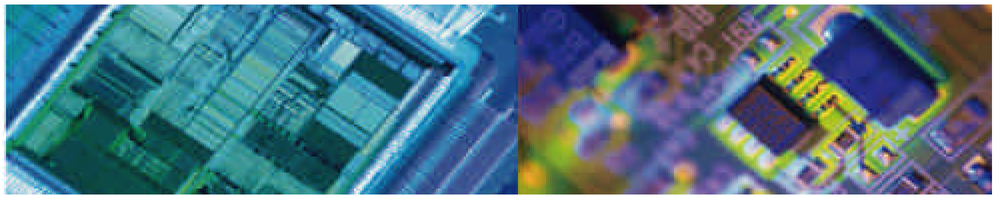
MATHSTAR™

Fast Silicon, Faster Solutions



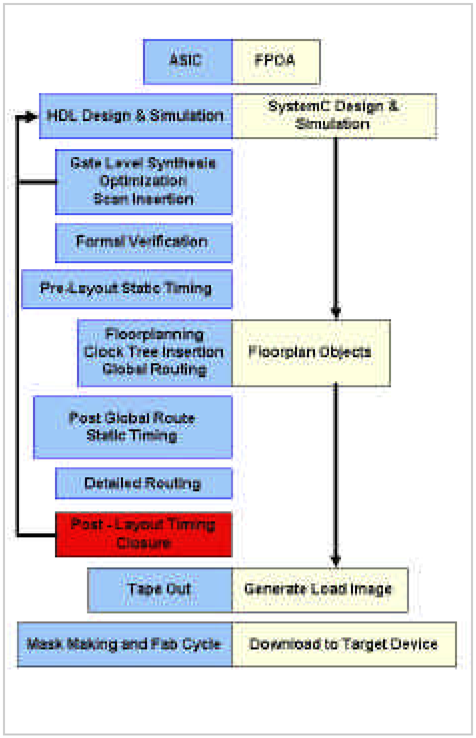
Tools Overview

Software Tools

High performance solutions with the lowest cost of ownership

Features

- Graphical and C language based design with EDA partners
- Verilog-like OHDL language based design
- No synthesis or gate level timing closure
- Cycle accurate simulation model support for
  - Summit Visual-Elite™
  - Synopsys VCS™
  - Cadence NC Verilog™
  - Mentor ModelSim™
- Mathstar supplied floorplanning, compilation, and debugging tools



ASIC
FPGA
HDL Design & Simulation
SystemC Design & Simulation
Gate Level Synthesis Optimization Scan Insertion
Formal Verification
Pre-Layout Static Timing
Floorplanning Clock Tree Insertion Global Routing
Floorplan Objects
Post Global Route Static Timing
Detailed Routing
Post - Layout Timing Closure
Tape Out
Generate Load Image
Mask Making and Fab Cycle
Download to Target Device

Figure 1. *Design Process*

Overview

The Mathstar Silicon Object development flow is a dramatic simplification from conventional ASIC or FPGA development flows.

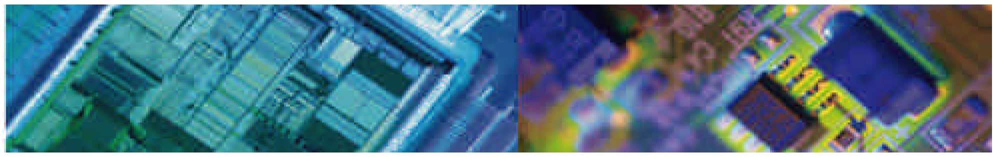
ASIC chip designers typically toil for months with complex tool chains from many vendors. Often teams appear to be spending as much time in tool chain maintenance as they do in actual design. The flow typically includes large numbers of iterations on system models, RTL models, gate models, library maintenance tasks, synthesis runs, floorplanning exercises, block connection, timing closure iterations, inter-block routes, global timing closure, clock tree design, and finally DRC/LVS rule checking. In recent years, reaching timing closure for both ASICS and high-end FPGA devices has become a significant project schedule risk and a major resource commitment. Months of effort are now common to reach performance and design density requirements.

Simplifying the Process

In contrast to FPGAs and ASICs, the vastly simplified flow from MathStar utilizes C combined with a Verilog-like hardware description language called OHDL, compiler, floorplanner, and powerful EJTAG chip debugger. The tools are designed to enable small teams to quickly design, program, verify, and debug algorithms and protocols on Silicon Object-based devices. Abstract representations are made in commonly available graphical or textual EDA modeling tools. That representation is then translated and mapped into the pre-timed hardware resources of the Silicon Objects. Unlike ASIC and FPGA designs, communication paths are chosen based on system clock cycle timing requirements, not on gate level propagation. In this way, tedious iteration to achieve timing closure is avoided. The code is loaded onto the array from a PROM or JTAG very similar to the methods used by FPGAs today. The simplicity of the design tools process allows changes to be incorporated on demand without extensive gate level timing implications.

