POWER ESTIMATION MODEL FOR FALL DETECTION SYSTEM
MÔ HÌNH ƯỚC TÍNH CÔNG SUẤT CHO HỆ THỐNG PHÁT HIỆN TÉ NGÃ

Nguyen Thi Khanh Hong

College of Technology – The University of Danang; ntkhong@dct.udn.vn

Abstract - In today's high-performance system designs, power consideration is becoming increasingly important ,leading to the development of the power consumption in MPSoC at an early step of the design. The aim of our work is to define a methodology to explore power estimation model for Fall Detection System. New power consumption models for processor core are defined according to different configurations of the target architecture (Zynq-7000 AP SoC platform) and the features of the applications like core frequency, number of processor cores, image resolution. Functional Level Power Analysis method is also applied to the extraction of the power models. In our work, we target accuracy based modeling style and analysis of information collected from measurement on real board to obtain sufficiently accurate power estimation for the Fall Detection System on a heterogeneous platform. Besides, to validate the accuracy of the proposed models, we also analyze the average error rate of these models that show that object segmentation (2.2%) , Filter (2.4%) , Feature Extraction (3.5%), and Recognition (2.9%)**.**

Key words - power estimation; fall detection system; FLPA; ARM processors; power model

Tóm tắt - Ngày nay, việc dự đoán công suất tiêu thụ của hệ thống MPSoC đóng vai trò quan trọng trong việc thiết kế những hệ thống hiệu suất cao. Mục đích của chúng tôi là xác định phương pháp để tìm ra mô hình ước tính công suất dành cho Hệ thống phát hiện té ngã. Mô hình công suất được xác định dựa vào quá trình thực nghiệm bằng cách thay đổi các thông số như: tần số của lõi, số lượng lõi xử lý, độ phân giải ảnh trên kit Zynq-7000 AP SoC. Phương pháp phân tích công suất ở mức chức năng (Functional Level Power Analysis) được áp dụng để thực hiện quá trình tìm mô hình này. Bên cạnh đó chúng tôi tiến hành kiểm chứng độ chính xác trung bình của mô hình với các giá trị như sau: khối tách đối tượng đạt 2,2%, Khối Lọc ảnh đạt 2,4%, Khối trích thuộc tính đạt 3,5% và khối nhận dạng là 2,9%**.**

Từ khóa - ước lượng công suất; hệ thống phát hiện té ngã; FLPA; vi xử lý ARM; mô hình công suất.

1. Introduction

Fall is common among elderly people and major obstacles to daily living, especially independent living. It is crucial to develop an automatic Fall Detection System to support the health care system, especially for the elderly people and rehabilitants. Most of the modern video applications use ARM processor, Digital Signal Processors (DSPs) and Field Programmable Gate Arrays (FPGAs) to hand over flexibility, real-time capability, and create custom embedded architectures for different requirements. ARM processors with various and high frequency executions are attractively chosen for real time embedded systems of computer vision applications. Recently, combination of the advantages of ARM processors for writing software and having accelerators in the FPGAs allows getting a higher performance in JPEG decoder for video processing as shown in [1]. Besides, another architecture is DSP/FPGA which was applied for JPEG 2000 and H.264SHD video coding in stimulating its performance such as real-time or near real time by using parallelization and reconfiguration [2]. With the same architecture, the paper [3] presents the fall detection system for elderly people using FPGAs for stereo matching and a DSP for neural network.

In addition, all modern systems from petascale supercomputers to handheld devices must have equilibrium between performance and power consumption. These system often requires access to real time power information. It is useful for the systems to use such information to execute workloads more efficiently.

Various design methods have been proposed to estimate power from low level to high levels that require design information in a language such as Verilog/VHDL/System C/C. However, the more detailed the model is, the slower the simulation and thus the slower the estimation. Therefore, to improve this process, it is desirable to define a power estimation methodology.

The accurate power estimation at a high level takes a significant role in any successful design methodology. Many researchers are interested in extending this area because of increasing the complexity of the MPSoC's architecture. There are some high-level power estimation approaches which consist of spreadsheet, Instruction Level Power Analysis (ILPA) and Functional Level Power Analysis (FLPA). In this paper, the FLPA is used as a methodology of our exploration.

