An Energy Characterization Framework for Software-Based Embedded Systems

Lee, Donghoon
Faculty of ISEE, Kyushu Univ.

Ishihara, Tohru
System LSI Research Center, Kyushu Univ.

Muroyama, Masanori
System LSI Research Center, Kyushu Univ.

Yasuura, Hiroto
System LSI Research Center, Kyushu Univ.

他

https://hdl.handle.net/2324/6794497

出版情報：ESTIMedia2006. 1, pp.59-64, 2006-10-26. ESTIMedia
バージョン：
権利関係：

An Energy Characterization Framework for Software-Based Embedded Systems

Donghoon Lee[1] ,
Tohru Ishihara[2] , Masanori Muroyama[2] , Hiroto Yasuura[2] ,
Farzan Fallah[3]

[1] Faculty of ISEE, Kyushu Univ.
[2] System LSI Research Center, Kyushu Univ.
[3] Fujitsu Labs. of America

Outline

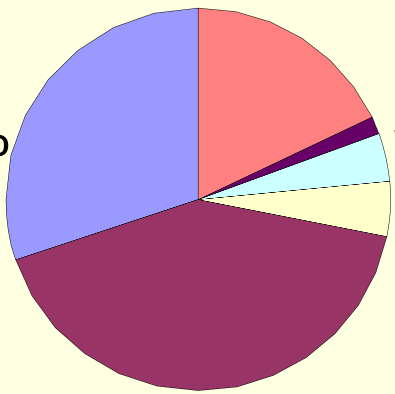
- Introduction
- Energy characterization framework
- Training bench generation
- Experimental results
- Summary

Back Ground (1/2)

Power consumption at a processor on embedded system has a big portion

Power Distribution in a PDA class sample device

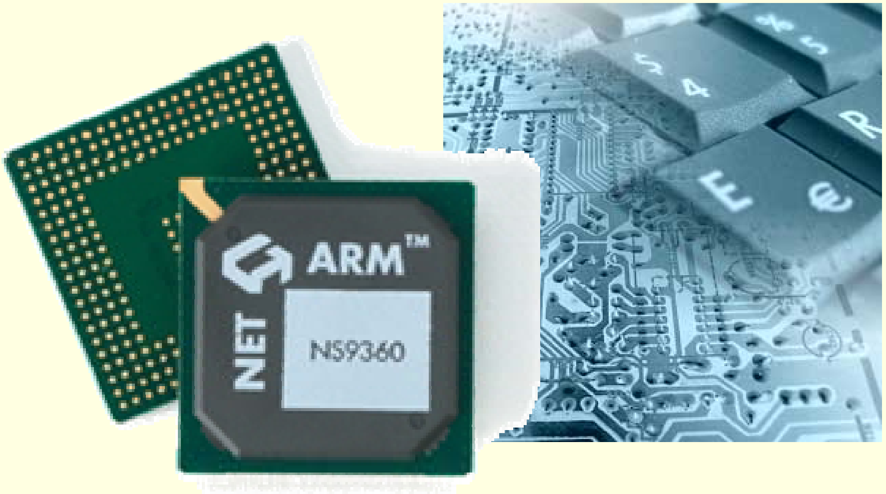
Cliff Brake, Accelent Systems, Inc. (May, 2003)



www.princeton.edu/~wolf/

Back Ground (2/2)

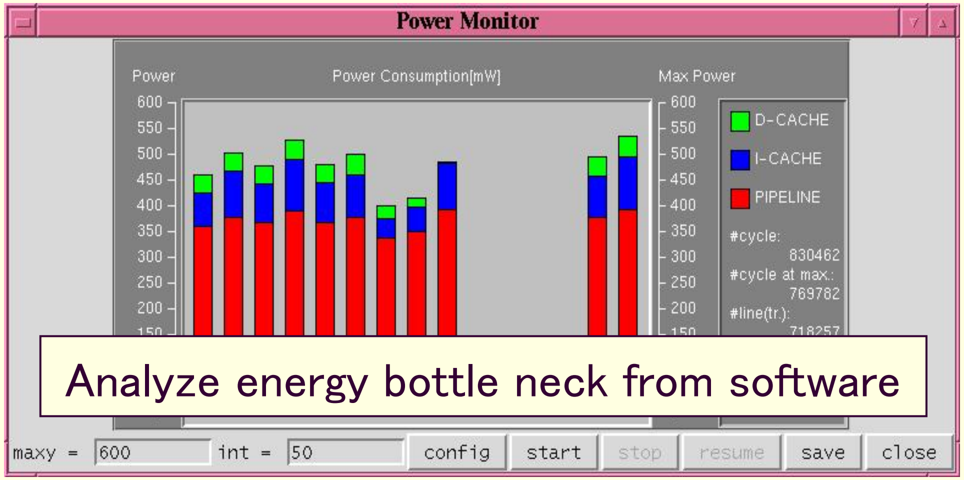
Power consumption at a processor depends on the software being executed

