SOPC WORLD 2004

Accelerating System-Level Design and Multi-Processor Systems in FPGA

ALTERA®

Agenda

Accelerating System-Level Design

- System Integration Issues
- SOPC Builder
- Interfacing to External Processors
- Summary

Multi-Processor Systems in FPGA

- Why Use Multiple Processors?
- Multi-Processor Architectures
- Design Considerations
- Software Development

System Integration Issues

- Matching I/O Specifications
  - Voltage Standards
  - Timing Requirements
- Connecting the Block Interfaces
  - Control Signal Types
  - Control Signal Assertion Levels
  - Data Widths
  - Transaction Timing
- Domain Crossing
  - Interface Domain
  - Clock Domain
- Time
  - Develop
  - Change Orders
  - Verification

SOPC WORLD 2004

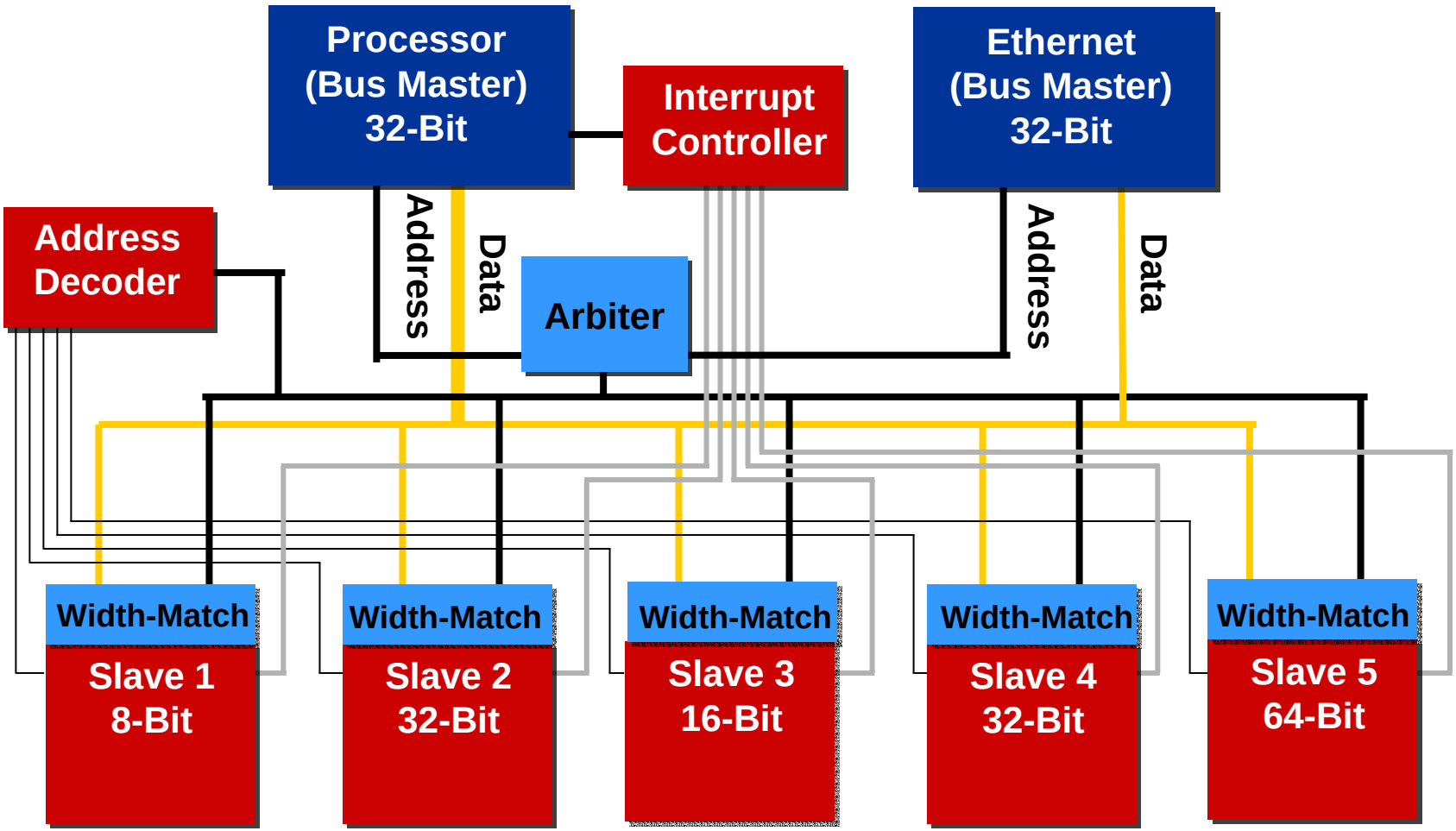
ALTERA

SOPC WORLD 2004

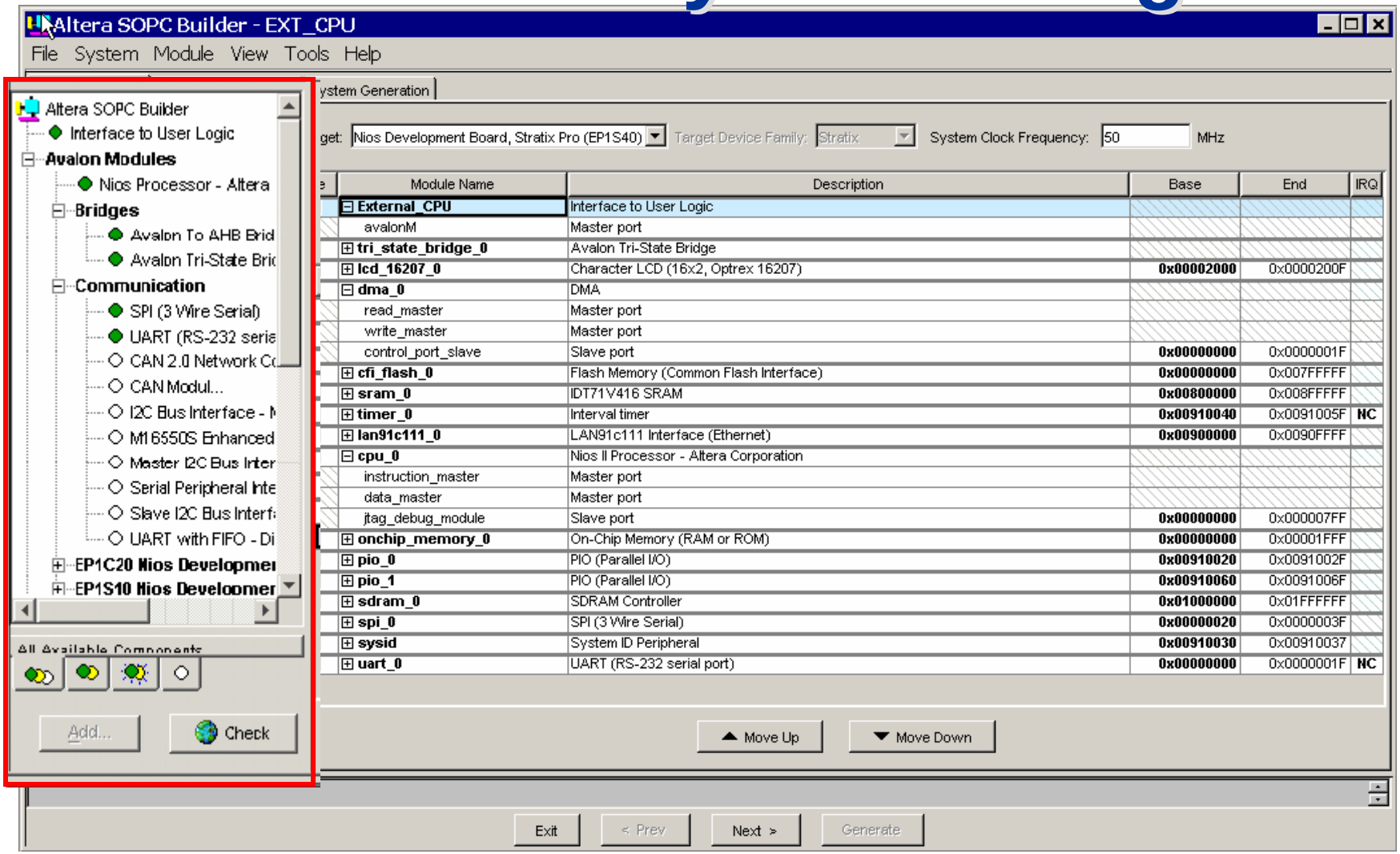
SOPC Builder

ALTERA®

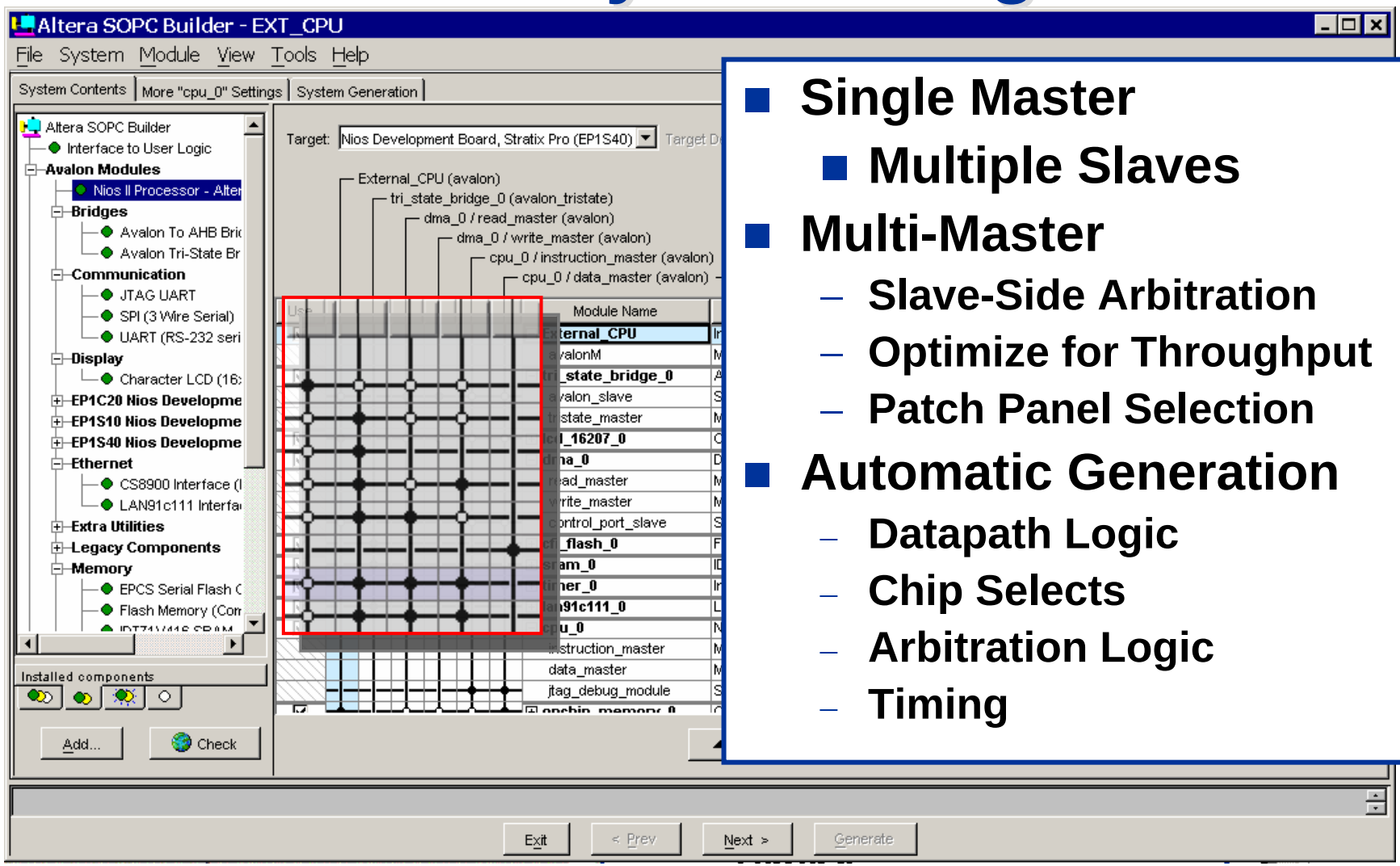
Embedded System Integration



SOPC Builder: System Design

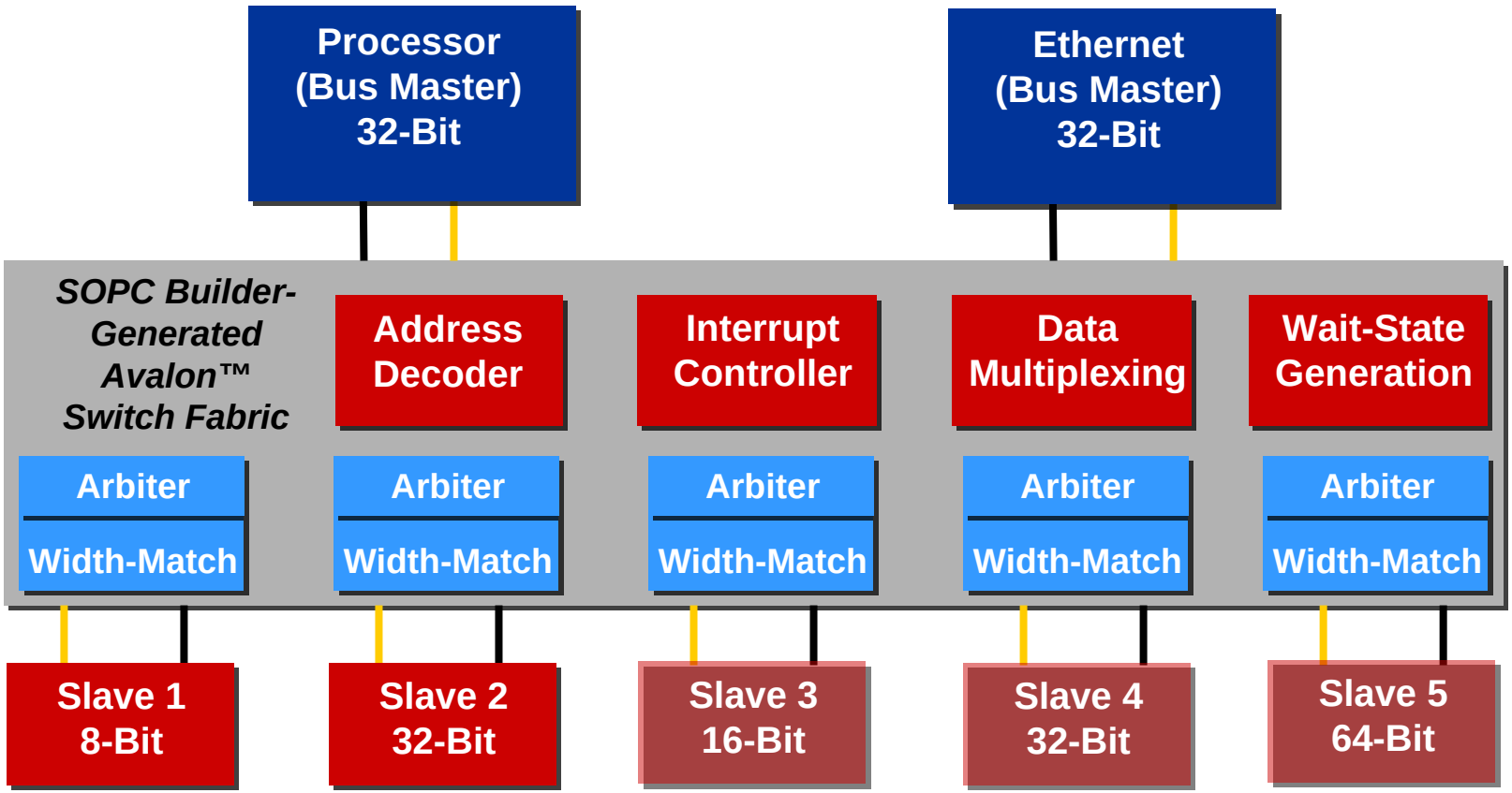


SOPC Builder: System Integration



- **Single Master**
  - **Multiple Slaves**
- **Multi-Master**
  - **Slave-Side Arbitration**
  - **Optimize for Throughput**
  - **Patch Panel Selection**
- **Automatic Generation**
  - **Datapath Logic**
  - **Chip Selects**
  - **Arbitration Logic**
  - **Timing**

SOPC Builder: Integration



Avalon Interface

- An Open Standard
  - Simple, Synchronous Interface
  - Switch Fabric Interconnect
    - Controls Signal Timing
    - Matching Logic Levels
- Many Signal Types Supported
  - Peripheral Uses Only the Signals it Needs
    - Any Combination of Signals Possible
  - Support for Arbitrary Setup Time, Hold Time & Wait States
- Switch Fabric Generated in SOPC Builder
  - Eliminates Tedious HDL Writing
  - Increases Intellectual Property (IP) Re-Use
  - Decreases Development Time

