Enabling New Low-Cost Embedded System Using Cyclone® III FPGAs

Unprecedented combination of low power, high functionality, and low cost to enable your new designs
Agenda

- Historical perceptions of FPGAs and current FPGA value proposition
- Hardware and software basis for making low-cost embedded system
- Embedded system design flow using FPGA
- Implementation examples and resources available
- Conclusion
Historical Perceptions of FPGAs

- In the past FPGAs have...
  - been too expensive
  - not offered enough performance
  - only been offered in low densities
  - consumed too much power
  - been challenging for which to design
# FPGA Value Proposition

<table>
<thead>
<tr>
<th>Value</th>
<th>Example end markets</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>Performance-to-price ratio</td>
<td>Video and medical imaging</td>
<td>- Parallel processing</td>
</tr>
<tr>
<td>Low cost and power per channel</td>
<td>Video surveillance, wireline, wireless</td>
<td>- Parallel processing</td>
</tr>
</tbody>
</table>
| Flexibility                  | Consumer, video and imaging, wireline, wireless | - Changing standards  
                                 |                                     | - Feature differentiation  
                                 |                                     | - Competitive response           |
| Obsolescence-proof           | Medical imaging, military, wireline, wireless | - Longevity vs. ASSPs                 |
Hardware and Software Basis for Making Low-Cost Embedded System
Unprecedented Combination

- **Low power**
  - TSMC 65-nm low-power (LP) process
  - Quartus® II software power-aware design flow
  - 120K logic elements (LEs) under ½ W static

- **High functionality**
  - Densities ranging from 5K to 120K LEs
  - Up to 4 Mbits of embedded memory
  - Up to 288 embedded multipliers for digital signal processing (DSP)

- **Low cost**
  - First low-cost 65-nm FPGA
  - Free Quartus II Web Edition software
  - Prices starting as low as $4.00

*Turn Your Ideas Into Revenue Faster*
Meeting the Needs of Emerging High-Volume Applications

2002
- 2 – 20K logic elements (LEs)
- 295-Kbits embedded RAM
- DDR support
- Nios® embedded processor

2004
- 5 – 70K LEs
- 1.1-Mbits embedded RAM
- 150 18 x 18 multipliers for DSP
- DDR2 support
- Nios II embedded processor

2007
- 50% lower power vs. Cyclone® II FPGAs
- 5 – 120K LEs
- 4-Mbits embedded RAM
- 288 18 x 18 multipliers for DSP
- Higher performance DDR2 support
- Nios II embedded processor
Cyclone III Key Architectural Features

- 65-nm low-power process
- Up to 4-Mbit embedded memory
- Up to 535 flexible user I/O pins
- Parallel and serial configuration with new remote update feature
- 400-Mbps external memory interfaces
- Up to 288 embedded multipliers for high-throughput DSP
- 5 – 120K LEs
- Dynamically configurable phase-locked loops (PLLs)
Memory Optimizations

- Increased memory block size
  - Allows for increased memory capacity

- Higher memory-to-logic ratio
  - Implement packet buffers
  - Integrate larger data and instruction caches for embedded processors
  - Integrate larger FIFO buffers

- Optimized memory-to-multiplier ratio for intensive processing applications
  - Video line buffers
  - Video and image processing

Cyclone II Family

- M4K
  - 4 Kbits
  - 18 LEs

Cyclone III Family

- M9K
  - 9 Kbits
  - 18 LEs

Up to 4 Mbits on-chip memory
Nios II Embedded Processor

- Configurable 32-bit RISC processor
- 3 members – choose for performance or size
- Library of peripherals with software support
- Perpetual use, royalty-free license
  - Altera® FPGA
  - HardCopy® structured ASIC

Implement a Processor in $0.25 of Logic
Processor Cost Reduction in Cyclone III FPGAs

Nios II/e (Economy) core

© 2007 Altera Corporation—Public
Altera, Stratix, Arria, Cyclone, MAX, HardCopy, Nios, Quartus, and MegaCore are trademarks of Altera Corporation
Processor Performance Boost in 65-nm Devices

Nios II CPU Performance (DMIPS*)

- 200
- 175
- 150
- 125
- 100

2002 2004 2007

Cyclone II

Cyclone III

15%

* Dhrystone 2.1 benchmark

© 2007 Altera Corporation—Public

Altera, Stratix, Arria, Cyclone, MAX, HardCopy, Nios, Quartus, and MegaCore are trademarks of Altera Corporation.
Multi-Core Designs in Cyclone III FPGAs

- **Nios II /e (economy) CPU**
  - 17 DMIPS
  - 4Kbytes on-chip memory

- **Nios II /f (fast) CPU**
  - 165 DMIPS
  - 64Kbytes on-chip memory
  - 4Kbytes I-cache
  - 2Kbytes D-cache