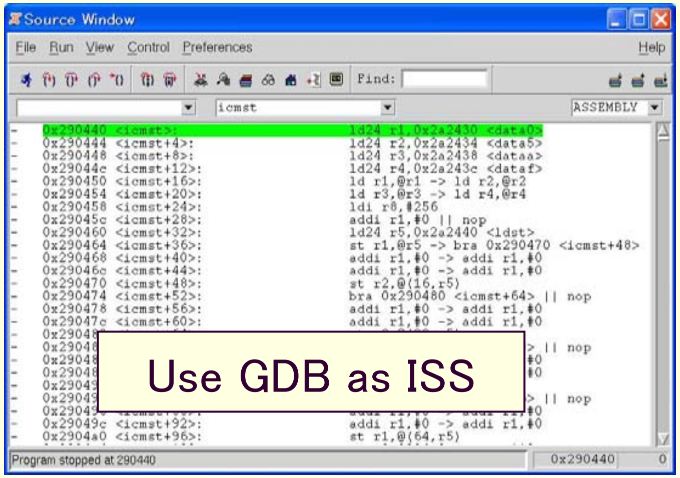


SW designer should think power dissipation by SW that he is developing

Energy Analysis Tool

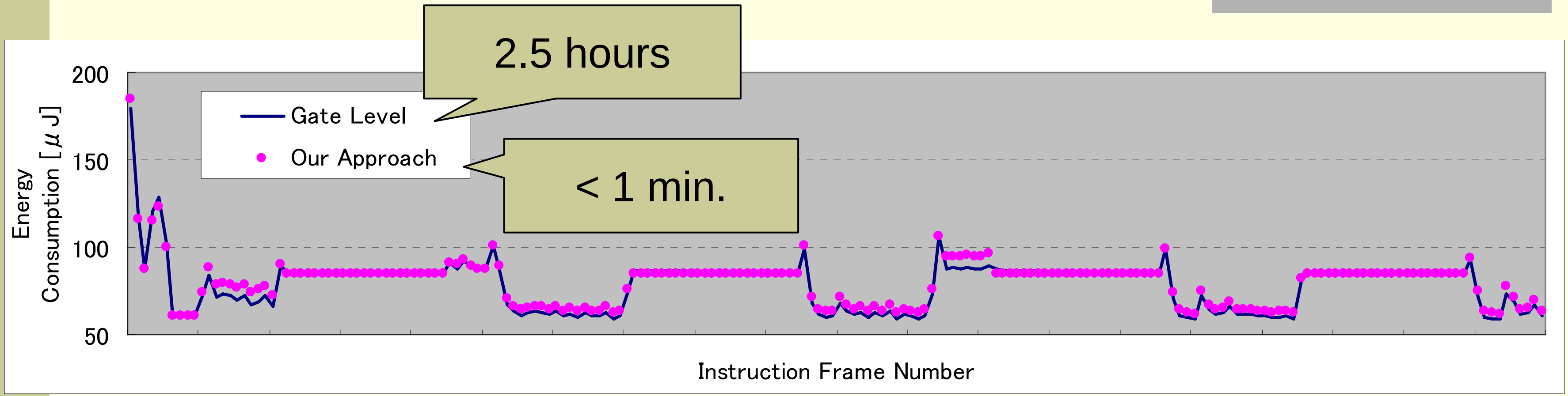
We propose a characterization technique to find a good energy model for a processor



