Whole Systems Energy Transparency

Kerstin Eder

Trustworthy Systems Laboratory, University of Bristol

Verification and Validation for Safety in Robots, Bristol Robotics Laboratory

University of Bristol

Department of Computer Science

The Royal Academy of Engineering

Whole Systems Energy Transparency

More *power* to software developers!

Kerstin Eder

Trustworthy Systems Laboratory, University of Bristol
Verification and Validation for Safety in Robots, Bristol Robotics Laboratory

University of BRISTOL

Department of COMPUTER SCIENCE

The Royal Academy of Engineering

Overview

- Introduction and Motivation
  - Energy consumption of Computing
  - Software vs Hardware

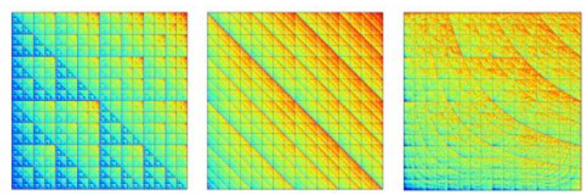
Overview

- Introduction and Motivation
  – Energy consumption of Computing
  – Software vs Hardware

Overview

- Introduction and Motivation
  - Energy consumption of Computing
  - Software vs Hardware
  - ***Energy Transparency***

- Measuring the energy consumption of software
- Energy modeling

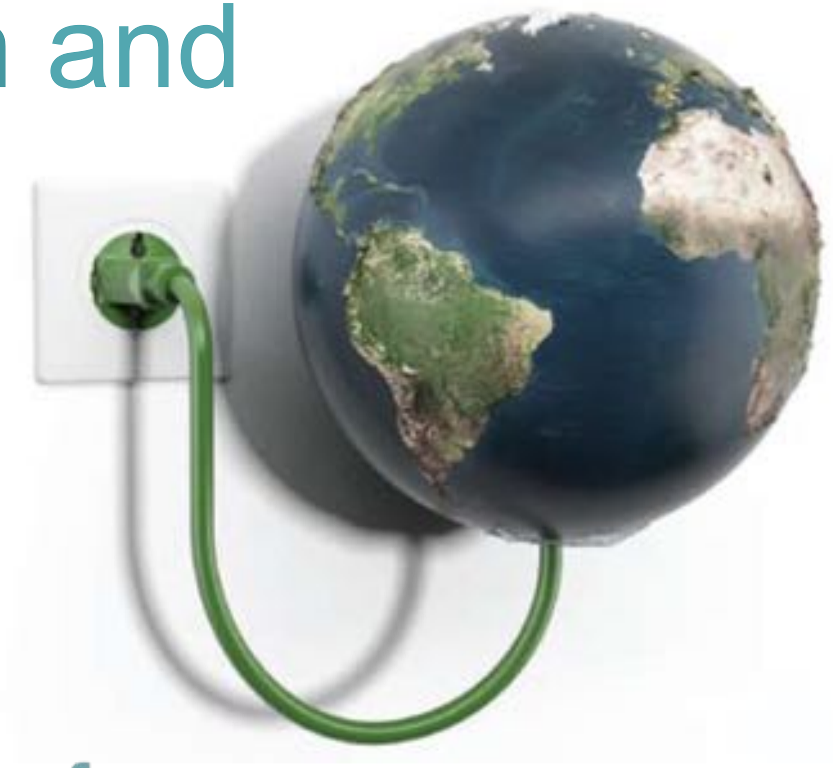


- Static energy consumption analysis of software
- Profile-based energy consumption prediction
- Research challenges

Learning Objectives

- Why software is key to energy efficient computing
- What energy transparency means and why we need energy transparency to achieve energy efficient computing
- How to measure the energy consumed by software
- How to estimate the energy consumed by software *without m*easuring
- How to construct energy consumption models
- Why timing and energy analysis differ

Introduction and Motivation

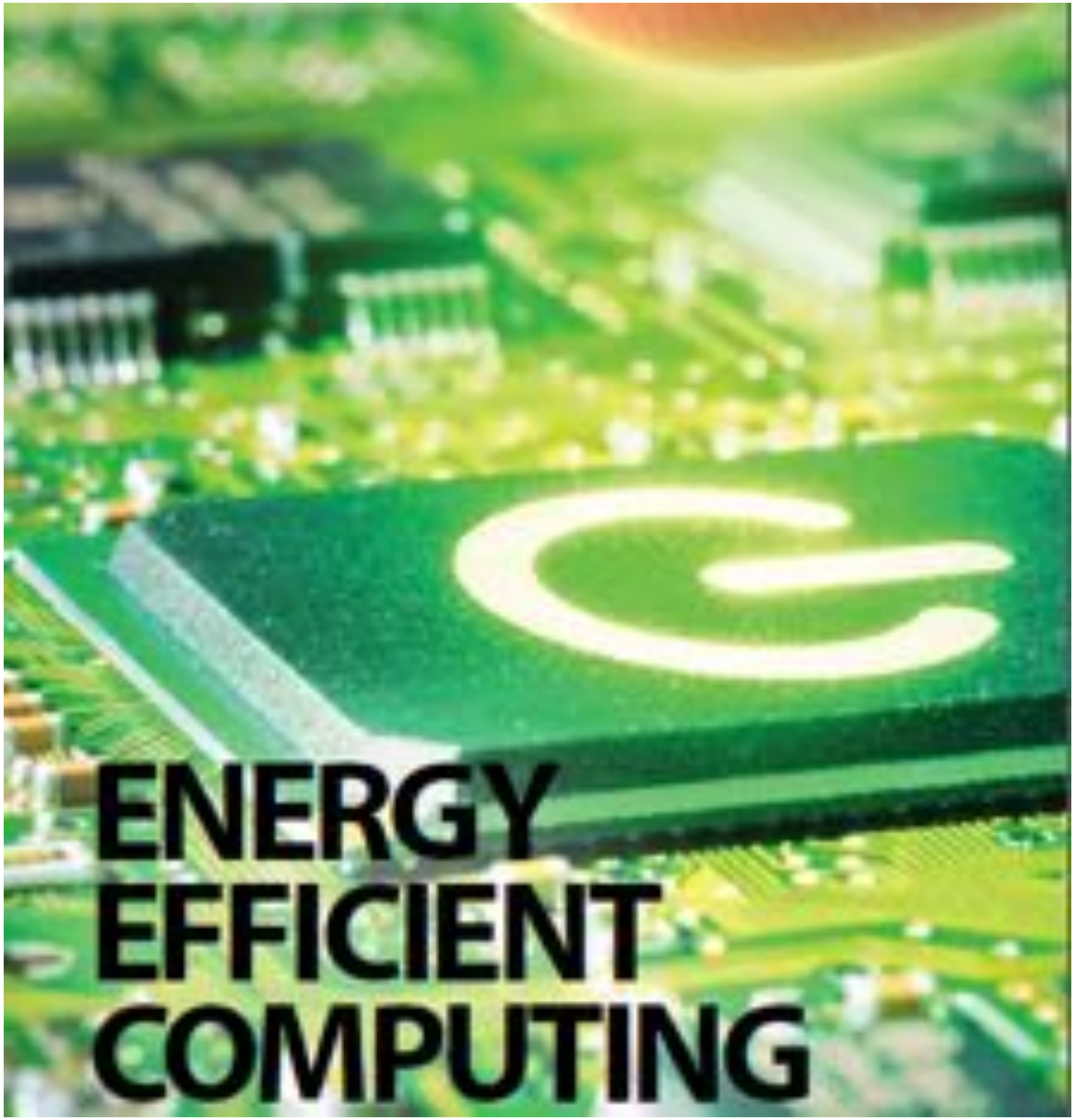


ICTenergy

http://www.ict-energy.eu/

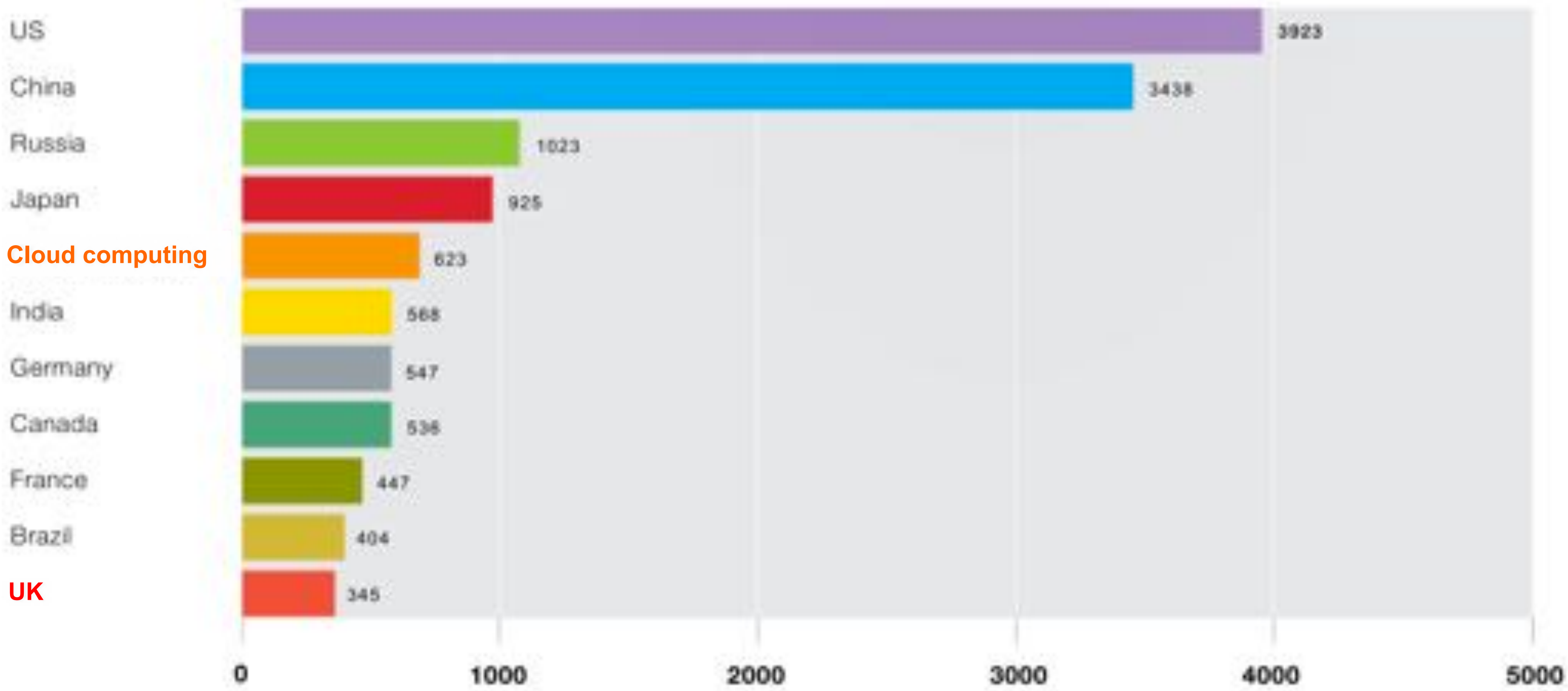
Pictures taken from the Energy Efficient Computing Brochure at:
https://connect.innovateuk.org/documents/3158891/9517074/Energy%20Efficient%20Computing%20Magazine?version=1.0

Pictures taken from the Energy Efficient Computing Brochure at:
https://connect.innovateuk.org/documents/3158891/9517074/Energy%20Efficient%20Computing%20Magazine?version=1.0



Pictures taken from the Energy Efficient Computing Brochure at:
https://connect.innovateuk.org/documents/3158891/9517074/Energy%20Efficient%20Computing%20Magazine?version=1.0

Electricity Consumption (Billion kwH, 2007)



Greenpeace: Make IT Green, March 2010
https://www.greenpeace.org/international/publication/7099/make-it-green-cloud-computing-and-its-contribution-to-climate-change/

Greenpeace: How clean is your cloud? April 2012
http://www.greenpeace.org/international/en/publications/Campaign-reports/Climate-Reports/How-Clean-is-Your-Cloud/

"Despite improved energy efficiency, **energy consumption through electronic devices will triple until 2030** because of a massive rise in overall demand."

Crowds in St. Peter's Square

2005

2013

BBC Mobile

News | Sport | Weather | iPlayer | TV |

NEWS TECHNOLOGY

Home World UK England N. Ireland Scotland Wales Business Politics Health Education Sci/Enviro

19 March 2012 Last updated at 17:34

1.7K Share

Free mobile apps 'drain battery faster'

Free mobile apps which use third-party services to display advertising consume considerably more battery life, a new study suggests.

Like many games, Angry Birds has a free version supported by targeted advertising

Researchers used a special tool to monitor energy use by several apps on Android and Windows Mobile handsets.

Findings suggested that in one case 75% of an app's energy consumption was spent on powering advertisements.

Report author Abhinav Pathak said app makers must take energy optimisation more seriously.

