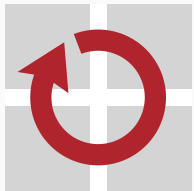
Energy-Consumption Modeling of Embedded Devices

A brief introduction to two different approaches

13th December, 2023

Martin Ottens

Friedrich-Alexander-Universität Erlangen-Nürnberg



Lehrstuhl für Verteilte Systeme
und Betriebssysteme

FRIEDRICH-ALEXANDER
UNIVERSITÄT
ERLANGEN-NÜRNBERG

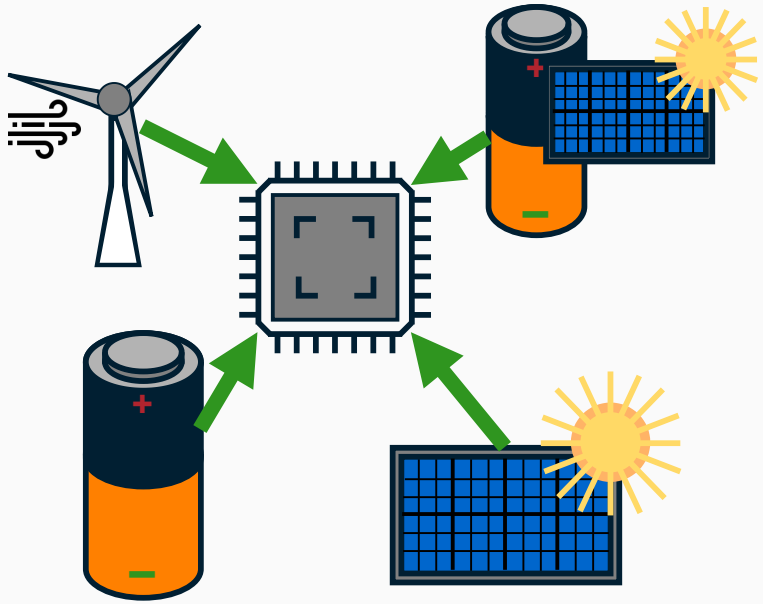
TECHNISCHE FAKULTÄT

Table of Contents

Introduction

Why do we need Energy Modelling for Embedded Devices?

- Embedded systems with ultra-low-power processors in a growing number of applications
  - IoT devices, wearables, ...
- Energy supply via battery and/or energy harvesting
- Required size of energy supply also depends on power consumption of the processor



→ Energy requirements needed to optimize size, weight and costs

Worst-Case Energy and Power Consumption

Analysis to determine **safe and accurate upper bounds** for ...

Worst-Case Energy Consumption (WCEC)

Maximum total energy consumption of a full program run

Peak Power Requirements (PPR)

The highest peak in energy consumption during execution

The basis for the analysis is always

- a specific program
- running on a processor with known characteristics

Where is the challenge?

WCEC and PPR depend on different factors:

