

**CSE 566 – Fall 2004**  
**Reconfigurable System On Chip Design**

**Lecture 9 : COREs :**  
**Logic Cores, Alliance Cores,**  
**and OpenCores.org**

**John W Lockwood**  
Washington University in St. Louis  
(includes material from Xilinx and OpenCores.org)  
[Lockwood@arl.wustl.edu](mailto:Lockwood@arl.wustl.edu)

Copyright 2004

<http://www.arl.wustl.edu/~lockwood/class/cse566-f04/>



## Introduction to Cores

- A *core* is a ready-made function that you can instantiate into your design as a “black box”
- Cores can range in complexity
  - Simple arithmetic operators, such as adders, accumulators, and multipliers
  - System-level building blocks, including filters, transforms, and memories
  - Specialized functions, such as bus interfaces, controllers, and microprocessors
- Some cores can be customized with parameters

## Benefits of Using Cores

- Save design time
  - Cores are created by expert designers who have in-depth knowledge of Xilinx FPGA architecture
  - Guaranteed functionality saves time for system verification
- Increase design performance
  - Cores that contain mapping and placement information have predictable performance
  - The data sheet for each core provides performance expectations

## Available Cores

- LogiCORE™ solutions  (Free)
  - DSP functions
    - Time skew buffers, FIR filters, correlators
  - Math functions
    - Accumulators, adders, multipliers, integrators, square root
  - Memories
    - Pipelined delay elements, single and dual-port RAM
    - Synchronous FIFOs
  - PCI master and slave interfaces, PCI bridge
- AllianceCORE™ solutions  (Not Free)
  - Peripherals
    - DMA controllers
    - Programmable interrupt controllers
    - UARTs
  - Communications and networking
    - ATM
    - Reed-Solomon encoders / decoders
    - T1 framers
  - Standard bus interfaces
    - PCMCIA, USB

## CORE Generator System

- Graphical User Interface (GUI) allows
  - Access to the cores themselves, with
  - Customizable parameters (available for some cores)
  - Data sheets
- Interfaces with design entry tools
  - Creates graphical symbols for schematic designs
  - Creates instantiation templates for HDL designs
- Web access from the Help menu
  - IP Center contains new cores to download and install
    - Provides access to the latest cores
  - Direct access to
    - <http://support.xilinx.com>

## CORE Generator GUI

The screenshot shows the Xilinx CORE Generator interface. On the left, a tree view displays core categories: Basic Elements, Clocking, Communication & Networking, Digital Signal Processing, Math Functions, Memories & Storage Elements, and Standard Bus Interfaces. A callout bubble labeled "Core Types" points to this tree. The main window shows the "Contents of: Communication & Networking - Telecommunications" section. Below this, a table lists various cores with their names, types, versions, and vendors. A callout bubble labeled "Cores organized by function, vendor, or device family" points to the tree view. Another callout bubble labeled "Core type, version, device support, vendor, and status" points to the table.

Name	Type	Version	Support	Vendor	Status
1024 Channel ADPCM	AUSI95C	1.0	✓	Amphion Semi...	✓
16 Channel ADPCM	AUSI95C	1.0	✓	Amphion Semi...	✓
256 Channel ADPCM	AUSI95C	1.0	✓	Amphion Semi...	✓
512 Channel ADPCM	AUSI95C	1.0	✓	Amphion Semi...	✓
768 Channel ADPCM	AUSI95C	1.0	✓	Amphion Semi...	✓
ADPCM32	ADPCM32	1.0	✓	Xilinx, Inc.	✓
BOOSTLife Bluetooth b...	BOOSTLife	1.0	✓	NewLogic Tec...	✓
Convolution Encoder	CONVENC	2.0	✓	Xilinx, Inc.	✓

# Customize cores with Parameters

Core Overview

Contact information for vendor

Parameters to customize the core

Web Links

Data sheet

# Documentation : CORE Data Sheets

**Features**

- Fully synchronous, drop-in modules for the Virtex™-II family using Virtex™-II Block Memory for performance and efficiency.
- Supports all three Virtex™-II write mode options: read after write, read before write, and no read on write.
- Supports data widths from 1 to 256 bits and memory depths from 512 to 1M words.
- Supports RAM functions enabling simultaneous write operations to separate locations and simultaneous read operations from the same location.
- The ports are completely independent of each other.
- Available in the Xilinx CORE Generator™ System V2.3.

