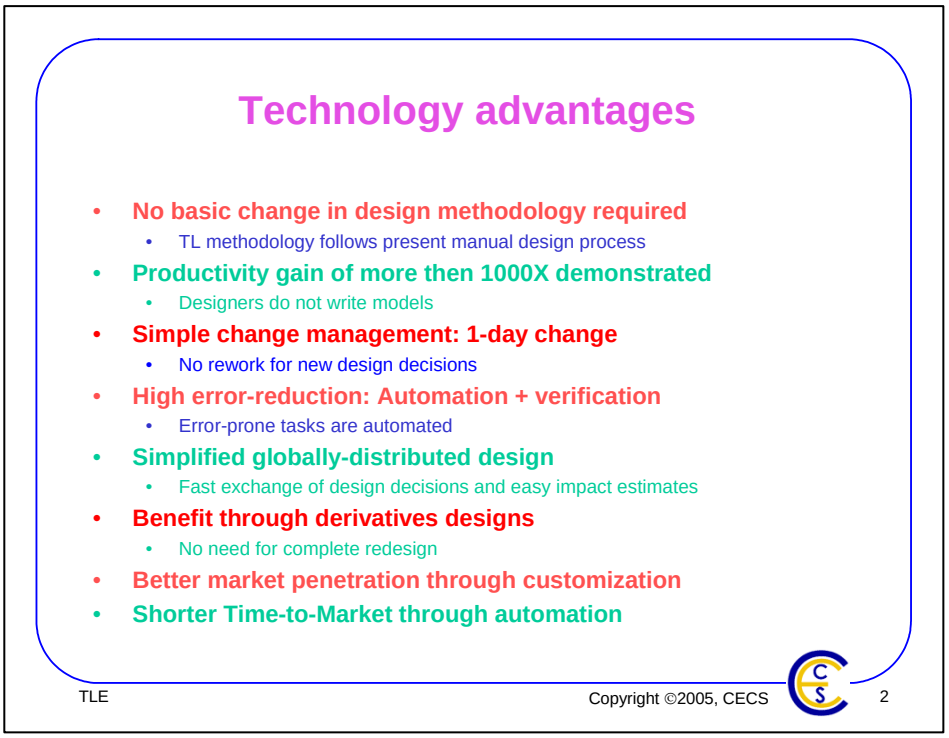
TL Environment

D. Gajski, A.Gerstlauer, R. Doemer, S. Abdi, J. Peng, and D. Shin

Technical Report CECS-05-10
July 30, 2005

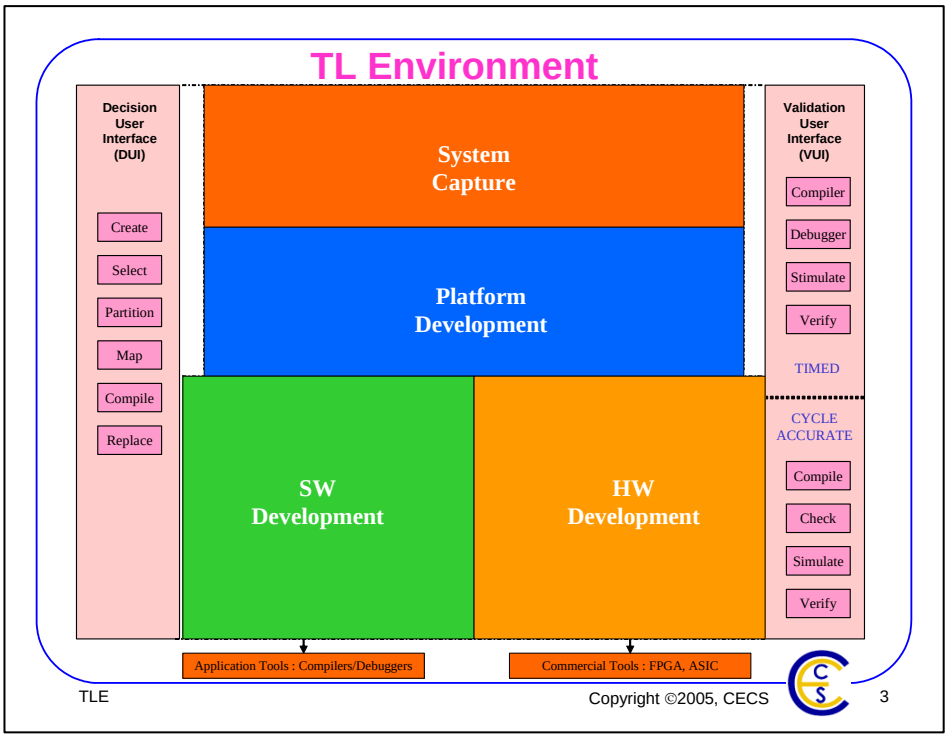
Center for Embedded Computer Systems
University of California, Irvine
Irvine, CA 92697-2425, USA