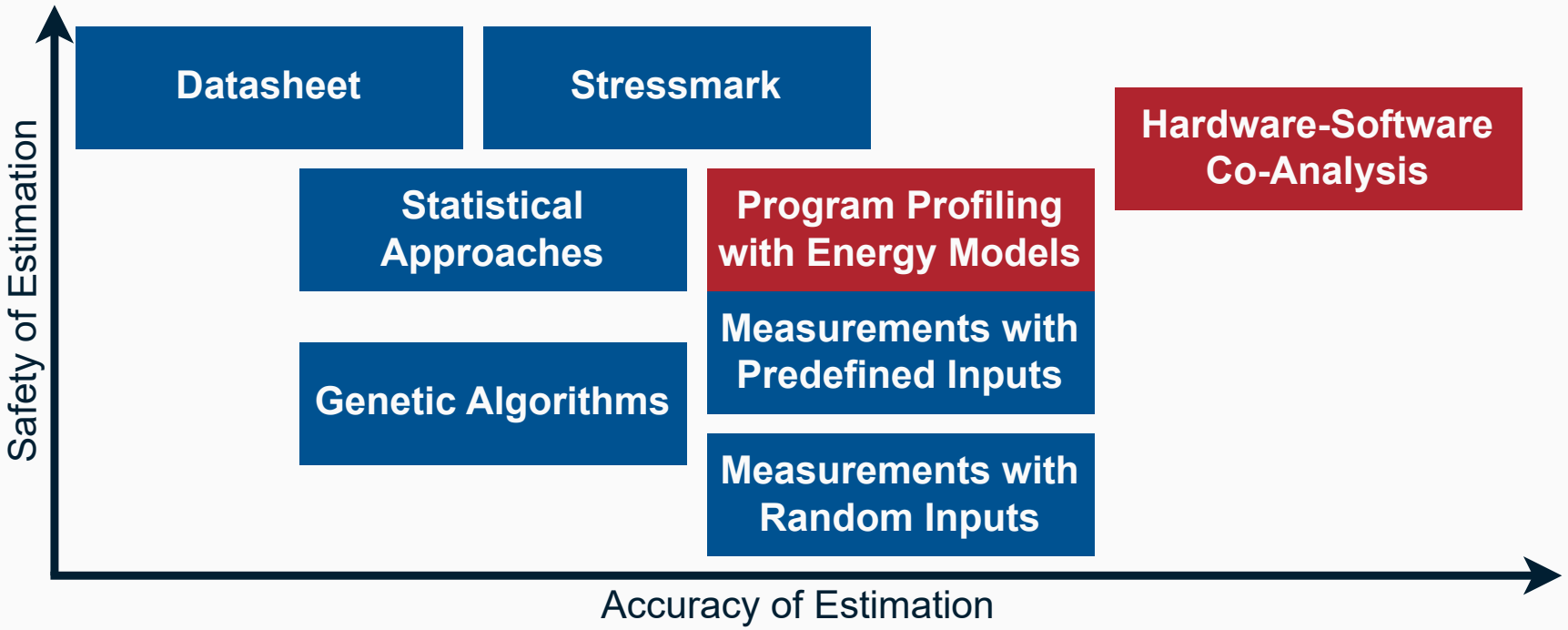
- Program
  - Control flows and instruction sequences
  - Input data $\rightarrow$ all possible combinations
- Processor
  - Design and basic electrical properties
  - Internal processor states $\rightarrow$ in every execution cycle
- Complex dependencies

