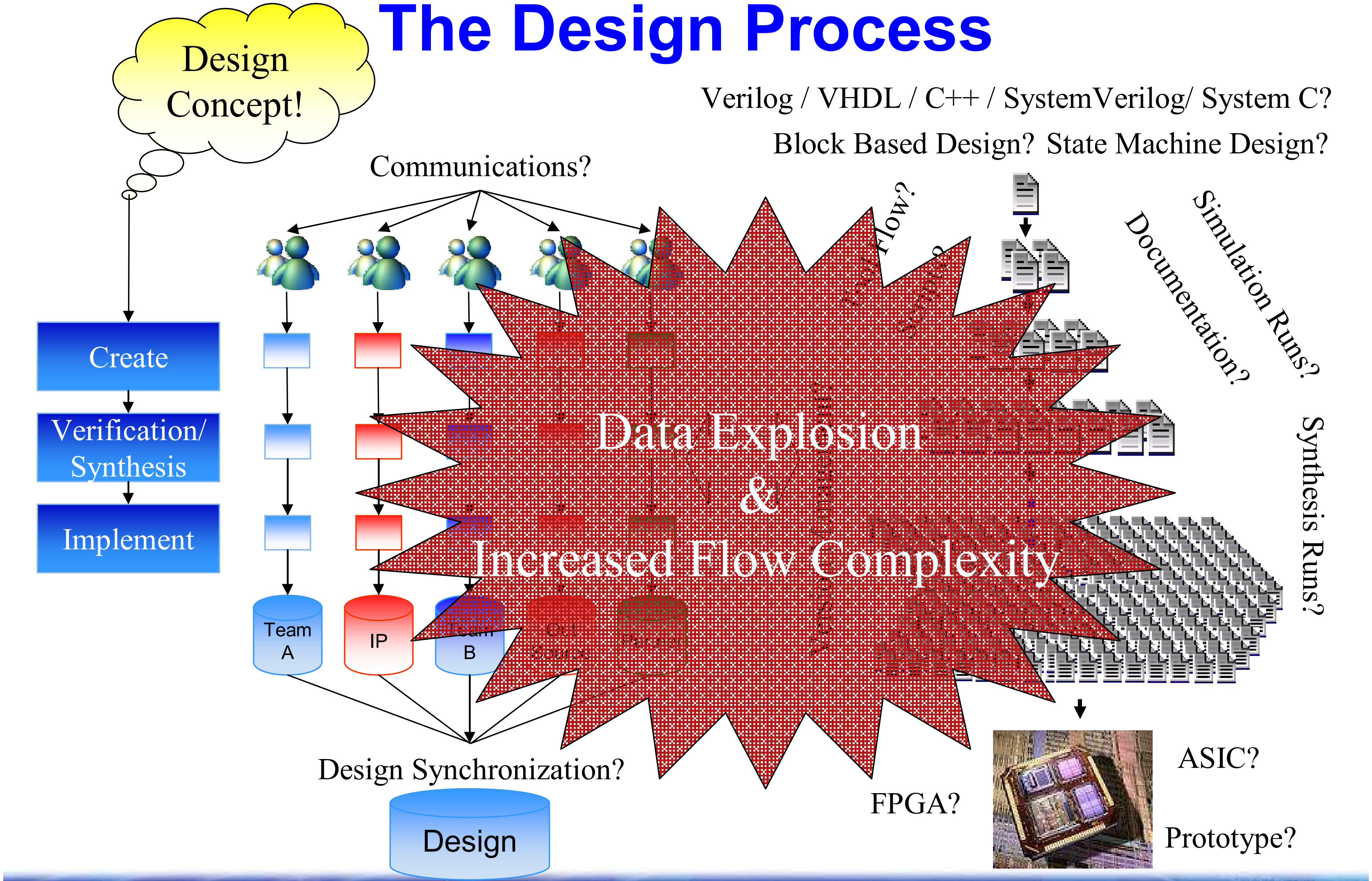
ASICentrum
Accelerate your RTL design with HDL Designer & DesignChecker

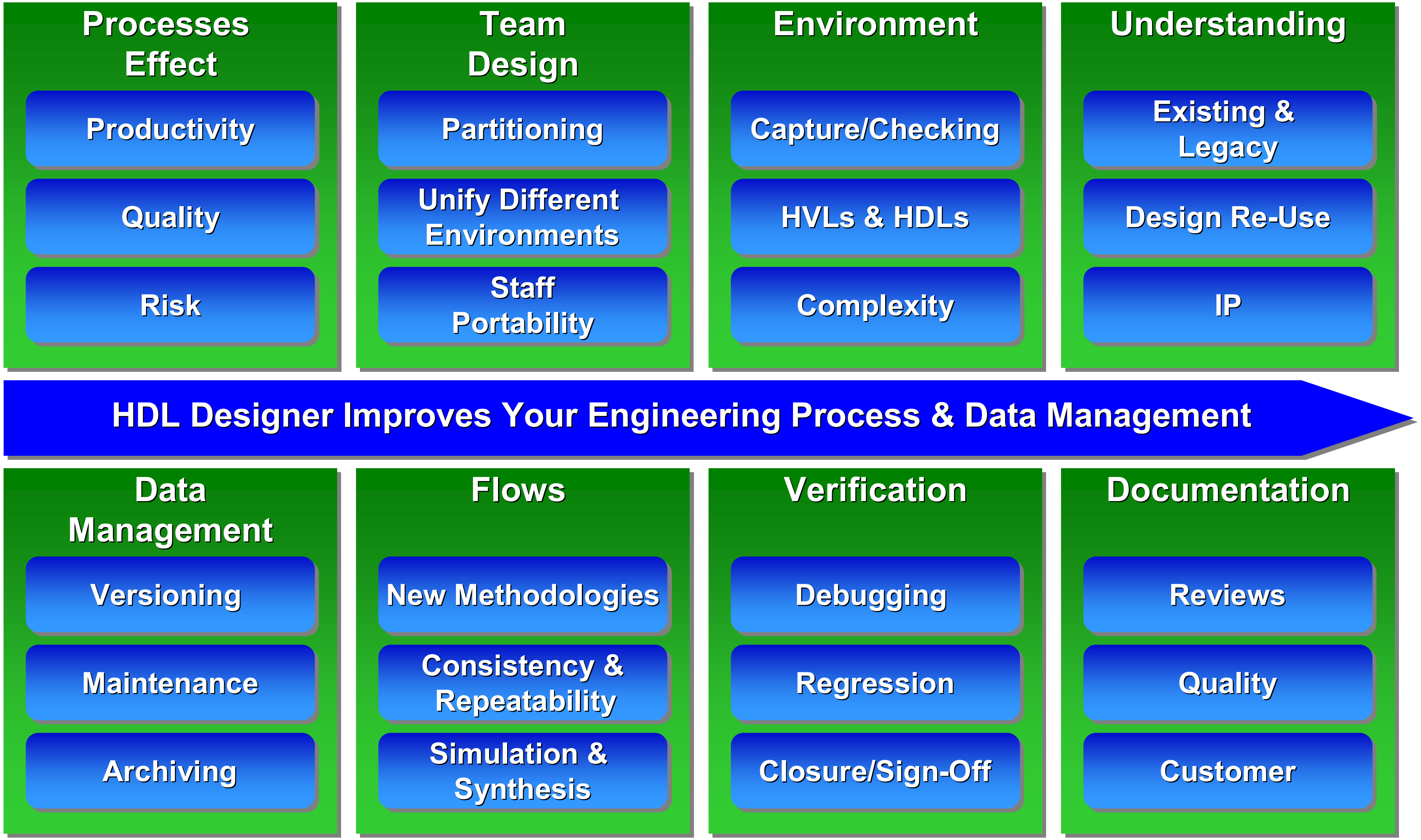
Ian Traynor – Technical Marketing

Mentor Graphics®

The Design Process



Can your processes be improved?



Processes Effect
- Productivity
- Quality
- Risk

Team Design
- Partitioning
- Unify Different Environments
- Staff Portability

Environment
- Capture/Checking
- HVLs & HDLs
- Complexity

Understanding
- Existing & Legacy
- Design Re-Use
- IP

HDL Designer Improves Your Engineering Process & Data Management

Data Management
- Versioning
- Maintenance
- Archiving

Flows
- New Methodologies
- Consistency & Repeatability
- Simulation & Synthesis

Verification
- Debugging
- Regression
- Closure/Sign-Off

Documentation
- Reviews
- Quality
- Customer

Quotes & statements to consider

Major European Chip Manufacture - ASIC Design Manager

"How do you manually interact with 1M lines of code and 10,000 related design files? Tools like HDS are critical in our design flow to access and manage design data"

PHILIPS

"..We have worked with Mentor to customize HDS to implement our Core ReUse Methodology."