Related Stories

Hot mobile trends for

Energy Aware Computing

Energy Efficiency of ICT

arChitecture

Blocks

gAtes

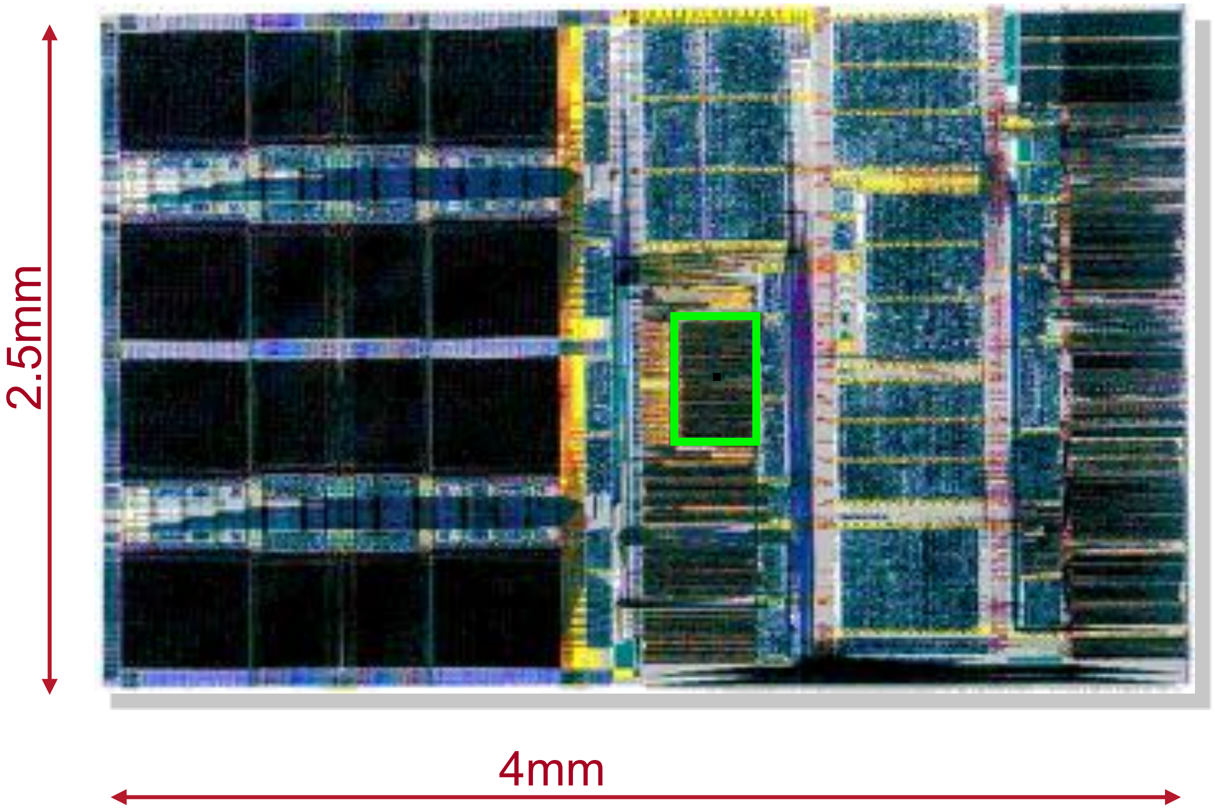
A historical perspective

(based on an inspiring talk by Steve Furber)

The Baby (1948)

- filled a medium-sized room
- executed 700 instructions per second

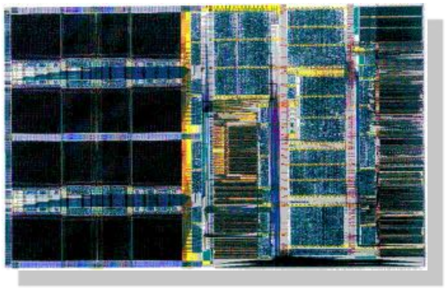
The ARM968 (2005)



- fills 0.4mm^2 of silicon
- executes 200,000,000 instructions per second
- ~300,000 times more than the Baby!

~60 years of progress

- **Baby, 1948:**
  – filled a medium-sized room
  – used 3.5 kW of electrical power
  – executed 700 instructions per second
- **ARM968, 2005:**
  – fills 0.4mm^2 of silicon (130nm)
  – uses 20 mW of electrical power
  – executes 200,000,000 instructions per second



Energy efficiency

- **Baby:**
  - 5 Joules per instruction
- **ARM968:**
  - 100 pico Joules per instruction

(James Prescott Joule born Salford, 1818)

Energy efficiency

- **Baby:**
  – 5 Joules per instruction
- **ARM968:**
  – 0.000 000 000 1 Joules per instruction

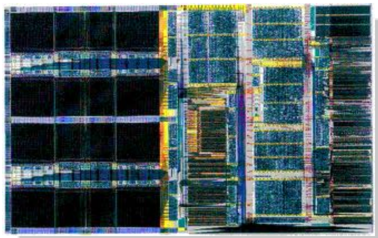
50,000,000,000 times better than Baby!

(James Prescott Joule born Salford, 1818)

10 more years of progress

- **Baby, 1948:**
  - filled a medium-sized room
  - used 3.5 kW of electrical power
  - executed 700 instructions per second

- **ARM968, 2005**
  - fills 0.4mm^2 of silicon (130nm)
  - uses 20 mW of electrical power
  - executes 200,000,000 instructions per second



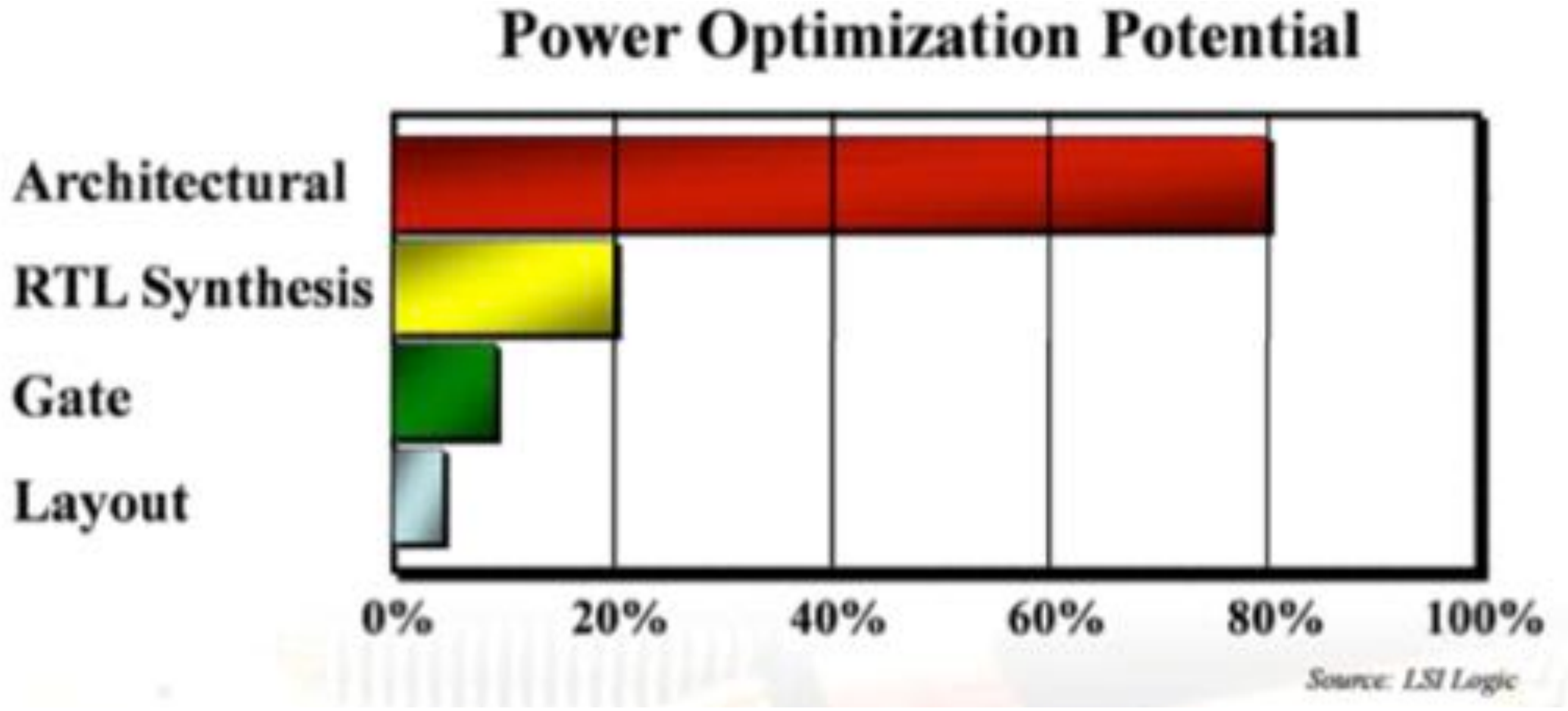
- **ARM Cortex-A35, 2015**
  - smallest area configuration <0.25mm^2
  - uses less than 4 mW of electrical power at 100 MHz
  - executes ~210,000,000 instructions per second

Hardware Design

- Power management largely in domain of Hardware Design
  - Considerations to minimize/optimize
    - Dynamic (switching) and static (leakage) power
  - On-chip power management
    - Modes: on, standby, suspend, sleep, off
- Development of low power electronics

Where can the greatest savings be made?

Greater Savings at Higher Levels



ElectronicsWeekly 15-21 June 2011 | No. 2472

LOW POWER

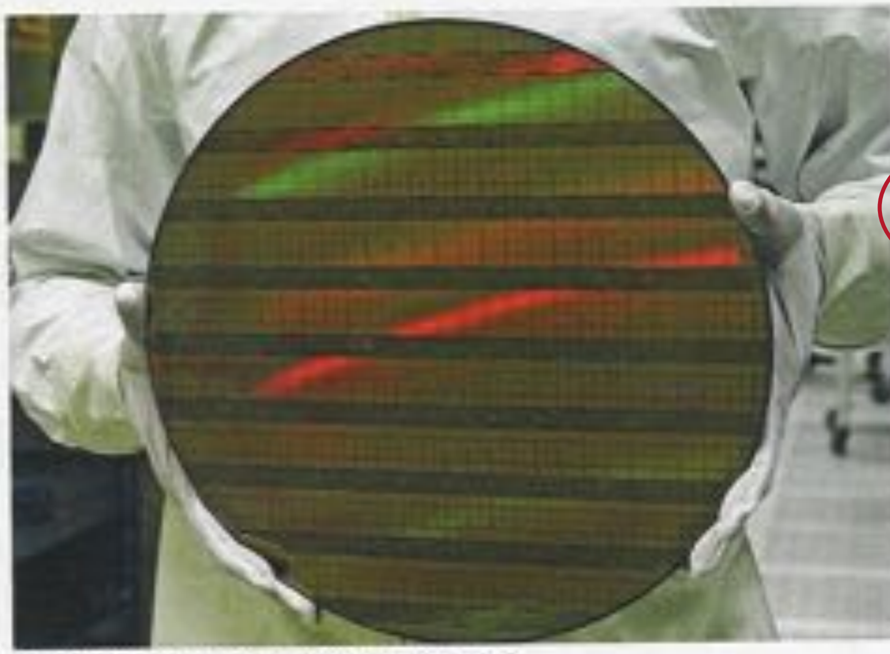
Lack of software support marks the low power scorecard at DAC

One of the panels at the Design Automation Conference (DAC), which took place in California in early June, set out to get an idea of how well the industry is doing at delivering lower-power systems.

It is becoming clear, *writes Chris Edwards*, that the system level is currently the missing link.

Processes can deliver some gains – and Globalfoundries' Andrew Brotman was able to outline some of the features that the foundry has put into its recently launched low-power high-k, metal gate (HKMG) process.

FinFETs should bring power down as those processes become available, although they are not the only options. But if the software keeps cores active for no good reason, the lower switching power per bit processed won't deliver a realised saving.



Intel waits for better low-power software control

In his keynote speech Gadi Singer, vice-president IAG and general manager of the SoC enabling group at Intel Corporation, said that with limited software support, dedicated low-power circuitry could save maybe 20% in a typical multimedia-oriented core.

Make the software controlling it better at controlling the power states and that difference could be three to five times.

During an afternoon panel discussion Ambrose Low, director of design engineering at Broadcom said: "We have hundreds of knobs in the hardware to turn power down.

"The question is whether we can take the actual use-cases into consideration and optimise the software to power the logic circuits down. We still have a long way to go."

Ruggero Castagnetti of LSI argued that the desire to do more in software will grow.

"As we see power limits and targets becoming unachievable, customers will be willing to go to that extra step. There is a challenge that needs to be addressed and we have to do more on the systems side," Castagnetti said.

"We should put a challenge to the software designers to see how much power they can save," he added.

