This is not true. However, some steps need to be taken to avoid problems during synthesis. Two common methods are:

• If you like to keep your assertions very close to the related HDL code, you can write them directly in the VHDL or Verilog source. To prevent the synthesis tool from reading them, you can place them in specially marked comments or surround them with pragmas disabling synthesis

• If you like to keep them separate from the related HDL code, simply place them in separate files and exclude those files from synthesis

How Much Time do I Need to Learn Assertions?

You need four to eight hours of guided training to learn everything you need to immediately start using properties, assertions, and coverage in your designs. Attaining a high proficiency level requires more time, but it is the breaking of old habits that really counts. The time spent once for learning will pay back repeatedly in faster, better work.

How Much Money do I have to Spend to Start Implementing Assertions?

Adding assertions to your design flow should have a minimal impact on your budget. As mentioned, the training required to be able to design using assertions is fairly minimal. What is needed are simulations tools that can interpret the properties you are specifying so you can make your simulator design intent "aware". This powerful verification technique is not available in the standard vendor offerings. Aldec® Active-HDL™ is one high performance, mixed-language RTL Simulator that provides the ability to include assertion and coverage capabilities into your design environment.

We hope this compiled list of questions helps clarify the world of assertions and design properties. Employing these techniques in your FPGA development environment can automate some of your most time consuming tasks, increasing your value as a designer and helping your company's bottom line.

For more information about Active-HDL and FPGA verification, please visit www.ema-eda.com/currents/aldec. For additional questions, please contact EMA at 877-362-3321 info@ema-eda.com. ●

Selecting a Product Lifecycle Management (PLM) Solution that Addresses Small Business Needs

By Chuck Cimalore, Chief Technology Officer, Omnify

A small or medium sized business (SMB) that is looking to gain a competitive advantage will often do so by improving their products and streamlining their operations. In these situations, decision makers will look toward Product Lifecycle Management (PLM) technology to support these changes. By implementing a Product Lifecycle Management system, companies can simplify and shorten each phase of the product development process. Selecting new technology is a challenge for any organization, but can be especially difficult for the SMB since most software (particularly PLM software) is designed to meet the needs of a large enterprise.

To begin the selection process, the SMB first needs to understand what the PLM system should accomplish for its organization. PLM can provide key functionality to streamline each phase of a product's lifecycle, from product conception and design to manufacturing and support, while improving communication across both internal and external constituencies. A successful PLM implementation can help reduce time-to-market and decrease product costs, and also dramatically reduce waste and rework. Once a company determines that PLM software will address the challenges within their organization, the question now becomes which PLM software is best suited for their company.

Because the SMB operates with a smaller cash flow, develops fewer products, and has a smaller customer and supplier base as compared to a large enterprise, the PLM system not only needs to address all development challenges, but it needs to be tailored to the distinct market segment. The increasing emphasis on outsourcing has dramatically changed the operational landscape of the SMB, and must be a key consideration when choosing the correct PLM system. Too often the SMB will select a PLM system that includes broad functionality, but fails to drill down into the daily challenges of engineers and developers to embrace the need to share product data outside the SMB's four walls.

PLM solutions designed for the large enterprise often manage few developmental logistics, and instead tout the capability to encompass supply chain management, packaging, and post-development phases. For the smaller business looking to improve products, the process should begin with the engineers who face issues with data management and communication with contract manufacturers and external partners.

A common misconception is that large PLM vendors with monolithic software systems are the only vendors that can address all product development needs. In reality, many of these large systems, which have been designed with large organizations such as automobile manufacturers in mind, include functionality that is irrelevant to the development needs of a small organization. Further, the SMB (with its limited cash flow) is paying for functionality that will likely never apply to its product design process.

Many smaller companies are beginning to look toward implementing PLM software designed for their market segment. In return, several large PLM and ERP vendors are attempting to "scale down" their products to address the PLM needs of the SMB. Unfortunately, because these systems are typically built on their legacy foundations, their heredity does not allow them to adapt easily to the evolving needs of the smaller organization. Moreover, these systems still require lengthy (and costly) implementation phases.

Implementation time and costs are critical factors for the SMB to consider when selecting a PLM system. SMBs cannot afford to reassign their valuable resources to long PLM implementation projects. Most software designed for the large enterprises have considerably lengthy implementation processes of more than six months. Alternatively, software designed for the SMB has a quicker implementation process and can have the company up and running within days or weeks, depending upon the vendor and level of integration with other systems.

