REC FPGA Seminar IAP 1998

Session 1:

Architecture, Economics, and Applications of the FPGA

Seminar Format

- Four 45 minute open sessions
  - two on Wed. January 21st, two on Fri. January 23rd
  - session 1: Architecture, Economics, and Applications of FPGAs
  - session 2: Design tools, Configuration and Practical Considerations
  - session 3: Advanced FPGA design techniques, Optimizations, and Tricks
  - session 4: Future Directions and Applications of FPGAs
- Hands-on session (preregistered only)
  - starts on Mon. January 26th in 6.004 lab, continues through the week
  - three 45 minute orientations for design tools and RHP4K board
  - one basic lab and one advanced lab assignment
    - “cylon” lab
    - pong, the video game
  - plus two days open lab time for your own project

Seminar Goals

- Teach practical FPGA knowledge
  - From design concept to finished product
  - When to use FPGAs
  - Which one to use
  - How to integrate them into your circuit
- Impart the ability to implement design solutions to real world problems in FPGAs

Architecture: Genesis

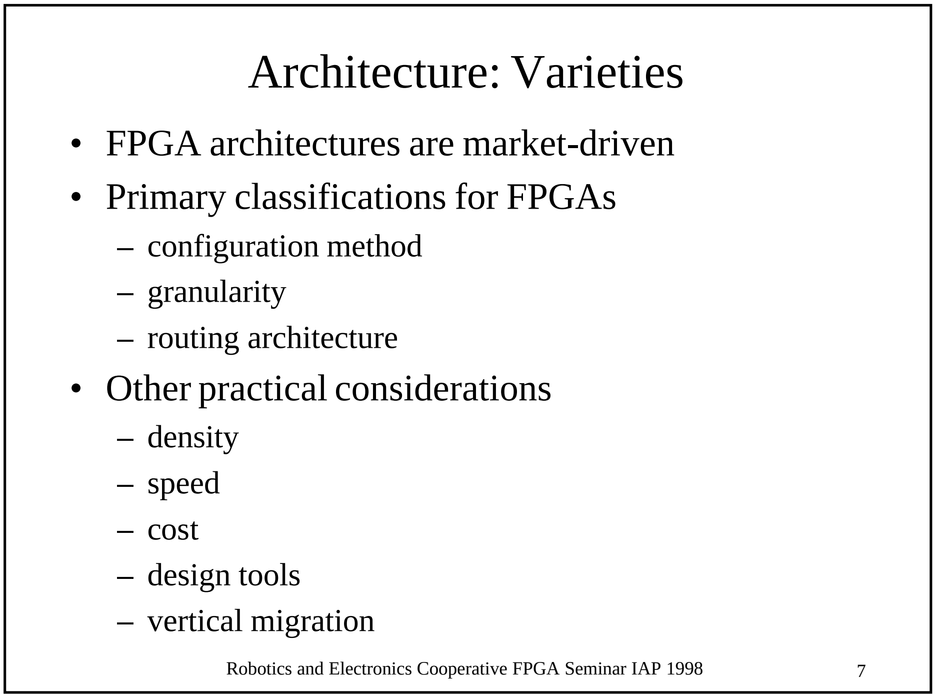
- Field Programmable Gate Array
  - Name belies genesis and original purpose
- Gate arrays are a kind of ASIC
  - ASIC types include CBIC, GA, and FC
  - GA: moderately fast-turn, cost effective, volume ASIC
    - Mfg. bakes a set of wafers with only transistor diffusions (first half of fabrication time and cost)
    - User specifies metalization layers for device application
    - Saves on NRE (fewer mask sets to make)

Architecture: Genesis

(gate array structure)