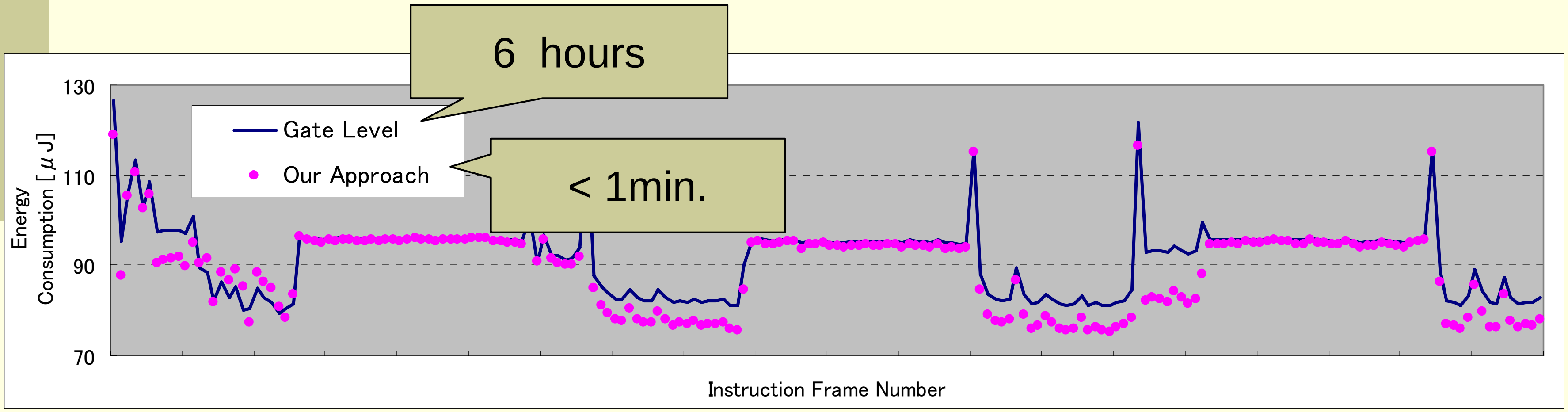


Fast, accurate, and processor-independent
Instruction level energy estimation

Experimental Result



Energy estimation for JPEG encoder executed on a M32R-II processor



Energy estimation for JPEG encoder executed on a SH3-DSP processor

Related Work (1/2)
High-level energy estimation

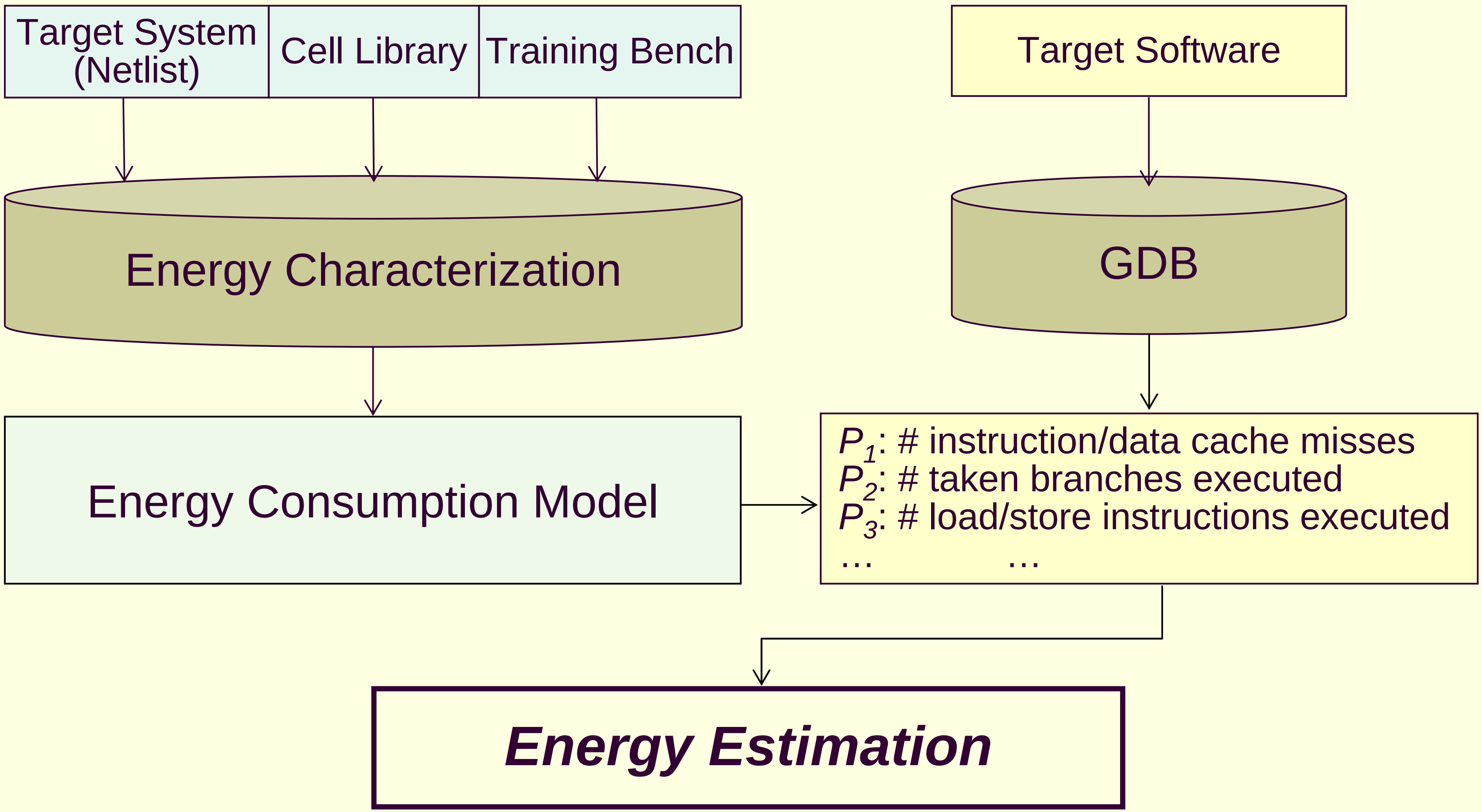
- Instruction-level modeling
  - Energy estimation by instruction-set simulator
  - Instruction level energy modeling by measuring the average power consumption of each instruction while executed in a loop

