

nbit_adder: adderx

GENERIC MAP (x => 0)

PORT MAP (AddSubR_n => AddSubR_n)

multiplexer: mux2to3

GENERIC MAP (x => 0)

PORT MAP (A_in => A_in, Z => Z)

AddSubR_n <= (OTHERS => AddSubR_n)

M <= XOR AddSubR_n

M <= XOR G(n-1), XOR G(n-2), XOR M(n-1);

Using FPGAs With Multiple Cores

Agenda

- Background: FPGA-based computing
- FPGA computing topologies and scalability
- Hardware acceleration
- Effect of multiple cores on productivity
- Conclusion
- Q&A


```
nbit_adder: adder8;
```

```
GENERIC MAP (x => 8)
```

```
PORT MAP (AddSubR_n => 0, M_n => 0);
```

```
multiplexer: mux2to3
```

```
GENERIC MAP (x => 3)
```

```
PORT MAP (A_in_1 => 0, S_in_1 =>
```

```
AddSubR_n <= (OTHERS => AddSubR_n
```

```
M_n <= 0); XOR AddSubR_n
```

```
XOR G_n <= XOR G_n <= XOR M(n-1);
```

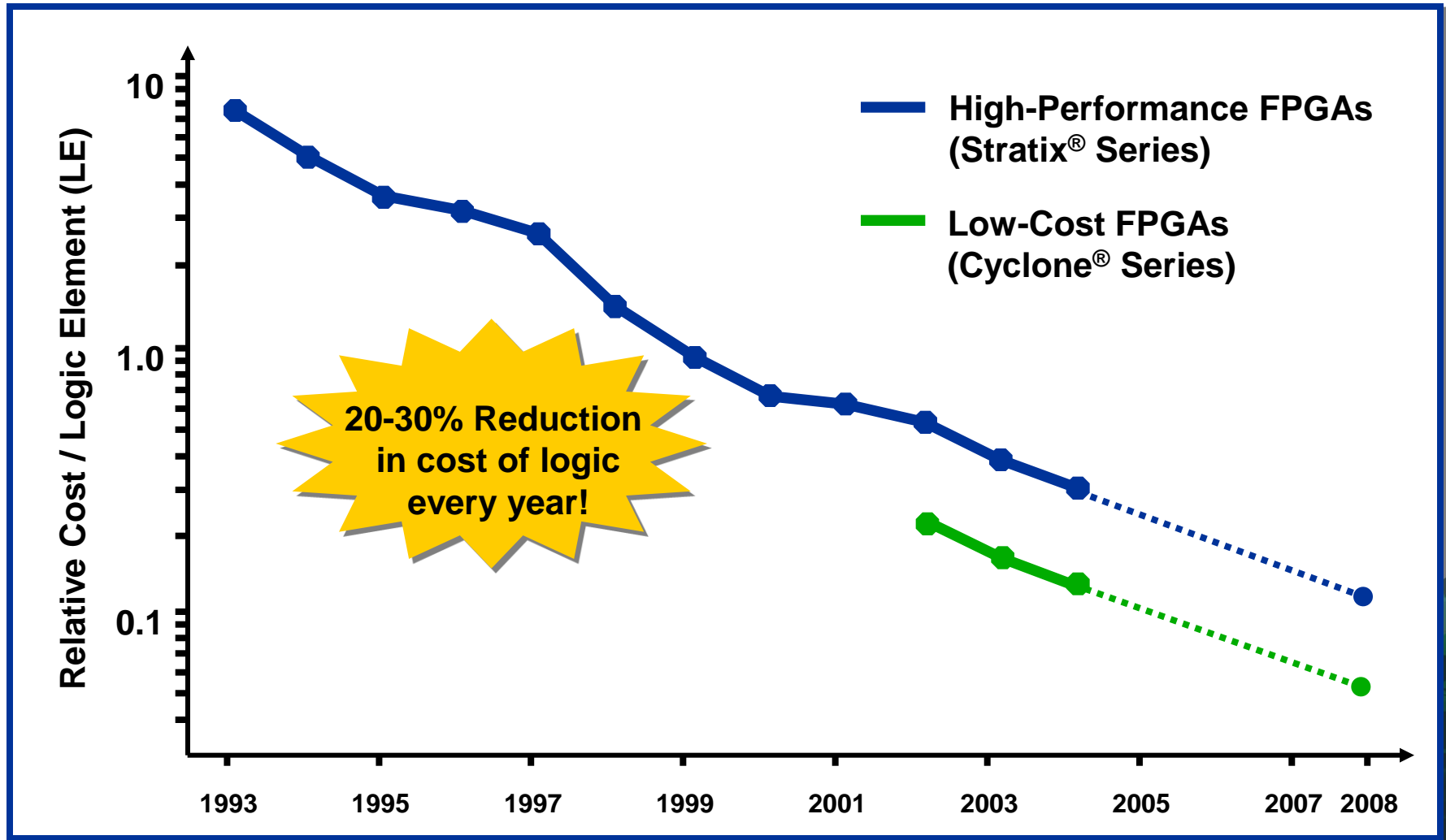
Background: FPGA-Based Computing

Today's FPGA Devices Meet Advanced System Requirements

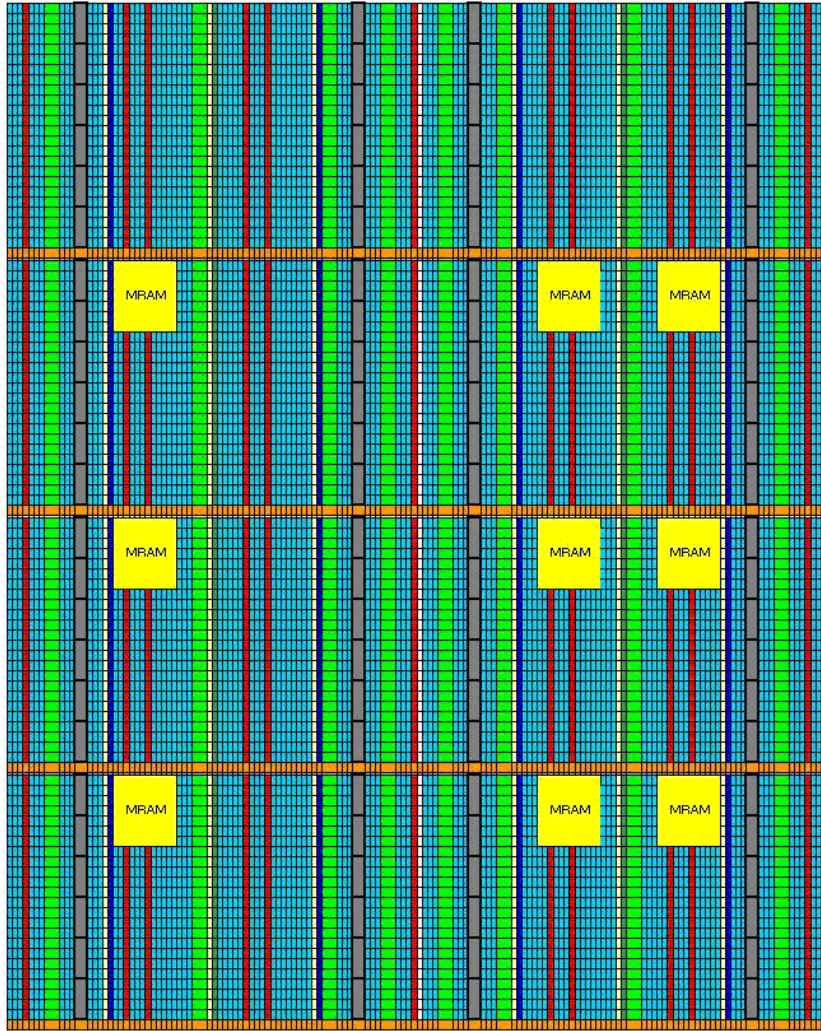
- Wide range of fast I/O
- Substantial embedded memory
- High-performance digital signal processing (DSP) blocks
- Abundant logic
- Low-cost FPGA and structured ASIC families



