ESL – The Next Leadership Opportunity For India

Alan Naumann
President and CEO, CoWare, Inc.

Competitive Strategies For The Electronics Industry
Agenda

• Introduction to CoWare
• India’s Opportunity
• The SoC Is Becoming The System
  – Embedded Software Challenge
  – Hardware Design Consequences
  – SoC Design Imperatives
• ESL Design & Verification
  – Software-Optimized SoC Architecture Design
  – Software/Hardware Co-Development
  – ESL Connects and Enables the Design Chain
• Algorithm Acceleration In Software And Hardware
  – Software-Optimized Processor Design
  – DSP Algorithm-Optimized Hardware Design
• India VLSI
  – SoC Design Audit
  – India VLSI: What Next?
CoWare: The Largest Private EDA Company

ESL Design Expertise Delivered Worldwide

CoWare’s Mission

• Lead next generation of EDA by uniquely serving S&SoC with system, SoC, processor, SW & algorithm-centric design
• Selected by global leaders; >4000 users in > 90 unique customers; >50% market share
• Largest team in ESL with >120 engineers: >90 R&D & >30 service/support
• 30+ Engineers in India: R&D + local customer support
India’s Opportunity

- India’s software development expertise is the springboard to leadership in SoC design:
  - Software delivers >50% of SoC functionality, and is critical differentiating intellectual property (IP)
  - Software processing and storage needs drive SoC architecture
  - Software developers’ expertise in leveraging system resources is key to the design of performance- and power-optimized SoC

- ESL is a critical enabler in embedded software-driven SoC design:
  - Tightly-coupled software & hardware design from system architecture and algorithm design through chip implementation
  - Optimum level of abstraction for fast design, evaluation and integration of complex IP
Embedded Software Challenge

Embedded software delivers >50% of SoC functionality, and is critical differentiating IP

Source: IBS, November 2002

Software effort ~55% of hardware effort @ 250nm

Software effort ~125% of hardware effort @ 90nm
The chip is becoming the system hardware

- SoC = multiprocessor system with non-trivial memory architectures & communications protocols.
  - Cannot be designed in RTL alone

- System’s silicon content (Gartner):
  - 1970’s: ~5%
  - Today: ~20%
  - Near future: 30% - 40%
Hardware Design Consequences

SoC architecture design = system architecture design

Relative Effort by Designer Role

Source: IBS, November 2002

Negligible architectural design effort @ 250nm

Architectural design effort exceeds physical design effort @ 90nm.
# SoC Design Imperatives

## The value-add is at the front end

<table>
<thead>
<tr>
<th>SoC Design &amp; Verification</th>
<th>Value-Add</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Embedded Systems Design Expertise</strong></td>
<td>▪ Highest productivity SoC design approach. Assures:</td>
</tr>
<tr>
<td>▪ Multi-processor architecture</td>
<td>– Functionality &amp; performance</td>
</tr>
<tr>
<td>▪ Complex algorithm development</td>
<td>– On-time quality SW</td>
</tr>
<tr>
<td>▪ Early embedded SW development</td>
<td>▪ Primary source of defensible differentiation</td>
</tr>
<tr>
<td>▪ HW/SW co-verification</td>
<td></td>
</tr>
<tr>
<td><strong>Intellectual Property</strong></td>
<td>▪ SoC design = IP-based design</td>
</tr>
<tr>
<td>▪ IP integration: system, SW, RTL</td>
<td>▪ IP ownership = more profit &amp; defensible differentiation than mere access</td>
</tr>
<tr>
<td>▪ HW IP: “Horizontal” – processors, memories; “Vertical” – video, wireless</td>
<td></td>
</tr>
<tr>
<td>▪ SW IP: Apps, middleware, RTOS</td>
<td></td>
</tr>
<tr>
<td><strong>Chip Design Expertise</strong></td>
<td>▪ No differentiation</td>
</tr>
<tr>
<td>▪ RTL-to-GDSII design &amp; verification</td>
<td>▪ Excess supply of expertise</td>
</tr>
</tbody>
</table>
ESL Design & Verification

ESL design is more than a level of abstraction above RTL!!!
SW-Optimized SoC Architecture Design

ESL TLM architecture enables:

- Performance analysis to optimize SoC architecture for speed & power, & identify the need for further processing resources:
  - Software-optimized processor(s)
  - Algorithm-optimized hardware
- Memory activity profiling to meet speed and power goals:
  - Identify SW optimization candidates
  - Determine cache architecture & sizes
- Early software development
- HW/SW co-verification ~1,000x faster than C/RTL
- Such modeling & analysis is impossible with RTL within time & budget constraints
Software/Hardware Co-Development Reduces Time-to-Design-Win

