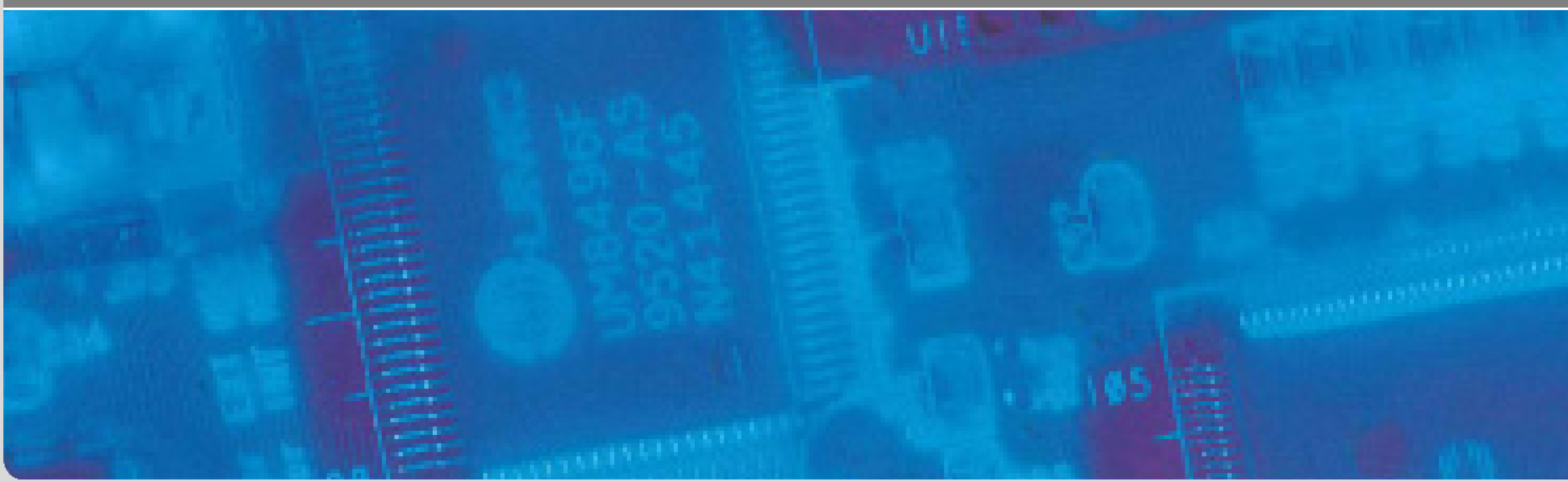
Low Power Design

Volker Wenzel on behalf of Prof. Dr. Jörg Henkel
Summer Term 2016

CES – Chair for Embedded Systems



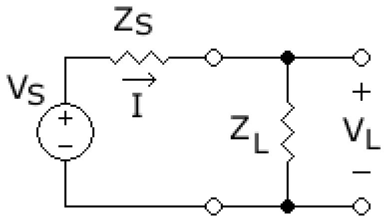
Lecture Slides Download

Overview Low Power Design Lecture

- Introduction and Energy/Power Sources (1)
- Energy/Power Sources(2): Solar Energy Harvesting
- Battery Modeling – Part 1
- Battery Modeling – Part 2
- Hardware power optimization and estimation – Part 1
- Hardware power optimization and estimation – Part 2
- Hardware power optimization and estimation – Part 3
- Low Power Software and Compiler
- Thermal Management – Part 1
- Thermal Management – Part 2
- Aging Mechanisms in integrated circuits
- Lab Meeting

Internal Resistance

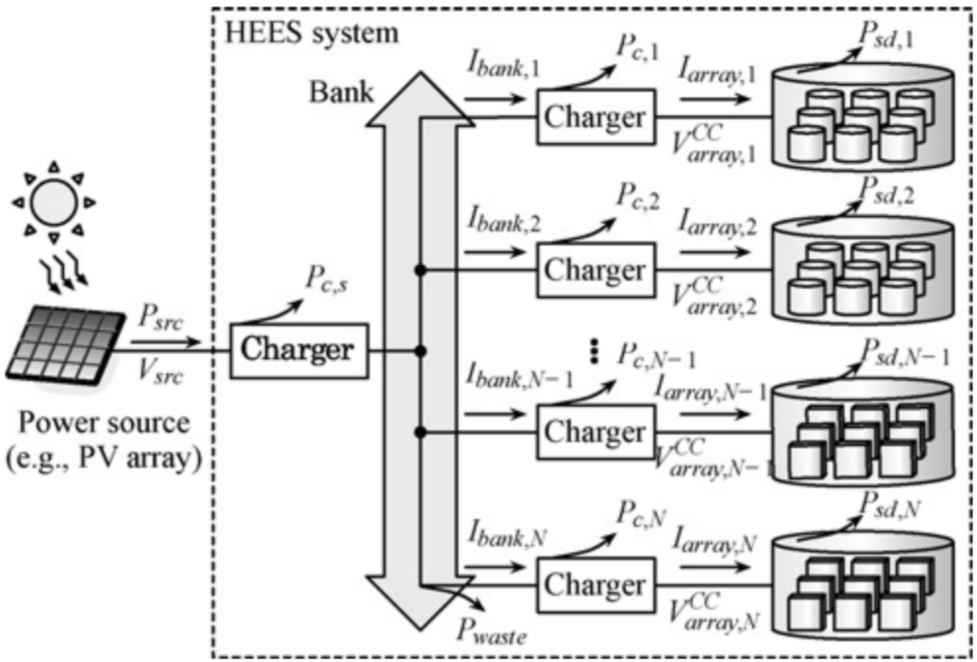
- Battery may be modeled as a voltage source in series with a resistance.
- internal resistance of battery dependent on
  - size
  - chemical properties
  - age
  - temperature
  - discharge current
  - ...



