The apeNEXT project

F. Rapuano
INFN
and
Dip. di Fisica
Università Milano-Bicocca

for the APE Collaboration

INFN Ferrara, Rome
DESY Zeuthen
Bielefeld University
Université de Paris-Sud, Orsay
The Group

• **Italy**
  • Roma: N. Cabibbo, F. di Carlo, A. Lonardo, S. de Luca, D. Rossetti, P. Vicini
  • Ferrara: L. Sartori, R. Tripiccione, F. Schifano
  • Milano-Parma: R. de Pietri, F. di Renzo, F. Rapuano

• **Germany**
  • DESY, NIC: H. Kaldass, M. Lukyanov, N. Paschedag, D. Pleiter, H. Simma

• **France**
  • Orsay: Ph. Boucaud, J. Micheli, O. Pene,
  • Rennes: F. Bodin
Outline of the talk
apeNEXT is completely operational and all its circuits are functioning perfectly
Mass production starting

- A bit of history
- A bit of HW
- A bit of SW
- Large installations
- Future plans
The Ape paradigm

- Very efficient for LQCD (up to 65% peak), but usable for other fields
  The “normal” operation as basic operation
    \[ a \times b + c \] (complex)
- Large number of registers for efficient optimization, Microcoded architecture (VLIW)
- Reliable and safe HW solutions
- Large software effort for programming and optimization tools
“The APE family”

Our line of Home Made Computers

Once upon a time (1984) Italian lattice physicists were sad …

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td># nodes</td>
<td>16</td>
<td>2048</td>
<td>2048</td>
<td>4096</td>
</tr>
<tr>
<td>Topology</td>
<td>flexible 1D</td>
<td>rigid 3D</td>
<td>flexible 3D</td>
<td>flexible 3D</td>
</tr>
<tr>
<td>Memory</td>
<td>256 MB</td>
<td>8 GB</td>
<td>64 GB</td>
<td>1 TB</td>
</tr>
<tr>
<td># registers (w.size)</td>
<td>64 (x32)</td>
<td>128 (x32)</td>
<td>512 (x32)</td>
<td>512 (x64)</td>
</tr>
<tr>
<td>clock speed</td>
<td>8 MHz</td>
<td>25 MHz</td>
<td>66 MHz</td>
<td>200 MHz</td>
</tr>
<tr>
<td>peak speed</td>
<td>1 GFlops</td>
<td>100 GFlops</td>
<td>1 TFlops</td>
<td>7 TFlops</td>
</tr>
</tbody>
</table>
\[(a + ib)(A + iB) + C + iD = (Aa - Bb + C) + i(Ab + Ba + D)\]
Lattice conferences as status checkpoints

- Lattice 2000 (Bangalore) FR, general ideas
- Lattice 2001 (Berlin) R. Tripiccione, clear ideas, VLSI design started, simulator running
- Lattice 2002 (Boston) D. Pleiter, all designs complete, most HW prototypes ready, VLSI design complete
- Lattice 2003 (Tsukuba) no talk, 6 months delay in VLSI. Delivered in December
apeNEXT Architecture

• 3D mesh of computing nodes, 64bit arithm
  • Custom VLSI processor - 200 MHz (J&T)
  • 512 registers
  • 1.6 GFlops per node (complex “normal”)
  • 256 MB ÷ 1 GB memory per node
  • 3.2 GB/s memory bandwidth (128 but chan)
  • Prefetch queues
• First neighbor communication network loosely synchronous (fifo based)
  • $\rho = 8\div16 \Rightarrow 200\text{ MB/s}$ per channel
• Scalable 25 GFlops ÷ 7 Tflops
  16 ÷ 4096 nodes
• Linux PCs as Host system
Topology

- Two directions (Y,Z) on the backplane
- Direction X through front panel cables

**System topologies:**

- Processing Board
  
  \[ 4 \times 2 \times 2 \sim 26 \text{ GF} \]

- subCrate (16 PB)
  
  \[ 4 \times 8 \times 8 \sim 0.4 \text{ TF} \]

- Crate (32 PB)
  
  \[ 8 \times 8 \times 8 \sim 0.8 \text{ TF} \]