Chris Edwards writes the Low-Power Design Blog (enabled by Mentor Graphics) on ElectronicsWeekly.com

www.electronicsweekly.com/ew-blogs/

Wasted Potential


news

Lack of software support marks the low power scorecard at DAC

Huge advances have been made in power-efficient hardware.

BUT – potential energy savings are wasted by

- software that does not exploit energy-saving features of hardware;
- poor dynamic management of tasks and resources.

Energy Efficiency of ICT

alGorithms

soFtware

compilErs

Drivers

arChitecture

Blocks

gAtes

http://static.datixinc.com/wp-content/uploads/2015/04/7.jpg

http://www.clker.com/cliparts/f/0/5/1/1298749934185335695circuit-board.jpg

The Focus is on Software

Lack of software support marks the low power scorecard at DAC

- Software controls the behaviour of the hardware
  - Algorithms and Data Flow
  - Compiler (optimizations)
    - Traditional SW design goals:

      performance,

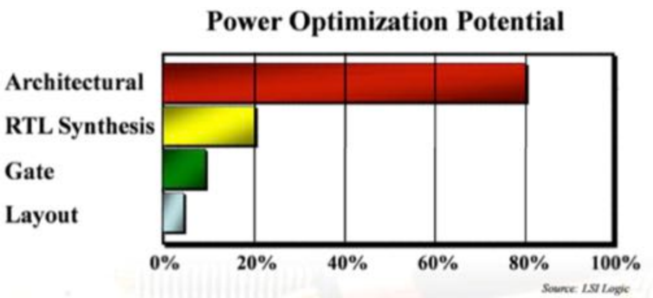
      performance,

      performance

The Focus is on Software


Lack of software support marks the low power scorecard at DAC

- Software engineers often **“blissfully unaware”**
  - Implications of algorithm/code/data on power/energy?
  - Power/Energy considerations
    - at best, secondary design goals
- BUT the **biggest savings** can be gained from optimizations at the higher levels of abstraction in the system stack
  - Algorithms,
  - Data and
  - SW



6.3. SOFTWARE DESIGN FOR LOW POWER

KAUSHIK ROY AND MARK C. JOHNSON
School of Electrical and Computer Engineering
Purdue University
West Lafayette, Indiana, U.S.A.

1. Introduction

It is tempting to suppose that only hardware dissipates power, not software. However, that would be analogous to postulating that only automobiles burn gasoline, not people. In microprocessor, micro-controller, and digital signal processor based systems, it is software that directs much of the activity of the hardware. Consequently, the software can have a substantial impact on the power dissipation of a system. Until recently, there were no efficient and accurate methods to estimate the overall effect of a software design on power dissipation. Without a power estimator there was no way to reliably optimize software to minimize power. Since 1993, a few researchers have begun to crack this problem. In this chapter, you will learn

Aligning SW Design Decisions with Energy Efficiency as Design Goal

Key steps*:

- *"Choose the **best algorithm** for the problem at hand and make sure it **fits well with** the computational **hardware**. Failure to do this can lead to costs far exceeding the benefit of more localized power optimizations.*
- *Minimize **memory size** and expensive **memory accesses** through algorithm transformations, efficient mapping of data into memory, and optimal use of memory bandwidth, registers and cache.*
- *Optimize the **performance** of the application, making **maximum use of available parallelism.***
- *Take advantage of **hardware support for power management**.*
- *Finally, select instructions, sequence them, and order operations in a way that **minimizes switching** in the CPU and datapath."*

* Kaushik Roy and Mark C. Johnson. 1997. "Software design for low power". In *Low power design in deep submicron electronics*, Wolfgang Nebel and Jean Mermet (Eds.). Kluwer Nato Advanced Science Institutes Series, Vol. 337. Kluwer Academic Publishers, Norwell, MA, USA, pp 433-460.

How much?

Picture from www.pd4pic.com

Energy Transparency

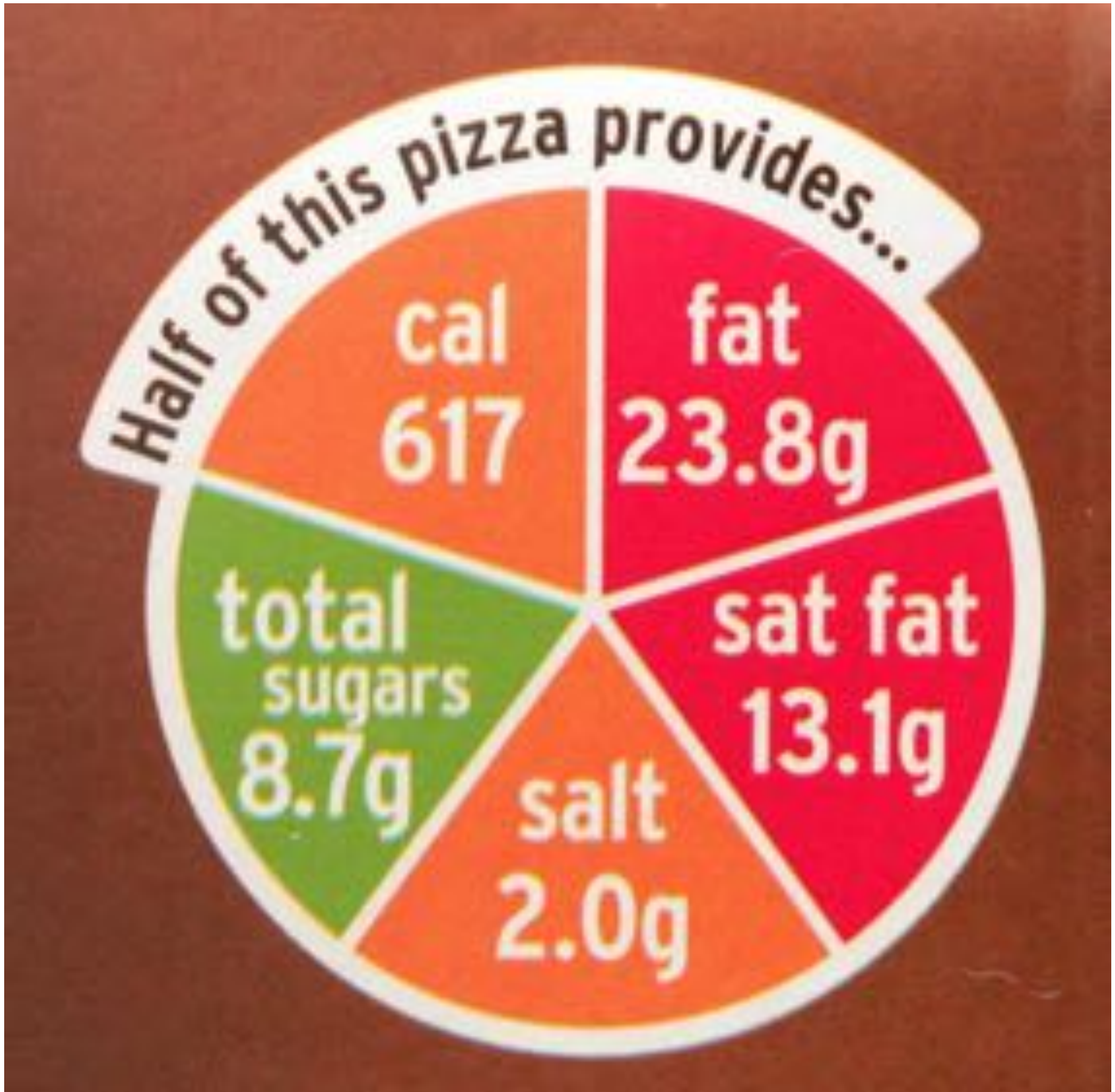
Energy Transparency

Information on energy usage is available for programs:

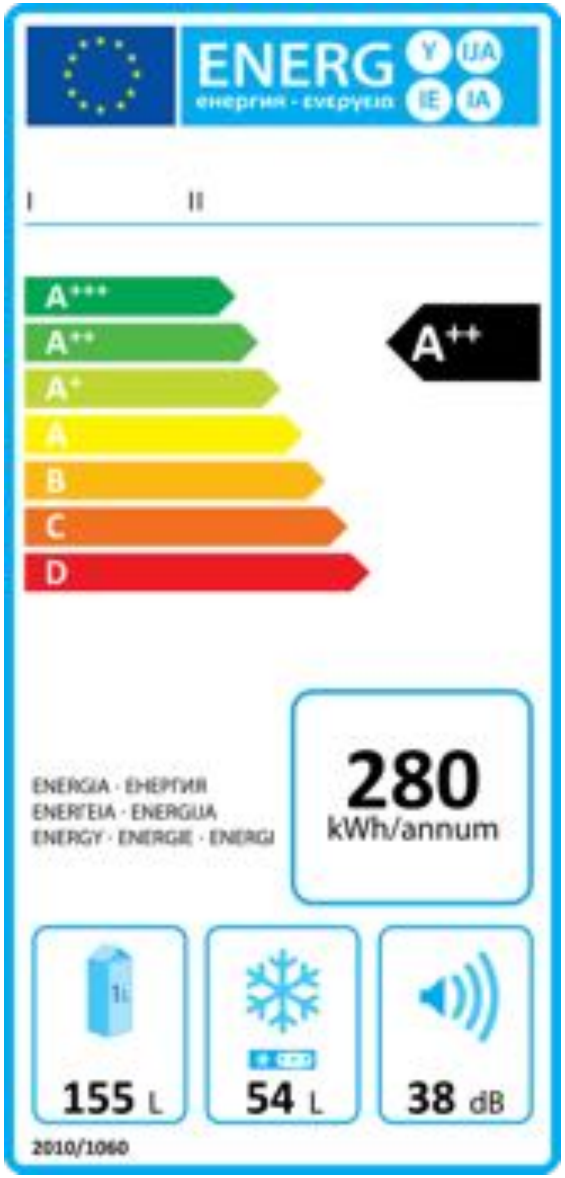
- ideally without executing them, and
- at all levels from machine code to high-level application code.

K. Eder, J.P. Gallagher, P. López-García, H. Muller, Z. Banković, K. Georgiou, R. Haemmerlé, M.V. Hermenegildo, B. Kafle, S. Kerrison, M. Kirkeby, M. Klemen, X. Li, U. Liqat, J. Morse, M. Rhiger, and M. Rosendahl. 2016. "ENTRA: Whole-systems energy transparency". *Microprocess. Microsyst.* 47, PB (November 2016), 278-286. https://doi.org/10.1016/j.micpro.2016.07.003

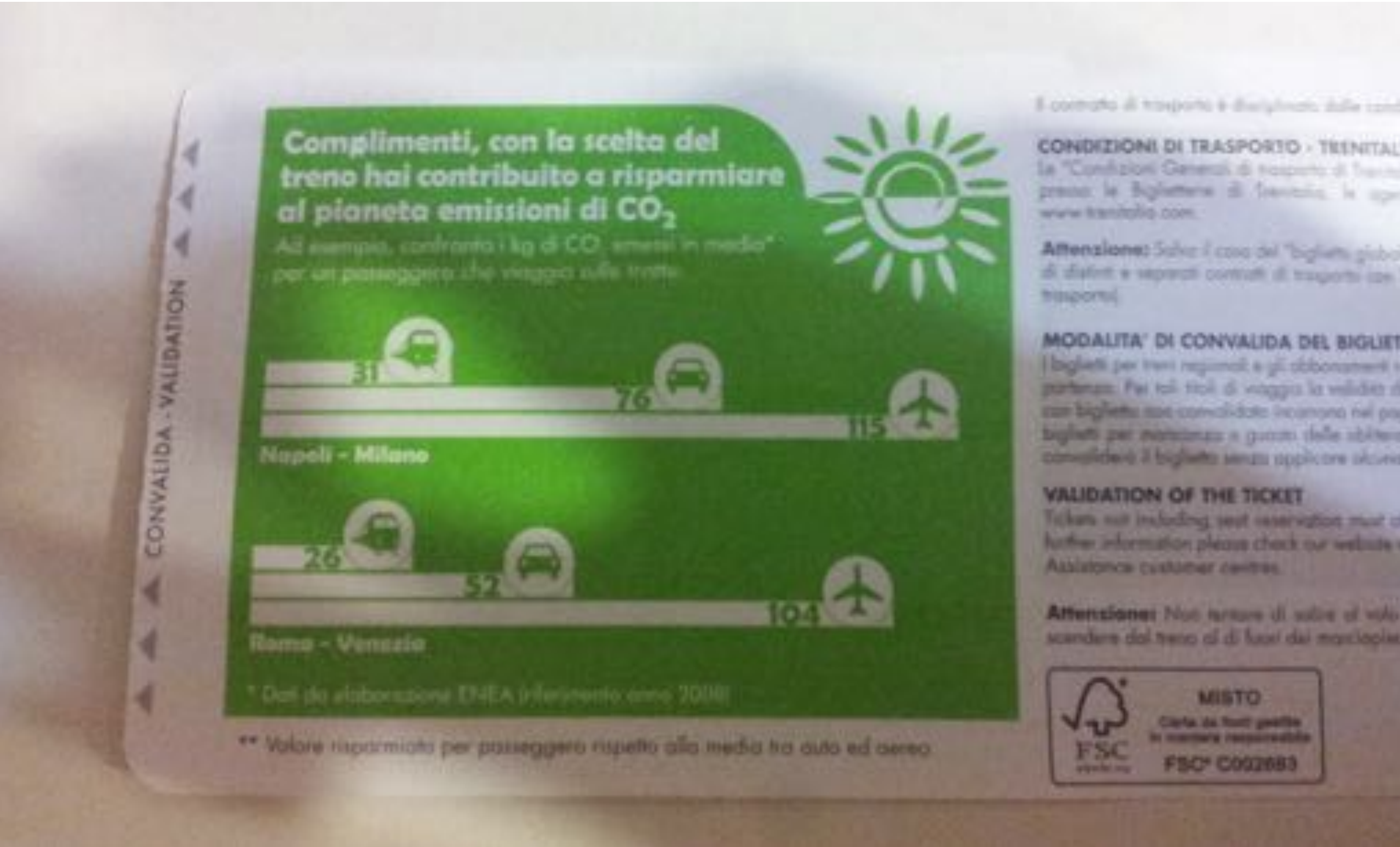
Transparency



Transparency



Transparency



Why Energy Transparency?