{gajski, gerstl, doemer, sabdi, pengj, dongwans} @cecs.uci.edu



Technology Advantages

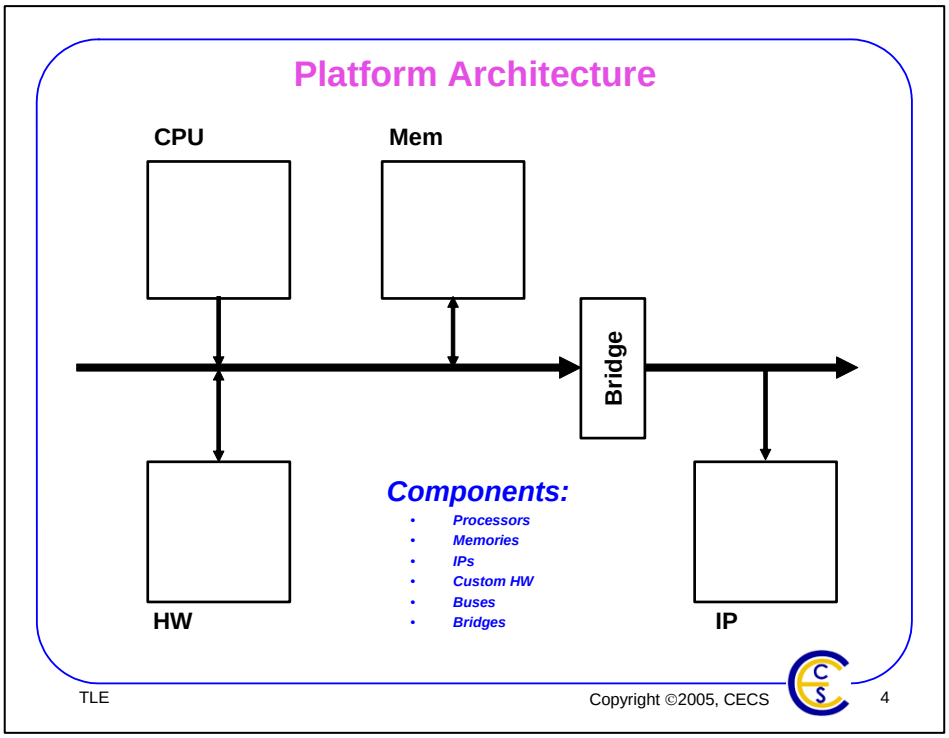
This new TL methodology does not require any changes in the present corporate methodology and offers three orders of magnitude of productivity gain. It also allows simple change management of few hours for small changes and few days for large changes. It reduces bugs since the mundane tasks of generating models and verifying them is automatic. Since all the models and changes are made automatically it is easy to ship those models around the world for design, checking and upgrades. However, the main advantage lies in the fact that every system or product can be easily upgraded with only few days of work. This type of customization allows better market penetration and shorter time-to-market.



