Power monitoring on ARM-based HPC clusters

Experiences from young and old

Filippo Mantovani

June 7th, 2017
Outline of the talk

About the Mont-Blanc project

- Overall contributions of the project
- ARM-based platforms for scientific computing / HPC
- System software to operate ARM clusters

Experiences power monitoring ARM based platforms

- The theory we would like to have...
- ...Fixing and patching to have it
- Combining performance with power analysis using BSC tools

Student Cluster Competition: young minds in action

Next steps & conclusions
Mont-Blanc projects in a glance

Vision: to leverage the fast growing market of mobile technology for scientific computation, HPC and non-HPC workload.
Mont-Blanc contributions

ARM-based prototypes
- Mobile technology
- Server technology
- Custom design

System software
- Scientific libraries
- Performance analysis tools
- Support for runtimes
- Power monitor

Scientific applications
- Porting and benchmarking of mini-apps and full scale applications
- Scalability study on real ARM-based platforms

Next-generation studies
- big.LITTLE studies
- Limitation analysis
- Performance projections

Resiliency
- Application based fault tolerance
- Fault tolerance support in the runtime
- Reliability study of the Mont-Blanc prototype
Prototypes are critical to accelerate software development

System software stack + applications

The Mont-Blanc prototype ecosystem

PRACE prototypes
- Tibidabo
- Carma
- Pedraforca

Mini-clusters
- Arndale
- Odroid XU
- Odroid XU-3
- NVIDIA Jetson

Mont-Blanc prototype
- 1080 compute cards
- Fine grained power monitoring system
- Installed between Jan and May 2015
- Operational since May 2015 @ BSC

ARM 64-bit mini-clusters
- APM X-GENE2
- Cavium ThunderX
- NVIDIA TX1

Mont-Blanc 3 demonstrator
- Based on new-generation ARM 64-bit processors Cavium ThunderX2 SoC
- Targeting HPC market

Timeline:
- 2012
- 2013
- 2014
- 2015
- 2016
- 2017

Trondheim, June 7th, 2017
Mont-Blanc prototype

**Local Storage**
- microSD up to 64 GB

**Memory**
- 4 GB LPDDR3-1600

**Exynos5 Dual SoC**
- 2x cores ARM Cortex-A15
- 1x GPU ARM Mali-T604

**Network**
- USB3.0 to 1GbE bridge

Operational since May 2015 @ BSC

- 2 Racks
- 8 BullX chassis
- 72 Compute blades
- 1080 Compute cards
- 2160 CPUs
- 1080 GPUs
- 4.3 TB of DRAM
- 17.2 TB of Flash
Fundamental limitations

SoC level

- Low # of cores per socket
- Low amount of memory
  - 32-bit memory controller
  - Even if ARM Cortex-A15 offers 40-bit address space
- Double precision FP performance / vectorization
- Several interconnect but no classical HPC I/O interfaces
  - Do NOT provide native Ethernet or PCI Express
- No network protocol off-load engine
  - TCP/IP, OpenMX, USB protocol stacks run on the CPU

Integration level

- Integration process is still completely “HPC style”
  - Thermal studies are needed for a denser integration
- No ECC protection in memory
Vision

- Most of the limitations will evolve, eventually
  - In the original market of the devices
  - When extending to the server market
  - Pushed by other markets (e.g. automotive)

- Programming model and runtime will help “overcome”
  - Asynchrony and overlap
  - Resilience
  - Variability / Load balancing

- Tools can help understand the real problems and suggest/evaluate alternatives
  - e.g. correlating performance and power

Cavium Thunder cluster (from server market)

- Based on Cavium ThunderX SoC
  - Core: ARMv8 custom implementation
  - 48 cores @ 1.8 GHz per SoC

- 1 cluster node = dual socket board
  - 1 board, 2 sockets, 96 cores
  - 128 GB of DDR3 RAM
  - Cache coherency protocol implemented
  - One instance of Linux

- Cluster deployed at BSC facilities
  - 4x dual socket boards ( +1 )
  - 384 cores in 2U
  - ~700W peak power consumption*

* On a reference design board + PASS1 SoC

Provided by:

NTNU - EECS Seminar
Trondheim, June 7th, 2017
Jetson TX1 cluster (from mobile/embedded market)