FPGA-Based System Economics



Modern FPGAs: Platforms for High-Performance Parallel Architectures



- Example: Altera's Stratix II Family
 - First 90-nm high-end FPGA
 - TSMC Fab
 - Shipped in June 2004
- Rich interconnect fabric
- Up to 180K LEs
- Up to 9 million memory bits
- Up to 96 flexible DSP blocks
- Up to 1,170 high-speed I/O pins
- Many clock trees and phase-locked loops (PLLs)

Soft-Core CPUs Are Affordable

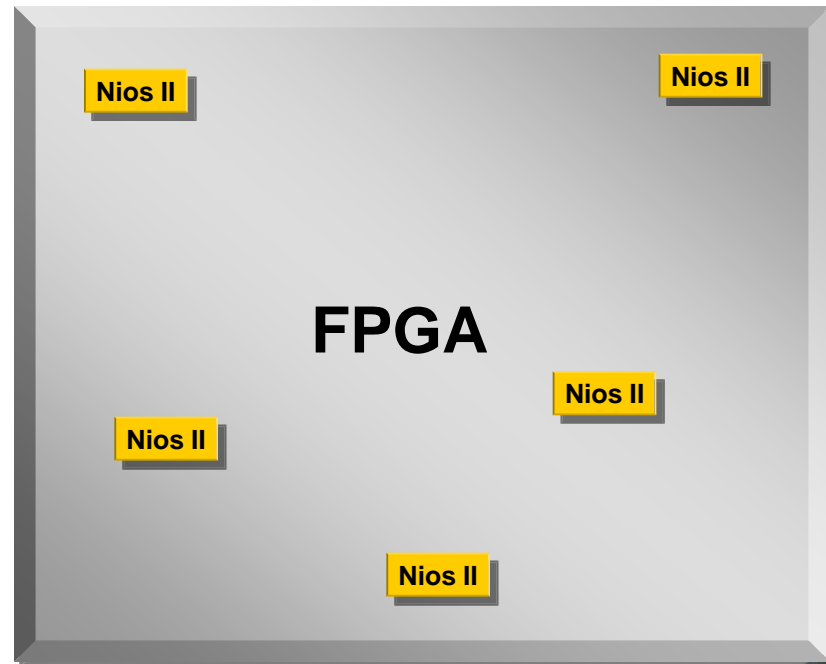
Small Cyclone II
4,600 Look-up Tables (LUTs)



600 LEs
13% of FPGA
Nios II/e "Economy"

35¢ in
Lowest-Cost
FPGA

Largest Stratix II
180,000 LUTs



1800 LEs, 1% of FPGA
Nios II/f "Fast"

FPGA = Fast Development

- Fast time to production
 - Prototype and production boards can be the same
 - Can accommodate late changes
 - Use/reuse intellectual property (IP)
- Fast design cycle
 - Develop → synthesize → test
- System-level design tools
 - Integration of IP and in-house designs
- Powerful system debug tools
 - Software and hardware



IP

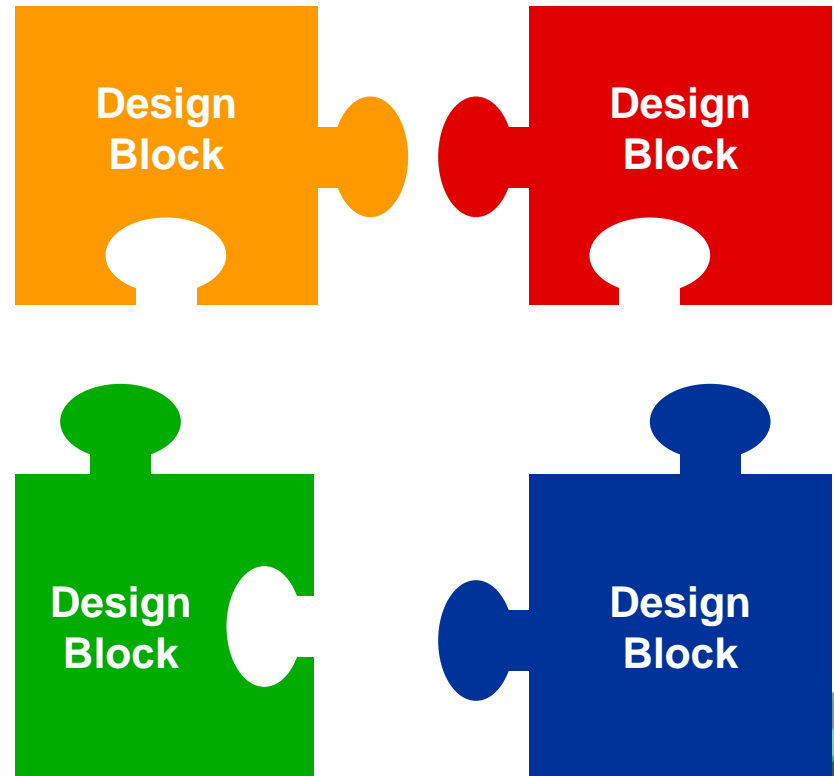


QUARTUS II



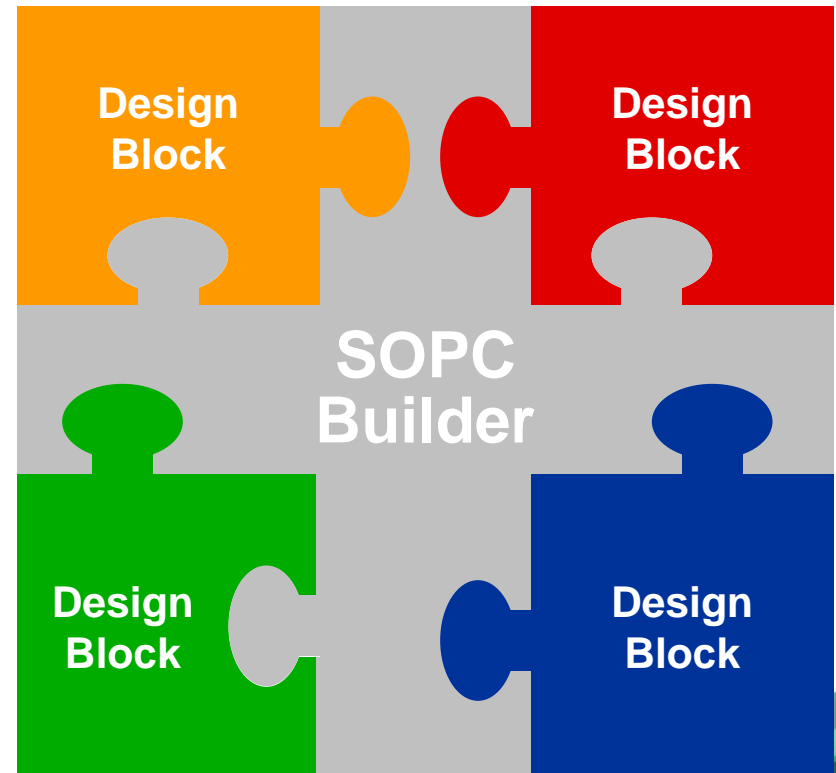
Using IP to Increase Productivity

- Not all blocks are designed with the same interface
- Design for reuse can lead to greater time and cost
- Blocks need integration to create system