- Measurement of the internal resistance of a battery is a guide to its condition, but may not apply at other than the test conditions.
- Internal resistance increases on depletion of battery

Connecting Non-Identical Batteries in parallel

- Non-identically charged batteries have different voltages
- → Batteries will mutually discharge



Schematic of the charge allocation process in a HEES system (src. [Xie])

Self-discharge of Batteries

- Self-discharge present in all batteries due to internal chemical side reactions, internal short circuits
  - chemical reaction depend on battery technology
  - increases with charge
  - increases with temperature (Arrenhius?)
- Self-discharge in Li Ion is around 1.5-2% per month
- Self-discharge in NiMH is around 15-100% per month

Overview for today

- How to read a paper?
- A Stochastic Battery Model (Panigrahi, D. et al.)
- Battery-aware scheduling (Luo, J. et al.)

How to read a paper?

3-Pass Approach:

1. Pass

- read title, abstract, conclusions
- read section headings
- glance at equations

2. Pass

- read paragraphs in detail
- make notes
- ignore technical details (proofs, appendix, etc.)

3. Pass

- read carefully, word-by-word if necessary
- make an attempt to virtually re-implement the paper

(src.: [42])

Recap: Rate dependent battery capacity

Rate: defines how fast the battery is discharged

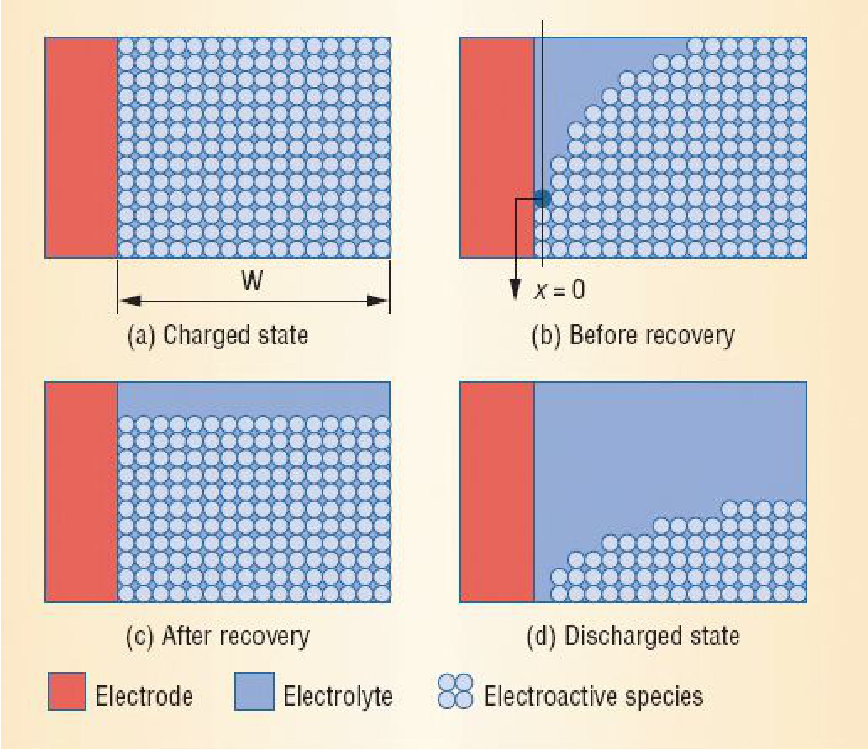
Shown is the mechanism that defines rate-dependent capacity

a) charged state

b) before recovery

c) after recovery

d) discharged state



(Src: [Rao03])

Recap: Battery Models – Comparison –

Model	Temperature effect	Capacity fading	Accuracy	Computational complexity	Configuration effort	Analytical insight	Applications
Physical							
Lithium-polymer-insertion cell (Doyle et al.)	Yes	Yes; support for Arrhenius temperature dependence and cycle aging added by Rong and Pedram	Very high	High	Very high (> 50 parameters)	Low	
Empirical							
Peukert's law	Yes; needs recalibration for each temperature	No	Medium (14% average error for constant load, 8% average error for interrupted and variable loads)	Low	Low (2 parameters)	Low	
Battery efficiency (Pedram and Wu)	Yes; needs recalibration for each temperature	No	Medium	Low	Low (2 parameters)	Low	Design of interleaved dual-battery power supply; load splitting for maximum lifetime of multibattery systems
Weibull fit (Syracuse and Clark)	Yes	No	Medium	Low	Low (3 parameters)	Low	

Recap: Battery Models – Comparison (cont'd)

Abstract							
Electrical-circuit (Gold)	Yes	Yes	Medium (12% error predicting cell voltage and thermal characteristics, 5% error predicting cycle aging)	Medium	Medium (> 15 parameters)	Medium	
Electrical-circuit (Bergveld et al.)	Yes	No	Medium	Medium	High (> 30 parameters)	Medium	Thermostatic charge method: high charging efficiency
Discrete-time (Benini et al.)	Yes	No	Medium (1% compared to Hspice continuous-time model)	Medium	Medium (>15 parameters)	Medium	Dynamic Power Management; multibattery discharge
Stochastic (Chiasserini and Rao)	No	No	High (1%)	Low	Low (2 parameters)	Medium (stochastic model of load pattern assumed)	Shaping load pattern to exploit charge recovery
Mixed							
Analytical high-level (Rakhmatov et al.)	No	No	High (5%)	Medium	Low (2 parameters)	High	Task scheduling by sequencing and *V/f* scaling; analysis of discharge methods for multibattery systems
Analytical high-level (Rong and Pedram)	Yes	Yes	High (3.5%)	Medium	Medium (> 15 parameters)	High	

Stochastic Battery Model

Idea:

- battery-life estimation of HW/SW embedded systems
- exploration of alternative implementations

Definitions

- charge unit: smallest amount battery may be discharged with
- T: number of maximum available charge units
- N: nominal capacity of charge units (nominal: for very small currents). In practice: N << T
- N, T vary dependent upon battery and discharge current
- State of charge is tracked via a discrete time transient stochastic process

(src.: [Pani01])

Side Remark: Fundamental Electricity Terms

- Voltage 1 Volt
- Current 1 Ampere
- Charge 1 Coulomb = 1 Ampere * 1 Second
- Capacity 1 Farad = 1 Coulomb / 1 Volt
- Energy 1 Joule = 1 Newton * 1m = 1 Volt * 1 Coulomb