EETIMES

THE INDUSTRY SOURCE FOR ENGINEERS & TECHNICAL MANAGERS WORLDWIDE

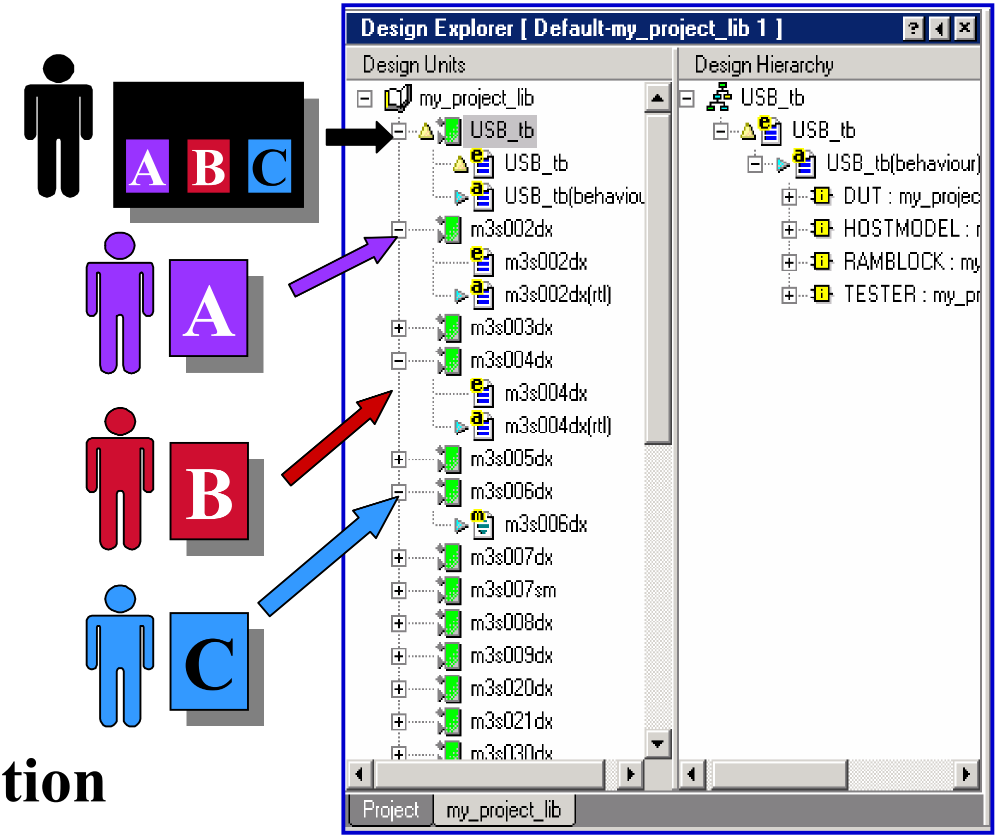
Over a third of all high-end ASIC designers now use FPGAs for prototyping 500,000-plus-gate designs.

CMP EEdesign An EE Times Community

"…Design managers report that at least 50% of their engineering resource is devoted to managing the push of design data through the tools in their flows." – **1/4/05**

Team Design

- Better use of Libraries
- Bottom-Up & Top-Down Design Styles
- Text and/or Graphics
- Environment & Flows Common
- Quality Design Documentation



Design Explorer [Default-my_project_lib 1]

Design Units

```
my_project_lib
  USB_tb
    USB_tb
    USB_tb(behaviou
  m3s002dx
    m3s002dx
    m3s002dx(rtl)
  m3s003dx
  m3s004dx
    m3s004dx
    m3s004dx(rtl)
  m3s005dx
  m3s006dx
    m3s006dx
  m3s007dx
  m3s007sm
  m3s008dx
  m3s009dx
  m3s020dx
  m3s021dx
  m3s030dx
```

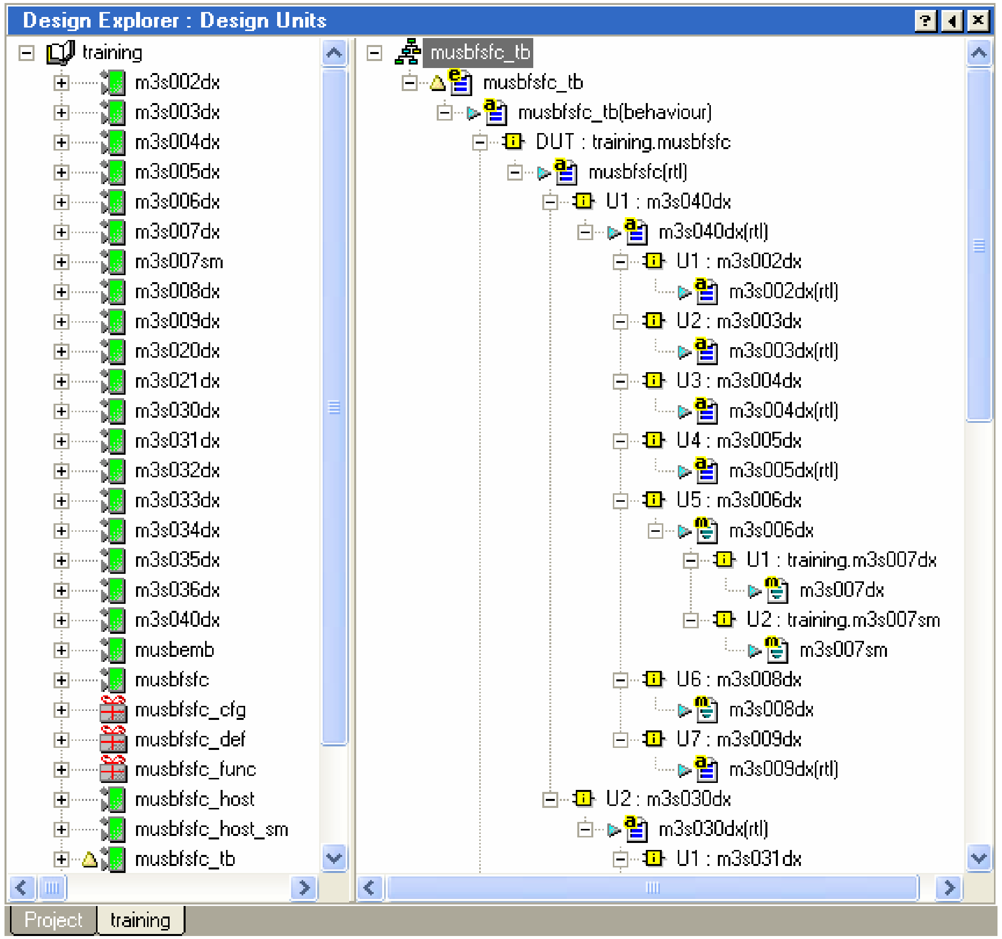
Design Hierarchy

```
USB_tb
  USB_tb
    USB_tb(behaviour)
      DUT : my_projec
      HOSTMODEL : r
      RAMBLOCK : my
      TESTER : my_pr
```

Project | my_project_lib

Understand & Manage Design Database

- Import/reference mechanisms
- HDL code parser
- Hierarchy understood
  - Unit/Hierarchy View
  - Cross Referencing/Highlight
  - Searches Find/Where used
  - VHDL Configurations
- Easy design navigation
  - Pure Text and/or Graphics
  - Rendering of Text2Graphics

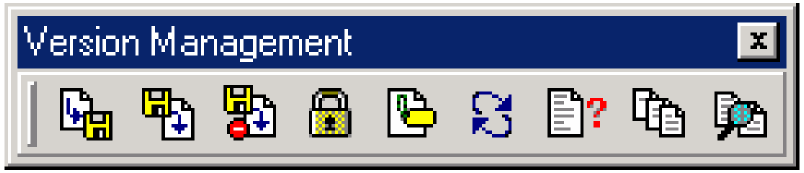


```
training
  m3s002dx
  m3s003dx
  m3s004dx
  m3s005dx
  m3s006dx
  m3s007dx
  m3s007sm
  m3s008dx
  m3s009dx
  m3s020dx
  m3s021dx
  m3s030dx
  m3s031dx
  m3s032dx
  m3s033dx
  m3s034dx
  m3s035dx
  m3s036dx
  m3s040dx
  musbemb
  musbfsfc
  musbfsfc_cfg
  musbfsfc_def
  musbfsfc_func
  musbfsfc_host
  musbfsfc_host_sm
  musbfsfc_tb
```

```
musbfsfc_tb
  musbfsfc_tb
    musbfsfc_tb(behaviour)
      DUT : training.musbfsfc
        musbfsfc(rtl)
          U1 : m3s040dx
            m3s040dx(rtl)
              U1 : m3s002dx
                m3s002dx(rtl)
              U2 : m3s003dx
                m3s003dx(rtl)
              U3 : m3s004dx
                m3s004dx(rtl)
              U4 : m3s005dx
                m3s005dx(rtl)
              U5 : m3s006dx
                m3s006dx
                  U1 : training.m3s007dx
                    m3s007dx
                  U2 : training.m3s007sm
                    m3s007sm
              U6 : m3s008dx
                m3s008dx
              U7 : m3s009dx
                m3s009dx(rtl)
          U2 : m3s030dx
            m3s030dx(rtl)
              U1 : m3s031dx
```

