

System Level Power and Energy Modeling for Signal Processing Applications

Jalel KTARI¹, Mohamed ABID¹
¹CES-National Engineers School of Sfax
Sfax, Tunisia

ktari@iuplo.univ-ubs.fr, mohamed.abid@enis.rmu.tn

Abstract—This article presents a new methodology of consumption and performance characterization of software's intellectual property (IPs) computing on DSPs. These IPs are generally submitted to various constraints especially the real time. The proposed approach exploits parametric models representing the consumption's behavior of both DSP's architecture and algorithm. This consists in releasing the laws of consumption on a high level. This approach makes it possible to deduce the power and the energy consumption of a code in an advanced language (Strict ANSI-C) for a given target. Feasibility and the interest of the approach are proved using a signal processing applications.

Keywords: Low power, DSP, MPEG2, Modeling, methodology.

I. INTRODUCTION

New applications such as laptop, wireless telecommunications (radiomobile, GSM, PDA, etc) are more and more increasing in the electronic domains. This type of applications integrates complex functionalities (coding speech or multimedia video, treatments, etc), which requires powerful computations, while adding strong constraints on the system consumption [1, 2].

The availability of high performance IP cores for System on Chip (SoC) devices is an important factor of the electronic market and has attracted significant research interest. Moreover, in order to maximise the operating time provided by the battery and to satisfy the real time constraint, we need high level IP modeling. So we can maintain performance, low power constraints as well as that of the battery and real time.

The objective of this work is to define an approach of IP consumption's estimation and modeling in the System level using FLPA. Initially, we point out the related works then the methodology "FLPA" (Functional Level Power Analysis). Then, we will introduce its transposition for Software IP in order to consider the IP's algorithmic specifications. Then, we will appraise the models suggested on signal processing applications. Finally, we will discuss the results obtained compared to the physical measurements carried out on the development board and their benefits in the design space exploration.

II. RELATED WORK

With the physics of the semiconductor working against the designer at nanometer process technologies, the designer must turn to the voltage and the clock frequency of the system-on-chip (SoC) to find power savings. In the simplest sense, designers want to be able to treat individual blocks of a SoC in much the same way they did when those parts were separate chips on a printed-circuit board. These chips could run at different clock rates and different voltages and could be turned off when not in use in order to save power. The majority of IP power management focus on dynamic voltage, frequency scaling and current consumption in standby mode. Currently, few work exist on the consumption modeling of software IP according to the parameters of the IP itself.

Existing energy estimation and monitoring techniques can be divided into two categories: simulation-based or measurement-based. Energy simulators such as Wattch [3] and SimplePower [4] estimate the energy consumption in reasonable time[5].

On RTL level, we can mention the DSP-PP [6], a tool for simulation allowing the estimate of the power dissipated by DSPs. It is composed of two components: the simulator of performance on the cycle level (CPS) and the estimator of the dissipation of power (PDE). It is written in C++ making it possible to consider abstract models. The components of the DSP are modeled like objects integrating the model of consumption. DSP-PP considers the simulation on the cycles level of all the DSP's components: the ways of data and the interconnection and estimate the value of dynamic power, short-circuit of each component of the DSP.

Representative researches in measure-based estimation techniques are SES [7] and PowerScope [8]. SES is an energy-monitoring tool, which collects energy consumption data in a cycle-by-cycle resolution and maps the collected energy consumption data to program structure. SES has a main advantage that the accuracy of analysis results is very high because profiling is performed in a cycle-by-cycle resolution. However, SES needs an extra profile acquisition module which consists

of measurement circuit, profile controller and acquisition memory.

PowerScope [8] is based on hardware instrumentation by using a digital multimeter with support of embedded operating system. Profiled energy costs are mapped in high-level language. In PowerScope, any extra hardware logic is not needed in embedded systems. Therefore, PowerScope is applicable to ordinary embedded systems. EPRO [5] employs measure-based estimation techniques used in SES and PowerScope. However, ePRO is distinct from SES because ePRO does not need any extra hardware module such as profile acquisition module in SES.

In addition, the increasing importance of the software part in these embedded systems requires the analysis of consumption on advanced level of the design. Moreover, optimizations at the architectural and algorithmic levels bring more important profits than those carried out at the technological level.