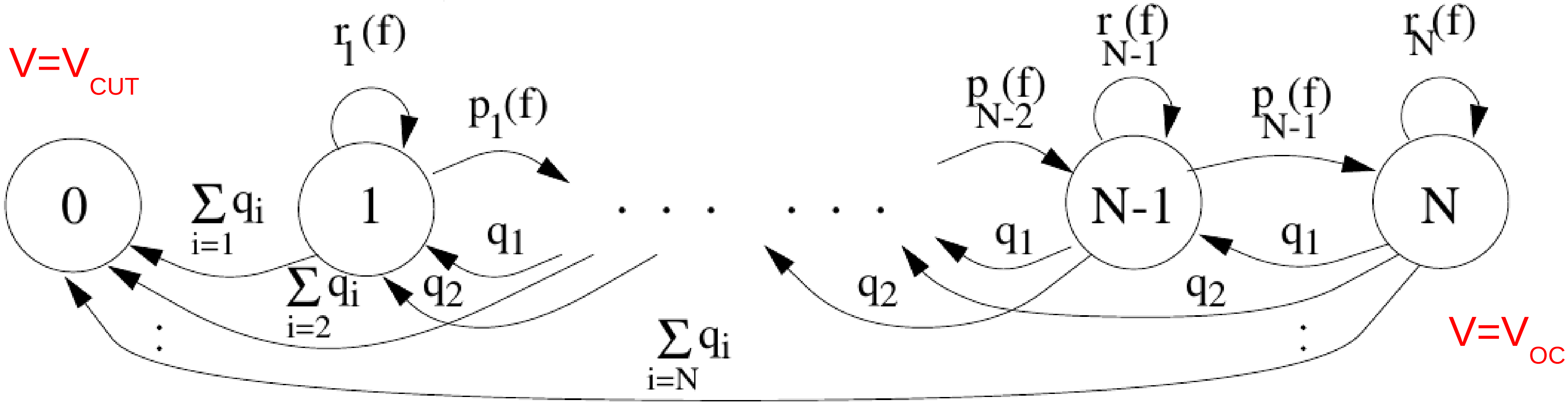
What is wrong here?

„We define the smallest amount of capacity that may be discharged as a charge unit“

Notations and Definitions

V_{OC}	open-circuit potential	initial potential of a fully charged cell wo/ load
V_{CUT}	cut-off potential	potential at which the cell is considered discharged
	theoretical capacity	[Ah]
	nominal capacity	can be achieved when discharging at rated current
C_{rated}	rated current	the nominal capacity is determined by discharging with C_{rated}
	battery lifetime	time until fully charged cell reaches V_{CUT}
	delivered specific energy	delivered energy over weight of battery
T	maximum „charge units“	
N	nominal „capacity“	N<<T
q_i	probability of demanding i charge units in time slot	

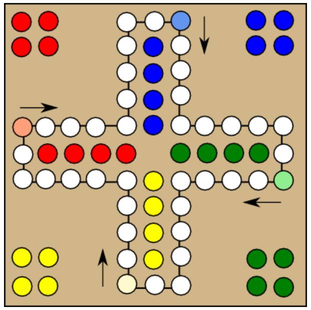
Stochastic Battery Model



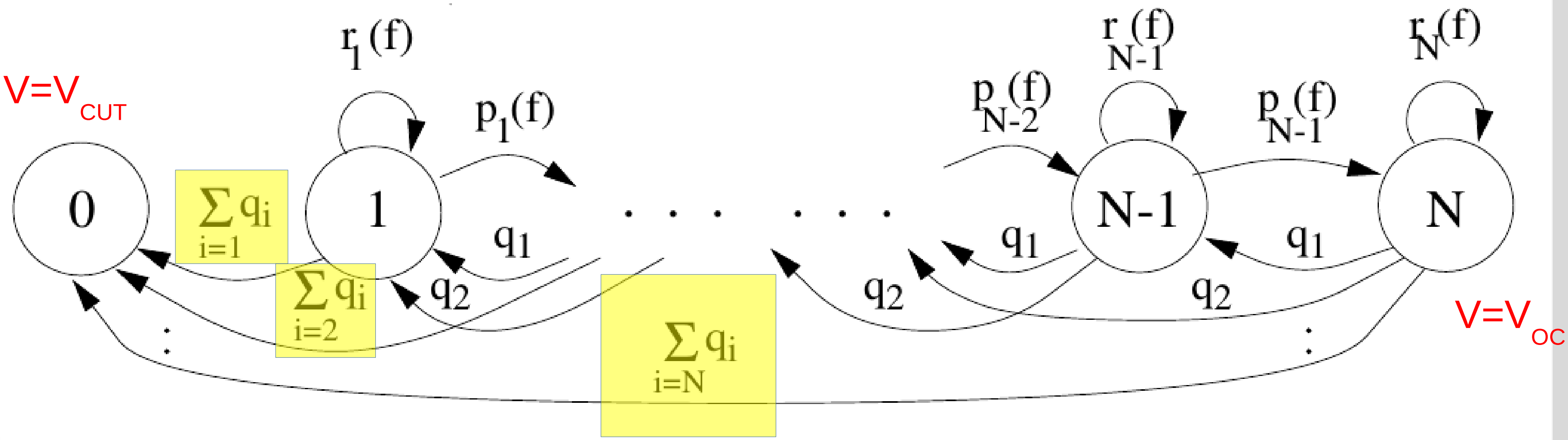
- Stochastic process starts from state of full charge ($V = V_{oc}$), denoted by N
- At each time unit, the state of charge decreases from state z to $z\text{-}n$ with n being the charge units demanded from the battery
- On the other side: if no charge units are demanded,
  battery may recover → state of charge z may increase
- Stochastic process stops at absorbing state ($V = V_{cut}$) OR
  the max available capacity T is reached.
- Allowing idling periods between discharges → battery recovers and # of charge units drained before reaching state *0* is greater than N

(src: [Pani01])

„discrete-time transient stochastic process“



Stochastic Battery Model



What is the problem here?