<table>
<thead>
<tr>
<th># of Processors</th>
<th>3C5</th>
<th>3C10</th>
<th>3C16</th>
<th>3C25</th>
<th>3C40</th>
<th>3C55</th>
<th>3C80</th>
<th>3C120</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>11</td>
<td>14</td>
<td>17</td>
<td>1</td>
<td>1</td>
<td>68</td>
<td>79</td>
<td>115</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>3</td>
<td>4</td>
<td>6</td>
</tr>
</tbody>
</table>

* Dhrystone 2.1 benchmark

© 2007 Altera Corporation—Public

Altera, Stratix, Arria, Cyclone, MAX, HardCopy, Nios, Quartus, and MegaCore are trademarks of Altera Corporation.
Embedded System Design Flow Using FPGAs
Embedded System Design Flow

**Hardware development**
- Processor library
  - Configure processor
- Peripherals library
  - Select and configure peripherals, IP
  - Connect blocks
  - Generate
- Custom instructions

**Software development**
- Synthesis and Fitter
  - HDL source files
  - Testbench
- Verification and debug
  - User design
  - Other IP blocks
  - JTAG, Serial, or Ethernet
  - On-Chip Debug
  - Software Trace
  - Hard Breakpoints
  - SignalTap® II
- Compiler, Linker, Debugger
  - User code
  - Libraries
  - RTOS
  - Executable code

**Tools**
- SOPC Builder GUI
- Quartus II
- Nios II IDE or Command Line
- Nios II C2H Compiler
- GNU Tools
- Altera FPGA
- Nios II EDS

**Files**
- Quartus II configuration file
- Nios II EDS

**Languages**
- C Header files
- Custom library
- Peripheral drivers

© 2007 Altera Corporation—Public
Altera, Stratix, Arria, Cyclone, MAX, HardCopy, Nios, Quartus, and MegaCore are trademarks of Altera Corporation
FPGA Hardware Development Design Flow

Design Specification

- SOPC Builder
  - Functional simulation (ModelSim®, Quartus II tools)
  - Verify logic model and data flow (no timing delays)

- Design entry/register transfer level (RTL) coding
  - Behavioral or structural description of design

- RTL simulation
  - Functional simulation (ModelSim, Quartus II tools)
  - Verify logic model and data flow (no timing delays)

- Synthesis
  - Translate design into device-specific primitives
  - Optimization to meet required area and performance constraints
  - Spectrum, Synplify, Quartus II software

- Placement and routing
  - Map primitives to specific locations inside target technology with reference to area performance constraints
  - Specify routing resources to be used
FPGA Hardware Development Hardware Design Flow

- **Timing analysis**
  - Verify performance specifications were met
  - Static timing analysis

- **Gate-level simulation**
  - Timing simulation
  - Verify design will work in target technology

- **Test FPGA on PC board**
  - Program and test device on board
  - Use SignalTap II logic analyzer and SignalProbe for debugging
  - Discussed in depth in advanced Quartus II software class
Using Quartus II Programmer

- Launch from Quartus II design software after compiling to program FPGA

<hardware>.sof programming file generated during the Quartus II hardware compile
SOPC Builder System Design Software

1. Select and configure IP
2. Select connections
3. Generate system

Easy, Flexible, Fast

© 2007 Altera Corporation—Public
Altera, Stratix, Arria, Cyclone, MAX, HardCopy, Nios, Quartus, and MegaCore are trademarks of Altera Corporation
Nios II IDE (Integrated Development Environment)*

- Leading-edge software development tool in the Nios II Embedded Design Suite
- Target connections
  - Hardware (JTAG)
  - Instruction set simulator
  - ModelSim-Altera software
- Advanced hardware debug features
  - Software and hardware breakpoints, data triggers, trace
- Flash memory and Quartus II programming support

* Based on Eclipse 3.2/CDT 3.1
User-Defined Custom Peripherals

- Add a peripheral not included with the Nios II system
  - To perform some kind of proprietary function or perhaps a standard function that is not yet included as part of the Nios II kit
  - To expand or accelerate system capabilities

- You are now going to learn how to connect your own design directly to the Nios II system via the Avalon™-Memory Mapped interconnect
  - Note: As many peripherals contain registers, you could also have chosen to use a programmed input/output (PIO) rather than connect directly to the bus
Custom Peripherals

- Map into Nios II memory space
- Can be on-chip or off-chip
  - HDL code or an external component on your board
    - HDL code can map inside SOPC Builder system or out
Creating Avalon Peripherals

- No need to worry about creating the bus interface to Avalon Interconnect inside your peripheral
  - Implement only the signals you need
  - Avalon Memory Mapped Interconnect will adapt to connect to the peripheral’s ports
  - Timing handled automatically
  - Fabric created for you
  - Arbiters generated as needed

Concentrate Effort on Peripheral Functionality!
Map Ports to Avalon Signal Types

Avalon Interconnect Fabric

module my_peripheral

  input clk, cs, wr_n, addr, clr_n;
  input [31:0] wr_data;
  output [31:0] rd_data;
  output [7:0] pwm_out;

  .
  .
  .