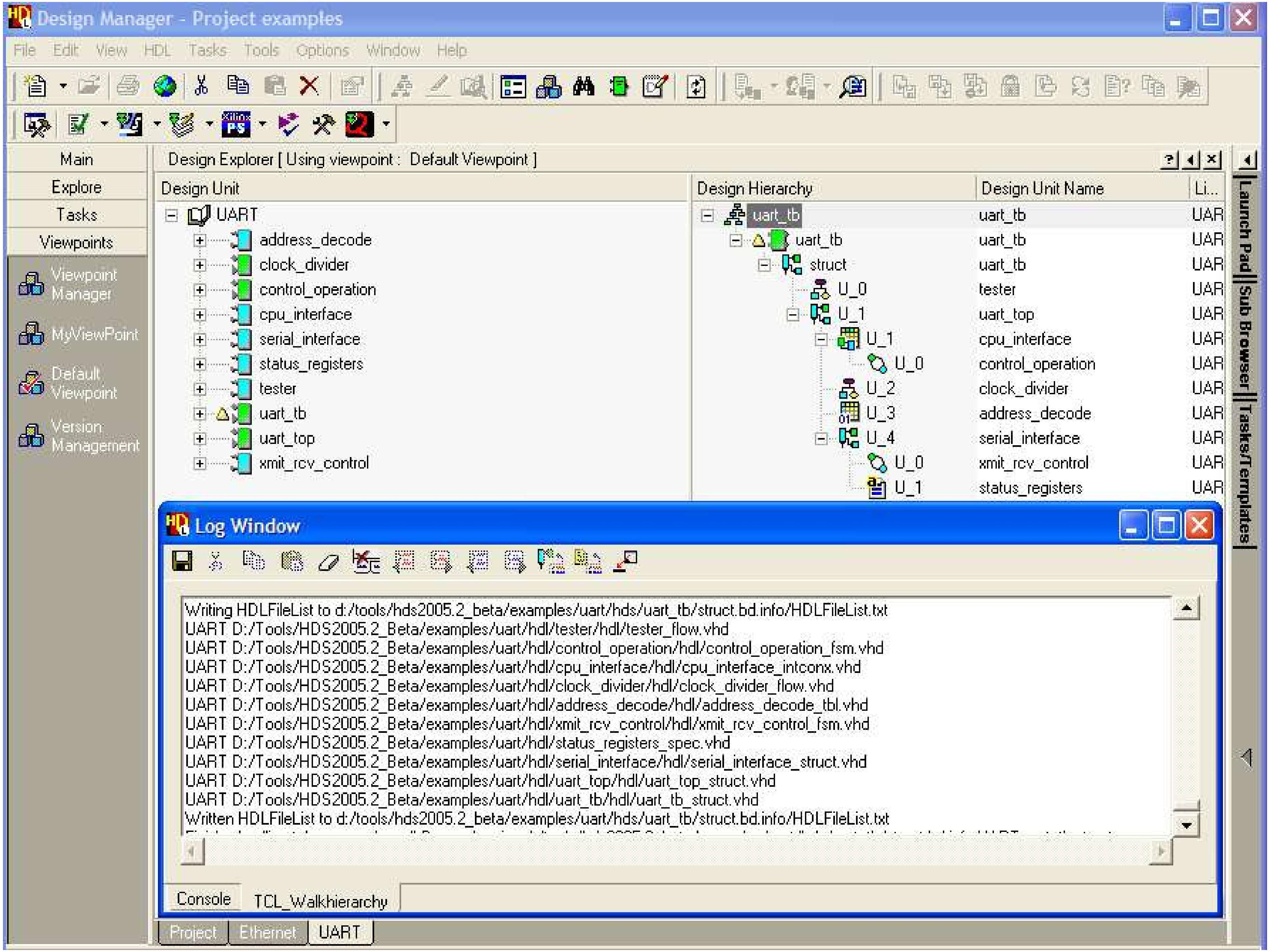
- Engineering Logbook
- Store relevant Documents/Scripts/Programs within Environment

Version Management



Check In, Check Out, Get, Change Lock,
Label, Synchronize, Status, History, Compare

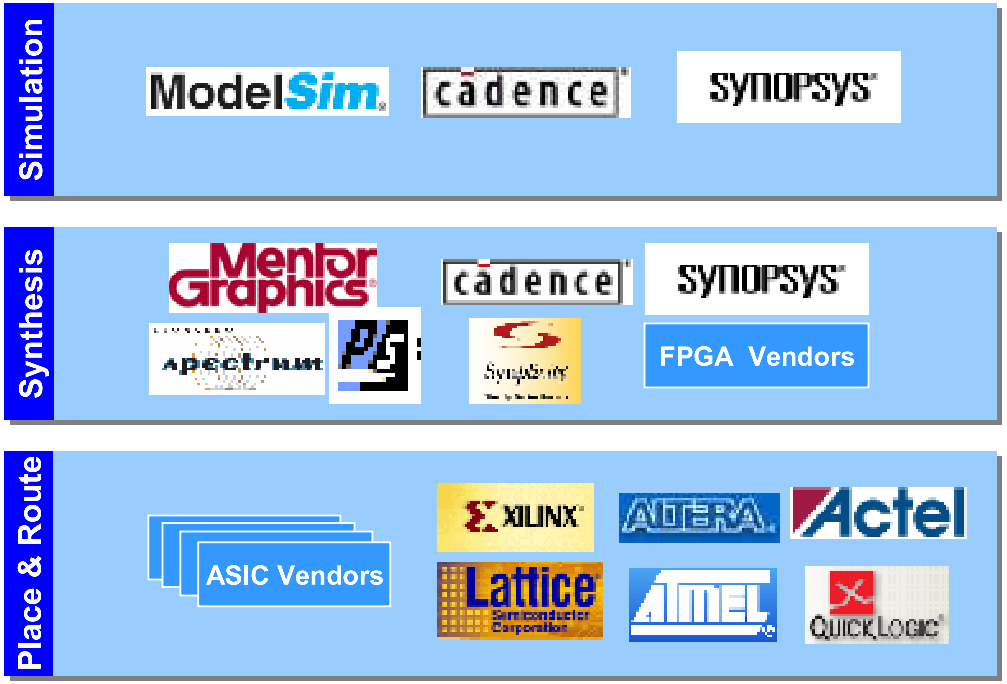
- Need to keep track of changes
  - Back track and/or modify existing design
- Simple interface
  - Easy to use and low impact
- 3rd-party version management:
  - GNU RCS or CVS
  - Microsoft Visual Source Safe™ (VSS)
  - Rational (IBM) ClearCase™
  - Synchronicity DesignSync™
  - ClioSoft SOS™



Design Hierarchy	Design Unit Name	Li...
uart_tb	uart_tb	UAR
uart_tb	uart_tb	UAR
struct	uart_tb	UAR
U_0	tester	UAR
U_1	uart_top	UAR
U_1	cpu_interface	UAR
U_0	control_operation	UAR
U_2	clock_divider	UAR
U_3	address_decode	UAR
U_4	serial_interface	UAR
U_0	xmit_rcv_control	UAR
U_1	status_registers	UAR

```
Writing HDLFileList to d:/tools/hds2005.2_beta/examples/uart/hds/uart_tb/struct.bd.info/HDLFileList.txt
UART D:/Tools/HDS2005.2_Beta/examples/uart/hdl/tester/hdl/tester_flow.vhd
UART D:/Tools/HDS2005.2_Beta/examples/uart/hdl/control_operation/hdl/control_operation_fsm.vhd
UART D:/Tools/HDS2005.2_Beta/examples/uart/hdl/cpu_interface/hdl/cpu_interface_intconx.vhd
UART D:/Tools/HDS2005.2_Beta/examples/uart/hdl/clock_divider/hdl/clock_divider_flow.vhd
UART D:/Tools/HDS2005.2_Beta/examples/uart/hdl/address_decode/hdl/address_decode_tbl.vhd
UART D:/Tools/HDS2005.2_Beta/examples/uart/hdl/xmit_rcv_control/hdl/xmit_rcv_control_fsm.vhd
UART D:/Tools/HDS2005.2_Beta/examples/uart/hdl/status_registers_spec.vhd
UART D:/Tools/HDS2005.2_Beta/examples/uart/hdl/serial_interface/hdl/serial_interface_struct.vhd
UART D:/Tools/HDS2005.2_Beta/examples/uart/hdl/uart_top/hdl/uart_top_struct.vhd
UART D:/Tools/HDS2005.2_Beta/examples/uart/hdl/uart_tb/hdl/uart_tb_struct.vhd
Written HDLFileList to d:/tools/hds2005.2_beta/examples/uart/hds/uart_tb/struct.bd.info/HDLFileList.txt
```

Downstream Tool/Flow Integration


Simulation

Synthesis

FPGA Vendors

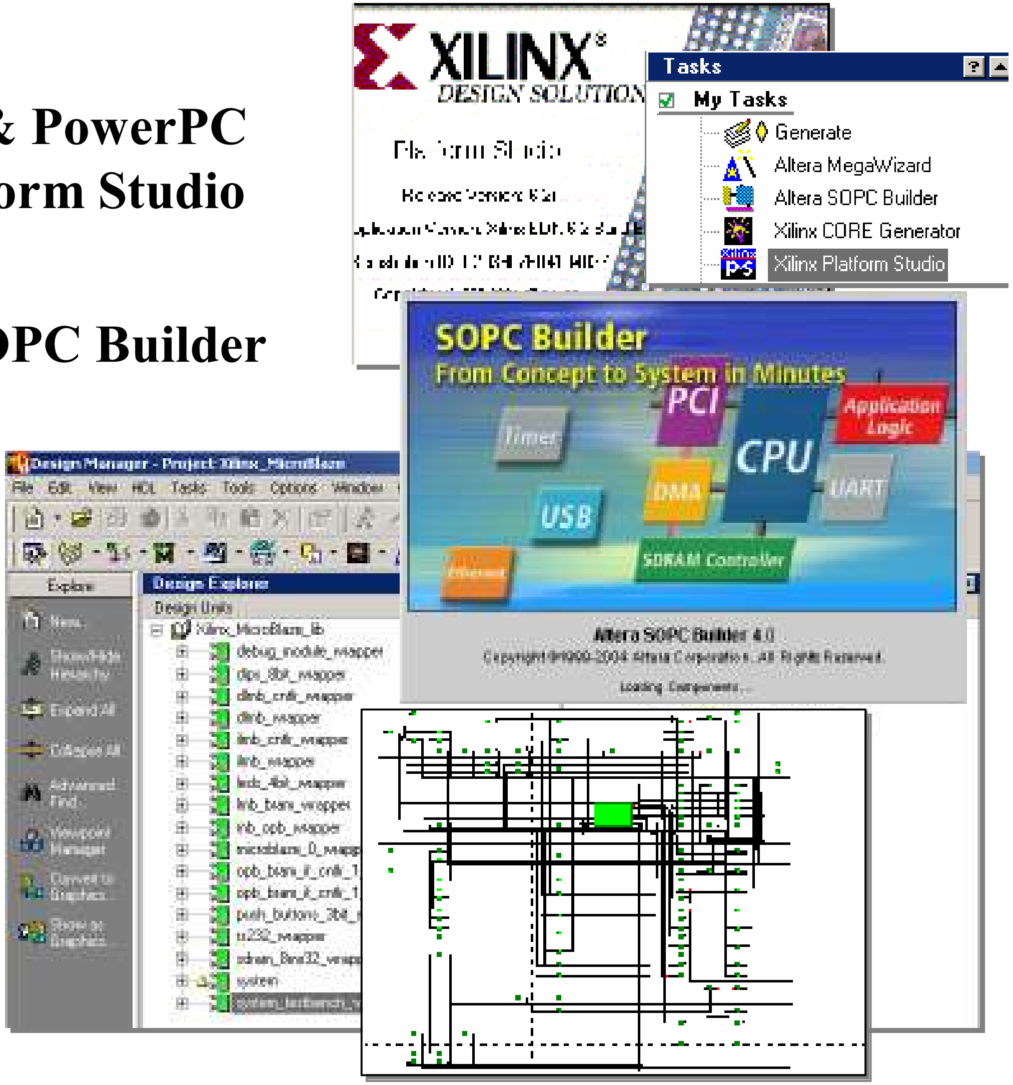
Place & Route

ASIC Vendors

Focus on system rather than process issues!