„Let us define q_i to be the probability that in one time unit, called slot, i charge units are demanded"

(src: [Pani01])

Stochastic battery model (cont'd)

Recovery Process

Is represented as a decreasing exponential function of the state of the battery (i.e. it is the smaller, the smaller the remaining charge of the battery is)

During discharge, different phases can be identified:

Each phase *f, (f=0, … , f_max)* starts right after ***d_f* charge units have been drained** from battery and ends when the amount of discharged capacity reaches *d_(f+1)* charge units

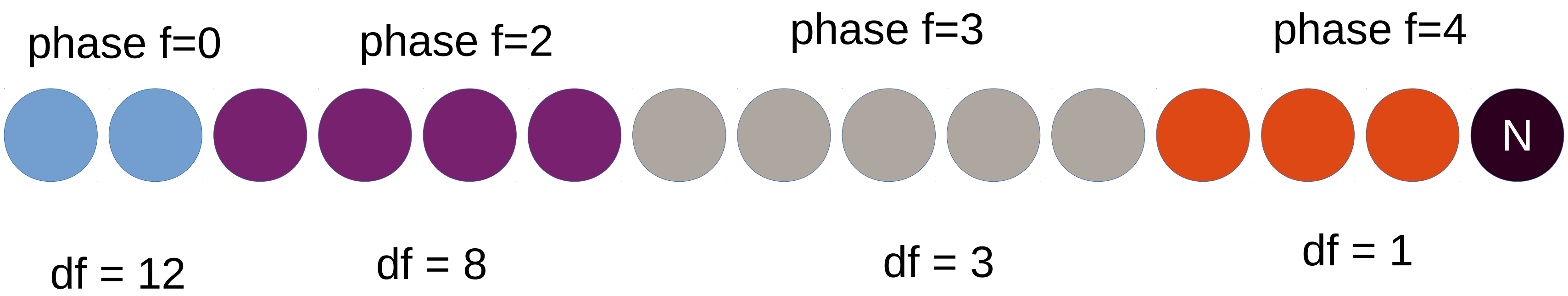
Probability of recovering 1 charge unit in a time slot dependent upon state *j (j=1, …, N-1)* and phase *f* is

$$p_j(f) = \begin{cases} q_0 e^{-g_N(N-j)-g_C(f)} & f = 0 \\ q_0 e^{-g_N(N-j)-g_C(f)d_f} & f = 1, \dots, f_{max} \end{cases}$$

(src: [Pani01])

g_N, g_C - parameters that depend upon the capability of recovery of the battery; a small g_N represents high cell conductivity (high recov. capability) and a large g_N represents high internal resistance.

Stochastic Battery Model



Stochastic Battery Model

probability of recovering 1 charge unit per time interval

$$p_j(f) = \begin{cases} q_0 e^{-g_N(N-j)-g_C(f)} & f = 0 \\ q_0 e^{-g_N(N-j)-g_C(f)d_f} & f = 1, \ldots, f_{max} \end{cases}$$

q_0 probability of idle discharging 0 charge units per interval

g_N constant

$g_C = g_C(f)$

(src: [Pani01])

Stochastic battery model (cont'd)

g_C - is related to the voltage drop of the battery cell during discharge

q_0 - is probability of an idle slot

There is a **probability to remain in the same state** when discharged (due to the recovery effect):

$$\begin{aligned} r_j(f) &= q_0 - p_j(f) \qquad j=1,\ldots,N-1 \\ r_N(f) &= q_0 \,. \end{aligned}$$

(src: [Pani01])

Assumption: g_N is constant;

g_C is a piecewise constant function of the number of charge units already drawn off the cell; it changes value in correspondence with d_f (f = 1, … , f_max). It is d_0=0 and d_(f-max+1)=T. Proper values are chosen according to the battery

Stochastic battery model (cont'd)

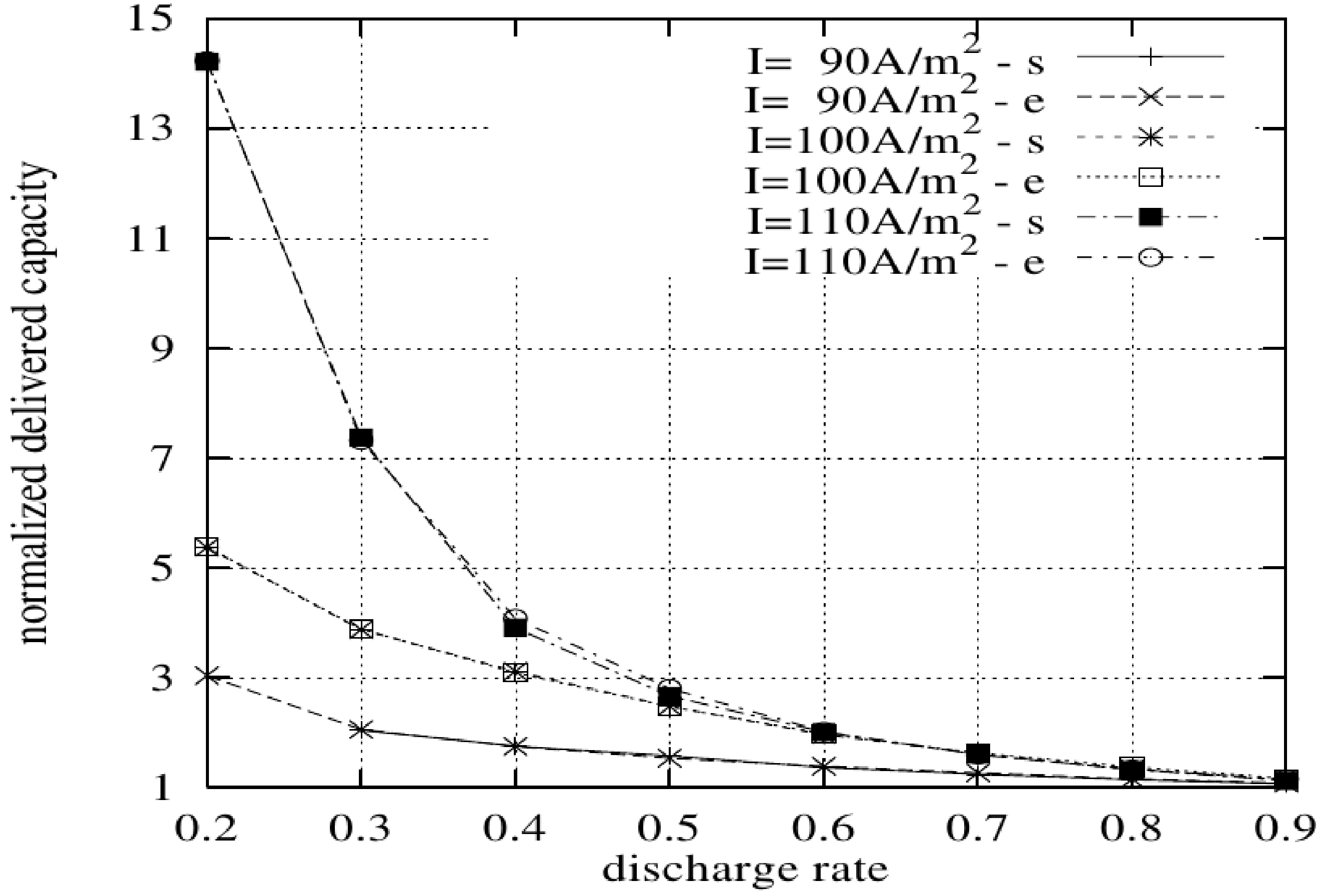
```
Simulation_Step
inputs:        Current_State,Recovery_Probability[],
Discharge_Rate
outputs: Next_State
begin
    Generate a random number R between 0 and 1;
    If (R < Discharge_Rate) then
        Next_State := Current_State - 1;
    else if ( R < Recovery_Probability[Current_State]) then
        Next_State := Current_State + 1;
    end if
end
```