**Functionality**

The Dual Port Block Memory module is composed of single or multiple Virtex II 18K6 blocks enabling a deeper and/or wider memory implementation. The Selectable memory is True Dual-Port™ SRAM, offering fast, discrete, and large blocks of memory in the Virtex-II device.

A memory module has two independent ports that enable shared access to a single memory space and is generated based on user-defined width and depth. Both ports are functionally identical, with each port providing read and write access to the memory. Simultaneous reads from the same memory location may occur, but all other simultaneous reading from and writing to the same memory location will result in correct data being written into the memory, but invalid data being read.

The Memory's Port A and Port B are configured to support user-defined data input and address widths. When both ports are disabled (EN\_A and EN\_B inactive) the memory contents and output ports remain unaffected. When either port is enabled (EN\_A or EN\_B asserted) all memory operations occur on the rising edge of the clock input.

**Schematic Symbol**

**Waveforms**

**Figure 7: No Read-on-Write Mode Waveform**

The other port must not address that memory location for a write/read while the clock-to-clock setup window. Conflicts do not cause any physical damage to BlockRAM cells. For more information on conflict resolution, refer to the Virtex™-II Handbook.

**Core Resource Utilization**

The number of BlockRAM primitives required is dependent on the values of the depth and data width fields selected in the CORE generator parameter window, and is at least  $(\text{depth} \times \text{width}) / 16382$  and will exceed this number for many configurations.

For some memory depths extra logic is required to decode the address and multiplex the outputs from various primitives. Virtex-II CLB slices are used to provide this functionality. The number of slices required depends on the way that the depth is constructed from the primitives, the data width, and the implementation of any decoding or multiplexing.

Parameter Name	Type	Notes
Component Name	String	Up to 256 characters
Width	Integer	Ranges from 1 to 256
Depth	Integer	Ranges from 512 to 1M (up to 10M) with increments of 512
Write Modes	String	Default: Read after Write. Options are: Read after Write, Read before Write, and No Read on Write
Enable EN_A, EN_B	Boolean	Default: false
Nondestructive Read	Boolean	Default: false
NO_READ_ON_WRITE	Boolean	Default: false
Register Input	Boolean	Default: false
Additional Output Pipe Stages	Integer	Default: 0
Synchronous initialization	Boolean	Default: false
Synchronous initialization value	Integer	Default: 0

Example of good documentation for a core.

## HDL Design Flow for Core Generation and Integration

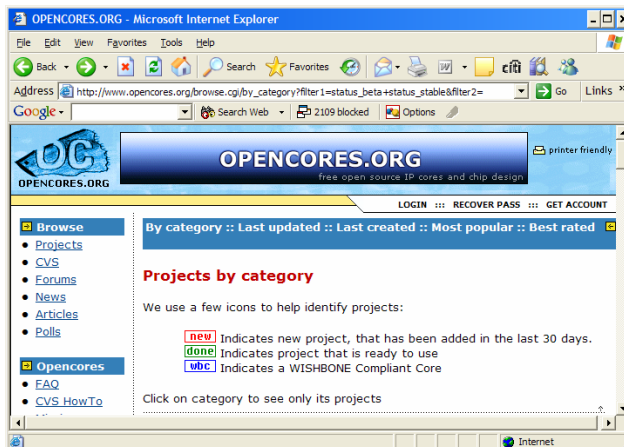
- Generate core and produce:
  - Instantiation template files (VHO or VEO)
  - Netlist file (EDN)
  - Behavioral simulation wrapper files (VHD or V)
- Instantiate the core into your HDL source
  - Template provided in the VEO or VHO file
- Wrapper files enable behavioral simulation
  - Analyze the wrapper file for each core before analyzing the file that instantiates the core
- Synthesize and implement using Netlist (EDN)

