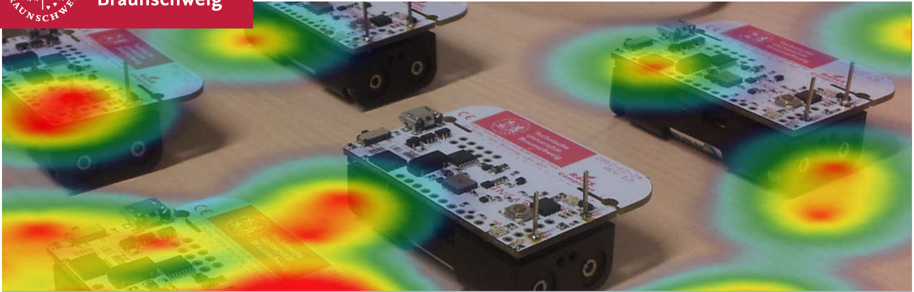
Institute of Operating Systems
and Computer Networks

Technische Universität Braunschweig



Undervolting in WSNs – A Feasibility Analysis

IEEE World Forum Internet of Things 2014

Ulf Kulau, Felix Büsching and Lars Wolf, March 8, 2014

Technische Universität Braunschweig, IBR

Undervolting in WSNs – Motivation

- Energy Efficiency in WSNs / IoT plays a significant role
  - Usability, Feasibility, Acceptance...
- Limping evolution of batteries (capacity)
- Various existing approaches on several layers

Undervolting in WSNs – Motivation

- Energy Efficiency in WSNs / IoT plays a significant role
  - Usability, Feasibility, Acceptance...
- Limping evolution of batteries (capacity)
- Various existing approaches on several layers

Application Layer	Data Management, Compression, ...
Network Layer	Routing, RPL, ...
Data Link Layer	TDMA, LPL (X-MAC), ...
Physical Layer	DPM, PVS, DVS, ...

Undervolting in WSNs – Motivation

Existing approaches are inflexible:

- Real environmental conditions (changes) are less considered
- They act conservatively (reliability)
- Usage comes often with some limitations (e.g. waiting periods)

Undervolting Basics - DVS

- ICs are mostly based on CMOS technology
  - Static power dissipation is negligible
  - Overall power consumption is dominated by

$$p_{dyn} = C_L \cdot f_{cpu} \cdot V^2$$

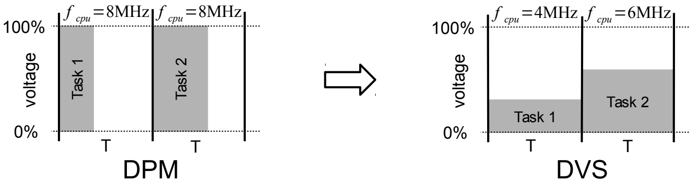
  - But the switching delay of CMOS gates depends on V
  - $\rightarrow V(f_{cpu})$ (Un-)Safe Operating Area

Undervolting Basics - DVS

- ICs are mostly based on CMOS technology
  - Static power dissipation is negligible
  - Overall power consumption is dominated by

$$p_{dyn} = C_L \cdot f_{cpu} \cdot V^2$$

  - But the switching delay of CMOS gates depends on V
  - $\rightarrow V(f_{cpu})$ (Un-)Safe Operating Area
- DVS: Adapting f_{cpu} to current Workload and scale $V(f_{cpu})$



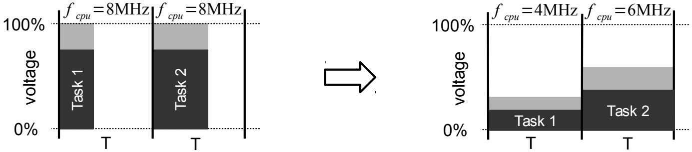
Undervolting Basics - DVS

- ICs are mostly based on CMOS technology
  - Static power dissipation is negligible
  - Overall power consumption is dominated by

$$p_{dyn} = C_L \cdot f_{cpu} \cdot V^2$$

  - But the switching delay of CMOS gates depends on V
  - $\rightarrow V(f_{cpu})$ (Un-)Safe Operating Area

- Undervolting: Violate specifications $V(f_{cpu}) \rightarrow V(f_{cpu}) - \Delta V$