Bernoulli arrival

(src: [Pani01])

What is the problem here?

Stochastic battery model (cont'd)

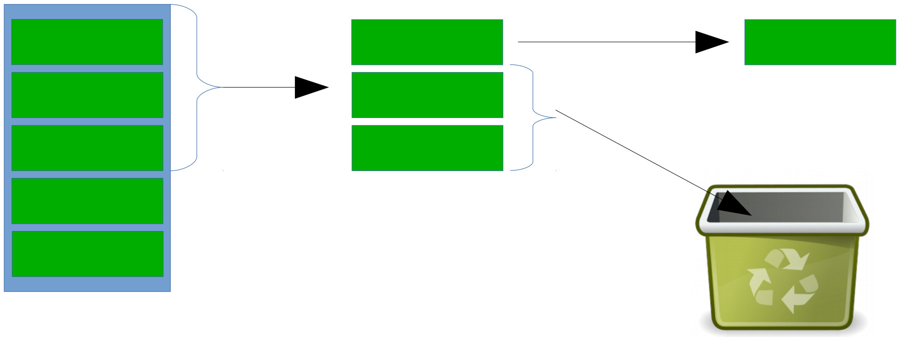


„It can be seen that curves obtained from the PDE and the stochastic models match closely"

(src: [Pani01])

Stochastic battery model (cont'd)

- next step: make battery model deterministic
- introduce battery efficiency to account for Rate Capacity effect
- build efficiency LUT using PDE model



Stochastic battery model (cont'd)

```
Simulation_Step
inputs: Current_State, Current_Demand, Recovery_Probability[], Efficiency_Table[]
outputs: Next_State
variables: Actual_Demand
begin
    Generate a random number R between 0 and 1;
    Actual_Demand := Efficiency_Table[Current_Demand];
    If (Current_Demand > 0) then
        Next_State := Current_State - Actual_Demand;
    else if ( R < Recovery_Probability[Current_State]) then
        Next_State := Current_State + 1;
    end if
end
```

(src: [Pani01])

Stochastic battery model (cont'd)

Table 3: Estimation of Battery Life and Delivered Energy Using Stochastic Model

System	Rate Capacity Effect		Recovery Effect		Rate Capacity & Recovery Effect		
	Delivered Spec.Energy (Wh/Kg)	Life Time (ms)	Delivered Spec. Energy (Wh/Kg)	Life Time (ms)	Delivered Spec. Energy (Wh/Kg)	Life Time (ms)	Packets Processed
SYS1	1.369	16875	13.357	163650	1.369	16875	20250
SYS2	3.754	67717	15.553	280543	3.754	67717	81260
SYS3	2.858	88383	32.924	1115616	4.974	153817	92290

Table 4: Comparison with PDE model : Speed and Accuracy

System	Delivered Spec. Energy (Wh/Kg)			Life Time (ms)			Computation Time	
	STOC	PDE	% Err	STOC	PDE	%Err	STOC	PDE
SYS1	1.36	1.33	2.25	16785	17264	2.85	18.62 sec	>1 Day
SYS2	3.75	3.79	1.06	67717	65723	2.94	19.52 sec	>1 Day
SYS3	4.97	5.07	2.01	153817	154956	1.00	40.35 sec	>2 Days

(src: [Pani01])

Battery-aware scheduling

Battery-aware scheduling

Idea:

- adjust task schedule such that battery's capacity as a function of current distribution is taken into consideration (see [Luo01])
- Basis is the following equation:

$$p^{act} = \int dI \, \frac{V \cdot I}{c(I)} \cdot \hat{P}(I)$$

(src: [Pedram99])

- V : voltage (assumed constant)
- I : actual current drawn (piece-wise constant)
- c(I) : utilization factor

  (i.e. ratio of battery capacity at discharge current I to standard capacity.) May be expressed through Peukert's law:

  $c(I) = k / I^b$ (normalized)

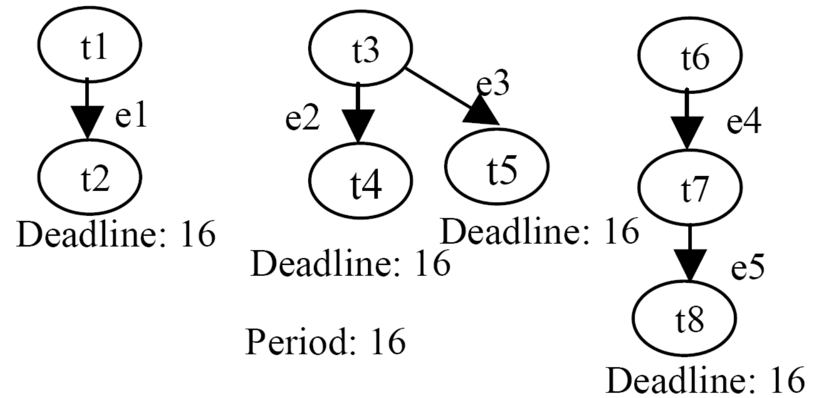
- $\hat{P}(I)$ is the probability density function of I (a measure of how evenly the value of the current is distributed)