J. Laurent, N. Julien et al., first introduced Functional Level Power Analysis (FLPA) method in [4]. The functional level power modeling approach is applicable to all types of processor architectures. Furthermore, FLPA modeling can be applied to a processor with moderate effort, and no detailed knowledge of the processor's circuitry is needed. This approach is based on a functional analysis of the core of the processor to determine a set of consumption rules. Their functional analysis presents a very efficient and straightforward method for energy optimization. The error rate between estimation and measurement is not higher than 7.4% for their considered application and architecture. The result of this method is applied to a FIR 16 filter on a TMS320C6201 DSP and has extended to other processors.

The recent work of M. E. A. Ibrahim et al. [5] presents a precise high-level power estimation methodology for the software loaded on a VLIW processor based on a FLPA. Their targeted processor is the TMS320C6416T DSP from Texas Instrument.

In the work of S. Rethinagiri et al. [6], they extend the FLPA to create generic power models for the different target processors such as ARM processors (ARM9, ARM Cortex-A8, and ARM Cortex-A9 processors), DSP processor, Heterogeneous multiprocessor (OMAP5912 and OMAP3530) under test. Their estimation of power and energy results provides a maximum error of 5% for mono-processors and 9% for heterogeneous multiprocessor based system when compared with the real board measurements.

Consequently, the target of our work is to define an efficient power model to estimate the power consumption of video applications in the Fall Detection System. In general, the aim of power estimation methodology mentions the speed and accuracy. In this paper, we target accuracy based modeling style and analysis information collected from measurement on real board to obtain sufficiently accurate power estimation for the Fall Detection System on processors based on Functional Level Power Analysis method (FLPA). Therefore, we experiment and verify the model's accuracy on Zynq-7000 AP SoC platform, to show the applicability of our model.

2. Power model for Fall Detection

2.1. Overview of Fall Detection System

Figure 1. *Block diagram of Fall Detection System*

The system in this paper is designed based on the Fall Detection Algorithms [7] which help to obtain signal results showing behaviours of object FALL, or NON FALL, and furthermore not transfer the private image to the other systems.

Object segmentation block is responsible for detecting and distinguishing between moving objects and the rest of the frame which is called background

Filter block improves the quality of image from the object binary image by using modules such as Morphology Mathematic (MM), Edge Detection Filter (Sobel, Canny, Prewit Filter). In this module the object will be smoothed, blobbed and the noise will be removed.

Feature Extraction block extracts five features such as Current angle, Coefficient of motion (Cmotion), Deviation of the angle (Ctheta), Eccentricity, Deviation of the centroid (Ccentroid).

2.2. Methodology of estimation power model for Fall Detection System

In our work, we select the Zynq 7000 AP SoC platform which has both processor cores and FPGA. The aim is to estimate the power consumption in order to evaluate the performance of different implementations.

For processor cores, we first need to realize the power/time characterization of the target. This methodology is based on physical measurements in order to guarantee realistic values with good accuracy. The FLPA methodology, as shown in Figure 2, has four main parts, which are given below:

- Firstly, a primary functional analysis helps the

designer to determine which relevant parameters have an impact on the power consumption. There are two types of parameter: algorithmic parameter values depend on the specificity of the application and architectural parameter values depend on the processor configuration settled by the designer.

- Then, they characterize the power consumption behavior and execution time (obtained by measurements) in varying independent parameters.

- Next, a mathematical model is determined by regression law.

- Finally, the accuracy of the determined model is validated against new measurement set.

In our system, we consider extracting the characteristics and determining the power consumption model for the Object Segmentation, Filter (Mathematical Morphology), Feature Extraction and Recognition tasks. Therefore, the number of experiments for exploring the best architecture of Fall Detection System is reduced. Two types of parameters are considered in this approach:

- Algorithmic parameters depend on the executed algorithm (typically the cache miss rate for the processor cores).