- Large systems
  
  \[ (8*n) \times 8 \times 8 \]
- Dominant Technologies:
  - LVDS: 1728 (16*6*2*9) differential signals 200MB/s, 144 routed via cables, 576 via backplane on 12 controlled-impedance (100Ω) layers
  - High-Speed differential connectors:
    - Samtec QTS (J&T Module)
    - Erni ERMET-ZD (Backplane)
- Collaboration with NEURICAM spa
Computing & control integrated

→ no *glue logic*

→ Reduced time for project, simulation and test of the prototype
J&T
the Arithmetic box

• Pipelined “normal” a*b+c (8 flops) per cycle
J&T
Remote I/O

• fifo-based communication:
  • LVDS
  • 1.6 Gb/s per link
    (8 bit @ 200MHz)
  • 6 (+1) independent bi-dir links
- CMOS 0.18µ, 7 metal (ATMEL)
- 200 MHz
- Double Precision Complex Normal Operation
- 64 bit AGU
- 8 KW program cache (user-controllable)
- 128 bit local memory channel
- 6+1 LVDS 200 MB/s links
- BGA package, 600 pins
J&T Module

- J&T
- 9 DDR-SDRAM, 256Mbit (x16) memory chips
- 6 Link LVDS up to 400MB/s
- Host Fast I/O Link (7th Link)
- I2C Link (slow control network)
- Dual Power 2.5V + 1.8V, 7-10W estimated
- Dominant technologies:
  - SSTL-II (memory interface)
  - LVDS (network interface + I/O)
# NEXT BackPlane

- 16 PB Slots + Root Slot
- Size **447x600 mm²**
- **4600 LVDS** differential signals, point-to-point up to **600 Mb/s**
- **16** controlled-imp. layers (32 Tot)
- Press-fit only
- Erni/Tyco connectors
  - **ERMET-ZD**
- Providers:
  - APW (primary)
  - ERNI (2nd source)

## Activity Table

<table>
<thead>
<tr>
<th>Activity</th>
<th>Status</th>
<th>Who</th>
<th>Cost</th>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>BP development</td>
<td>Done</td>
<td>APW(ERNI)</td>
<td>32 KEuro</td>
<td></td>
</tr>
<tr>
<td>BP prototypes (3)</td>
<td>Done</td>
<td>APW</td>
<td>41 KEuro</td>
<td></td>
</tr>
</tbody>
</table>

- Connector kit cost: **7KEuro (!)**
- PB Insertion force: **80-150 Kg (!)**
Host I/O interface

I2C: bootstrap & control

7th-Link (200MB/s)
Host I/O Interface

- Altera APEX II based
- PCI Interface 64bit, 66Mhz
  - PCI Master Mode for 7th Link Intf
  - PCI Target Mode for I2C Intf
- 7th Link: 1(2) bidir chan. (200*9 M/s)
- I2C: 4 independent ports
- QuadDataRate Memory (x32)
Programming Languages

- Tao (was Apese)
  - Fortran-like, very simple to learn
  - Dynamical grammar, OO-style programming, QCDlib
  - Many tens thousand lines of codes existing all over Europe
  - All APEmille code compiles with no changes

- C
  - Based on lcc
  - Language extensions (complex vector, ~, where (), all() …)

- SASM
  - High level assembly (should never be needed!!)
Software Overview

- **rtc**
  - Zz parser
  - kernel (wraps)
  - asm generator
  - *.sasm

- **nlcc**
  - frontend
  - (trees, symb.)
  - backend
  - *.sasm

- **mpp**
  - macro expansion
  - label analysis
  - cache utilities
  - *.masm

- **sofan**
  - muladd fusion
  - move removal
  - dead code removal
  - AGU-optimisation
  - *.masm

- **shaker**
  - scheduling
  - register allocation
  - label resolution
  - compression
  - *.no
  - *.nex

- **sf**
  - functional simulation

- **linker**
  - *.nex
Assembler Optimizer: Sofan

- Optimization operating on low-level assembly
- Based on optimization toolkit SALTO (IRISA, Rennes)
- Optimization steps:
  - merging APE-normal operations
  - removing dead code
  - eliminating register moves
  - optimizing address generation:
  - instruction pre-scheduling
  - ...