Battery-aware scheduling (cont'd)

- goal: extend battery lifespan
- means: schedule transformations
- terms:
  - task graph
  - schedule
  - PE
- assumptions:
  - 2 PEs are connected via 1 bus
  - Intra-task communication costs are 0
  - Power drawn during each task execution is constant
  - Notion: tx (y) means:

    task ‘x’ has power consumption

    of ‘y’ units



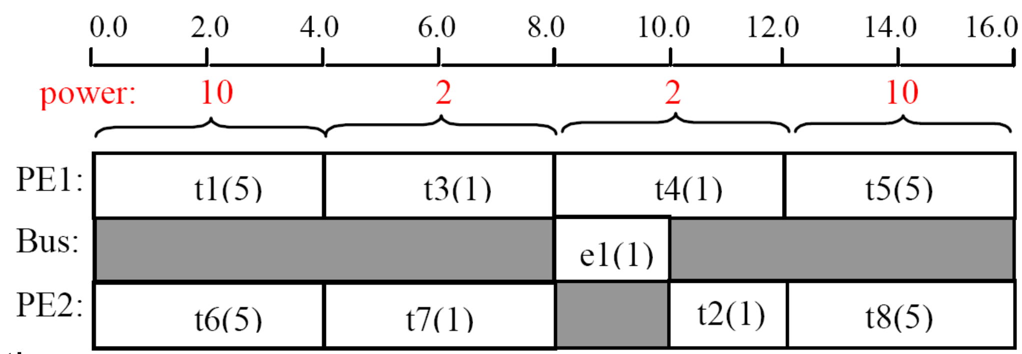
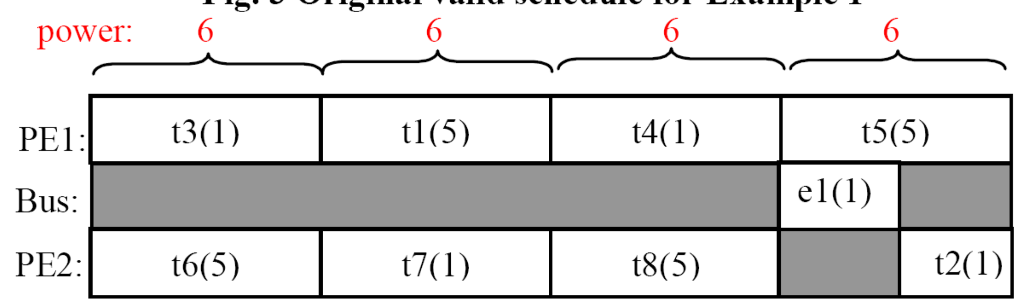


Fig. 3 Original valid schedule for Example 1



Battery-aware scheduling (cont'd)

Example 1: (cont'd)

Two different valid schedules are shown

Using equations 1 and 2 (and appropriate parameters) it turns out that the lower schedule is 15% more power efficient

=> obviously equations 1 and 2 can be used in a cost function of a schedule to minimize the power consumption through considerations of battery effects

Example 2:

Same assumptions as before except for

t1, t3, t4, t5, t7 -> 0.2sec worst-case execution time (WCET)

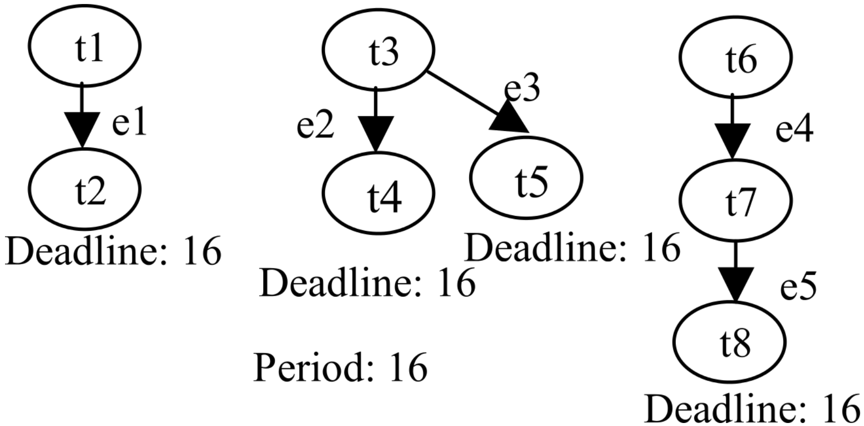
T2, t6 -> 0.3sec

Edges (communication) e1, e2 -> 0.1sec

Task graph as shown on next slide

P (average) of each task is 1 unit; … of each edge is 0.2units

Battery-aware scheduling (cont'd)