- The component configuration set by the designer (i.e., Clock frequency) is the dependent of architectural parameters.
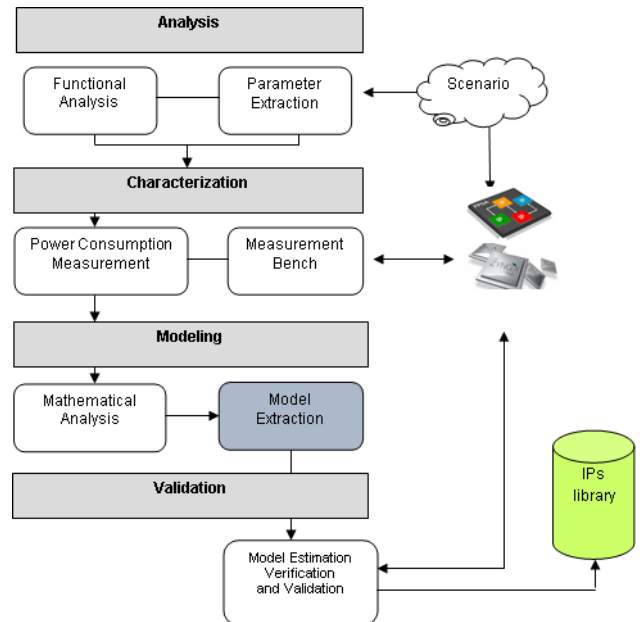

Figure 2. *Functional Level Power Analysis Methodology [8]*

2.3. Scenario Implementation

To extract the performance of processors which include a set of metrics such as Data cache access, data cache refill, total instruction, total cycle and data memory access, etc, we use the optional non-invasive debug component, Performance Monitors Extension. In ARMv7, the Performance Monitors Extension is an optional feature which helps to derive the specification of the earlier ARM implementations[9].

In our work, the four tasks of Object Segmentation,

Filter, Feature Extraction, and Recognition are independently executed, with negligible interference from other tasks. The system is composed of low power processor with N cores, operating at clock frequency F, where F∈{Fmin, Fmax}.
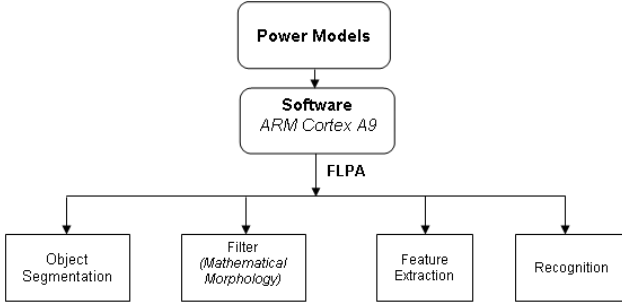

Figure 3. Framework for extracting power models

Firstly, different functional blocks are divided into the memory unit, clock system unit as shown in Figure 3. These parameters are indicated by each functional block of the processor and they are λ_1 and λ_2 respectively for L_1 and L_2 cache miss rates, Instruction per cycle (IPC) for all the activated cores and F for clock unit. The second step is the characterization of the power model by varying the parameters. The scenario of our test is also the two separate modules of Fall Detection System with different resolution images and number of cores. In our work, characterization is accomplished by measurement on Zynq 700 AP SoC platform.

Table 1. Model parameters

Symbol	Power Model Description
λ	Caches miss rate for processor
IPC	Instructions per cycle
s	Resolution Images
F_{core}	Frequency of the core
N	Number of cores

2.4. Proposal Power Estimation Model for Fall Detection System

2.4.1. General Power model

The power consumption models are determined from all experiments by using regression analysis. Regression analysis is a statistical process for estimating the relationships among variables in statistics. It includes many techniques for modeling and analyzing several variables, when the focus is on the relationship between a dependent variable and one or more independent variables. Regression analysis is widely used for prediction and forecasting. In restricted circumstances, regression analysis is used to infer causal relationships between the independent and dependent variables [10].