TL Environment

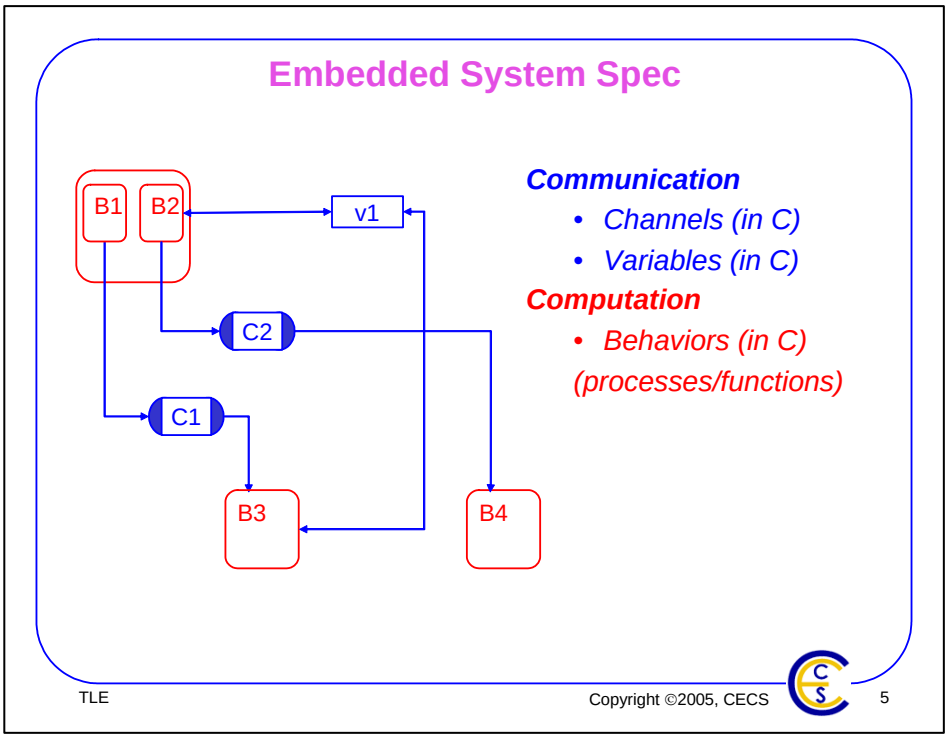
The TLE consists of four basic components supported by two interfaces. **System Capture** is a new graphical user interface for capturing the definition of the platform architecture and product/system specification. **Platform Development** tool generates TL models of the platform architecture running the product specification defined by the capture tool. The **HW Development** component is used to generate cycle-accurate or RTL description of the HW components which can be further refined by commercially available tools for ASIC or FPGA manufacturing. **SW Development** generate firmware necessary to run communication and application SW on the platform.

Validation User Interface is used to debug and validate developed SW and HW. **Decision User Interface** is used by the designer, to estimate the quality metrics and make decisions such as component selection, task scheduling, mapping of SW functions to HW components and others.



