CE233(CAD), Lecture 14:

IP-cores and design reuse

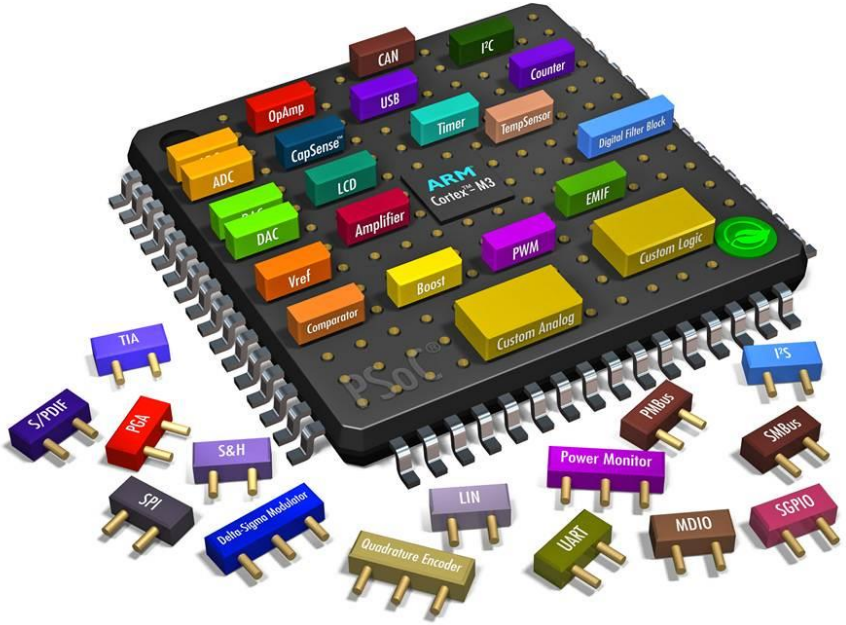
Mehdi Modarressi

Department of Electrical and Computer Engineering,

University of Tehran

Introduction

- Introduction to SoCs
- Design reuse and IP-cores



Picture source: http://www.cypress.com

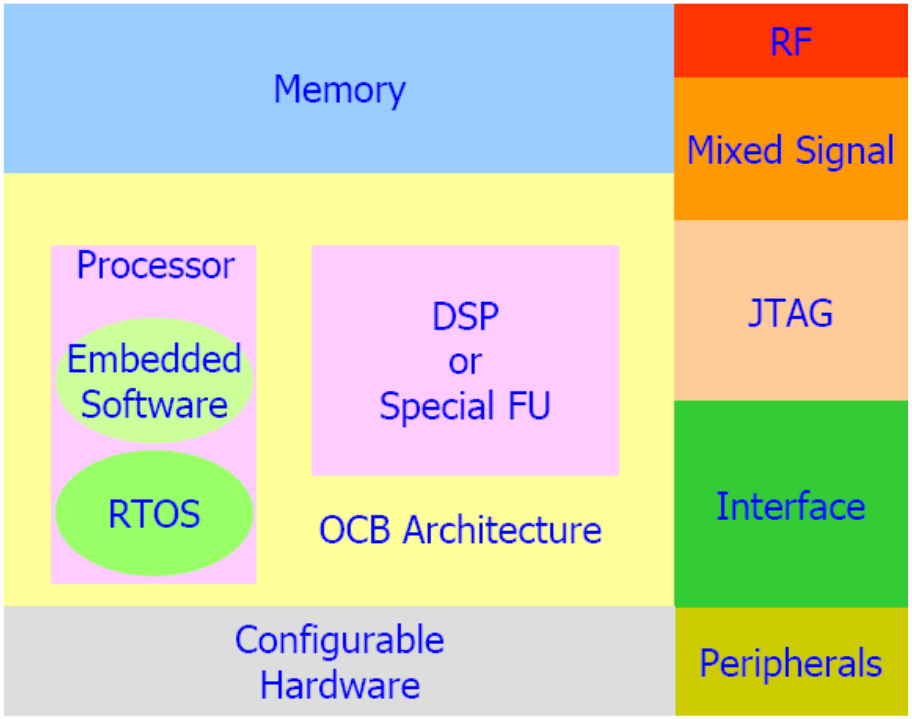
Introduction

- Advances in semiconductor technology
  - Today's chip can contains 7 billion transistors
- The consequences
  - Components connected on a printed circuit board can now be integrated onto single chip
  - Emergence of systems-on-chip