Undervolting

Undervolting – Basics

Temperature Dependency

- Specification of $V(f_{cpu})$ is given in Datasheets
- Specification does not include the temperature $V(f_{cpu}, T)$
  - Threshold Voltage V_{th} of CMOS is temperature dependent

$$V_{th}(T) = V_{th0} + \alpha \cdot (T - T_0)$$

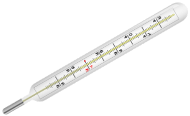
Undervolting – Basics

Temperature Dependency

- Specification of $V(f_{cpu})$ is given in Datasheets
- Specification does <u>not</u> include the temperature $V(f_{cpu}, T)$
  - Threshold Voltage V_{th} of CMOS is temperature dependent

$$V_{th}(T) = V_{th0} + \alpha \cdot (T - T_0)$$

MCUs cover a widespread temperature range with a fixed voltage level $V(f_{cpu})$



$\rightarrow$ MCUs must be able to run below $V(f_{cpu})$ (under *normal* conditions)

Challenges and Issues

Undervolting will lead to a higher unreliability:

- Operating devices outside their specification
- Calculation errors, losses, resets, failures may affect the application

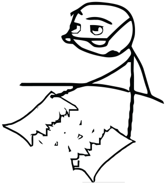
Challenges and Issues

Undervolting will lead to a higher unreliability:

- Operating devices outside their specification
- Calculation errors, losses, resets, failures may affect the application

Our Perspective:

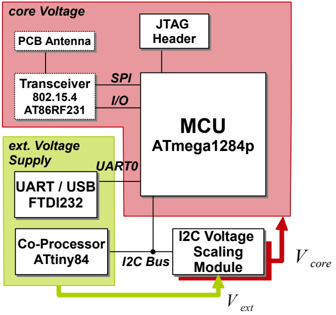
- WSNs are designed to be fault tolerant per se (protocols, algorithms, applications, ...)
- WSNs need increased energy efficiency and offer fault tolerance (ideal)

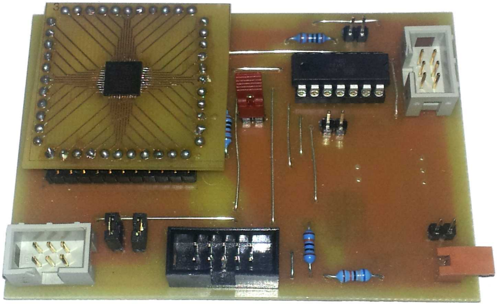


Theory and Practice

Preparations:

- Ordering of ATmega1284p MCUs from different distributors
- Implementation of a prototype to analyze the effect of Undervolting





Detecting failures caused by Undervolting

- Continuous (periodic) observation (counter-check) of...
  - Busses (I2C, SPI)
  - GPIOs
  - Clock rate
  - ALU failures (calculation errors)

[1] A. Rohani and H.-R. Zarandi, "An analysis of fault effects and propagations in avr microcontroller atmega103(l),"

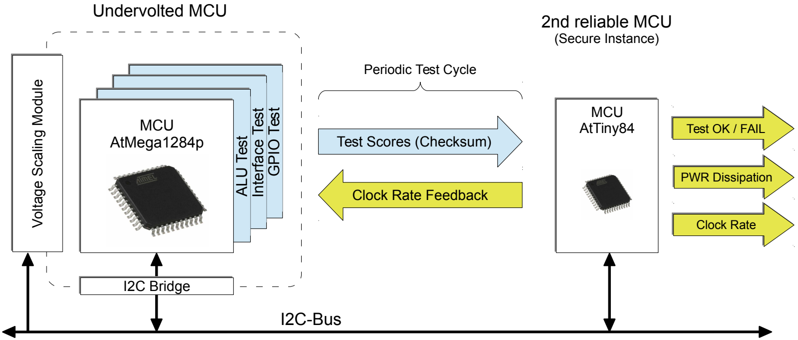
Detecting failures caused by Undervolting

- Continuous (periodic) observation (counter-check) of...
  - Busses (I2C, SPI)
  - GPIOs
  - Clock rate
  - ALU failures (calculation errors)
- How to detect ALU failures by software?
  - A complete test is not adequate ($(2^n)^m$)
- Sufficient error detection through checksum calculation[1]

$$checksum = det(A \cdot B) \;\text{ with }\; A, B \in \mathbb{R}^{n \times n}$$

[1] A. Rohani and H.-R. Zarandi, "An analysis of fault effects and propagations in avr microcontroller atmega103(l),"