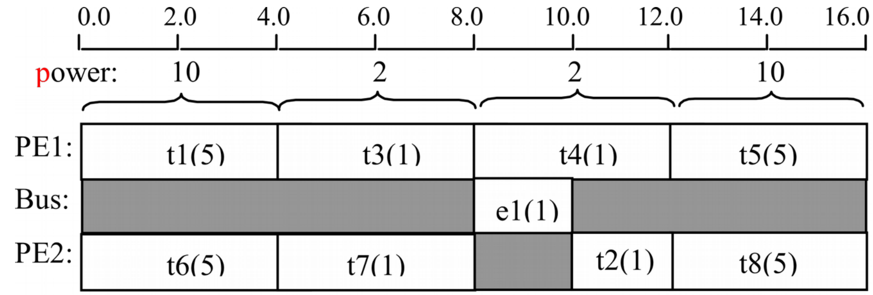


Fig. 3 Original valid schedule for Example 1

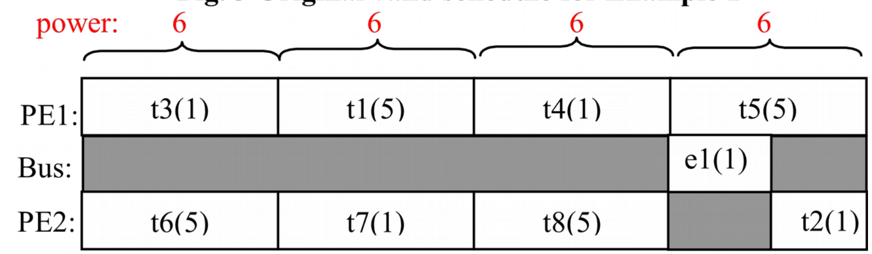


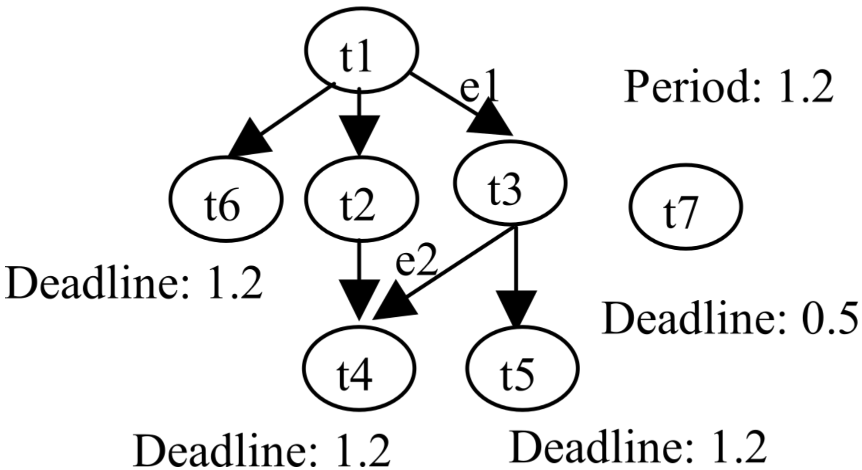
Fig. 4 New valid schedule for Example 1

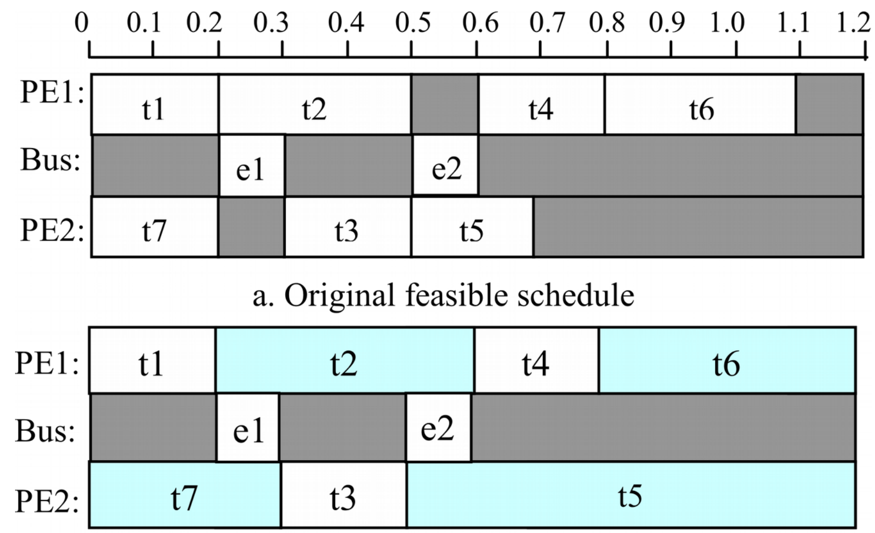
- Total power is the same,
- but power density is different
- second schedule: 15% improvement in power-drawn

$$p^{act} = \int dI \, \frac{V \cdot I}{c(I)} \cdot \hat{P}(I)$$

(src.: [Luo01])

Battery-aware scheduling (cont'd)





a. Original feasible schedule

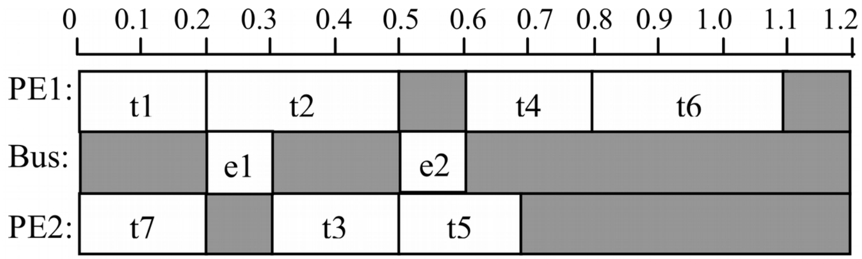
b. Corresponding variable-voltage schedule

- first schedule: ASAP
- voltage scaling extends execution time to latest finish time

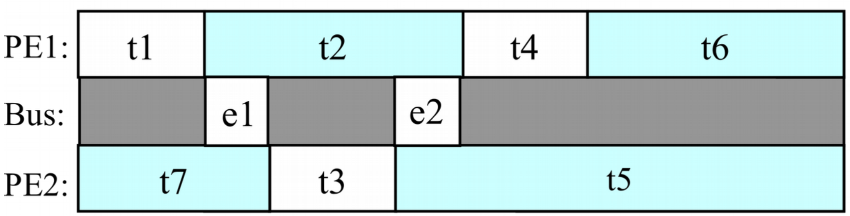
blackboard

(src.: [Luo01])

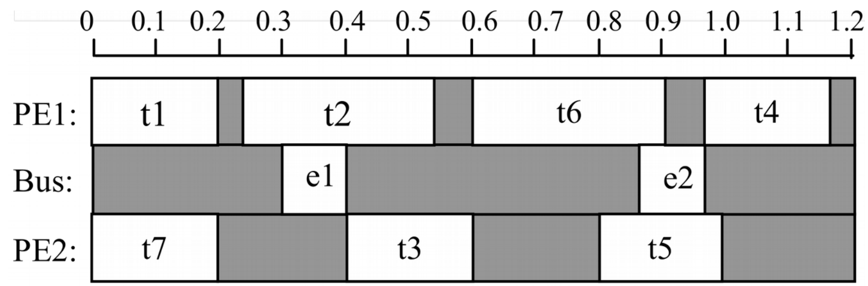
Battery-aware scheduling (cont'd)