Picture source: www.xbitlabs.com

SoC

- An SoC may contain
  - Custom hardware blocks
    - A.K.A. Hardware accelerator
  - DSP blocks
  - Embedded CPU
  - Embedded Memory
  - Real World Interfaces (USB, PCI, Ethernet)
  - Software (both OS and Applications)
  - Mixed-signal Blocks
    - DAC, ADC
  - FPGAs



SoC benefits

- Reduce size
- Low cost
- Reuse
- Low power
- High performance
- High reliability

SoPC

- System-on-Programmable Chip (SoPC): FPGA vendors provide an SoC on an FPGA chip
- In addition to configurable part there are also:
  - Processor cores
  - Peripherals
  - On-chip buses
  - DSP blocks
  - A rich range of soft IP cores as a library can be implemented on the configurable part
  - Operating systems and software drivers
  - …

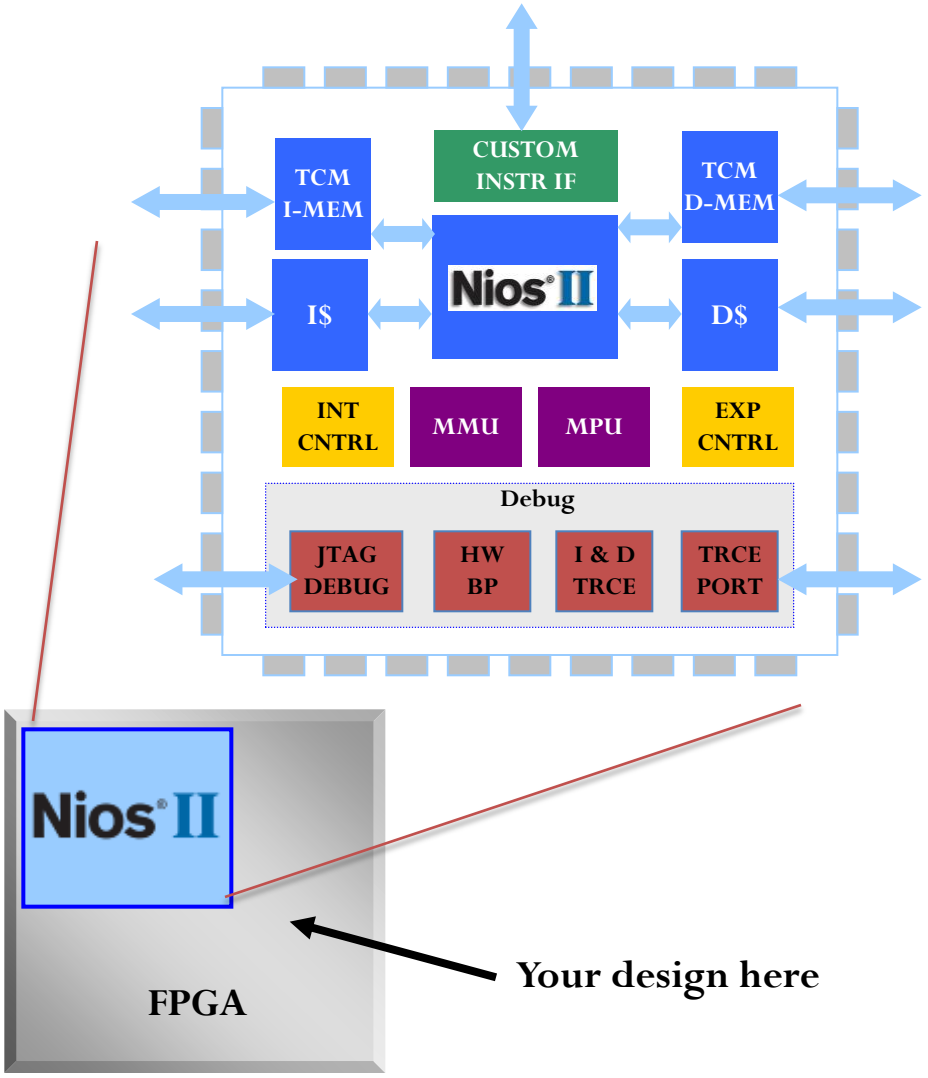
SoPC

- Xilinx SoPCs:
  - 2 or 4 PowerPC 405 and 440 cores and many peripherals
  - Xilinx EDK for development
- Altera SoPCs:
  - NIOS II or ARM Cortex cores and many peripherals
    - ARM Cortex is a dual core processor
  - Altera SoPC Builder for development