Energy transparency enables a deeper understanding of how algorithms and coding impact on the energy consumption of a computation when executed on hardware.

K. Eder, J.P. Gallagher, P. López-García, H. Muller, Z. Banković, K. Georgiou, R. Haemmerlé, M.V. Hermenegildo, B. Kafle, S. Kerrison, M. Kirkeby, M. Klemen, X. Li, U. Liqat, J. Morse, M. Rhiger, and M. Rosendahl. 2016. “ENTRA: Whole-systems energy transparency”. *Microprocess. Microsyst.* 47, PB (November 2016), 278-286. https://doi.org/10.1016/j.micpro.2016.07.003

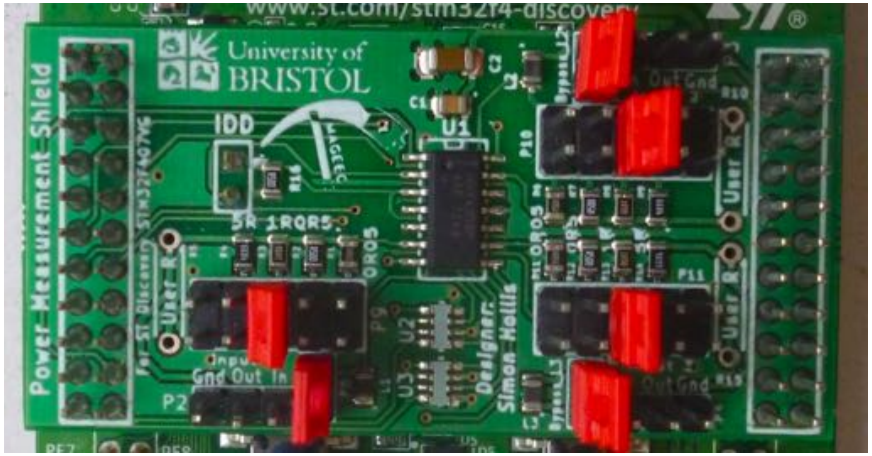
Learning Objectives

- ✓ Why software is key to energy efficient computing
- ✓ What energy transparency means and why we need energy transparency to achieve energy efficient computing
- How to measure the energy consumed by software
- How to estimate the energy consumed by software *without m*easuring
- How to construct energy consumption models
- Why timing and energy analysis differ

Learning Objectives

- ✓ Why software is key to energy efficient computing
- ✓ What energy transparency means and why we need energy transparency to achieve energy efficient computing
- How to measure the energy consumed by software
- How to estimate the energy consumed by software *without m*easuring
- How to construct energy consumption models
- Why timing and energy analysis differ

Measuring the Energy Consumption of Computation



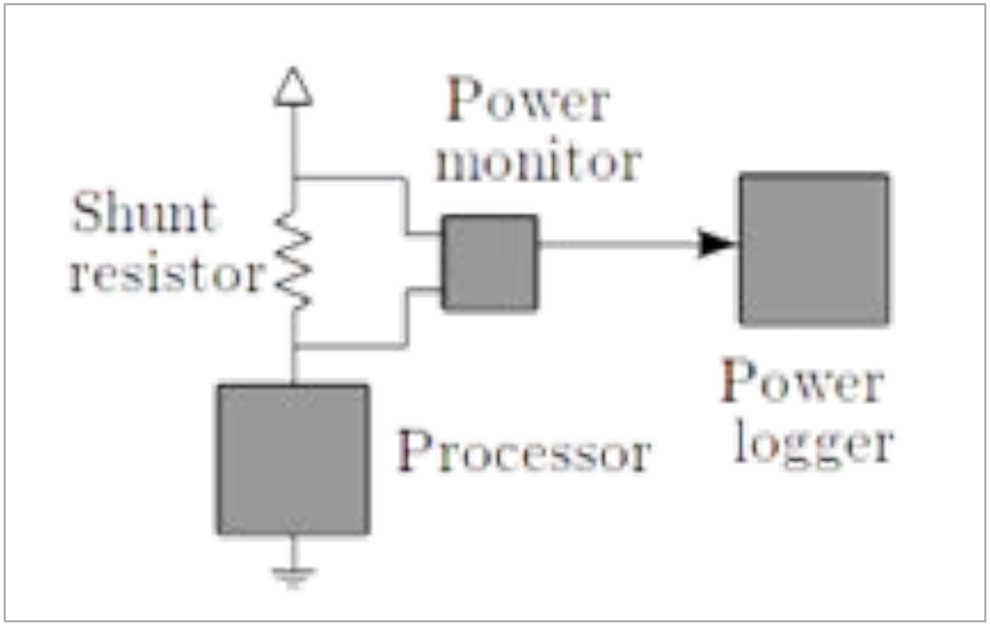
Measuring Power

Measure voltage drop across the resistor

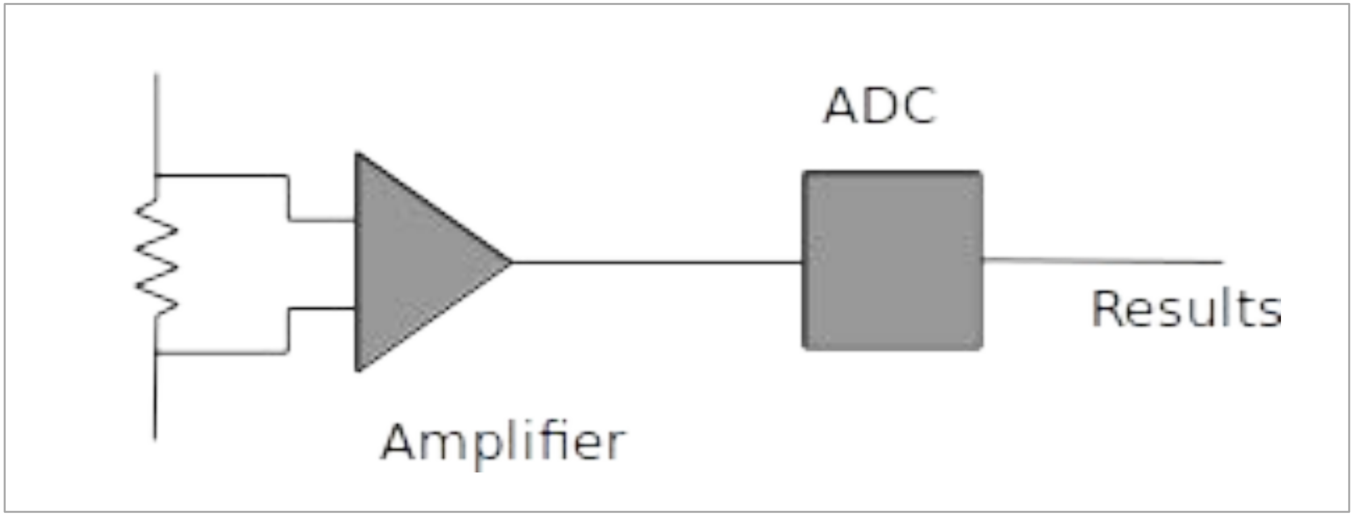
$I = V_{shunt} / R_{shunt}$ to find the current.

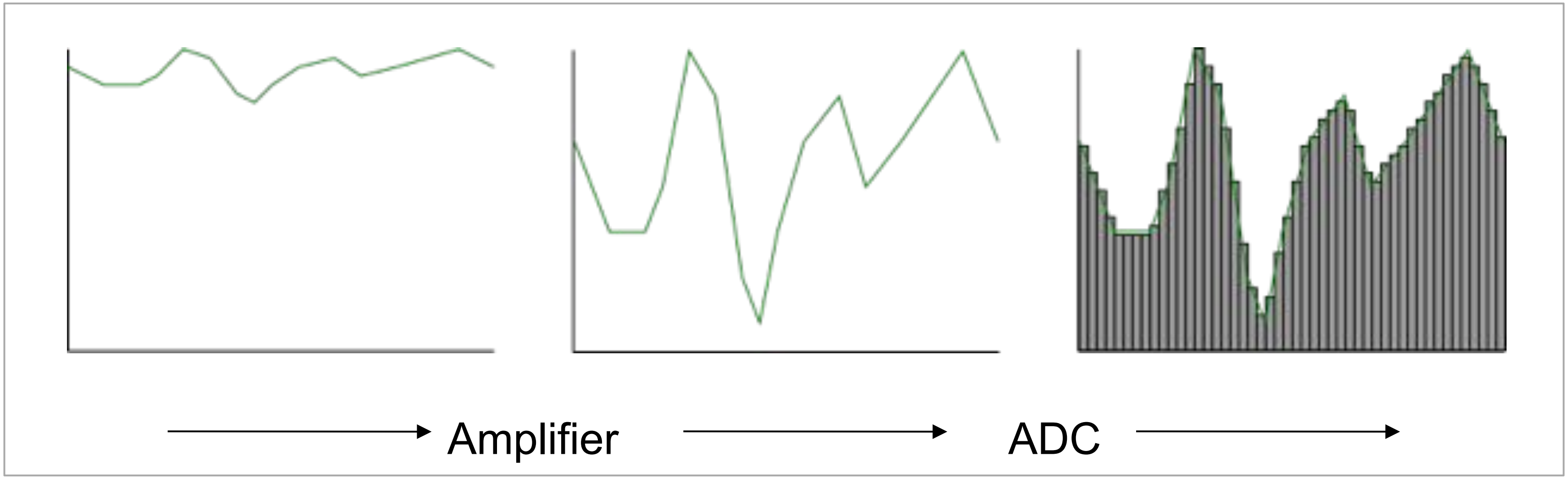
Measure voltage at one side of the resistor

$P = I \times V$ to calculate the power.