V. Tiwari, S. Malik, and A. Wolfe, "*Power analysis of embedded software: a first step towards software power minimization*," IEEE Tr. On VLSI, vol. 2, no. 4, pp. 437-445, Dec. 1994.

- Structural modeling of the underlying hardware architecture
  - Make power models by estimating capacitance on the circuit
  - Keep track of which units are accessed per cycle by cycle-level performance simulation

D. Brooks, V. Tiwari, and M. Matonosi, "*Wattch: A Framework for Architectural-Level Power Analysis and Optimization*," in Proc. Of ISCA, pp. 84-94, June. 2000.

Our Approach



Related Work (2/2)
Characterization-based energy estimation

- Characterization-based macro-modeling
  - Regression analysis to model software energy
  - Model the energy consumption using linear expression

$$E = \sum_{j=1}^{p} c_j P_j$$

- P_j : parameters of the model
- c_j : corresponding coefficients
- p : number of parameters

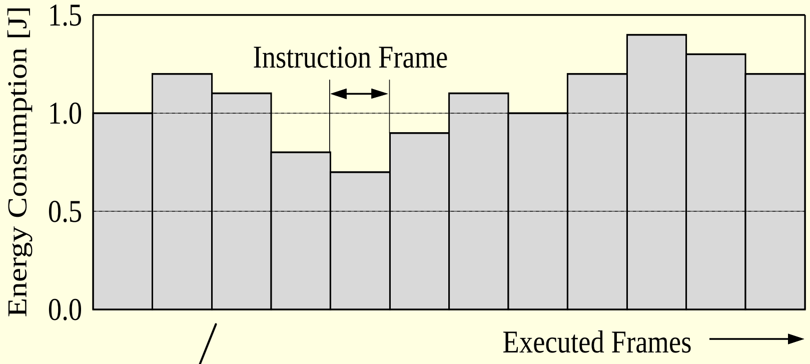
$$\mathrm{P} = \begin{pmatrix} P_{1,1} & P_{1,2} & \cdots & P_{1,p} \\ P_{2,1} & P_{2,2} & \cdots & P_{2,p} \\ \cdots & \cdots & \cdots & \cdots \\ P_{n,1} & P_{n,2} & \cdots & P_{n,p} \end{pmatrix} \qquad \mathrm{E} = \begin{pmatrix} E_1 \\ E_2 \\ \cdots \\ E_n \end{pmatrix} \Rightarrow C = \begin{pmatrix} c_1 \\ c_2 \\ \cdots \\ c_n \end{pmatrix} = \left[\mathrm{P}^T \mathrm{P}\right]^{-1} \mathrm{P}^T \mathrm{E}$$

Evaluate set of parameters	Obtain the energy consumption by low-level estimator	Obtain coefficient using regression analysis

T. K. Tan, A. Raghunathan, G. Lakshminarayana, and N. K. Jha, "*High-level software energy macro-modeling*," in Proc. Design Automation Conf., 2001, pp. 605--610.