Picture source: http://www.altera.com

Altera Startix

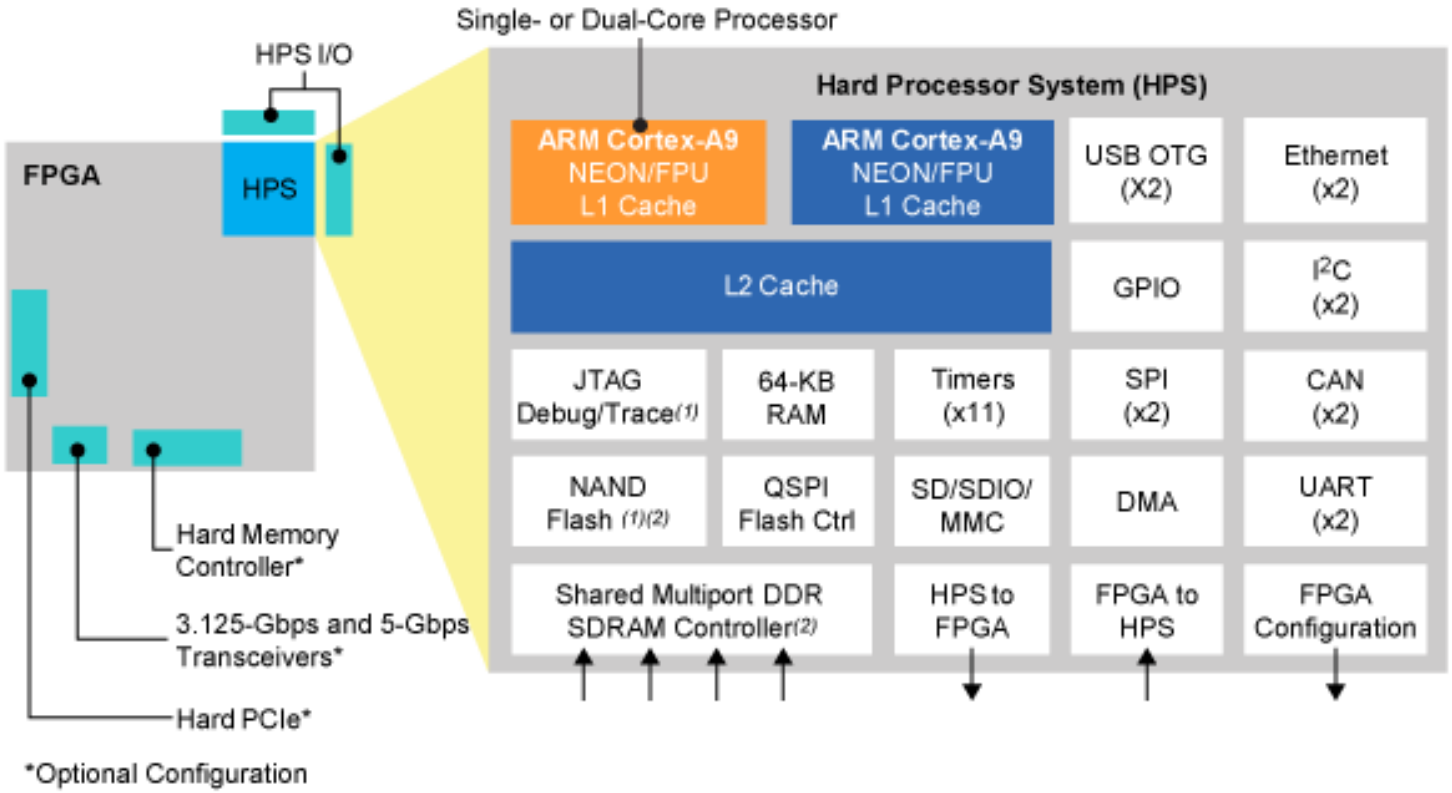
- 32-bit embedded NIOS-II processor



Picture source: http://www.altera.com

Altera Cyclone V

- Integrate an ARM-based hard processor system (HPS) consisting of processor, peripherals, and memory interfaces with the FPGA fabric