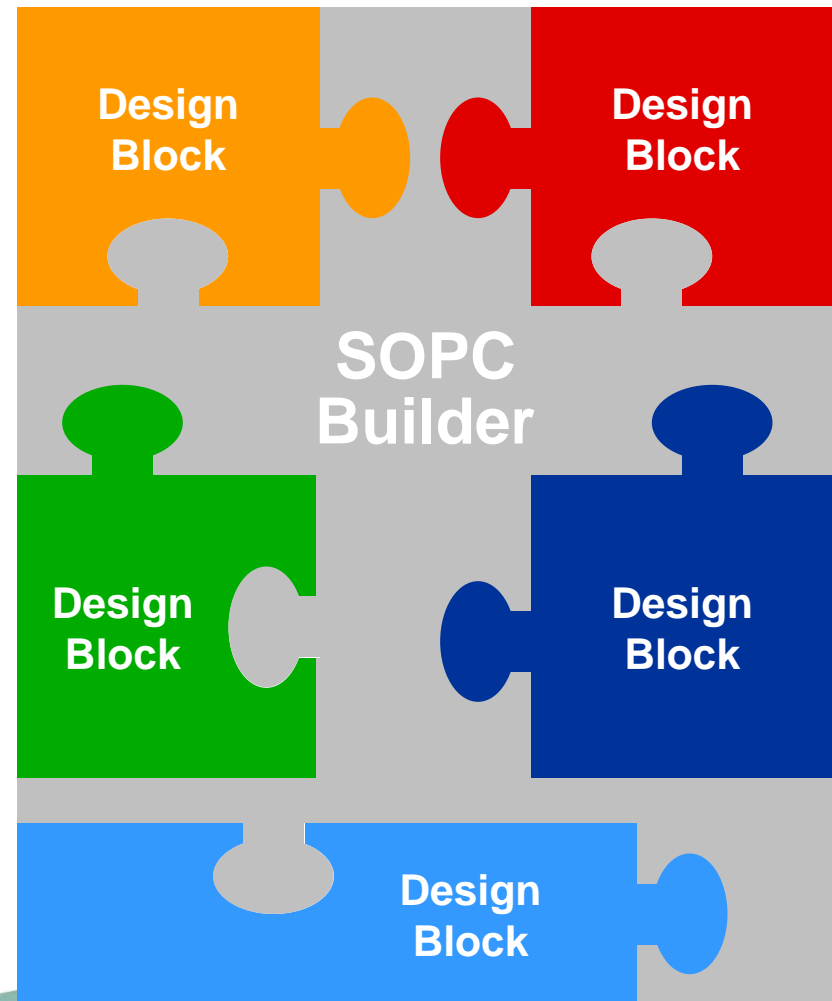
SOPC Builder - Automated Integration

- Generates interconnect between IP blocks
 - Supports third party and in-house IP
- Automatically resolves interface issues
 - Clock domains
 - Bit widths
 - Arbitration
 - Other issues

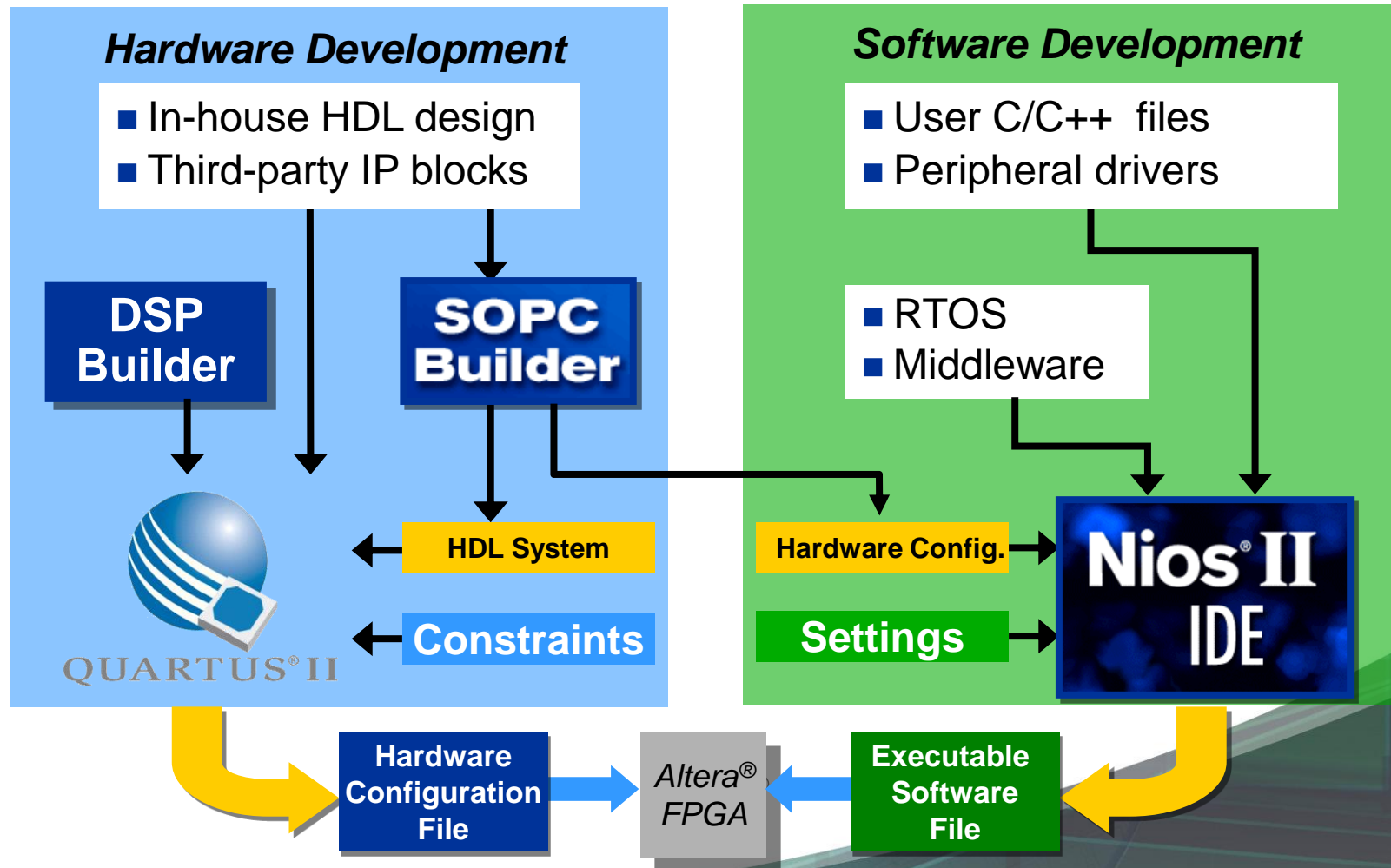


Updating the System

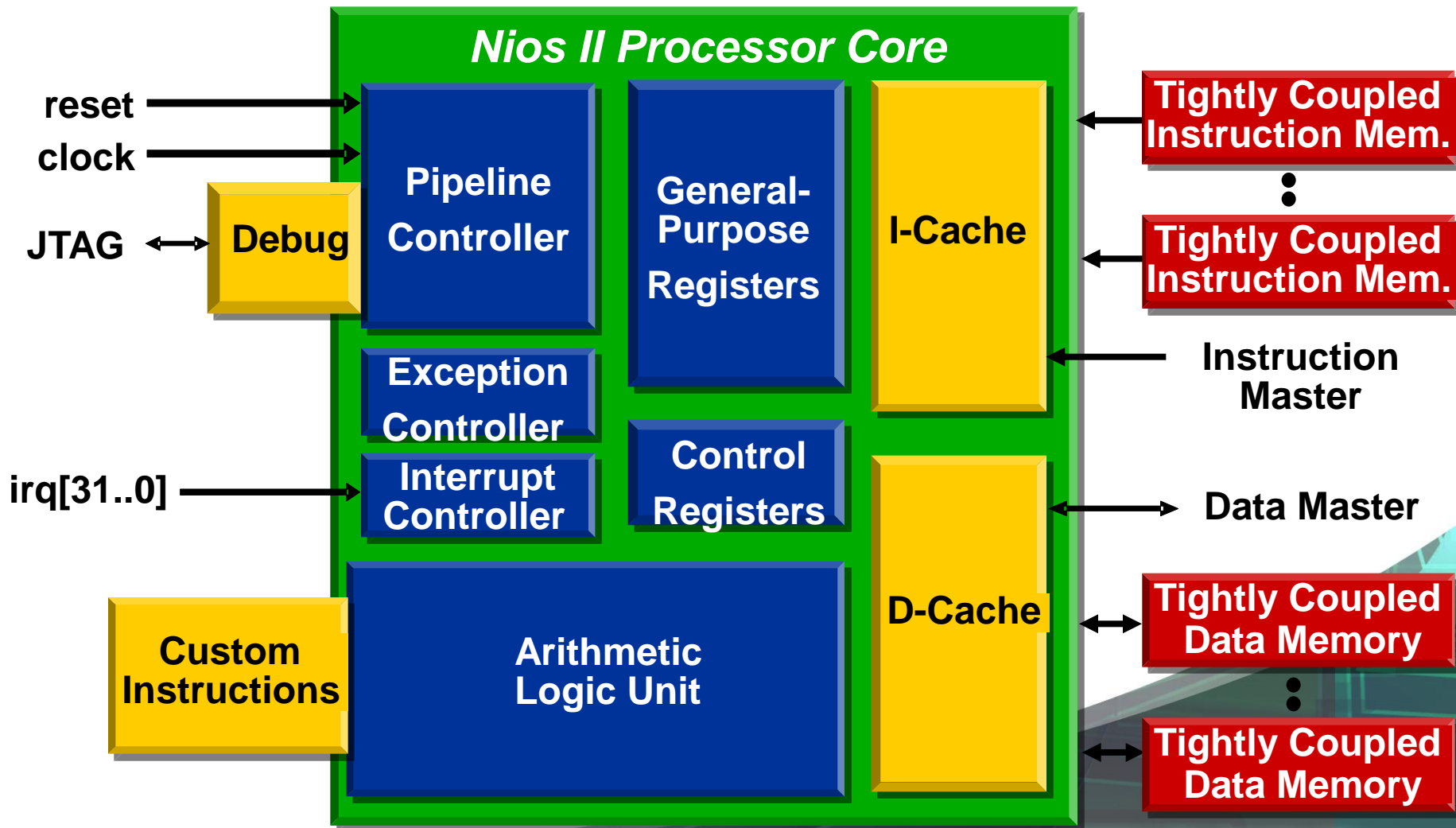
- SOPC Builder systems can be easily extended
- No need to “hack” into existing system
- Minimal disruption to the rest of the system



