Oct 28th, 9:00 AM - Oct 30th, 5:00 PM

Models for reducing power consumption in CPLD and FPGA devices

Marsida Ibro
Aleksander Moisiu University, marsida.ibro@gmail.com

Luan Karçanaj
Professional College, Tirana, luankarcanaj@gmail.com

Models for reducing power consumption in CPLD and FPGA devices

Marsida Ibro[1], Luan Karçanaj[2]

[1]University "Aleksander Moisiu", Durrës, Department of Information Technology
[2]Professional College, Tirana
{marsida.ibro[1], luankarcanaj[2]}@gmail.com

Abstract. Usage of programmable logic devices PLD has increased in the latest years because of the ability to quickly implement complex types of electronic systems while reducing cost and time of synthesis. This technology enables dynamic reconfiguration of different applications according to specific requirements. Also, power consumption and its loss is becoming an increasingly important requirement in the design of systems for portable applications fed by batteries.
Other factors to be taken into account in the consumption of power are elements that are used for manufacturing, packaging, and cooling systems. Power consumption must be taken into consideration especially for wireless applications where battery technologies provide power 20 W/h and voltage 1.2 volts. Despite improvements in battery technology, the development of methods for reducing power consumption plays a decisive role in portable applications.
Therefore, modeling of power consumption has become a requirement with the highest impact in the performance of FPGA elements. Despite generated models of the different manufacturers of these elements, this article will appear comparisons of models based on experimental measurements performed on both CPLD and FPGA elements. Based on these models is selected to simulate a system that will be implemented in two elements and see how reduced power consumption, without affecting system performance. Experimental results show that FPGA elements have better performance and significantly reduce the power consumption.

Keywords: models, power consumption, CPLD/FPGA

1. Introduction

Recent years have seen an increase in the use of embedded systems, specifically in the field of electronics. This growing trend on the use of embedded systems is a consequence of very good performance and low power consumption are thus motivated to further research on advanced techniques for these systems. Many of these systems implement digital signal processing and require implementation of many mathematical calculations. Since digital signal processing (DSP) is already integrated into many devices it is necessary to achieve optimal designs to meet market demand. Software can enable flexibility in design, allowing continuous changes after design is complete. Software sequentially executed in hardware and allows sequential execution but in parallel. Also, the creation of integrated circuits for specific applications (ASIC) requires more time and after finishing it

is impossible to change the design. In this case they come to the aid of programmable logic elements, which provide a good solution combining hardware and software.
Digital signal processors have found applications in many areas of technology because of the short development time, low power consumption and low cost. Due to the design requirements of DSP systems programmable logic elements have become very necessary. Due to the development of fabrication technologies, FPGA contain many programmable logic blocks (CLB) and are becoming platforms suitable for a wide range of applications. Processors typically perform arithmetic operations through software, and the idea to realize these actions by hardware require a very long time to be designed. FPGA development platforms enable the best combination possible of both cases. Configurable hardware, such as FPGA, provides very high performance and therefore is faster than traditional microprocessors.
Finally, multiprocessor software technology use resources available in programmable devices. Based on their suitability and ability to support parallel, they serve as best platforms for fast prototype development and provide sufficient space for the design of complex systems.
Often these microprocessors can be implemented using FPGA as to enable reconfiguration when new functionality is needed. Digital signal processors performed by software should be simple architecture which provides good performance mainly to non critical calculations. In general, the FPGA can implement complex applications with high throughput, but the best performance requires high power consumption and this is the main challenge of today's electronic systems, especially in mobile devices [1-3].

2. Power consumption calculation

In this paper we focused on the aspect of physical-synthesis implementation of changes in logic circuits. The aim of these changes is to improve former county synthesis in order to significantly reduce dynamic power of it dissipates. In the following are some of the methods used for reducing dynamic power dissipated in circles Field-Programmable Gate Array. These methods use the theory to detect glitch probability and then apply techniques to reduce their use D-FF with negative triggered exit. These methods reduce number of logical transitions that occur, thus reducing the dynamic power dissipated by a logic circuit.
Most of the proposed techniques for reducing dynamic power at the regional level, including the establishment of a control logic that synchronizes entries logical blocks and reduce tension food sources [4]. At the logical level, the dynamic power dissipation achieved during synthesis and mapping technology [5, 6]. Rewiring up during synthesis reduces toggle rate of internal signals district, reducing the number of logic gates, the output of which toggles often and redundant gates are just added to reduce toggle.
Power dissipation consists of three basic components: static, short-circuit and dynamic. Static power dissipates when transistor current is flowing in even when it is closed. Static Power Static power is the power consumed by the FPGA when no signals are toggling. Both digital and analog logic consume static power. Short-circuit power is dissipated by a CMOS gate during a short period of time when pull-up and pull-down pass current. Dynamic power is dissipated whenever capacitor is charged and discharged in the circuit [7]. By experiments and surveys a FPGA with 4-LUT has power dissipation is 40% due to static power and 60% due to dynamic power [8] while the short circuit is negligible compared with the first two [9]. Dynamic power is the additional power consumed through the