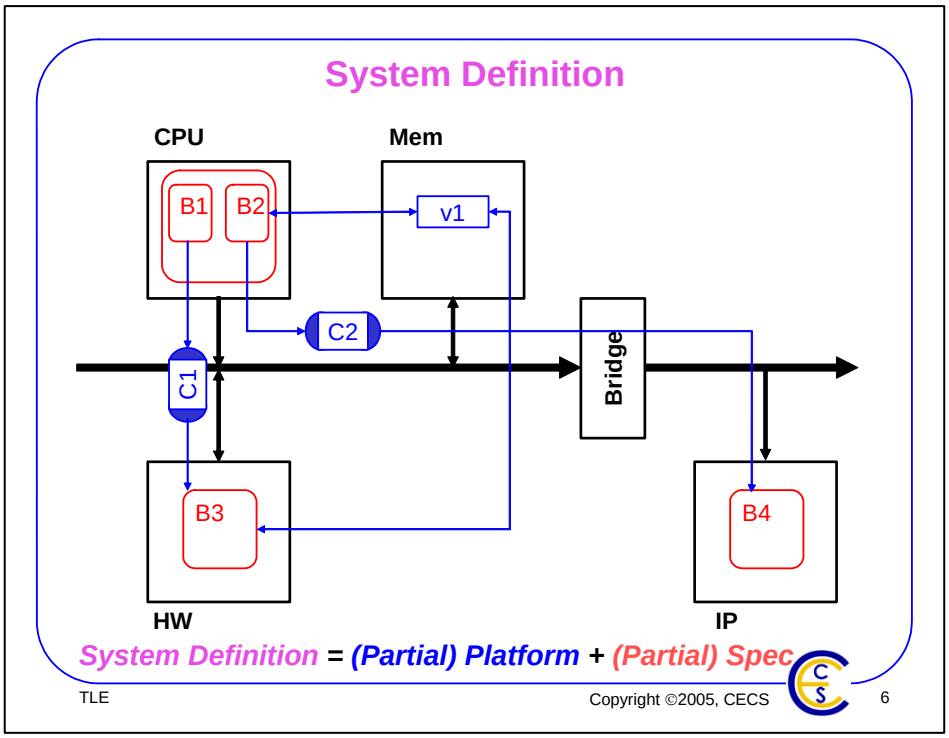
Platform Architecture

Platform architecture consists of set of components and set of connections selected from the library or defined by the user. The platform architecture can be completely or partially defined. More components and connections can be added later. TLE will upgrade the TLM and PAM models automatically to satisfy the new upgrades.



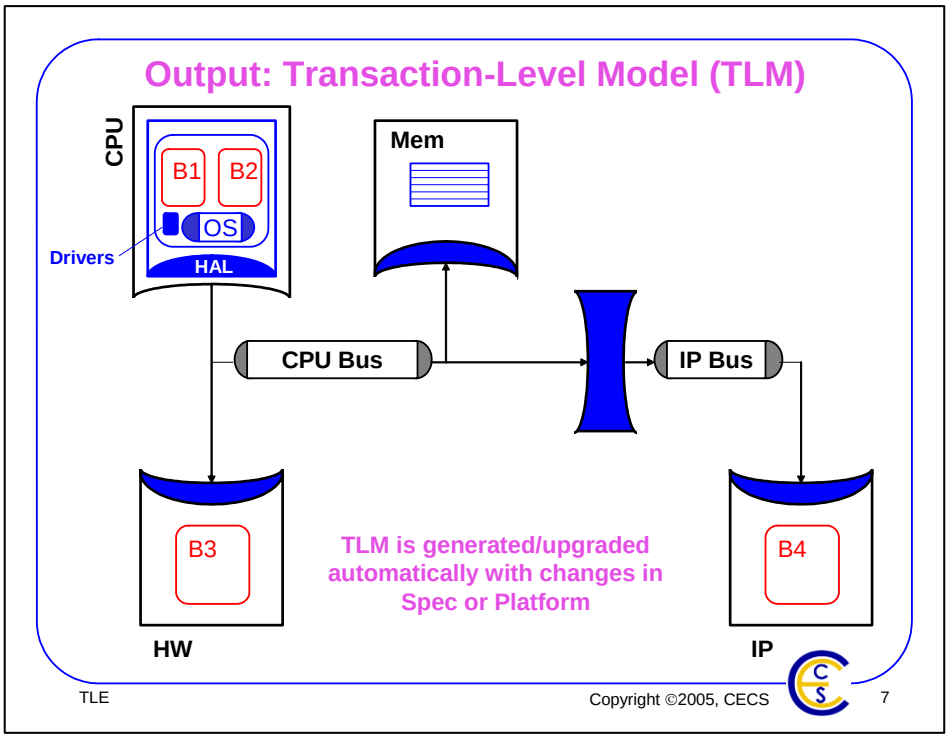
System Specification

System Spec consists of computation components called behaviors (such as processes, tasks or function) in a programming language (such as C) and communication components such as channels or just plain variables also written in a programming language (C) . The Spec can be partial or complete. As the application code is developed it is added as new behaviors to the already existing Spec.