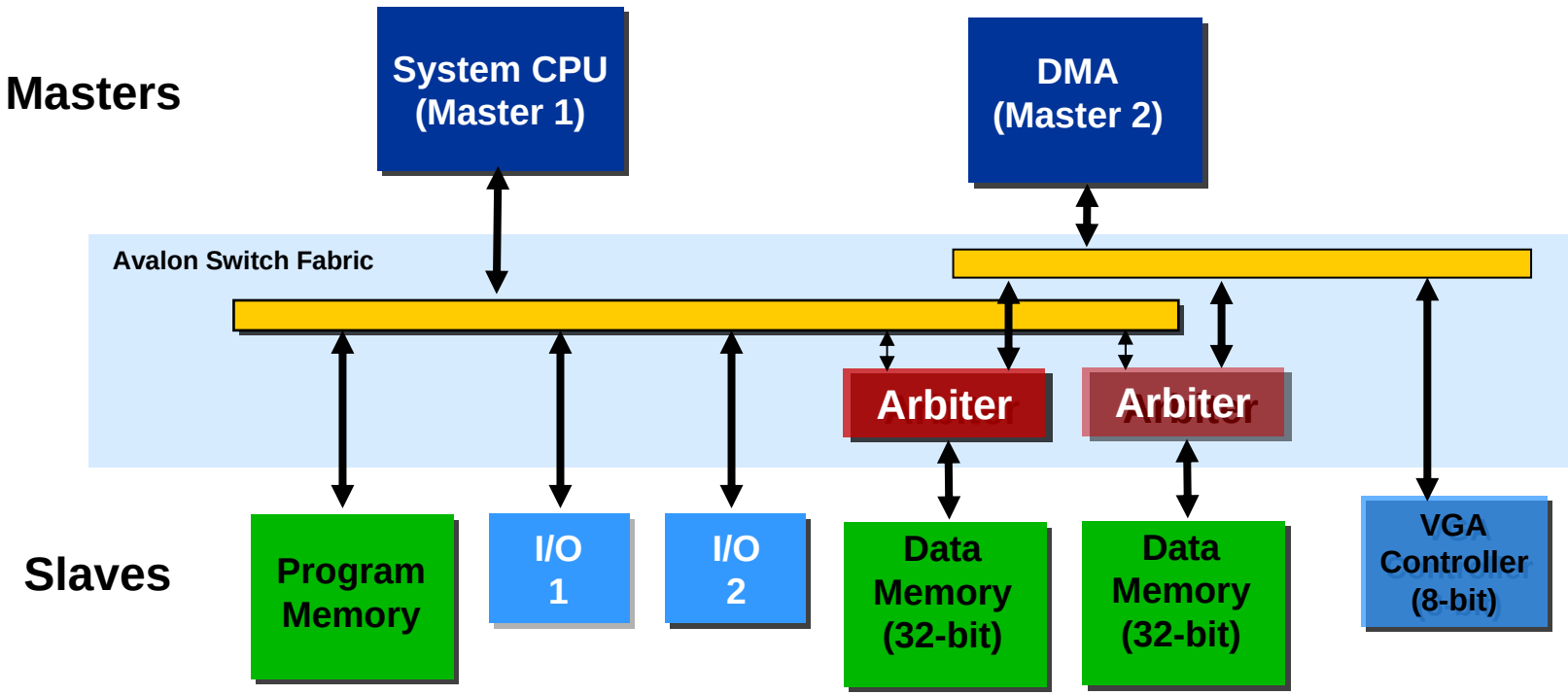
Avalon Signal Types

```
reset
chipselect
address
byteenable
read
readdata
write
writedata
data
waitrequest
readyfordata
dataavailable
datavalid
flush
begintransfer
endofpacket
irq
irqnumber
clk
resetrequest
```

All Signals Available In Positive or Negative Form

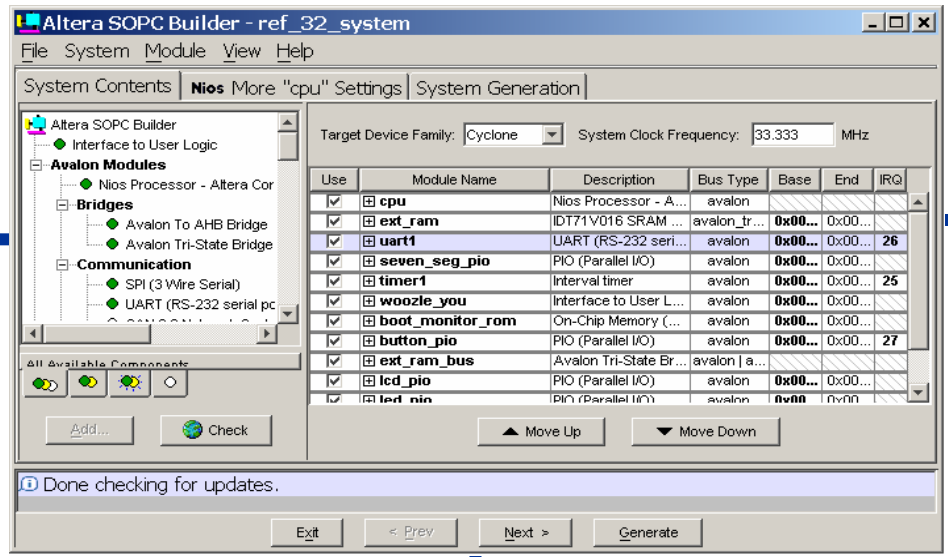
Avalon Switch Fabric

- Peripherals Can Be Dedicated or Shared
  - Shared Connections Arbitrated on the Slave-Side
- Simultaneous Data Transfers Boost Throughput

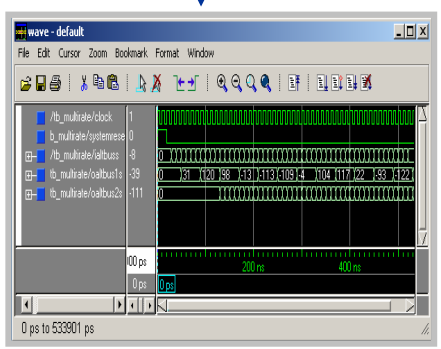


SOPC Builder: System Generation

Simulation Test Bench-Generation



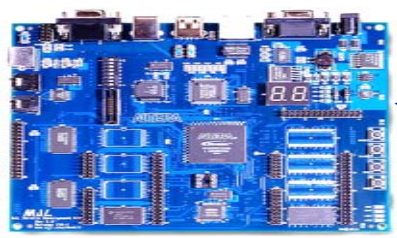
Software Development-Kit Generation



System Generation VHDL/Verilog HDL

QUARTUS® II

- C Header files
- Peripheral Drivers
- Software IDE
- Software Debuggers
- RTOS

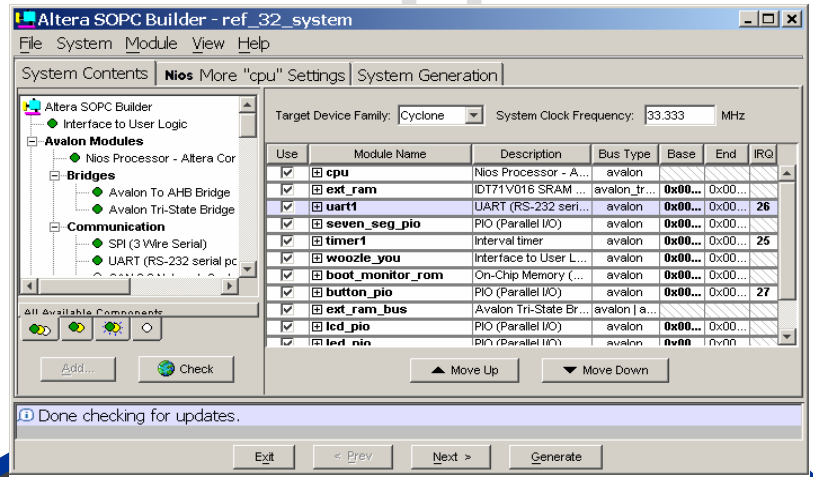


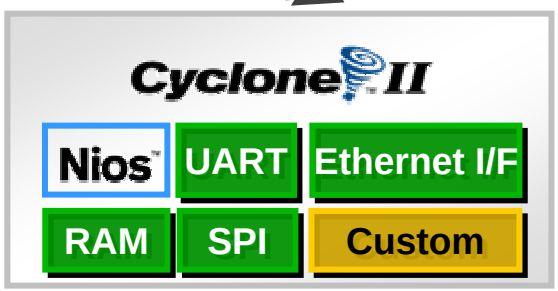
Model Technology
A MENTOR GRAPHICS COMPANY

SOPC WORLD 2004

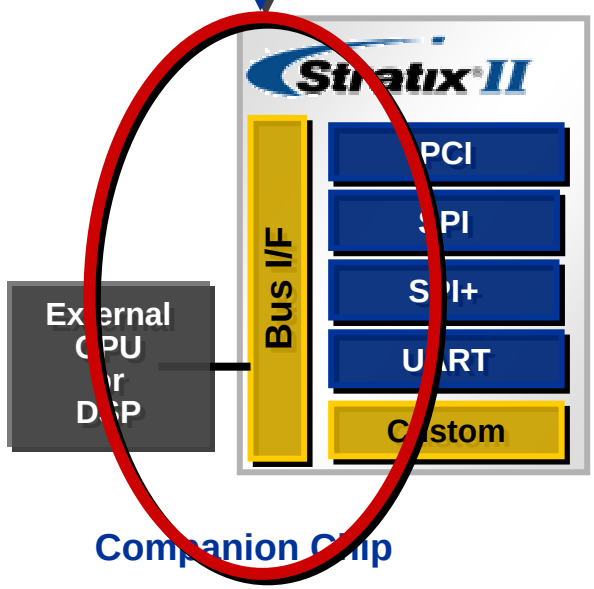
ALTERA®

SOPC Builder: Applications

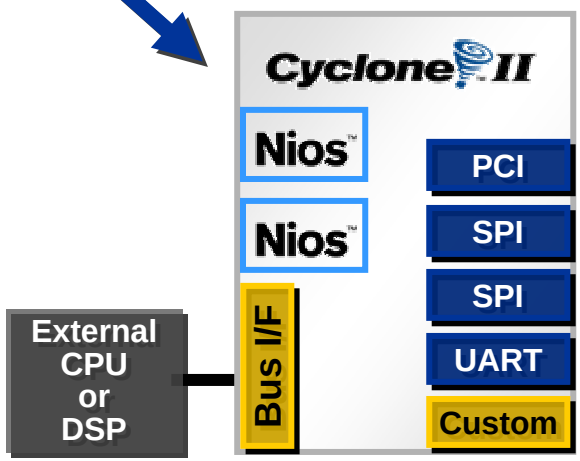




Custom Microcontroller



Companion Chip



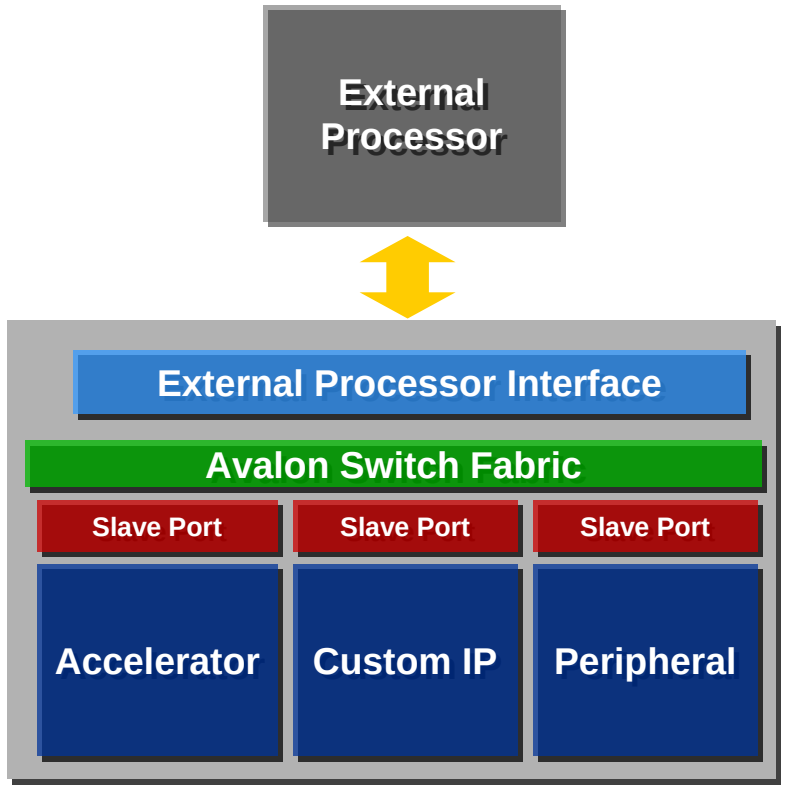
Multi-Processor

SOPC WORLD 2004

ALTERA

Interfacing External Processors

- Incumbent Processor or Digital Signal Processor
- Add System Elements
  - Peripherals
  - Custom Logic
  - Accelerators
- SOPC Builder Glues it Together
- Missing Element is the Interface



Classes of Processor Interfaces

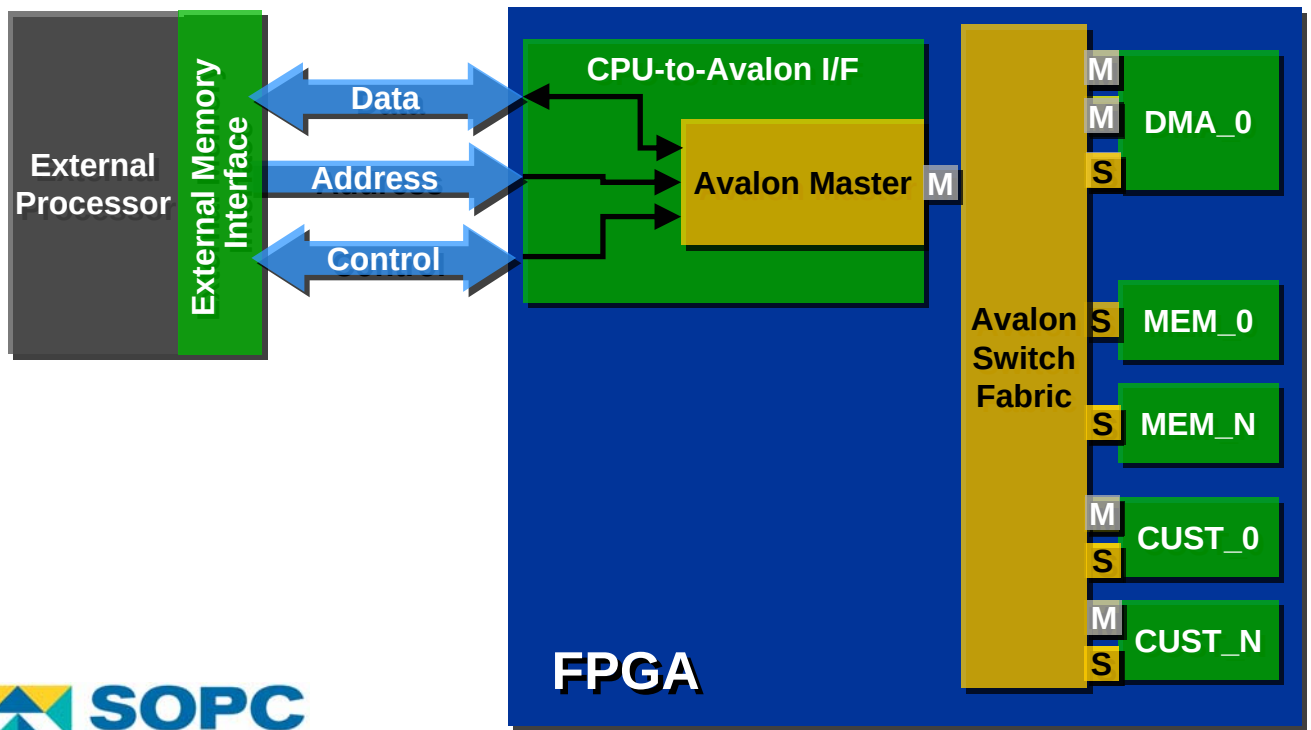
- Complex & Simple Interfaces
  - Complex
    - PCI(X)(Express), Rapid I/O™ & HyperTransport™ Technologies
    - Solved with Specialized Logic Blocks & Software Protocols
  - Simple
    - Common on Most Embedded Processors
    - Address, Data, Chipselect, R/W control, Etc.
    - Simple Software Control of Memory-Mapped Peripherals

Processor Interface Issues

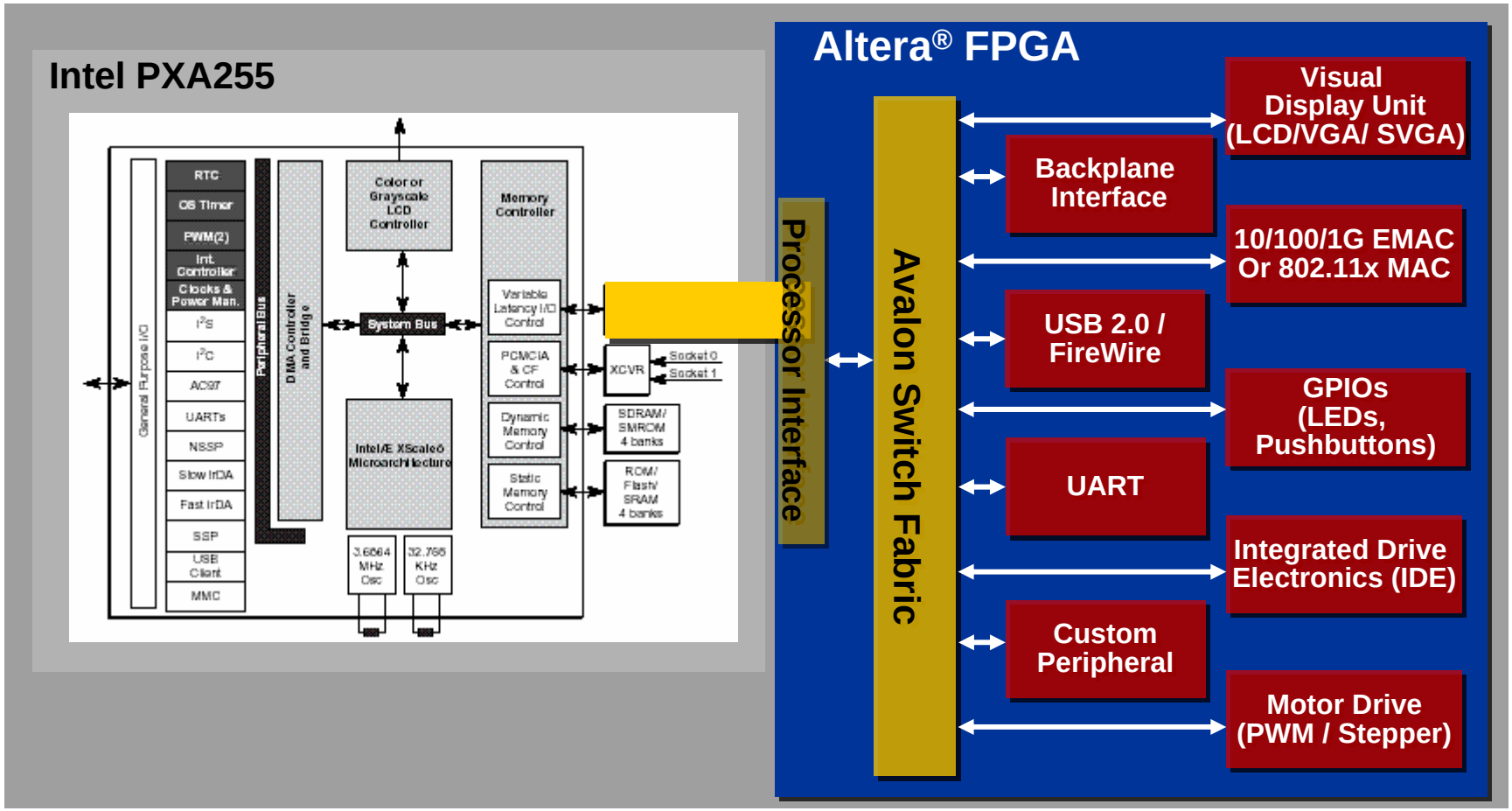
- Bus Frequency, I/O Timing, & I/O Standard
  - FPGAs Are Rich in I/O Standards
  - FPGA Tools Provide I/O Timing
  - High-Speed Busses Require High-Speed Design Techniques
    - See High-Speed Design Seminar Notes
- Control Signal Matching
- Data Transfer Latency
- Clock Domain Crossing