Embedded System Design Flow

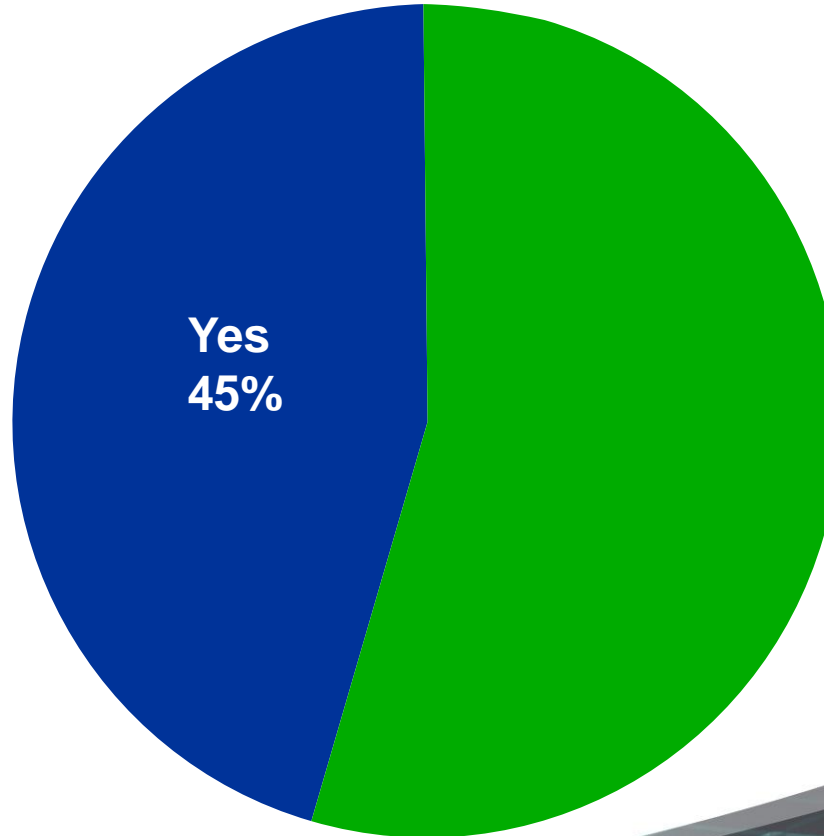


FPGA-Based Soft CPU: Nios II



FPGA-Based Soft CPU Usage Trend

Percent of EE Times embedded systems designers who say they “will likely use a processor inside an FPGA”



Source: CMP 2005 Embedded Market Study

nbit_adder: adder

GENERIC MAP (x => 8)

PORT MAP (AddSubR_n => 0, M_n => 0)

multiplexer: mux2to1

GENERIC MAP (x => 8)

PORT MAP (A_in => Z, S_in => G)

AddSubR_n <= (OTHERS => AddSubR_n)

M_n <= (OTHERS => XOR AddSubR_n)

M_n <= XOR G_n, XOR M_n, XOR M(n-1);

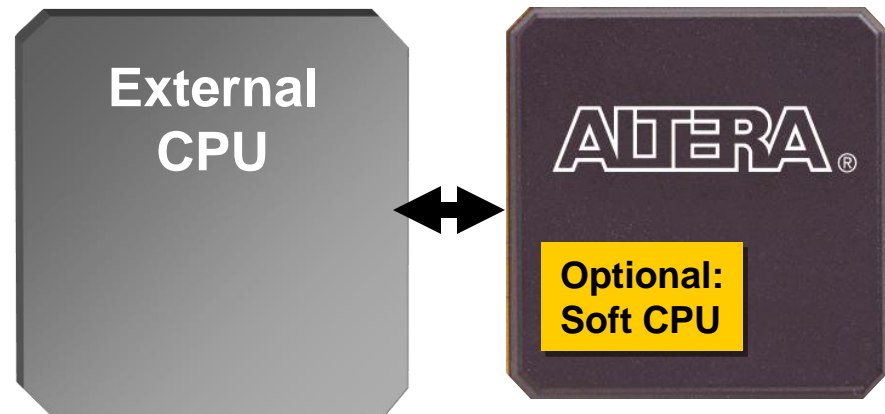
FPGA Computing Topologies

Two FPGA Computing Topologies

Standalone Computing



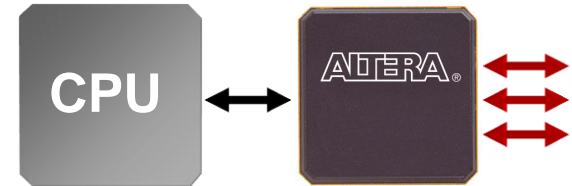
Companion Computing



Enhancing Your System With FPGAs

■ Peripheral expansion

- Extends life of CPU
- Enhances flexibility
- Avoids costs and risks of using new part

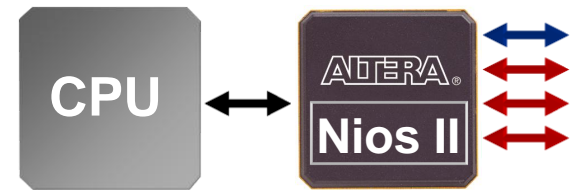
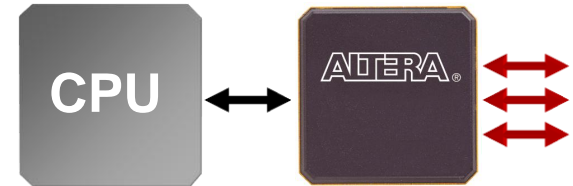


Enhancing Your System With FPGAs

■ Peripheral expansion

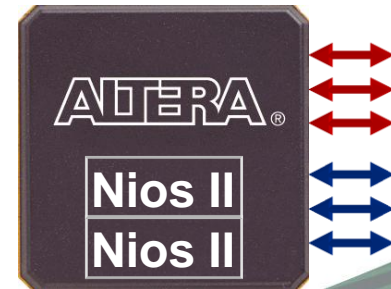
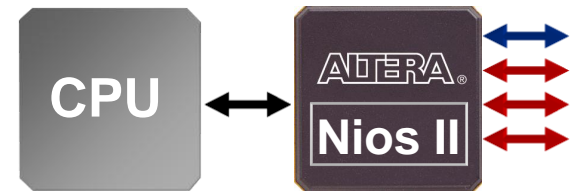
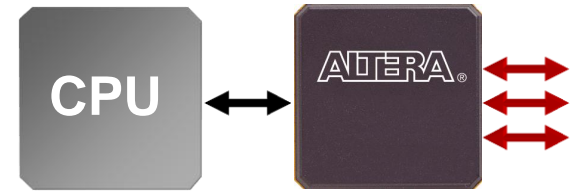
■ CPU offload