operation of the device caused by signals toggling and capacitive loads charging and discharging. Dynamic power decreases with Moore's law by taking advantage of process shrinks to reduce capacitance and voltage. The challenge is that as geometries shrink with each process shrink, the maximum clock frequency increases [10].
Dynamic and I/O power dominate the FPGA's total power consumption. In Figure 1 shows the contribution of each component in FLEX10K and Spartan 6 elements. Power consumption distribution of the five elements forming a FPGA (Logic Cells, interconnect resources, clock tree, I/O cells and Memory cells).

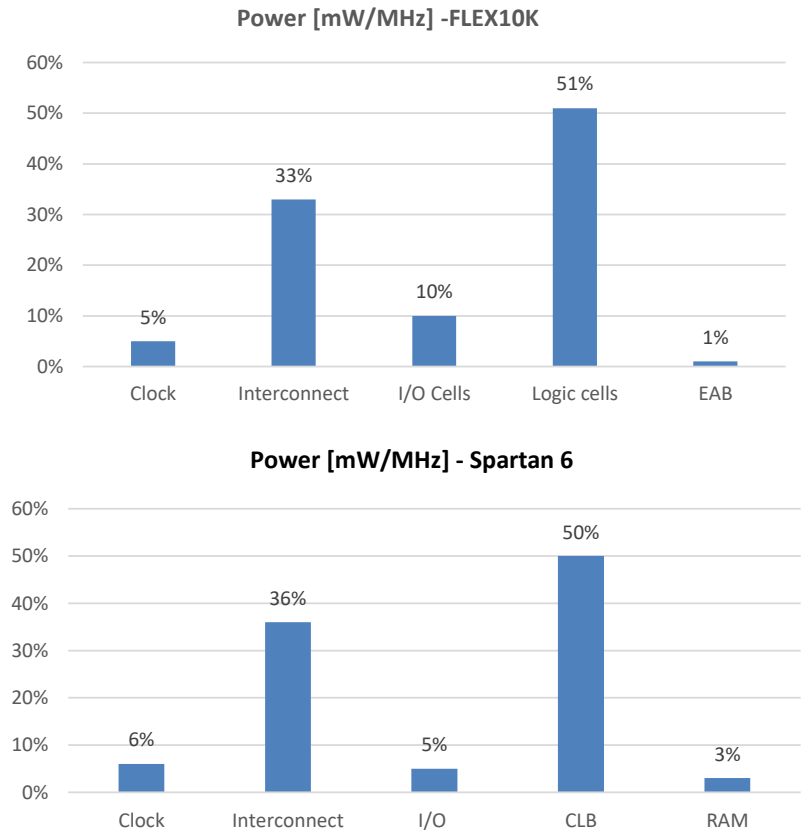


Fig.1 Power consumption of each component

With I/O toggling at higher data rates and logic toggling at faster frequencies, the charging and discharging of loads on and off chip becomes the main consumer of FPGA power. To effectively reduce total FPGA power, both static and dynamic power must be addressed while ensuring the FPGA's performance still meets design requirements. Dynamic power consumption can be calculated:

$$P_{dyn} = \left[\frac{1}{2} C \cdot V^2 + Q_{short-circuit} \cdot V\right] \cdot f \qquad (1)$$