Exact determination of WCEC and PPR is often impossible

$\rightarrow$ **Analysis methods focus on safe and accurate estimations**

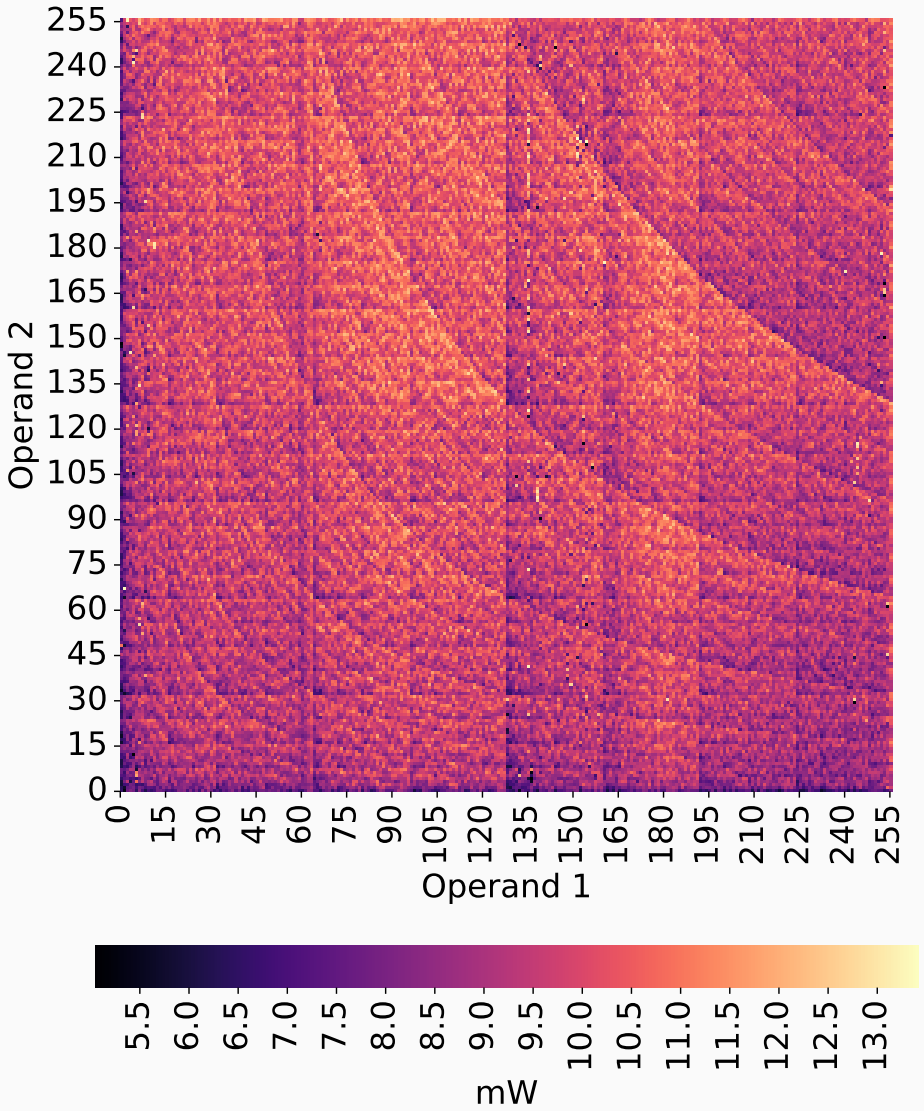
Previous Approaches

Several different hardware-focused and software-focused approaches to estimate the WCEC and PPR:

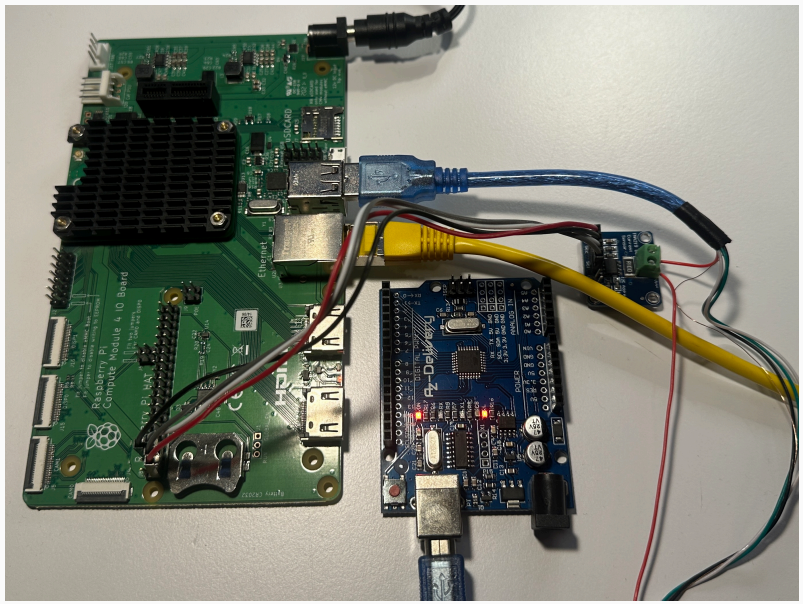


Instruction and Data Dependent Modeling

Data Dependent Power Consumption – An Experiment



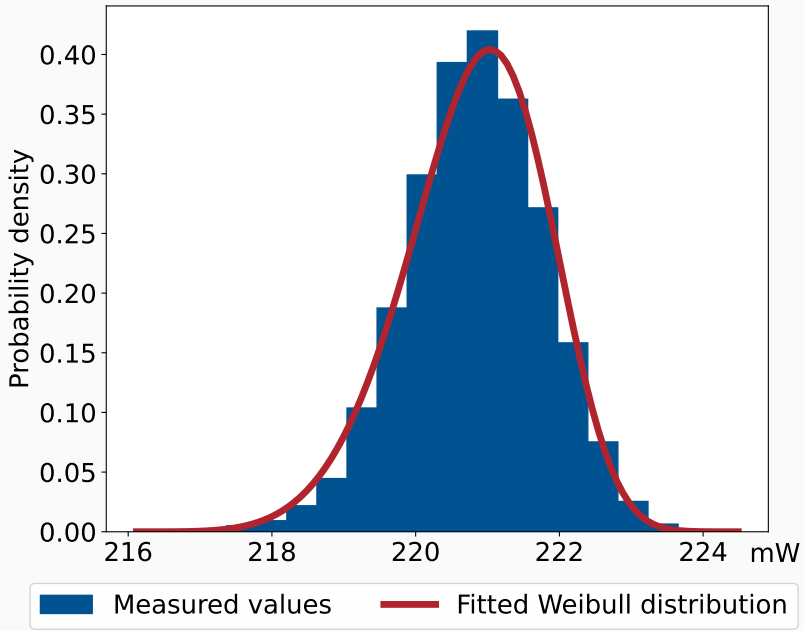
- 8-bit AVR *Atmel ATmega328P* development board
- Power consumption during `mul` instructions with different operand values



