

Optimisation of battery operating life considering software tasks and their timing behaviour

Dissertation zur Erlangung des Grades eines
Doktors der Ingenieurwissenschaften

von

Dipl.-Math. Henrik Lipskoch

Gutachter:

Prof. Dr.-Ing. Wolfgang Nebel

Prof. Dr.-Ing. Frank Slomka

Tag der Disputation: 19.02.2010

Bibliographic information published by the Deutsche Nationalbibliothek

The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data are available in the Internet at <http://dnb.d-nb.de>.

Lipskoch, Henrik

Optimisation of battery operating life considering software
tasks and their timing behaviour

ISBN 978-3-941274-95-2

All Rights Reserved

1. Edition 2011, Göttingen

© Optimus Verlag

URL: www.optimus-verlag.de

Printed in Germany

Paper is FSC certified (wood-free, chlorine free and acid-free,
and resistant to aging ANSI 3948 and ISO 9706)

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, scanning, or otherwise without the prior written permission of the Publisher. Request to the Publisher for permission should be addressed to info@optimus-verlag.de.

Acknowledgements

To all those persons who gave me insights, were patient with me, or were just there for me, I am thankful. First, I would like to thank my supervisors Wolfgang Nebel and Frank Slomka who both equally influenced my work. Frank was my first supervisor before he joined the University of Ulm, and Wolfgang took over for the second half of my time. Wolfgang provided me with a good environment, and Frank with enthusiasm.

Next, I would like to thank my colleagues for many honest comments and plenty of discussions. Especially, I need to thank four persons: first, Mario Korte who looked this work over; Thomas Weißmüller and Nils Schröder who together made such a reliable and fully automated device, which enabled my battery measurements; and last, Johannes Faber who brought me to the sport of running and ran with me twice a week under all weather conditions. He was never tired to talk over our progress on the street and the progress of our research activities.

I would like to thank my friends, especially Christian Lüpkes and Christian Oppermann, who reminded me many times that there are things beyond work that are necessary and enjoyable.

Last, I would like to thank my brothers and their families who believed in my work the whole time, my mother who supported me with everything, my father who was eager to hold this thesis in his hands, and Kinga for without her this book would not be.

Abstract

Users of mobile embedded systems have an interest in long battery operating life. The longer a system can operate without need for recharge or battery replacement, the more will maintenance cost and the number of faults due to insufficient power supply decrease. Operating life is prolonged by saving energy, which may reduce available processing time. Mobile embedded systems communicating with other participants like other mobile systems or radio stations have to satisfy time guarantees ensuring reliable communication. Thus, methods that save energy by reducing processing time are not only subject to available processing time but subject to the embedded system's time guarantees.

By performing parameter optimisations offline, run-time computation can be avoided, decisions can be taken early, design complexity can be reduced, and energy can be saved. Especially, by computing processor shutdown durations offline, no extra circuitry to monitor system behaviour and to wake up the processor needs to be designed, deployed, or power supplied. The resulting optimisation solutions of this research work are designed to depend on minimal extra circuitry: only an energy counter and a programmable timer are required.

For this thesis, software tasks which share one processor are considered. The scheduling algorithm earliest deadline first is assumed, and per task, a relative deadline is assumed. Tasks may be instantiated arbitrarily as long as this occurrence behaviour is given in the notion of event streams. Especially, the theory and the solutions presented in this work do not require tasks to occur periodically.

Scaling the processor's voltage and shutting down the processor are taken into account as methods for saving energy. With given per task worst-case execution times and the tasks' event streams, the real-time feasibility of each energy optimised solution is proven.

The used real-time feasibility test has the advantage that it can be approximated by a combination of step-functions and linear functions. The approximation provides an adjustable small error, and it yields an adjustable number of linear optimisation constraints.

Reducing the processor's voltage reduces processor frequency, therefore, execution time increases. The resulting slowdown becomes the optimisation variable, either global, for all tasks, or local, individually per task. It is shown that a global slowdown factor can be computed from linear constraints together with a linear objective in a linear program, whereas local slowdown factors are shown to require a non-linear objective that is still convex.

Processor shutdown switches off the processor for a specific time; it blocks the processor from processing software. Duration and occurrence become optimisation parameters, and the linearised real-time feasibility test is used to decide whether a parameter setting is feasible or violates deadlines.