- No increase in clock speed
- Preserves real-time set up
- Low power consumption
- No upgraded components required
- Increases flexibility



Enhancing Your System With FPGAs

- Peripheral expansion
- CPU offload
- Full system integration
 - Very flexible
 - Fast development
 - No obsolescence issues
 - High performance/low cost

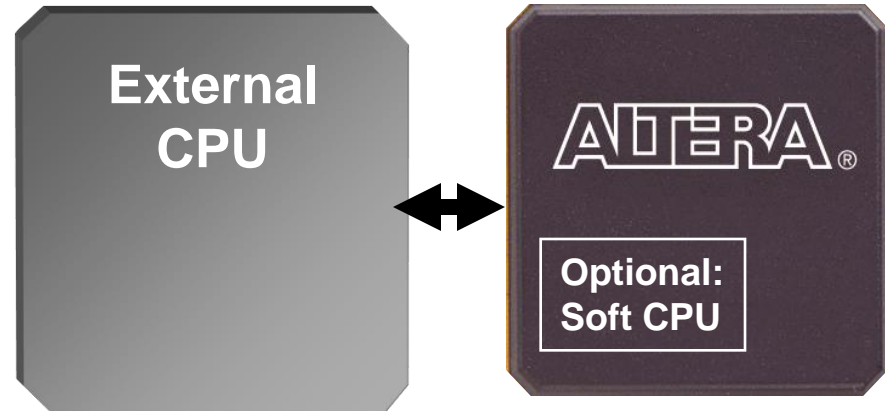


Multiple-Core Scalability

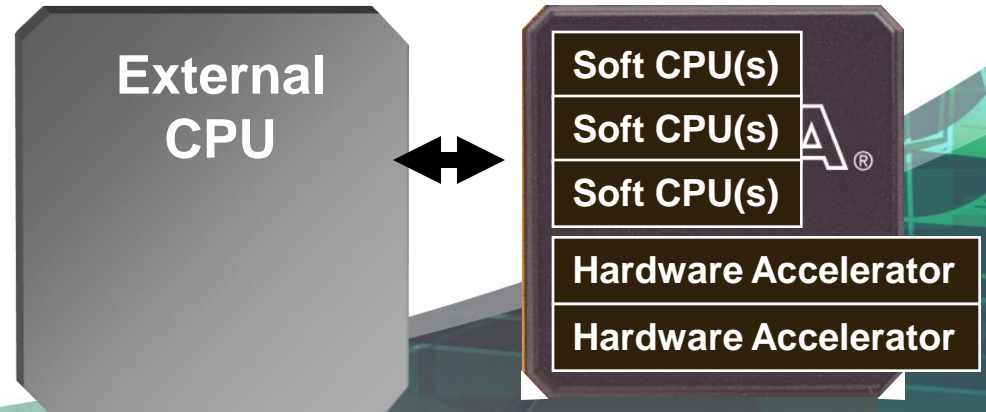
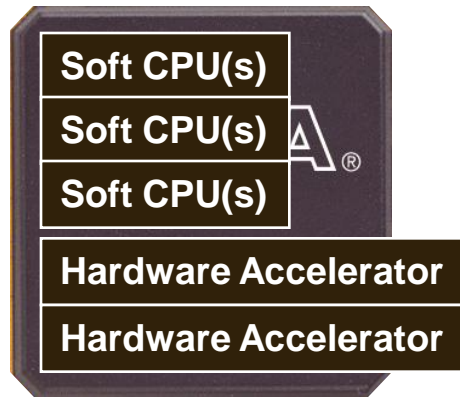
Standalone Computing



Companion Computing

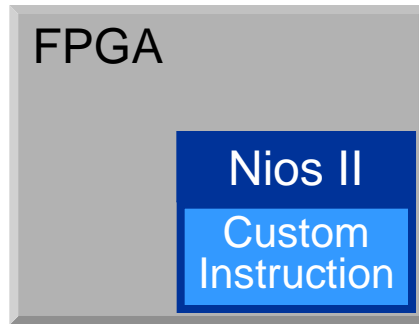


Add Compute Power As Needed



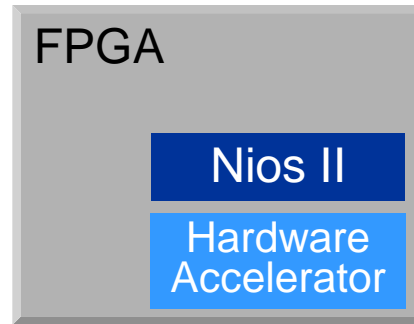
Three Ways to Scale Performance

Custom Instructions



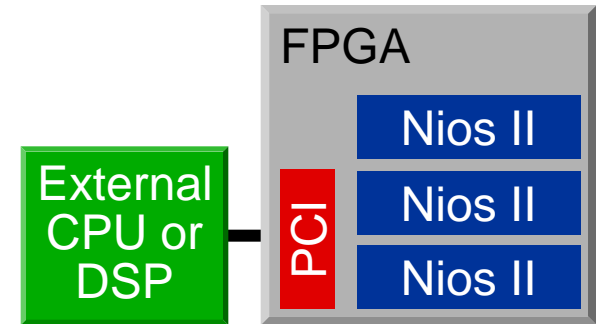
- Accelerate CPU processing performance with application-specific hardware

