

# **HIGH PERFORMANCE SYSTEM MODELING**

## **A**

### **VHDL BASED METHODOLOGY**

**Paul Clemente, Michael Hajj, Wayne Jessop**  
**Raytheon Company**  
1001 Boston Post RD  
Marlborough, MA. 01752  
(508) 490 - 3902

#### **ABSTRACT**

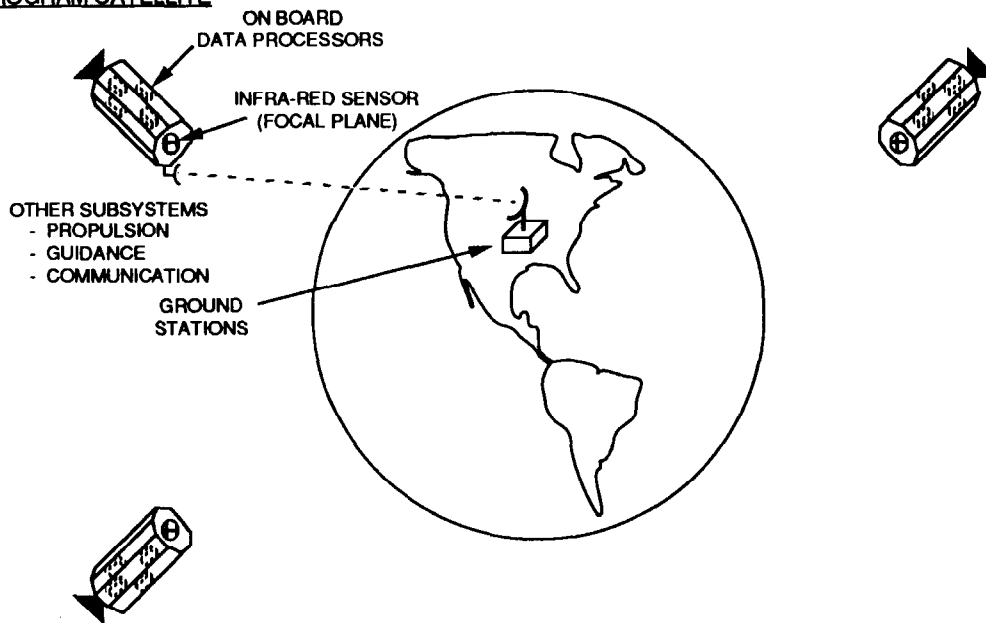
With increasing emphasis in the design environment being placed on requirement flow down and traceability through the design hierarchy, new methodologies are evolving to meet this goal. These new methodologies are based on a shift to a new design paradigm that no longer is based on gate level design, but is now based on requirements analysis, behavioral design, logic synthesis, hardware acceleration and computer aided hardware prototyping as the foundation.

Over the past several years Raytheon has been involved in creating a design environment that supports top down modeling of system requirements and creating executable specifications at the behavioral and RTL levels to meet the DoD standards initiative. This allows design tradeoffs and spec. compliance to be validated at the highest level. Raytheon has explored a few different ways of generating VHDL compliant code. One method uses manually generated, RTL level, synthesizable code. Another method uses statecharts to provide a graphical entry and simulation environment which may then be automatically translated into RTL level VHDL. Associated testbenches may also be generated in the same fashion for either method. These methods generate correct and synthesizable code targeted to any one of several synthesis vendors. VHDL then forms the software backplane for the rest of the seamless design environment. This hardware description language is used as the input to the logic synthesis tool to automatically generate a gate level implementation. These gate level models are simulated, verifying functional equivalence to the behavioral model. This is accomplished using a standard workstation or accelerated on a hardware accelerator to accommodate larger, more complex problems. Additionally, computer aided hardware prototyping systems are used to generate functionally equivalent, gate level, hardware implementations. The verification of the design implementation can be extended from ASIC and module level to operating system and application software execution.

Raytheon has helped to develop some of these tools as well as applying this methodology to several different projects. This paper will focus on this top down design methodology, the heterogenous EDA tools used to create this seamless environment and highlight a few test cases focusing on this new design paradigm.