## Open Cores Website

- **OPENCORES.ORG**
  - <http://www.opencores.org/>

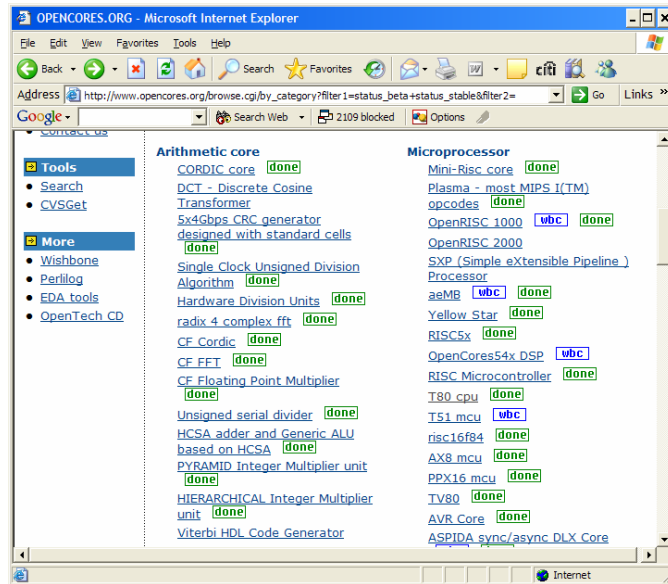
– Open source cores in VHDL and Verilog source form

- Arithmetic
- Microprocessors
- Co-processors
- Communications
- Crypto
- DSP
- Memory
- AMBA bus
- ...



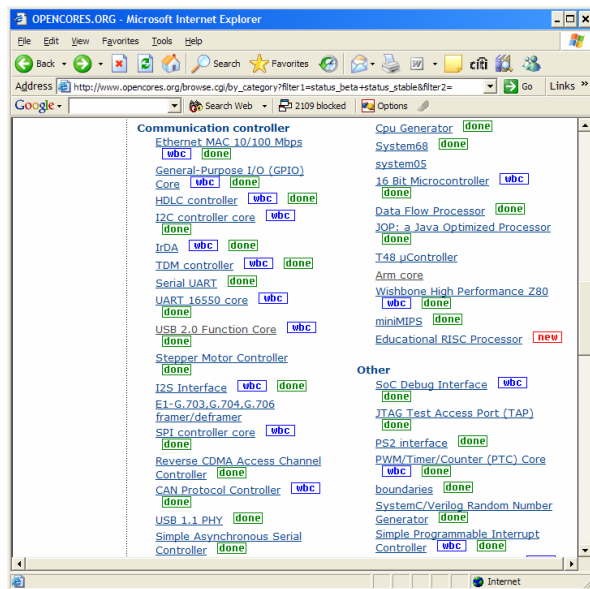
## Open Arithmetic & Microprocessor Cores

- Arithmetic
  - Dividers
  - CRC
  - FFT
  - ...
- Micro-processors
  - RISC
  - Z80
  - 32-bit
  - 16-bit
  - ARM
  - miniMIPS
  - ...