Peripheral’s ports (mapped to Avalon interconnect)
Component Editor

Two Uses:

1. Create a wrapper file that connects Avalon bus to peripheral living outside SOPC system (on- or off-chip)

2. Create direct *on-chip* connection between Avalon bus and user HDL code
Custom Peripheral Integration Into Avalon
Component Editor

- Writes a TCL script file instead of proprietary class.ptf file

```
# TCL File Generated by Component Editor on:
# Wed Jan 17 10:18:07 PST 2007
# DO NOT MODIFY

set_source_file "/data/korthner/SPR/230791/tb_sopc/my_onchip_mem.vhd"
set_module "my_onchip_mem"
set_module_description ""
set_module_property instantiateInSystemModule true
set_module_property version 1.0
set_module_property group ""
set_module_property editable true
set_module_property libraries "altera.altera_europa_support_lib.all"

# Module parameters

# Interface avalon_slave_0
add_interface "avalon_slave_0" "avalon" "slave" "asynchronous"
set_interface_property "avalon_slave_0" "interleaveBursts" "false"
set_interface_property "avalon_slave_0" "addressAlignment" "DYNAMIC"
set_interface_property "avalon_slave_0" "isNonVolatileStorage" "false"
```

Scripting interface

- Well-defined TCL API to describe components and their interfaces
- Build your own TCL-defined components
  - Automatically found by SOPC Builder
Device Driver for PWM Peripheral

- "avalon_pwm_regs.h"
  - Manually add to software project
  - Loads peripheral registers to run `pwm`

```c
#ifndef __ALTERA_AVALON_PWM_REGS_H__
define __ALTERA_AVALON_PWM_REGS_H__
#include <io.h>

#define IORD_ALTERA_AVALON_PWM_DIVIDER(base)            IORD(base, 0)
#define IOWR_ALTERA_AVALON_PWM_DIVIDER(base, data)      IOWR(base, 0, data)

#define IORD_ALTERA_AVALON_PWM_DUTY(base)       IORD(base, 1)
#define IOWR_ALTERA_AVALON_PWM_DUTY(base, data) IOWR(base, 1, data)

#endif /* __ALTERA_AVALON_PWM_REGS_H__ */
```
Manually Add Driver Code to Project

- Using same method as adding application code
Custom Instructions

- Add custom functionality to the Nios II design
  - To take full advantage of the flexibility of FPGA

- Dramatically boost processing performance
  - With no increase in $f_{MAX}$ required

- Application examples
  - Data stream processing (e.g. network applications)
  - Application-specific processing (e.g. MP3 audio decode)
  - Software inner loop optimization
Custom Instructions

- Augment Nios II instruction set
  - Multiplexing user logic into arithmetic logic unit (ALU) path of processor pipeline
Custom Instructions Tab

- Enabled from the **Custom Instructions** tab in the Nios II CPU Wizard in SOPC Builder

Add a custom instruction from built-in library

Or import your own user logic
C Language Software Interface

- Nios II IDE generates macros automatically during build process
- Macros defined in `system.h` file

```c
#define ALT_CI_<your_instruction_name>(instruction arguments)
```

- Example of user C-code that references Bitswap custom instruction:

```c
#include "system.h"
int main (void)
{
    int a = 0x12345678;
    int a_swap = 0;

    a_swap = ALT_CI_BSWAP(a);
    return 0;
}
```

Assembly language interface also available
Verilog and VHDL Templates Available

C:\altera<ver#>\nios2eds\examples\verilog\custom_instruction_template\
C:\altera<ver#>\nios2eds\examples\VHDL\custom_instruction_template\
Accelerate Software Execution

- Example: CRC Algorithm (64 Kbytes)

27 Times Faster

530 Times Faster
Implementation Examples and Resources Available
WiMAX Pico-Cell Base Transceiver Station

Downlink

MAC Layer

FEC Encoding → Symbol Mapper → DL OFDM Engine

Sub Channelization → IFFT, Insert CP → DUC

UL OFDM Engine

Desub Channelization → FFT, Remove CP → DDC

Channel Estimation, Equilization

Downlink

Uplink

Ranging

FEC Decoding → Symbol Demapper → OFDMA Symbol Rate Processing

Bit Rate Processing

Intermediate Frequency (IF) Processing

© 2007 Altera Corporation—Public

Altera, Stratix, Arria, Cyclone, MAX, HardCopy, Nios, Quartus, and MegaCore are trademarks of Altera Corporation