Overview of Energy Characterization

- Energy consumption model based on linear expression
- Evaluate energy from each divided instruction frames



Extract by instruction-set simulation

Estimate energy at gate level

Instruction frame 1 ⇨ $E_1 \longleftarrow E'_1 = c_1 P_{11} + c_2 P_{12} + c_3 P_{13} \cdots$

Instruction frame 2 ⇨ $E_2 \longleftarrow E'_2 = c_1 P_{21} + c_2 P_{22} + c_3 P_{23} \cdots$

…

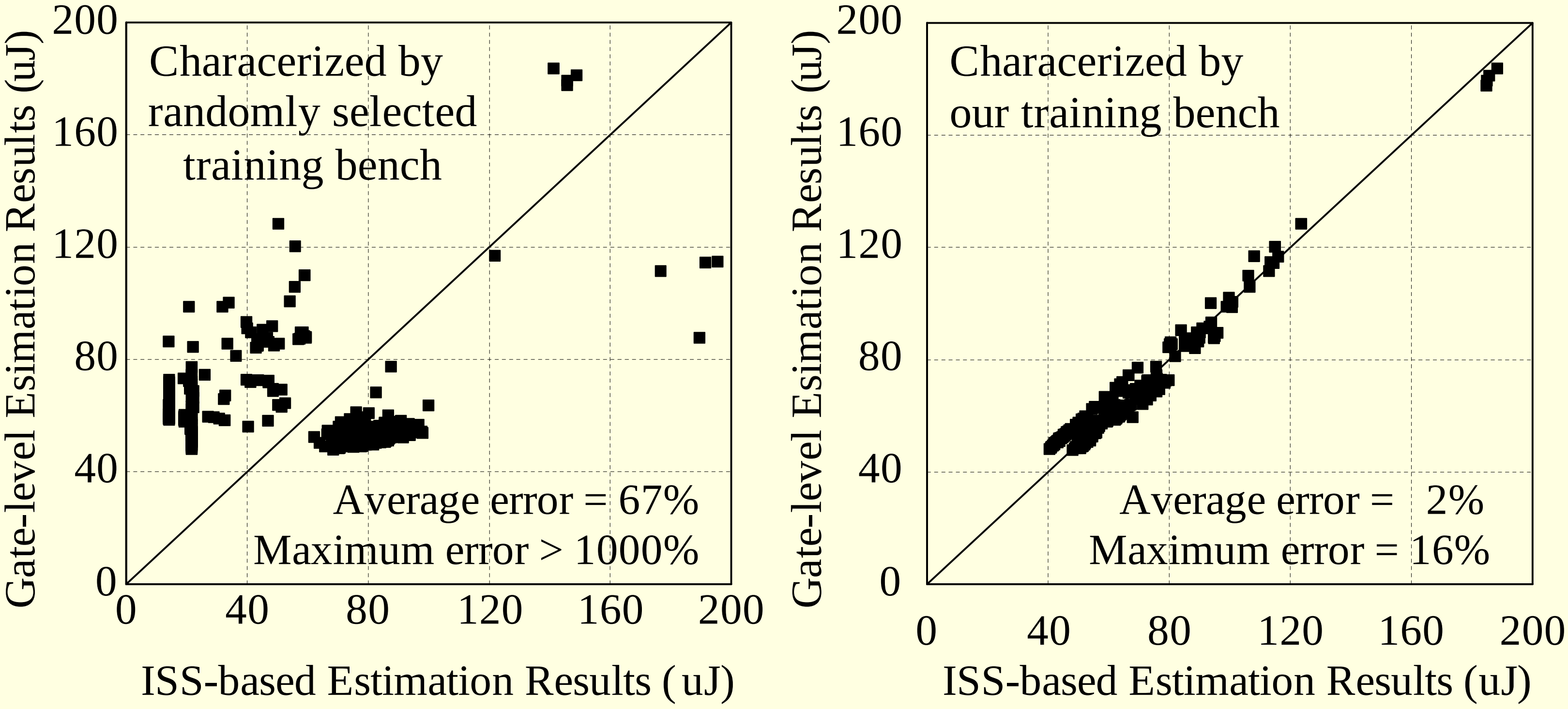
Instruction frame n ⇨ $E_n \longleftarrow E'_n = c_1 P_{n1} + c_2 P_{n2} + c_3 P_{n3} \cdots$

Solve the set of c_i which minimizes $\sum_{i=1}^{n} | E'_i - E_i |$

Error Sources of this characterization

- Parameter set selection
- Non-linear effects
- Training bench
- …

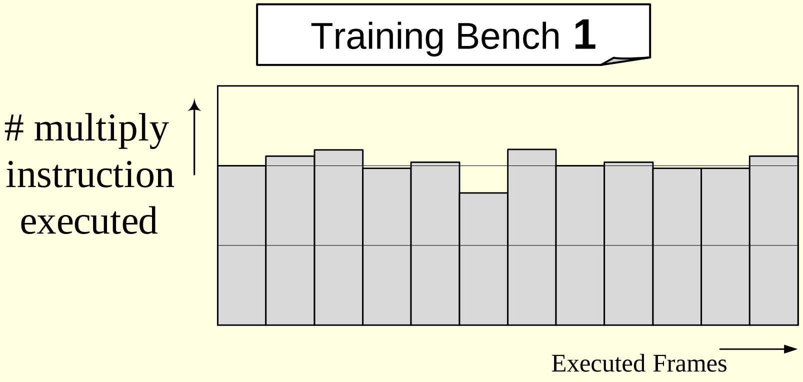
Motivational Example



Training bench dominates model accuracy !!