For example, JouleTrack [9] does not model the program; the model of processor is simply made up of the frequency and the voltage supply.

On algorithmic level, we can mention SoftExplorer[10], a tool that can estimate the power and energy consumption of an algorithm directly from the C program, or from the assembly code. Estimation is based on a power model of the targeted processor, obtained through an FLPA methodology for some Texas Instrument's DSP. This methodology FLPA allows to establish a model of consumption at high level of a given processor. In this approach, the architecture of the processor is decomposed into various independent functional blocks. Each of these blocks is stimulated separately thanks to a scenario of instructions in assembler code so as to obtain the model of consumption.

The objective of this work is to define an approach of consumption's estimation in the system level. For this reason, we do not propose models of consumption of the target architecture but a model of the algorithms themselves. Parametric models binding consumption (power and energy) with the architectural aspects of the target (DSPs) have been used [10]. Here, we do not apply this method to architecture but directly to the algorithm in order to characterize its consumption.

III. METHODOLOGY

Starting from the functional analysis, the "FLPA" [10] methodology allows us to develop a parametric model, which represents the consumption behavior of a target. In fact, this methodology is composed of three steps.

- Functional analysis determines the effective parameters to take into account in the power model.
- Characterization of each parameter is tuned to qualify the output variations either by measurements on board or by low-level simulation. Each parameter that does not impact significantly the characteristics is then discarded.

- The general model is established according to the available parameters.

The transposed methodology for IP is composed also of the same three steps. (Fig. 1)