38
Enabling the Highest Integration

Abundant Memory, Multipliers, and Logic
To Do More For Less

© 2007 Altera Corporation—Public
Altera, Stratix, Arria, Cyclone, MAX, HardCopy, Nios, Quartus, and MegaCore are trademarks of Altera Corporation
Wireless Applications Resources

- Altera and partner intellectual property (IP) cores
  - FEC, FFT/IFFT, FIR, NCO, CIC, and more
- Low-cost FPGA Starter Kit, Cyclone III Edition
- *Design Low-Cost, Low-Power Wireless Systems with New FPGAs* QuickCast
- *Using Cyclone III FPGAs for Emerging Wireless Applications* white paper

www.altera.com/cyclone3-markets
H.264 Encoder Block Diagram

- Macroblock of input image signal
- Prediction error signal
- Quantized coefficients
- Entropy coding
- Video over IP
- DDR2 SDRAM

Processing-intensive blocks

© 2007 Altera Corporation—Public
Altera, Stratix, Arria, Cyclone, MAX, HardCopy, Nios, Quartus, and MegaCore are trademarks of Altera Corporation
Enable Low-Cost H.264 Encoding

**Processing-intensive blocks**

*Implement SD H.264 Encoder in a Single Device for Under \( \frac{1}{4} \) W and $5 Per Channel*
Video and Image Processing Resources

- Video and image processing IP
  - Library of nine common video and image processing functions from Altera
  - Compression IP available from Altera partners including ATEME, Barco, 4i2i, and CAST

- Video processing reference design

- Video training course
  - Advanced DSP design: using FPGAs to architect and optimize a video and image processing system

- Low-cost FPGA Starter Kit, Cyclone III Edition

- Video daughtercard

- Design Video and Image Processing Systems with Low-Cost Cyclone III FPGAs QuickCast

- White papers
  - Video and Image Processing Design Using FPGAs
  - Video Surveillance Implementation Using FPGAs
  - Medical Imaging Implementation Using FPGAs

www.altera.com/cyclone3-markets
Universal, Flexible, and Scalable Display Controller

Integrate the exact display functions you need

Add a custom video and image processing algorithm

Support the right memory for your application

Support for multiple simultaneous displays in every resolution, including non-standard

Display panel

Integrated mini-LVDS RSDS PPDS

© 2007 Altera Corporation—Public

Altera, Stratix, Arria, Cyclone, MAX, HardCopy, Nios, Quartus, and MegaCore are trademarks of Altera Corporation
Display Application Resources

- Video and image processing IP suite
  - Library of nine common video and image processing functions optimized for Altera FPGAs
- Video processing reference design
- Low-Cost FPGA Starter Kit, Cyclone III Edition
- Microtronix ViClaro II HD Video Enhancement Development Platform
- Develop a Display System Using Low-Cost Cyclone III FPGAs QuickCast
- White papers
  - Cyclone III FPGAs Enable a New Class of LCD HDTVs
  - A Flexible Architecture to Drive Sharp Two-Way Viewing Angle and Standard LCDs
  - Satisfy the Demand for Rapid Feature Enhancement in Consumer Display Products

www.altera.com/cyclone3-markets
eCos RTOS

- Commercial port from eCosCentric
  - Open source RTOS
    - Designed for deeply embedded applications
    - Configurable down to 10s of Kbytes
    - Commercially supported and maintained
      - Support and maintenance contract in place for Nios II embedded processor v7.1 and v7.2
eCosPro Starter Kit (Free Version)

- Available for download from eCosCentric website
- Features:
  - eCos kernel and hardware abstraction layer (HAL)
  - ISO C and math libraries
  - Memory-based file systems
  - RedBoot bootloader
  - BSP support for on-board LAN91C111 Ethernet, RS232, and flash devices (Cyclone II and Stratix® II kits)
  - Debug connections: USB Blaster (JTAG), Ethernet, and serial
  - eCos RTOS graphical configuration tool
  - Windows and Linux host development support
  - POSIX compatibility layer
eCosPro Developer Kit (Paid Version)

- Includes all the eCosPro Starter Kit features plus:
  - Product support
    - Incident support (bug fixes)
    - Advice line service (email support)
  - Additional peripherals
    - Triple speed Ethernet media access control (TSE MAC)
    - Watchdog timer
  - Additional software
    - JFFS2 journaling flash file system
- Additional fee-based services
  - Device driver/BSP development
  - Application consulting
  - On-site training
Conclusion
Conclusion

- Altera FPGAs adding value to external processors
  - Focus in most common interface cores
  - Support coprocessing and peripheral expansion
  - Drag-and-drop ease of use with SOPC Builder

- 65 nm + Nios II process expands Altera’s embedded market
  - New device families reduce cost, increase performance
  - New ecosystem partners added per customer demand
Thank You!