Power IC Model Library for Cadence PSpice Technology Adds TI Model Line-up in 3.0 Release

Version 3.0 Power IC Model Library™ has over 300 time-domain simulation models for power electronic designs. Several previously unavailable Texas Instruments (TI) models debut in version 3.0, and future library updates are anticipated to focus on many of TI's newest and most popular components.

The Power IC Model Library includes model netlists in PSpice syntax, schematic symbols for both Cadence OrCAD® Capture and legacy schematics, and a set of example application circuits for many of the IC models. The models are compatible with OrCAD Capture version 16.x software.

"The Power IC Model Library for PSpice has models of parts that are simply not available from any other EDA company," said Manny Marcano, President and CEO of EMA Design Automation. "AEi Systems has proprietary relationships with nearly all of the top analog IC manufacturers. These relationships provide unique access to the part characteristics needed to produce models with the accuracy our customers expect.

"This library makes the PSpice simulator even more compelling for engineers in the power supply market. Many EDA vendors only have access to information in the manufacturer data sheets. This is simply not sufficient to create a precise model of a controller or regulator," stated Charles Hymowitz, Managing Director of AEi Systems. "Data sheets

EMA Currents

do not have the level of detail required, so those companies and modelers who rely solely on data sheet input will necessarily produce inferior, inaccurate models."

AEi Systems and EMA are keeping the price of version 3.0 of the Library the same as the current version: $1,995, plus yearly maintenance of $495. For more information on the Power IC Model Library, please visit www.ema-eda.com/currents/powerIC, call 877.362.3321, or e-mail info@ema-eda.com.

Switch Mode Power Supplies (SMPS) Workshop

EMA recently hosted a series of SMPS workshops, which showed how to use PSpice A/D and Cadence PSpice Advanced Analysis to design and analyze switch mode power supplies.

Attendees learned:

• What is needed for SMPS simulation

• What PSpice technology offers for SMPS design simulation

• Advantages of a unified simulation/implementation environment

You can request a free version of these workshop materials by filling out this form: www.ema-eda.com/currents/smps.

Stay tuned for news on additional SMPS workshops taking place throughout North America over the rest of this year.

SMBs often have a competitive advantage over their larger competitors when it comes to customer support and responsiveness. SMBs can react quicker to the evolving needs of their customers and target market, with intimate support and new or enhanced products. This should also be an important consideration for the SMB when choosing a PLM system. SMBs should consider such purchases to be a "partnership" with the vendor. For the SMB, their PLM system needs to be easy to use, have a low total cost of ownership (TCO), and be from a vendor who is dedicated to the success of that SMB.

Finally, companies frequently overlook the importance of employee buy-in when selecting a new PLM application. If it is necessary for employees to learn an encyclopedia of information before using the software, the staff will be more resistant to using the tool and improving the product development process will be a greater challenge. In order to see a quicker return on investment (ROI), employees need a user-friendly PLM tool that requires limited ramp-up-time. If employees feel comfortable using the software, they will be more likely to use it for a range of purposes. Within a smaller business, employees are expected to take on multiple functions, so it is likely they will need to understand many aspects of the software. If the software is easy to learn and easy to use on a daily basis, companies will see a greater and quicker transformation across the development process.

Selecting a PLM system can be a challenge if a company is not aware of the options available within the marketplace. But if an organization starts the process by understanding what the technology should address, and further understand the needs of their unique business and market segment, they can select a PLM tool that will allow the organization to develop products more efficiently and ultimately transform their organization.

The PLM solution from Omnify® helps manufacturers manage their product data from concept to obsolescence in a unified environment. Omnify supplies a single, secure location to manage all the essential information for designing, manufacturing, and supporting your products, and enhances visibility into all aspects of product development with features such as:

• Product Data Management

• Bill of Material (BOM) Management

• Engineering Change Management

• Document Management

• Project Management

• Quality/CAPA Management

• Training Records Management

• Compliance Management

For more information on Omnify, please visit www.ema-eda.com/currents/omnify. ●

Students Use OrCAD to Design Solar Powered Car

By Joe Mignano, Marketing Specialist, EMA Design Automation

EMA established the OrCAD North America Academic Program to help colleges and universities better prepare their future graduates for the PCB design workplace. This program provides industry-standard, commercial-grade software tools at reduced prices to accredited colleges and universities for educational purposes.

The Coppell Solar Racing Team of Coppell High School in Coppell, Texas (just north of Dallas, Texas), is one academic institution currently utilizing this program. They used the Cadence OrCAD PCB Design Suite for Academic Institutions to design a single-passenger electric vehicle recharged only by solar power.