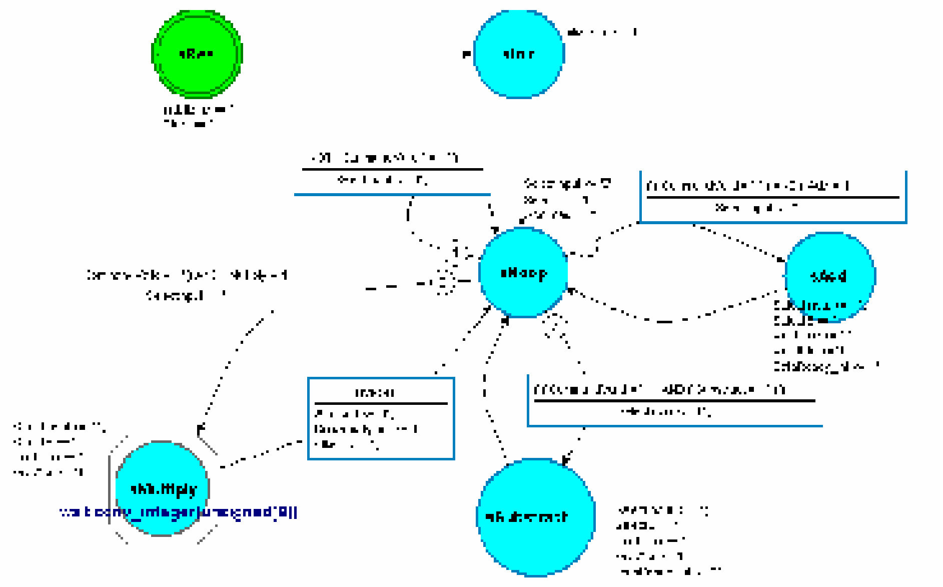
Xilinx & Altera CPU Integration

- Supports Xilinx MicroBlaze & PowerPC CPU's with Xilinx EDK Platform Studio
- Supports Altera CPU with SOPC Builder
- ModelSim simulation view in HDL Designer browser
- Synthesis view for Precision Synthesis & LeonardoSpectrum in Side Data

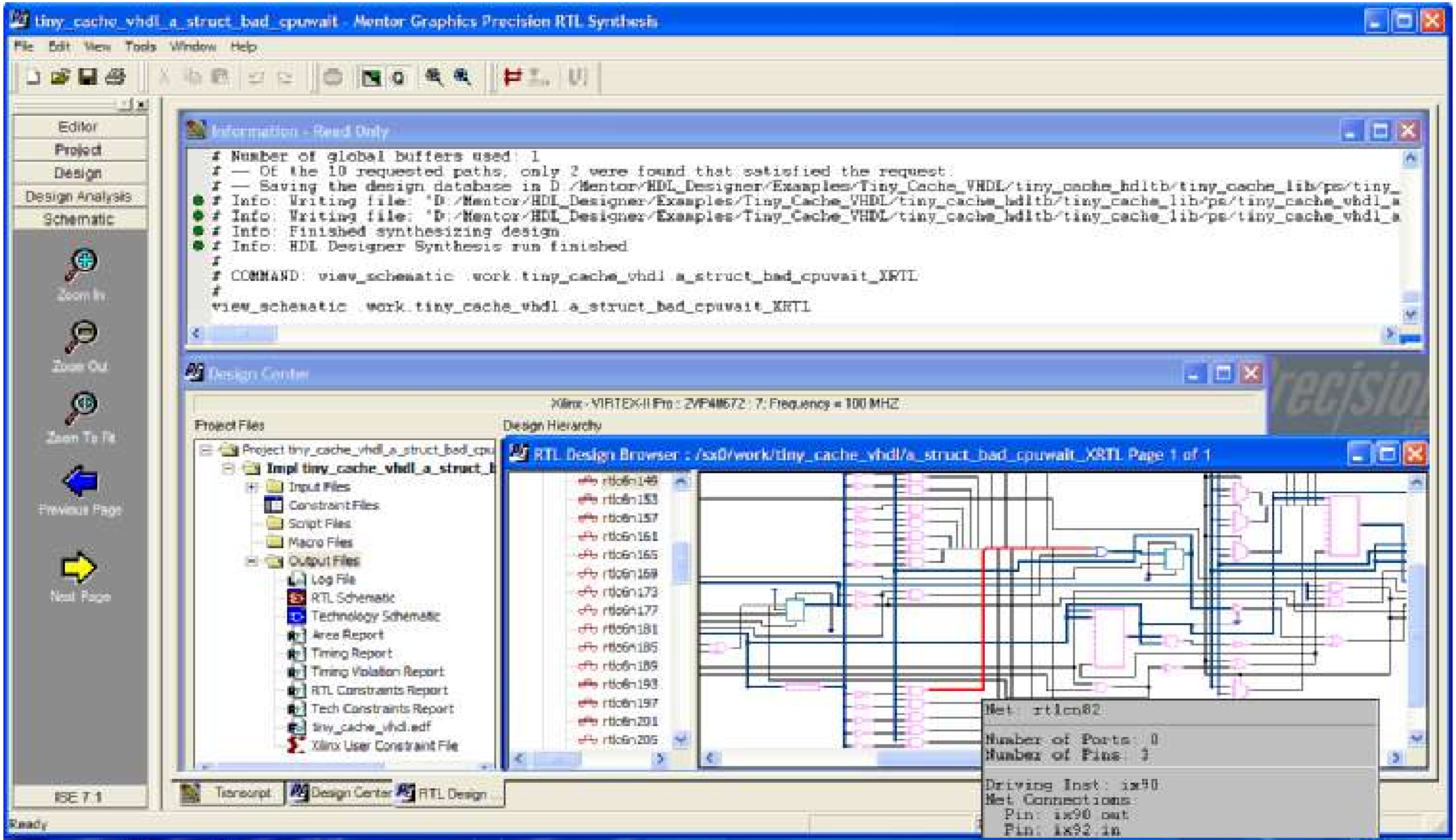


Automated Design Documentation

- Make design communications easier for all
  - — Design Comprehension
  - — Design review
  - — Multi-site design sharing
  - — Lifecycle maintenance
- HDL Designer helps
  - — Renders graphics
    - Block diagrams
    - State machine charts
  - — Dynamic HTML Export



HDL Designer provides a complete Design, Data & Process Management Environment



tiny_cache_vhdl_a_struct_bad_cpuwait - Mentor Graphics Precision RTL Synthesis

Information - Read Only

```
# Number of global buffers used: 1
# -- Of the 10 requested paths, only 2 were found that satisfied the request.
# -- Saving the design database in D:/Mentor/HDL_Designer/Examples/Tiny_Cache_VHDL/tiny_cache_hdltb/tiny_cache_lib/ps/tiny_
# Info: Writing file: 'D:/Mentor/HDL_Designer/Examples/Tiny_Cache_VHDL/tiny_cache_hdltb/tiny_cache_lib/ps/tiny_cache_vhdl_a
# Info: Writing file: 'D:/Mentor/HDL_Designer/Examples/Tiny_Cache_VHDL/tiny_cache_hdltb/tiny_cache_lib/ps/tiny_cache_vhdl_a
# Info: Finished synthesizing design.
# Info: HDL Designer Synthesis run finished
#
# COMMAND: view_schematic .work.tiny_cache_vhdl.a_struct_bad_cpuwait_XRTL
#
view_schematic .work.tiny_cache_vhdl.a_struct_bad_cpuwait_XRTL
```

Design Center

Xilinx - VIRTEX-II Pro : 2VP4ff672 : 7: Frequency = 100 MHZ

RTL Design Browser : /sx0/work/tiny_cache_vhdl/a_struct_bad_cpuwait_XRTL Page 1 of 1