Instruction and Data Dependent Modeling

- Power consumption of an instruction depends on:
  - Input data
  - Previous instruction
- Model power consumption of instruction pairs using probability distributions

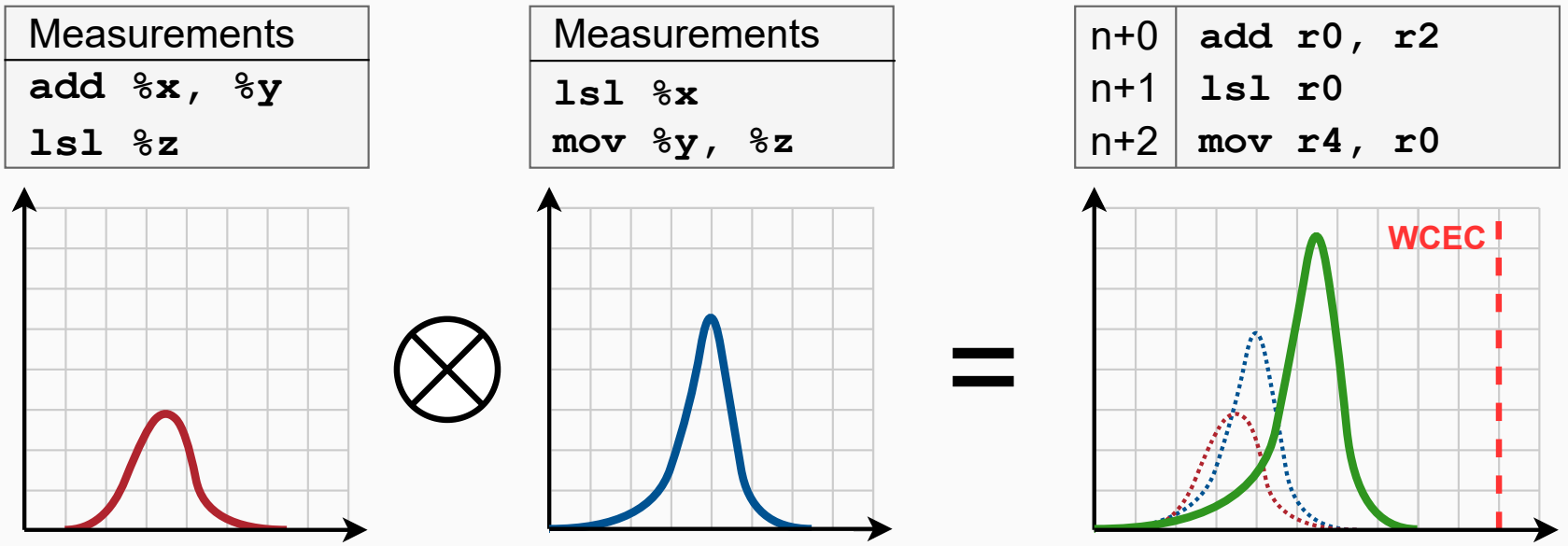
→ **Weibull distribution**



J. Pallister, S. Kerrison, J. Morse, and K. Eder.
Data dependent energy modeling for worst case energy consumption analysis.
In *Proceedings of the 20th International Workshop on Software and Compilers for Embedded Systems*, pages 51–59, 2017.