Hardware Accelerators



- Add external coprocessing hardware to accelerate data functions

Multiprocessor System



- Add more processors (internal and/or external) to increase processing power

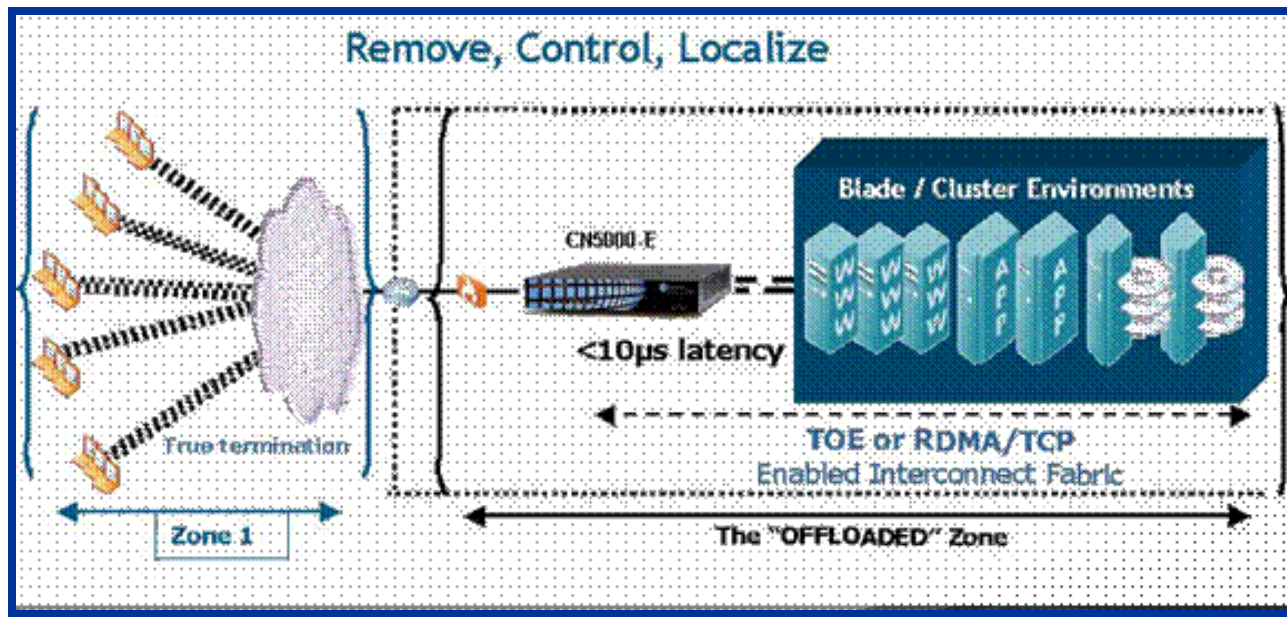
Multi-Channel Processing

Crescendo *Maestro* CN-5000E

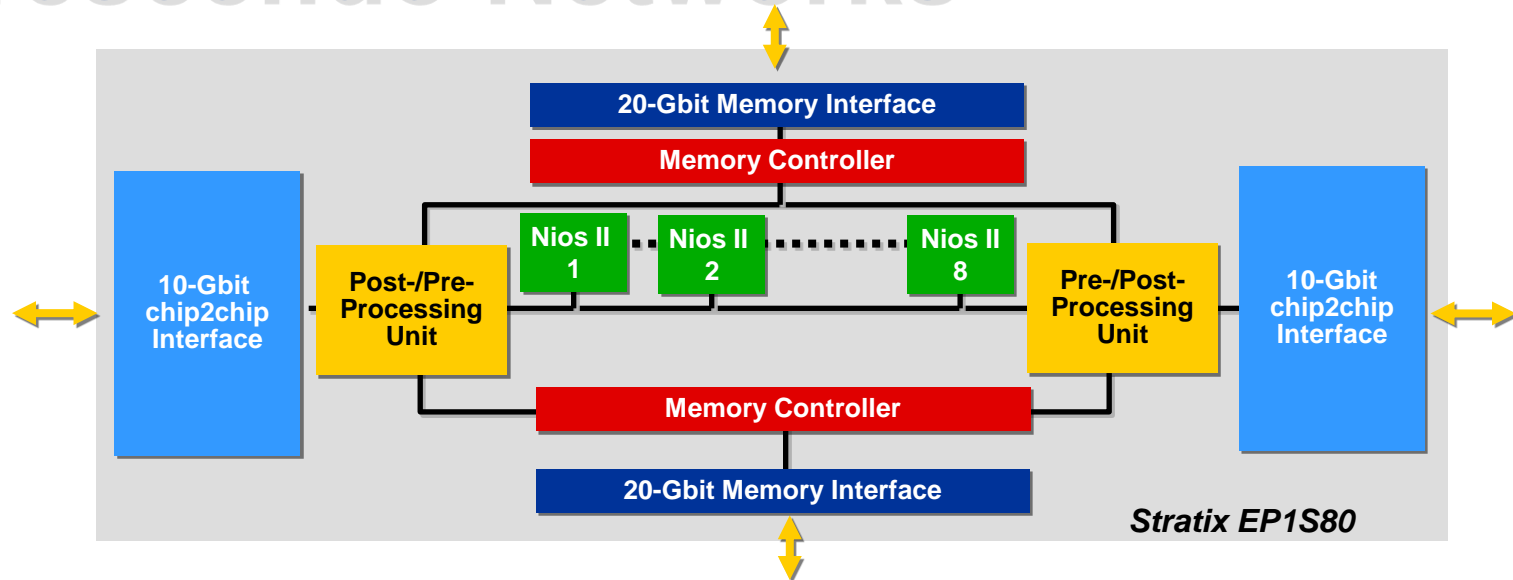


Application:

Application Acceleration Platform



Crescendo Networks



■ Eight Nios II Processors

- TCP/IP offload engine (TOE)
- Complex memory management
- Header parsing

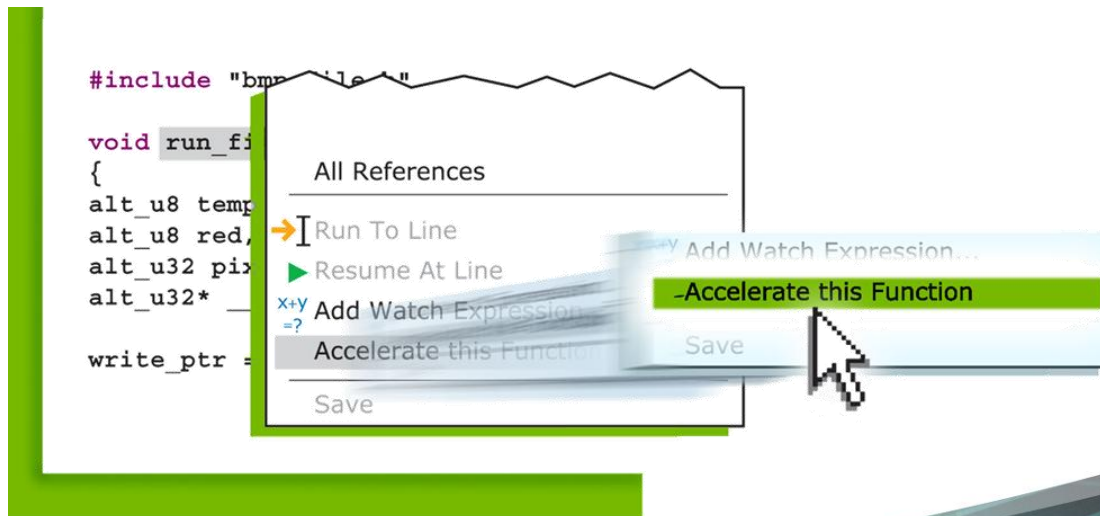
■ Comparisons

- ASIC
 - Lower development cost
 - Faster time to market
- ASSP
 - More flexible
- Off-the-shelf processor
 - Able to support multi-Gbit bandwidth

Hardware Acceleration

Advanced C-to-Hardware Acceleration (C2H) Productivity Tool

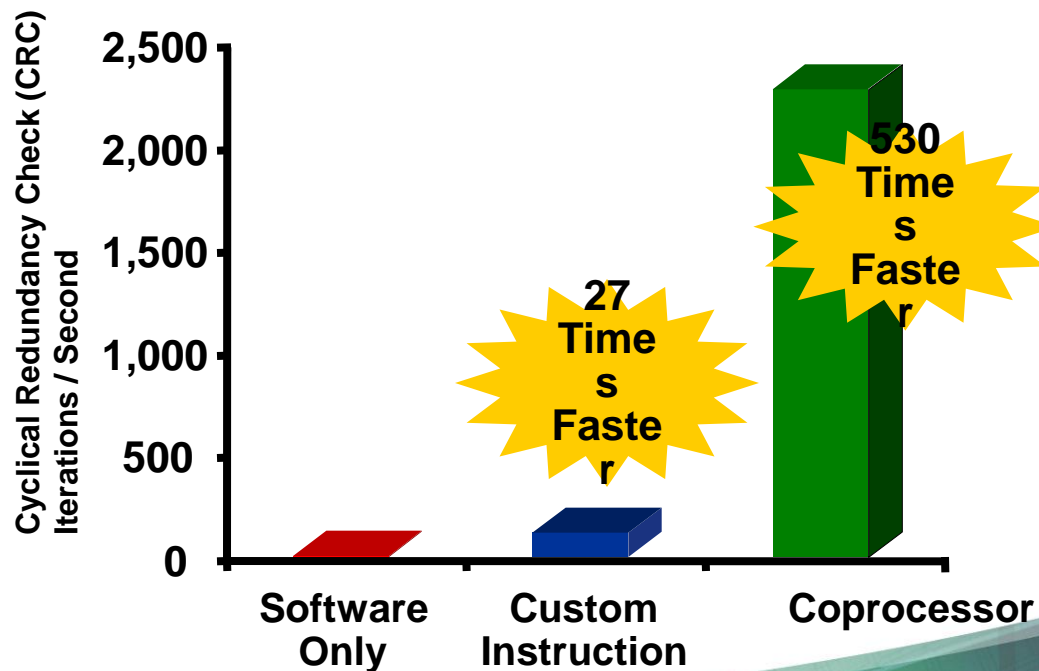
- Automates creation and integration of hardware accelerators
- Intuitive user interface streamlines C acceleration
- Uses familiar Nios II Integrated Development Environment (IDE)
- Support for standard ANSI C language