The models in our work are defined and are related to the indicated parameters such as core frequency, the number of cores, Instruction per Cycle, Cache miss rate, resolution images (as shown in Table 1). Therefore, the power model for the ARM Cortex A9 processor is created by (see more in Equation 1):

$PPS\ (mW) =$

$31.7\ +\ 0.42*F\ +\ 52.9*N\ +\ 7.7*(\lambda 1 + \lambda 2)\ +\ 68.3*IPC$ (1)

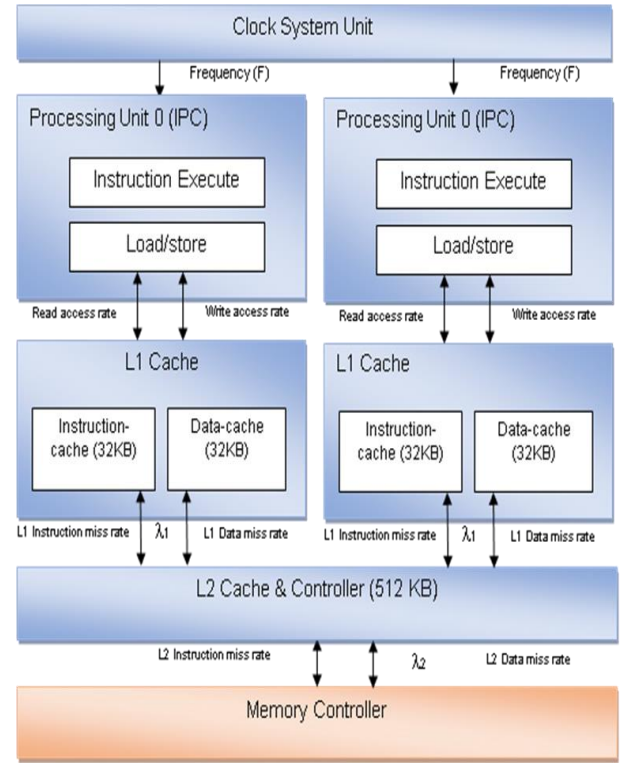Where, PPS: Power consumption on processor cores.

Figure 4. Functional Blocks of Dual Core ARM Cortex A9 processor [11]

Table 2 shows the numbers of experiments measured on ARM Cortex A9. The results obtained for the twelve experiments (for Object Segmentation and Filter tasks) and six experiments (for Feature Extraction and Recognition tasks) validate our approach. Furthermore, it also describes the parameter numbers including image resolution, the number of processor cores and frequency of processor cores for each model.

Table 2. Maximum and average errors for power consumption model on processors

Applications	Processors	Measurement numbers	Parameter numbers
Object Segmentation	ARM Cortex A9	12	4
Filter (Mathematical Morphology)	ARM Cortex A9	12	4
Feature Extraction	ARM Cortex A9	6	4
Recognition	ARM Cortex A9	6	4

2.4.2. Power model for Fall Detection System

We estimate the power consumption of processor cores for Fall Detection System based on the power model as illustrated in Equation 2. The new modeling not only relates frequency of core, a number of cores but also image

resolution parameters are considered in this subsection. The image resolution is one of the factors impacting on the accuracy of our system. Therefore, the power consumption model for each task of Fall Detection System extended from Equation 1 is determined as follows:

$$P(T_i)=P_{core}(N,F_{core},s)=152+0.000416*s+0.39*F_{core}+30.5*N \quad (2)$$

Where, $P(\tau_i)$ is the power consumption on task τ_i. ; τ_i is i^{th} of task and i= (1:4); N is the number of processor cores; F_{cores} is the frequency of processor cores; the image size or the image resolution is assigned by s.

The evaluation of the general model is analysed by the real measurement on processor cores with the maximum error at 3.5%. This error rate is not too high, therefore it is a good adequacy in extending power consumption models for the Fall Detection System.

3. Result and Discussion

The different power models are validated by the real board measurement in order to find the efficiency of FLPA modeling for processor cores applied in this paper. The video applications are compiled for processor cores on Zynq 7000 AP SoC platform. While these applications are running, the power consumption is measured online. Finally, the measurement of the experiment from the platform is compared with the estimation from the power consumption model which extracts the useful activities of the power model.