Simple Processor Interface

- Maps FPGA to a Memory Region, SOPC Builder Decodes Each Element
- Operates Synchronous to the Processor's Bus Clock
- Translates Processor Signal Types to Avalon Equivalents

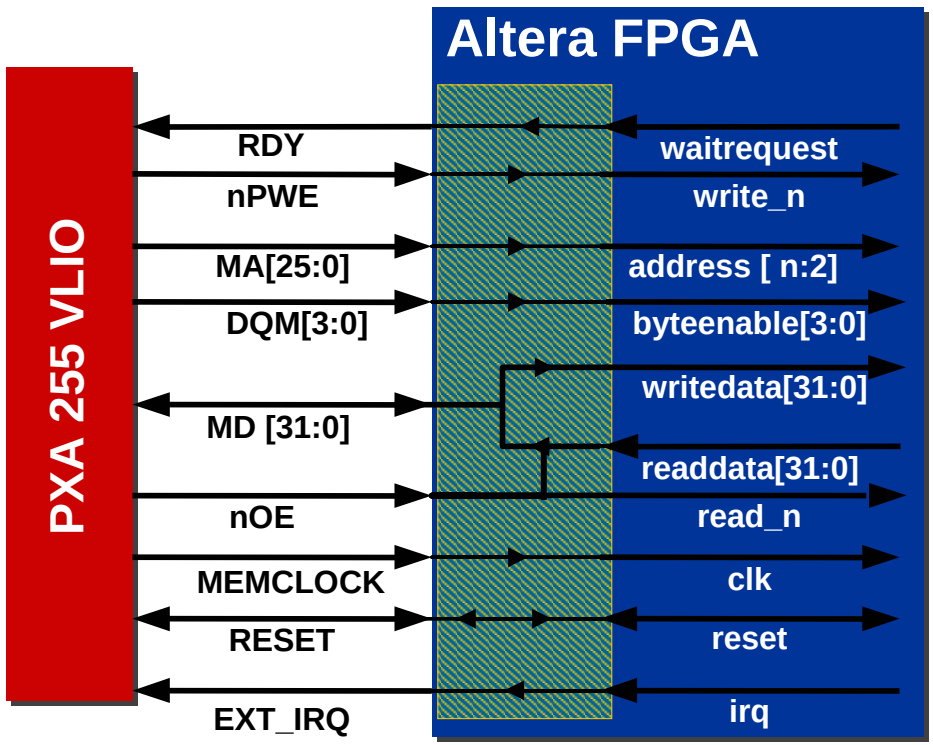


Intel PXA255 Example



VLIO as an Avalon Master Port

- Intel PXA255 Variable Latency I/O (VLIO) Uses a Bi-Directional Data Path, RDY Signal to Add Wait States
- Interface Separates DATA into Read Data & Write Data Paths



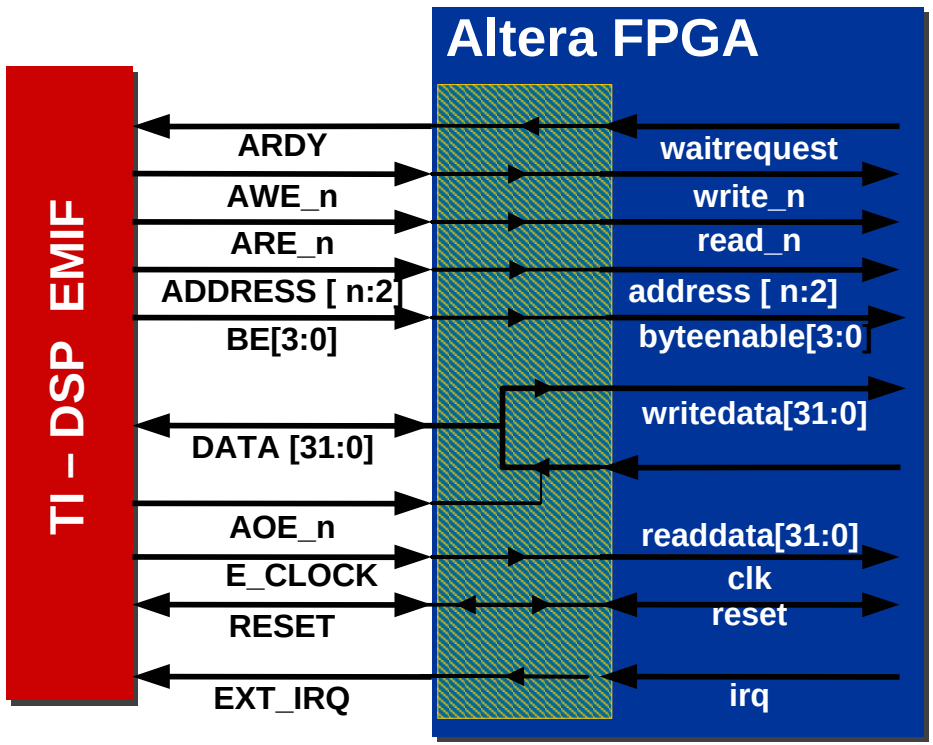
VLIO	Avalon
RDY	waitrequest
MA [25:0])	address
nPWE	write_n
MD[31:0]	writedata
nOE	read_n
MD [31:0]	readdata
DQM[3:0]	Byteenble[3:0]
EXT_IRQ	irq
MEMCLOCK	clk
RESET	reset

Relevant Verilog Code to Implement

```
module ext_proc_if (
    //port declarations
    )
    //signal declarations
    assign a_write_n = e_pwe_n;
    assign a_read_n = e_oe_n;
    assign a_addr = {e_ma, 2'b0};
    assign e_rdy = a_waitrequest;
    assign a_be_n = e_dqm_n;
    // work out the bi-directional data bus
    // if output enable is low, then get the data from the readdata path of the Avalon switch fabric
    assign e_data = (!e_oe_n &&  !e_cs_n)? a_data_read : 'bz;
    // assign the Avalon Switch Fabric write data path to athe a_data_write net (i.e. the incoming data..
    assign a_data_write = a_write_data;
    always @(!e_cs_n)
          begin
        if (!e_pwe_n ) a_write_data = e_data;
          end
    endmodule
```

EMIF to Avalon Master Port

- Texas Instruments (TI) DSP External Memory Interface (EMIF) Uses a Bi-Directional Data Path
- Interface Separates DATA into Read Data & Write Data Paths



EMIF	Avalon
ARDY	waitrequest
ADDRESS [21:2])	address
AWE_n	write_n
DATA [31:0]	writedata
ARE_n	read_n
DATA [31:0]	readdata
BE[3:0]	byteenble
EXT_IRQ	irq
E_CLOCK	clk
reset	reset

Relevant Verilog Code to Implement

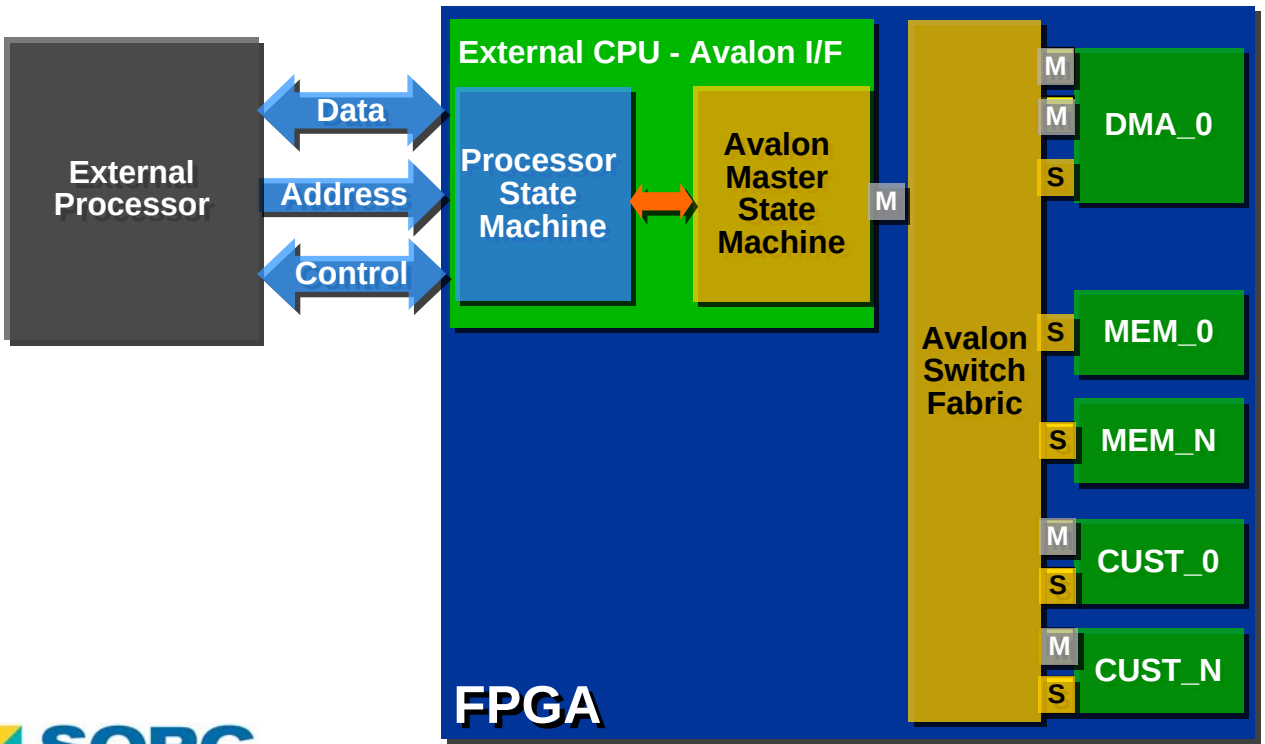
```
module ext_proc_if (
    //port declarations
    )
    //signal declarations
    assign a_write_n = e_write_n;
    assign a_read_n = e_read_n;
    assign a_addr = {e_addr, 2'b0};
    assign e_rdy = a_waitrequest;
    assign a_be_n = e_be_n;
    // work out the bi-directional data bus
    // if output enable is low, then get the data from the readdata path of the Avalon switch fabric
    assign e_data = (!e_oe_n && !e_read_n && !e_cs_n)? a_data_read : 'bz;
    // assign the Avalon Switch Fabric write data path to athe a_data_write net (i.e. the incoming data..
    assign a_data_write = a_write_data;
    always @(!e_cs_n)
        begin
      if (!e_write_n ) a_write_data = e_data;
        end
    endmodule
```

Simple Processor Interface

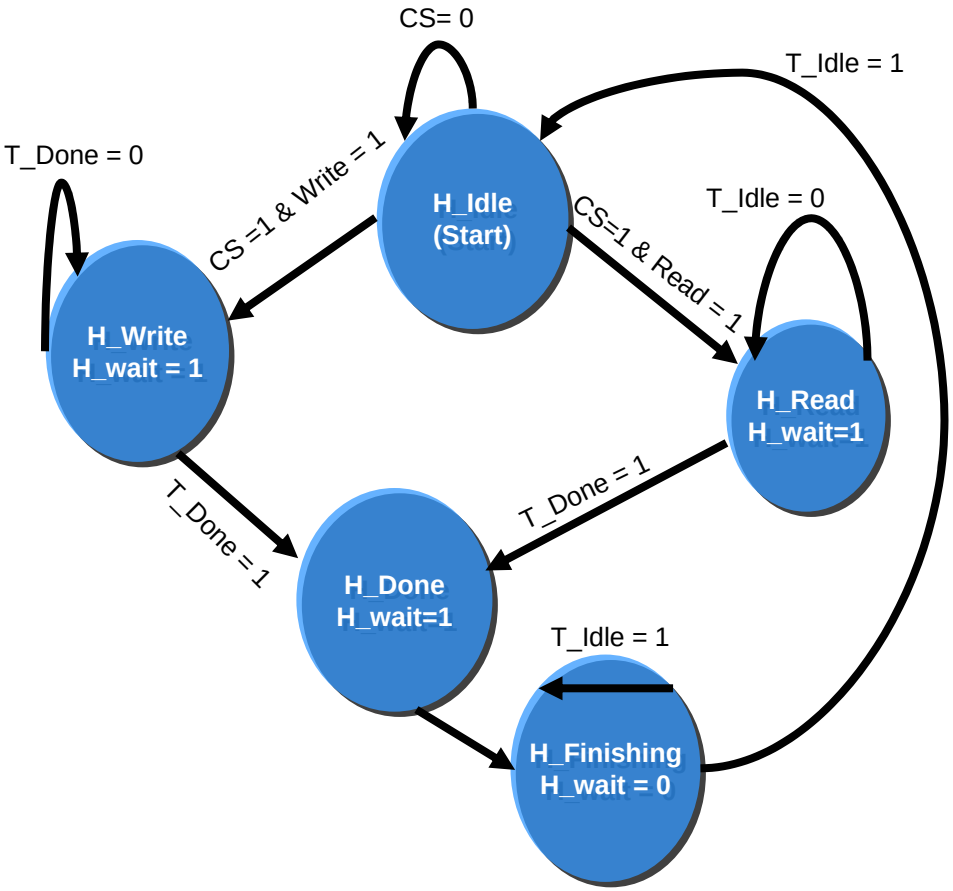
- Advantages
  - Simple as Wires—1 Logic Element (LE) to Control Data Bus I/O Direction
  - Easy Access to Peripherals for the Incumbent Processor
  - Limitless Options for Adding Components
- Restrictions
  - Processor Bus & FPGA Logic Operate on Single Clock Domain
  - Processor Must Honor *waitrequest*

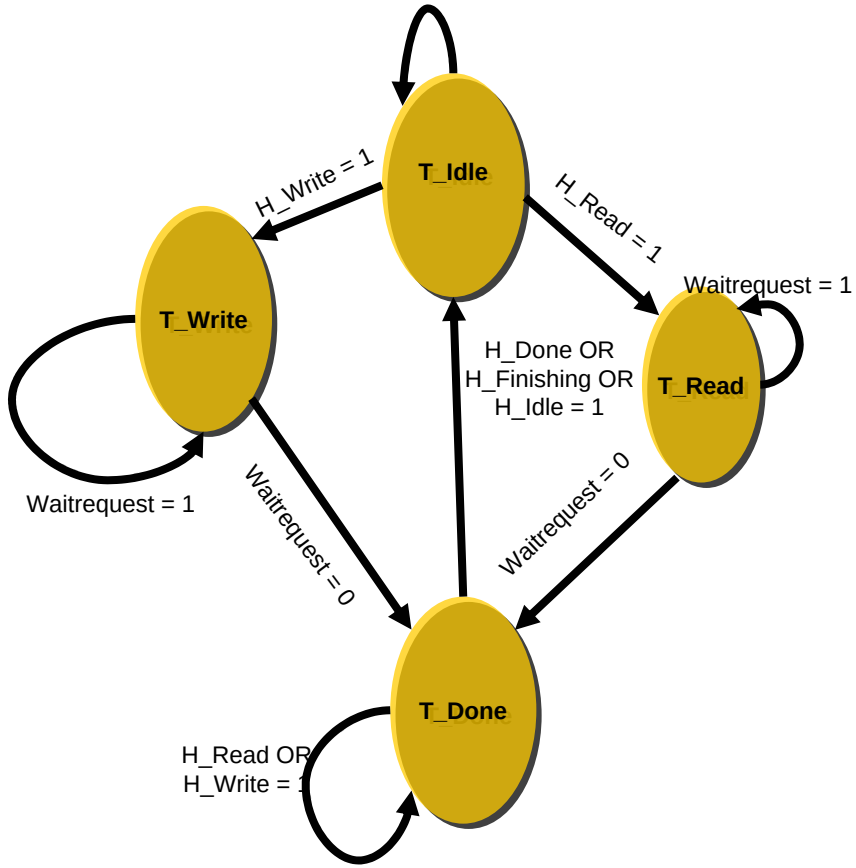
Removing Clock Domain Restrictions

- State Machines Pass Transactions Between Clock Domains
- Processor State Machine Initiates Transaction
  - Waits for Avalon Master State Machine To Complete
- Avalon Master State Machine Initiates Transaction with Avalon Switch Fabric
  - Waits for Avalon Switch Fabric to Complete



State Machine Control Diagram





External CPU or Digital Signal Processor (Host) SM Diagram

Avalon Master SM Diagram

SOPC WORLD 2004

ALTERA

State Machine Control

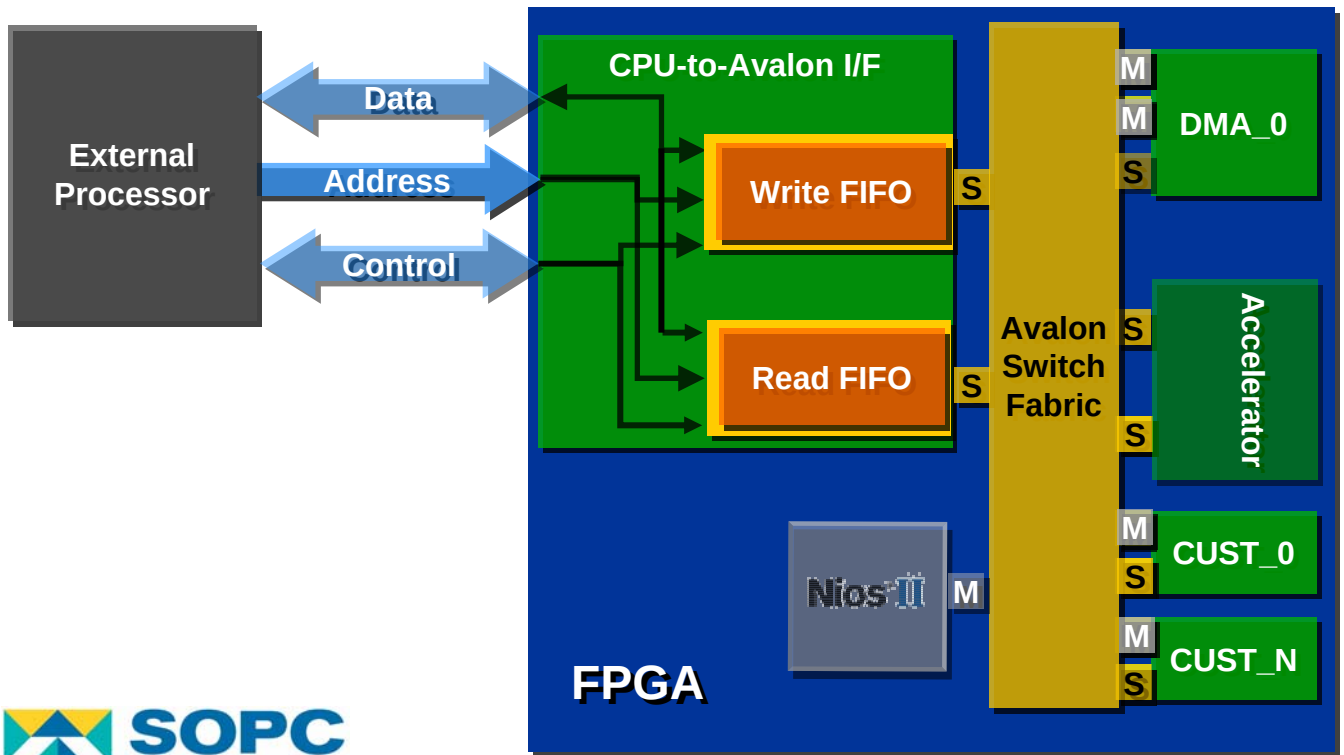
- Advantages
  - Processor & FPGA Operation Decoupled
  - Each Side Free to Operate at f_{MAX}
- Restrictions
  - Adds Size & Complexity to Core
  - Adds 3-5 Cycles of Latency per Transaction