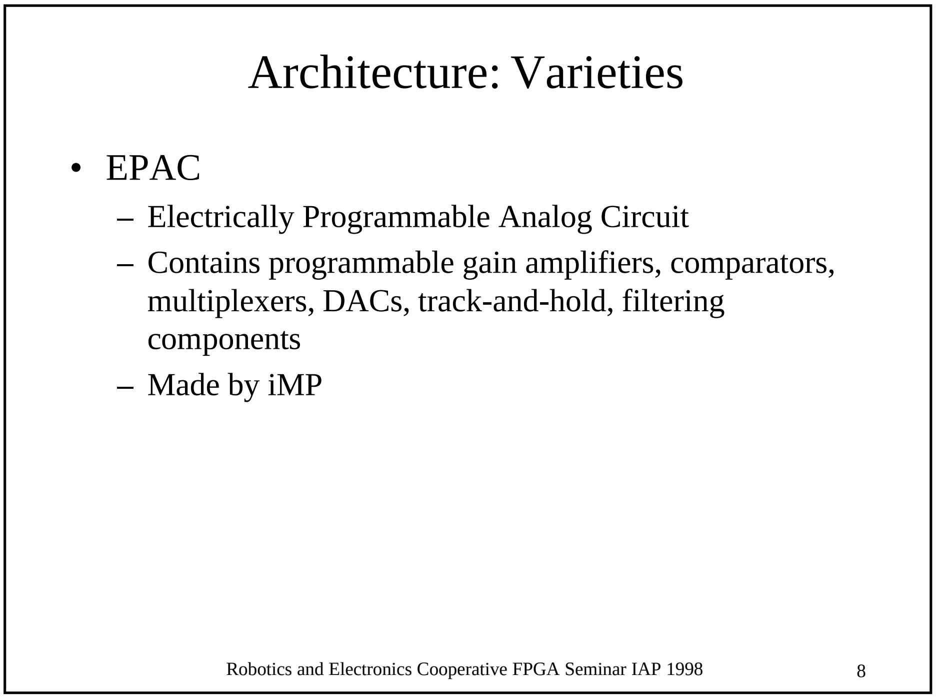
Architecture: Genesis

- Industry generally considers FPGA to be a kind of ASIC
  - extremely fast turn, low volume ASIC
    - sold as a completed chip in package
    - FPGA has a “sea of gates” structure with user-defineable interconnections
      - fuses, antifuses, or pass gates
  - target market: low volume, fast turn operations
    - higher cost per device countered by lower NRE cost
    - many FPGA vendors make OTP devices because they are cheaper per device (Actel’s anti-fuse)



Architecture: Varieties

- FPGA architectures are market-driven
- Primary classifications for FPGAs
  - configuration method
  - granularity
  - routing architecture
- Other practical considerations
  - density
  - speed
  - cost
  - design tools
  - vertical migration



Architecture: Varieties

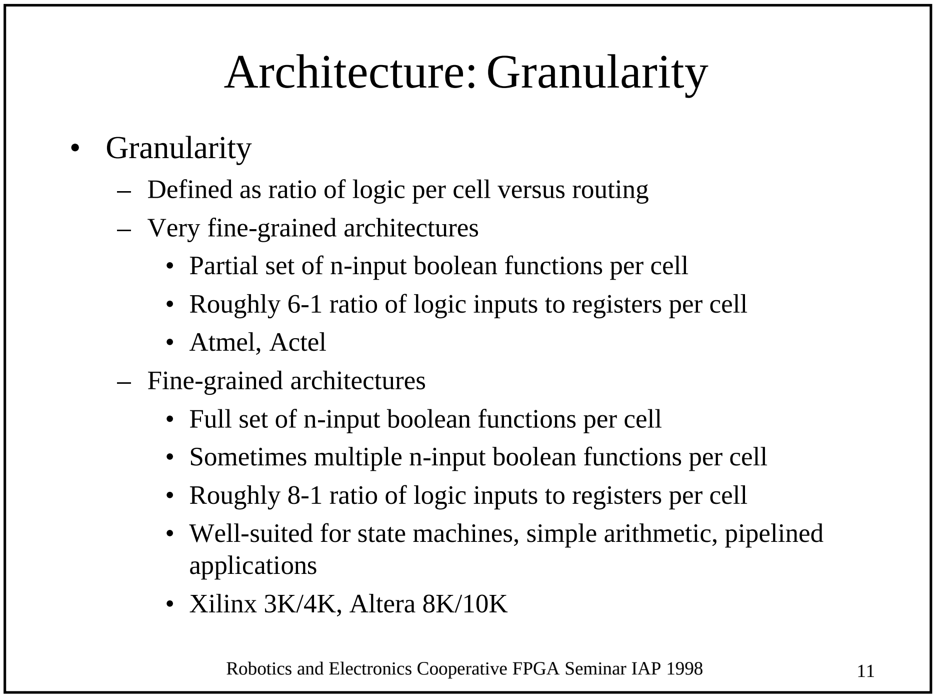
- EPAC
  - Electrically Programmable Analog Circuit
  - Contains programmable gain amplifiers, comparators, multiplexers, DACs, track-and-hold, filtering components
  - Made by iMP