Dynamic and I/O power dominate the FPGA's total power consumption. Because high-end FPGA designs tend to push the envelope in terms of bandwidth and performance, they use more logic running at a higher clock f. With I/O toggling at higher data rates and logic toggling at faster frequencies, the charging and discharging of loads on and off chip becomes the main consumer of FPGA power. To effectively reduce total FPGA power, both static and dynamic power must be addressed while ensuring the FPGA's performance still meets design requirements. The calculated average power consumption is expressed in the equation:

$$P_{avg} = \frac{1}{2}\sum_{i=1}^{nnumber\ of\ nets}(C_i f_i V^2) \quad (2)$$

where P_{avg} is the average value of dynamic power dissipation, C_i is the capacitance of the net, f_i is the average toggle rate and V is power supply value. It is important to use software tools that accurately predict the dynamic power consumption of a design. Unsophisticated power analysis tools simply model each circuit as a lumped capacitance. In contrast, Altera Power Play and Xilinx XPower Estimator are power analysis tools that use detailed dynamic power models: simulation-based power models and empirical power models.

3. Models for power consumption

In [11], gives a detailed power model that estimates static and dynamic power of the logic, routing, and clock network for a range of FPGAs with different architecture parameters is described. For static power, the model uses a first-order analytical technique which calculates leakage based on transistor size and various technology-specific parameters. For dynamic power, the model uses transistor-level capacitance information from the place and route tool and switching activity information.
In [12], a similar FPGA power model estimates static and dynamic FPGA power by calculating the power for each clock cycle using simulated switching activity information. This power model has been enhanced to support FPGAs with a programmable supply voltage and programmable threshold voltages.
In [13, 14, 15], high-level FPGA power models that use macro-models to estimate power are described. These models characterize the power consumption of various FPGA components, such as adders, multipliers, and programmable logic, for low power high-level synthesis. Based on this related works, in the explained models and the architecture of the programmable logic elements we have proposed to implement pipeline. First we have considered two pipeline structures: (1) 16-bit adder with a comparator and (2) between the adder and the absolute comparator, a stage of pipeline can be inserted after the adder. Apart the pipeline hardware implementation and I/O power supply we will focus also in the clock frequency parameter.

4. Experiment and measurements

Another factor is the majority of glitches were found on near-critical paths because they generally consisted of the largest number of LUT. For getting good results we used different designs realized

with Altera FLEX10K element and Power play analyzer estimation tool. Using a simple calculation of power, we see how small differences in structure affect a logical circuit. In this particular case the focus was on dynamic power dissipation of a logic circuit. Focusing on the problem, can locate and glitch nets can reduce the loss of their dynamic power. This estimate shows that this reduction in power consumption is achieved at the expense of increasing the delay to 3.23%. Experiments show that this small change in the structure of the circuit does not affect the operation of its logical and this technique can be used for optimization of power even in other elements of programmable logic. We have used an empirical calculation methodology used to obtain the current consumed by each internal sub-element of a CPLD/FPGA by starting with a simple design containing a minimum number of internal elements, fix toggling rate and constant room temperature. The static current measured for the Flex device is from 1.5 to 2.4 mA; and static current of the Spartan 6 devices is between 7.2 and 7.6 mA. The difference in the current consumption is computed to estimate the power consumed by this element. Interconnect must be the first to be measured because if the power consumed by interconnect is estimated first, then the power consumed by the others must be easily estimated.

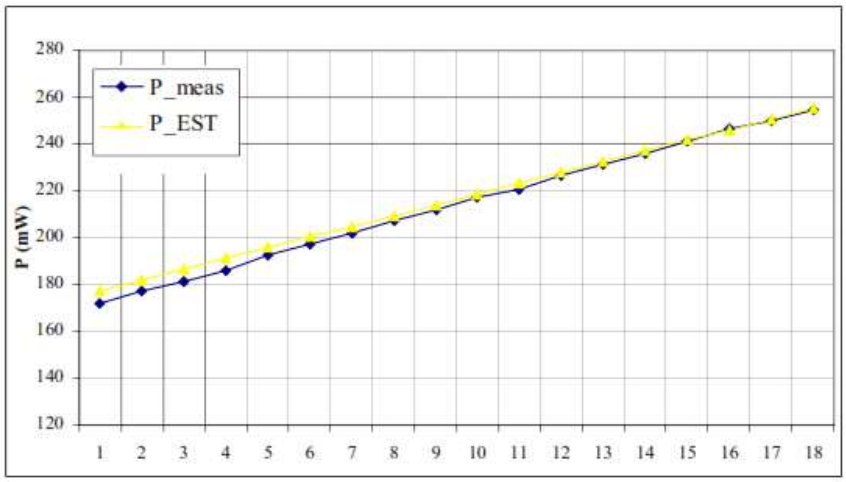


Fig.2 Power estimation and measurement for FLEX10K

After this measurement we have take in consideration pipeline and clock frequency 100 MHz, power consumption is measured for each supply voltage value from 5 volts to the minimum supply voltage is shown in the figure below.