Removing *waitrequest* Restriction

- Fixed Timing Between Processor & Interface
  - FIFO or DPRAM
  - Additional Logic Required to Move Data In/Out of FIFO / DPRAM
- FIFO-Based Interface
  - Single Address, No Random Access
  - Excellent for Data Streaming Applications
- Dual-Port RAM-Based Interface
  - Allows Fully Loaded Memory with Random Access
  - Excellent for Multi-Processor Applications

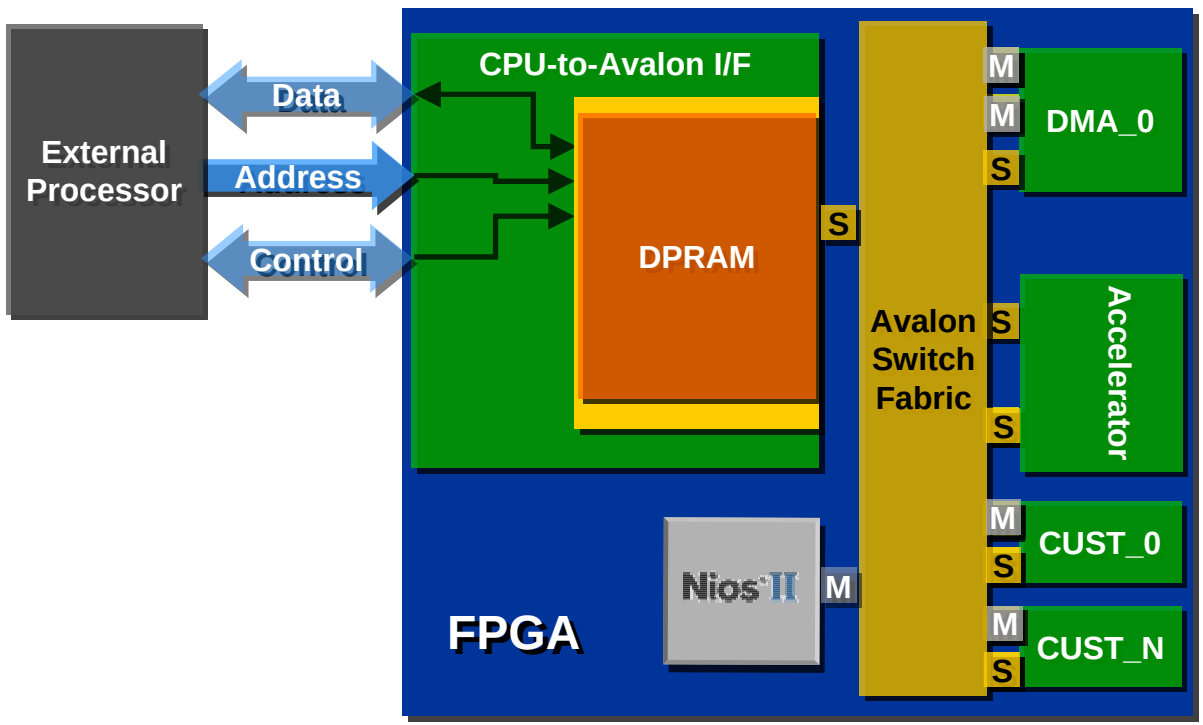
FIFO-Based Interface

- FIFO Used to Stream Data to/from a Fixed Destination
  - Custom Accelerators
  - Streaming Devices
- Data Movement Controlled by Nios II Processor, DMA Controllers, Custom Logic
- FIFO Depth Affects Data Throughput



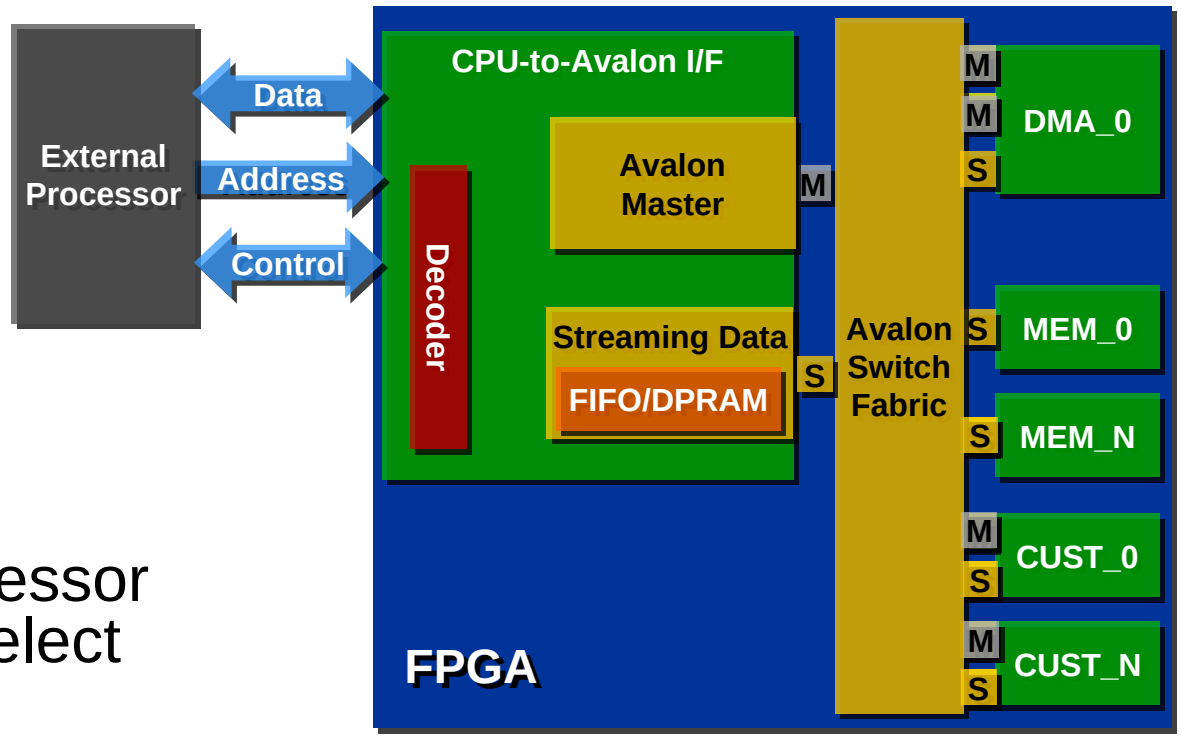
DPRAM-Based Interface

- DPRAMs
  - Random Access
  - Mailbox Access
- DPRAM Has Fixed Cycle Timing
- Data Movement on DPRAM Slave Controlled by:
  - Soft Embedded Processor
  - DMA on FPGA
  - IP Accelerator



Combination of Interfaces

- Variable Cycle Interfaces
  - Random Access Peripherals
  - Control Variables
- FIFO/DPRAM
  - Low-Latency
  - Streaming Data
- Use External Processor Chip_Selects to Select the Interface



Interface Bandwidth & Throughput

- Determined by Your Processor Bus
  - Width of Bus, Frequency of Bus, Number Cycles per Transfer
- Dedicated or Shared Peripheral
  - SOPC Builder Easy to Define
  - Slave-Side Arbitration
- Latency
  - Processor: Inherent by Design
  - Processor Interface: Depends on Type Used
  - Avalon Switch Fabric: 0 Cycles, Combinatorial
  - Peripherals: Defined by the Peripheral

Summary

- Multiple Processor Interface Options
  - Simple Processor Interface
  - Asynchronous Interface with State Machines
  - FIFO-Based Interface
  - DPRAM-Based Interface
  - Combination Interfaces
- SOPC Builder Automatically Integrates Additional Components
  - Converts Your Concepts into Systems in Minutes

Additional Resources

- SOPC Builder
  - Included in All Quartus II Products
- Evaluation Nios II & IP Cores
  - MegaCore IP Library CD
  - Nios II Embedded Processor Evaluation Edition
- www.altera.com
  - Literature Section – SOPC Builder
  - Complete List of Application Notes
  - Wide Range of Tutorials

SOPC WORLD 2004

Multi-Processor Systems in FPGAs: Implementation & Debug

ALTERA®

Multi-Processor Systems Address these Challenges

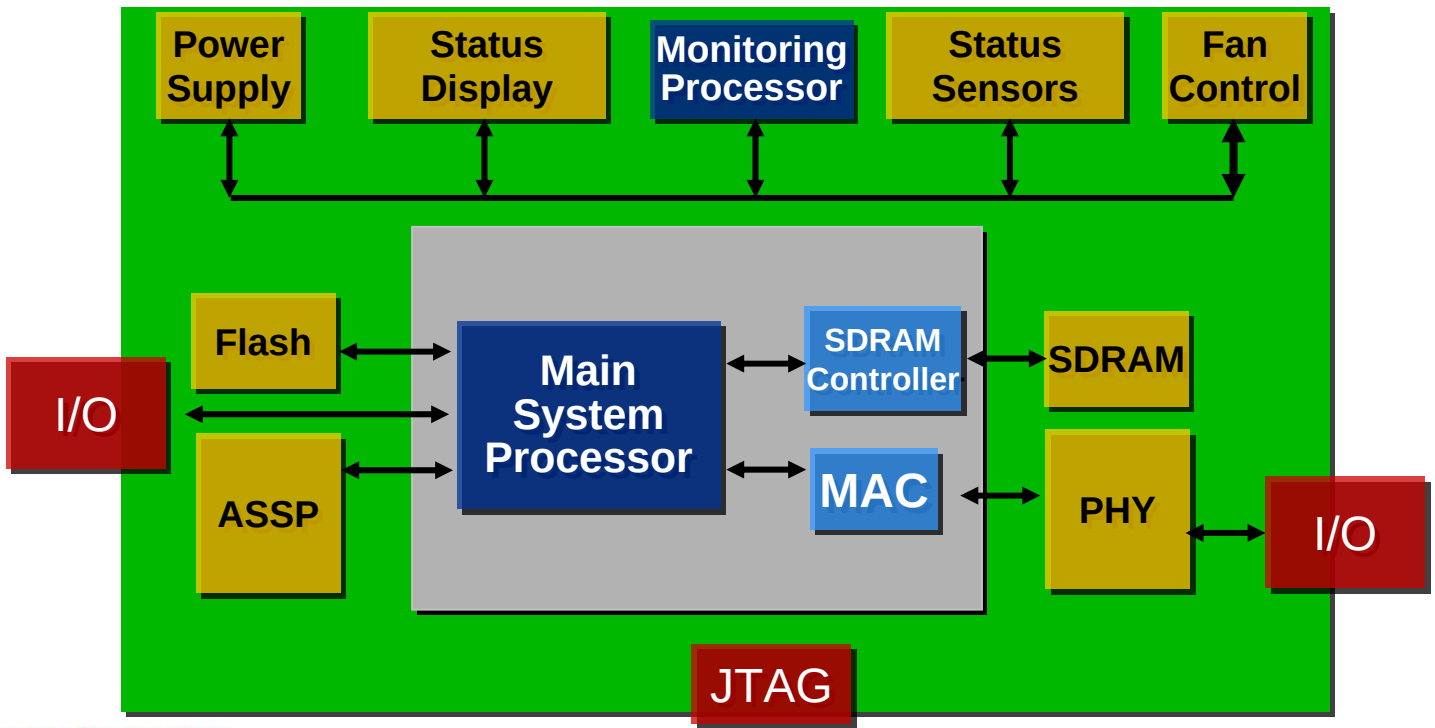
- Increasing Application Requirements
  - Single Processor → Faster Clock Rate → Higher Power
  - Multi–Processors → Slower Clock Rate → Lower Power
  - Execute Tasks in Serial or Parallel for Highest Performance
- Expanding Product Functionality
  - Custom-Fit Processors for Specific Functions
- Simplify Application Software
  - Limit Code to the Task at Hand

Multi-Processor Architectures

- Multiple Independent Processors
- Host with Off-Load Processors
- Serially Linked Processors
- Multi-Channel Processors

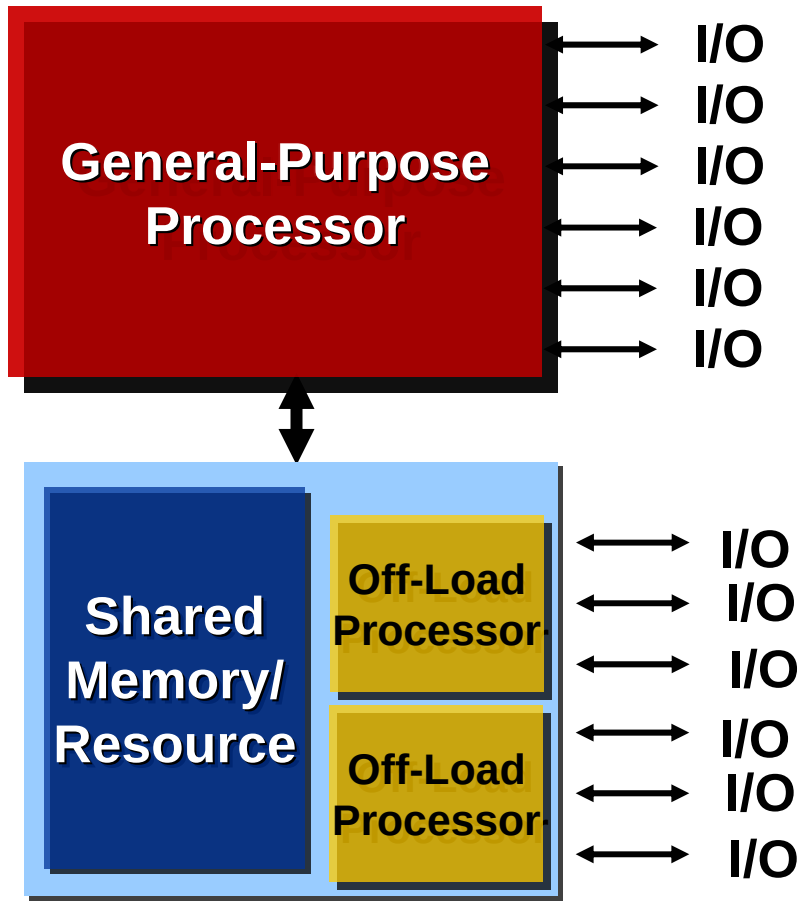
Multiple Independent Processors

- Main System Processor
  - Primary Application
- Secondary Processors
  - System Monitoring & Reporting Functions



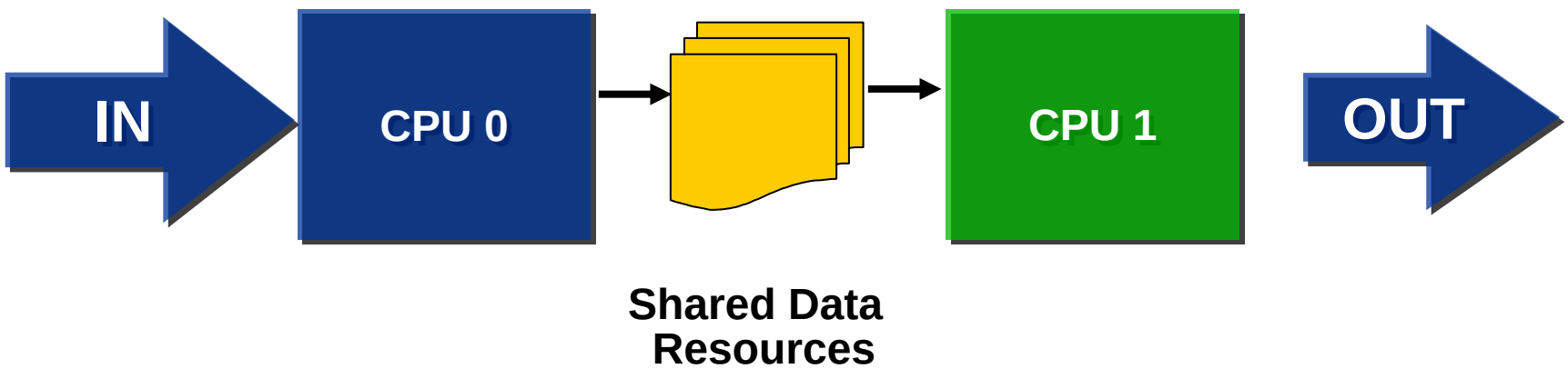
Off-Load of Host Processor

- Application or I/O Requirements Not Being Met by CPU
- Add an Off-Load CPU
- Additional Off-Load CPU(s) to Further Address Performance or Functional Requirements