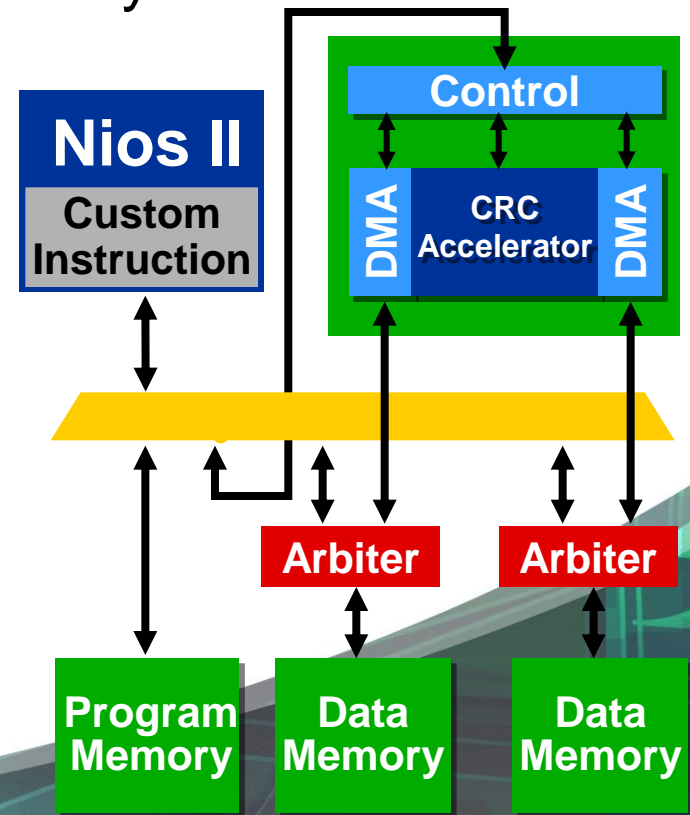
**Right-Click to
*Accelerate***

Add Hardware Accelerator

- CPU starts/stops accelerator
- Accelerator fetches data and stores results
- CPU runs application code concurrently
- Ideal for block data operations

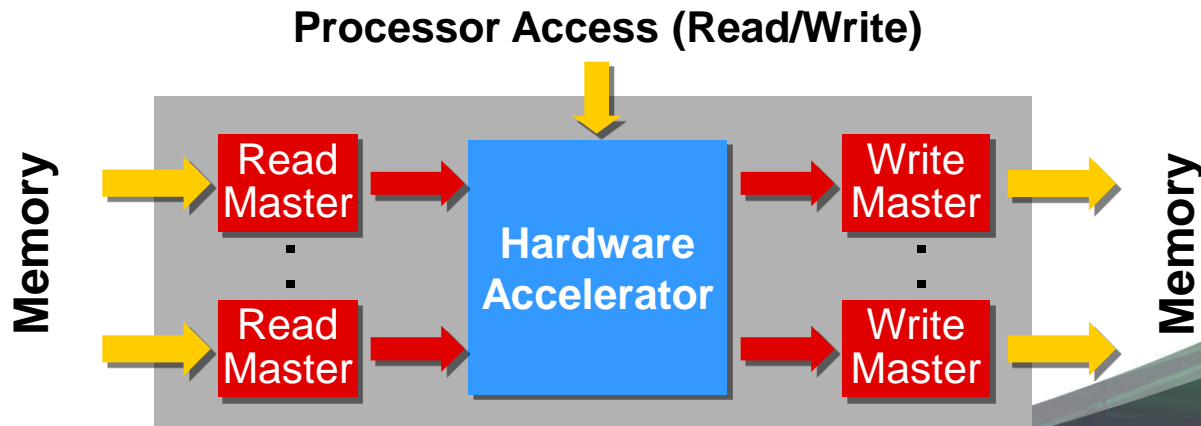


* Processor running at 100 MHz



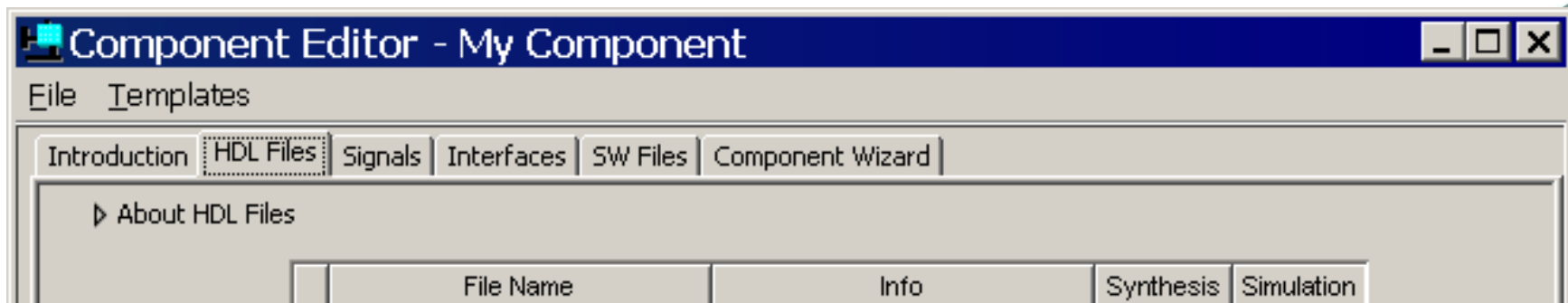
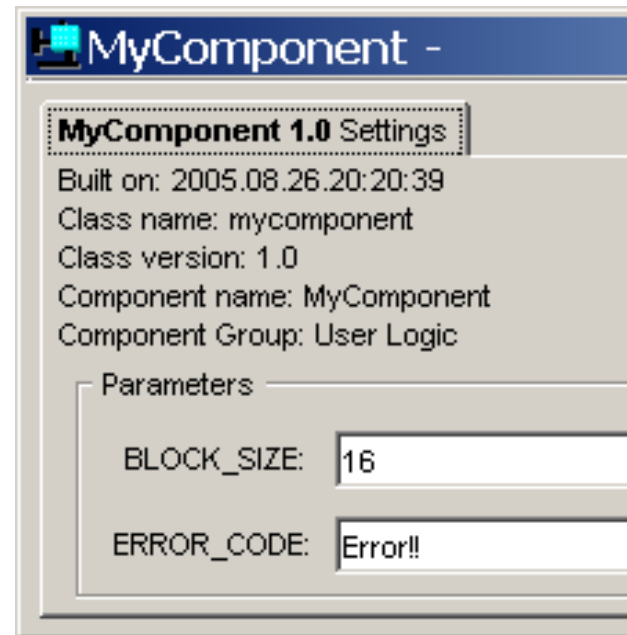
Anatomy of a Hardware Accelerator

- Multiple Avalon[®] master interfaces to memory
 - Accelerator pipelined to match memory latency
 - Can sustain extremely high throughput
- Slave interface for CPU control
 - Memory mapped registers such as configuration, start, stop, status, etc.
- Software wrapper function
 - Controls hardware via slave interface



Adding SOPC Builder Components

- Import hardware design files
- Map signals to Avalon switch fabric
- Set additional timing parameters
- Add software files (optional)
- Add component information
- Add to SOPC Builder library



Performance Comparison

■ Mandlebrot algorithm

- Pentium M running 1,600 MHz, entirely out of cache (best case)
- Cyclone II system running Nios II processor at 75 MHz with acceleration blocks

	Floating-Point Software	Integer Software	Accelerator Hardware	Direct Memory Access (DMA) I/O Offload Hardware
Pentium at 1,600 MHz	30 ms	110 ms	NA	NA
Pentium at 75 MHz	640 ms	2347 ms	NA	NA
Nios Processor at 75 MHz	351,789 ms	243,017 ms	211 ms	35 ms

nbit_adder: adder

GENERIC MAP (x => 8)

PORT MAP (AddSubR_n => 0, M_n => 0)

multiplexer: mux2to1

GENERIC MAP (x => 8)

PORT MAP (A_in => Z, S_in => G)

AddSubR_n <= (OTHERS => AddSubR_n)