The solar electric vehicle designed by the team was entered in the 2010 Hunt-Winston School Solar Car Challenge race. The goal of the race was to teach high school students around the world how to build roadworthy solar cars. The race began on July 18, starting at Texas Motor

Continued from page 6

Speedway in the Dallas/Ft. Worth area and ending at the National Renewable Energy Laboratory in Golden, Colorado. The team finished first in the Advanced Division, averaging 34.73 mph during the race.

There were several reasons why the Coppell Solar Racing Team chose OrCAD products to assist in their electrical design. The biggest factor was an electrical team mentor's positive experience with OrCAD products throughout his professional career.

"I've used OrCAD PCB design technologies quite a bit in my career, and have always found them easy to use," said Gary Kidwell, an electrical team mentor. Kidwell currently works as a principal member of the technical staff at Interphase Corp in Plano, Texas, and has worked for 30 years in embedded computer technology. "What I like best about using OrCad for this project is you do not have to have a big library set up to begin using it. I thought that made it a good fit for the high school kids who have zero experience with schematic capture, and don't even really know what it is."

"The students needed the OrCAD PCB Design Suite primarily to draw out and insert pages into the electrical design, such as fuses, switches, motors, and connectors. As a result, I wanted to pick out a tool they could learn fairly quickly and have the basic components already built that they could put in their schematic. Some of the parts we have built ourselves, but most came with the library that comes with OrCAD Capture," said Kidwell. "It was very easy to build a new part within OrCAD Capture when we needed to."
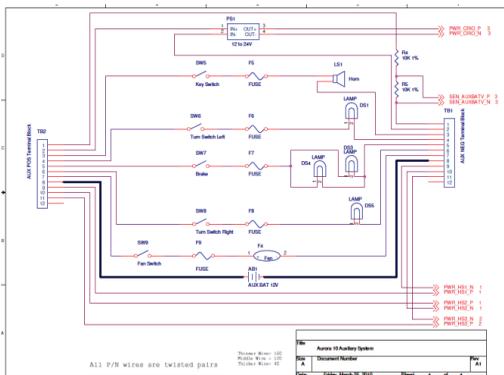
Figure 1: Image of Schematic for Solar Car

The students liked the software and were able to get up to speed very quickly. The car was designed to have two electrical systems running

(one at 72 volts and one at 12 volts). The 12 volt system was the auxiliary system that powered the fans, brake lights, flashers, and telemetry equipment. The 72 volt system was the main power systems for the car, which included an 1170 watt photovoltaic (solar) array, a 5 kilowatt battery array, the Maximum Power Point Trackers (MPPT) that adjusted the voltages from the solar array to be compatible with the battery array, and a motor and controller.
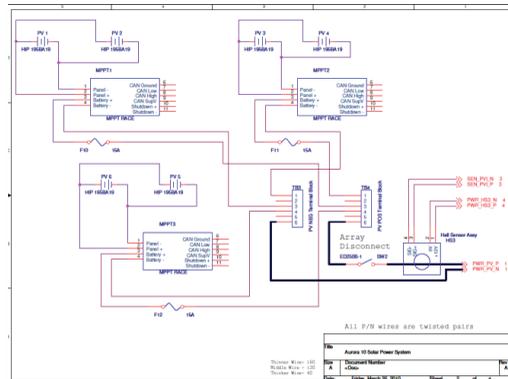
Figure 2: Image of Schematic for Solar Car

Deciding on the correct software package was important to the team, but being able to acquire and install it efficiently was also important. "Our Account Manager was great," said Michael Yabubovsky, lead engineering instructor for the team and a pre-college engineering teacher at Coppell High School. "He helped us find the right product and assisted us as we purchased and installed it. The installation was a little tricky because we were installing it on the school's computer and had to work in a pretty strict environment. The technical support provided by EMA was helpful, and we were able to install all our software without any issues."

"After using this software from EMA, the students on the team have a much greater understanding of the electrical system of a car and how it works. Before we began using the OrCAD PCB Design Suite, they were not nearly as knowledgeable about the electrical system and how it really functioned. They only knew that it did. In previous competitions, they used the same setup that other teams have used without knowing why," said Yabubovsky. "Electronics is not an easy subject for most students, and these students are finding it easier to understand after using the OrCAD software."

The schematics for the main and auxiliary power systems were completed at the end of March. In

Stay Connected with EMA and "Team OrCAD"

Several social media pages have been launched that are aimed at keeping customers and others in the EDA and MCAD industries informed of the latest news, events, promotions, and technical tips.