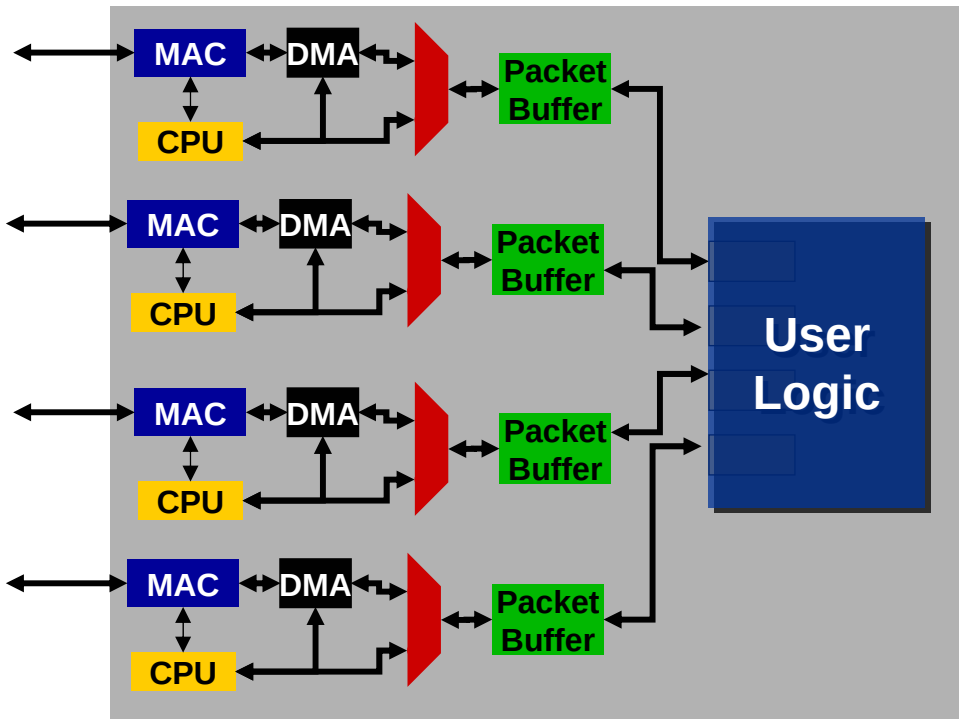
Serial Processing

- Application Divided into Sequential Tasks
  - One Processor per Task
  - Simplify Software
- Processors Communicate Through Shared Data/ Resources



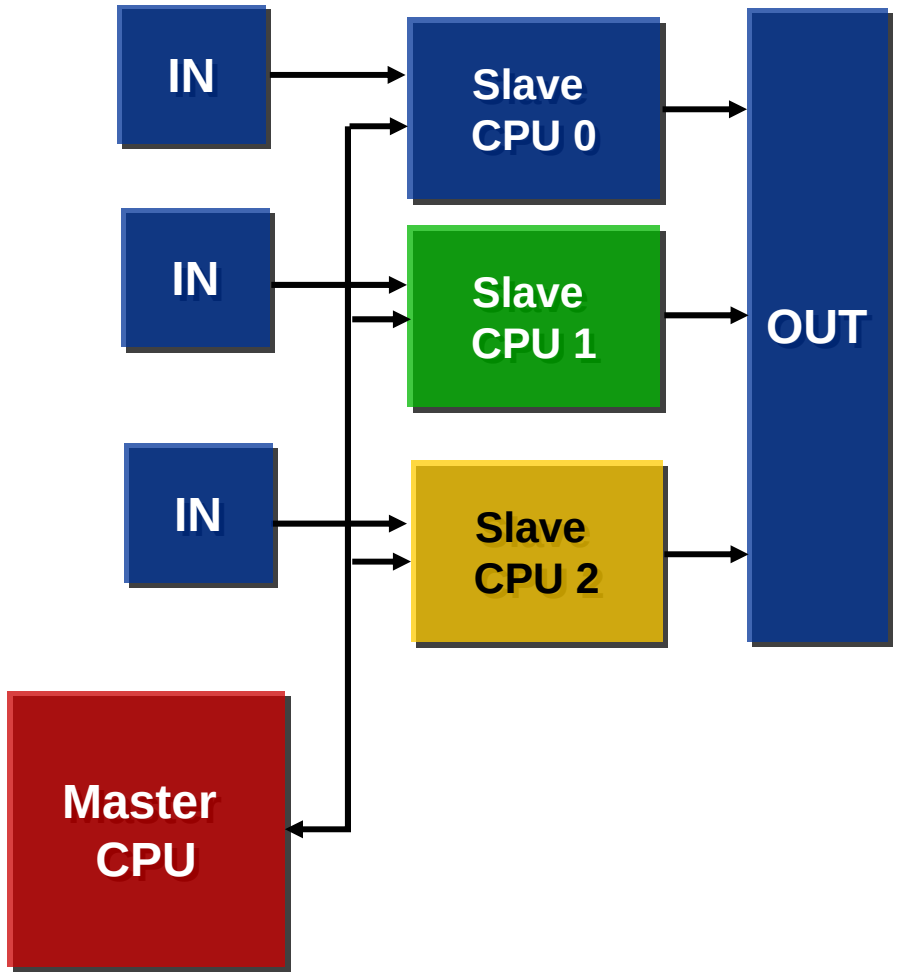
Multi-Channel Processing

- Develop a Single Channel Solution
- Replicate to More Channels



Master/Slave Processor System

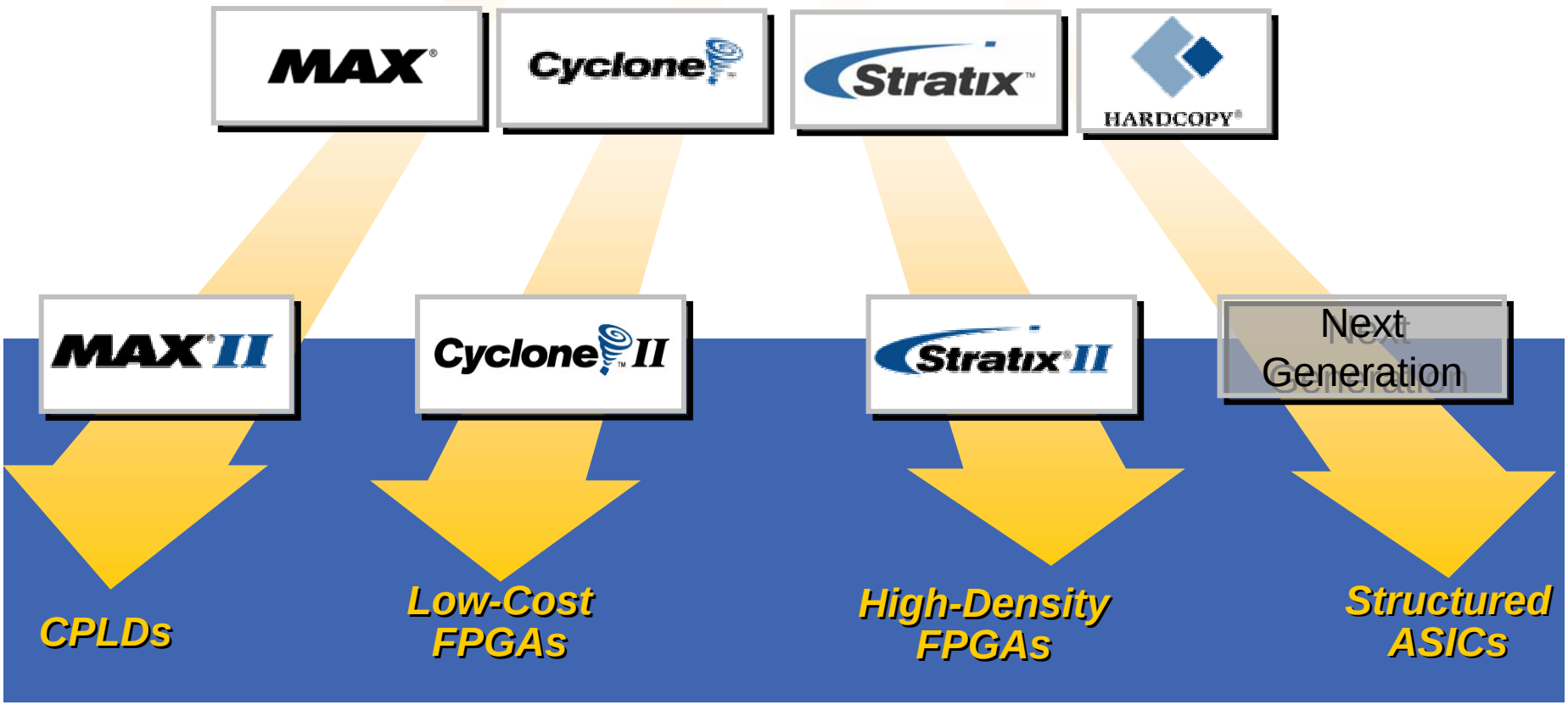
- Master Processor
  - Running Operating System
  - Primary Application(s)
- Slave Processors
  - Operate Under Master Processor Control
- Examples
  - Load Balancing in Web Server
  - Control Processor with Multiple Processor Engines for Packet Processing



Design Considerations

- System Development Environment
  - FPGAs
  - Soft Embedded Processors
  - System Development Tools
- Inter-Processor Communication
- Resource Sharing
- Software
  - Project Management
  - Collective Run & Stop Control

Altera's Product Offerings



Nios: Most Popular Soft Processor Ever

- Over 14,000 Kits Shipped
  - Over 4,000 Unique Nios® Licensees (Companies)
  - More than All Other Embedded Soft Processors Combined *
- EDN Hot 100 Product of 2003
- Keys to Success
  - Complete Kit
  - Simple, Capable CPU
  - Easy to Use
  - Low Cost ($995)
  - Perpetual License
  - No Royalties

* Semico: Soft-Core Processor Licensees

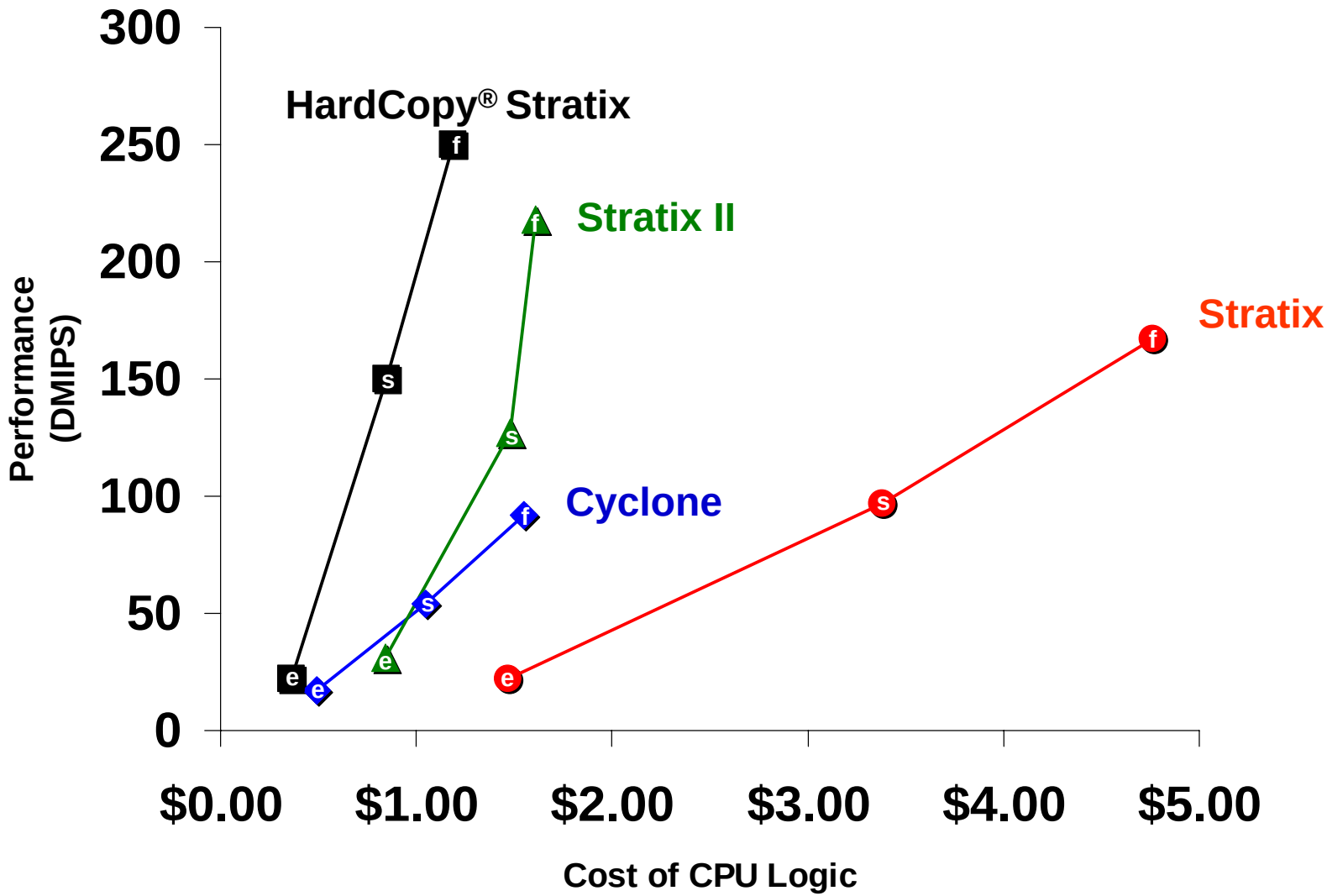
SOPC WORLD 2004

ALTERA

Nios II Overview

- Family of 3 Processor Cores
  - Performance Over 200 DMIPS
  - Cost as Low as 35¢ of Logic
- Powerful New Software Development Tools
  - IDE/Debugger, RTOS, TCP/IP Stack
- Complete Portfolio of Development Kits
  - Stratix® , Stratix II & Cyclone™ Kits
  - Application-Specific Expansion Boards

Processor Cost vs. Performance



FPGA Processor Core Advantages

Low-Cost Embedded Solution

Processor?