Composition and Full Program Analysis

- Distributions for all instruction pairs measured using a large set of random input data
- Convolute distributions for all pairings in program flow
- WCEC estimations can be calcultated using 99^{th} percentile



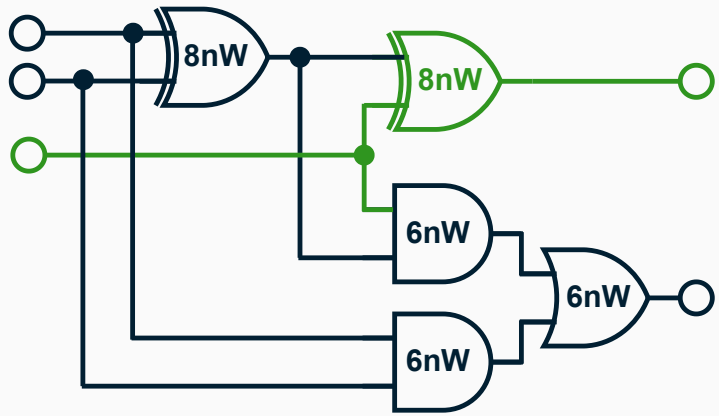
Instruction and Data Dependent Modeling – Conclusion

+ Approach allows **static analysis**
  $\rightarrow$ Measurements required only once for a processor model
+ Analysis with given distributions is comparatively simple
– Measurements for large instruction sets are **time-consuming**
  $\rightarrow$ Unclear scalability
– Data dependencies between instructions ignored
  $\rightarrow$ Safe but potentially **overestimated WCEC**

Symbolic Simulation Based Energy Modeling

Symbolic Simulation Based Energy Modeling

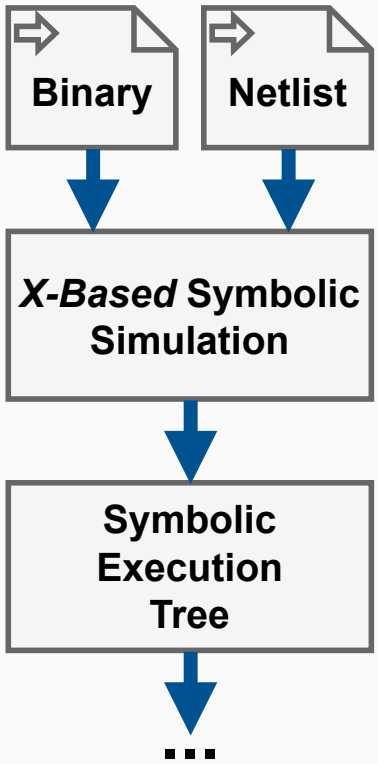
- Simulation of all possible gate switching activities
- Co-Simulation of Hardware and Software
  - using a full program binary
  - and the processor netlist
- Independet from exact input data, but application-specific



H. Cherupalli, H. Duwe, W. Ye, R. Kumar, and J. Sartori.
Determining application-specific peak power and energy requirements for ultra-low-power processors.
ACM Transactions on Computer Systems, 35(3):1–33, 2017.

X-Based Symbolic Simulation

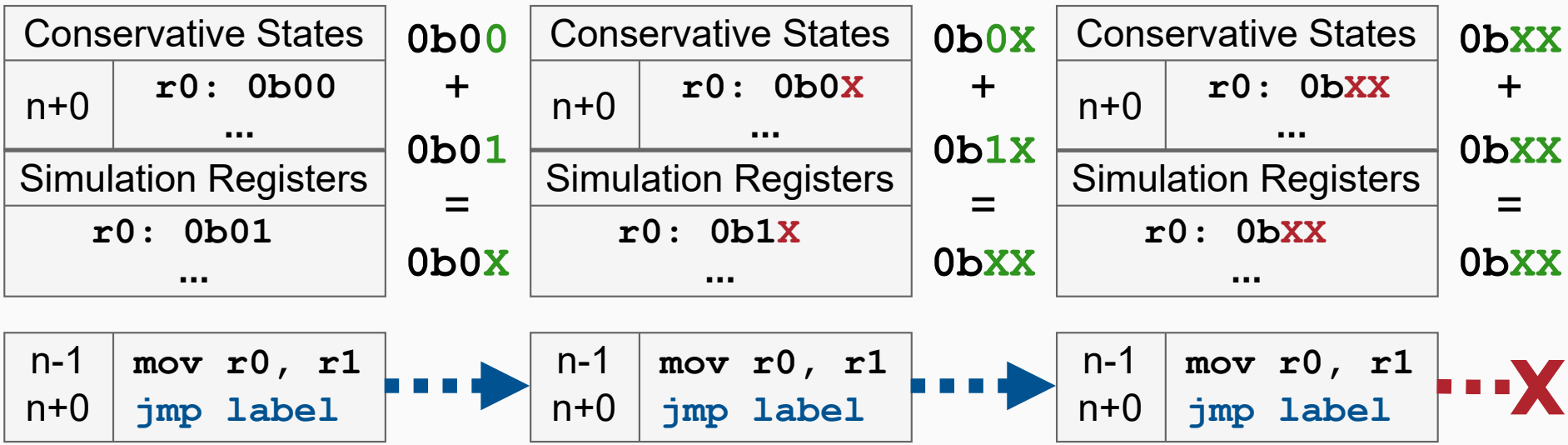
- *X-Based:* Unknown logical values (*X*s) at the processor inputs
- Simulation algorithm builds an Symbolic Execution Tree
- Cycle-accurate simulation of gate activities
- Branches: Simulate all program paths using a stack
- Execution Tree contains all reachable processor states with *X*s and explicit values