For Twitter users, one page has been created for those involved with EDA (www.twitter.com/ema_eda) and another has been created for those involved with MCAD (www.twitter.com/ema_mcad). These two pages provide a steady stream of information on all the EDA and MCAD technologies sold and supported by EMA.

For YouTube users, there is a "Team OrCAD" YouTube Channel (www.youtube.com/user/TeamOrCAD) packed with all the latest OrCAD PCB design technology movies from Cadence and its worldwide channel partners (including EMA). EMA's YouTube Channel (www.youtube.com/EMADesignAutomation) contains movies focused on all the technology sold and supported by EMA, as well as product integrations and enhanced capabilities provided to users of OrCAD technology.

Lastly, there is now a Facebook page for OrCAD users. The "Team OrCAD" Facebook page (www.facebook.com/TeamOrCAD) is designed for OrCAD users to interact with users and resellers across the world. The page is continuously updated with movies, application notes, and other useful information from Cadence and their worldwide channel partners, including EMA.

Stay tuned for additional ways to use social media to stay connected with EMA and "Team OrCAD"!

EMA Now Offers Autodesk's 2011 Digital Prototyping Software

EMA recently announced it will offer Autodesk's latest 2D and 3D design and engineering software for manufacturers.

Digital Prototyping with Autodesk Inventor software gives manufacturers the ability to digitally design, visualize, and simulate how a product will work under real world conditions before it is built. Digital Prototyping reduces reliance on physical prototypes, which helps reduce cost and speed time to market in highly competitive industries.

"The EMA/Autodesk solution for Digital Prototyping provides a revolutionary design experience that offers significant benefits to engineers," said Greg Roberts, director of marketing at EMA. "The 2011 release of Autodesk products unites direct manipulation with the more traditional parametric based workflows for a faster and easier to use interface. This release allows designers and engineers to capture and embed engineering and product knowledge directly into virtual models, and it provides a new and easy-to-use product for creating compelling product documentation."

For more information on EMA's Autodesk offerings, please visit www.ema-eda.com/products/autodesk.aspx, call 877.362.3321, or e-mail info@ema-eda.com.

the future, the Coppell Solar Racing Team plans to use PSpice A/D and OrCAD PCB Editor to design custom interface boards to gain an even better understanding of electronic design.

If you are part of an academic institution with an electronic design program, please contact EMA at 877.362.3321 or info@ema-eda.com for more information how you can acquire OrCAD products at an extremely affordable price. We are also interested in hearing about how other academic institutions are using their OrCAD products and what they are using them for. ●
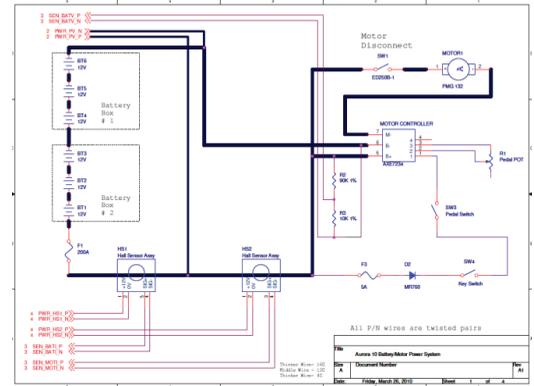
Figure 3: Image of Schematic for Solar Car

Scripting Capability in OrCAD Capture

By Cadence Design Systems, Inc.

As part of the Cadence 16.3 release, OrCAD Capture was enhanced to include support for Tcl/Tk scripting language. This capability provides immense power for users to interact with both the OrCAD Capture user interface and the design database, and greatly extends a user's ability to develop custom applications and scripts for their design environment.

The desire for specialized productivity improvements and tool efficiency drives the need for scripting and customization in a user's design environment. With scripting and customization, designers can apply automation to manual processes and complete projects faster, difficult operations can be streamlined and custom features that do not exist natively can be created, further enhancing and extending the OrCAD Capture environment.

TCL Commands
The Tcl/Tk language provides an extensive set of scripting functionality. The core functionality includes procedures and commands for data manipulation, control constructs, mathematical expressions, file I/O routines, system calls, registry handling, GUI designing, and many more. The always growing additional packages of Tcl/Tk are just making almost everything possible that can be done using any procedural language.

The integrated TCL interpreter in OrCAD Capture allows any commands from these Tcl/Tk packages to run seamlessly. On top of that, OrCAD Capture

provides a rich set of its own TCL commands that gives immense power to the users to interact with both Capture UI and database through the scripting interface.