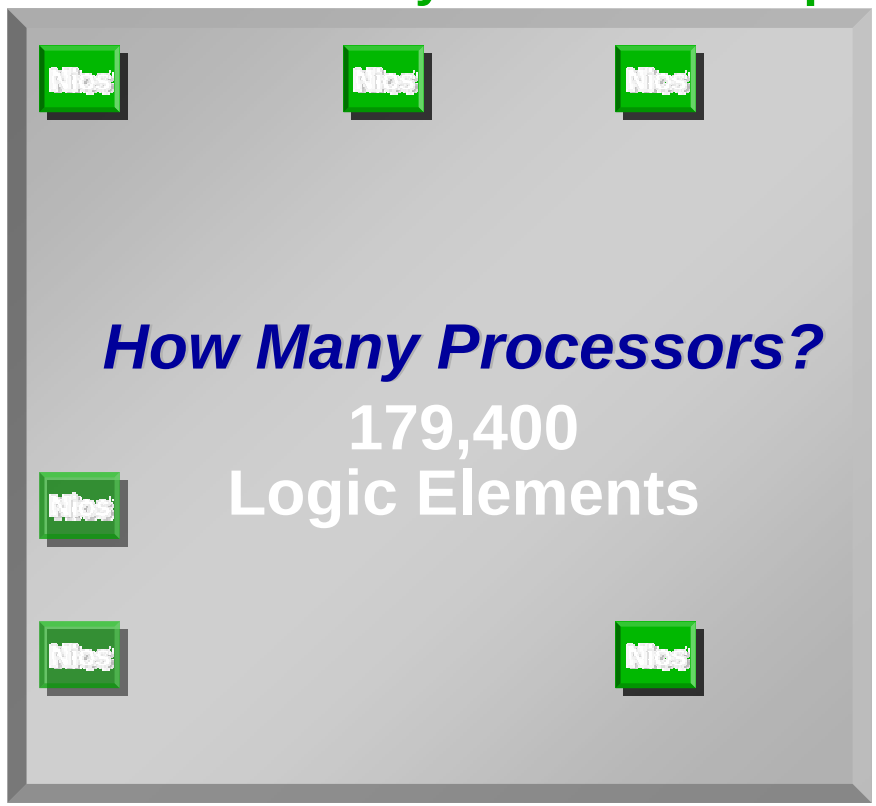
Cyclone Series & Nios II Economy

CPU < 20% of Device

20 DMIPs

As Low as 35¢

Complex Embedded System-on-a-Chip

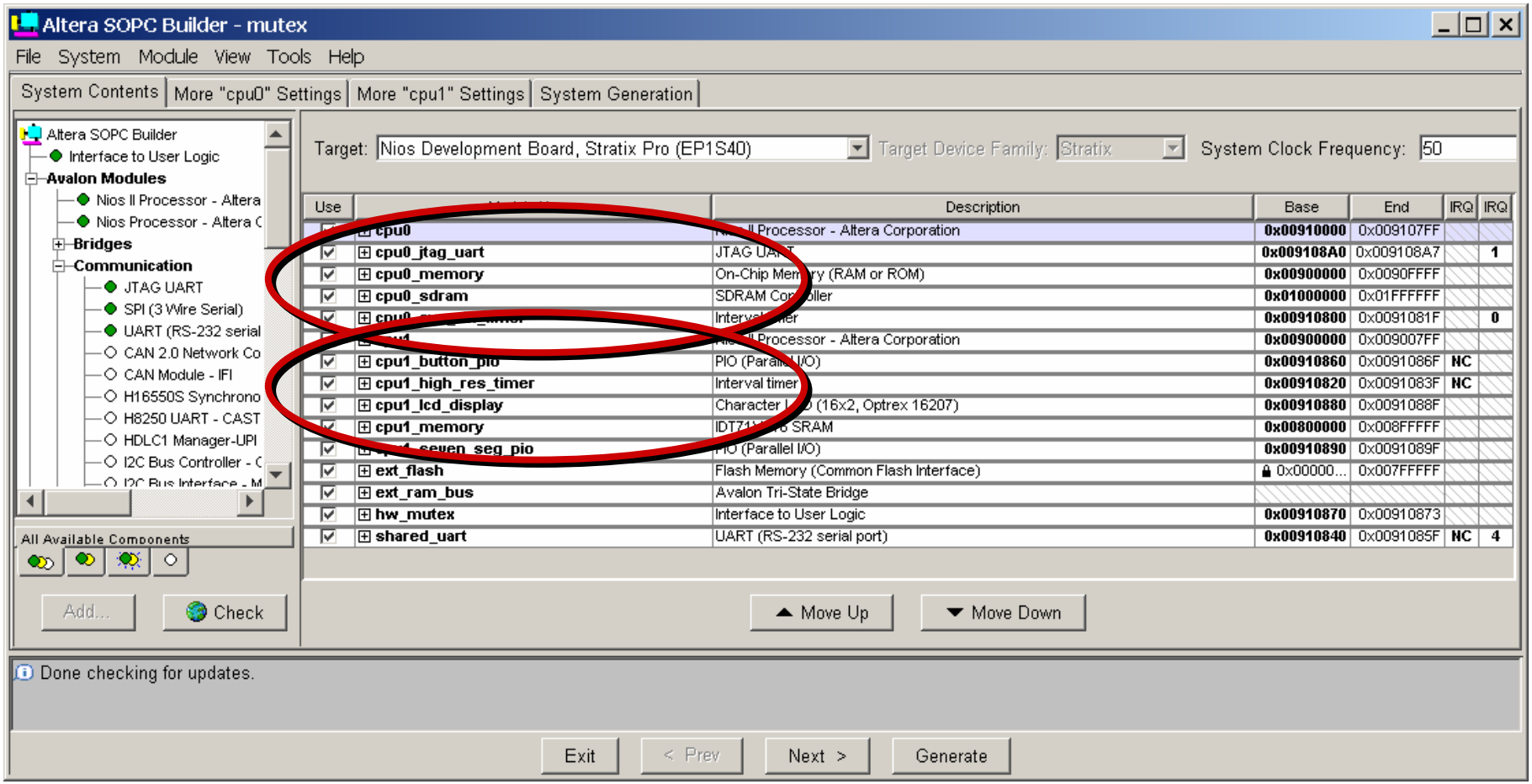


Stratix II EP2S180 & Nios II Fast

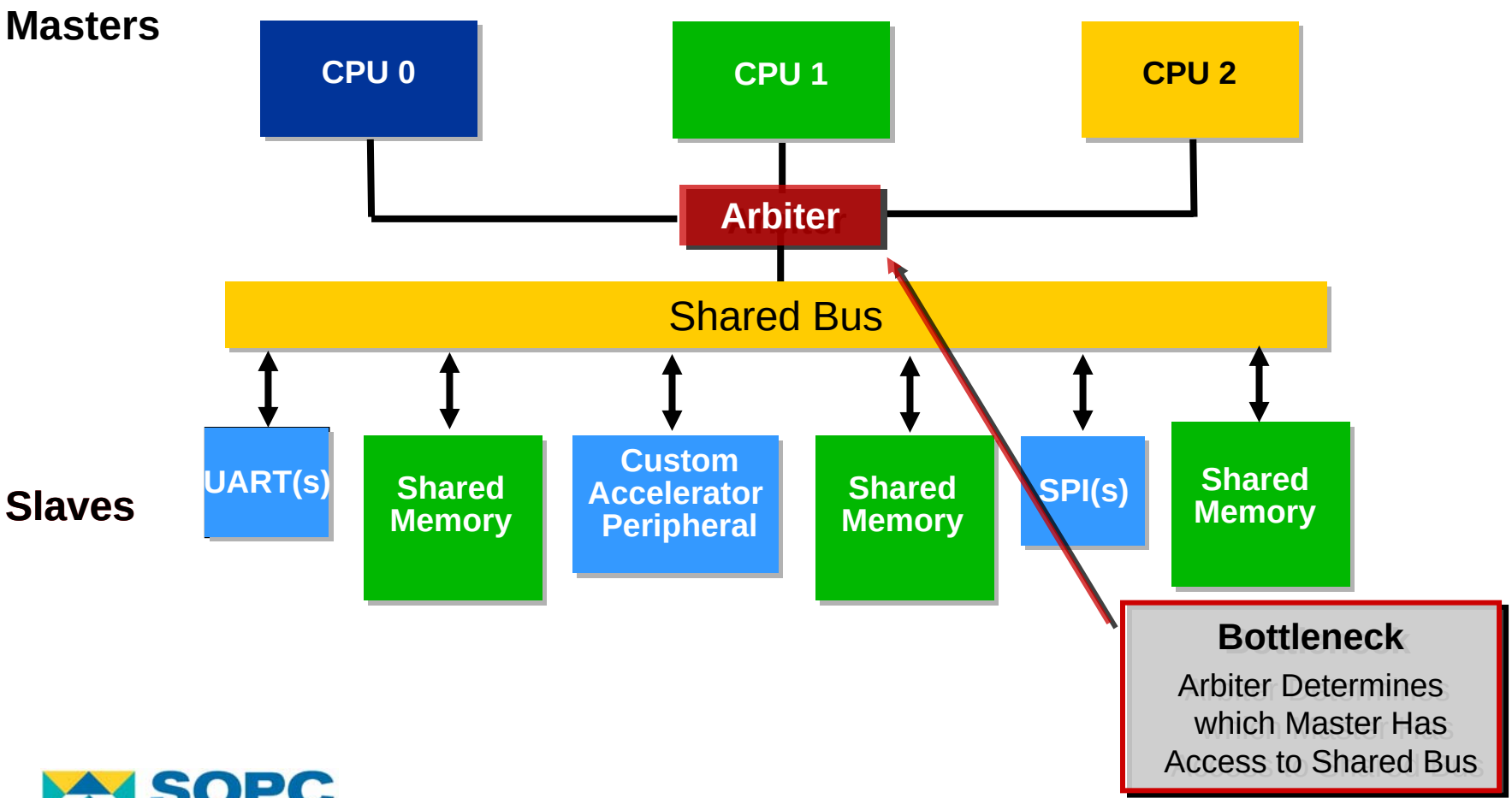
CPU 1% of Device

220 DMIPs (each)

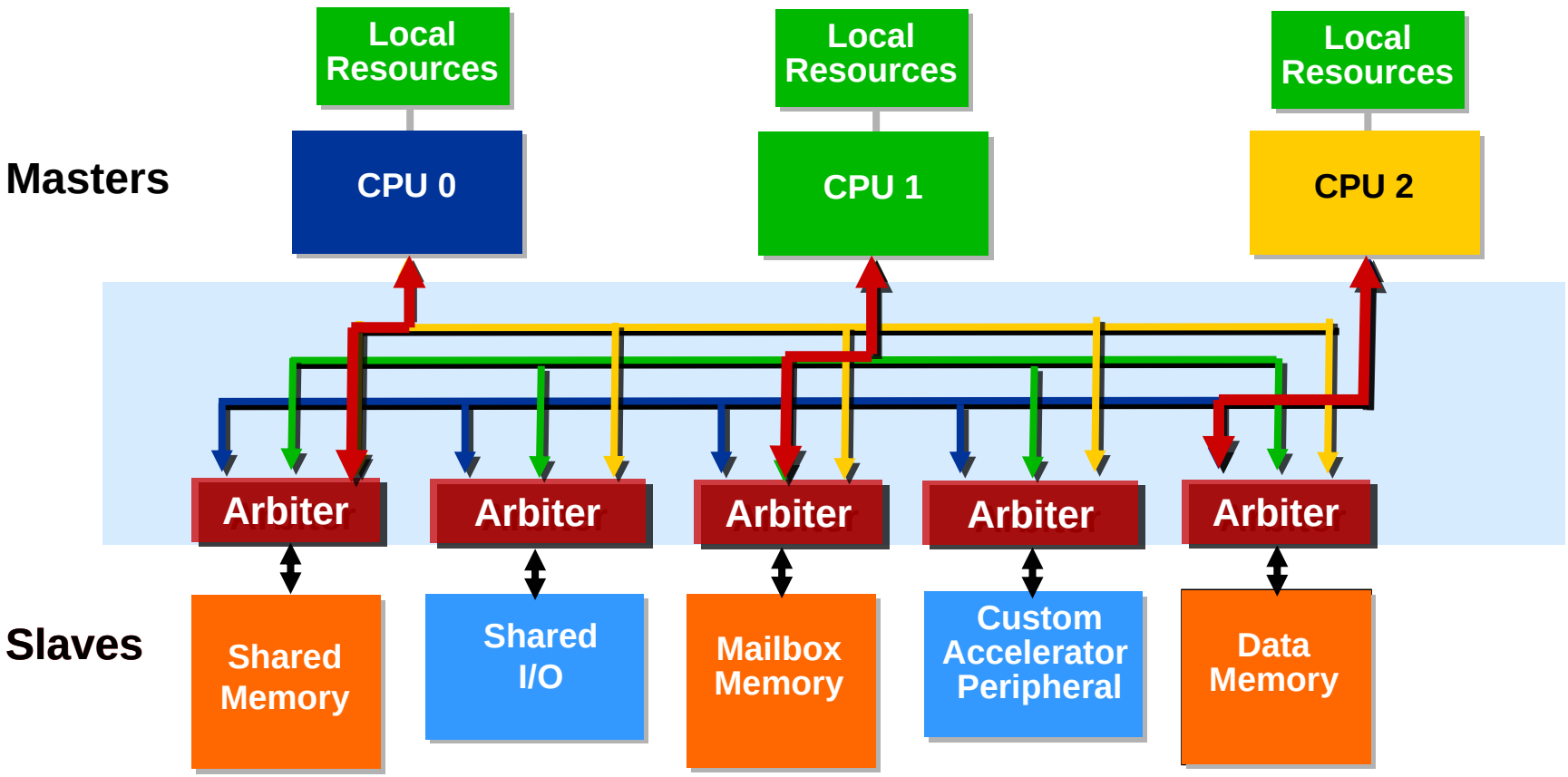
Development Tools – SOPC Builder



Host-Side Arbitration Scheme

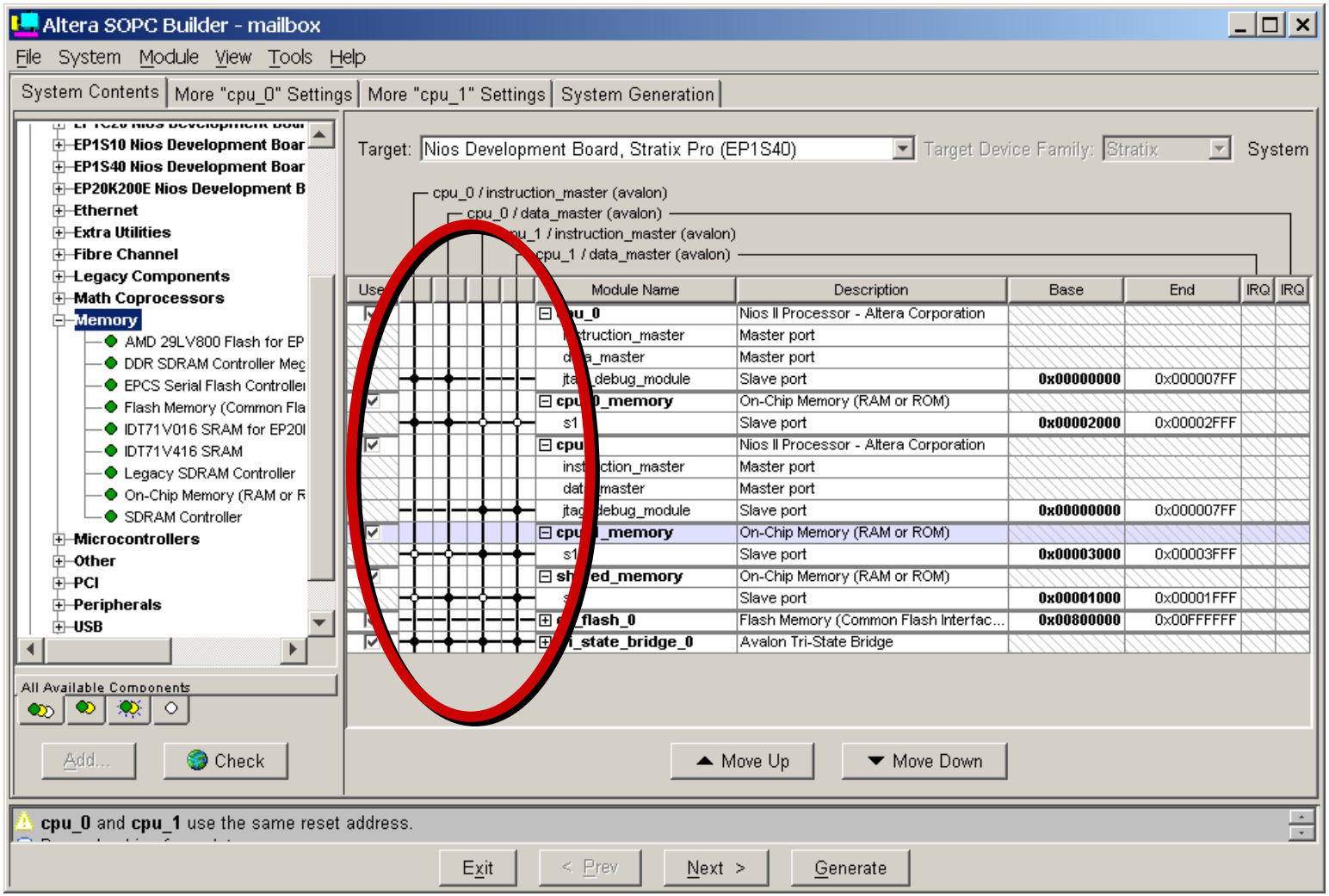


Slave Side Arbitration



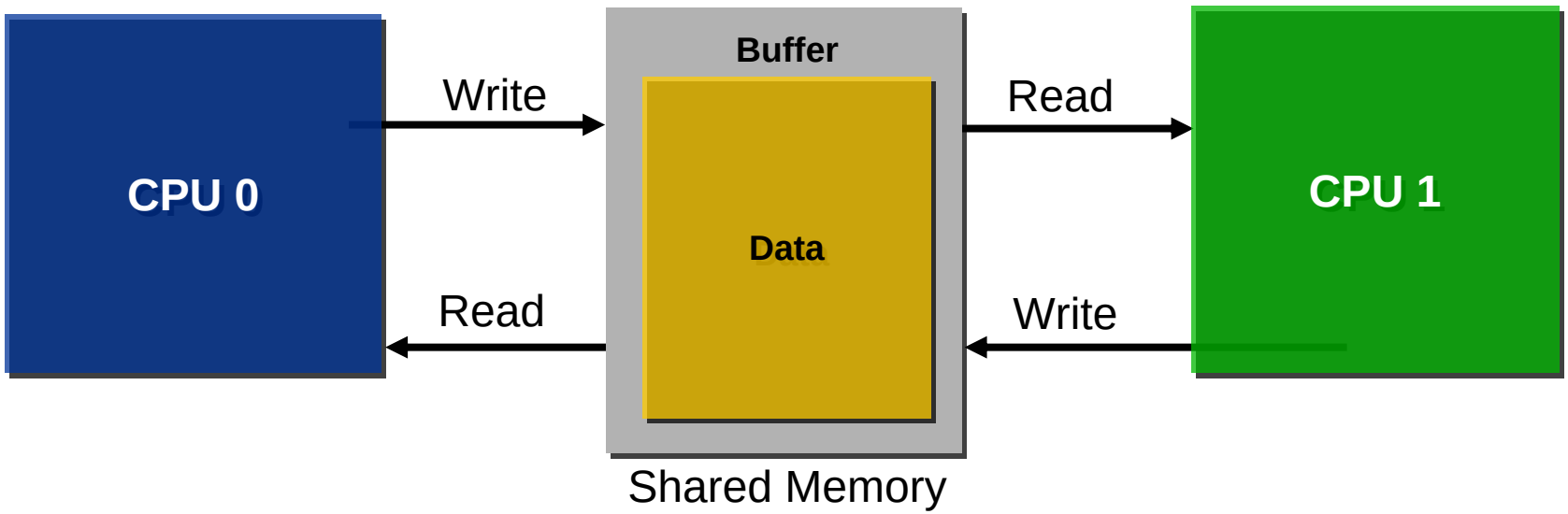
Simultaneous Operation for All Masters

SOPC Builder – Slave Side Arbitration



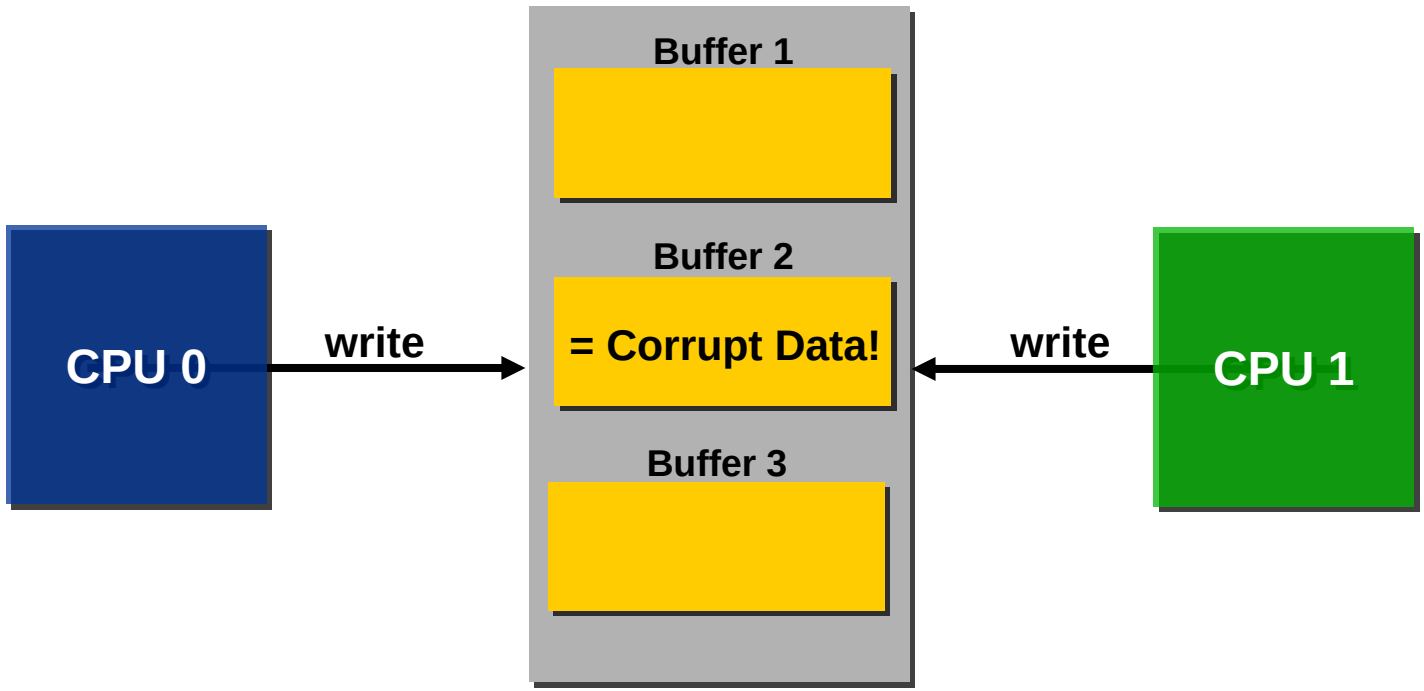
Inter-Processor Communications

- Memory Management
  - Access Control
  - Data Integrity
  - Amount of Memory



Sharing Memory

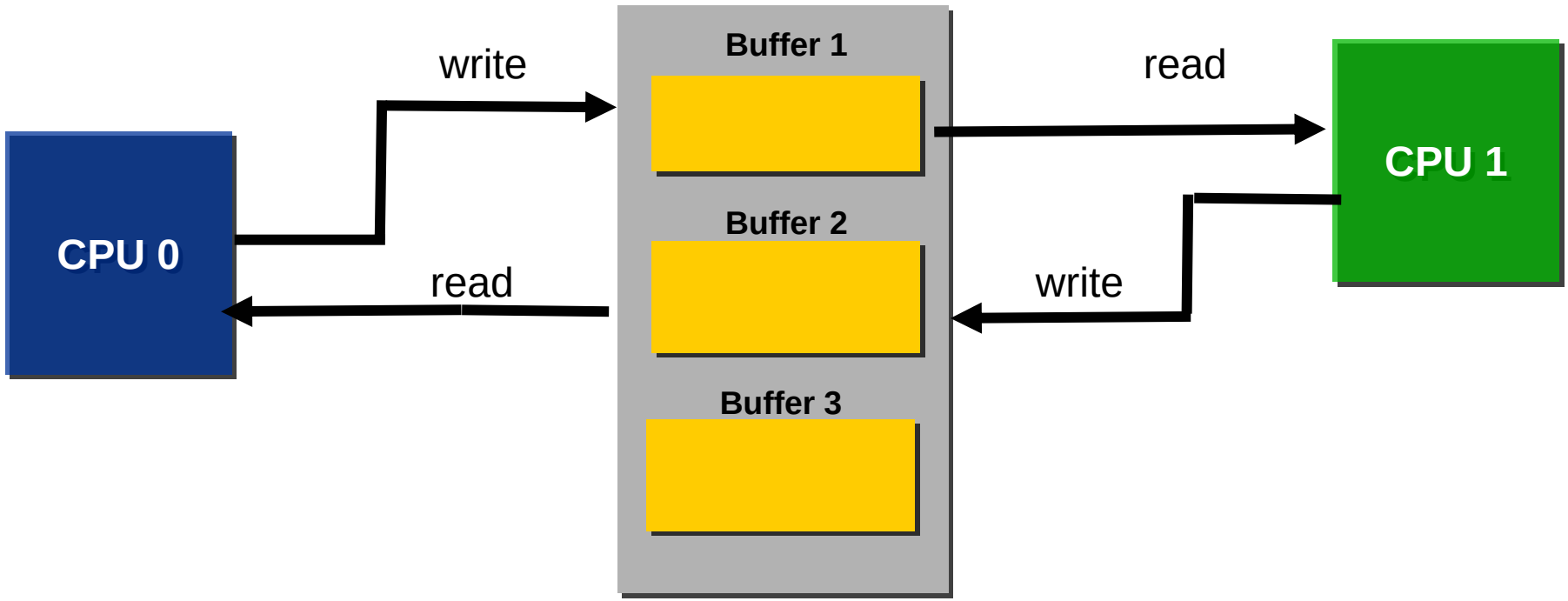
- Potential Problems with Shared Memory



Processors Writing to the Same Buffer

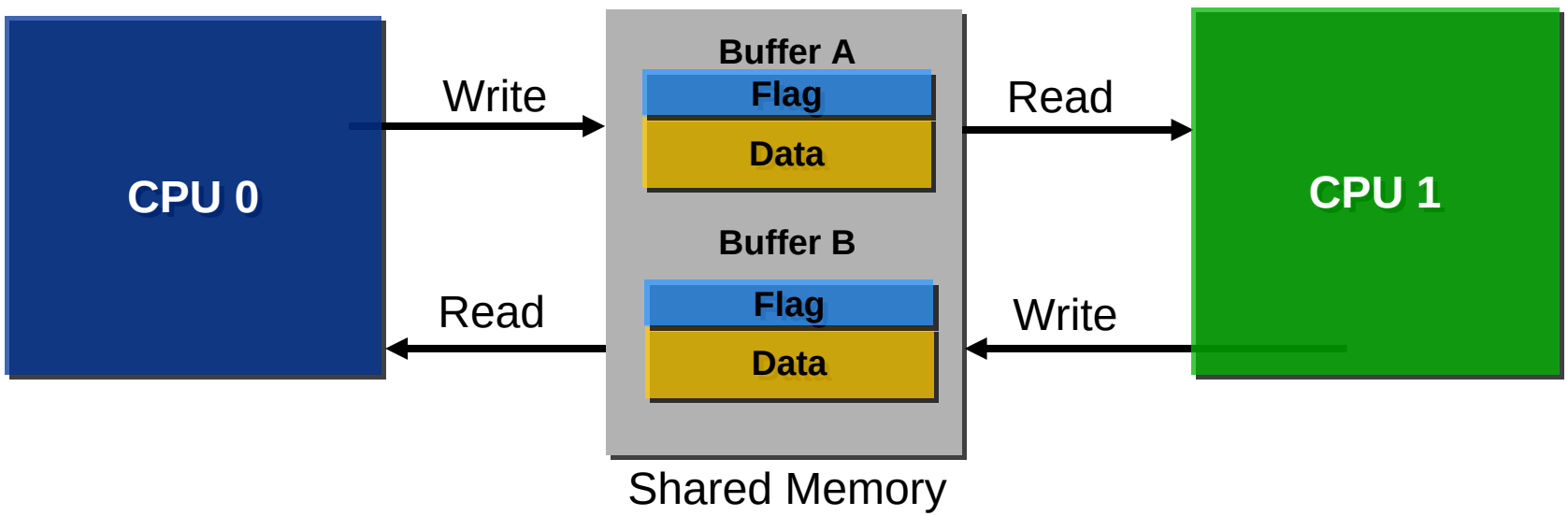
Sharing Memory

- Partition & Allocate Memory by Processor
- Limit Access in Application Software

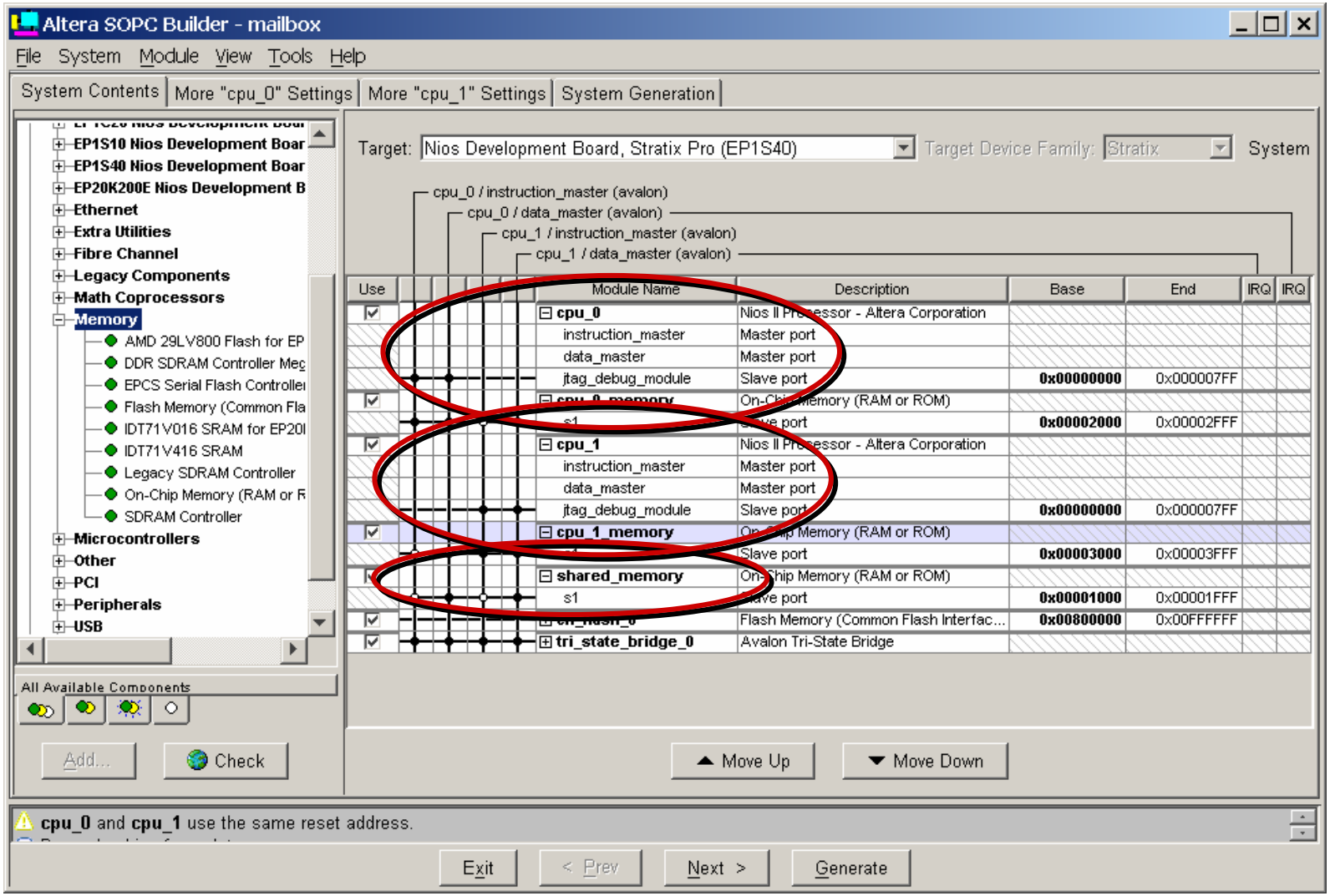