With this thesis, two shutdown policies are introduced. Periodic shutdown switches off the processor periodically, thus, besides duration, the period is to be optimised. Task-dependent shutdown switches off the processor with an inherited occurrence behaviour from a certain task, thus, besides duration, the parent task needs to be chosen, as well as

the number of instances of the parent task to fall between two consecutive shutdowns.

In this thesis, a battery model is developed. Influencing factors, design and usage dependent, are discussed. The Peukert-formula for constant discharge, an approximation for battery life known since over 100 years, and its derivation introduced by Rossander and Forsberg for variable currents are evaluated with experiments with non-constant discharge currents, showing significant error of more than 20%. A known theory concerning the influence of periodically discharge is also shown to not hold for chosen commercially available batteries designed for mobile systems. Based on the experimental results, a battery model in closed form is derived which is as simple as the Peukert-formula and provides a significantly less error. An error of less than 7% is achieved.

Each of the presented methods for saving energy, global or local slowdown, or periodic or task-dependent shutdown, yields a different system configuration, out of which the best with respect to operating life is chosen with the help of the developed battery model. Discharge profiles serve as a connection between information in the notion of event streams, task power consumptions, and the battery model.

Zusammenfassung

Benutzer mobiler eingebetteter Systeme haben ein Interesse an einer langen Batterienutzungsdauer. Je länger ein System ohne Nachladen oder Batteriewechsel operieren kann, desto geringer werden Wartungskosten und durch Energiemangel bedingte Ausfälle. Energiesparmethoden helfen die Nutzungsdauer zu verlängern, können jedoch zu einer Reduzierung der verfügbaren Rechenzeit führen. Die Systeme können zeitlichen Anforderungen unterworfen sein, etwa um eine zuverlässige Kommunikation sicherzustellen. Somit ist der Einsatz von Energiesparmethoden neben der verfügbaren Rechenzeit auch an die zeitlichen Anforderungen gebunden.

Indem Optimierungen zur Designzeit durchgeführt werden, lassen sich früh Entscheidungen treffen und weitere Berechnungen zur Laufzeit vermeiden. Insbesondere die Parameterberechnung für die Prozessorabschaltung zur Designzeit ermöglicht ein Auskommen ohne zusätzliche systemüberwachende Schaltkreise, die ihrerseits entworfen und mit Energie versorgt werden müssten. Die sich aus dieser Arbeit ergebenden Optimierungslösungen sind dahin entworfen minimale Hardware-Anforderungen zu haben: es sind nur ein Energiezähler und eine programmierbare Echtzeituhr nötig.

Für diese Arbeit werden Software-Tasks betrachtet, die sich einen Prozessor teilen. Für die Ablaufplanung wird ausschließlich der Algorithmus „nach frühester Frist zuerst“ betrachtet. Es wird den Software-Tasks ein beliebiges Auftreten zugestanden, solange sich dieses mit der Ereignisstrommethodik formulieren lässt. Insbesondere ist es für die präsentierte Theorie und auch für die sich ergebenden Optimierungslösungen nicht erforderlich, dass Tasks periodisch auftreten.

Als Energiesparmethoden werden Prozessorspannungsskalierung und Prozessorabschaltung betrachtet. Mit Hilfe von als gegeben vorausgesetzten Task-Laufzeitoberranken und auf Basis der Ereignisströme, wird die Echtzeitfähigkeit der energieoptimierten Lösungen geprüft.

Der verwendete Echtzeittest hat den Vorteil durch Treppenfunktionen und lineare Funktionen approximierbar zu sein. Die Approximation bietet einen einstellbar kleinen Fehler und liefert für die Optimierung eine einstellbare Menge linearer Bedingungen.

Prozessorverlangsamung reduziert die Taktfrequenz und erhöht damit die Rechenzeit der Tasks. Der sich aus der Frequenz ergebende Verlangsamungsfaktor wird zur Optimierungsvariablen, entweder global, d.h. für alle Tasks gleichermaßen, oder lokal, d.h. für jede Task individuell. Ein globaler Faktor lässt sich mit Hilfe einer linearen Zielfunktion und den linearen Bedingungen aus dem Echtzeittest in einem linearen Programm optimieren. Wie in dieser Arbeit gezeigt, benötigt dagegen die Optimierung für lokale Verlangsamungsfaktoren eine nicht-lineare Zielfunktion, welche aber noch konvex ist.