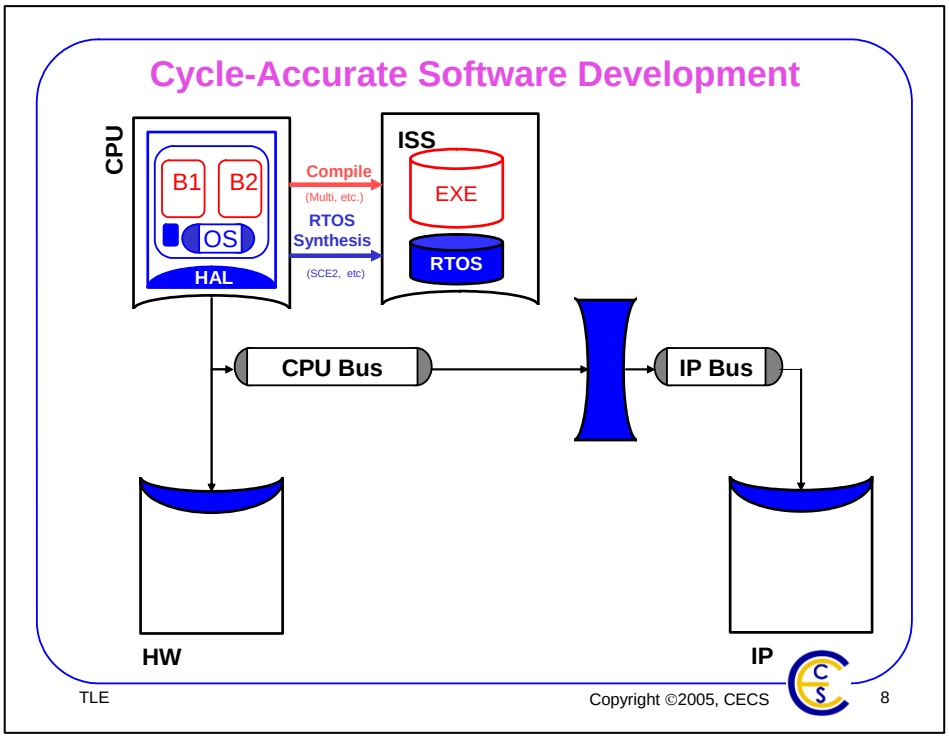
System Definition

The Platform and the Spec combined define the new product. The assignment of specification components (SW) to platform component (HW) is performed by the user. (It can also be performed automatically.) The user can upgrade either HW or SW parts at any time.