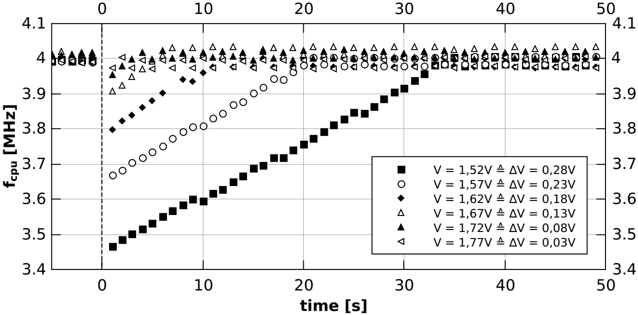
Testbench Implementation



Periodic Test Cycle (1Hz):

- Execute tests on undervolted MCU and use reliable MCU for validation
- Measure time between test cycles and generate binary feedback for clock rate adjustment

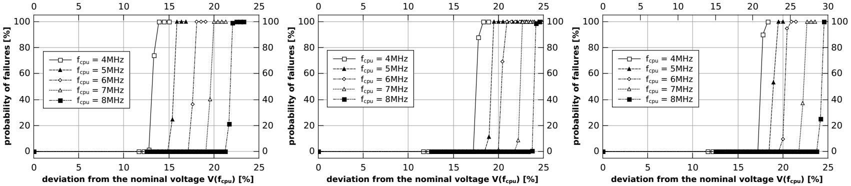
Results – Clock Rate Recalibration



- Linear recalibration of the clock rate (binary feedback)
- Constant clock rate despite using undervolting

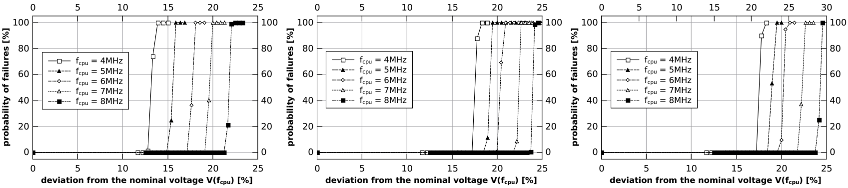
Results – Functionality analysis 1/2

Evaluation of three MCUs from different distributors:



Results – Functionality analysis 1/2

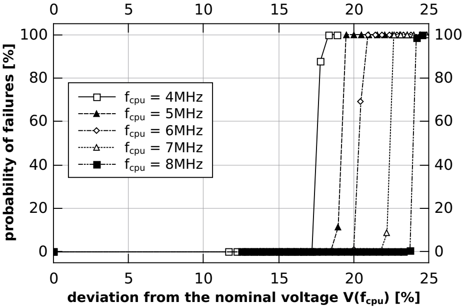
Evaluation of three MCUs from different distributors:



→ Similar but individual results even with same kind of MCU!

Results – Functionality analysis 2/2

Exemplary Results of MCU2:

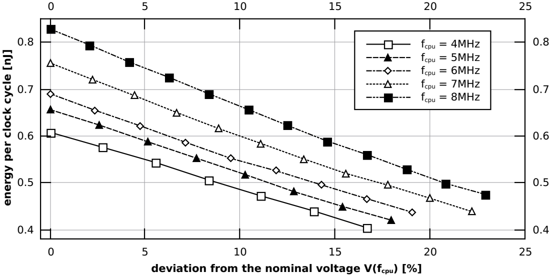


Statements:

- Undervolting is possible
- Malfunction appears in a small, sharp region
- Individual characteristic (even for same kind of MCUs)
- Possible deviation growth with clock rate

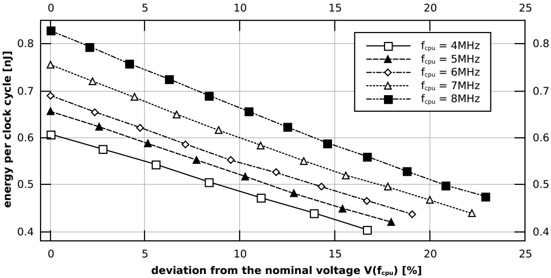
Results – Energy Savings

Averaged energy per clock cycle:



Results – Energy Savings

Averaged energy per clock cycle:



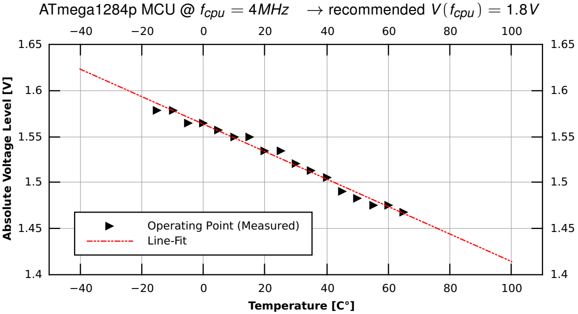
Compared to recommended voltage level $V(f_{cpu})$:

$f_{cpu}[MHz]$	4	5	6	7	8
$E_{min}[pJ]$	404.0	420.2	437.0	466.8	475.0
$max(\delta_e)[\%]$	33.52	35.96	36.67	38.23	42.66

Results – Temperature Dependency

Presumption: Threshold voltage depends on temperature

- Measurement of the minimum (stable) operating point in a climatic chamber



Nodes are exposed to various environmental conditions

Results – Wireless Communication

Undervolting and transceiver unit:

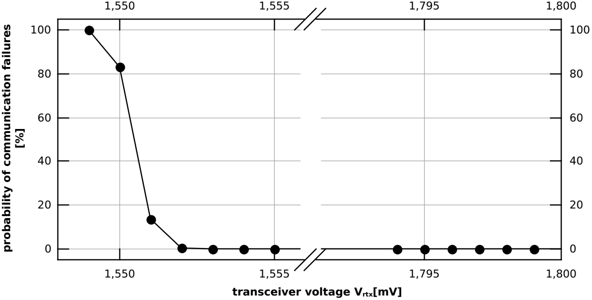
- Not mainly based on CMOS (RF-section, amplifiers, ...)
- How to detect errors?
  - Increased packet loss indicates communication errors

Results – Wireless Communication

Undervolting and transceiver unit:

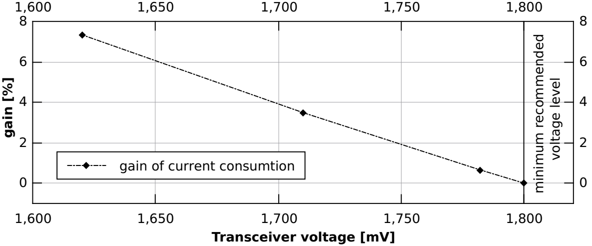
- Not mainly based on CMOS (RF-section, amplifiers, ...)
- How to detect errors?
  - Increased packet loss indicates communication errors

AT86RF231 IEEE 802.15.4 Transceiver, nominal minimal voltage level 1.8 V, Contiki + RIME-Stack



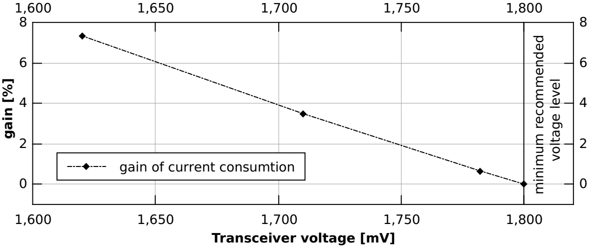
Results – Wireless Communication

- The transceiver has a fixed transmit and reception power
  - Power dissipation is less bound to the voltage level
  - Nevertheless:



Results – Wireless Communication

- The transceiver has a fixed transmit and reception power
  - Power dissipation is less bound to the voltage level
  - Nevertheless:



Only few experiences:

- Undervolting of other parts (e.g. transceiver) is possible

Summary

Legitimation for undervolting in WSNs:

- Using safety margin of CMOS parts
  - Temperature dependencies
  - Individual tolerances
- WSNs are fault tolerant

Summary

Legitimation for undervolting in WSNs:

- Using safety margin of CMOS parts
  - Temperature dependencies
  - Individual tolerances
- WSNs are fault tolerant

Prototype implementation and evaluation of COTS MCUs:

- Undervolting of MCUs is possible
- May influence parts of the MCU (e.g. RC-oscillator)
- Energy savings up to 42%

Summary

Legitimation for undervolting in WSNs:

- Using safety margin of CMOS parts
  - Temperature dependencies
  - Individual tolerances
- WSNs are fault tolerant

Prototype implementation and evaluation of COTS MCUs:

- Undervolting of MCUs is possible
- May influence parts of the MCU (e.g. RC-oscillator)
- Energy savings up to 42%

Usage of experimental results on higher layers:

- Undervolting adds heterogeneity
  - New approaches for load balancing, routing, ...

Summary

Legitimation for undervolting in WSNs:

- Using safety margin of CMOS parts
  - Temperature dependencies
  - Individual tolerances
- WSNs are fault tolerant

Prototype implementation and evaluation of COTS MCUs:

- Undervolting of MCUs is possible
- May influence parts of the MCU (e.g. RC-oscillator)
- Energy savings up to 42%

Usage of experimental results on higher layers:

- Undervolting adds heterogeneity
  - New approaches for load balancing, routing, ...

Thank you for your attention! Questions?

Ulf Kulau

kulau@ibr.cs.tu-bs.de