Conservative States

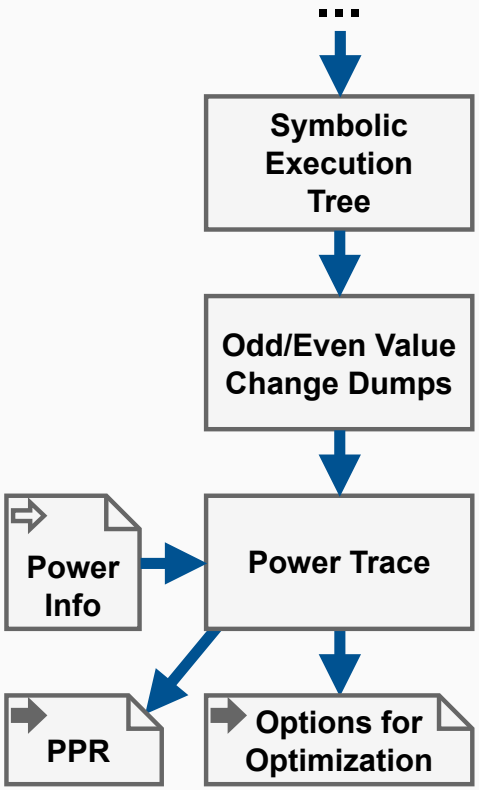
- Complex control flows such as input-dependent loops
  $\Rightarrow$ *X-Based* symbolic simulation will never terminate
- Possible processor states at jumps are typically finite

$\rightarrow$ *Conservative States* to scalable compute worst-case gate toggling activities in such cases



Estimation of PPR and WCEC

- Maximum power consumption in a cycle
  $\Leftrightarrow$ Maximum gate toggles
- Odd/Even Value Change Dumps
- Generate Power Traces using information from Standard Cell Library
- PPR = Maximum entry from Power Traces
- WCEC estimation is more complex



Cycle	n+0	n+1	n+2	n+3	n+4
Value	0	x	x	0	1

Toggles at odd cycles maximized

Cycle	n+0	n+1	n+2	n+3	n+4
Value	0	1	1	0	1

Toggles at even cycles maximized

Cycle	n+0	n+1	n+2	n+3	n+4
Value	0	0	1	0	1

Symbolic Simulation Based Energy Modeling – Conclusion

+ Approach can provide **accurate bounds** for PPR and WCEC
+ Provides indications for optimization potential of a program
– Requires detailed knowledge of **internal processor structure**
  $\rightarrow$ Processor designs are often proprietary
– *Conservative States* will introduce inaccuracies
  $\rightarrow$ Possible **Overestimation** of PPR and WCEC

Conclusion and Outlook

Conclusion and Outlook

- Approaches allow static analysis
  $\rightarrow$ Integration in design flows
- WCEC needs program path analysis
  $\rightarrow$ Progress from Worst-Case Execution Time research
- Limited to basic embedded processors
  $\rightarrow$ Challenges with non-determinism (e.g. caches)
- External system components are ignored

$\Rightarrow$ Further research and optimization needed

$\Rightarrow$ Interesting steps towards more compact & secure embedded systems

Questions?