The Power Monitor



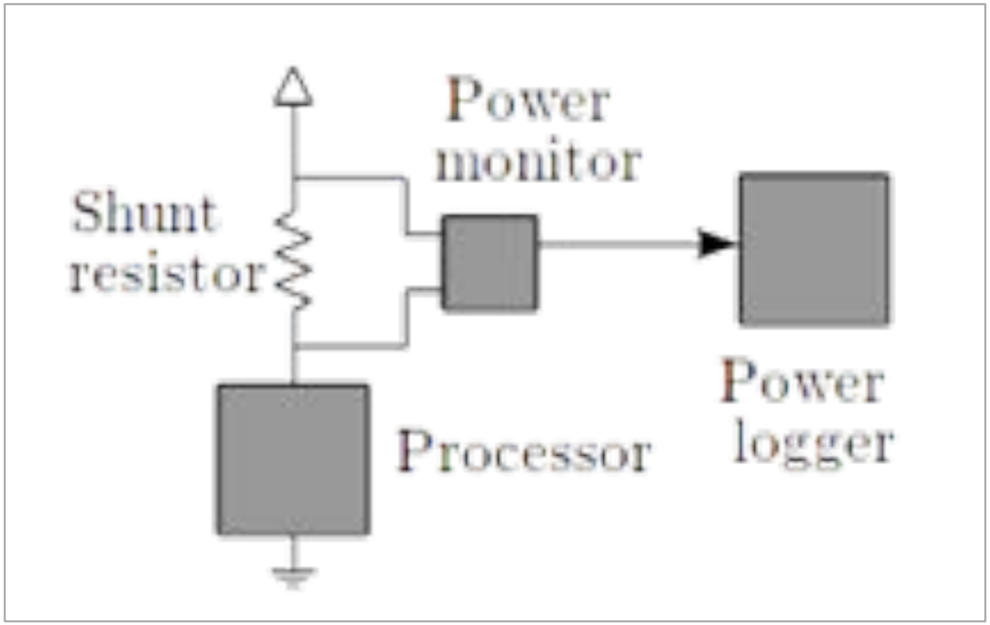


Measuring Power

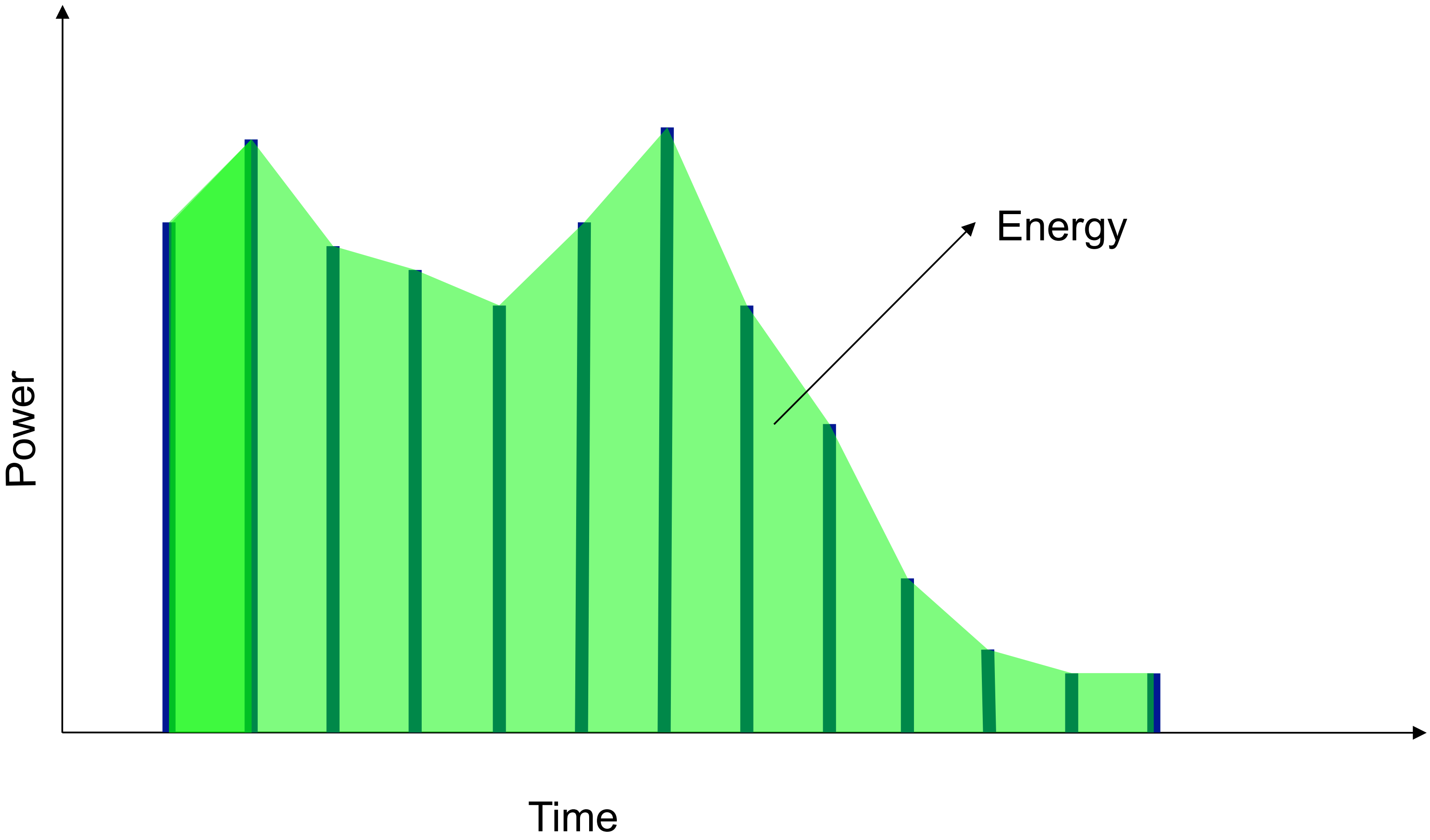
Repeat frequently, timestamp each sample

Measure voltage drop across the resistor — $I = V_{shunt} / R_{shunt}$ to find the current

Measure voltage at one side of the resistor — $P = I \times V$ to calculate the power



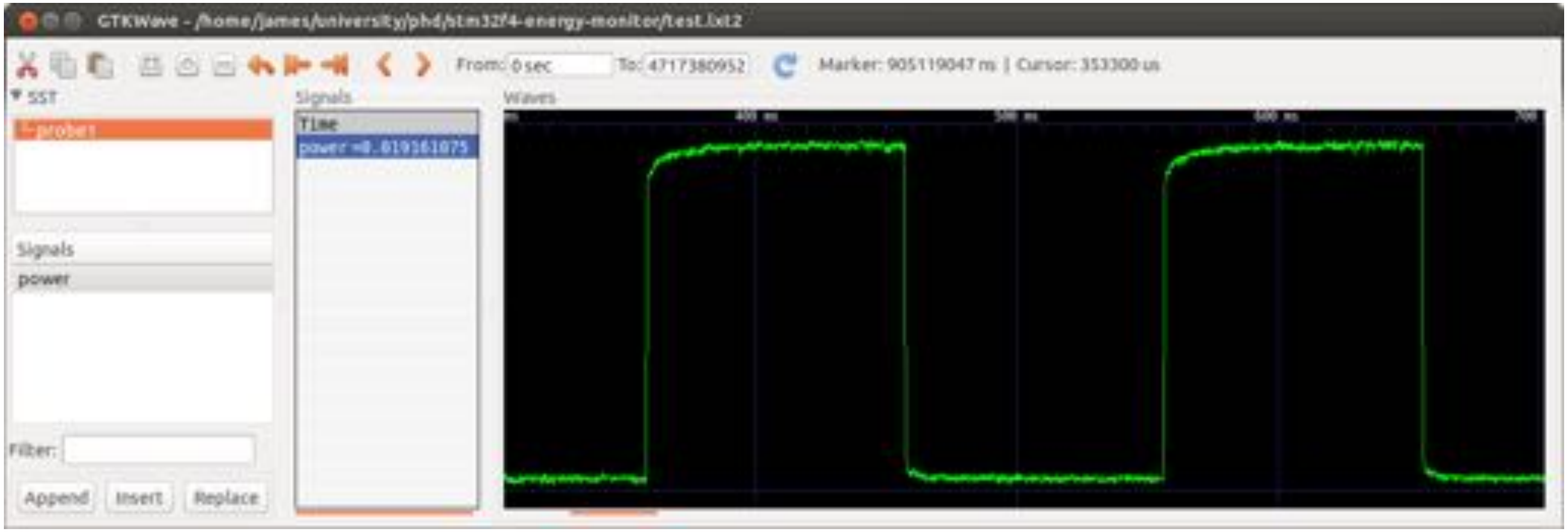
Measuring Energy



How much data?

Currently 500,000 Samples/second

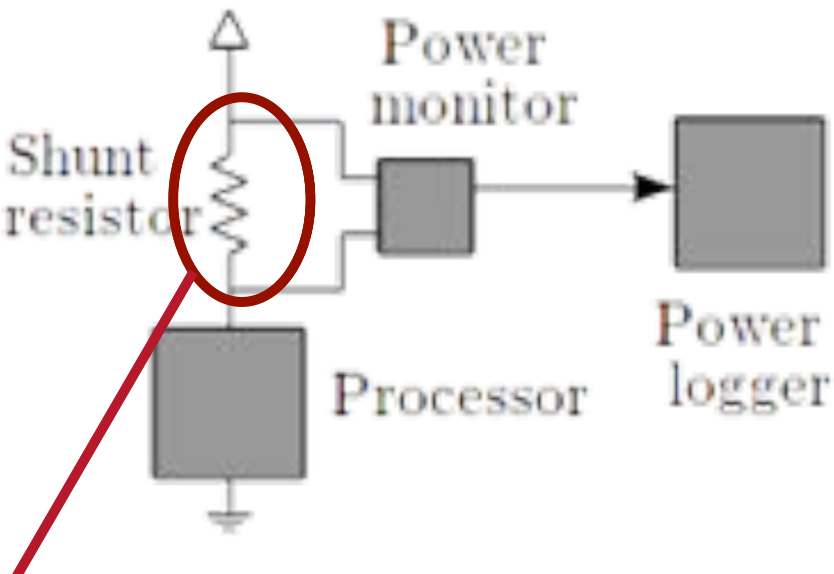
6,000,000 S/s possible in bursts



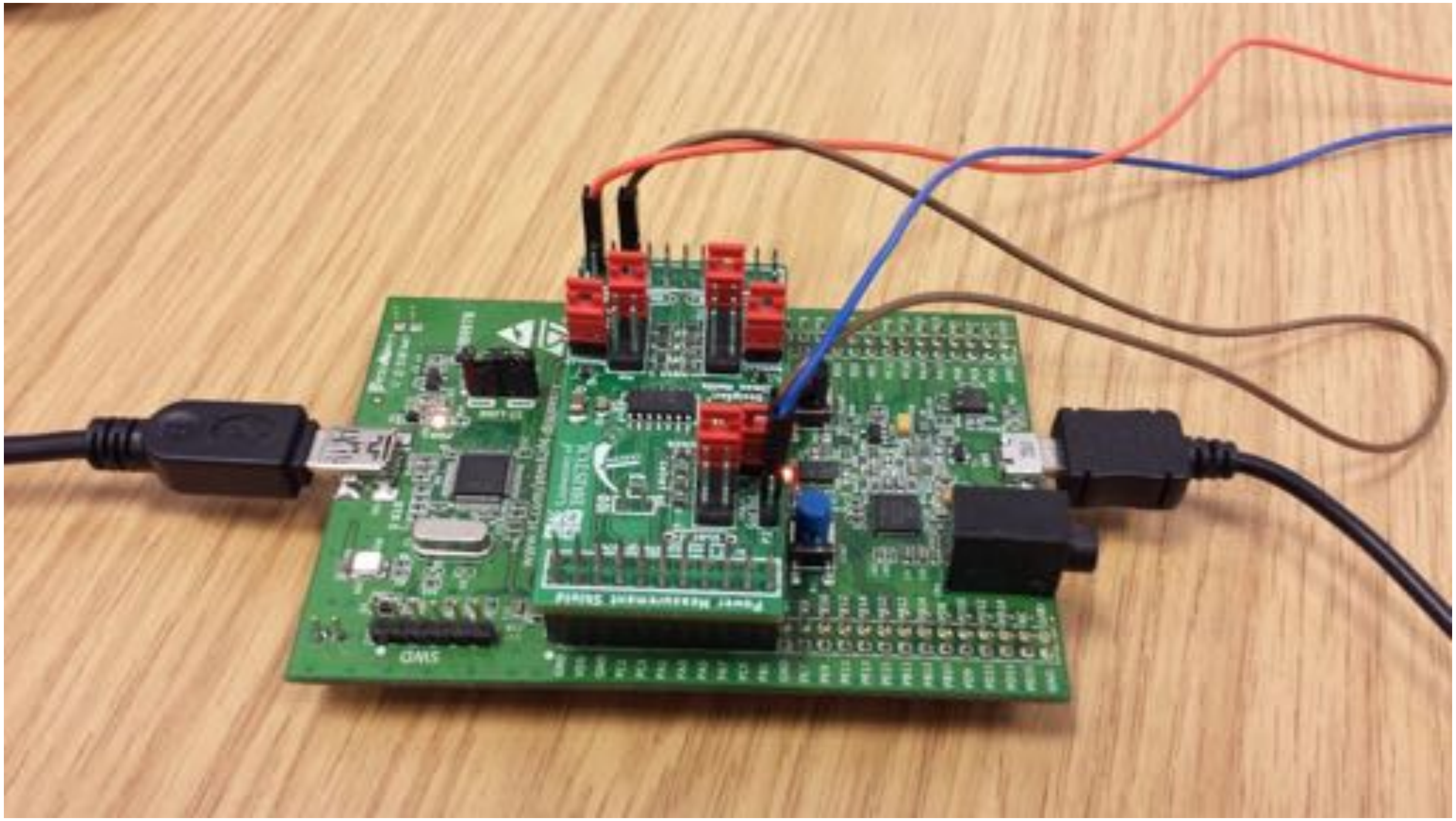
Summary: Energy Measurement

- We can *directly measure* the energy consumed during the execution of a program.
- The accuracy of the measurements depends on the sampling frequency, on the measuring hardware and on the characteristics of the target you want to measure.
- In many cases, specialized hardware and/or modifications to hardware are still required to enable energy measurement.