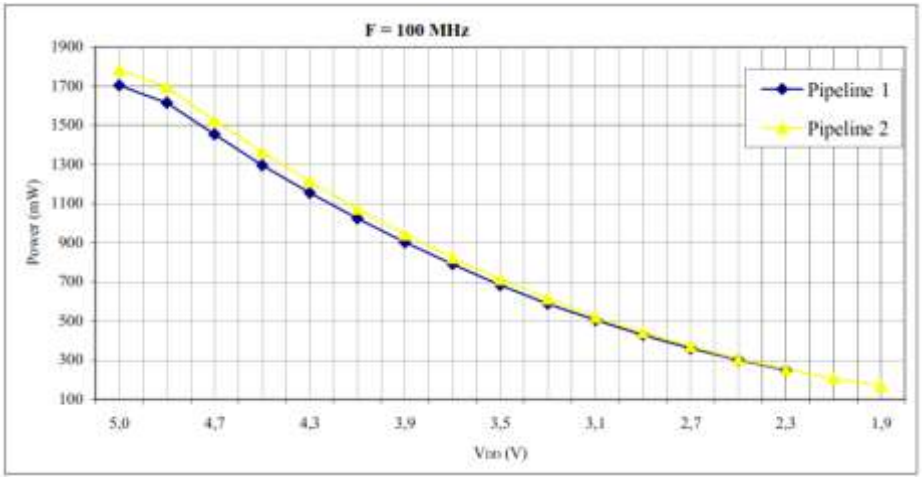


Fig.3 Power consumption for pipeline architecture

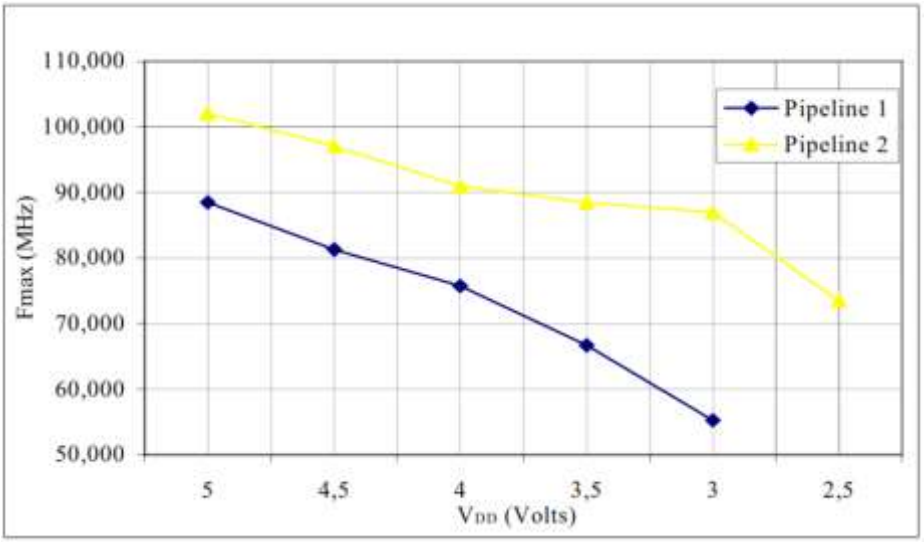
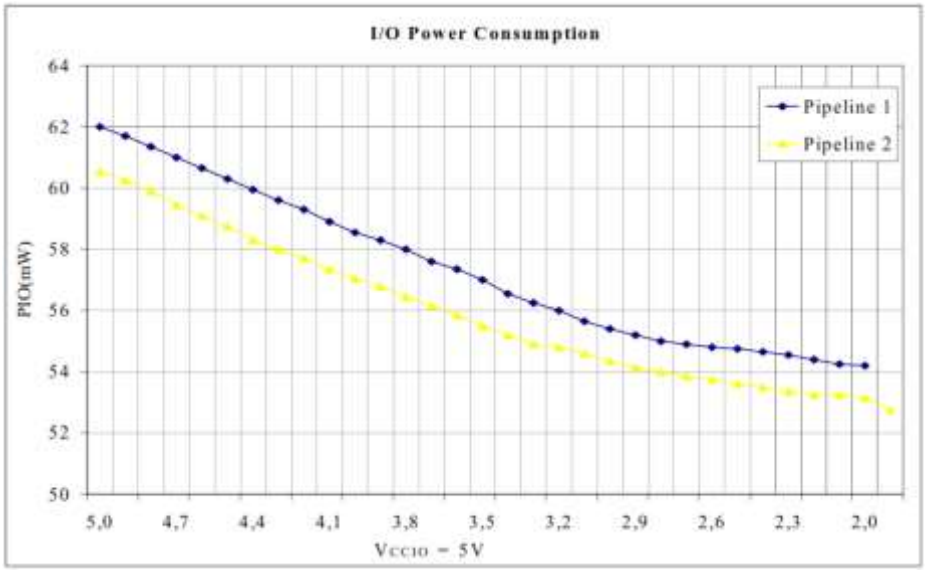