OrCAD Capture provides two sets of pre-defined Tcl commands:

• User-action Tcl commands

• Database Tcl commands

User-action Tcl Commands
User-action Tcl commands correspond to the operations within the GUI performed by the user. Every user action that can be performed in OrCAD Capture is now executable in the form of a Tcl command. These capabilities allow users to perform a large set of pre-defined and customized operations automatically for specific needs. They also help designers when debugging custom scripts, as they provide a definite mechanism for step-by-step script re-creation.

The Journaling option provides the facility to capture, store, and later replay a customized command or copy for use in an external script. This option is set through the TCL command "SetOptionBool Journaling TRUE" and allows manual actions to display the corresponding TCL commands. All these TCL commands can be replayed individually. OrCAD Capture also automatically stores these commands in a TCL script file that can be replayed to repeat the steps.

For example, if a user performed the following action:

Click the "Place Wire" icon and then draw a wire from coordinates 4.00, 1.00 to 6.00, 1.00

the action is captured and stored as a Tcl command in the command window as "PlaceWire 4.00 1.00 6.00 1.00". Executing this Tcl command would have the exact same effect as placing the wire manually with the "Place Wire" command.
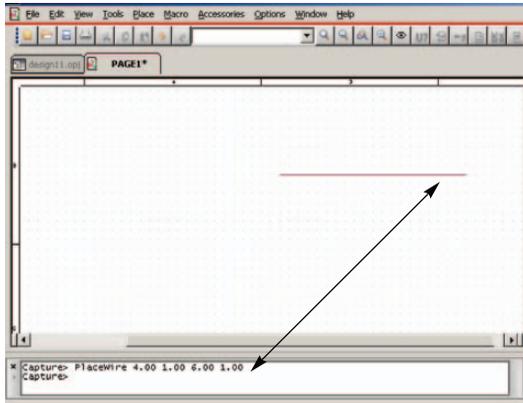
Figure 1: User Action TCL Command

Database Tcl Commands

OrCAD Capture also provides a rich set of Tcl commands that allow users to interact with design and library databases directly. These database Tcl commands allow for both querying and manipulation of the design and library objects. All types of design and library objects (schematics, pages, parts, pins, nets, wires, globals, off-pages, ports, library packages, graphic objects, etc.) and their properties can be queried, iterated, and manipulated using Tcl commands.

The database Tcl commands provide the same power as OrCAD Capture software design kit application programming interfaces (SDK APIs). Due to their ability to directly interact with the database objects, engineering organizations and end users can build large sets of customized design verification and manipulation procedures using these Tcl commands.

As an example, if a user wants to simultaneously manipulate all the parts on a schematic page and add a custom property "PartVersion" with a value of "1.1", the following custom Tcl procedure can be utilized for this:

```
proc addPropertyToAllPartsOnPage { pPage } {

    set lNullObj NULL
    set lStatus [DboState]
```

```
    set pPartInstsIter [$pPage NewPartInstsIter
        $lStatus]
    set pInst [$pPartInstsIter NextPartInst
        $lStatus]

# iterate over all parts

    while {$pInst!=$lNullObj} {
        set lPropNameCStr [DboTclHelper_
            sMakeCString "PartVersion"]
        set lPropValueCStr [DboTclHelper_
            sMakeCString "1.1"]

#add the property to part

    set lStatus [$pInst SetEffectivePropStringValue
        $lPropNameCStr $lPropValueCStr]
    set pInst [$pPartInstsIter NextPartInst
        $lStatus]

    }

    delete_DboPagePartInstsIter $pPartInstsIter

    $lStatus -delete

}
```

These database Tcl commands can be run within the OrCAD Capture command window as well as in a standalone Tcl shell. This window allows users to modify their custom Tcl procedures as needed within a session. The modified script can be sourced again in the command window to display the changed behavior. Any Tcl-aware application can also use these commands directly within its process space. Tcl commands are available to get active sessions, create new sessions, get active design/schematic/pages, open any design or library in the session, and then perform the iterations to retrieve and manipulate the desired set of database objects.

Automatic Script Loading and Procedure Calls

The scripting framework in OrCAD Capture provides various mechanisms for automatic script loading and automatic calling of Tcl procedures. This framework allows users to have customized sets of operations and behaviors automatically available when starting OrCAD Capture and for various types of trigger events. If a user wants to always place a set of custom title blocks on a new page, they can use this feature to run a custom Tcl procedure automatically whenever a new page is created.

New On-Demand Demos

From beginners to advanced users, there are demos for every skill level ready to be viewed at: www.ema-eda.com/currents/multimedia.