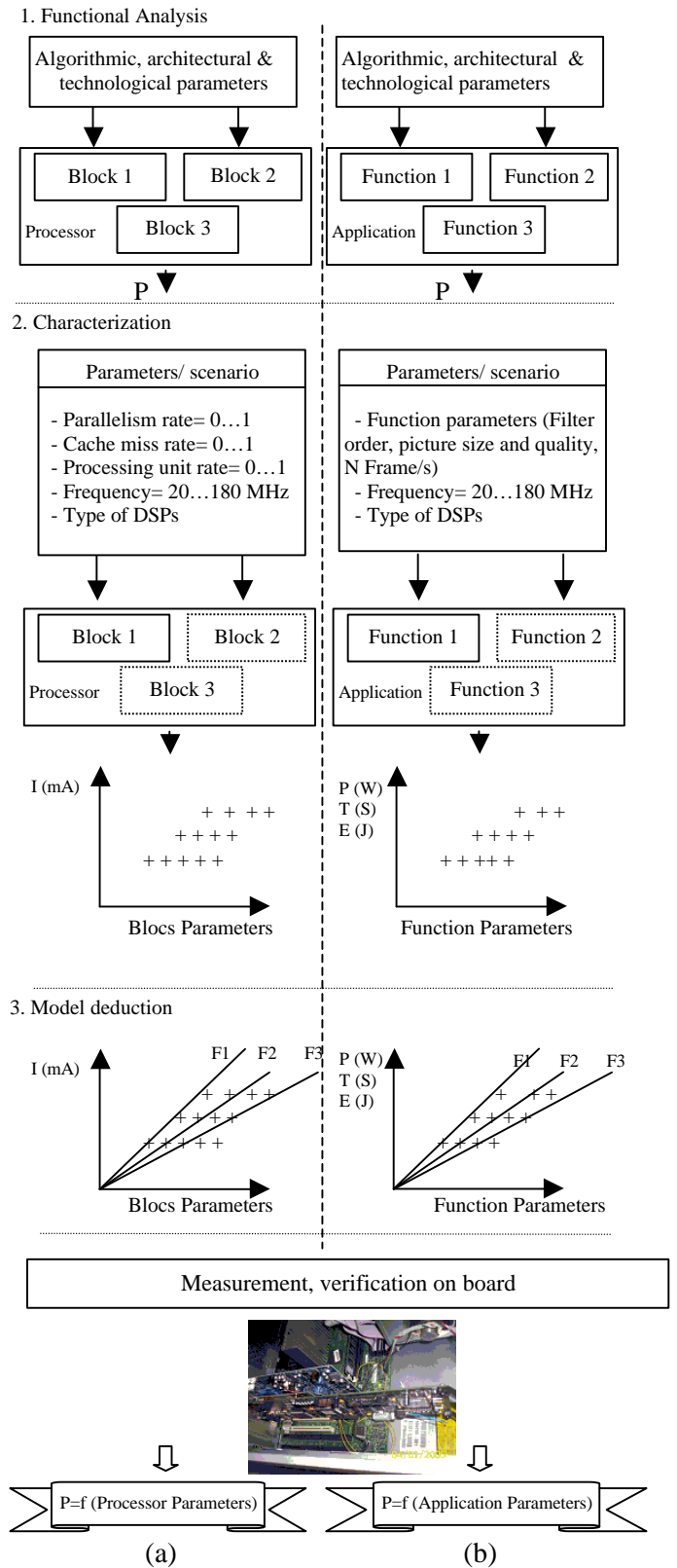


Figure 1. (a) FLPA Methodology for processor (b) Transposed FLPA for IP(SW)

Thus, we can take account of the algorithmic specifications, in order to appraise the consumption at the algorithmic level according to the IP's parameter variations.

This work is based on using 2 models of DSPs: Texas Instruments (C5510 & C6701) integrated in the tool SoftExplorer [10]. They exploit this methodology of functional analysis "FLPA".

We appraise the models suggested on 2 signal-processing applications: a filter FIR and a FFT. The modeling of applications written in C relates to the research of the parameters influencing consumption. Through this study and thanks to Soft Explorer, a model of consumption is established according to these parameters.

The application parameters considered are:

- Algorithmic: the filter order N (from 8 up to 256), and (from 32 up to 2048 points (Np)) for the FFT.
- Architectural: the clock frequency (30 MHz up to Fmax of the DSP) and
- Technological: type of the DSP used (C6701 & C5510). Indeed, this model is deduced by varying these parameters and by exploiting the estimation given by SoftExplorer for various frequency and targets. In fact, on the basis of measurements on boards (C5510 and C6701) and of these estimates, the variations of consumption and execution time are deduced for many values of frequency.

IV. MODELS

The consumption models of FIR and FFT (time, power and energy) for the two processors (C6701 & C5510) are given in table I.

Fig. 2 illustrates the consumption's variation (energy) of the FFT on the DSP C6701 according to the frequency and the number of points from 32 to 2048. In order to reduce the modeling error of FFT consumption laws on the DSPs, a multi-linear model is adopted to time and energy. This energy model is quasi-invariant in segments even if the frequency changes.

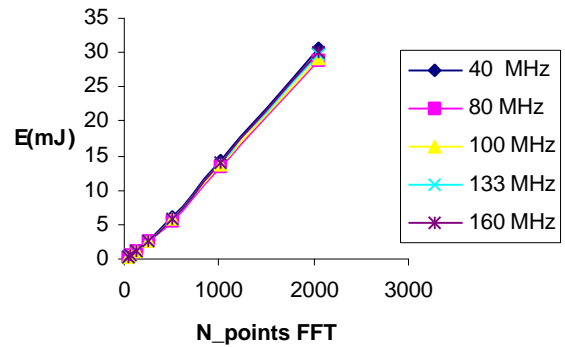
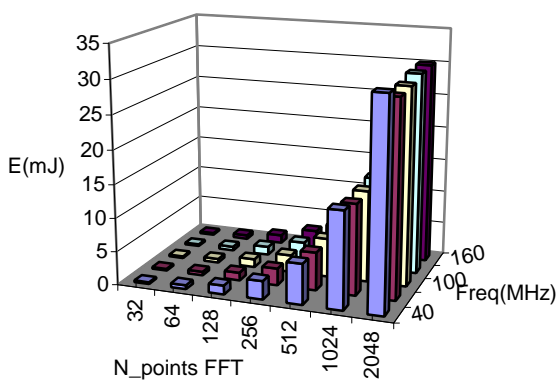


Figure 2. The consumption's variation of the FFT on the DSP C6701

The consumption models (time, power and energy) for the two processors (C6701 & C5510) are given in table I.