- Same SoC of NVIDIA Shield console
  - 1x NVIDIA Tegra X1
    - 4x Cortex-A57 @ 1.73GHz
    - 1x Cortex-A53 (not usable)
  - 1x NVIDIA Maxwell GPU
    - 256 CUDA cores
- 4 GB LPDDR4
- 1GbE Network

- Cluster deployed at BSC facilities
  - 16x NVIDIA Jetson TX1 boards
  - Mont-Blanc software stack available
# System software stack for ARM

**Source files (C, C++, FORTRAN, Python, …)**

<table>
<thead>
<tr>
<th>Compilers</th>
<th>Scientific libraries</th>
<th>Developer tools</th>
<th>Cluster management</th>
<th>Runtime libraries</th>
<th>Hardware support / Storage</th>
<th>Linux OS / Ubuntu</th>
</tr>
</thead>
<tbody>
<tr>
<td>GNU</td>
<td>ATLAS</td>
<td>Extrae</td>
<td>Nagios</td>
<td>Nanos++</td>
<td>Power monitor</td>
<td>CPU</td>
</tr>
<tr>
<td>JDK</td>
<td>FFTW</td>
<td>Perf</td>
<td>Puppet</td>
<td>OpenCL</td>
<td>DVFS</td>
<td>GPU</td>
</tr>
<tr>
<td>Mercurium</td>
<td>HDF5</td>
<td>DDT</td>
<td>OpenLDAP</td>
<td>CUDA</td>
<td>NFS</td>
<td>Network</td>
</tr>
<tr>
<td></td>
<td>cIblas</td>
<td>Scalasca</td>
<td>Ganglia</td>
<td>MPI</td>
<td>Lustre</td>
<td></td>
</tr>
<tr>
<td></td>
<td>cIFFT</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Based on open-source packages**

**Tested on several ARM-based platform**

**But…**

- More than 10 prototypes
- More than 5 years
- More than 4 different ways of measuring the power…

...and still no standards!
Outline of the talk

→ About the Mont-Blanc project
  ▪ Overall contributions of the project
  ▪ ARM-based platforms for scientific computing / HPC
  ▪ System software to operate ARM clusters

Experiences power monitoring ARM based platforms
  ▪ The theory we would like to have...
  ▪ ...Fixing and patching to have it
  ▪ Combining performance with power analysis using BSC tools

→ Student Cluster Competition: young minds in action

→ Next steps & conclusions
Power monitoring approaches

- Monitor total power consumption of nodes
  - Coarse grained $\rightarrow O(s)$
  - Including the whole node power consumption
  - Out-of-band access (e.g. via IPMI, MQTT)
  - Mont-Blanc prototype, Cavium ThunderX + external power meter

- Monitor computational elements using on board devices
  - Medium granularity $\rightarrow O(ms)$
  - Including what the board producer decides to include
  - In-band access (e.g. via I2C) or out-of-band (with smarter BMCs)
  - JetsonTX

- Accessing power monitoring registers of the SoC
  - Fine grained $\rightarrow O(\text{cycles})$
  - Not including memory and accelerators
  - Requires standard tools/interfaces (RAPL / PAPI)
  - Currently not available in Mont-Blanc ARM-based platforms
    - Mostly political restrictions, i.e. SoC producers not sharing this info
Power monitor on the Mont-Blanc prototype (1)

Credits: Axel Auweter, Daniele Tafani (LRZ)

NTNU - EECS Seminar
Trondheim, June 7th, 2017
Power monitor on the Mont-Blanc prototype (2)

Field Programmable Gate Array (FPGA)
- Collects power consumption data from all 15 power measurement
- Sampling interval: 70ms

Board Management Controller (BMC)
- Collects 1s averaged data from FPGA
- Stores measurement samples in FIFO

Mont-Blanc Pusher
- Collects measurement data from multiple BMCs using custom IPMI commands
- Forwards data using MQTT protocol through Collect Agent into key-value store

Credits: Axel Auweter, Daniele Tafani (LRZ)
What can we do with this?

Power profile Mont-Blanc

- 1 core @ 1.6 GHz
- 2 cores @ 1.6 GHz
- 1 core @ 0.8 GHz
- 2 cores @ 0.8 GHz
- 1 core @ 1.6 GHz + GPU
- 1 core @ 0.8 GHz + GPU

Static Idle Power > 5W
What can we do with this?