Den Prozessor abzuschalten, heißt alle Berechnungen für eine bestimmte Zeit aussetzen. Dauer und Häufigkeit der Abschaltung werden Optimierungsvariablen, und deren Echtzeitfähigkeit wird mit Hilfe des linearisierten Echtzeittests überprüft.

Mit dieser Arbeit werden zwei Abschaltverfahren vorgestellt. Bei periodischem Abschalten wird, neben der Abschaltdauer, die Periodendauer optimiert. Task-abhängiges

Abschalten koppelt die Abschaltung an das Auftreten einer bestimmten Task. Neben der Abschaltdauer, werden Task und die Anzahl der Task-Instanzen, die zwischen zwei Abschaltungen fallen sollen, bestimmt.

Jedes der präsentierten Energiesparverfahren, ob globale oder lokale Verlangsamung, ob periodische oder task-abhängige Abschaltung, liefert eine andere Systemkonfiguration. Aus welcher sich die längste Batterienutzungsdauer ergibt, wird mit Hilfe eines in dieser Arbeit entwickelten Batteriemodells ermittelt. Entladeprofile dienen der Kopplung der Ereignisströme, der Verlustleistung pro Task und des Batteriemodells.

In dieser Arbeit wird ein Batteriemodell entwickelt. Beeinflussende Faktoren, abhängig von Design und Nutzung, werden diskutiert. Die seit über hundert Jahren bekannte Peukertsche Formel zur Abschätzung der Batterienutzungsdauer für konstante Ströme und die Weiterentwicklung von Rossander und Forsberg für variable Ströme werden anhand variabler elektrischer Ströme evaluiert, wobei die hier gezeigten Messungen signifikante Abweichungen von mehr als 20% aufzeigen. Ein bekannter Ansatz, Batterien periodisch zu entladen, wird auf Signifikanz hin untersucht und dessen Unwirksamkeit für ausgewählte, für den mobilen Einsatz gedachte Batterien gezeigt. Basierend auf den Experimenten wird ein, Batteriemodell in geschlossener Form entworfen, welches wie die Peukertsche Formel einfach ist, und welches einen Fehler von weniger als 7% aufweist.

Contents

1. Introduction	17
1.1. Assumptions and Targeted Problem	18
1.2. Achievements	19
1.3. Overview of the Work	19
2. Preliminaries	21
2.1. Scheduling	22
2.2. Event Streams	23
2.3. Approximating Event Streams	27
2.4. Demand Bound Function	29
2.5. Energy Consumption of Software	33
2.5.1. Power dissipation	33
2.5.2. Power Consumption of Software	36
2.5.3. Methods for Power Saving	38
2.6. Batteries as Energy Sources	43
2.6.1. Function Principle	43
2.6.2. Construction	45
2.6.3. Battery Capacity	47
2.6.4. Discharge	49
2.7. Summary	50
3. State of the Art	51
3.1. Hard Real-Time	51
3.2. Power Dissipation	53
3.2.1. Voltage Scaling	54
3.2.2. Power Management	56
3.3. Battery Modelling	59
3.3.1. Physical Models	59
3.3.2. SPICE-Models/Electric Circuit	61
3.3.3. Abstract Models	62
3.3.4. Analytical Models	63
3.3.5. Coulomb counters	64
3.3.6. Summary	65
3.4. Summary	65