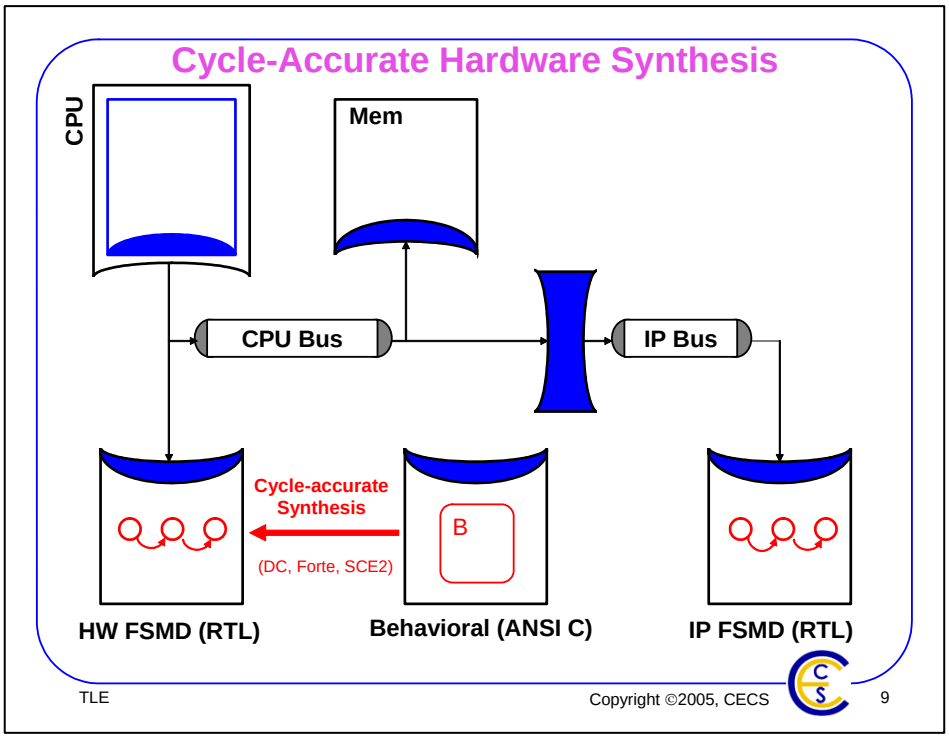
Transaction-Level Model

The TLE generates automatically the transaction-level and pin-accurate models of the system for debugging and validation. This model is used also for application SW development as well as for design of custom HW and interfaces. This way SW and HW can be developed concurrently. It simulates very fast so that productivity of developers increases by an order of magnitude (from days to hours).



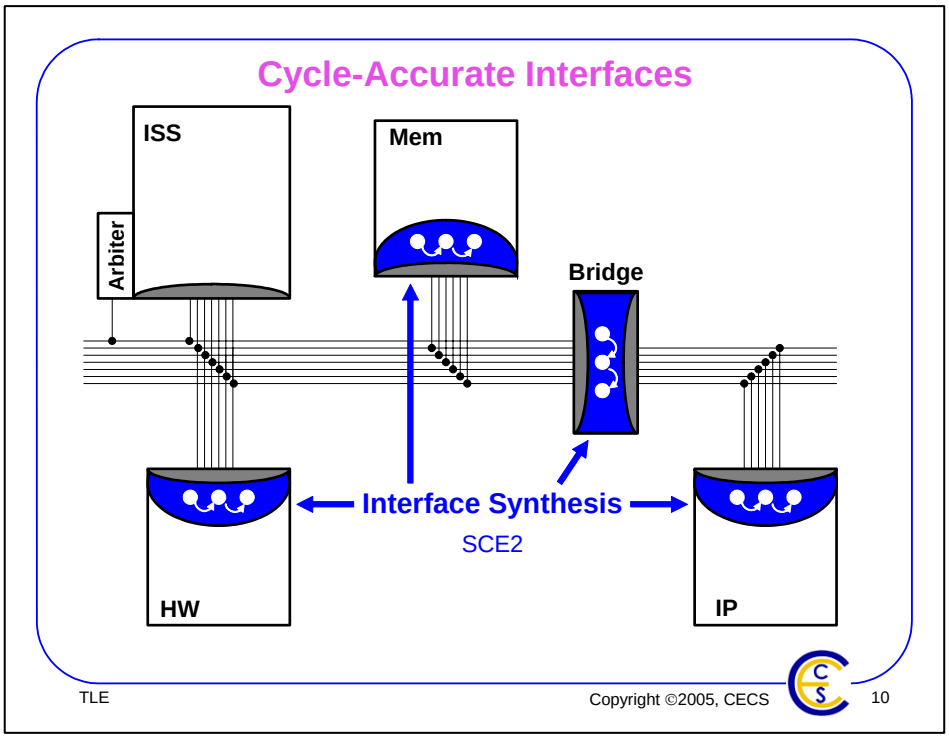
CA SW model

The application SW can be also developed using the commercially available tools, compiled and inserted into the model. TLE will upgrade the model by inserting the Instruction Set Simulator and compiled binaries. Similarly, users can develop custom operating system and insert it into the model. Obviously, CA HW and SW models run much slower and should be inserted selectively.