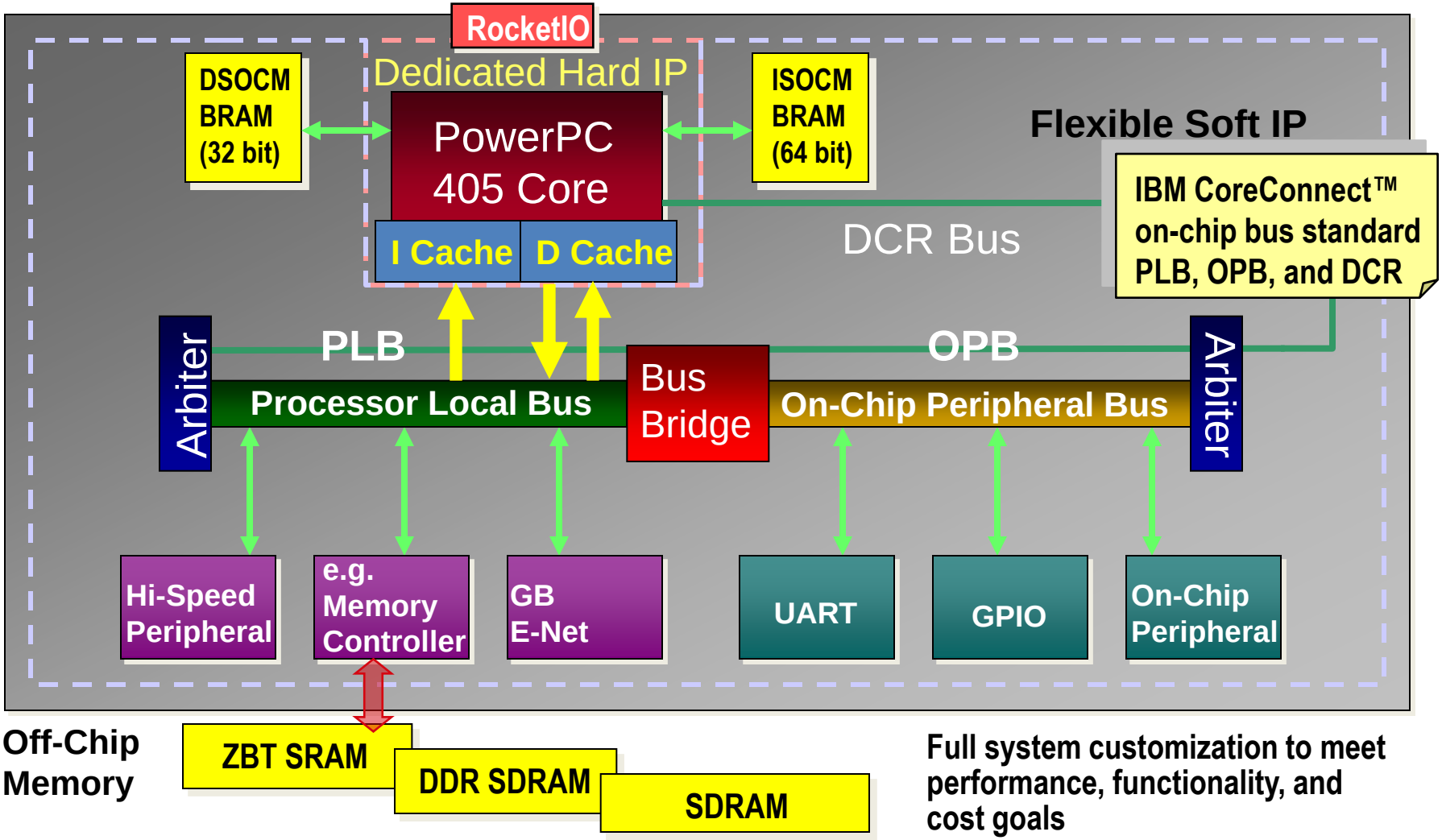


Picture source: http://www.altera.com

Xilinx SoPCs

- Some Xilinx FPGA families have a processor core and a large selection of components
- Example: Virtex-II family of Xilinx FPGAs has one PPC-405 core
  - A large variety of peripherals
  - Uses IBM CoreConnect bus
    - Fast PLB : Processor Local Bus
    - Slower OPB : On-chip Peripheral Bus
- Xilinx EDK tool to program these SoPCs