4. Concept Presentation and Classification	67
4.1. Task Graphs, and Levels of Abstraction	67
4.2. Premises	69
4.3. Objective	70
4.4. Conceptual Work-Flow	71
4.5. Comparison to Related Work	73
4.5.1. Voltage Scaling/Processor Slowdown	73
4.5.2. Power Management/Processor Shutdown	74
4.6. Summary	78
5. Modelling Processor Slowdown	79
5.1. Energy Bound	79
5.2. Characterisation of Processor Slowdown	80
5.2.1. Voltage and Frequency Scaling	81
5.2.2. Global Voltage Scaling	82
5.2.3. Local Voltage Scaling	83
5.3. Constraints for Processor Slowdown	85
5.3.1. Constraints for Global Slowdown	85
5.3.2. Constraints for Local Slowdown	86
5.4. Comparing Linear and Non-Linear Objective for Local Slowdown	87
5.5. Summary	89
6. Modelling Processor Shutdown	91
6.1. Characterisation of Processor Shutdown	91
6.1.1. Efficiency of Processor Shutdown Methods	92
6.1.2. Energy Consumed with Application of Processor Shutdown . . .	92
6.2. Periodic Processor Shutdown	93
6.2.1. Periodic Shutdown of the Processor	93
6.2.2. Determination of the Parameters ‘Period’ and ‘Duration’	94
6.3. Task-Dependent Shutdown of the Processor	99
6.3.1. The Case of Two Never Overlapping Consecutive Executed Tasks	100
6.3.2. Processor Shutdown in Between Two Jobs of the Same Task . . .	105
6.3.3. Shutdown in Between Many Instantiations of the Same Task . . .	107
6.3.4. Shutdown After Execution of Tasks Sharing Occurrence Behaviour	110
6.4. Summary	112
7. Modelling the Battery as Energy Source	113
7.1. Factors Influencing the Operating Life of a Battery	113
7.1.1. Battery Design Dependent Factors	113
7.1.2. Usage Dependent Factors	114
7.1.3. Influence of Pulse Duration Concerning Pulsed Discharge	119
7.1.4. Selection of Influences	123

7.2. Battery Modelling	125
7.2.1. Voltage, Current, and Time Dependence of a Battery Model . . .	125
7.2.2. Methodology of Event Streams	126
7.2.3. Discharge Profiles and Final Battery Model	129
7.3. Summary	135
8. Experimental Setup and Evaluation	137
8.1. Evaluation of Presented Concept	137
8.1.1. Implementation	138
8.1.2. Task Sets for Evaluation	143
8.1.3. Evaluation of Processor Slowdown	147
8.1.4. Evaluation of Methods for Processor Shutdown	154
8.2. Battery Modelling	161
8.2.1. Measurement Device	161
8.2.2. Measurements of Batteries	165
8.3. Conclusion	173
8.3.1. Methods for Saving Energy	174
8.3.2. Battery Model	174
9. Conclusion	175
9.1. Summary	175
9.2. Outlook	177
A. Efficiency of DC-DC-Converters	179
B. Hierarchical Event Streams with Sequences	185
C. Deterioration of a SANYO HR4U cell	187

List of Figures

2.1. Transformation From Explicit Schedule to Event Stream Description . . .	25
2.2. Alternating Event Densities	26
2.3. Likely Overestimations of an Event Stream	28
2.4. CMOS Inverter, Loading Capacities	34
2.5. CMOS Inverter, Short Circuit Phase	34
2.6. Leakage Currents in an NMOS-Transistor	35
2.7. Power Consumption by Time Diagram for Duration of Mode-Switch . . .	41
2.8. Procrastination	42
2.9. Schematic of an Electrochemical Cell	44
2.10. Construction of Batteries	46
2.11. Temperature-Capacity Curve of a Common NiCd-Cell	48
2.12. Temperature-Capacity Curve of a Common Lithium Cell	48
4.1. Battery and Processor, Discharge Relation	67
4.2. Battery Process	68
4.3. Scheduling Process	68
4.4. Conceptual flow	72
5.1. Using Two Different Objectives to Obtain per Task Slowdown Factors, Remaining Idle Time for the Aircraft Controller Example	88
5.2. Using Two Different Objectives to Obtain per Task Slowdown Factors, Resulting Average Power Consumption for the Aircraft Controller Example	88
6.1. Periodic Shutdown	94
6.2. Several Local Extrema of Utilisation by Efficiency Graphs in Case of Periodic Shutdown	97
6.3. Modified Versus Unmodified Algorithm for the Computation of Parame- ters for Periodic Shutdown	99
6.4. Task-Dependent Shutdown, Every Instantiation of ρ	100
6.5. Relations of Three Consecutive Tasks	101
6.6. Never-Overlapping Execution: Two Worst-Cases	102
6.7. Task-Dependent Shutdown, Every 2nd Instantiation	108
6.8. Evaluating Task-Dependent Shutdown: Every Instance vs. Varying Number	110
6.9. Precedence Constraints, Consecutive Instantiation	111
6.10. Precedence Constraints, Parallel Instantiation	111