Architecture: Configuration

- Configuration method
  - In-circuit programmable methods
    - SRAM based (Xilinx 2K/3K/4K, Altera 8K/10K, Lucent Orca)
      - volatile, but fast configuration times
      - must reprogram on every power-up
      - some architectures offer partial reconfiguration (Atmel)
      - most expensive in terms of area and timing costs
      - standard CMOS process
    - EEPROM based (Altera 7K/9K)
      - nonvolatile, slow config; sometimes requires extra voltages for programming and erasing
      - special silicon processing required

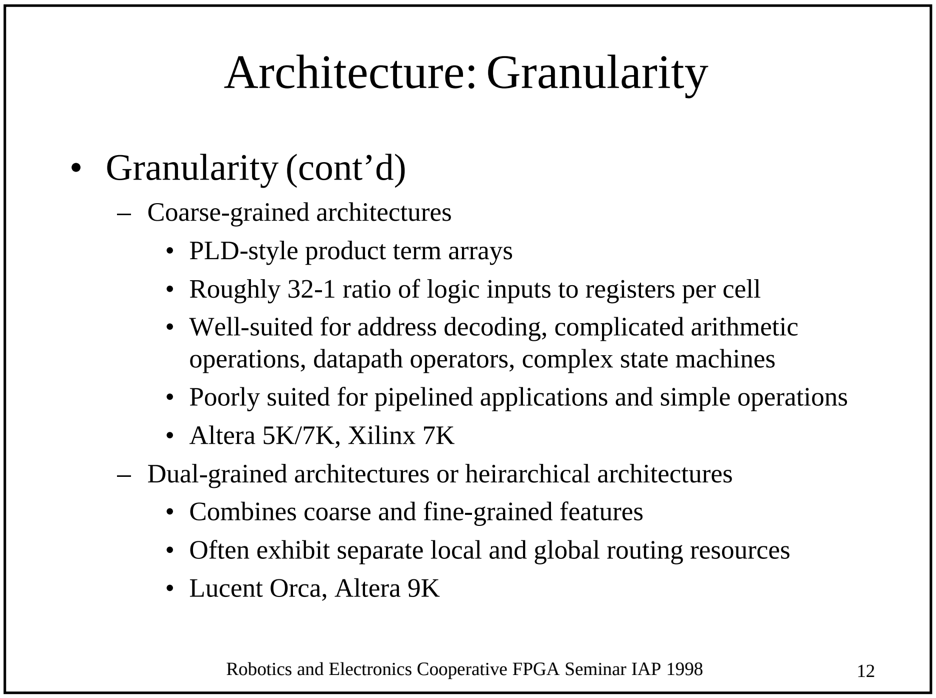
Architecture: Configuration

- Configuration method (cont'd)
  - Pre-assembly programmable methods
    - Antifuse based (Actel, Quicklink FPGAs)
      - nonvolatile, very fast links
      - permanent configuration (OTP)
      - smallest link size (lower cost)
      - special silicon processing technology required
    - (E)EPROM based (Altera 5K, 7K, Xilinx 7200, 7300)
      - nonvolatile, moderate performance
      - reprogrammable after special erase cycle
      - medium-sized link
      - special silicon processing technology required



