

Energy-Aware Computing Systems

Energiebewusste Rechensysteme

I. Introduction

Timo Hönig

2018-10-17



EASY



©thoenig

EASY (WS 2018, Lecture 1)

Preface

2-23

Agenda

Preface

Motivation

Contents

Organization

Summary

Energy-Aware Computing Systems

meaning of the lecture labelling in linguistic terms:

en·er·gy (gr.) *energeia*: word based upon *ergon*, meaning *work*

1. capacity for the exertion of power
2. a fundamental entity of nature that is transferred between parts of a system in the production of physical change within the system

aware (old en.) *gewær*

1. having or showing realization, perception, or knowledge
2. state of being conscious of something

com·put·ing (lat.) *computare*: *com* (together) + *putare* (to settle)

1. task of making a calculation
2. to use a computer

sys·tems plural of (gr.) *systemas*: to place together

1. a regularly interacting or interdependent group of items forming a unified whole
2. a group of devices (...) or an organization forming a network especially for distributing something or serving a common purpose



©thoenig

EASY (WS 2018, Lecture 1)

Preface

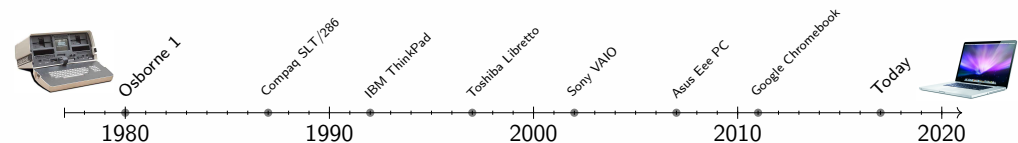
3-23



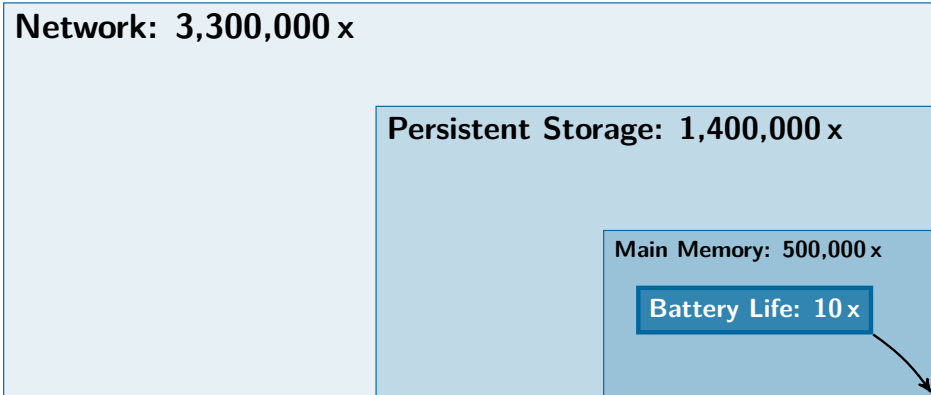
1980s



2010s



Technological Progress in Recent Decades



Transmission speed improved by a factor of approx. **3.3 million**

➡ 300 bit/s vs. 1 gigabit/s

Storage capacity increased by a factor of approx. **1.4 million**

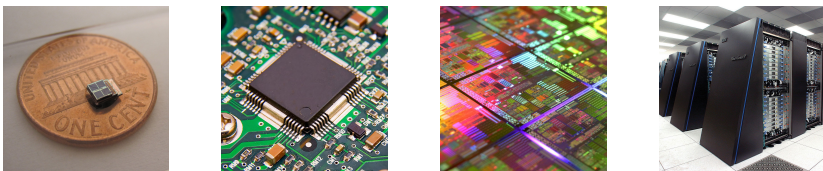
➡ 360 KiB vs. 500 GiB

Memory capacity improved by a factor of approx. **0.5 million**

➡ 4 KiB vs. 2 GiB

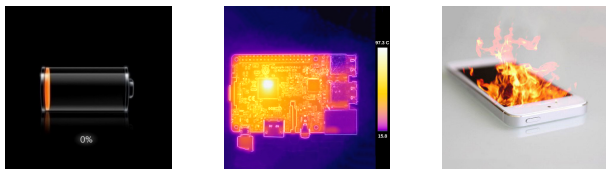
Electrical Energy: Basic Operating Resource