The tools, programming model, and programming process for a Mathstar Silicon Object device are both revolutionary and easy to comprehend. Significant time and cost savings over conventional ASIC and FPGA development processes are realized. No synthesizers, timing closure issues or extensive million-gate route bottlenecks are required to realize extremely high-performance, functionally dense designs.

MATHSTAR™
Fast Silicon, Faster Solutions



OHDL Code Example

```
module fifo128x512s1 (
    in bit reset,
    in bit re,
    in bit we,
    in word addr, word depth ) {
    nets {
        bit rd_cntr_en,
        bit wr_cntr_en,
        word rd_addr_cntr,
        wr_addr_cntr;
    }
    // instances
    addr_ctrl wr_addr(
        clock_enable => wr_cntr_en,
        reset => reset,
        count => wr_addr_cntr
    );
    addr_ctrl rd_addr(
        clock_enable => rd_cntr_en,
        reset => reset,
        count => rd_addr_cntr
    );
}   // fifo128x512s1

Alu module addr_ctrl(
    in bit reset,
    in bit clock_enable,
    out-local word count
) {
    Initial
        count = 0x00ff;
        K0 = 0x01ff;
    Truth Tables
        while(1) {
            if (reset) {
                XOR count, count;
            }
            if(!clock_enable) {
                count = PASS count;
            } else {
                count = UCW count, K0;
            };
        };
};
```

Mathstar Supplied Tools and Languages

OHDL

OHDL is a Hardware Description Language structurally similar to Verilog. Like Verilog, it contains structural and behavioral modules. Communications are effected through ports and signals with arguments in named sets. The language supports hierarchies, macros, and libraries to enable either top-down decomposition or bottoms-up design styles. Coding OHDL is like working with a subset of Verilog RTL, and is much simpler than coding for gate level synthesis. OHDL is a text based source level language that easily integrates most code control systems. OHDL exposes the architecture of the underlying hardware directly giving the designer complete control of resources and features. The language is essentially an API to the hardware, coded as an RTL level HDL. Because of this, there is no need for complex and expensive tool chains to support gate synthesis or gate level timing closure.

COAST

COAST is a graphical floorplanning editor that assigns functional modules to physical resources on the Silicon Objects array. COAST includes an interactive GUI and powerful analysis and guidance software to assist the user. In addition to floorplanning, COAST enables designers to alter, upgrade, or optimize existing designs as requirements change during the design process. In COAST, the user is presented with graphical representations of the design hierarchy and the array resources. The user selects hierarchical instances from the design tree and assigns them to a grid representing the object array elements in a drag-and-drop fashion. Fly lines and other visual cues as well as automation support enhance user productivity.

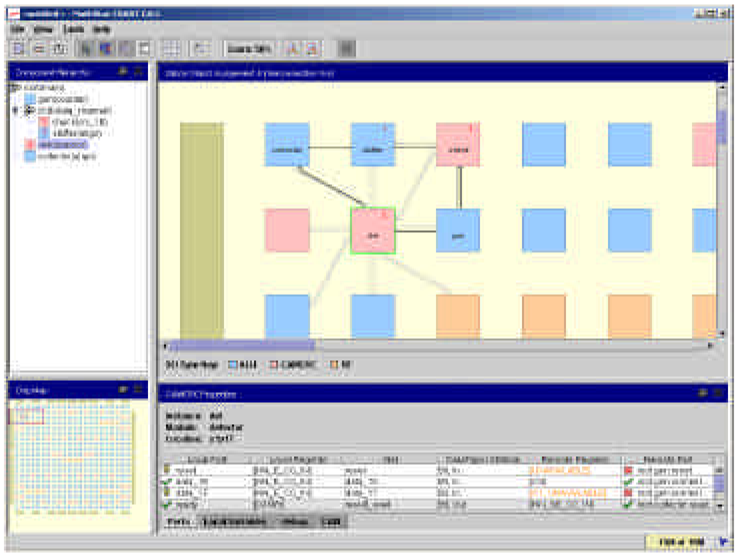


Figure 2. *COAST Diagram*

Contact Us...

Corporate Headquarters
5900 Green Oak Drive
Minneapolis, MN 55343
P: 952.746.2200

Sales Office
1901 Wright Place
Carlsbad, CA 92008
P: 760.602.5250

Website
info@mathstar.com
www.mathstar.com