7.1. Pulsed Discharge Versus Constant Discharge	116
7.2. Variation of Pulse Width	121
7.3. Discharge Curve for a SONY LiMn Cell	126
8.1. Process Structure for Optimisation of Operating Life	138
8.2. Palm-Pilot Example, Test Index Versus Remaining Idle Time	148
8.3. Palm-Pilot Modification 1, Test Index Versus Remaining Idle Time	148
8.4. Palm-pilot Modification 2, Test Index Versus Remaining Idle Time	149
8.5. Satellite Example, Test Index Versus Remaining Idle Time	150
8.6. Aircraft Controller, Test Index Versus Remaining Idle Time	150
8.7. Aircraft Controller, Demand Bound Function for Test Indices 3 and 90	151
8.8. Test Index Versus Average Power Consumption, Palm-Pilot Modification 2	152
8.9. High Power Task by Average Power Consumption, Palm-Pilot Modification 2	153
8.10. High Power Task by Average Power Consumption, Original Palm-Pilot	153
8.11. Efficiency of Shutdown Methods Applied on the Aircraft Controller Task Set Using a Set of Various Test Indices	155
8.12. Efficiency by Break-Even Time Diagram, Several Shutdown Methods	157
8.13. Efficiency by Break-Even Time Diagram, Periodic, Task-Dependent, and Ideal Shutdown, Modified Palm-Pilot Task Sets	158
8.14. Efficiency by Break-Even Time Diagram, Periodic, Task-Dependent, and Ideal Shutdown, Olympus Task Set	159
8.15. Efficiency by Break-Event Time Diagram, Periodic, Task-Dependent, and Ideal Shutdown, Aircraft Controller Task Set	160
8.16. Raise and Fall Times for Pulsed Discharge	163
8.17. Raise and Fall Times for Pulsed Discharge Current, a Closer Look	164
A.1. DC-DC-Converter in Circuit	179

List of Tables

2.1. Categorisation of Low-Power Modes	39
2.2. Electric Currents and Transition Times of the STM32F103 Microcontroller's Power Modes	40
4.1. Processor Slowdown/Dynamic Voltage Scaling, Comparison to Related Work	75
4.2. Processor Shutdown/Power Management, Comparison to Related Work	77
7.1. Pulsed Discharge vs. Constant Discharge, Measured with a SONY 18650 LiMn Cell	118
7.2. Fitting the Peukert-Coefficient for the SONY Lithium-Manganese Cell	119
7.3. Pulsed Discharge vs. Constant Discharge, Using Similar Loads at Cut-Off Potential	120
7.4. Equidistant Pulses and the Capacity Obtained	122
7.5. Battery Surface Temperatures for Measurements of Table 7.4, in °C	123
7.6. Repetition of Measurements of Table 7.4	124
7.7. Computation of Discharge Times to the Discharge of Specific Amounts of Normalised Battery Capacity, Currents 0.675 A, 1.35 A, 2.7 A	130
7.8. Discharging the Same Amount of Normalised Capacity with Varying Pre-Discharge Currents and Corresponding Normalised Capacities	131
7.9. Computation of Discharge Times to Discharge Specific Amounts of Normalised Battery Capacity, Currents 1 A, 2 A, 3 A	131
7.10. Discharging Specific Amounts of Normalised Capacity with Varying Pre-Discharge Currents and Corresponding Normalised Capacities	132
7.11. Discharging the Same Amount of Normalised Capacity with Varying Pre-Discharge Currents and Corresponding Normalised Capacities, Repetition and Extension of Previous Measurements	132
7.12. Discharge of the ULT 18650 FP LiFePO ₄ Secondary Battery	133
7.13. Pre-Discharging and Discharging the ULT 18650 FP Battery	134
8.1. Task Set of the Palm-Pilot	144
8.2. Modification 1 of the Palm-Pilot	144
8.3. Modification 2 of the Palm-Pilot	145
8.4. Task Set of the Olympus Attitude and Orbital Control System	145
8.5. Task Set of the Aircraft Controller	146