- electrical energy is *the* basic operating resource of today's computers

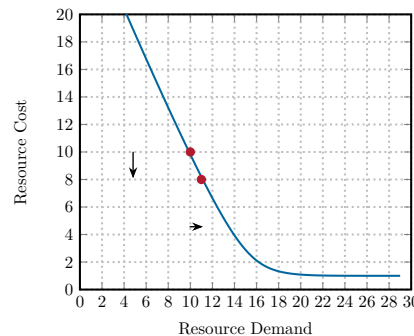


embedded — laptop/desktop — cluster

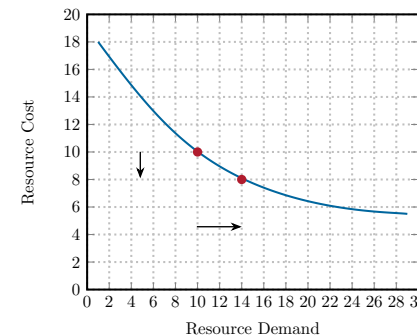
- **but:** excessive power dissipation leads to uncontrollable situations



Jevons Paradox

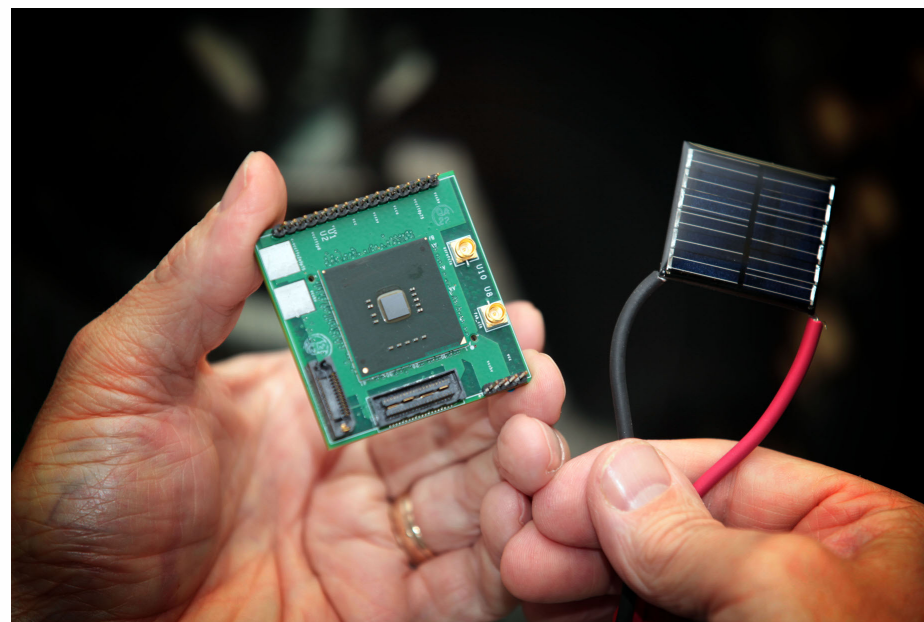


increase efficiency by 20 %
 ⇒ increase demand by 10 %



increase efficiency by 20 %
 ⇒ increase demand by 40 %

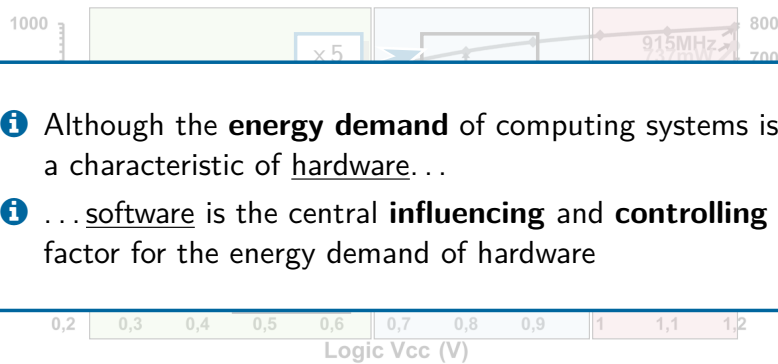
- improve efficiency by reducing costs
- Jevons paradox: efficiency gain ⇒ increase of demand
- rebound effect: increase of demand outweighs efficiency gain



► Shailendra Jain, Surhud Khare, Satish Yada et al.
A 280mV-to-1.2V Wide-Operating-Range IA-32 Processor in 32 nm CMOS
 IEEE International Solid-State Circuits Conference (ISSCC), 2012.