Architecture: Granularity

- Granularity
  - Defined as ratio of logic per cell versus routing
  - Very fine-grained architectures
    - Partial set of n-input boolean functions per cell
    - Roughly 6-1 ratio of logic inputs to registers per cell
    - Atmel, Actel
  - Fine-grained architectures
    - Full set of n-input boolean functions per cell
    - Sometimes multiple n-input boolean functions per cell
    - Roughly 8-1 ratio of logic inputs to registers per cell
    - Well-suited for state machines, simple arithmetic, pipelined applications
    - Xilinx 3K/4K, Altera 8K/10K



Architecture: Granularity

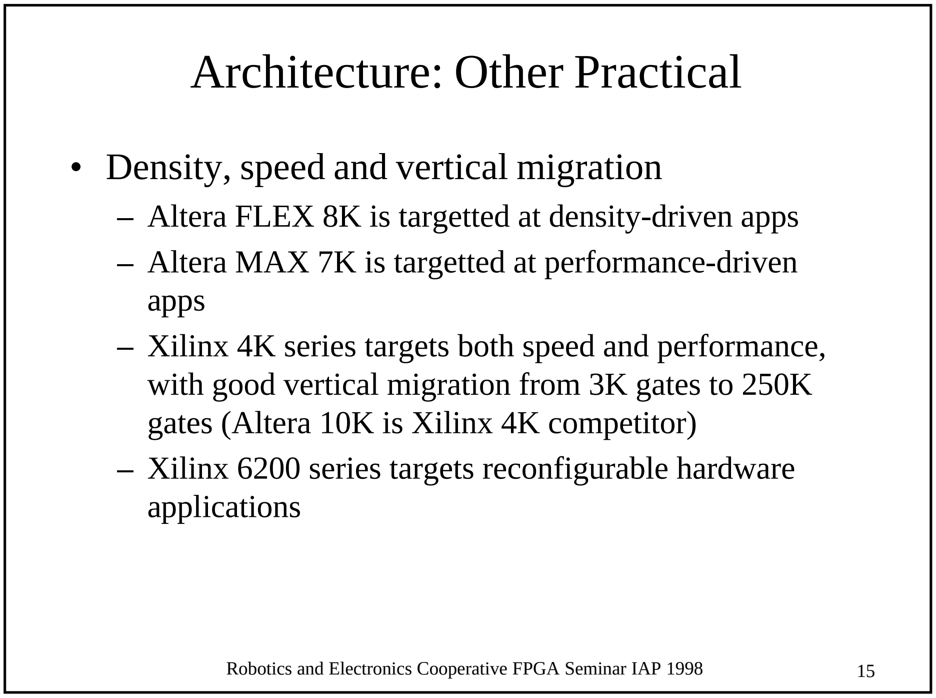
- Granularity (cont'd)
  - Coarse-grained architectures
    - PLD-style product term arrays
    - Roughly 32-1 ratio of logic inputs to registers per cell
    - Well-suited for address decoding, complicated arithmetic operations, datapath operators, complex state machines
    - Poorly suited for pipelined applications and simple operations
    - Altera 5K/7K, Xilinx 7K
  - Dual-grained architectures or heirarchical architectures
    - Combines coarse and fine-grained features
    - Often exhibit separate local and global routing resources
    - Lucent Orca, Altera 9K

Architecture: Routing

- Routing method
  - Fine-grained
    - Short hops (1 to 8 logic cells spanned per track)
    - Path-dependant timing
    - Exhibits high density
    - Flexible switch matrices
    - Less logic placement constraints
  - Coarse-grained
    - Tracks span entire chip
    - Fixed timing regardless of logic placement
    - Lower density
    - Logic placement constrained by routing availability