Signal Navigation and Intersheet References in OrCAD Capture
Are you spending time trying to track signals that span multiple schematic pages? This presentation shows how OrCAD Capture allows you to navigate design connectivity across multiple schematic pages quickly and effectively. www.ema-eda.com/currents/movie1.

Autowiring in OrCAD Capture
Spending too much time manually defining connectivity in your schematics? This presentation demonstrates how unique automated wiring capabilities in OrCAD Capture free you from tedious point and click methodology, allowing you to focus on engineering and adding value to your product. www.ema-eda.com/currents/movie2.

Automate your New Part Introduction Process with OrCAD
CIP provides OrCAD Capture CIS users instant access to millions of orderable parts in the Future Electronics, Newark, Digi-Key, and Premier Farnell databases, with device parameters, RoHS compliance status, cost, quantity on hand, and mechanical dimensions. This demonstration will highlight how easy it is to download part distributor information directly into OrCAD Capture CIS. You also will discover how CIP can help automate new part requests and streamline the part selection process, how you can get better control of the CIS database, and how to ensure a more reliable and shorter design cycle. www.ema-eda.com/currents/movie3.

EDA Industry Focus Shifts to Integration and Profitability

Today, systems and semiconductor companies are undergoing a disruptive transformation so profound that even the best-known companies will be impacted. The EDA industry now stands at a crossroads where it also must change in order to continue as a successful, independent business. Without that change, EDA will become a fragmented industry offering suboptimal, poorly targeted solutions that fail to solve customer problems. As a result, the huge leap forward provided by the electronics revolution will come to a standstill. The result? A squandered opportunity for technology innovation, and a diminished contribution by the electronics industry to re-build the global economy.

The disruptive transformation we are speaking of is not about EDA developing new design tools. It is not about new methodologies. It is not about the functional verification crisis, or the move to electronic system level (ESL) design, or any of the issues that have dominated discussions about EDA to date. It is about something much larger. It begins with a shift from design creation to integration in the electronic systems industry, and results in a new focus on profitability. This realization, in turn, opens the way to EDA360, a new vision for what the EDA industry can become.

To tell the story, however, we must take a step back for a moment to where everything starts—the electronics consumer.

To read the rest of this article, please visit www.cadence.com/eda360.

Framework examples include:

- Automatic loading of scripts at startup: OrCAD Capture automatically sources capture/tclscripts\capinit.tcl file at startup. Also, all Tcl files present inside the capture\tclscripts\capAutoLoad folder are sourced automatically during startup. Using this method, users can make custom scripts and procedures available by default.

- Automatic calling of TCL procedures on event triggers: OrCAD Capture provides Tcl procedure hooks to execute scripts upon event triggers. The custom Tcl callback procedure is registered using the "RegisterAction" Tcl command. Any number of Tcl callback procedures can be registered against an event. They all are called in sequence when the particular event is triggered.

To place a set of custom title blocks on a new page, the addition could call a custom Tcl procedure 'AddPageCustomTitleblocks' automatically at every new page creation. The registration of the Tcl procedure would be performed with the following command:

    RegisterAction"_cdnOrOnNewSchematicPage" "capTrue" "" "AddPageCustomTitleblocks" "PM"

where

    New page creation hook unique name = _cdnOrOnNewSchematicPage

Custom TCL procedure to call on new page creation = AddPageCustomTitleblocks

- Adding a custom menu within the Accessory menu: Tcl customization allows users to add custom menus within the Accessories menu through the command "AddAccessoryMenu". These commands can be used with Register Action hooks "_cdnCapTclAddPageCustom Menu" and "_cdnCapTclAddDesignCustom Menu" to automatically create these menus when OrCAD Capture is started.

For example, to add a custom menu "Capture Communication Server" with submenus "Start" and "Stop" as well as their associated Tcl callback procedures, the following Tcl command would be used:

    AddAccessoryMenu "Capture Communication Server" "Start" "::capCommServer::StartServer"

    AddAccessoryMenu "Capture Communication Server" "Stop" "::capCommServer::StopServer"

Figure 2: Custom Menus

- Run OrCAD Capture with a replay script: OrCAD Capture can be started from the command line along with a replay Tcl script file. All the user actions mentioned in the script file automatically run as OrCAD Capture starts.

Inter-Process Communication with OrCAD Capture using Tcl
OrCAD Capture provides a sample Tcl package named "capCommServer" that can be used as a bidirectional communication mechanism between OrCAD Capture and any other standard Tcl-aware application. Users can write custom client and server-end Tcl procedures and use this mechanism as a framework for establishing bidirectional communication between the two applications. Any of the user-action or database level Tcl commands can be run inside OrCAD Capture from other applications using this approach.