Sharing Memory – Mailboxes

- Software Protocol Controls Exchange of Data
  - Flags Signal Data Ready or Taken
- Independent Data Buffers
  - N * (N-1) Buffers Required
    - N = Number of Processors Assumes All Shared

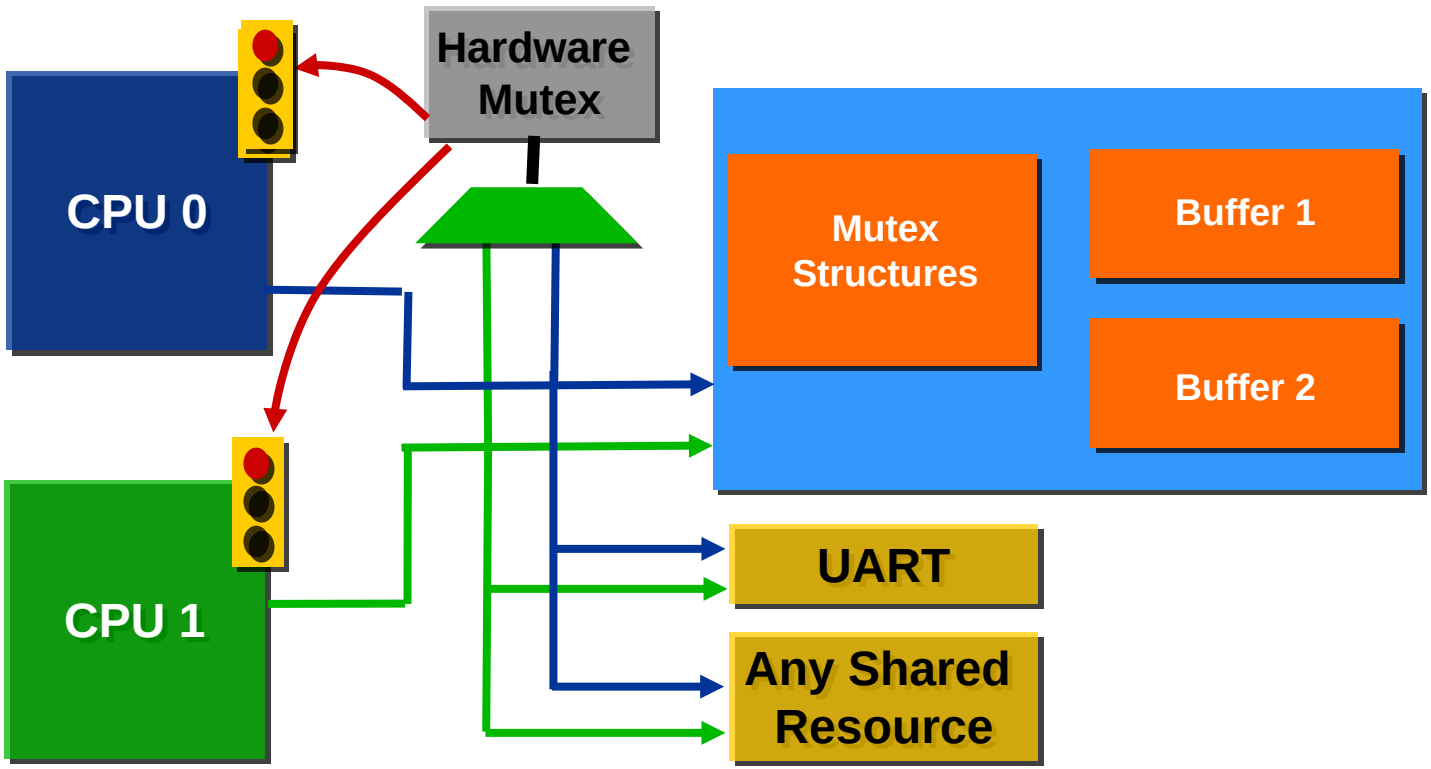


Mailbox in SOPC Builder

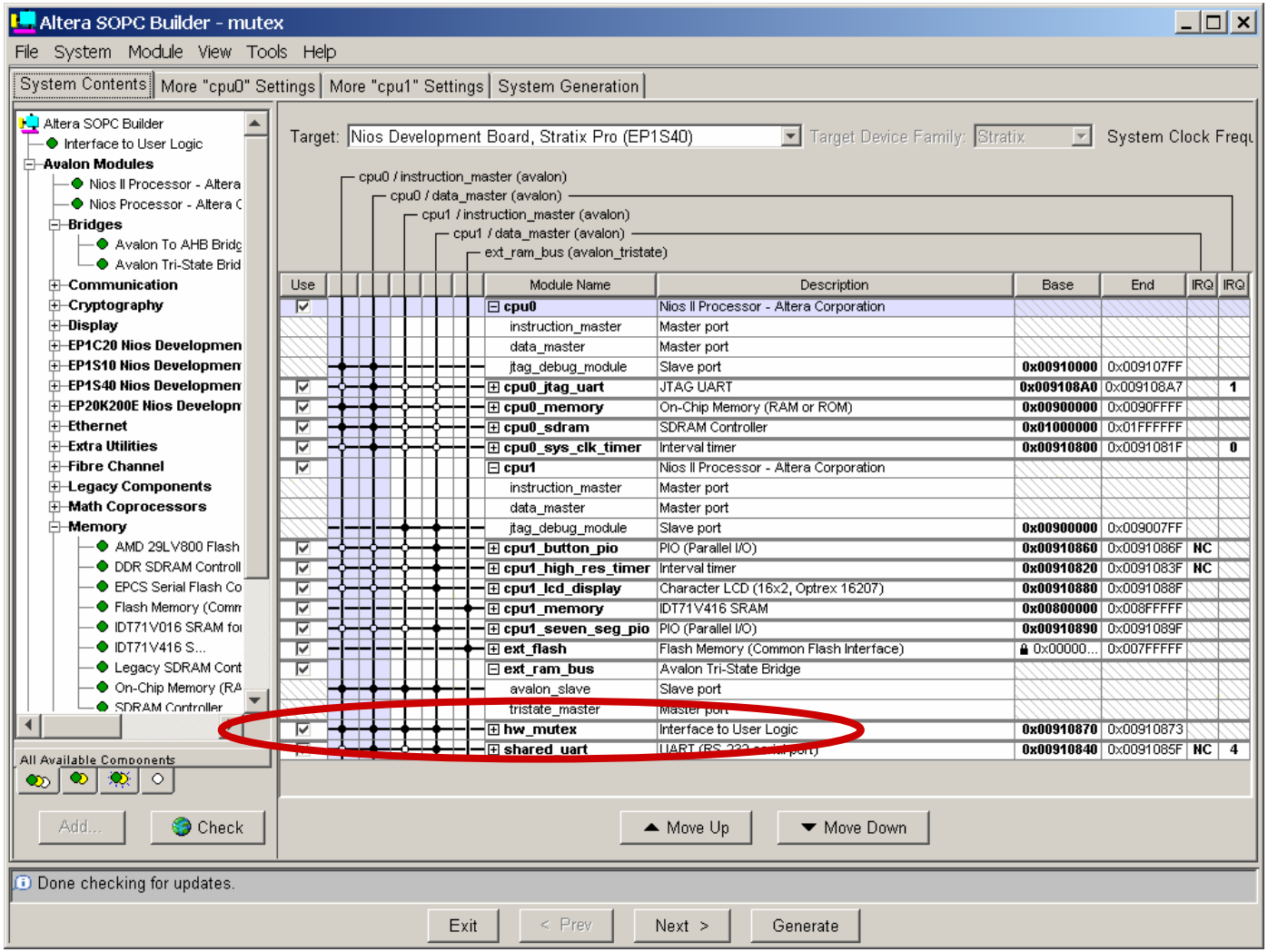


Sharing Resources – Mutex

- Controls Access to Shared Memory & Resources



Mutex System in SOPC Builder



Software Development Challenges

- Software Development
- Maintaining Hardware/Software Coherency
- Partitioning Software
  - Sorry, That's Your Job

Nios II Integrated Development Environment (IDE)*

- Features
  - Project Management
  - Editor & Compile
  - Debugger
  - Flash Programmer
- Single JTAG Communication Link
  - HW/SW Download
  - HW/SW Debug
    - SignalTap® II Embedded Logic Analyzer
    - JTAG Debug Module
  - JTAG UART
- Advanced Hardware Debug Features
  - Hardware Break Points, Data Triggers, Trace

SOPC WORLD 2004

* Based on Eclipse Project Integrated Development Environment (IDE)

ALTERA.