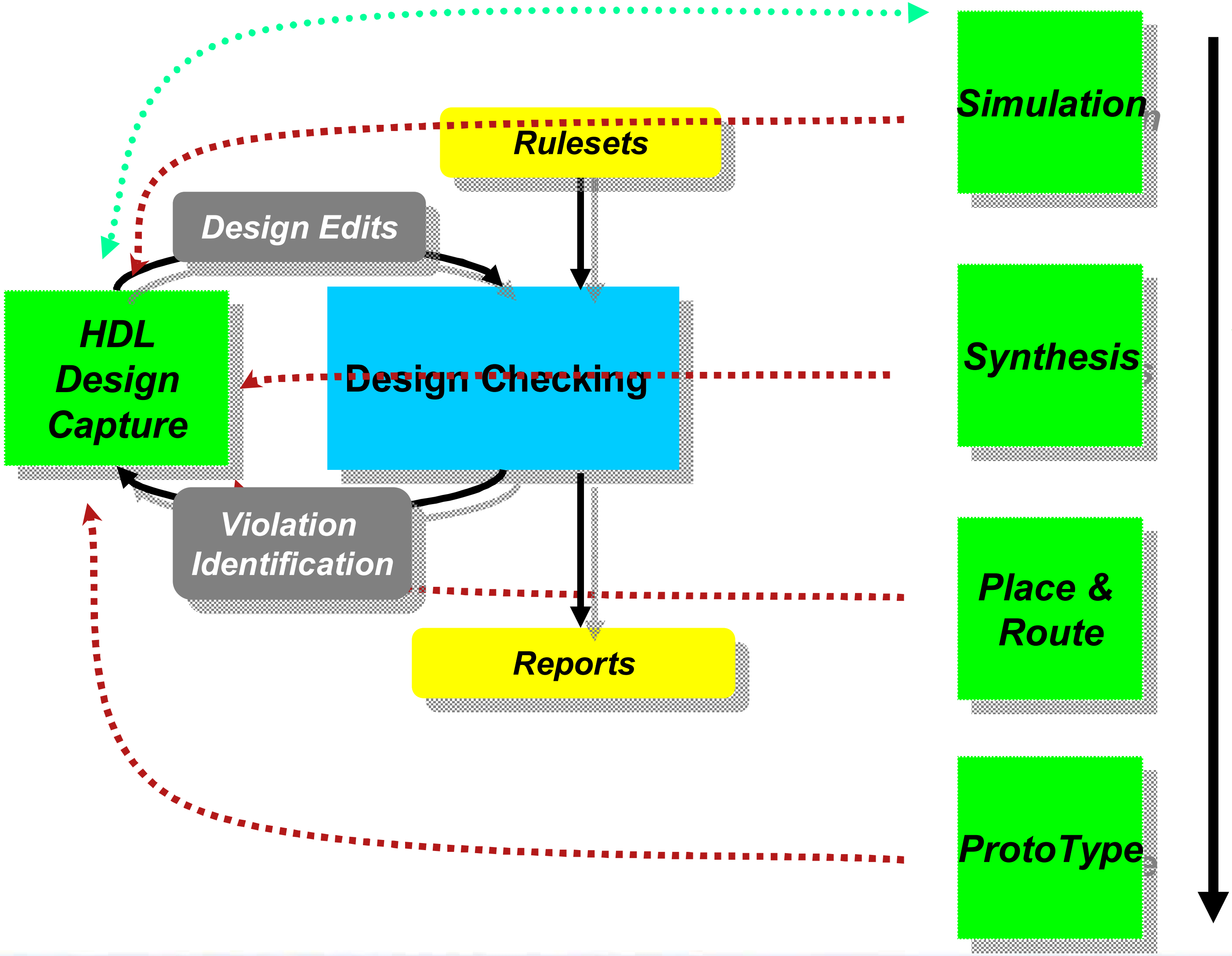
```
Net: rtlcn82

Number of Ports: 0
Number of Pins: 3

Driving Inst: ix90
Net Connections:
   Pin: ix90 out
   Pin: ix92 in
```

NEW

HDL Designer - DesignChecker

- Check for issues as the code is written!
  - Identifies coding/design bugs earlier
  - Checking at the end of a project is too late!
- Result Better Quality code at the time of creation

Where Does Static Design Checker Fit?



Finding problems later increases the number of iterations ☹

Early checking reduces the number of iterations ☺

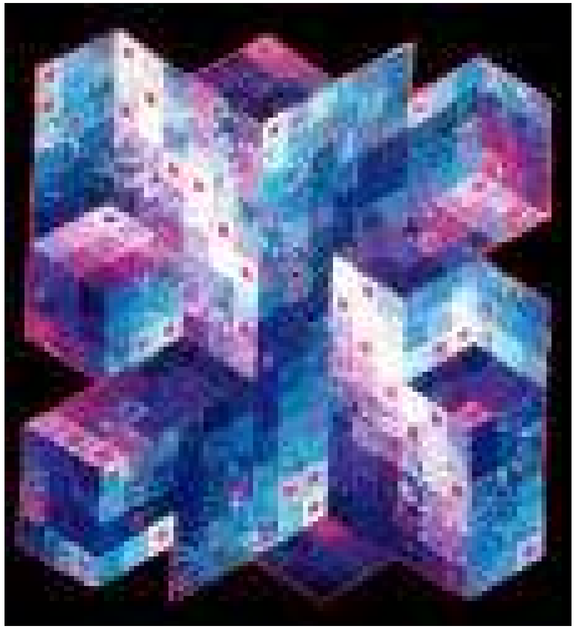
Detect with Static Design Checking

Checks that:

- **Validates Code Standalone or Hierarchically**
- **NO translation to a gate-level netlist**
- **NO synthesis, simulation, DFT, or formal engines**

What Types of Checks can be Performed?

- Allow
- Assignments
- Clocks & Resets
- Conditions
- Configurations
- Declarations
- Directives
- Format
- FSM
- Gates
- HDL Syntax & Semantics



- Instances
- Labels
- Naming
- Order
- Partitioning
- Pragmas
- Race Conditions
- Ranges
- Registers
- Sensitivity
- Subprograms
- VITAL

Built-In Rulesets

Supports out-of-the-box use

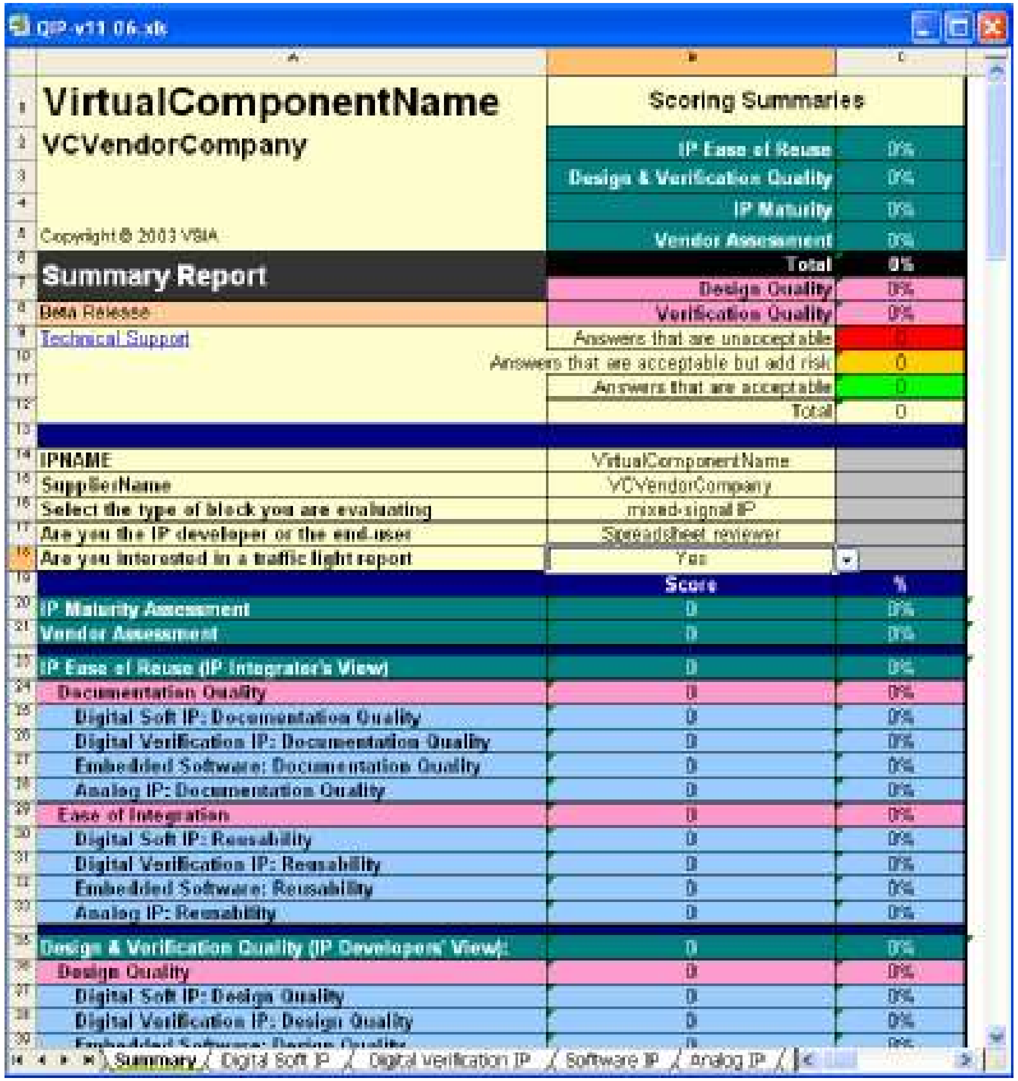
Built using base rules

- Reuse Methodology Manual- 3rd Edition
- Essentials – default, core checks
- Altera
- Xilinx
- Quality IP scoring matrix VSI Alliance (soon)

QIP Support

- Quality IP scoring matrix from VSI Alliance
- Manually fill out the dynamic spreadsheet & get a score
- If the score is low, add significant time to your $\text{Time}_{\text{Reuse}}$ variable

www.vsi.org



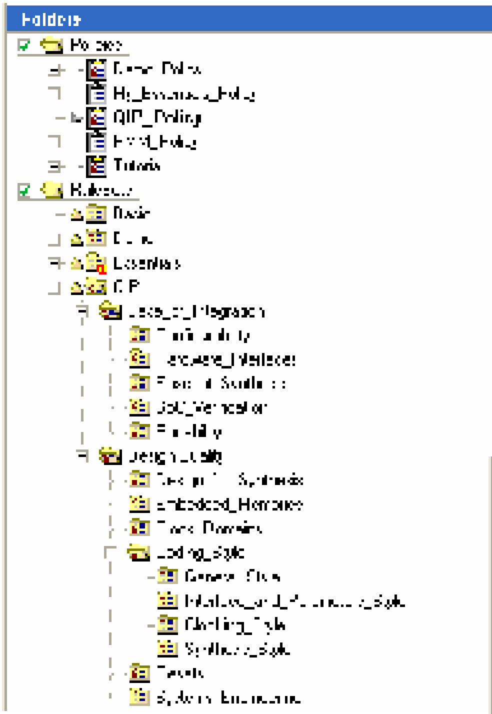
	A	B	C
1	VirtualComponentName	Scoring Summaries	
2	VCVendorCompany	IP Ease of Reuse	0%
3		Design & Verification Quality	0%
4		IP Maturity	0%
5	Copyright © 2003 VSIA	Vendor Assessment	0%
6	Summary Report	Total	0%
7		Design Quality	0%
8	Beta Release	Verification Quality	0%
9	Technical Support	Answers that are unacceptable	0
10		Answers that are acceptable but add risk	0
11		Answers that are acceptable	0
12		Total	0
13			
14	IPNAME	VirtualComponentName	
15	SupplierName	VCVendorCompany	
16	Select the type of block you are evaluating	mixed-signal IP	
17	Are you the IP developer or the end-user	Spreadsheet reviewer	
18	Are you interested in a traffic light report	Yes	
19		Score	%
20	IP Maturity Assessment	0	0%
21	Vendor Assessment	0	0%
23	IP Ease of Reuse (IP Integrator's View)	0	0%
24	Documentation Quality	0	0%
25	Digital Soft IP: Documentation Quality	0	0%
26	Digital Verification IP: Documentation Quality	0	0%
27	Embedded Software: Documentation Quality	0	0%
28	Analog IP: Documentation Quality	0	0%
29	Ease of Integration	0	0%
30	Digital Soft IP: Reusability	0	0%
31	Digital Verification IP: Reusability	0	0%
32	Embedded Software: Reusability	0	0%
33	Analog IP: Reusability	0	0%
35	Design & Verification Quality (IP Developers' View):	0	0%
36	Design Quality	0	0%
37	Digital Soft IP: Design Quality	0	0%
38	Digital Verification IP: Design Quality	0	0%
39	Embedded Software: Design Quality	0	0%