SYSTEM DIAGRAM

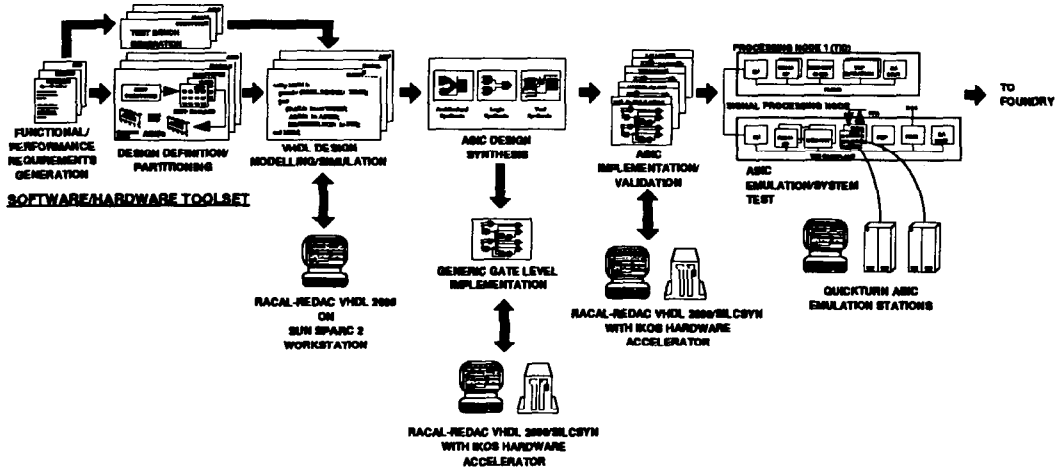
PROGRAM SATELLITE



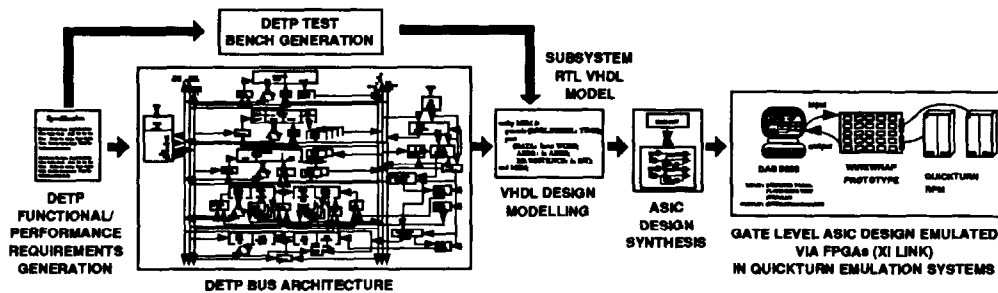
PROGRAM DESIGN CHALLENGES

- ❑ Model Complete System in VHDL.
  - ❑ 12 Different ASICs Targeting 4 Different Foundries.
  - ❑ All ASICs Should Be Implemented in RAD-Hard Submicron Fabrication (Down to 0.5 micron).
  - ❑ ASICs Must Be Testable Within Their Own Module/Subsystem Environments.
  - ❑ Real Sensor Data is Used as Stimulus to Validate & Verify ASIC/System Design.
  - ❑ Tight Time Schedule (12 to 18 Months).
- ❑ These Challenges Could Not be Met Without Top-Down VHDL Based Methodology.