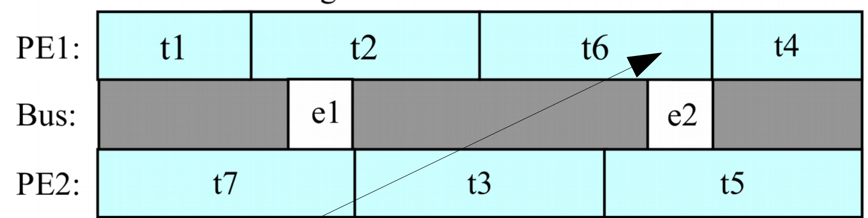
a. Original feasible schedule



b. Corresponding variable-voltage schedule



a. Original feasible schedule



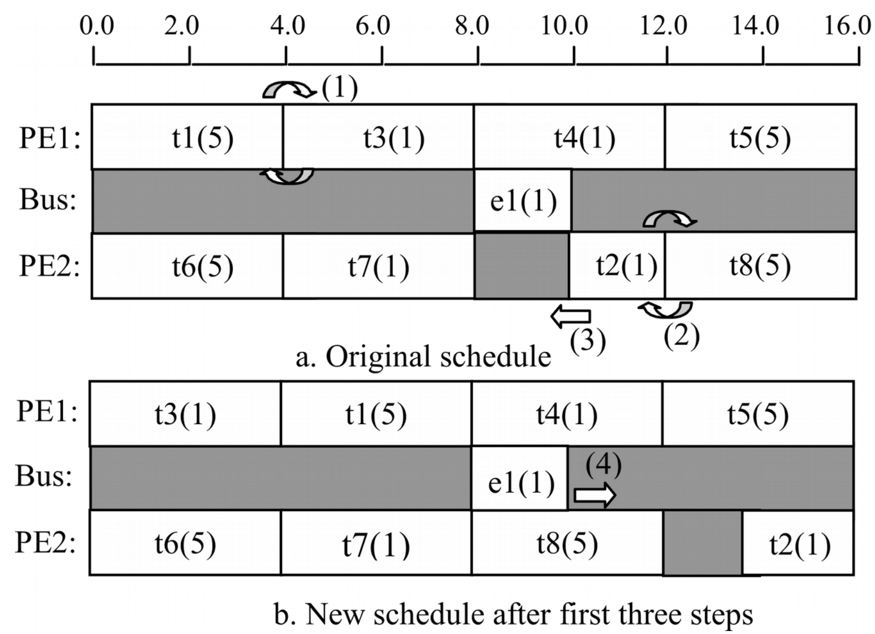
- even more improvement possible with slot shifting and swapping

(src.: [Luo01])

Battery-aware scheduling (cont'd)

- battery-aware improvements of slack-based list scheduling
- reduce actual power

$$p^{act} = \frac{1}{hyperperiod} \int_0^{hyperperiod} dt \, \frac{p(t)}{c_p(t)}$$



(src.: [Luo01])

Demo: Linux power supply class

Demo: Linux power supply class

- class used to represent battery, UPS, AC or DC power supply
- available via sysfs (/sys/class/power_supply)
- All voltages, currents, charges, energies, time and temperatures in µV, µA, µAh, µWh, seconds and tenths of degree Celsius unless otherwise stated.
  - CHARGE_* attributes represents capacity in µAh only.
  - ENERGY_* attributes represents capacity in µWh only.
  - CAPACITY attribute represents capacity in **%**, from 0 to 100.

Demo: Linux power supply class (cont'd)

- Alos possible to query battery information on higher abstraction levels:
- **ACPI**
  - sudo apt-get install acpi
  - watch --interval=5 acpi -V
- **Upower**
  - ...

Sources

[Piguet04] Ch. Piguet (Ed.), “Low Power Electronics Design”, CRC Press, ISBN 0-8493-1941-2, 2004.

[Park05] Chulsung Park, Jinfeng Liu, Pai H. Chou, “B#: A Battery Emulator and Power-Profiling Instrument”, IEEE Design & Test of Computers, Volume: 22 , Issue: 2, pp.150 - 159, Feb. 2005.

[Luo01] Luo, J. Jha, N.K., Battery-aware static scheduling for distributed real-time embedded systems, IEEE/ACM Proc. Of Design Automation Conference (DAC’01), pp.444 – 449, June 2001.

[Pani01] Panigrahi T,D., Panigrahi D., Chiasserini, C., Dey, S., Rao R., Raghunathan A., Lahiri K., "Battery life estimation of mobile embedded systems", 14th. IEEE VLSI Design International Conf. 2001, pp.57-63, 2001.

[Rao03] Rao, R. Vrudhula, S. Rakhmatov, D.N, “Battery modeling for energy aware system design”, IEEE ComputerMagazine, Dec. 2003, Volume: 36 , Issue: 12 , pp.77 – 87.

[Benini01] Benini, L. Castelli, G. Macii, A. Macii, E. Poncino, M. Scarsi, R, “Discrete-time battery models for system-level low-power design”, IEEE Tr. on Very Large Scale Integration (VLSI) Systems, Volume: 9, Issue: 5, pp.630-640, 2001.

[Rakh01] Rakhmatov, D.N.; Vrudhula, S.B.K.; "An analytical high-level battery model for use in energy management of portable electronic systems", IEEE/ACM International Conference on CAD (ICCAD2001), 4-8 Nov. pp.488-493, 2001.

[Pedram99] Pedram, M.; Qing Wu; "Design considerations for battery-powered electronics", IEEE/ACM Proc. of 36th. Design Automation Conference (DAC99), pp.861-866, 1999.

[42] http://blizzard.cs.uwaterloo.ca/keshav/home/Papers/data/07/paper-reading.pdf