![Diagram of memory controller and register file]
# Benchmarks: Linear Algebra

<table>
<thead>
<tr>
<th>operation</th>
<th>IO-Op</th>
<th>Flop</th>
<th>sustained performance</th>
<th>measured</th>
</tr>
</thead>
<tbody>
<tr>
<td>vnorm</td>
<td>1</td>
<td>4</td>
<td>50%</td>
<td>37%</td>
</tr>
<tr>
<td>zdotc</td>
<td>2</td>
<td>8</td>
<td>50%</td>
<td>41%</td>
</tr>
<tr>
<td>zaxpy</td>
<td>3</td>
<td>8</td>
<td>33%</td>
<td>29%</td>
</tr>
<tr>
<td>$U V$</td>
<td>27</td>
<td>202</td>
<td>92%</td>
<td>65%</td>
</tr>
</tbody>
</table>

"maximum" sustained performance ← ignoring latency of floating point pipeline and loop overhead

**Optimization “tricks”:**
- loop unrolling
- burst memory access
- instructions kept in buffer

**Performance limitations:**
- start-up latency
- loop overhead

From Pleiter, Simma,...
## Benchmarks: Results from C

<table>
<thead>
<tr>
<th>operation</th>
<th>assembler</th>
<th>C</th>
<th>C + Sofan</th>
</tr>
</thead>
<tbody>
<tr>
<td>vnorm</td>
<td>37%</td>
<td>31%</td>
<td>34%</td>
</tr>
<tr>
<td>zdotc</td>
<td>41%</td>
<td>28%</td>
<td>40%</td>
</tr>
</tbody>
</table>

→ Assembler programming not required
Benchmarks: Wilson-Dirac Operator

\[ \Psi_x = D_{xy}[U] \Phi_y \]

Consider worst case: local lattice size \(16 \times 2^3\)

- Measured sustained performance: 55%
- Measured number of stretch cycles: 4%

Optimization "tricks":

- keep gluon fields local
- pre-fetching 2 sites ahead
- orthogonal communication directions
- some unrolling
Operating system
Step Scaling Function for the running coupling constant in SU(3), 16 node apeNEXT
Costs

- 1700 KEuro developments
  - 550 KEuro + 1050 KEuro
    - Non VLSI   VLSI

  NO SALARIES

- Prototype production cost ~ 0.6-0.7 Euro/Mflops
  Large scale as low as ~ 0.5, see next
• Like APEmille, apeNEXT will be commercially available.
• Slow EU procedure for official tender (start 03/04, end ~ 08/04) to choose the company
• Committee (Vicini, Simma, FR, INFN administratives) at work
• Machines by Nov-Dec 2004 at a rate of 2x512-node/Month

<table>
<thead>
<tr>
<th>Location</th>
<th>GFlops</th>
<th>Crates</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bielefeld</td>
<td>130</td>
<td>2</td>
</tr>
<tr>
<td>Zeuthen</td>
<td>520</td>
<td>8</td>
</tr>
<tr>
<td>Milan</td>
<td>130</td>
<td>2</td>
</tr>
<tr>
<td>Bari</td>
<td>65</td>
<td>1</td>
</tr>
<tr>
<td>Trento</td>
<td>65</td>
<td>1</td>
</tr>
<tr>
<td>Pisa</td>
<td>325</td>
<td>5</td>
</tr>
<tr>
<td>Rome 1</td>
<td>520</td>
<td>8</td>
</tr>
<tr>
<td>Rome 2</td>
<td>130</td>
<td>2</td>
</tr>
<tr>
<td>Orsay</td>
<td>16</td>
<td>1/4</td>
</tr>
<tr>
<td>Swansea</td>
<td>65</td>
<td>1</td>
</tr>
</tbody>
</table>

APEmille in Europe

• INFN has already funded apeNEXT per un totale di circa 10 Tflops in Italy to be installed at “la Sapienza”. More funds may come
• Germany and France are still contracting with their funding agencies
Plans

• Physics
  • LQCD of course (so many groups), see Lattice 2005
  • Turbulence (Fe)
  • Complex System (Rm)

• apeNEXT\(^2\)
  • Activity will continue
  • Intermediate 2-4 x machine?
  • 100TF project???

• QBIO
  • Protein (mis)folding
  • Drug docking
  • See QBIO archive @ LANL
Structure Explorer - 1BDD

Title: Staphylococcus Aureus Protein A, Immunoglobulin-Binding B Domain, NMR, Minimized Average Structure
Classification: Immunoglobulin-Binding Protein
Compound: Mol_Id: 1; Molecule: Staphylococcus Aureus Protein A; Chain: Null; Fragment: B Domain; Engineered: Yes
Exp. Method: NMR, Minimized Average Structure