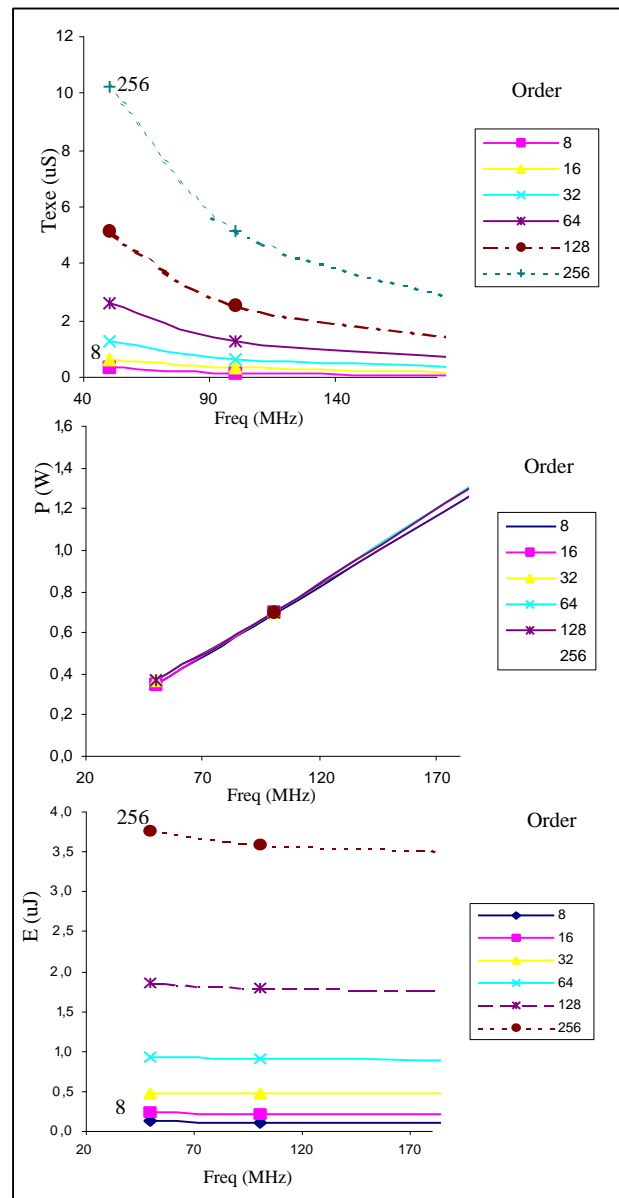


Figure 3. The consumption's variation of the FIR on the DSP C6701

Fig. 3 illustrates the consumption's variation (time, power and energy) of the FIR on the DSP C6701 according to the frequency and the order of the filter from 8 to 256. For the FIR, it is well noticed that the execution time model varies linearly with the order. Energy is quasi invariant even if the frequency changes.

Concerning the order 8, the maximum error in theoretical modeling of the execution time compared to SoftExplorer is 9.5% whereas starting from order 16, the maximum error is only 3.6% for both DSPs.

Table I: Models of consumption of the FIR and FFT

F(MHz)	T (uS)	P (mW)	E (nJ)	
FIR / C6701				
Model	2.006 * order/F	7.20 * F	14.4 * order	
Error max/measurement	9.5%	6.5%	10%	
Average error	5.2%	2.9%	7.1%	
FIR / C5510				
Model	3.013 * order/F	2.758 * F	8.312 * order	
Error max/measurement	9%	6.6%	13.1%	
Average error	5%	3%	7.6%	
FFT / C5510				
Model	Np < 64	2469 * Np/F	2.53 * F	6246.5 * Np
	64 < Np < 512	3180 * Np/F		8045 * Np
	512 < Np < 2048	4320 * Np/F		10929.6 * Np
Error max / measurement	Np < 64	7%	5%	10.6%
	64 < Np < 512	8.6%		13.4%
	512 < Np < 2048	9%		13%
Average error	5.6%	3.56%	6.6%	
FFT / C6701				
Model	Np < 64	1668 * Np/F	*F	8181.5 * Np
	64 < Np < 512	2120 * Np/F		10398.6 * Np
	512 < Np < 2048	2880 * Np/F		14126.4 * Np
Error max / measurement	Np < 64	5%	6.3%	9%
	64 < Np < 512	7%		10.6%
	512 < Np < 2048	3.5%		8%
Average error	3 %	2.91%	4%	

This can be explained by the fact that for the little orders, the compiler is not able to compile the code in order to use maximum parallelism whereas SoftExplorer considers that the architecture is used as well as possible. The FIR power model is only a function of the frequency and its average error is 7% compared with the estimates provided by SoftExplorer.