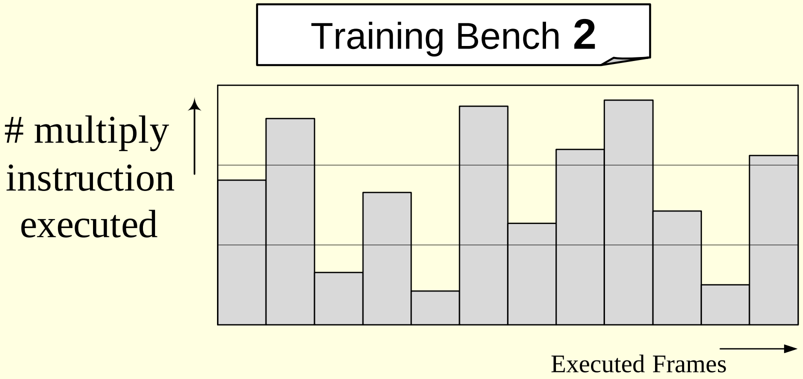
What is expected for a ‘good’ training bench ?

Criteria on Training Bench (1/2)



If parameter value of each frames is constant or mono-tone

Difficult to derive energy consumed by parameter



If parameter value of each frames is randomized

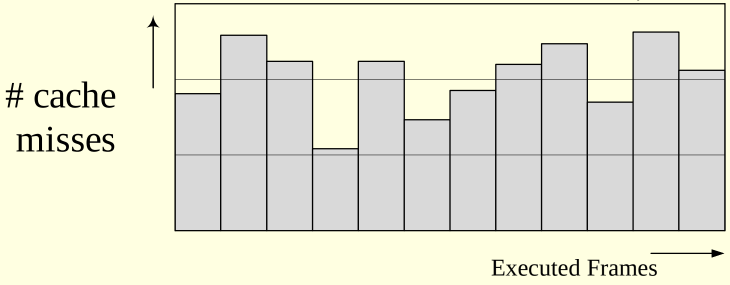
Suitable to derive energy consumed by parameter

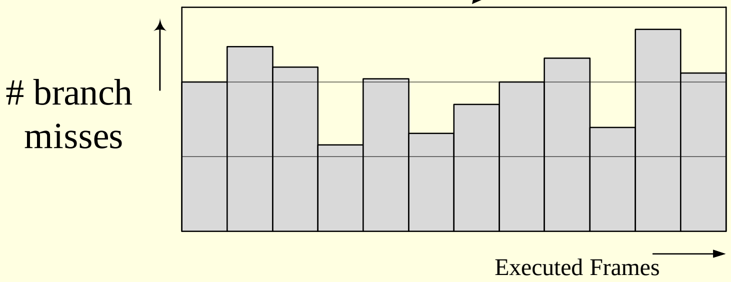
Criterion 1

Standard deviation of parameter value

Criteria on Training Bench (2/2)

$$E = c_1 P_1 + \ldots + c_{cache\_miss} \boxed{P_{cache\_miss}} + c_{branch\_miss} \boxed{P_{branch\_miss}}$$