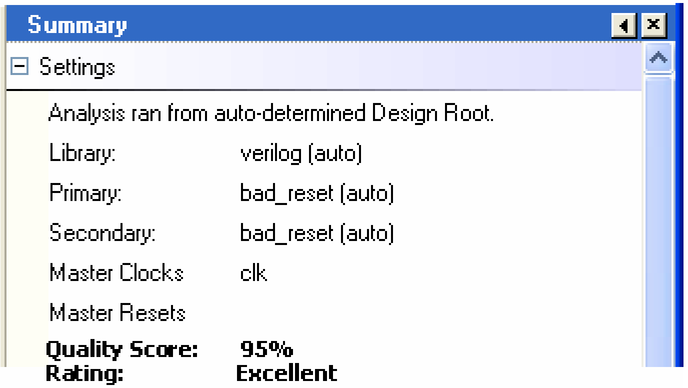
QIP Scoring

- Score X weight for each rule
- Rule pass is counted, rule fail is uncounted
- Percentage of total is the final score

$$\text{QIP Score} = \frac{\text{Design score total}}{\text{Total available score}} \text{ X } 100$$

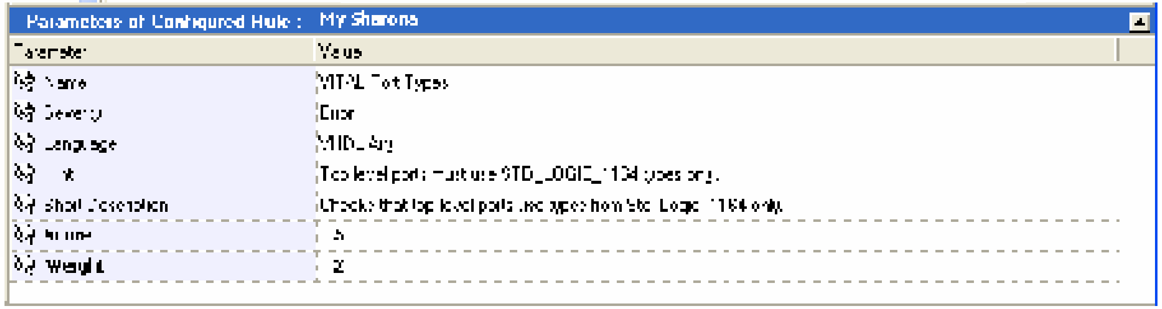
QIP – How it Works





Analysis ran from auto-determined Design Root.

Library:	verilog (auto)
Primary:	bad_reset (auto)
Secondary:	bad_reset (auto)
Master Clocks	clk
Master Resets	
Quality Score:	**95%**
Rating:	**Excellent**


Parameters of Configured Rule : My Sharons

Parameter	Value
Name	VITAL Port Types
Severity	Error
Language	VHDL_Any
Text	Top level ports must use STD_LOGIC_1164 types only.
Short Description	Checks that top level ports use types from Std_Logic_1164 only.
Score	5
Weight	2

- Built-in QIP ruleset & policy
- Score & weight parameter for any rule
- Score & rating text in Summary
- Export report
- Considering bi-directional Excel feature

Top 5 Design Mistakes

1. **Clock domain crossing**
2. **Unsafe state machines**
3. **Combinational feedback**
4. **Unused or multiply-driven signals**
5. **Incorrect sensitivity lists**

Incorrect Sensitivity Lists

Typical Causes:

- Edited code, but forgot sensitivity list
- Extra signals in the sensitivity list, just to be safe

Main Effects:

- Can infer unintended latches
- Incorrect functional behavior
- RTL & Netlist simulation mismatches
- Extra signals slow simulation

Unused or Multiply-Driven Signals

Typical Causes:

- Miss-spelling or connected wrong signal in an assignment
- Extra component port defined that is not needed
- Forgot to connect a port

Main Effect:

- Unknown (X) values in simulation
- Place & route errors
- Unintended behavior

Combinational Feedback

Typical Causes:

- No assignment inside a process sensitive to a clock
- Assignment is incorrect thus forming an unintended loop

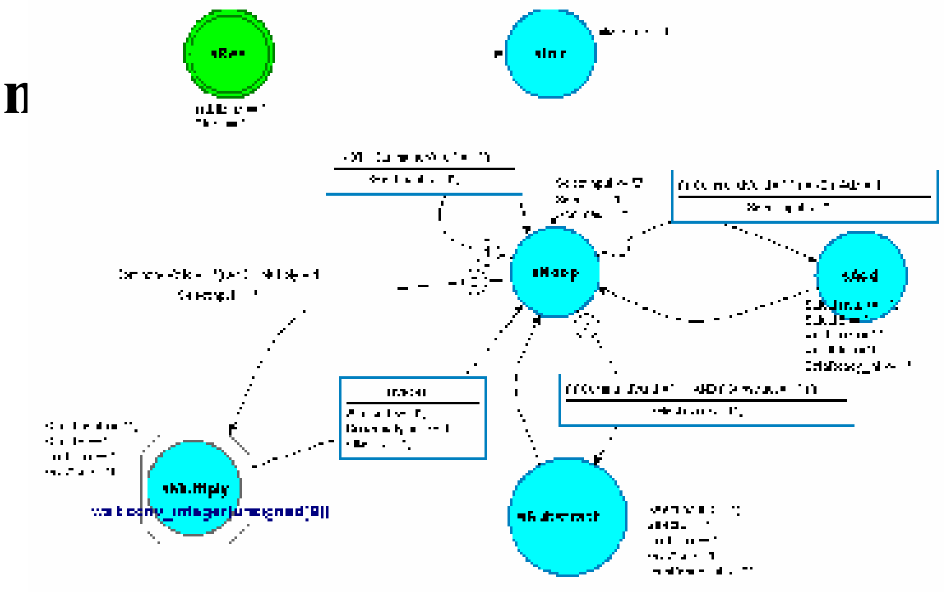
Main Effect:

- Static timing analysis difficult or impossible
- Probably causes a functional bug

Unsafe State Machines

Typical Causes:

- Incomplete specification
- No default case
- No recovery state
- Unreachable state



Main Effect:

- Unrecoverable state

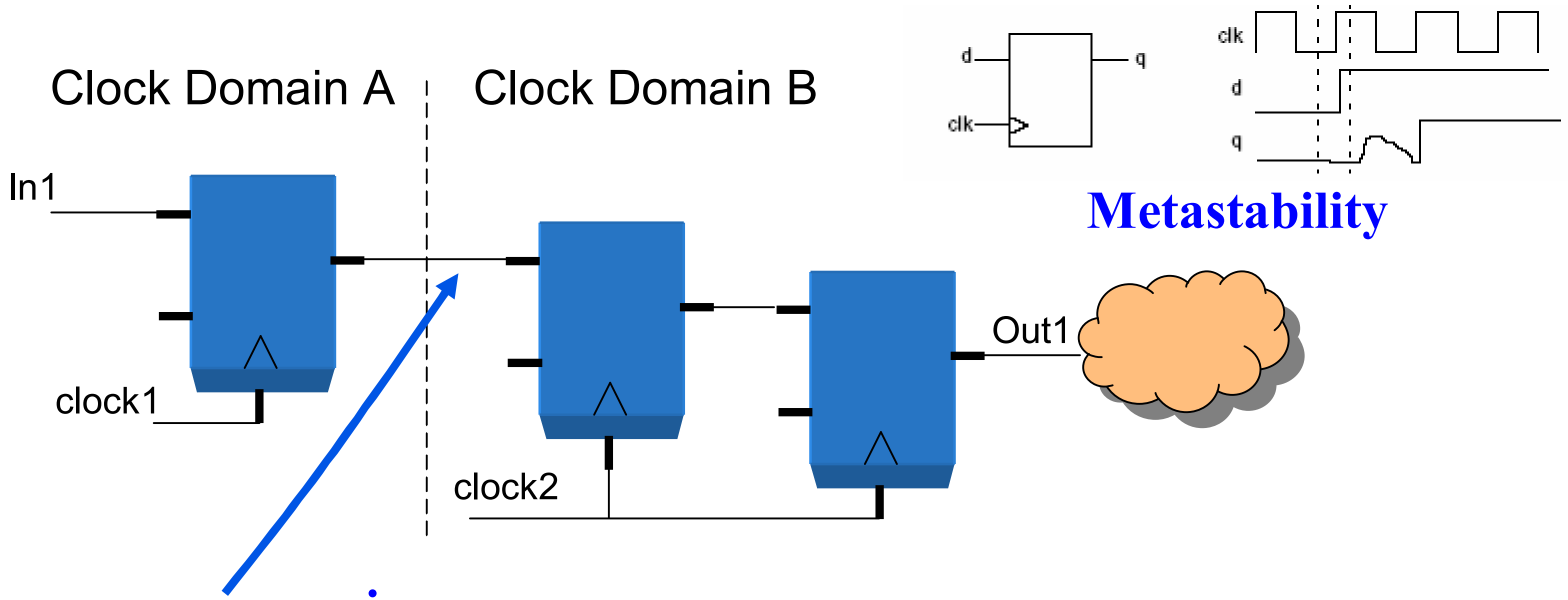
Clock Domain Crossing

Typical Issues:

- Metastability neither a 1 or 0 is at the input & the wrong value is placed on the output
- Not typically detected by simulation

Main Effect:

Design creates incorrect logic at some point in time



Metastability

Add one or more syncronizers

Demo

Design Import/Reference/SendTo

Hierarchy

Cross Reference, Where Used, Search, Unbound

Render Graphics – Documentation

VHDL Configurations – Generation Application

Version Management

Design Checking – As you write!