Intel Claremont: Variable Energy Demand of Systems

- energy demand as an important non-functional system property
- energy-efficient systems require adjustable computing processes



i Although the **energy demand** of computing systems is a characteristic of hardware . . .

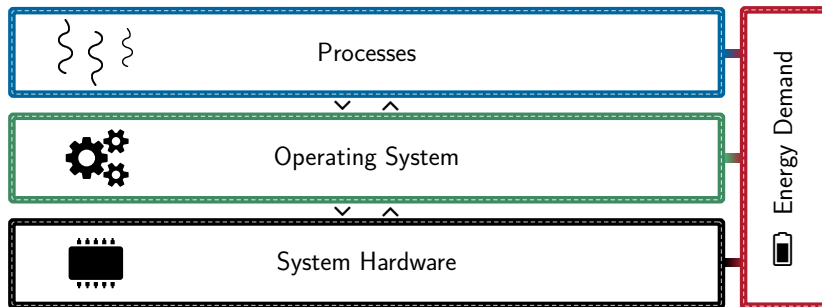
i . . . software is the central **influencing** and **controlling** factor for the energy demand of hardware

Shailendra Jain, Surhud Khare, Satish Yada et al.
A 280mV-to-1.2V Wide-Operating-Range IA-32 Processor in 32 nm CMOS
IEEE International Solid-State Circuits Conference (ISSCC), 2012.

Energy Demand as a System Property

- energy demand** is a physical property of integrated transistor circuits that construct hardware components
 - type**
 - static energy demand
 - dynamic energy demand
 - form**
 - effective energy → maximize
 - energy loss → minimize
- duality and principle of causality**: software and hardware activities
 - software activities ⇒ hardware activities
 - hardware activities ⇒ software activities
- software: **two dimensions** of influence
 - quantitative amount of energy demand
 - control system: energy demand must be under strict governance

System Characteristics



- design and structure** of energy-aware system software
 - interfaces for higher-level abstractions (upwards towards applications)
 - controlling of system-level activities to enforce system strategies (downwards towards the hardware)

Fundamentals

Introduction:

Lecture 1 Overview, Organization

General Topics and Basic Principles:

Lecture 2 Principles of Energy-Aware Computing Systems

- terminology, metrics
- assessing of power and energy demand

Lecture 3 Energy Demand Analysis

- awareness of energy demand at system level
- physical and logical means to determine energy demand

Lecture 4 Energy Management

- hardware power and energy management
- energy accounting at operating-system level

Systems

Energy-Aware Components, Subsystems, and Systems:

- Lecture 5** Components and Subsystems
 - energy-aware system components (e.g., memory, caches)
 - subsystems to integrate energy-aware components
- Lecture 6** Cyber-Physical Systems
 - energy-constraint systems from the embedded domain
 - energy-aware sensors and actuator in control systems
- Lecture 7** Cluster Systems
 - resource allocation in cluster computing environments
 - assessment of remote execution



Software Systems

Energy-Aware System Software:

- Lecture 8** External Lecture, Excursion
 - TBA
- Lecture 9** System Software
 - energy-aware operating systems
 - accounting and enforcement of energy demand
- Lecture 10** Energy-Aware Programming
 - constructive approaches towards energy-aware software
 - software design and restructuring for low energy
- Lecture 11** Research Lecture, In-Depth Paper Review
 - TBA



State of the Art and Advanced Topics

Tie Points, Industry Experience, and Remarks

- Lecture 12** Infrastructure
 - impact of renewable energy, electricity-grid evolution
 - supplementary, fact-related research areas
- Lecture 13** Industry Lecture
 - TBA
- Lecture 14** Research Projects
 - current DFG funded projects at the chair
 - Bachelor's and Master's theses
- Lecture 15** Remarks
 - retrospection and lessons learned
 - wrap-up and perspectives



Language of Instruction

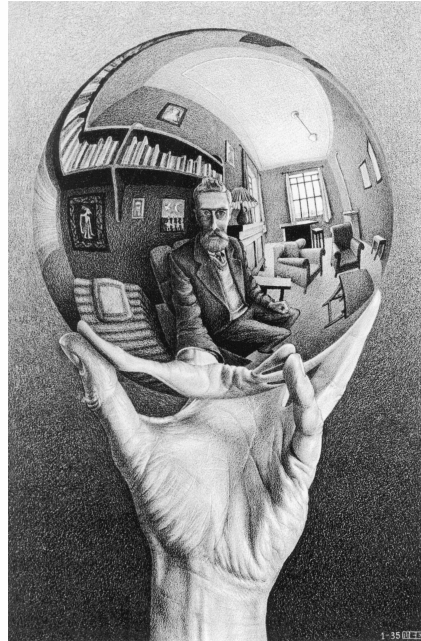
(Ger.) *Unterrichtssprache*



- language of instruction for the lecture
 - English** ■ primary working language
 - German** ■ in case of doubt, German is the fall-back position
- written material (slides or handouts, resp.) will be English
 - with technical terms also stated in German, where applicable