As for the FFT application, the time model follows a multi-linear law. However, the power model follows a linear law. The maximum error of energy model is 10.6 % in the C67 and 13% in the C55 against measurements. That of power doesn't exceed 6.3%.

Physical measurements on boards of development (containing C6701 & C5510) are made in order to check the validity of the model established with SoftExplorer according to the parameters of the FIR application. These measures are established thanks to:

- The evaluation platform « Code Composer » of TI, which supplies the cycles number necessary for the execution of the application on the DSP boards.
- The logic analyzer, which provides the current level on the DSP core.

The consumption models, having been developed by using the tool, can be refined by using physical measurements of consumption carried out on the board of development. The same improvements are made for the model of the application realized on C55. Whatever the model is, the max error either in the execution time or in the power and energy remains lower than 13% for the FIR and FFT.

It should be noted that this approach could reduce the design time through modeling in the system level and the SoftExplorer tool. Indeed, by just annotating the dynamic parts of the C code [11], we will obtain the estimates (time of estimate lower than the second); whereas by using physical measurement, the modeling time is more important and requires platform for measurements.

V. SPACE EXPLORATION

The design space exploration consists in analyzing the possible solutions to deduct the optimal solution according to a function of cost: performance, surface and power. So, the two main parameters of the conception to be respected are the application specification and its constraints. (fig.4)

Moreover, to limit the space of solutions and to be able to choose effective and realistic solutions, target architecture are considered.

In order to develop this methodology through an example, let us take the case of the FIR running on the DSP C6701. Let us admit that the application is subjected to two constraints:

- Average power should not exceed 1 W,
- Real time processing should not exceed 2 uS.

VI. CONCLUSION

As computation and communication have been steadily moving toward mobile and embedded platforms, realizing low power consumption has become a critical concern in designing modern embedded systems, which is added to surface and performance.

Through this work, we have shown the interest and the feasibility of IP consumption's modeling at a high level. The models and the environment suggested make it possible to estimate the consumption of the application at the system level as a function of the frequency and the order.

It is necessary to have a reliable high-level estimate, which allows the designer not only to choose the most adapted processor to its application but also to regulate its parameters according to the constraints.

Such models can be used, for example, by an operating system, which could choose the parameters of the algorithm to respect the constraints of consumption according to the context. We would have thus an approach of power and energy management at the algorithmic level in order to carry out an adaptive control. In fact, by taking this FLPA tool (SoftExplorer), and by running it on a filter and FFT for different filter/FFT parameters, we can produce estimations for the performance in relation to these algorithmic parameters. This estimation was fit to curves with higher-level parameters. Using the curves, we can then have an estimation of power vs. number of filter taps, for example. This higher-level modeling could be used in design exploration. A vendor of IP could provide these higher-level models to the customer.

This work opens new possibilities in taking consumption into consideration in the applications' design flow. So we can add new dimensions to solution selection, namely the guarantee of QoS (Quality of Service) from both application quality and real time issue points of view.

REFERENCES

- [1] J.M. Rabaey, M. Pedram, *Low Power Design Methodologies*, Kluwer Academic Publisher, 1996.
- [2] N. Julien, J. Laurent, E. Senn, D. Elleouet, Y. Savary, N. Abdelli, J. Ktari "Power/Energy Estimation in SoCs by Multi-Level Parametric Modeling", ReCoSoC'05, Juin2005, France
- [3] D. Brooks, V. Tiwari, M. Martonosi, "Wattch: A Framework for Architectural-Level Power Analysis and Optimizations", In Proceedings of International Symposium on Computer Architecture (ISCA), 2000.
- [4] W. Ye, N. Vijaykrishnan, M. Kandemir, M.J. Irwin, "The Design and Use of SimplePower: A Cycle Accurate Energy Estimation Tool", In Proceedings of 37th Design Automation Conference (DAC), 2000.
- [5] W. Baek, Y. Kim, J. Kim, "ePRO: A Tool for Energy and Performance Profiling for Embedded Applications", In Proc.