8.6. Aircraft Controller, Performance Improvements of Local Slowdown Method	151
8.7. Validating the Influence of Pulsed Discharge on the ENERGIZER NH15 Cell	166
8.8. Fitting the Peukert-Coefficient for the ENERGIZER Cell	167
8.9. Validating the Influence of Pulsed Discharge on the SANYO HR4U Cell .	168
8.10. Discharging the Battery of Type SANYO HR4U to Cut-Off Potential, with and without Pre-Discharge	170
8.11. Validating the Influence of Pulsed Discharge on a ULT LiFePO ₄ Cell . . .	172
8.12. Fitting the Peukert-Coefficient for the ULT 18650 FP Cell	173
A.1. Efficiency of some Linear DC-DC-Converters, Manufactured by Analog Devices (Models AD...), MAXIM Dallas (Models MAX...), and Linear Technology (Models 1...)	180
A.2. Efficiency of some Switching DC-DC-Converters, Manufactured by MAXIM Dallas	181
A.3. Efficiency of further Switching DC-DC-Converters, Manufactured by MAXIM Dallas, and Analog Devices (Models AD...)	182
A.4. Efficiency of further Switching DC-DC-Converters, Manufactured by Analog Devices (Models AD...), and Linear Technology (Models 3...) . .	183
C.1. Discharging a SANYO HR4U Cell to Cut-Off Potential, Capacity Degradation	188
C.2. Discharging a SANYO HR4U Cell to Cut-Off Potential, Further Capacity Degradation	189

1. Introduction

A traffic bridge often serves for many years millions of cars to pass rough terrain. When such a bridge is built, cranes lift and shift large structures. Persons control these cranes to move bridge components into the right positions. Communication between controller and crane is done via radio. If the radio fails or transmits twisted messages communication is interrupted: structures might get broken; people might get severe injuries. Damaged material and hurt humans are the consequences of unreliable communication.

One of the reasons why communication eventually fails is a weak battery, as mobiles and radios usually are powered by batteries. Another reason is the user had no dependable information on the remaining energy within the battery.

On the one hand, methods for saving energy can prolong battery operating life, and the most energy surely is saved by switching the device off, but with the drawback that communication is lost: energy savings must retain the radio's correct functionality.

Modern embedded systems are implemented partly in software and partly in hardware. The software is divided into pieces of execution called tasks. These tasks are assigned to a processor which they share. Each task may include communication. For successful operation, communication between tasks, data exchange or communication to the outside of the processor induce timing constraints, i.e. deadlines. Deadlines become part of the system's specification to ensure that the system operates reliably. Methods for saving energy are subject to these constraints, especially those methods which affect a software's processing time. In this thesis, the latter are considered.

On the other hand, a forecast on battery operating life can provide a guarantee on the system's minimal run-time, and an early forecast, at design time of the system, allows for an early maintenance estimation which helps to avoid shortages when the system is in use.

A kind of forecast could be to measure the energy consumption per hour of the embedded device and to divide the specified capacity of the battery to use by this amount, with the result being equal to the amount of time the system could operate. But this simple formula does not yield dependable results: experience and research show that the operating life of a battery heavily depends on the current drawn from it; the operating life depends non-proportionally on the current. For a reliable forecast, a battery model is to be used. This thesis presents a model that is developed from experiments.

Energy savings, timing constraints, and battery modelling are the three topics that are to be brought together to form the optimisation problem. The resulting methodology will provide reliable energy optimised solutions.

1.1. Assumptions and Targeted Problem

For this work, three basic components are considered. These are a processor with several levels of speed and energy consumption, a set of software tasks with deadlines being scheduled earliest deadline first, and a battery. These are shortly discussed next.

To begin with the software tasks, beyond deadlines, parameters that describe embedded software include a description of the software's occurrence behaviour which depict the frequency for each task to become instantiated and be ready for processing. This behaviour can be periodic; the period itself can be affected by deviations (jitters), e.g. caused by time-slot-based communication; the behaviour can be sporadic, yet with a minimal time to pass between two consecutive task instantiations; or, for example, the behaviour can be affected by time shifts, e.g. caused by a clock with skew. Further, parameters include upper bounds on execution times. The bound on a task's execution time and its occurrence behaviour form a task's timing behaviour. Both can be combined to compute upper bounds on the processing demand. The sum of all tasks' demands yields the overall demand, and thus the available idle time, which is to be exploited by methods for saving energy. For the work in this thesis, it is assumed that deadline, worst-case execution time, and occurrence behaviour are given for each task of the task set.

Parameters describing the processor include the processor's ability to perform voltage and frequency scaling, the ability to transit into low-power modes, and the amount of modes available. The processor's power consumption differs for various modes and for different software. Information on power consumption and timing behaviour combines to an energy demand estimation: this is an aim of this thesis, therefore, power consumptions for different modes and software are assumed to be given.