Source: STMicroelectronics
ESL Connects And Enables The Design Chain

Enables multi-location design, verification & evaluation by multi-disciplinary teams

• In your company
  – Common executable specification for all groups

• With your customer
  – Early customer design-in
  – Executable specification as design service sign-off

• With your partners
  – Third-party SW support

• Between your customers & their customers
  – Early customer approval
Algorithm Acceleration In SW & HW

**Software**
- Coprocessor augments GP CPU with processing capacity → higher performance
- Easy modification & re-use
- Software-optimized processor bestows:
  - Defensible differentiation
  - IP ownership – no license fees & royalties

**Hardware**
- Algorithm-optimized HW achieves greater speed at less power than a processor
- Often implements a standard-compliant algorithm, e.g. UWB.
- Non-trivial modification & re-use
- Algorithm-optimized HW bestows:
  - Defensible differentiation
  - IP ownership – no license fees & royalties
Software-Optimized Processor Design

- Enables software developers to drive differentiation & optimizations
  - Familiar software environment, tools and debugger
  - Enables design of processor optimized for performance & flexibility
- Enables rapid exploration of instruction alternatives
  - Achieves the optimal performance for the application
  - Trade off extra instructions vs. physical implementation results
- Reduces time to market through automation
Software-Optimized Processor Design

Deploy an application-specific processor synthesis tool

- Slashes processor HW design time by months
- Eliminates engineer-years from SW tool generation effort
- Synthesizes DSP, RISC, SIMD, VLIW & superscalar processors
- Instructions and/or architecture drive synthesis of:
  - Instruction set simulator
  - Software development tools, including C compiler
  - RTL implementation
- “Golden source” model guarantees compatibility of ISS, software tools and RTL implementation

Profile the application

Enter User Defined Instructions in a GUI

Automatic coupling to simulator with user adding new instructions as C macros

Yes

Measure performance

Performance target met?

No

Generate RTL

Verify with Simulation

Automated Capability
DSP Algorithm-Optimized Hardware Design

Nearly one-third of designs deploy two or more DSP Processor Core and DSP Accelerator.

Deploy DSP algorithm-specific IP design & verification tool

- Floating point & fixed point design and simulation
- HW/SW partitioning
- Library of customizable DSP algorithms for communications & multimedia applications.
- Library of standards-compliant algorithms, such as 3G W-CDMA, GSM/EDGE, IS-95 CDMA, IEEE 802.11/a/b/g Wireless LAN, Bluetooth, and UWB.
- Micro-architecture libraries for RTL implementation.
## India VLSI: SoC Design Audit

India can command the SoC design flow from the top

<table>
<thead>
<tr>
<th>SoC Design &amp; Verification</th>
<th>India’s Position</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Embedded Systems Design Expertise</strong></td>
<td></td>
</tr>
<tr>
<td>- Multi-processor architecture</td>
<td></td>
</tr>
<tr>
<td>- Complex algorithm development</td>
<td></td>
</tr>
<tr>
<td>- Embedded software development</td>
<td></td>
</tr>
<tr>
<td>- HW/SW co-verification</td>
<td></td>
</tr>
<tr>
<td><strong>Intellectual Property</strong></td>
<td></td>
</tr>
<tr>
<td>- IP integration: system, SW, RTL</td>
<td></td>
</tr>
<tr>
<td>- HW IP: “Horizontal” &amp; “Vertical”</td>
<td></td>
</tr>
<tr>
<td>- SW IP: Applications, middleware, RTOS</td>
<td></td>
</tr>
<tr>
<td><strong>Chip Design Expertise</strong></td>
<td></td>
</tr>
<tr>
<td>- RTL-to-GDSII design/verification</td>
<td></td>
</tr>
<tr>
<td><strong>India’s Position</strong></td>
<td></td>
</tr>
<tr>
<td>- World class embedded software development and hardware modeling expertise</td>
<td></td>
</tr>
<tr>
<td>- Strong architecture and algorithm research</td>
<td></td>
</tr>
<tr>
<td>- World class integration expertise</td>
<td></td>
</tr>
<tr>
<td>- Adequate IP access</td>
<td></td>
</tr>
<tr>
<td>- Inadequate IP ownership - little defensible differentiation</td>
<td></td>
</tr>
<tr>
<td>- Especially difficult for design services companies</td>
<td></td>
</tr>
<tr>
<td>- World class expertise</td>
<td></td>
</tr>
</tbody>
</table>
India VLSI: The Leapfrog Strategy

The Profitable Objective

• Lead SoC & IP design by leveraging world class SW development expertise

• Reap profits & establish defensible differentiation - define, implement & OWN application-specific architectures, IP & algorithms

• Drive ESL design & verification methodology into your organizations RIGHT NOW!

The Leapfrog Strategy

2005: Advanced

Leapfrog!