1. learn → new information
2. relate → to existing knowledge
3. reflect



Exercise

Experimental Learning

- deepen knowledge by means of direct experience: „learning by doing”

behavior and operational ability is less a matter of easy instruction but rather functional copy, practice, and use. (Aristotle [1])

- deepen technical discussion of research papers
- consolidation of the lecture and discussion of assignments
- **blackboard practice** under guidance of an exercise instructor
 - registration through [WAFFEL](https://www4.cs.fau.de/Lehre/WS18/V_EASY/)¹, URL see web page: https://www4.cs.fau.de/Lehre/WS18/V_EASY/
 - assignments are to be processed in teamwork: discretionary clause
 - depending on the number of participants
- **computer work** under individual responsibility
 - registration is not scheduled, reserved workplaces are available
 - in case of questions, a exercise instructor is available

¹abbr. for (Ger.) *Webanmeldefrickelformular Enterprise Logic*



- acquire new knowledge
 - prepare next reading on one’s own initiative
 - attend presentation, listen, and discuss topics treated
 - **reading and discussing research papers** on a regular basis
 - jointly with the exercises discussed papers transfer theory to practice
 - reinforce learning matter, reflect
- relate it with previous knowledges
 - computer architecture (GRA) 13
 - system programming (SP, SPiC, GSPiC) 14
 - operating systems (BS), operating-systems engineering (BST) 14
 - modeling, optimization and simulation of energy systems (MOSES) 17
- teaching material presented in the **lecture room**:
 - follow “Lehre” (Eng. *teaching*) at <https://www4.cs.fau.de>
 - copies of the slides are made available as handouts free of charge
 - supplemented by secondary literature as and when required



Requirements

Exercises

- **hard skills** (computer-science expertise)
 - mandatory
 - **structured** computer organization
 - algorithm design and development
 - principles of programming in C → V_SP, V_SPiC, V_BS, V_BST, V_CS
 - utilization of GNU/Linux → V_SP, V_BS, V_BST, V_CS, P_PASST
 - ↳ knowledge gaps will not be closed actively: no extra tuition
 - beneficial
 - basic knowledge of at least one scripting language (e.g. shell, Python, Perl) → V_SP, P_PASST, V_BS, V_CS
 - basic knowledge of a version control system, (preferably) GIT or SVN → V_SP, V_SPiC, V_BS, V_BST, P_PASST
 - optional
 - assembly language (absolute) programming
 - ↳ as appropriate, knowledge gaps will be closed on demand by the instructors
- **soft** (personal, social, methodical) **skills**
 - staying power, capacity of teamwork
 - structured problem solving



- achievable credit points
 - 5 ECTS (*European Credit Transfer System*)
 - corresponding to a face time of 4 contact hours per week
 - lecture and practice, with 2 SWS² (i.e., 2.5 ECTS) each
- German or English, **thirty-minute oral examination**
 - date by arrangement: send e-mail to thoenig@cs.fau.de
 - propose desired date within the official audit period
 - the exception (from this very period) proves the rule...
- examination subjects
 - topics of lecture, blackboard practice, but also computer work
 - brought up in the manner of an “expert talk”
 - major goal is to find out the degree of understanding of inter-relations
- registration through “mein campus”: <https://www.campus.fau.de>

²abbr. for (Ger.) *Semesterwochenstunden*

- energy-aware computing systems
 - fundamental understanding for analyzing and improving the energy demand of computing systems
 - comprehend factors and causality for energy demand that is exhibited by different computing systems
- structured analysis of system designs
 - reading and understanding of subject-related research papers to gain knowledge and relate to own work on exercises and assignments
 - bridging the gap from theory to practice
- reading list for Lecture 2:
 - ▶ Mark Horowitz et al.
Low-power Digital Design
Proceedings of IEEE Symposium on Low Power Electronics, 1994.

Reference List I

- [1] ARISTOTLE:
Nicomachean Ethics.
c. 334 BC