### ASIC VHDL DEVELOPMENT METHODOLOGY



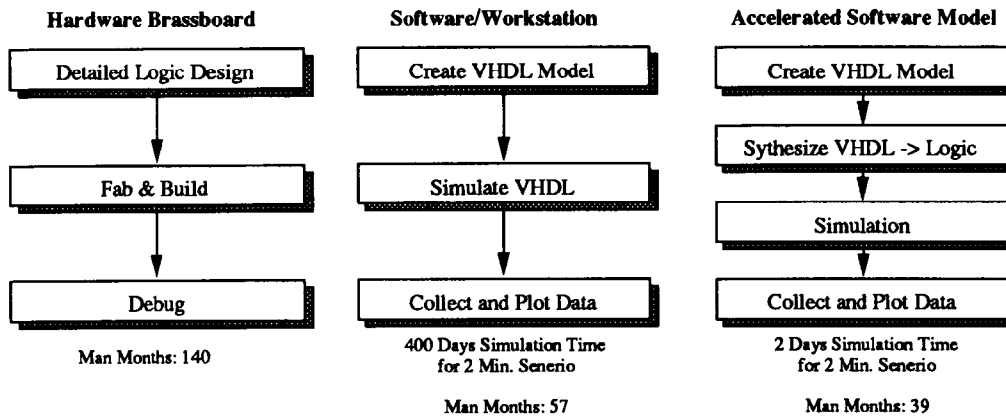
### METHODOLOGY PATHFINDING DEMONSTRATED



## RAYTHEON VHDL EXPERIENCE 63K GATE DETECTION PROCESSOR

In 1991 Raytheon Demonstrated the Following Objectives:

- ☐ VHDL Modeling and Simulation was a Viable Alternative to Hardware Prototyping.
  - 3x Cost, 2x Prototype Improvement on Comparable Prototype.
  - Flexibility in Model Allows Architecture Iteration.
- ☐ VHDL Synthesis & Acceleration can Achieve Rapid ASIC Implementation & Verification.



## VHDL CAD TOOLS EVALUATION

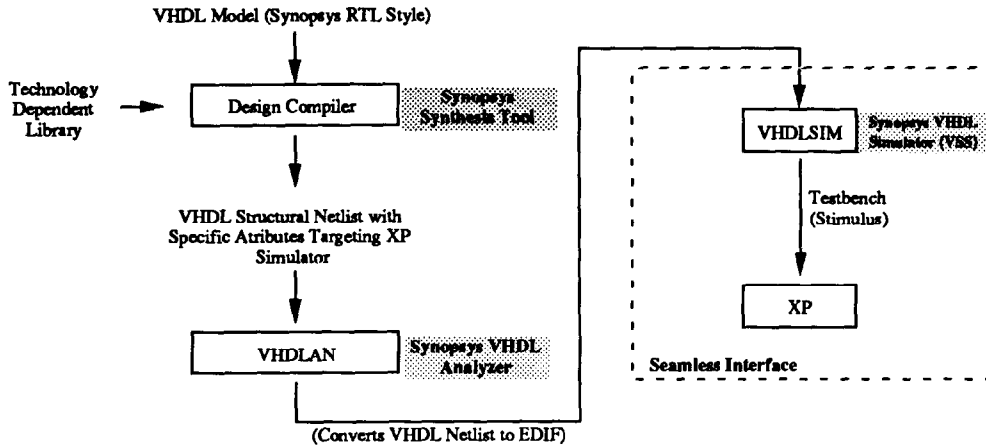
- ☐ In 1992, Hardware Accelerator Vendors Introduced Seamless Support for VHDL.
- ☐ Both Zycad and IKOS Introduced Top-Down Design Environments using VHDL for System and ASIC Simulation With These Claims:
  - Seamless and Integrated Interface to Accelerate VHDL Simulation (No Manual Work).
  - VHDL Testbenches can be Used for Acceleration.
  - Source Level Debugging and Signal Traceability.
  - Both Use Front End VHDL Simulators and Synthesis Tools to Achieve VHDL Acceleration.
    - + Zycad Uses Synopsys VHDL Simulator (VSS) and Synthesis Tool (Design Compiler).
    - + IKOS Uses RACAL-REDAC VHDL Simulator (VHDL 2000) and Synthesis Tool (SilcSyn).
- ☐ To Meet FEWS Challenges, Raytheon Evaluated Both VHDL Environments With Their Respective Synthesis Tools.

### Criteria:

- Simulation Speed Achieved (Main Criteria).
- Accuracy Of Simulation.
- Capability Of Handling Large Size Designs (Subsystems) with Reasonable Turn Arounds.
- How Seamless and Integrated Each Vendors' Interface to Accelerator (Human Interface).
- Low Level Synthesis Capabilities (Gate Level Synthesis).
- Vendor Support.