- Fine grained power monitor infrastructure…
  - …integrated with standard tools…
    - SLURM plugin for jobs energy accounting
    - Paraver for correlating performance and power consumption (we will see it later)
  - …for the development of energy aware scheduling policies at datacenter level

Credits: Nikola Rajovic
Experimental setup with external power monitor

- Not “platform specific”
  - Cavium ThunderX
- Full node measurements
  - Including PSU losses

- Serial Interface
  - 3 sample/sec
Jetson TX1: “old school” hacking...

› Voltage monitor on-board component
  § Texas Instruments INA3221
  § Connected via I2C
  § No support provided by NVIDIA
  § Hand-written support…

› Measurements validated with external setup

› So we are now able to get power traces on Jetson TX1
  § 😊 O(0.1 sec) granularity
  § 😞 In-band measurements, potential conflicts with application execution
Meeting BSC performance analysis tools

- **Extrae: binary instrumentation**
  - `.trace.sh you-binary` → Run your application and generate a trace
  - Traces are collections of timestamped events
  - In the trace are collected several events specified in a xml config file
    - Beginners like me mostly get PAPI counters

- **Paraver: graphical trace visualizer**
  - Post-mortem analysis
  - Allow analysis applying different semantics / filters / histograms

Can we correlate performance and power?
Correlating performance and power

Credits: Enrico Calore
Leaving the system free to decide...

Histogram of cycles per us (i.e. frequency)
Outline of the talk

⇒ About the Mont-Blanc project
  ▪ Overall contributions of the project
  ▪ ARM-based platforms for scientific computing / HPC
  ▪ System software to operate ARM clusters

⇒ Experiences power monitoring ARM based platforms
  ▪ The theory we would like to have...
  ▪ ...Fixing and patching to have it
  ▪ Combining performance with power analysis using BSC tools

Student Cluster Competition: young minds in action

⇒ Next steps & conclusions
Mont-Blanc is not only research...

⇒ 12 teams of 6 undergraduate students
  ▪ From all over the world
  ▪ At the largest supercomputing conference of Europe

⇒ 3 kW power budget

⇒ 3 applications + 2 benchmarks
  ▪ Some known in advance
  ▪ Some “secret” application
  ▪ Some coding challenge

⇒ 3 awards to win
  ▪ Highest HPL
  ▪ 1st, 2nd, 3rd overall places
  ▪ Fan favorite
Outline of the talk

➡️ About the Mont-Blanc project
  ▪ Overall contributions of the project
  ▪ ARM-based platforms for scientific computing / HPC
  ▪ System software to operate ARM clusters

➡️ Experiences power monitoring ARM based platforms
  ▪ The theory we would like to have...
  ▪ ...Fixing and patching to have it
  ▪ Combining performance with power analysis using BSC tools

➡️ Student Cluster Competition: young minds in action

➡️ Next steps & conclusions
Next steps

Short term:
- Deeper understanding of governors
- Implementing easy access to Energy to Solution and Energy Delay Product
- Liaising with companies for standardize access to power data
- Profiling power of “real” production codes

Ideally targeting three levels of power optimizations:

From the application
- Access to an energy register, PAPI style
- Possibility of easily powering on-off / change the frequency of cores

From the runtime (within Task Based Prog. Model e.g OmpSs)
- Direct access to the power registers
- Possibility of easily powering on-off cores (without kernel support)

From the outside
- Gather power data of larger systems “a la Mont-Blanc”
- Targeting power aware job scheduling
Conclusions

Highlight of Mont-Blanc activities have been presented

- Even with low-end hardware components it is possible to achieve decent performance in parallel computation
- Main-line of Mont-Blanc 3 activity is targeting high-end server market
- Still researching in cost-efficient platforms

3 ARM-based platforms for scientific computing have been introduced

- With focus on power monitoring
- There is still a long way for real power aware programming
  - Getting fine grained (RAPL style) + node level power measurements is key

Young minds need to be educated to power sensibility

“The secret is to win going as slowly as possible.”

Niki Lauda

montblanc-project.eu @MontBlanc_EU filippo.mantovani@bsc.es