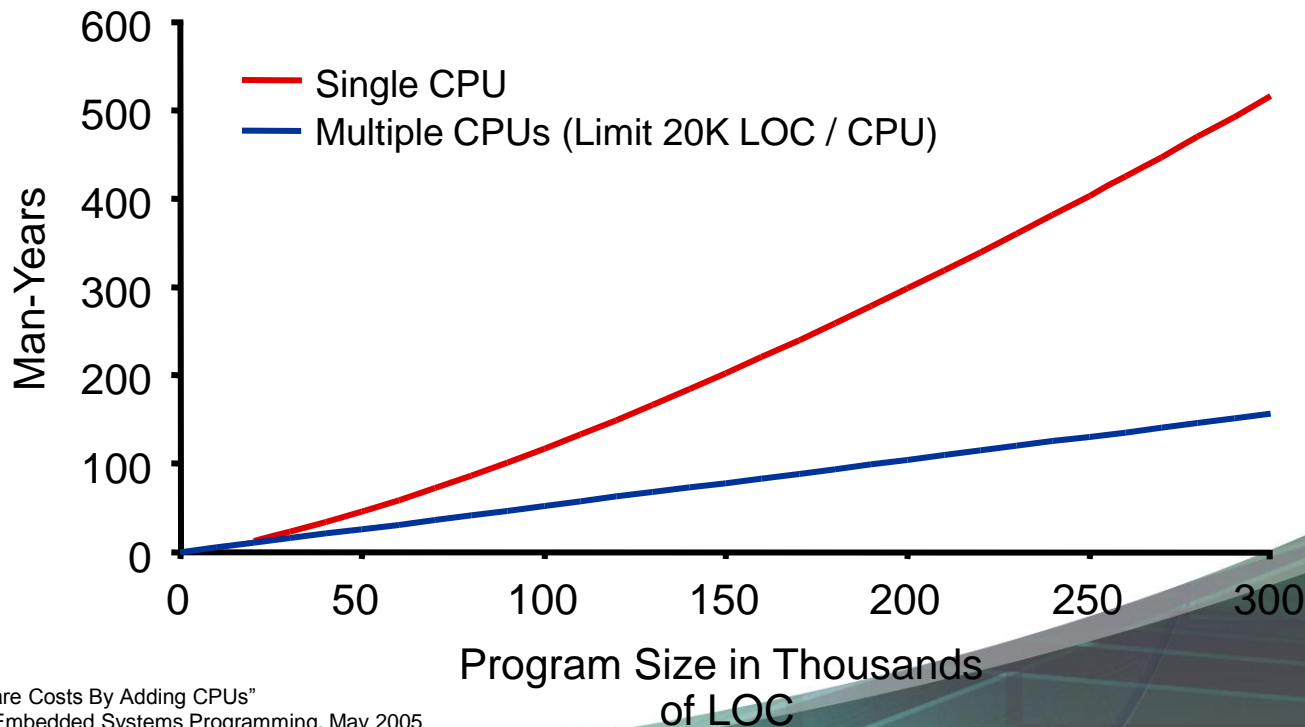
M_n <= (OTHERS => M_n)

XOR M(n-1);

Effect of Multiple Cores on Productivity

Multiple Cores Drive Team Productivity

- Each processor runs fewer lines of code
- Smaller projects are faster to develop
 - “Double the lines of code (LOC) and multiply man-months by four” *
- Improves reliability and maintainability
 - Large function error rates 2X to 6X higher than smaller routines **



* “Subtract Software Costs By Adding CPUs”

-- Jack Ganssle, Embedded Systems Programming, May 2005

** “An Empirical Study of Operating System Errors”

-- Chu, Yang, Chelf, Hallem, Proceedings of the 18th ACM Symposium on Operating System Principles, October 2001

Multiprocessor System Design Made Easy

- Building multiple-CPU systems with SOPC Builder
 - As many Nios II CPUs as you like
 - Communicate via shared memory mailbox
 - Share hardware resources with mutex core
- Software development:
 - Integrated drive electronics with multiprocessor debug support
 - Software drivers for mailbox and mutex
 - Enhanced tool support from Lauterbach
- Example designs and application notes

Nios II Multiprocessor System

Altera SOPC Builder - std_1s40

File Module System View Tools Help

System Contents | **Nios II More "cpu1" Settings** | Nios II More "cpu2" Settings | Nios II More "cpu3" Settings | Board Settings | System Generation

Target: Board: Nios Development Board, Cyclone II (EP2C35)
Device Family: Cyclone II ☐ HardCopy Compatible

Clock	Source	MHz	Pipeline
sysclk	External	50.0	<input type="checkbox"/>
click to add...			

Altera SOPC Builder
Create New Component...
Avalon Components
Nios II Processor - Altera
Bridges
Communication
JTAG UART
SPI (3 Wire Serial)
UART (RS-232 serial)
16550S Enhanced UART
CAN 2.0 Network Controller
D16550 UART with 16550S
DI2CM I2C Bus Interface
DI2CSB I2C Bus Interface
DSPI Serial Peripheral Interface
H16550S UART -- CAST, Inc.
H8250 -- CAST, Inc.
I2C Bus Controller -- CAST, Inc.
Multi Channel HDLC -- CAST, Inc.
Nios_CAN -- IFI
T1 Framer -- Adaptiv

Cryptography
Display
EP1C20 Nios Development Board
EP1S10 Nios Development Board
EP1S40 Nios Development Board
EP20K200E Nios Development Board

All Available Components
☐ ☐ ☐ ☐
Add... Check

Use	Module Name	Description	Input Clock	Base	End	IRQ	IRQ	IRQ
<input checked="" type="checkbox"/>	cpu1	Nios II Processor - Altera Corporation	sysclk					
	instruction_master	Master port						
	data_master	Master port						
	jtag_debug_module	Slave port						
<input checked="" type="checkbox"/>	cpu1_prog_mem_64_kbytes	On-Chip Memory (RAM or ROM)	sysclk	0x00840...	0x00840FFF			
<input checked="" type="checkbox"/>	cpu1_sys_timer	Interval timer	sysclk	0x00800...	0x00800FFF			
<input checked="" type="checkbox"/>	cpu2	Nios II Processor - Altera Corporation	sysclk					
	instruction_master	Master port						
	data_master	Master port						
	jtag_debug_module	Slave port						
<input checked="" type="checkbox"/>	cpu2_prog_mem_64_kbytes	On-Chip Memory (RAM or ROM)	sysclk	0x00860...	0x00860FFF			
<input checked="" type="checkbox"/>	cpu2_sys_timer	Interval timer	sysclk	0x00820...	0x0082003F			
<input checked="" type="checkbox"/>	cpu3	Nios II Processor - Altera Corporation	sysclk					
	instruction_master	Master port						
	data_master	Master port						
	jtag_debug_module	Slave port						
<input checked="" type="checkbox"/>	cpu3_prog_mem_64_kbytes	On-Chip Memory (RAM or ROM)	sysclk	0x00810...	0x00810FFF			
<input checked="" type="checkbox"/>	cpu3_sys_timer	Interval timer	sysclk	0x00850...	0x00850081F			
<input checked="" type="checkbox"/>	ext_ram_bus	Avalon Tristate Bridge	sysclk					
	avalon_slave	Slave port						
	tristate_master	Master port						
<input checked="" type="checkbox"/>	ext_flash	Flash Memory (Common Flash Interface)		0x00000...	0x007FFFFFFF			
<input checked="" type="checkbox"/>	ext_ram	IDT71V416 SRAM		0x00900...	0x009FFFFFFF			
<input checked="" type="checkbox"/>	ext_ram_mux	Mux	sysclk	0x00830...	0x00830043			

▲ Move Up ▼ Move Down

Benefits of Multiprocessor Design

- Can increase performance
 - More work per clock cycle
- Can scale system design
 - Add CPU when required
- Can reduce cost
 - External processor upgrade not required
- Can simplify software development

```
nbit_adder: adderx
```

```
  GENERIC MAP (x => 8)
```

```
  PORT MAP (AddSubR_n => H, M => carryout,
```

```
  multiplexer: mux2to3)
```

```
  GENERIC MAP (x => 32)
```

```
  PORT MAP (A => Z, S => R, G
```

```
  AddSubR_n <= (OTHERS => AddSubR_n
```

```
  H <= Z XOR AddSubR_n
```

```
  carryout XOR G(n-1) XOR H(n-1) XOR M(n-1);
```

Conclusion

Conclusion

- FPGAs can be viewed as a flexible, scalable computing platform
- FPGAs support general and specialized computing cores
- FPGA-based soft CPU cores are maturing, moving into mainstream
- FPGAs provide flexible way for leveraging productivity advantages of multi-core computing

Call for Actions

- Please download Nios II Embedded Processor
 - <http://www.altera.com/download>
- Please visit SOPC World Booth for Demos

Thank You
Q & A