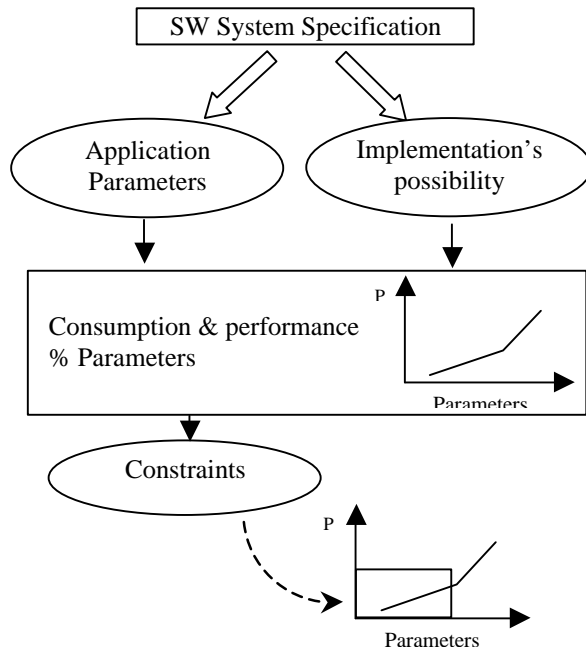


Figure 4. Software design space exploration

Considering the constraint of power, the maximum authorized frequency must be lower than 137 MHz in order not to exceed 1 W. Moreover, by studying the FIR's time model according to the frequency and N , and in order to respect the time constraint (2 μ S), a new law is established $N_{max} = f$ (Frequency). (Fig. 3)

In our case, N_{max} can in no situation exceed 136 points, indeed for $F=137$ MHz and $N=136$, running time is $1,99 < 2\mu$ S.

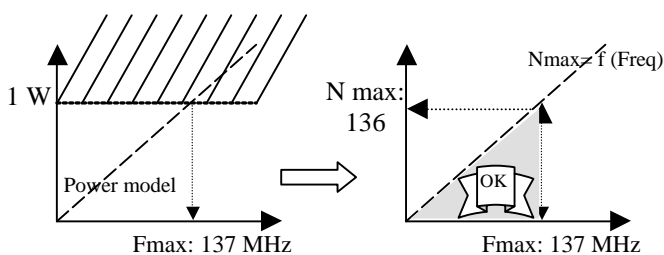


Figure 5. Design exploration applied on the FIR

Thus, with this methodology, the designer could know the filter's maximum order and also the frequency field allowing to respect the constraints of the application at a high level and in a short time. So, the designer can build a trade-off between the frequency and the order. With this method, the feasibility of the design's space exploration on simple applications was demonstrated. This method will be tested for more complex application in the future.

of International SoC Design Conference (ISOCC'04), Seoul, Korea, October 2004, pp. 372-375

[6] D Q. Minh, L. Bengtsson, P. Edefors, "DSP-PP: A simulator/estimator of power consumption and performance for parallel DSP architectures", Proc. 21st IASTED international Conference Applied Informatics, Austria, February 2003.

[7] D. Shin, H. Shim, Y. Joo, "Energy-Monitoring Tool for Low-Power Embedded Programs", IEEE Design and Test of Computers, July 2002, pp. 7 – 17.

[8] J. Flinn, "PowerScope: A Tool for Profiling the Energy Usage of Mobile Applications", In Proceedings of the Second IEEE Workshop on Mobile Computer Systems and Applications, 1999.

[9] A. Sinha, A. P. Chandrakasan, "JouleTrack - A Web Based Tool for Software Energy Profiling", in Proc. DAC, June 2001, p220.

[10] J. Laurent, N. Julien, E. Senn, E. Martin, "Functional Level Power Analysis: Efficient year Approach for Modeling the Power Consumption of Complex Processors", IEEE DATE 2004, Paris, February 2004.

[11] <https://www.softexplorer.fr>