Tcl/Tk Setup
OrCAD Capture TCL/TK framework works with the Tcl/Tk libraries version 8.4, which is part of the standard 16.3 installation.

Additionally, application developers can use advanced Tcl features or Tk GUI extensions (ttk, BWidgets, etc.) to develop their custom applications. To do this, developers are advised to download and install ActiveState ActiveTCL version 8.4, which is available for free at www.activestate.com/activetcl/downloads.

Details of the setup process are provided in the scripting interface help document that comes with the download. TCL/Tk specific information in this document applies to OrCAD Capture versions 16.3-S013 and later.

Samples and Application Information
A detailed application note along with several scripting samples is available at www.ema-eda.com/currents/tcltk. For more information on Tck/Tk scripting capabilities, please contact EMA at 877.362.3321 or e-mail us at info@ema-eda.com. ●
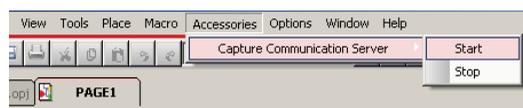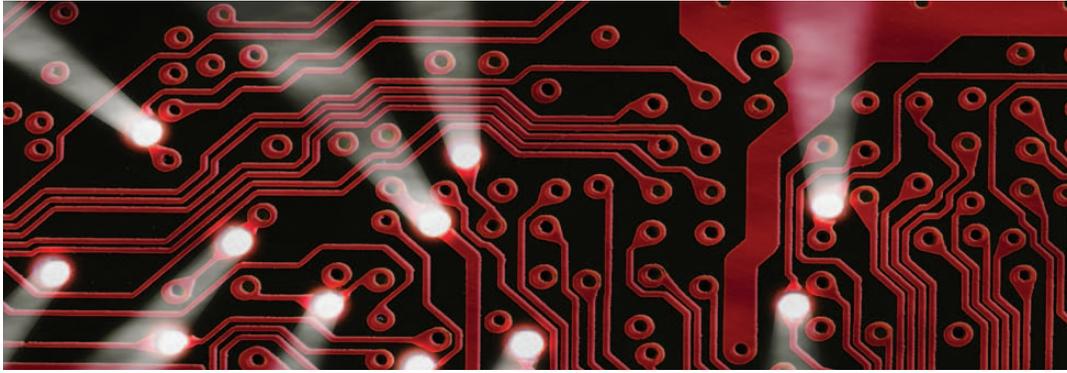
OrCAD PCB Editor Time Saving Tips

By Bill Zembek, Technical Support Specialist, EMA Design Automation

In today's design industry, resources are limited and time schedules are more critical than ever. Your PCB tool needs to keep pace by making every keystroke count. OrCAD PCB Editor provides a very flexible environment to help with design efficiency and productivity.

Once you have mastered the basics of OrCAD PCB Editor, there are several ways to take full advantage of this technology's functionality. One such way is the use of "hot key" shortcuts. These shortcuts allow you to get away from using the mouse to perform tedious tasks that can be performed quicker with the use of the keyboard.

> OrCAD PCB Editor provides a very flexible environment to help with design efficiency and productivity.

Below are a few tasks made easier through the use of "hot key" shortcuts. To begin using these shortcuts, simply type the lines of code indicated below into your local env file.

Change Active/Alternate Layers
You can increment or decrement the active or alternate layer in your PCB design by using the following shortcuts:

- To increment the active subclass, hit the "+" character after implementing this code: funckey + subclass -+

- To decrement the active subclass, hit the "-" character after implementing this code: funckey - subclass --

- To increment the alternative subclass, hit the "a" character after implementing this code: funckey a altsubclass -+

To directly change the active layer, use the following shortcuts:

- To make the top layer the active later, hit "1" after implementing this code: funckey 1 options subclass TOP

- To make 2nd layer the active layer, hit "2" after implementing this code: funckey 2 options subclass SIGNAL_2

You can program function keys for as many layers as are on your board. Another method to change the active subclass is by clicking the right mouse button (RMB):

RMB > Quick Utilities > Change Active Subclass

Adding Vias
Adding a via to your design has traditionally been done with a double click of the left mouse button (LMB). However, the use of the space bar can save you thousands of mouse clicks per year. To set up this functionality in OrCAD PCB Editor, add this shortcut to your local env file:

funckey " " "pop bbdrill -cursor"

The spacebar entry is represented by " ". Other keys can be assigned as a shortcut, but the space bar is easy to use without having to look down at your keyboard.