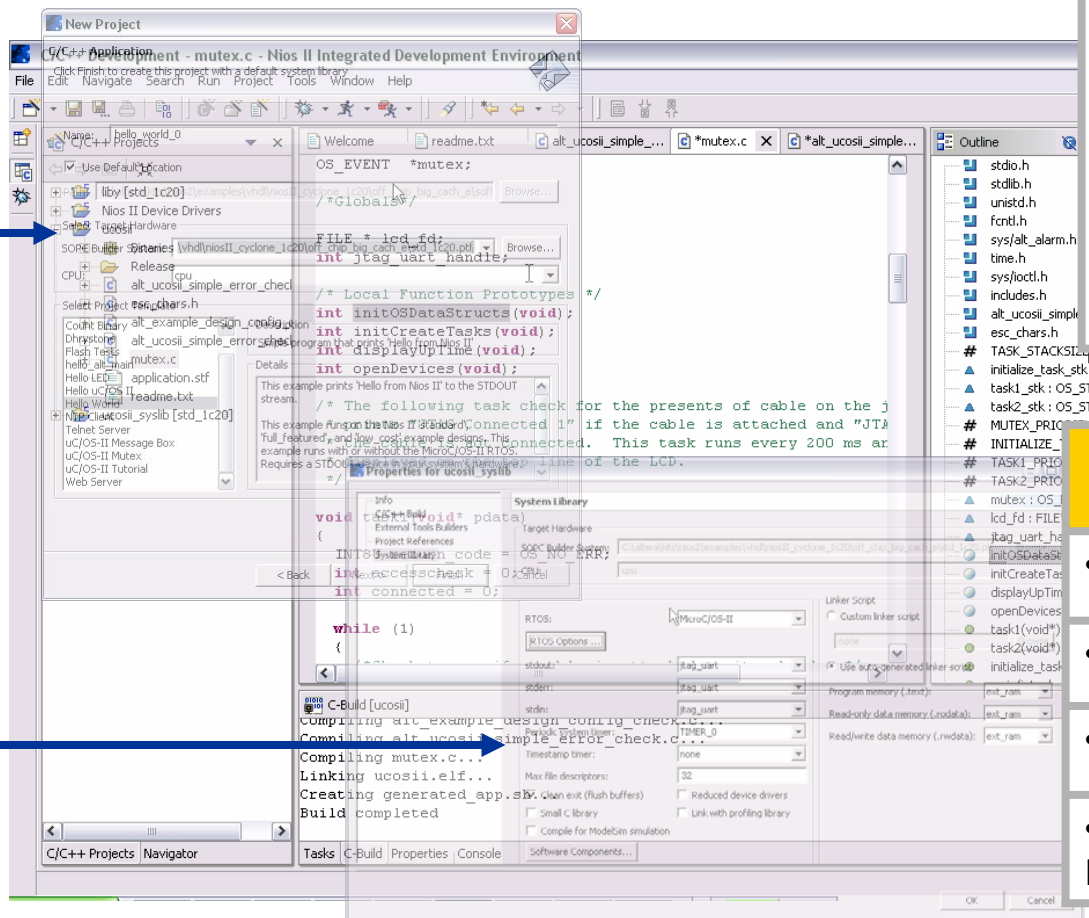
Nios II IDE: Project Management

Software Templates

- Provide Easy-to-Use Examples
- Users May Add Their Own Templates

Software Components

- System Library Customized to Hardware
- Painless Configuration



Nios II IDE

Features:

- Project Manager
- Editor & Compiler
- Debugger
- Flash Programmer

Included Software Components

- LWIP TCP/IP Stack
- MicroC/OS-II RTOS*
- Altera ZIP FS
- Nios II Run-Time Library

Integrated Software Project Management

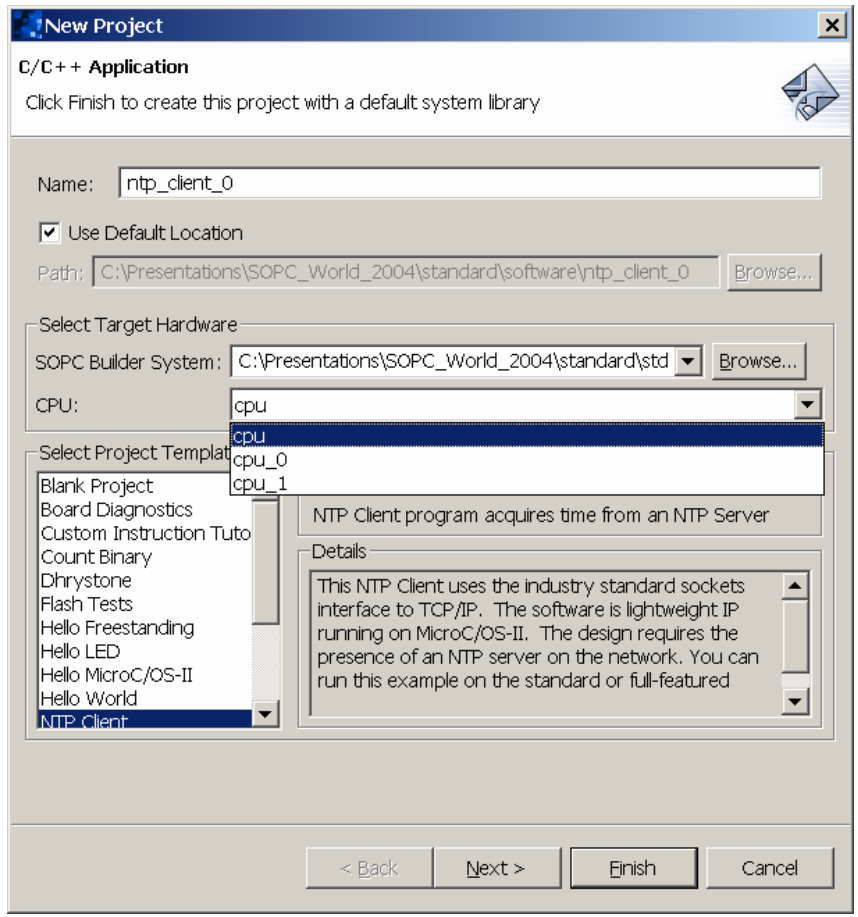
* MicroC/OS-II requires a separate license for product deployment!

SOPC WORLD 2004

ALTERA

Nios II IDE – Multi-Processors

- Multiple Processors Support
- System Library Customized to the SOPC Builder Project
  - HW/SW Coherency
- Assign Regions of a Shared Memory to Individual Processors
  - Linker Script



Nios II Hardware Abstraction Layer

HAL Mailbox Routines

- **inter_processor_mbox_init()**
  - Initialize a Mmailbox
- **inter_processor_mbox_get()**
  - Copy a Message from Mailbox. Block Until Requested bBtes are Available.
- **inter_processor_mbox_tryget()**
  - Copy a Message from Mailbox. Fail if Requested Bytes are Not Available.
- **inter_processor_mbox_peek_item()**
  - Copy a Message from Mailbox. Fail if Requested Bytes are Not Available. Do not Remove the Message from the Mailbox.
- **inter_processor_mbox_put()**
  - Copy a Message to Mailbox. Block until the Message Can Be Placed in the Mailbox.
- **inter_processor_mbox_tryput()**
  - Copy a Message to Mailbox. Fail if the Message Will Not Fit in the Mailbox.
- **inter_processor_mbox_peek()**
  - Retrieve the Number of Bytes Available for Getting from the Mailbox.

HAL Mutex Routines

- **inter_processor_mutex_init()**
  - Initialize a Mutex, & Associate it with the Supplied Hardware Component.
- **inter_processor_mutex_destroy()**
  - Destroy a Mutex, Disassociating it with the Supplied Hardware Component.
- **inter_processor_mutex_lock()**
  - Acquire the Lock for the *Mutex.* Block until The Lock Can Be Acquired.
- **inter_processor_mutex_trylock()**
  - Acquire the Lock for the *Mutex.* Return if the *Mutex* is Unavailable.
- **inter_processor_mutex_unlock()**
  - Release the Lock for the *Mutex.*

Nios II IDE: Debugger

Basic Debug

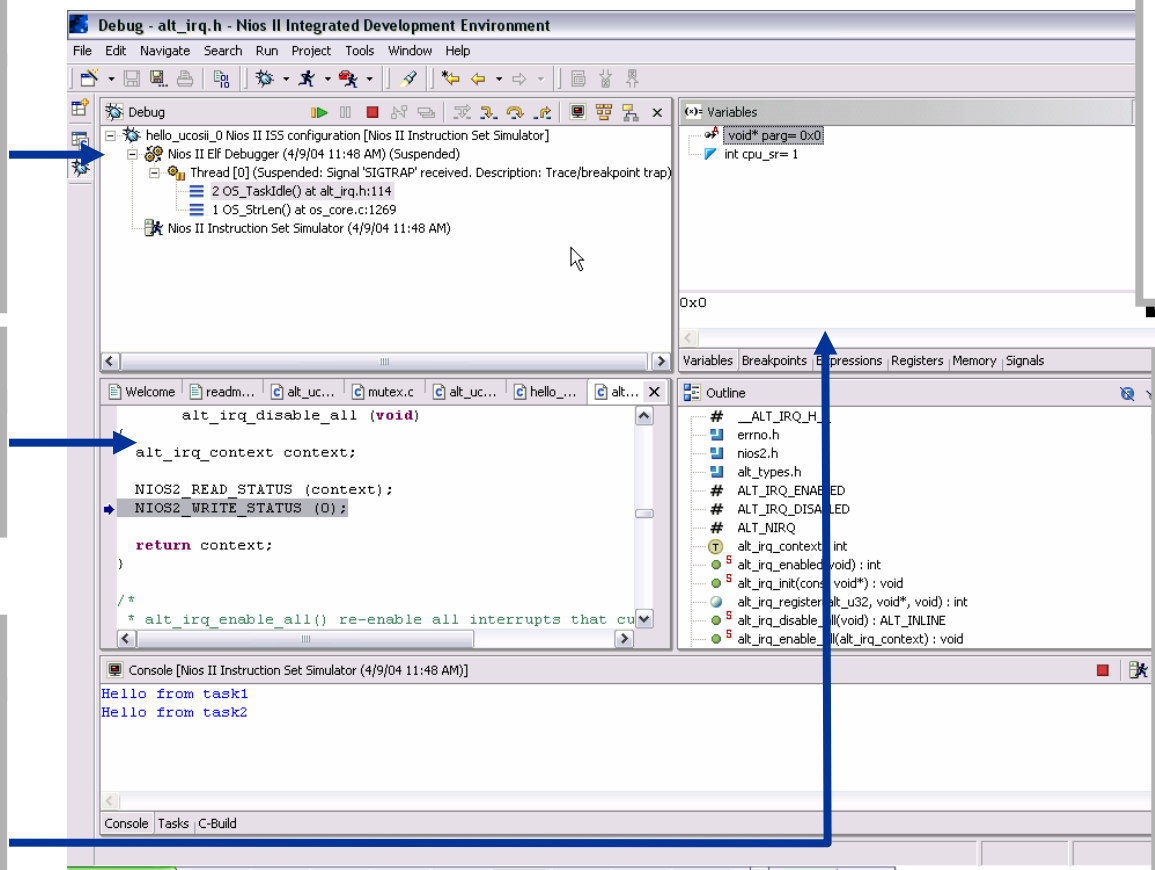
- Run Controls
- Stack View
- Active Debug Sessions

Source View

- Breakpoints
- Code Position

Memory View

- Variables
- Registers
- Signals



Nios II IDE

Features:

- Project Manager
- Editor & Compiler
- **Debugger**
- Flash Programmer

Debug Scope

Debug Targets:

- Hardware (JTAG)
- Instruction Set Simulator
- Logic Simulator (ModelSim)

Debug Scope:

- HW/SW breakpoints
- H/W Data Triggers
- Watchpoints
- Instruction Trace

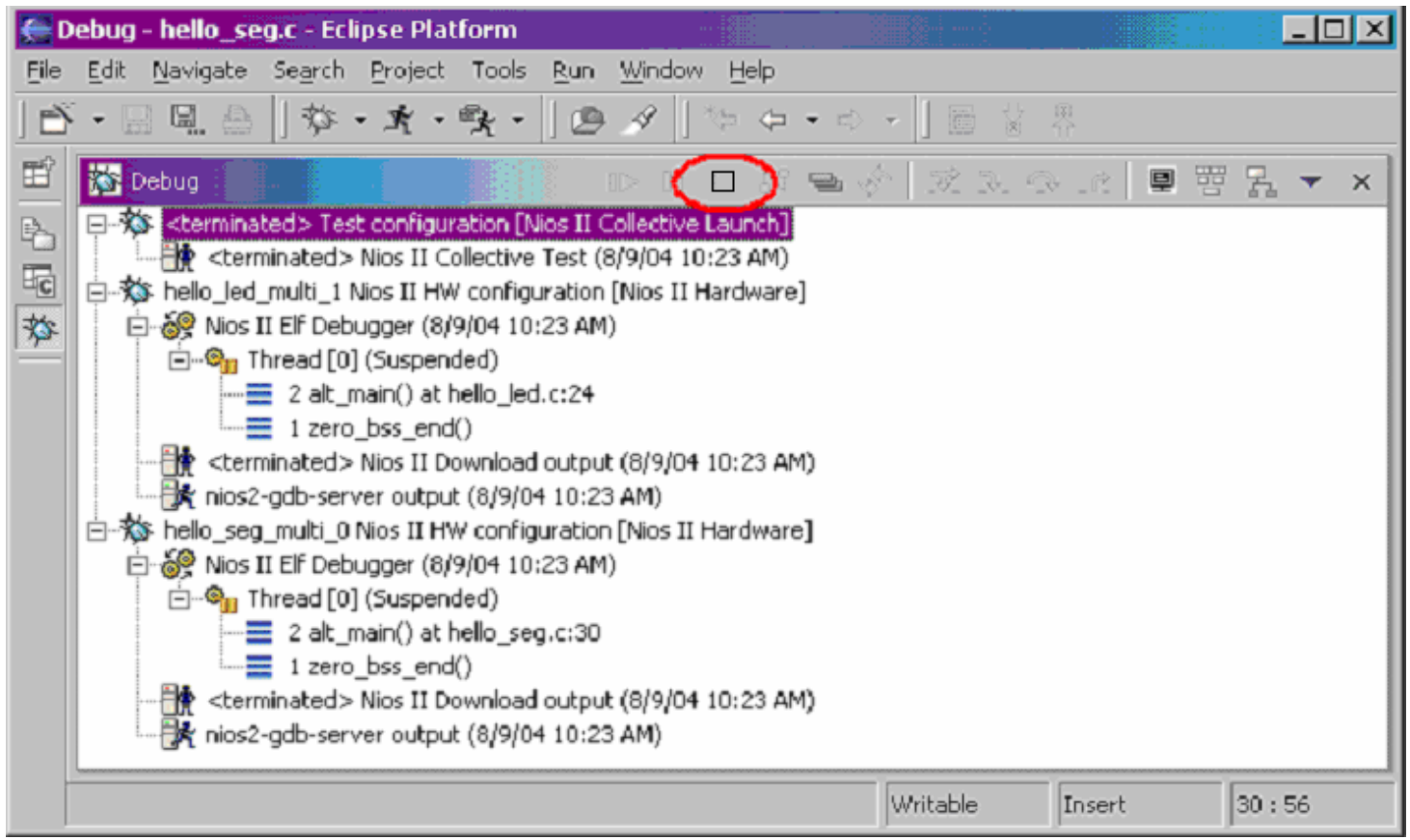
Integrated, Broad Capability Debug

SOPC WORLD 2004

ALTERA

Debug – Multi-Processor Support

- Collective Launch & Terminate



Multi-Processor Systems Using FPGAs

- Multi-Processors
  - Address Increasing Application Requirements
  - Expand Product Functionality
  - Simplify Software Development
- FPGAs, Nios II, SOPC Builder
  - The *Perfect-Fit* Solutions

Additional Resources

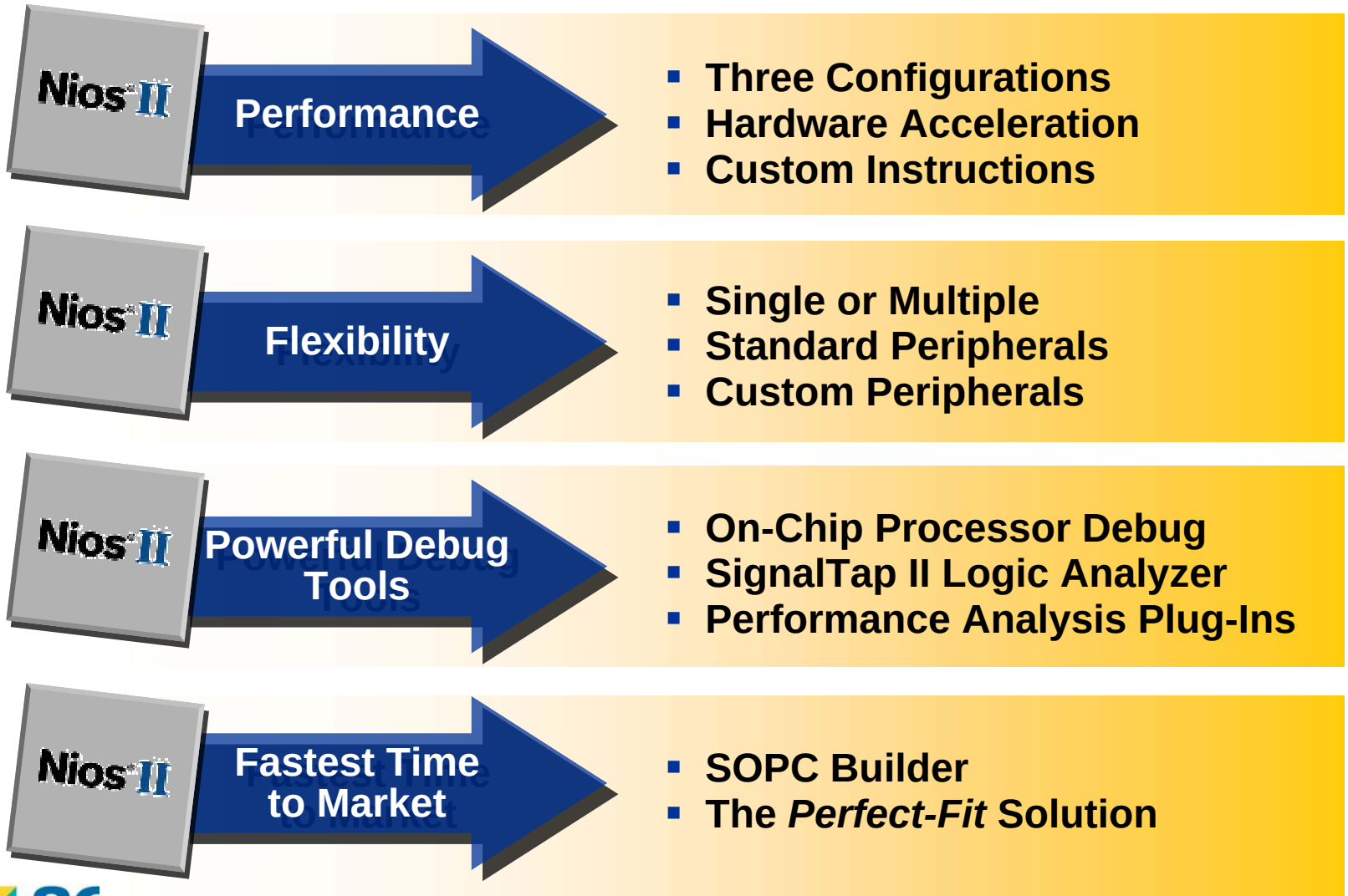
- Nios II & IP MegaCore® Functions
  - Nios II Embedded Processor Evaluation Edition
  - MegaCore IP Library CD
- SOPC Builder
  - Included in All Quartus® II Products
- www.altera.com Literature Section
  - Multi-Processor Systems with Nios II Application Note
  - Nios II Handbook
  - SOPC Builder Documentation

SOPC WORLD 2004

Multi-Processor Systems in FPGAs: Implementation & Debug

ALTERA®

FPGAs for Multi-Processor Designs



SOPC WORLD 2004

ALTERA

SOPC WORLD 2004

Thank You !

ALTERA®