Table 3 shows the maximum and average errors obtained with our approach modeling against measurements on ARM Cortex A9. The results obtained for the experiments (as described in Table 2) validate our approach. With each model the power estimation is obtained. Our power modeling approach has a negligible maximum error equal to 3.5 %.

Table 3. Maximum and average errors for power consumption model on processors

Applications	Processors	Maximum error (%)	Average error (%)
Object Segmentation	ARM Cortex A9	5.5	2.2
Filter (Mathematical Morphology)	ARM Cortex A9	5.3	2.4
Feature Extraction	ARM Cortex A9	6.2	3.5
Recognition	ARM Cortex A9	5.8	2.9

4. Conclusion

In this paper, we specific the separate video tasks which are used in the Fall Detection System to extract the general power consumption models. The modeling methodology is defined by analysing processor cores (based on FLPA)

with the aim to combine them with heterogeneous architecture. To define power consumption models for processor cores, the different scenery of experiments are implemented according to the different configurations offered by the ARM Cortex A9 processor of Zynq platform. On the basis of the FPLA techniques, power consumption models have been extracted for the different tasks of the Fall Detection System.

Moreover, these models are extended for the Fall Detection System regarding the features of the target architectures and the considered applications such as image resolutions, core frequency and a number of activated cores. The analysis of the error rate shows a maximum of 3.5% for the power consumption. The error rates offer a good quality models on processor cores.

The future work will be a new exploration methodology for low-cost architectures of the Fall Detection System with both execution time and power consumption models. An accurate model for this system is also proposed by applying Design Space Exploration (DSE) Methodology for the Fall Detection System.

REFERENCES

[1] F. Eberli, "Next Generation FPGAs and SOCs - How Embedded Systems Can Profit," 2013 IEEE Conf. Computer Vision Pattern Recognition Work, pp. 610–613, June. 2013.

[2] G. Baruffa, F. Fiorucci, F. Frescura, P. Micanti, L. Verducci, and B. Villarini, "A reprogrammable computing platform for JPEG 2000 and H.264 SHD video coding," 2010 8th IEEE Work. Embed. Syst. Real-Time Multimed., pp. 107–113, Oct. 2010.

[3] F. Humenberger M., Schraml, S., Sulzbachner, C., Belbachir, A.N., Srp A., and Vajda, "Embedded fall detection with a neural network and bio-inspired stereo vision," 2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, p. 60–67, June 2012.

[4] J. Laurent, E. Senn, N. Julien, and E. Martin, "High-Level Energy Estimation for DSP Systems," in PATMOS. IEEE, 2001.

[5] M. E. a. Ibrahim, M. Rupp, and H. a. H. Fahmy, "A Precise High-Level Power Consumption Model for Embedded Systems Software," EURASIP J. Embed. Syst., vol. 2011, pp. 1–14, 2011.

[6] S. K. Rethinagiri, O. Palomar, J. A. Moreno, O. Unsal, and A. Cristal, "System-Level Power and Energy Estimation Methodology for Open Multimedia Applications Platforms," in IEEE Computer Society Annual Symposium on VLSI (ISVLSI), 2014, pp. 442–449.

[7] Y. T. Ngo, H. V. Nguyen, and T. V. Pham, "Study on fall detection based on intelligent video analysis," 2012 Int. Conf. Advance Technology Communication, pp. 114–117, Oct. 2012.

[8] N. Julien, J. Laurent, E. Senn, and E. Martin, "Power Consumption modeling and Characterization of the TI C6201," IEEE Micro, no. 5, pp. 40–49, 2003.

[9] "ARM Architecture Reference Manual," 2014. [Online]. Available: http://infocenter.arm.com/help/index.jsp?topic=/com.arm.doc.ddi0406c/index.html.

[10] J. Aldrich, "Fisher and Regression," Stat. Sci., vol. 20, no. 4, pp. 401–417, Nov. 2005.

[11] http://www.design-reuse.com/articles/16875/the-arm-cortex-a9-processors.html