The Showstopper ☹

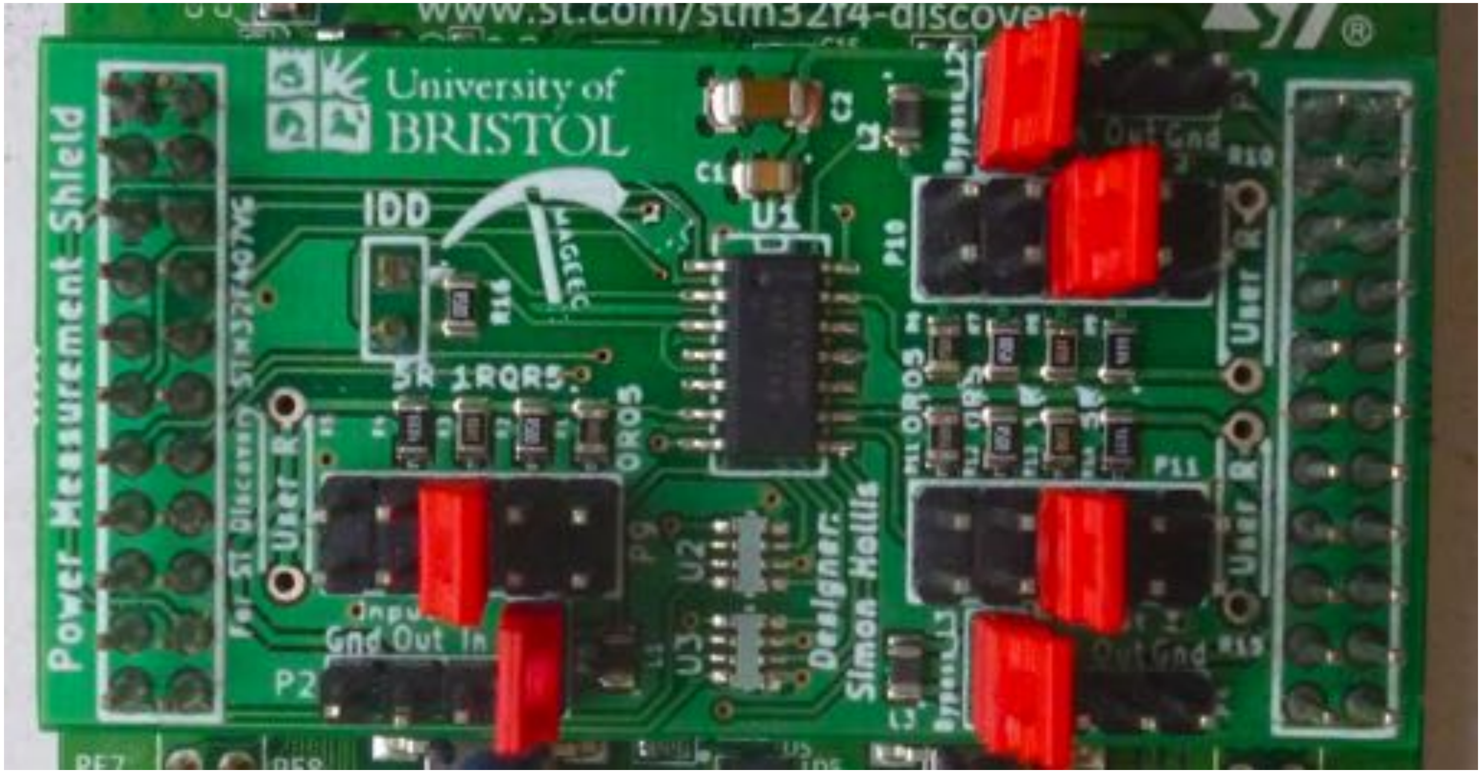


Open Energy Measurement Board



http://mageec.org/

Open Energy Measurement Board



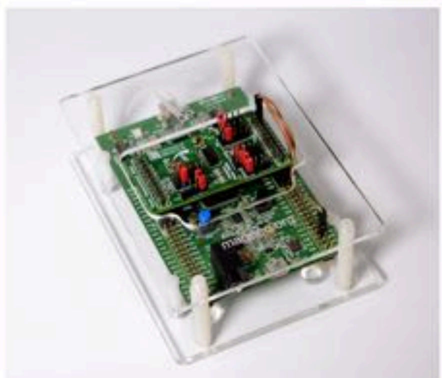
http://mageec.org/

Open Energy Measurement Board

Ground Electronics

Home Blog Catalog About Us

Home > Products > **MAGEEC Energy Measurement Kit**



MAGEEC Energy Measurement Kit

£43.50

Add to cart

The MAGEEC WAND is capable of measuring energy consumption at 3 independent points and with simultaneous measurement of targets at 2,000,000 samples/second.

The platform is comprised of an ARM Cortex M4-based STM32F4DISCOVERY board plus a custom shield, which is connected via USB to a host computer.

The shield, STM32F4DISCOVERY firmware, and a Python framework and applications, were developed as part of the MAGEEC project.

Hardware has been made available to members of the MAGEEC project, other research groups and as part of a workshop at FOSDEM 2014. Embecosm have funded the production of a limited number of kits which are now being made generally available at cost. There are no plans to produce any more once these are sold.

For further details, including a bill of materials, see the WAND Kit GitHub repository.

Like 1 Twe

http://groundelectronics.com/products/mageec-energy-measurement-kit

Dynamic Energy Monitoring for desktop applications

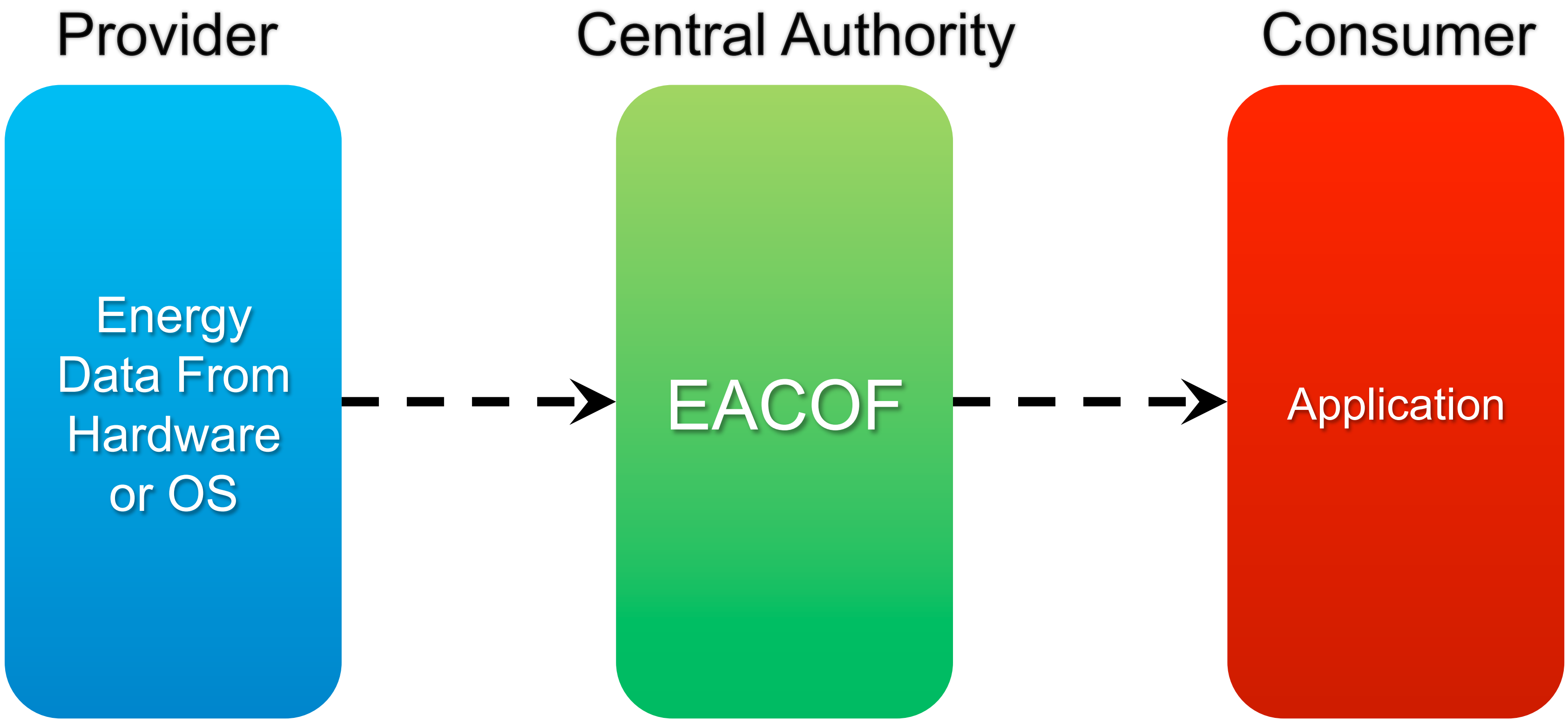
(using the Intel Power Gadget API)

The EACOF

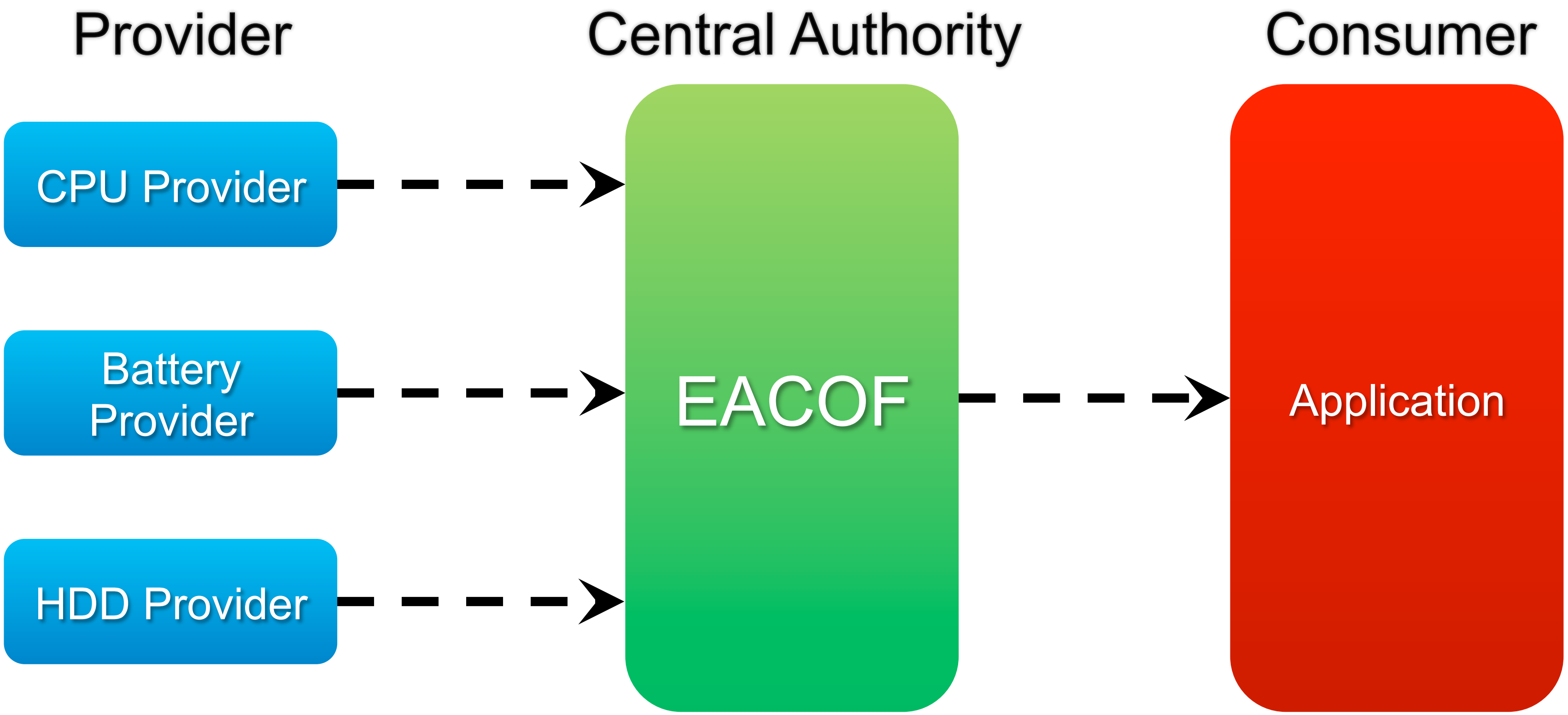
A simple Energy-Aware COmputing Framework