Add Connect and Adding a Vertex
Instead of using the left mouse button (LMB) to add a vertex point during Add Connect, consider hitting the "x" key every time you want to add a

PSpice Technology to be Used by STMicroelectronics

Cadence announced in April that STMicroelectronics, a global leader in integrated circuits for communications, consumer, computer, automotive, and industrial applications, has selected PSpice technology to provide simulation capabilities to its customers to evaluate the company's analog and power IC's. Customers will be able to run a free version of PSpice technology, a robust and widely used platform, within specific circuit testbenches.

"Recommending PSpice technology to our customers is a great way to enable them to test-drive our analog and power IC products," said Carmelo Papa, corporate vice president and general manager of STMicroelectronics' Industrial and Multi-Segment Sector (IMS). "Evaluating our chips with Cadence software will strengthen our customers' confidence in receiving the highest quality ICs needed for their success."

"We are honored and excited by STMicroelectronics recommending that their customers use PSpice technology in an integrated evaluation environment," said Charlie Giorgetti, corporate vice president of product marketing at Cadence Design Systems. "STMicroelectronics' decision to recommend PSpice software confirms that our global PSpice simulation environment is the first choice for today's analog design engineers."

The first product families that can deploy PSpice technology will be DC-DC converters and ViPERTM. All industrial and multi-segment sector analog and power IC products will eventually be included. For more information, please visit www.ema-eda.com/currents/stmicro

Cadence OrCAD Classroom Training

Increase your PCB design productivity by attending our in-depth, hands-on classroom training sessions. All our classroom courses are taught by professional instructors with years of training and application experience. Learn more at htpp://www.ema-eda.com/currents/training.

EMA classroom training is coming soon to a city near you…

Seattle, WA
September 13 – September 24

Ottawa, ON
September 20 – September 24

Sunnyvale, CA
September 27 - October 8

St. Paul, MN
September 27 - October 8

EMA | **Design Automation**™
ema-eda.com

EMA Design Automation, Inc.
225 Tech Park Drive
Rochester, New York 14623

Phone: 585.334.6001
Toll Free: 877.362.3321
Fax: 585.334.6693
eMail: info@ema-eda.com
Web: www.ema-eda.com

vertex during routing. Below is the code to add to your env file for this shortcut:

```
funckey x "pick_to_grid -cursor"
```

Deleting Elements

I use this function key more than any other one. Just pass your cursor over a cline, segment, via, text, or shape and hit "d" to delete it. No click of the mouse required! Here is the code for your env file:

```
funckey d "prepopup; pop dyn_option_select @:@Delete"
```

Rotating and Mirroring a Component

You can hit "r" to rotate a component during movement. Here is the code for your env file:

```
funckey r iangle 90
```

Hit "m" to mirror a component during movement. Here is the code for your env file:

```
funckey m "pop mirror"
```

Snapping

While moving an object, you can use a shortcut as opposed to your mouse to snap to various elements:

- To snap to the figure element, hit "f" after implementing this code: funckey f "prepopup; pop dyn_option_select 'Snap pick to@:@ Figure'"

- To snap to the intersection element, hit "i" after implementing this code: funckey i "prepopup;pop dyn_option_select 'Snap pick to@:@Intersection'"

- To snap to the arc/circle center element, hit "c" after implementing this code: funckey c "prepopup;pop dyn_option_select 'Snap pick to@:@Arc/Circle Center'"

- To snap to the via element, hit "v" after implementing this code: funckey v "prepopup;pop dyn_option_select 'Snap pick to@:@Via'"

Assign Commands to the Middle Mouse Wheel

The Button command can be used to assign the Middle Mouse Wheel scroll commands. This command works with SHIFT, CONTROL and SHIFT-CONTROL combinations while scrolling with the mouse in some cases:

- To move the middle mouse wheel up a subclass using the shift button, implement this code: button Swheel_up subclass -+

- To move the middle mouse wheel down an alternative subclass using the shift button, implement this code: button Swheel_down altsubclass -+

- To scroll up with the middle mouse wheel while pressing the control key, implement this code: button Cwheel_up "roam y -$roamInc"

- To scroll down with the middle mouse wheel while pressing the control key, implement this code: button Cwheel_down "roam y $roamInc"

- To scroll up with the middle mouse wheel while pressing the shift and control keys, implement this code: button SCwheel_up "roam x -$roamInc"

- To scroll down with the middle mouse wheel while pressing the shift and control keys, implement this code: button SCwheel_down "roam x $roamInc"

Please contact EMA Technical Support at 877.362.3321 or techsupport@ema-eda.com to learn more about these and other time saving ideas in OrCAD PCB Editor. ●