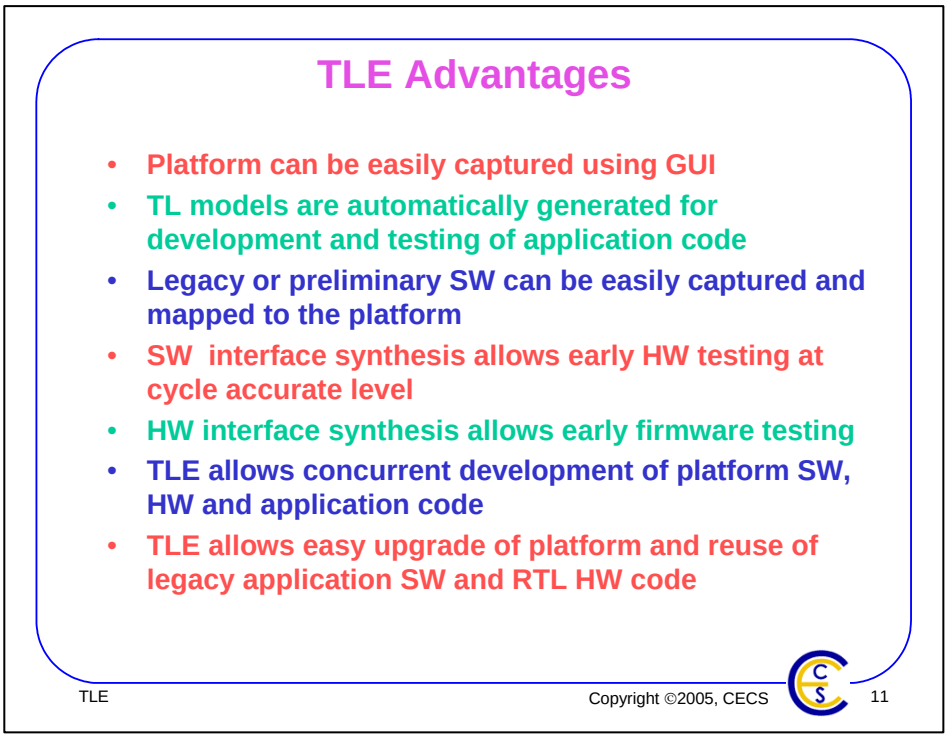
CA HW Synthesis

The HW components can be designed using standard CA synthesis tools or with our HW synthesis tools. The HW models are cycle accurate and can be easily plugged into the TL model when needed. These CA models can be easily inputted to standard, commercially available RTL synthesis tools for ASICs or FPGAs.



Interface Synthesis

In the final step TLE automatically synthesizes the interfaces for communication between components. This way users can test the communication between newly developed CA HW and SW. This is the final step in the development of the system on a chip or a prototype in a FPGA. Once the SW and HW components are debugged and tested on the CA level, users can switch back to TL model for development of application SW. Similarly, upgrading the system could also start on the TL level.



TLE Advantages

There are numerous advantages for using TLE.

The product specification and implementation is easily captured with proprietary GUI. All models are generated automatically after design decisions are made by the users. This saves enormous amount of time in learning modeling languages and writing and interfacing appropriate models. TLE allows parallel development of SW, HW and Application code. In other words, it allows early testing of each thus allowing faster and globally distributed development. Similarly, the upgrades can be easily developed any time and anywhere.