https://github.com/eacof

High Level

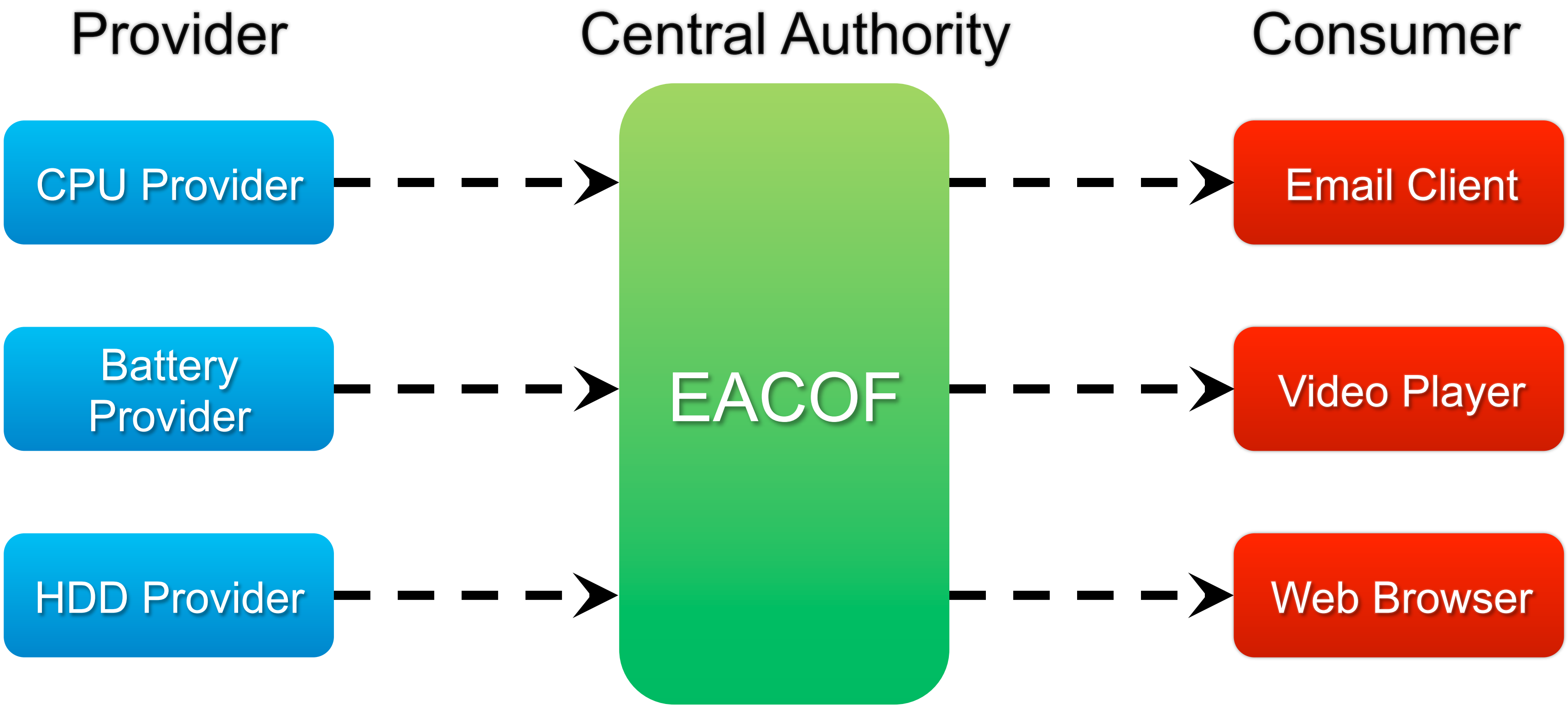

Provider
Central Authority
Consumer
Energy Data From Hardware or OS
EACOF
Application

Providers

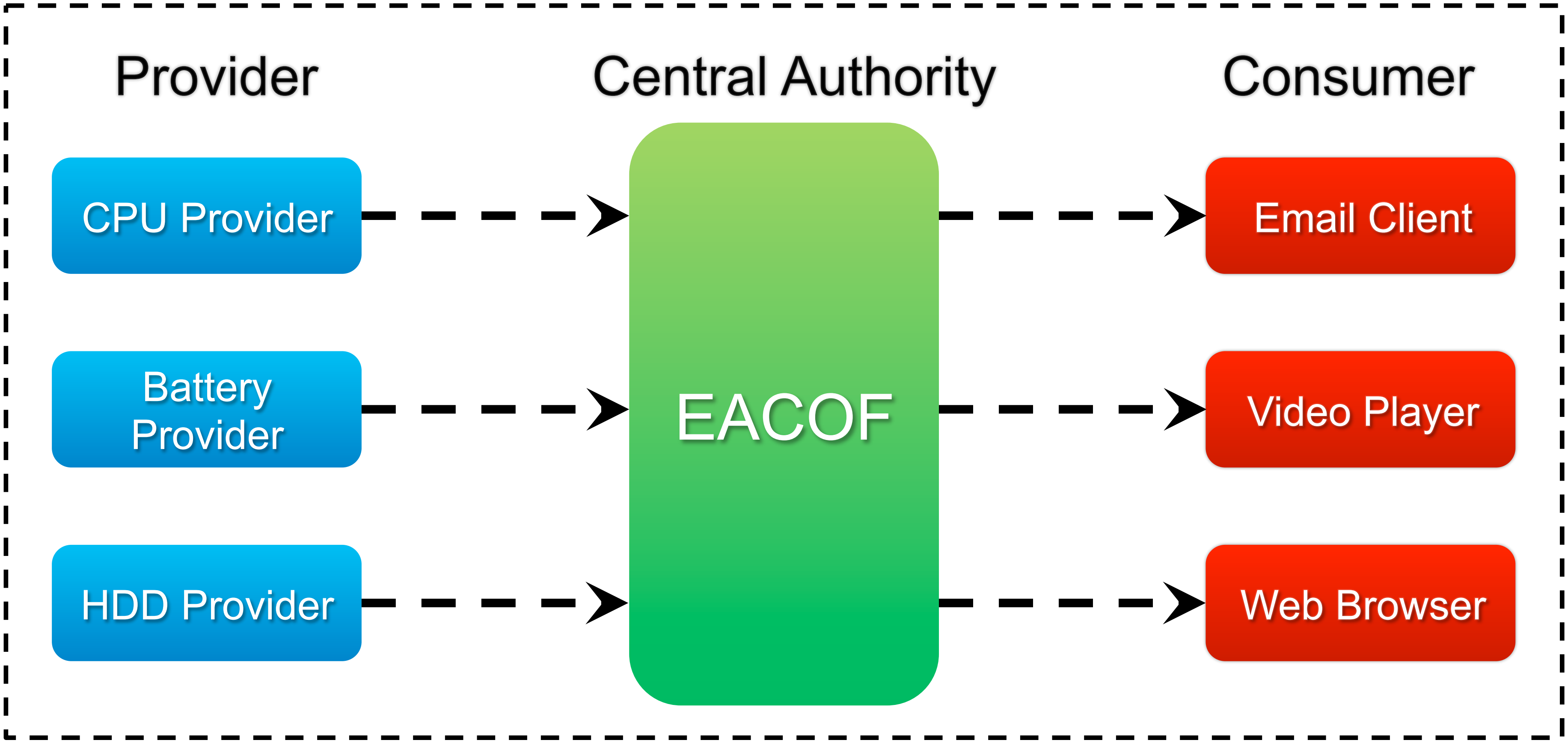


Provider
Central Authority
Consumer
CPU Provider
Battery Provider
HDD Provider
EACOF
Application

Consumers

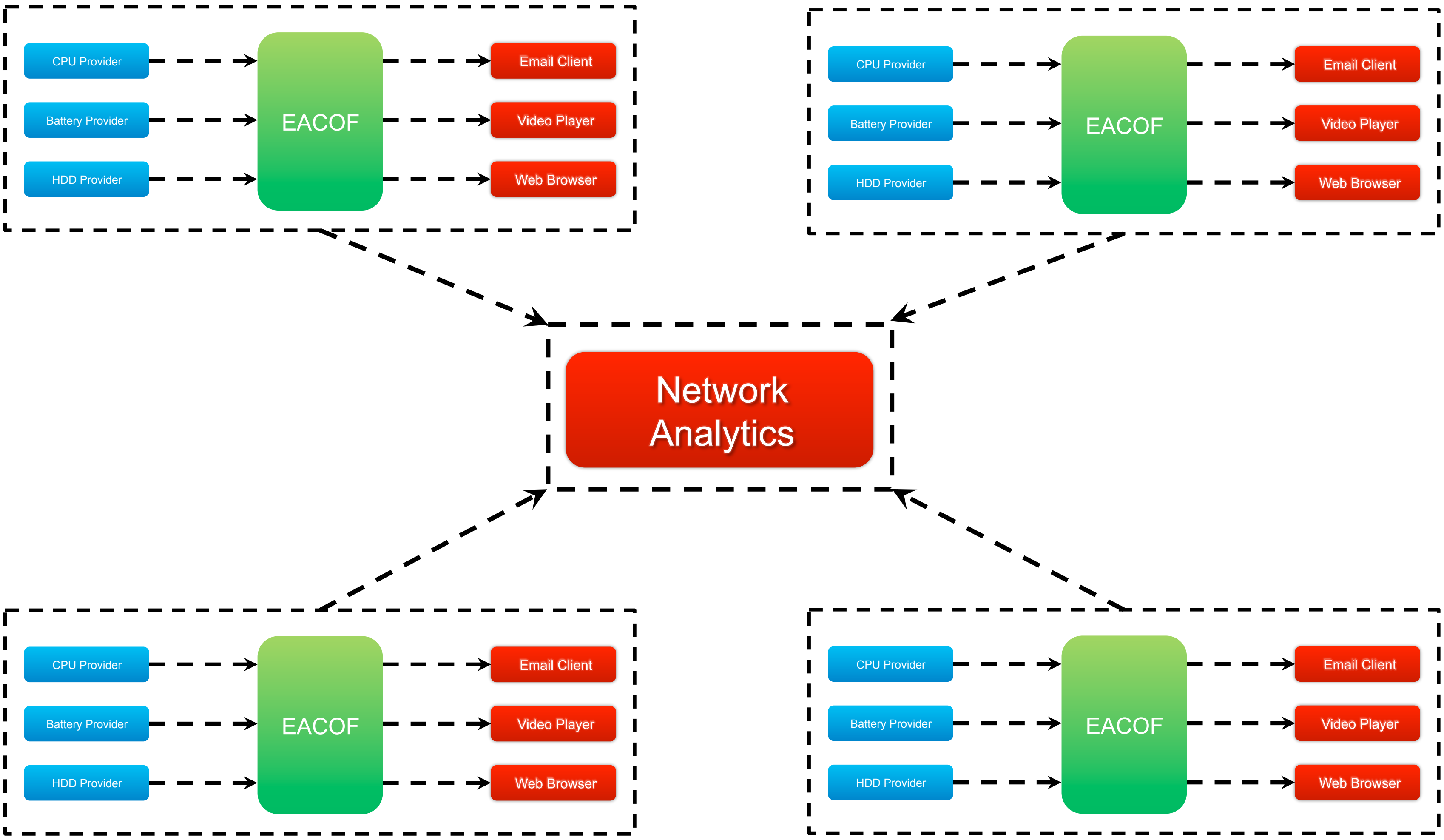

Provider
Central Authority
Consumer
CPU Provider
Battery Provider
HDD Provider
EACOF
Email Client
Video Player
Web Browser

One Machine



Provider
Central Authority
Consumer
CPU Provider
Battery Provider
HDD Provider
EACOF
Email Client
Video Player
Web Browser

Networked


CPU Provider
Battery Provider
HDD Provider
EACOF
Email Client
Video Player
Web Browser
CPU Provider
Battery Provider
HDD Provider
EACOF
Email Client
Video Player
Web Browser
Network Analytics
CPU Provider
Battery Provider
HDD Provider
EACOF
Email Client
Video Player
Web Browser
CPU Provider
Battery Provider
HDD Provider
EACOF
Email Client
Video Player
Web Browser

How to use EACOF

Simple Provider Example

```
while(1) {
  collectEnergyData();
  waitABit();
}
```

Simple Provider Example + EACOF

```
#include <eacof.h>
eacof_Probe *probe;
eacof_Sample sample;
initEACOF();
createProbe(&probe, 1, EACOF_DEVICE_BATTERY_ALL);
while(1) {
   sample = collectEnergyData();
   addSample(probe, sample);
   waitABit();
}
deleteProbe(&probe);
```

Simple Consumer Example

```
for (int i = 0; i < 10000; i++) {
   printf("Hello EACOF!");
}
```

Simple Consumer Example + EACOF

```
#include <eacof.h>
eacof_Checkpoint *checkpoint;
eacof_Sample sample;
initEACOF();
setCheckpoint(&checkpoint, EACOF_PSPEC_ALL, 1,
   EACOF_DEVICE_BATTERY_ALL);
for (int i = 0; i < 10000; i++) {
   printf("Hello EACOF!\n");
   sampleCheckpoint(checkpoint, &sample);
}
deleteCheckpoint(&checkpoint);
```