If correlation is strong between two parameters

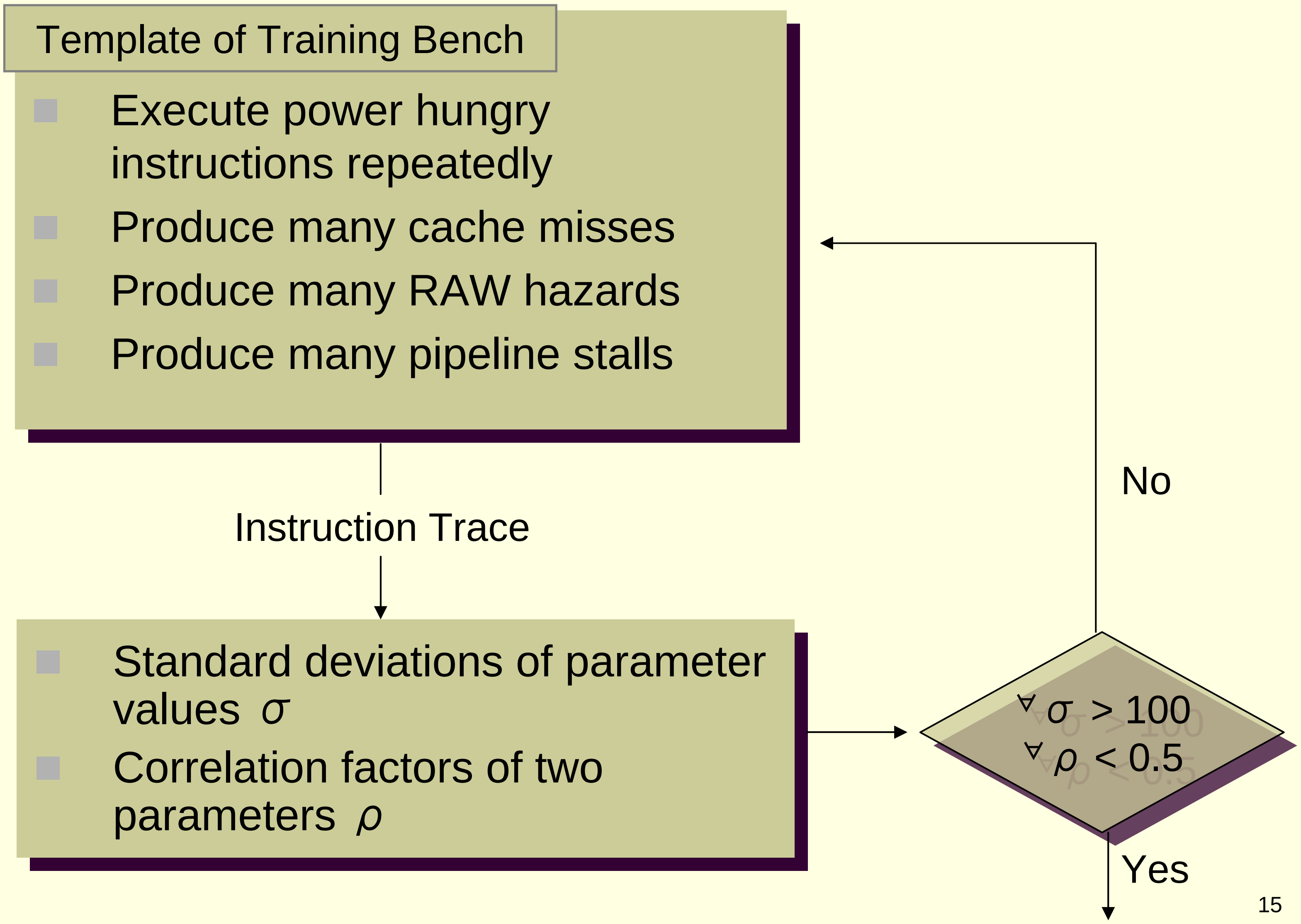
⇓

Difficult to derive energy consumption by each parameters

Criterion 2

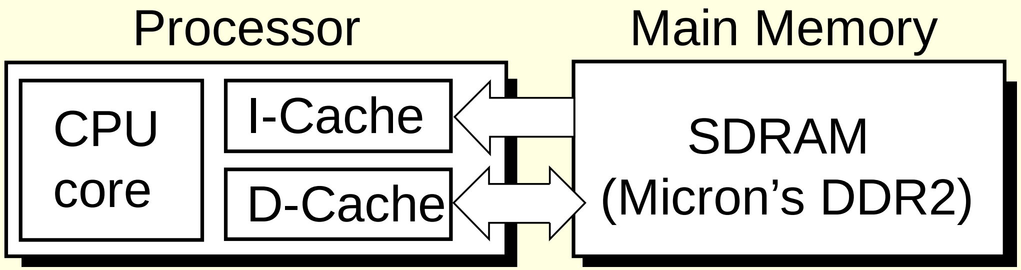
Correlation between parameters

Training Bench Generation



Experiment

- Target system



  - Processors
    - M32R-II, SH3-DSP
  - 0.18 μ m CMOS library

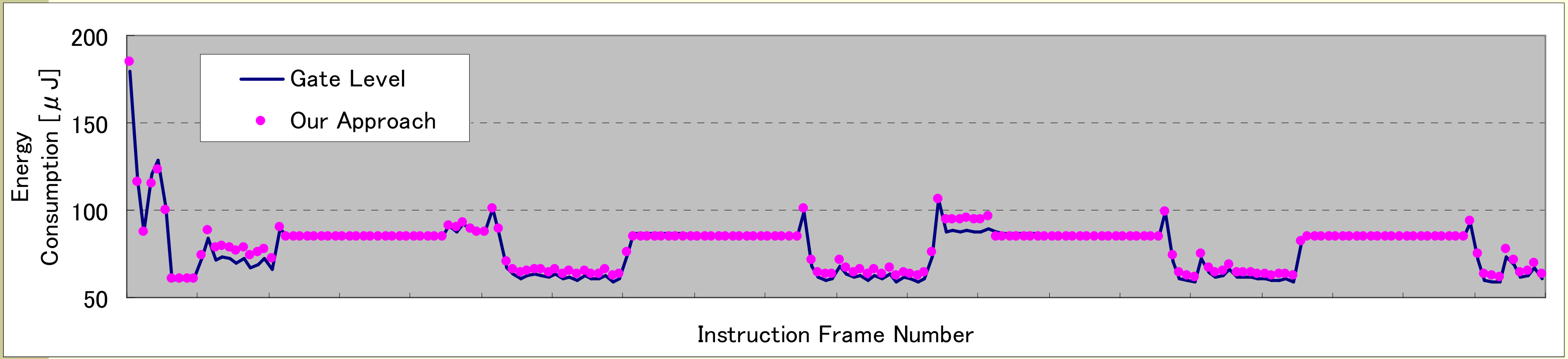
Experimental Result - Energy estimation error

	M32R-II		SH3-DSP	
	Average Error	Maximum Error	Average Error	Maximum Error
JPEG	2.70 %	10.32 %	3.17 %	11.89 %
JPEG_O	6.09 %	16.46 %	6.33 %	10.02 %
MPEG2	1.54 %	3.97 %	1.32 %	3.41 %
MPEG2_O	1.78 %	5.15 %	1.31 %	5.63 %
compress	5.00 %	6.41 %	5.73 %	10.84 %
compress_O	4.35 %	7.18 %	1.73 %	15.15 %