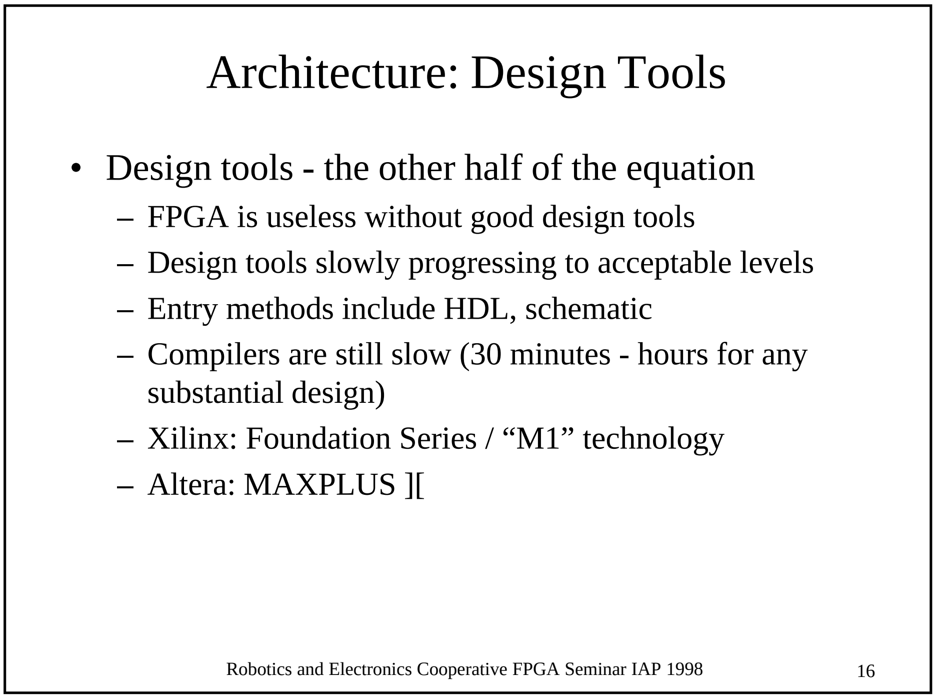
Architecture: Routing

- Routing method (cont’d)
  - Heirarchical routing
    - Local, fine-grained routing between cells
    - Global, coarse-grained routing between groups of cells
    - Usually path-dependant timing
    - Best of both worlds, but can be difficult to utilize efficiently



Architecture: Other Practical

- Density, speed and vertical migration
  - Altera FLEX 8K is targetted at density-driven apps
  - Altera MAX 7K is targetted at performance-driven apps
  - Xilinx 4K series targets both speed and performance, with good vertical migration from 3K gates to 250K gates (Altera 10K is Xilinx 4K competitor)
  - Xilinx 6200 series targets reconfigurable hardware applications



Architecture: Design Tools

- Design tools - the other half of the equation
  - FPGA is useless without good design tools
  - Design tools slowly progressing to acceptable levels
  - Entry methods include HDL, schematic
  - Compilers are still slow (30 minutes - hours for any substantial design)
  - Xilinx: Foundation Series / “M1” technology
  - Altera: MAXPLUS][

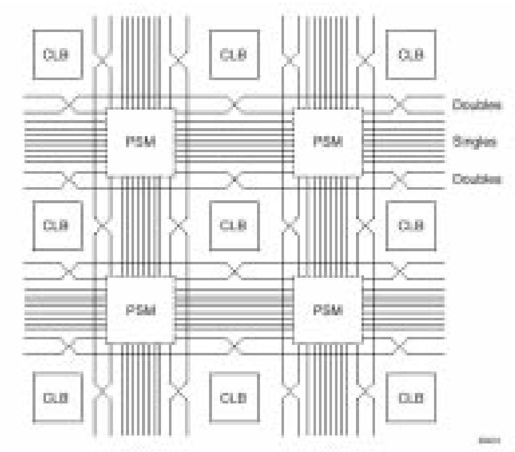
Architecture: Cost

- Cost formulas for FPGAs are complex
  - OTP FPGAs tend to be cheaper
  - Established lines are cheaper than new lines
  - Cost increases exponentially with performance and density
  - Some lines are targetted at cost-sensitive applications (Altera 7K)
  - Not all speed grade-density combos available from manufacturers