The EACOF API

```
#include <eacof.h>
initEACOF();
createProbe(); deleteProbe();

activateProbe(); deactivateProbe();

addSample();

setCheckpoint(); deleteCheckpoint();
sampleCheckpoint();
```

Comparing Sorting Algorithms

- Sorting of integers in [0,255]

		Data Type											
		uint8_t			uint16_t			uint32_t			uint64_t		
Algorithm	Num Elements	Total Time (s)	Total Energy (J)	Average Power (W)	Total Time (s)	Total Energy (J)	Average Power (W)	Total Time (s)	Total Energy (J)	Average Power (W)	Total Time (s)	Total Energy (J)	Average Power (W)
Bubble Sort	50,000	5.53	66.66	12.03	5.39	65.29	12.09	5.66	69.05	12.19	5.78	71.83	12.41
Insertion Sort	200,000	7.98	■102.18	12.75	7.98	■103.00	12.85	7.46	■98.81	13.21	7.54	■105.03	13.89
Quicksort	2,000,000	5.51	61.73	11.20	5.53	61.90	11.19	5.52	61.60	11.15	5.51	62.90	★11.42
Merge Sort	60,000,000	•6.06	•72.33	11.93	6.07	72.46	11.93	6.12	75.65	12.36	•5.93	•76.98	★12.98
qsort	100,000,000	•5.84	•72.39	12.37	6.15	76.90	12.48	6.79	86.29	12.69	•5.69	•73.25	12.86
Counting Sort	200,000,000	0.23	♦2.92	12.75	0.24	♦3.16	13.23	0.25	♦3.58	14.15	0.35	♦5.12	14.44

■ Insertion Sort: 32 bit version more optimized

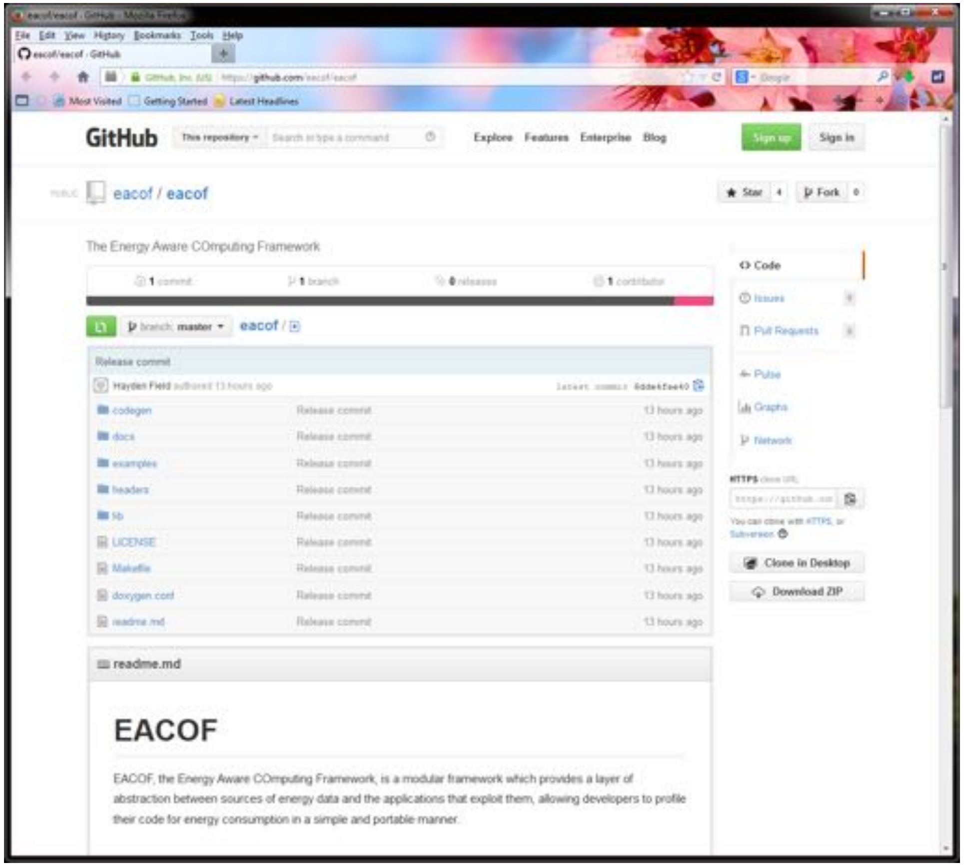
♦ Counting Sort:
75% more energy for 64 bit compared to 8 bit values

• Sorting 64 bit values takes less time than sorting 8 bit values, but consumed more energy

★ Average power variations between algorithms

H. Field, G. Anderson and K. Eder. "EACOF: A Framework for Providing Energy Transparency to enable Energy-Aware Software Development". *29th ACM Symposium On Applied Computing*. pp. 1194–1199. March 2014, ACM. DOI: 10.1145/2554850.2554920

Invitation: EACOF is open source!



github.com/eacof

Learning Objectives

- ✓ Why software is key to energy efficient computing
- ✓ What energy transparency means and why we need energy transparency to achieve energy efficient computing
- ✓ How to measure the energy consumed by software
- How to estimate the energy consumed by software *without m*easuring
- How to construct energy consumption models
- Why timing and energy analysis differ

Learning Objectives

- ✓ Why software is key to energy efficient computing
- ✓ What energy transparency means and why we need energy transparency to achieve energy efficient computing
- ✓ How to measure the energy consumed by software
- How to estimate the energy consumed by software *without* measuring
- How to construct energy consumption models
- Why timing and energy analysis differ

BREAK

(with the next two slides serving as screen cover during the break)

ICTenergy

http://www.ict-energy.eu/

University of BRISTOL

Energy Aware COmputing (EACO) research at the UNIVERSITY OF BRISTOL includes both Computer Science and Electronic Engineering, with significant cross-departmental expertise and collaboration in energy monitoring and modelling, static analysis and compilers, processor architectures and embedded multi-core system design.

The EACO Workshop series at the University of Bristol brings together academia and industry to identify and address intellectual challenges in Energy Aware Computing with the aim to reduce the energy consumption of computation. Topics of EACO Workshops span the entire system stack from application software and algorithms, via programming languages, compilers, operating systems, instruction sets and micro architectures to the design of hardware.

University of Bristol contact: **Kerstin Eder**

University of Glasgow

The UNIVERSITY OF GLASGOW's James Watt Nanofabrication Centre use micro- and nano-technology research and manufacturing facilities to develop technology including Terahertz optics and Silicon nano wires, healthcare applications and energy harvesting.

The Centre coordinates the Generate Renewable Energy Efficiently using Nanofabricated Silicon (GREEN Silicon) project, where the Seebeck effect is used to produce thermoelectric generators using Si/SiGe heterolayer technology, resulting in more efficient energy harvesting.

University of Glasgow contact: **Douglas Paul**

Tyndall

TYNDALL NATIONAL INSTITUTE is one of Europe's leading centres in ICT research and development. Applying an "atoms to systems" philosophy, energy research in Tyndall includes advanced concepts for low-power computing and efficient power supplies, energy storage and harvesting solutions, and technologies for wireless sensor networks applied to energy and resource optimisation in buildings and factories.

Tyndall coordinates a number of projects in the ICT Energy field including the NANOPOWER, SNAPS, SQWIRE, PowerSWIPE and DEEPEN projects.

Tyndall National Institute contact: **Giorgos Fagas**

Coordinating research efforts towards

LOW ENERGY ICT

The goal of the ICT-Energy project is to create a coordination activity among researchers working on energy reduction in ICT from Nanoscale Devices to Exascale Computing.

By bringing together the **Toward Zero-Power ICT** community with the **MINECC** (MINimizing Energy Consumption of Computing) community this project enables a concerted effort to lower energy consumption across the ICT sector.

Our aim is to assess the impact of existing research efforts and propose measures to increase the visibility of ICT-Energy related initiatives to the scientific community, targeted industries and to the public at large through the exchange of information, dedicated networking events, education and media campaigns.

coordinator: Prof. Luca Gammaitoni
NiPS Laboratory, Dipartimento di Fisica
Università di Perugia
Via A. Pascoli, 1 - 06123 Perugia, Italy
telephone: +39-075 585 2733
fax: +39-0755848458
email: luca.gammaitoni@nipslab.org

www.ict-energy.eu

The UNIVERSITY OF PERUGIA's Noise in Physical Systems (NiPS) Lab studies the effects of fluctuations in electrical fields, heat, sound and other mediums. This has led to the development of novel energy harvesting and noise sensing devices.

The NiPS Laboratory coordinates the LANDAUER project where the operation of basic physical switches below the Landauer limit is studied to investigate conceptually new devices and novel computing paradigms with radically improved power efficiency.

University of Perugia contact: **Luca Gammaitoni**

ROSKILDE UNIVERSITY's Programming, Logic and Intelligent Systems (PLIS) group focus on the theoretical aspects of programming languages and their applications. PLIS has significant expertise in software verification, program analysis and transformation.

The PLIS group coordinates the Whole Systems Energy Transparency (ENTRA) project where advanced program analysis and energy modelling techniques are used to predict the energy consumption of programs early on during software development. This enables energy-aware software engineering.

Roskilde contact: **John Gallagher**

The UNIVERSITY OF HEIDELBERG's Engineering Mathematics and Computing Lab (EMCL) applies numerical analysis to optimize the performance and energy consumption of High Performance Computing (HPC) as used in leading edge scientific programming.

The EMCL coordinates the EXA2GREEN project which aims to drastically reduce the energy consumed in HPC by developing advanced power consumption monitoring and profiling, and designing a smart, power-aware scheduling technology for HPC.

University of Heidelberg contact: **Vincent Heuveline**

HITACHI

At the HITACHI CAMBRIDGE LABORATORY (HCL) researchers investigate new designs of micro and optoelectronic devices, based on entirely new concepts, such as single electron logic circuits. Revolutionising the electronic devices used to power information technology has the potential to cut energy consumption by orders of magnitude.

HCL coordinates the Towards Low Power ICT (TOLOP) project which aims at the realization of novel low power devices (single electron transistors and single atom transistors), including implementation theory and the corresponding design architectures.

HCL contact: **David Williams**

Barcelona Supercomputing Center

BARCELONA SUPERCOMPUTING CENTER (BSC) uses HPC expertise to develop entirely new system architecture models for low-energy HPC.

The BSC coordinates the Parallel Distributed Infrastructure for Minimization of Energy (ParaDIME) project where radical software-hardware co-design techniques are being developed that are driven by future device characteristics on one side, and by a programming model based on message passing on the other side. This approach is expected to yield dramatic energy savings in heterogeneous distributed systems.

BSC contact: **Adrián Cristal Kestelman**

AALBORG UNIVERSITY

AALBORG UNIVERSITY's Center for Embedded Software Systems (CISS) improves embedded systems development through the use of model-driven design tools. These allow designs to be written in a verifiable way, and analysed for energy consumption and performance.

The CISS coordinates the Self Energy-Supporting Autonomous Computation (SENSATION) project which aims at increasing the scale of systems that are self-supporting by balancing energy harvesting and consumption. The research addresses the challenge of programming systems that reconfigure themselves in response to changing tasks, resources, errors and available energy.

Aalborg University contact: **Kim Guldstrand Larsen**

EPFL

ECOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE (EPFL) specialises in embedded and low-power systems, efficiently designed software algorithms and system level optimisation.

EPFL coordinates the PHIDIAS project which proposes the development of an ultra-low power smart bio-sensing wireless body sensor network, making use of new signal processing models and methods for efficient data handling. This enables long term low energy monitoring of bio-signals.

EPFL contact: **Pierre Vandergheynst**

www.ict-energy.eu

If you want an ultimate low-power system, then you have to worry about *energy usage at every level in the system design*, and you have to get it right from top to bottom, because any level at which you get it wrong is going to lose you perhaps an order of magnitude in terms of power efficiency.

The hardware technology has a first-order impact on the power efficiency of the system, but you've also got to have software at the top that avoids waste wherever it can. You need to avoid, for instance, anything that resembles a polling loop because that's just burning power to do nothing.

I think one of the hard questions is whether you can pass the responsibility for the software efficiency right back to the programmer.

Do programmers really have any understanding of how much energy their algorithms consume?

I work in a computer science department, and it's not clear to me that we teach the students much about how long their algorithms take to execute, let alone how much energy they consume in the course of executing and how you go about optimizing an algorithm for its energy consumption.

Some of the responsibility for that will probably get pushed down into compilers, but I still think that fundamentally, at the top level, **programmers will not be able to afford to be ignorant about the energy cost of the programs they write.**

What you need in order to be able to work in this way at all is instrumentation that tells you that running this algorithm has this kind of energy cost and running that algorithm has that kind of energy cost.

You need tools that give you feedback and tell you how good your decisions are.

Currently the tools don't give you that kind of feedback.

February 2010, acmqueue Interview with Steve Furber
The designer of the ARM chip shares lessons on energy-efficient computing at: http://queue.acm.org/detail.cfm?id=1716385

Steve Furber