Parameters describing the battery include brand and capacity for various discharge currents. Because the capacity depends on the current to be drawn, the capacity is rated (i.e. determined) for different currents, which results in the rated capacity for a battery. A model based on these parameters is capable of estimating the battery's operating life for a given energy demand. Therefore, these parameters are assumed to be given as well.

Problem statement The problem to be solved is increasing a battery's operating life by applying methods for saving energy on an earliest-deadline-first scheduled set of tasks, with each task not necessarily occurring periodically, and with hard real-time constraints in form of a relative deadline for each task. Different methods may yield different solutions. To rate the possible solutions of the optimisation problem, a battery model is to be developed.

The stated problem is to be tackled by the work presented in this thesis with the help of the theory of event streams. The theory follows the idea of covering all cases by only inspecting worst-cases: if an assertion holds for a worst-case, so does it for all cases included in the worst-case. A successful worst-case test will provide a reliable guarantee for all situations that might happen while the embedded system is running.

Note, event-stream theory can also be applied on preemptive fixed-priority systems.

1.2. Achievements

This work brings together real-time feasibility, energy optimisation, and battery modelling for embedded systems. And it contains the following contributions.

First, a convex optimisation problem for voltage scaling per task and per task set is presented in Chapter 5. The linearity is shown for global slowdown. The convexity is shown for local slowdown. The solutions have reduced complexity compared to the state-of-the-art: the complexity equals the number of tasks times the number of approximations steps. And this is the first work to apply voltage scaling per task to non-periodic task sets with arbitrary but fixed deadlines. A comparison to related work in this field is depicted in Table 4.1 on page 75.

Second, two optimisation problems for processor slowdown, periodically and task-dependent, are presented in Chapter 6, and this is the first presentation of processor shutdown applied to non-periodic task sets with arbitrary but fixed deadlines. Periodic shutdown is shown to be a convex optimisation problem. Task-dependent shutdown is shown to be a non-convex optimisation problem; variation of the variables deadline and worst-case execution time together appears to lead to a non-convex problem. All solutions have reduced hardware requirements for processor shutdown compared to related work: a programmable real-time clock (this is the least possible requirement for external circuitry). A comparison to related work in this field is depicted in Table 4.2 on page 77.

Third, a battery model for predicting operating life is presented in Chapter 7. Measurements are presented with samples of today's batteries for mobile use. These involve constant currents and highly non-constant currents reflecting energy consumptions of today's embedded systems. The measurements for the model introduced by Rossander and Forsberg (Peukert-formula for non-constant currents) show an error of 20%. The measurements for the developed model show an error of 6.7%, instead. A state-of-the-art approach to prolong battery life using a discharge with high pulses is falsified by measurements. And finally, the developed battery model is not based on simulation; it is in closed form instead, as opposed to solutions of current research.

This thesis presents an offline optimisation of processor slowdown and shutdown that allows for predicting the energy demand already at design time, and without computational intensive simulation, see Chapter 4 for the concept and Chapter 8 for implementation and experiments.

1.3. Overview of the Work

The work is structured into eight chapters, each composed of sections, subsections, sub-subsections, and paragraphs. Each chapter ends with a summary or a conclusion. Chapter 2 includes the foundations helping to understand the rest of the work. Chapter 3 is a discussion of works dealing with similar problems as the one described in this thesis; it gives a survey on battery models.

The concept behind the work presented in this thesis is shown and classified in Chapter

4. This chapter also includes an overview of the assumptions made as well as a comparison to related work. The reader will learn about the interaction of processor slowdown and the real-time feasibility test in Chapter 5; regardless of the occurrence behaviour, optimisation of processor slowdown is done per task and per task set, with the real-time test providing the optimisation constraints. The reader will find in Chapter 6 the idea to model processor shutdown as a hard real-time task, and the optimisation of its occurrence behaviour and execution time in several directions. Chapter 7 discusses properties of batteries affecting their operating life, enriched by experimental results; the chapter presents the chosen battery model and the coupling of task timing and power parameterisation to a battery model in the form of discharge profiles. The reader will find the descriptions of experimental setups and further experimentation results in Chapter 8.

Chapter 9 depicts the conclusion of the results and the achievements presented in this thesis, further, it provides an outlook on future work.

The appendix contains further notable material beyond the thesis. The appendix is followed by a list of bibliographic references. This thesis ends with the index containing frequently used notions and symbols.