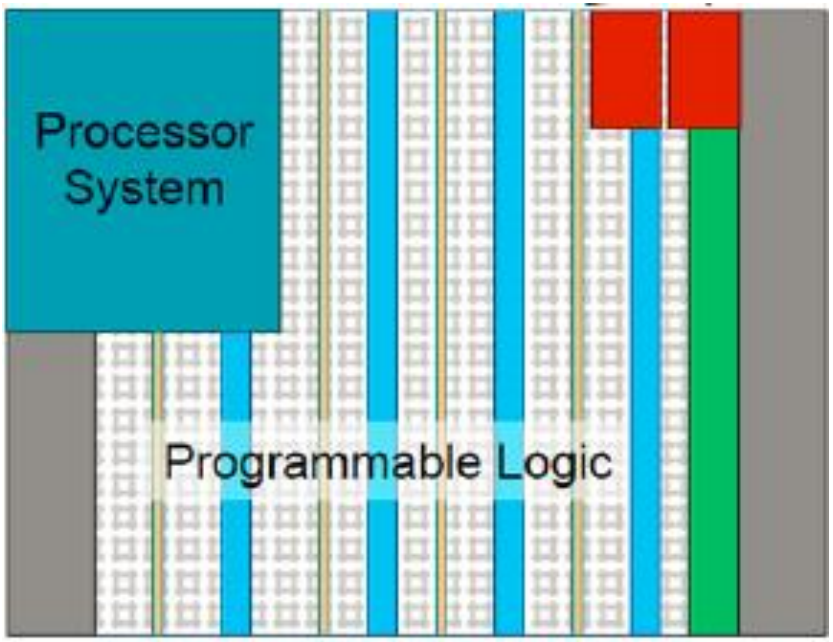
PowerPC-based Virtex-II

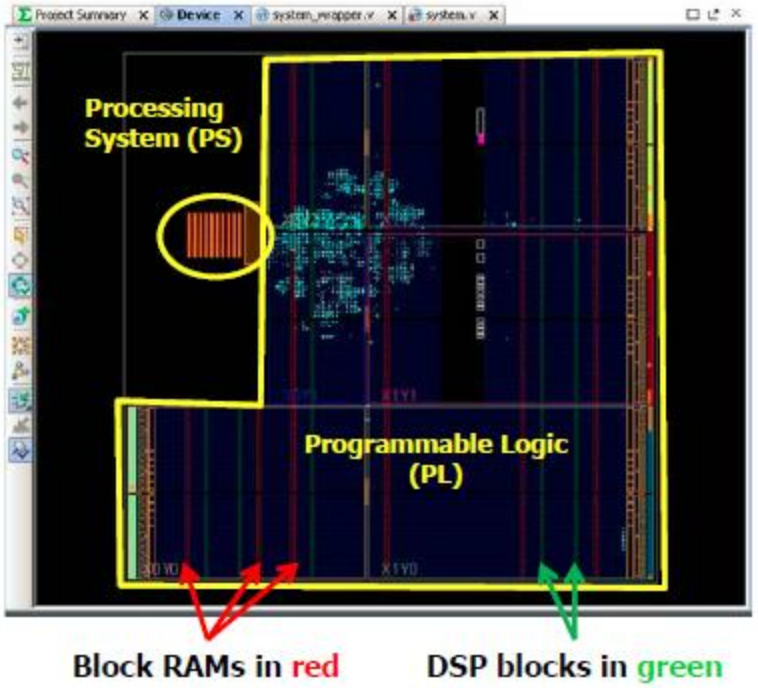


Picture source: http://www.xilinx.com

Xilinx Zynq

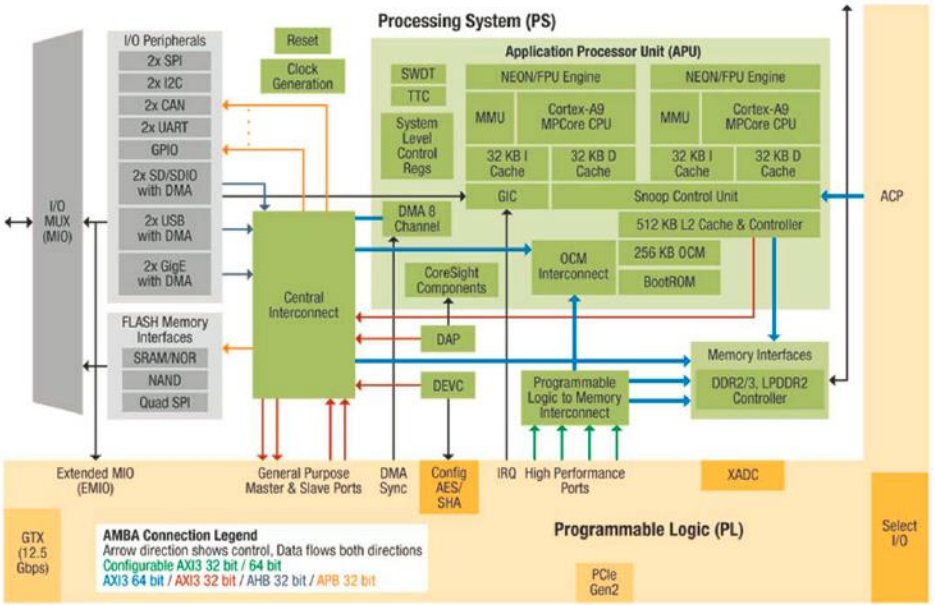
- A modern SoPC with many hardcore logic





Xilinx Zynq

- FPGA fabric with embedded ARM processor on single die
  - Programmable logic (PL)
  - Processing system (PS)
- PS based on dual-core ARM Cortex-A9
  - A rich set of peripherals: USB, CAN, SPI, Flash memory interface, DAM controller, memory controller, …..



SoC Challenges

- Productivity gap
- Time-to-market pressure
- Difficult verification due to increasing complexity
- Difficult timing closure due to deep submicron
- Difficult integration due to various levels and areas of expertise

Solution:

Block-based design with reusable IPs

Design reuse

- Using IP-cores
- "IP" means "intellectual property"
- Circuit designs cleaned up and made available to other engineers
- IP examples:
  - Microprocessor cores
  - DSP cores
  - Ethernet, USB, FireWire, 802.11 cores
  - DMA, timer, UART, and other generics
  - Special-purpose cores
    - MPEG-2 compression, decompression, encryption cores, filters,etc…

Hard IP-cores

- Delivered in physical form
- Fully designed, placed and routed, and characterized for timing, power, etc.
- Tied to a manufacturing process
- Actual physical layout
  - Fixed shape
- Technology specific layout
  - Size and speed determined

Soft IP-cores

- Circuit design delivered in "source code" format instead of physical film or physical-layout information
  - In forms of RTL code or netlist
  - technology dependent gate-level netlists are called firm-core
- Functions are fixed, may be parameterizable
  - Size and shape are determined when synthesizing
- Equivalent to delivering C source code
  - The program doesn't change at all, only the details of the executable binary you create

Tutorial- reusing IP-cores

- Keyboard and LCD driver IP cores
- Some existing free IP-cores can be selected to be used to design and implement an embedded system
  - The cores can be either selected from the free cores included in Xilinx tools or be picked up from some free IP-core websites, such as *www.design-reuse.com* or *opencores.org*