FFT	1.55 %	6.87 %	1.27 %	3.26 %
FFT_O	1.45 %	5.59 %	1.15 %	4.75 %
DCT	1.42 %	8.58 %	1.12 %	2.20 %
DCT_O	1.47 %	8.07 %	1.51 %	3.04 %
Total	**2.74 %**	**16.46 %**	**2.47 %**	**15.15 %**

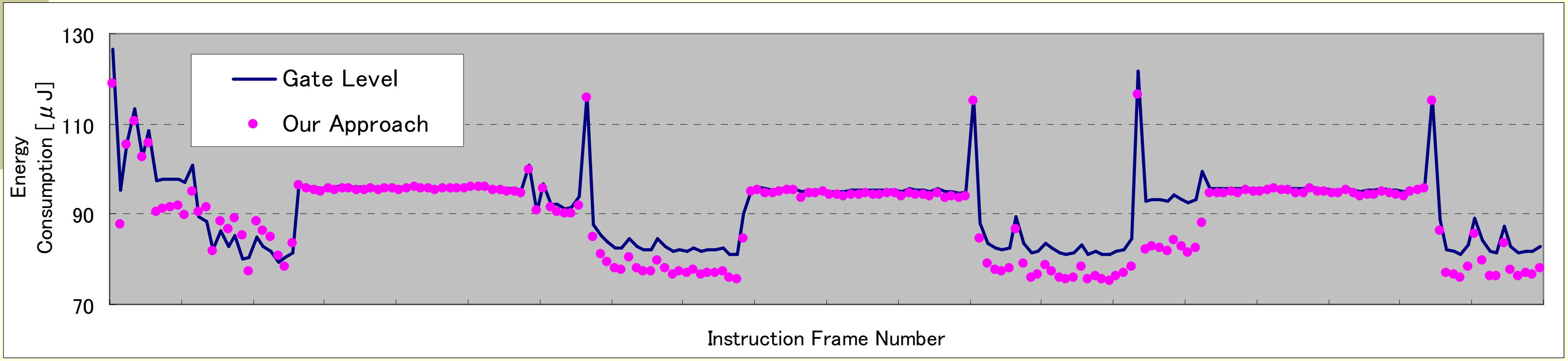
Compared to the gate level estimation

*_O : compiled with a "-O3" optimize option

Experimental Result



Energy estimation for JPEG encoder executed on a M32R-II processor



Energy estimation for JPEG encoder executed on a SH3-DSP processor

Summary

- Proposed energy characterization framework for processor-based embedded system
- Error is on an average 3% and worst case 16%

- Future work
  - Compare result to board level measurement
  - Extend current work to multi-core processor systems
  - Extend to systems running on RTOS

Thank You !