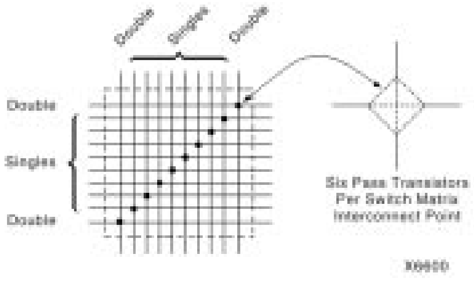
Detailed Architecture: Xilinx 4000E

- Fine-grained logic, SRAM based, with fine-grained routing
  - Array of CLBs embedded in single length / double length / quad length / longline routing resources + PSM
    - CLB = Configurable Logic Block
      - Two 4-input LUTs (LookUp Tables) and one 3-input LUT
      - Two SR D-type flip flops
      - Bypass paths and carry/cascade logic
    - PSM = Programmable Switch Matrix
      - 10 interconnect points per matrix
      - Each interconnect contains six pass transistors for full connectivity between four directions
      - Located at intersections of single and double length lines

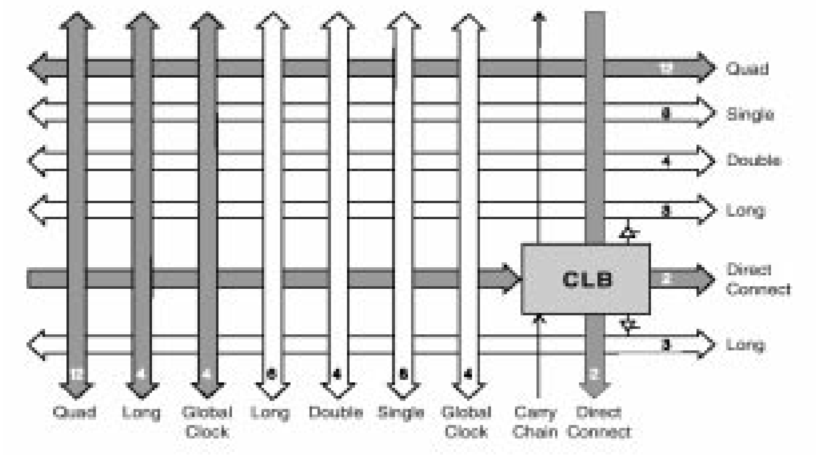
Detail Architecture: Xilinx 4000E



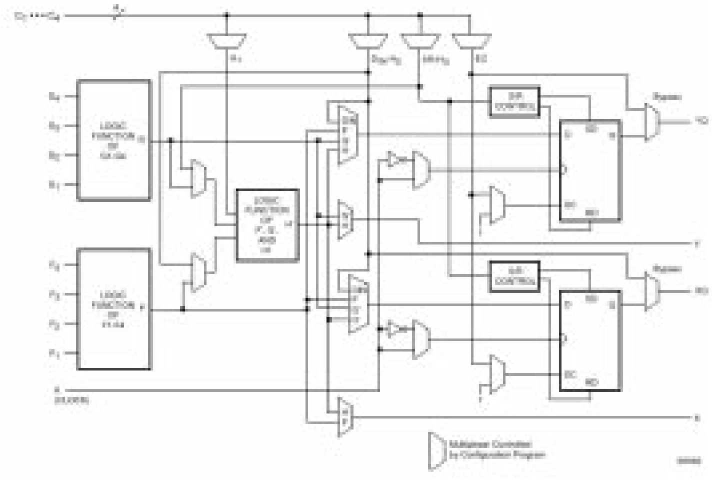
Detailed Architecture: Xilinx 4000E



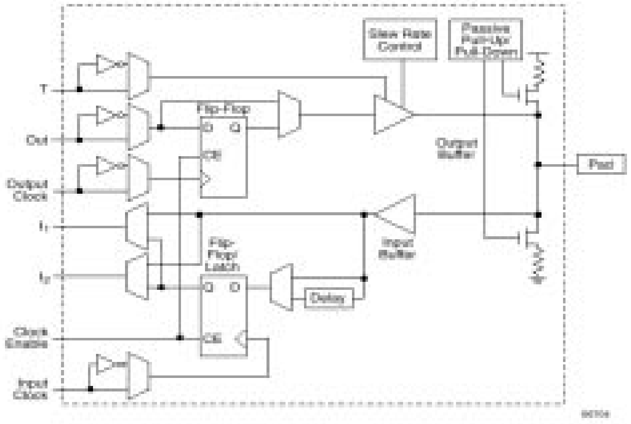
Detailed Architecture: Xilinx 4000E



Detailed Architecture: Xilinx 4000E


LOGIC FUNCTION OF G1-G4
LOGIC FUNCTION OF F1-F4
LOGIC FUNCTION OF F', G', AND H1
S/R CONTROL
S/R CONTROL
YQ
Y
XQ
X
Multiplexer Controlled by Configuration Program

Detailed Architecture: Xilinx 4000E



Detailed Architecture: Xilinx 4000E

- Configuration
  - total (device) reconfiguration (no partial reconfig)
  - several configuration modes available
    - parallel and serial modes
    - master and slave modes
    - daisy chain ability
  - device bitstreams between 50Kbits and 400Kbits
  - config rate around 10 Mbit/sec
    - max reconfig rate in a few tens of milliseconds
    - typical reconfig in a couple of seconds

Detailed Architecture: Xilinx 4000E

- Other features
  - distributed RAM
    - CLB LUTs can function as a 32x1, 16x1, or 16x2 RAM
    - synchronous RAM options available
  - internal tri-state buffers
  - global routing resources
  - JTAG boundary scan
  - configuration readback
  - programmable slew rate and logic levels in IOBs
  - common per-package pinout for all devices
    - allows for easy vertical migration

Applications

- Fast-turn, low volume ASIC (uninteresting)
- Reconfigurable Hardware Processors
- One-connector I/O solutions
- Rapid prototyping

Applications: RHP

- Direct implementation of algorithms in hardware
  - circumvents instruction fetch, decode, issue overhead
  - unrestricted parallelism