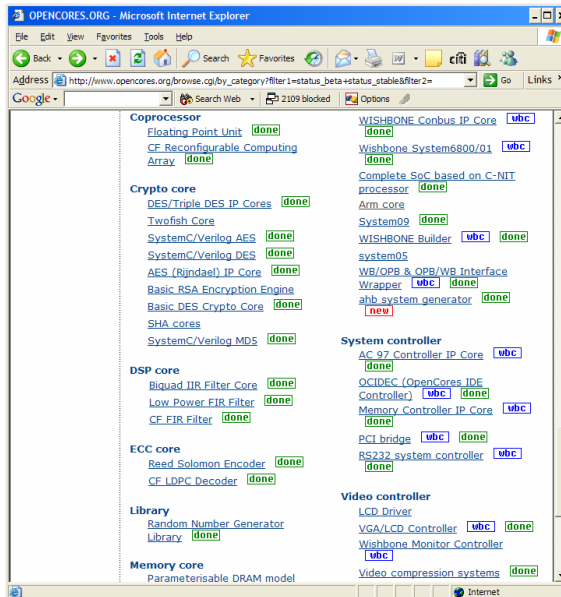
## Open Communication Controllers Cores

- Communications
  - Ethernet 10/100 Mbps
  - HDLC
  - UART
  - USB 2.0
  - ...
- Other
  - Random number generator
  - Interrupt controllers
  - ...



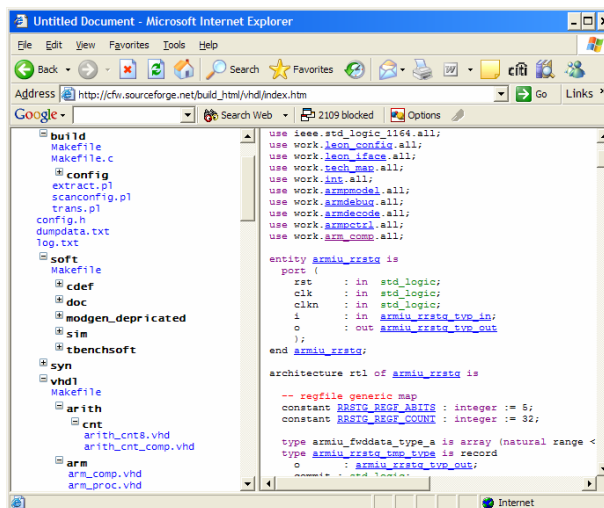
## Open Crypto and Controller Cores

- **Cryptography**
  - DES/3DES
  - AES ...
- **Bus Controllers**
  - Wishbone
  - AHB ...
- **Memory Controllers**
  - SDRAM
  - DDR ...
- **System Controllers**
  - PCI



## Annotated VHDL source on SourceForge

- **Open source ARM VHDL model**
  - [http://cfw.sourceforge.net/build\\_html/vhdl/index.htm](http://cfw.sourceforge.net/build_html/vhdl/index.htm)
- **Website provides access to VHDL source code**
  - Color coded syntax
  - Hyperlinks



## Washington University FPX Cores

- **Layered Internet Protocol Wrappers**
  - Process packets directly in hardware at Gigabit/second rates
- **TCP/IP Protocol processing wrappers**
  - Process millions of traffic flows in hardware at Gigabit/second rates
- **Bloom filters**
  - Find thousands of strings anywhere in data
  - Find longest matching prefixes of packets
- **FP-Grep & FP-Sed**
  - Search for Regular Expressions with wildcards
- **Internet Worm Detection**
  - Find malware passing through a network
- **Switching and routing cores**
  - Route traffic between FPGA components
- **More information**  
<http://www.arl.wustl.edu/arl/projects/fpx/reconfig.htm>

## License Requirements for Cores

- **Commercial Cores**
  - Integrators pay to use the core (per core or per copy)
  - Authors receive royalty for use of core
- **BSD-style license**
  - No warranty
  - May be distributed in binary or source form
  - Authors must be acknowledged
- **GNU Public License (GPL)**
  - All modifications must be distributed under the GPL
  - Cannot take away rights to access a modified version of core
  - If you use an item under GPL item, you **MUST** distribute all information about it and **NOT** prevent others from redistributing or modifying it.
- **Washington University FPX cores**
  - Free for non-commercial use (i.e., university research)
  - License required for commercial use (i.e., product sales)
- **Other licenses**
  - Depend on the terms set by the author(s)