ICT-ENERGY Doctoral Symposium — Bristol, 16 September 2015

ICT-ENERGY LETTERS

Worst case energy modelling

James Pallister, Steve Kerrison, Jeremy Morse and Kerstin Eder
firstname.lastname@bristol.ac.uk
University of Bristol

***Abstract*—The majority of current instruction level energy models for processors do not consider how data values affect the energy consumption of the instruction. This is important when attempting to find the largest possible energy an instruction sequence can consume (worst case energy consumption). This paper outlines the research questions and direction needed to address this problem.**

I. Introduction

Energy consumption of software running on embedded devices has become increasingly important, as a growing number of devices rely on batteries. Of particular interest is bounding this energy consumption — finding an upper bound for the amount of energy an application could consume, and the conditions under which this happens.

Traditional energy models typically assign a single energy value to each instruction. Energy predictions are made by counting the number of times each instruction would be executed in a program and applying these energy values to them [1]. However, this approach is simplistic and potentially inaccurate or unsafe when a worst case energy figure is needed — each instruction takes various parameters and is affected by the state of the computation. The effect of data on the energy consumption of an instruction is particularly important, with a potentially large difference in energy consumption achievable due to different datasets being given to the same application.

Figure 1 shows the effect that data can have on a single instruction. These data were gathered from an XMOS-L1 processor, with the actual energy measured. The graph shows additional power (compared to all zero input) required by different data inputs. It can be seen that power changes by up to 15% depending on the data operands given to the instruction.

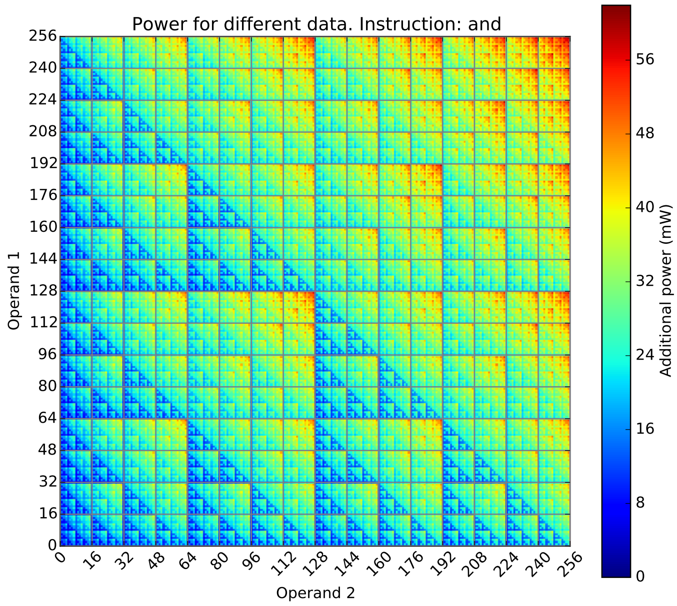

Figure 1: The effect of data values on the `and` instruction, for the XMOS-L1 processor. These values are on top the base power of 350mW.

II. Research Challenges

A worst case energy model (WCEM) should incorporate this data effect to obtain bounds that are both safe and tight, which ensures dependable and meaningful predictions for the dataset an application operates on. Energy is a more challenging metric than time due to its dependence on both time and power. A WCEM may require two components, a time analysis, plus an analysis to determine the worst possible power [2].

Modelling individual instructions. The power values seen in Figure 1 need to be modelled such that the power can be predicted, given operand values.

Instruction sequences. The model needs to be able to compose individual instructions into instruction sequences, so that larger programs can be modelled.

Dependencies between instructions. While an individual instruction could be modelled with a lookup table (making the problem a data gathering exercise), the output from one instruction provides the input to another. This can make it challenging to find the input data that gives the maximum energy for instruction sequences with data dependencies, without bruteforcing every input. The maximum energy for such instruction sequences is potentially less than the sum of individual instruction's maximum energy. How can data dependencies be adequately reflected in an energy model?

Type of bound. Energy is hugely dependent on environmental factors, as well as being challenging to model in fine detail. This means that it is likely infeasible to construct a reasonable hard bound on the maximum energy. Instead, it may be better to use a probabilistic bound, where a tight bound can be given to a high level of confidence [3].

These topics must be addressed to create a satisfactory worst case energy model.

III. Towards a WCEM

In a first step towards a WCEM we have investigated techniques to predict the worst case energy consumption of instruction sequences. Our approach is based on the observation that the transition between instructions is significant and can not be represented using a single energy cost [4]. Instead, we use a distribution dependent on the in-

put and output values of the two consecutive instructions. To validate our findings, the transition distributions for several instructions in the AVR instruction set have been characterised. This gives a tight and safe upper bound on the energy consumption. The distributions can be composed to form a distribution for a larger sequence of instructions. The top tail of the distribution can then be used to compute a maximum energy for a given confidence level.

Our approach still requires any dependencies between instructions to be addressed. This is motivated by experimental results that show that the maximum energy when composing multiple instructions is lower than the prediction when there are many data dependencies between instructions. While the predicted bound without considering dependencies is still valid, it is not as tight as one that may be obtained using a model which does include instruction dependencies.

Overall this approach models the distribution of an entire program's energy consumption, allowing the largest possible consumption of energy to be probabilistically estimated.

REFERENCES

[1] Tiwari et al. "Power analysis of embedded software: a first step towards software power minimization". In: *IEEE Transactions on VLSI Systems* 2.4 (Dec. 1994), pp. 437–445.

[2] Jayaseelan et al. "Estimating the Worst-Case Energy Consumption of Embedded Software". In: *Symp. Real-Time and Embedded Technology and Applications*. IEEE, 2006, pp. 81–90.

[3] M. Pedram. "Maximum power estimation using the limiting distributions of extreme order statistics". In: *Proc. 1998 Design and Automation Conference.* IEEE, 1998, pp. 684–689.

[4] James Pallister et al. "Data dependent energy modelling: A worst case perspective". In: *CoRR* abs/1505.03374 (2015).