  - disadvantage: little hardware abstraction, difficult to use
- RISC framework with reconfigurable instruction set
  - user-defined instructions depending on process context
  - prevents the MMX disease
  - easier to use, more hardware abstraction, but lower performance

Applications: RHP

- Optimal ISA
  - compiler analyzes code and chooses an ISA optimal for the problem, and bundles the hardware description for the ISA with the code object
- Configurable memory management and caching
  - useful for implementing special OS features
  - VM paging schemes directly in hardware
- Ultimate RHP-one processor, any ISA
- In the future - possibly adaptive processors which automatically optimize their architecture per application

Applications: Direct Hardware

- Ideal for implementing simple, repetative operations (overhead operations)
  - time synchronization on Novell networks
  - CAM lookup tables for IP routing and neural nets
  - encryption/decryption
  - FEA (finite element analysis)
  - Relaxation networks
  - database searching
    - higher peformance with special architectures (embedded RAM)

Applications: I/O solutions

- One-connector I/O solutions
  - use a single connector with any protocol desired
    - ex: a DB-25 which can do SCSI, IEEE1284 parallel, serial
  - ideal for space-limited applications
  - “Object oriented hardware”
    - devise a system such that a device plugged into the I/O port uploads the hardware configuration necessary to implement the communications protocol
    - protocol upgrades are a cinch
  - limited by electrical signalling compatibility issues
  - drawback - can be confusing to users, potentially damaging to hardware

Applications: EA

- Evolutionary algorithms
  - some research done on FPGAs already
    - tone recognition application
  - possibly requires intimate knowledge of FPGA hardware
    - vendor licencsing issues
  - EA apps do not map well into current FPGA architectures
    - however, with the right FPGA EA could yield very interesting results

Applications: Rapid Prototyping

- FPGAs are a handy thing to have on the lab bench
  - simple digital circuits no longer require wiring or order parts, etc
  - modifying and duplicating existing designs is relatively easy
  - with the right design tools, hardware design re-use is an additional benefit