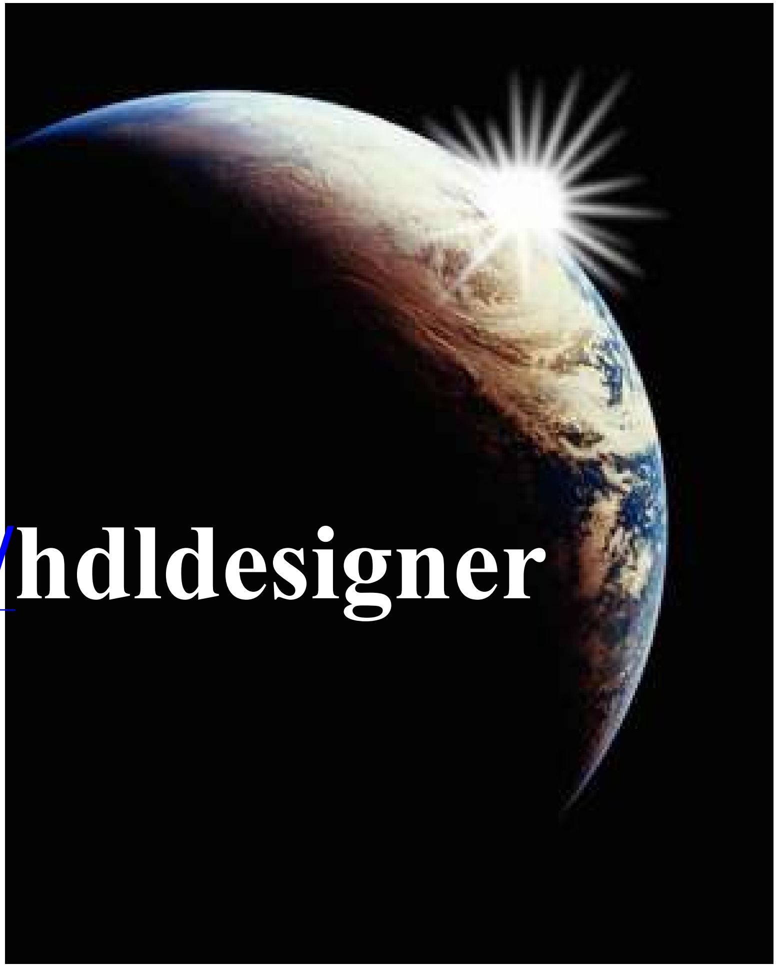
Properties/SideData

Flow customerisation – API, Simulation, Synthesis

Advanced Verification Techniques

Learning More

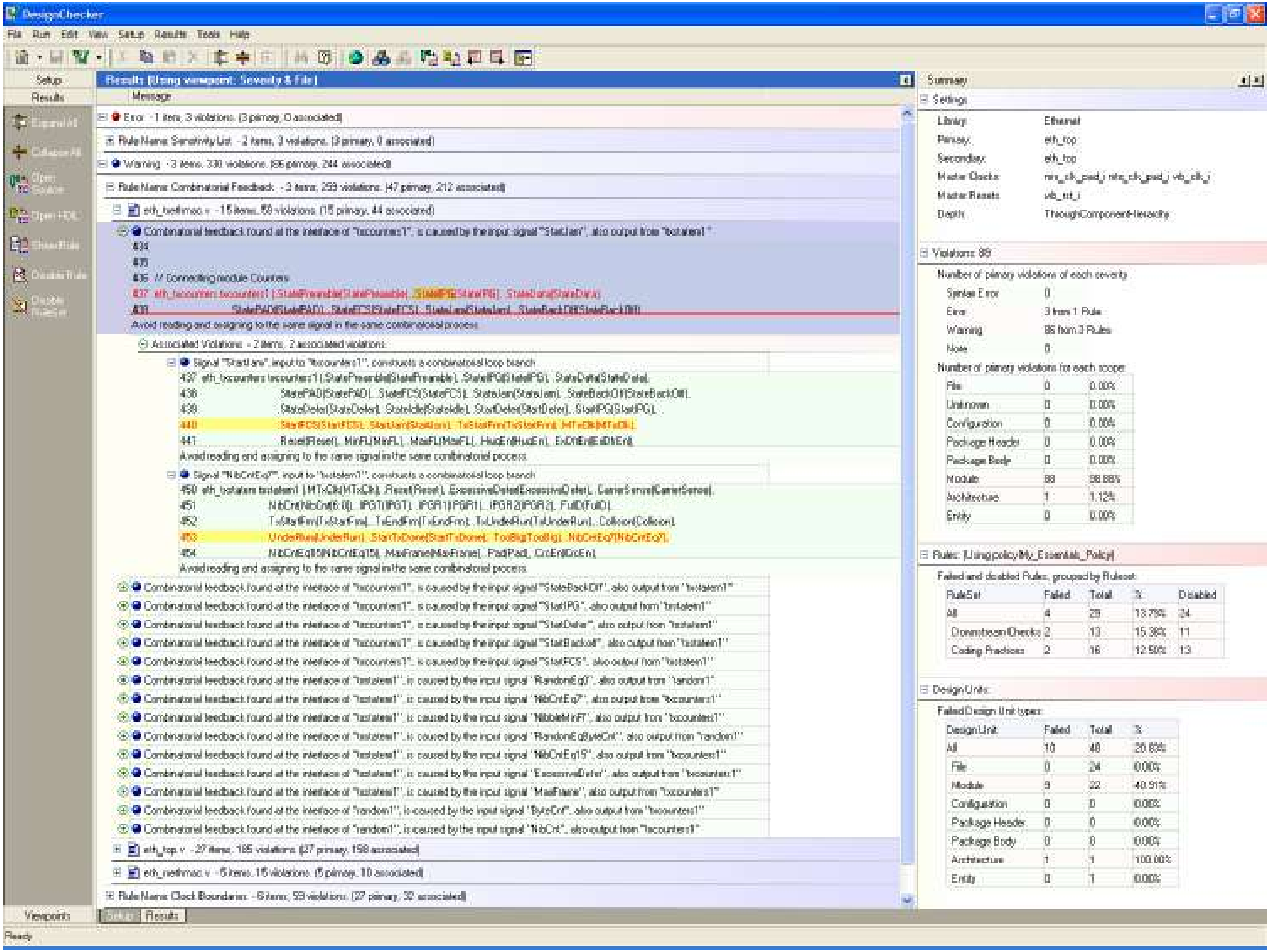
HDL Designer & DesignAnalyst information is available at:



www.mentor.com/hdldesigner

Mentor Graphics®

www.mentor.com



Results (Using viewpoint: Severity & File)

```
Message
⊟ Error  - 1 item, 3 violations. (3 primary, 0 associated)
  ⊞ Rule Name: Sensitivity List  - 2 items, 3 violations. (3 primary, 0 associated)
⊟ Warning  - 3 items, 330 violations. (86 primary, 244 associated)
  ⊟ Rule Name: Combinatorial Feedback  - 3 items, 259 violations. (47 primary, 212 associated)
    ⊟ eth_txethmac.v  - 15 items, 59 violations. (15 primary, 44 associated)
      ⊖ Combinatorial feedback found at the interface of "txcounters1", is caused by the input signal "StartJam", also output from "txstatem1".
        434
        435
        436  // Connecting module Counters
        437  eth_txcounters txcounters1 (.StatePreamble(StatePreamble), .StateIPG(StateIPG), .StateData(StateData),
        438                     .StatePAD(StatePAD), .StateFCS(StateFCS), .StateJam(StateJam), .StateBackOff(StateBackOff),
        Avoid reading and assigning to the same signal in the same combinatorial process.
        ⊖ Associated Violations  - 2 items, 2 associated violations:
          ⊟ Signal "StartJam", input to "txcounters1", constructs a combinatorial loop branch
            437  eth_txcounters txcounters1 (.StatePreamble(StatePreamble), .StateIPG(StateIPG), .StateData(StateData),
            438                     .StatePAD(StatePAD), .StateFCS(StateFCS), .StateJam(StateJam), .StateBackOff(StateBackOff),
            439                     .StateDefer(StateDefer), .StateIdle(StateIdle), .StartDefer(StartDefer), .StartIPG(StartIPG),
            440                     .StartFCS(StartFCS), .StartJam(StartJam), .TxStartFrm(TxStartFrm), .MTxClk(MTxClk),
            441                     .Reset(Reset), .MinFL(MinFL), .MaxFL(MaxFL), .HugEn(HugEn), .ExDfrEn(ExDfrEn),
            Avoid reading and assigning to the same signal in the same combinatorial process.
          ⊟ Signal "NibCntEq7", input to "txstatem1", constructs a combinatorial loop branch
            450  eth_txstatem txstatem1 (.MTxClk(MTxClk), .Reset(Reset), .ExcessiveDefer(ExcessiveDefer), .CarrierSense(CarrierSense),
            451                    .NibCnt(NibCnt[6:0]), .IPGT(IPGT), .IPGR1(IPGR1), .IPGR2(IPGR2), .FullD(FullD),
            452                    .TxStartFrm(TxStartFrm), .TxEndFrm(TxEndFrm), .TxUnderRun(TxUnderRun), .Collision(Collision),
            453                    .UnderRun(UnderRun), .StartTxDone(StartTxDone), .TooBig(TooBig), .NibCntEq7(NibCntEq7),
            454                    .NibCntEq15(NibCntEq15), .MaxFrame(MaxFrame), .Pad(Pad), .CrcEn(CrcEn),
            Avoid reading and assigning to the same signal in the same combinatorial process.
      ⊕ Combinatorial feedback found at the interface of "txcounters1", is caused by the input signal "StateBackOff", also output from "txstatem1"
      ⊕ Combinatorial feedback found at the interface of "txcounters1", is caused by the input signal "StartIPG", also output from "txstatem1"
      ⊕ Combinatorial feedback found at the interface of "txcounters1", is caused by the input signal "StartDefer", also output from "txstatem1"
      ⊕ Combinatorial feedback found at the interface of "txcounters1", is caused by the input signal "StartBackoff", also output from "txstatem1"
      ⊕ Combinatorial feedback found at the interface of "txcounters1", is caused by the input signal "StartFCS", also output from "txstatem1"
      ⊕ Combinatorial feedback found at the interface of "txstatem1", is caused by the input signal "RandomEq0", also output from "random1"
      ⊕ Combinatorial feedback found at the interface of "txstatem1", is caused by the input signal "NibCntEq7", also output from "txcounters1"
      ⊕ Combinatorial feedback found at the interface of "txstatem1", is caused by the input signal "NibbleMinFl", also output from "txcounters1"
      ⊕ Combinatorial feedback found at the interface of "txstatem1", is caused by the input signal "RandomEqByteCnt", also output from "random1"
      ⊕ Combinatorial feedback found at the interface of "txstatem1", is caused by the input signal "NibCntEq15", also output from "txcounters1"
      ⊕ Combinatorial feedback found at the interface of "txstatem1", is caused by the input signal "ExcessiveDefer", also output from "txcounters1"
      ⊕ Combinatorial feedback found at the interface of "txstatem1", is caused by the input signal "MaxFrame", also output from "txcounters1"
      ⊕ Combinatorial feedback found at the interface of "random1", is caused by the input signal "ByteCnt", also output from "txcounters1"
      ⊕ Combinatorial feedback found at the interface of "random1", is caused by the input signal "NibCnt", also output from "txcounters1"
    ⊞ eth_top.v  - 27 items, 185 violations. (27 primary, 158 associated)
    ⊞ eth_rxethmac.v  - 5 items, 15 violations. (5 primary, 10 associated)
  ⊞ Rule Name: Clock Boundaries  - 6 items, 59 violations. (27 primary, 32 associated)
```

Summary

⊟ Settings

Library:	Ethernet
Primary:	eth_top
Secondary:	eth_top
Master Clocks:	mtx_clk_pad_i mrx_clk_pad_i wb_clk_i
Master Resets:	wb_rst_i
Depth:	ThroughComponentHierarchy

⊟ Violations: 89

Number of primary violations of each severity

Syntax Error	0
Error	3 from 1 Rule
Warning	86 from 3 Rules
Note	0

Number of primary violations for each scope

File	0	0.00%
Unknown	0	0.00%
Configuration	0	0.00%
Package Header	0	0.00%
Package Body	0	0.00%
Module	88	98.88%
Architecture	1	1.12%
Entity	0	0.00%

⊟ Rules: (Using policy My_Essentials_Policy)

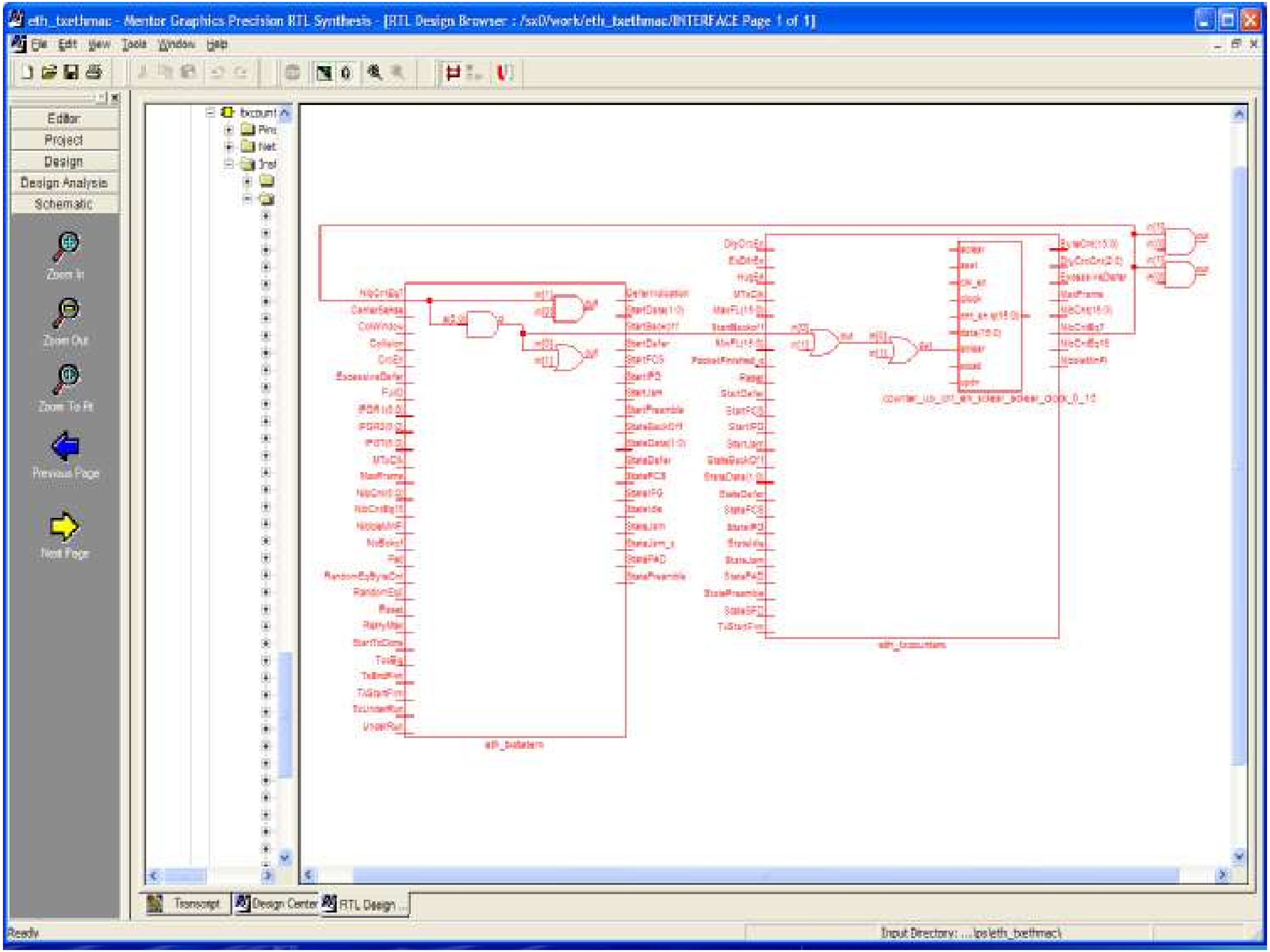
Failed and disabled Rules, grouped by Ruleset:

RuleSet	Failed	Total	%	Disabled
All	4	29	13.79%	24
DownstreamChecks	2	13	15.38%	11
Coding Practices	2	16	12.50%	13

⊟ Design Units:

Failed Design Unit types:

Design Unit	Failed	Total	%
All	10	48	20.83%
File	0	24	0.00%
Module	9	22	40.91%
Configuration	0	0	0.00%
Package Header	0	0	0.00%
Package Body	0	0	0.00%
Architecture	1	1	100.00%
Entity	0	1	0.00%


eth_txethmac - Mentor Graphics Precision RTL Synthesis - [RTL Design Browser : /sx0/work/eth_txethmac/INTERFACE Page 1 of 1]
File
Edit
View
Tools
Window
Help
Editor
Project
Design
Design Analysis
Schematic
Zoom In
Zoom Out
Zoom To Fit
Previous Page
Next Page
txcount
Pins
Net
Inst
NibCntEq7
CarrierSense
ColWindow
Collision
CrcEn
ExcessiveDefer
FullD
IPGR1(6:0)
IPGR2(6:0)
IPGT(6:0)
MTxClk
MaxFrame
NibCnt(15:0)
NibCntEq15
NibbleMinFl
NoBckof
Pad
RandomEqByteCnt
RandomEq0
Reset
RetryMax
StartTxDone
TxStartFrm
TxEndFrm
TxUnderRun
UnderRun
in(1)
in(2)
out
in(0)
DeferIndication
StartData(1:0)
StartBackoff
StartDefer
StartFCS
StartIPG
StartJam
StartPreamble
StateBackOff
StateData(1:0)
StateDefer
StateFCS
StateIPG
StateIdle
StateJam
StateJam_q
StatePAD
StatePreamble
eth_txstatem
DlyCrcEn
ExDfrEn
HugEn
MTxClk
MaxFL(15:0)
StateBackOff
MinFL(15:0)
PacketFinished_q
Reset
StartDefer
StartFCS
StartIPG
StartJam
StateBackOff
StateData(1:0)
StateDefer
StateFCS
StateIPG
StateIdle
StateJam
StatePAD
StatePreamble
StateSFD
TxStartFrm
aclear
aset
cnt_en
clock
cnt_en_sig(15:0)
data(15:0)
sclear
sload
updn
counter_up_cnt_en_aclear_sclear_clock_0_15
ByteCnt(15:0)
DlyCrcCnt(2:0)
ExcessiveDefer
MaxFrame
NibCnt(15:0)
NibCntEq7
NibCntEq15
NibbleMinFl
eth_txcounters
Transcript
Design Center
RTL Design ...
Ready
Input Directory: ...\ps\eth_txethmac\