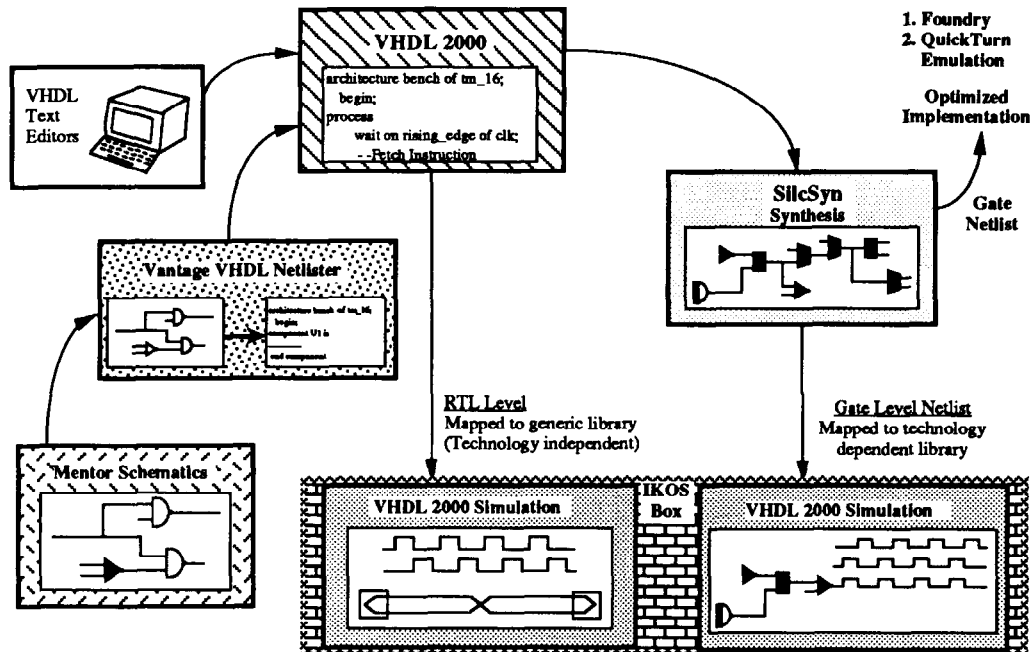
## ZYCAD VHDL DEVELOPMENT

- ❑ Zycad Provides Interface to Their XP Simulation Accelerator Through Synopsys VHDL Simulator (VSS 2.2).
- ❑ The Link is Transparent.
  - The XP is Transparently Controlled from VSS.
  - Designs Residing on the XP Look Like a Design Being Simulated in VSS.



## IKOS VHDL ENVIRONMENT

- ❑ IKOS Provides Interface to Their Accelerator Box Through RACAL-REDAC's VHDL Simulator (VHDL 2000).
- ❑ Link Is Transparent.
  - IKOS is Transparently Controlled from VHDL 2000.
- ❑ IKOS Provides 2 Distinctive Accelerations:
  - VHDL RTL Acceleration.
  - Gate Netlist Acceleration (Structural).
- ❑ VHDL RTL Acceleration is Performed Through the Integration Of RACAL-REDAC's Microarchitectural Synthesis with the IKOS Hardware Accelerator.
  - Microarchitectural Synthesis (High Level Synthesis) is a Unique Feature in RACAL-REDAC's Synthesis (Compiling RTL VHDL into RTL Primitives).
  - RTL Structure is Simulated on IKOS Without the Loss of the User Interface Function.
    - + Source Debugging is Performed From the VHDL Source Code, Not Gate-level!
    - + Breakpoints Can Be Set in the Source Code.
    - + Tracing Signals Can Be Accomplished Even if Described as an Enumerated Type.

PROGRAM VHDL DESIGN  
TOOL SUITE

## SUMMARY

- ☐ VHDL Based Top Down Design - Bottom Up Verification Methodology Developed and Demonstrated.
- ☐ Methodology Employs VHDL Modelling -> Logic Synthesis -> Hardware Acceleration -> ASIC Emulation Tool Set.
- ☐ Methodology is a Viable Alternative to Hardware Prototyping.
- ☐ Methodology is Currently Being Applied to Major DoD Program.