Fig.4 Maximum clock frequency of pipeline architecture



Conclusions

This technique proposes the use of pipeline architectures coupled with very low supply voltages to save power in FPGAs without performance loss. In order to improve this, the minimum supply voltage and the most optimal pipeline have to be applied. The I/O power consumption is similar to logic power because it depends on switching, load capacitance, frequency, and voltage. By reducing any one of these components, one can reduce the I/O power. As we can see from the graphics, the right pipeline architecture implementation and the low power supply for I/O component can lead to approximately in a reduction of 5% of power consumption. Finally, the best technique for reducing power consumption is low power supply for the element but this change is possible only in fabrication.

References

13. Underwood, K.D. Hemmert, K.S. 2004. Closing the Gap: CPU and FPGA Trends in Sustainable Floating-Point BLAS Performance. International Symposium on Field-programmable Custom Computing Machines (FCCM), California, USA.
14. Zhuo, L. Prasanna, V.K. 2005. Sparse Matrix-Vector Multiplication on FPGAs. International Symposium on Field Programmable Gate Arrays (FPGA), Monterey, CA, USA.
15. Alsolaim, A., Becker, J., Glesner, M., Starzyk, J. 2000. Architecture and Application of a Dynamically Reconfigurable Hardware Array for Future Mobile Communication Systems.

International Symposium on Field Programmable Custom Computing Machines (FCCM). Napa, CA.

16. R. I. Bahar, M. Burns, G. D. Hachtel, E. Macii, H. Shin, and F. Somenzi, "Symbolic Computation of Logic Implication for Technology-Dependent Low-Power Synthesis," Proceedings of the International Symposium On Low Power Electronics and Design, Monterey, 1996, pp. 163-168.
17. K.W. Kim, T. Kim, T.T. Hwang, S.M. Kang, C.L. Liu, "Logic Transformation for Low Power Synthesis," ACM Transactions on Design Automation of Electronic Systems, 2002, pp. 265-283.
18. H. Li, W-K. Mak, and S. Katkoori, "LUT-Based FPGA Technology Mapping for Power Minimization with Optimal Depth," IEEE Computer Society Workshop on VLSI, 2001, pp.123-128.
19. Z-H. Hong Wang, E-C. Liu, J. Lai, and T-C. Wang, "Power Minimization in LUT-Based FPGA Technology Mapping," ACM/IEEE Asia South Pacific Design Automation Conference, 2001, pp. 635-640.
20. A. P. Chandrakasan, M. Potkonjak, R. Mehra, J. Rabaey and R. Brodersen, "Optimizing Power Using Transformations," IEEE Transaction on Computer Aided Design of Integrated Circuits and System, Vol. 14, No. 1, January 1995,pp. 12-31.
21. FPGA Power Management and Modeling Techniques, Altera White Paper, www.altera.com/literature/wp/wp-01044.pdf
22. J. H. Anderson and F. N. Najm, "Switching Activity Analysis and Pre-Layout Activity Prediction for FPGAs," ACM/IEEE Workshop on System Level Interconnect Prediction (SLIP), Monterey, CA, 2003, pp. 15 - 21.
23. K.K.W. Poon, S.J.E. Wilton, and A. Yan, “A detailed power model for field-programmable gate arrays,” ACM Trans. on Design Automation of Electronic Systems, vol. 10, no. 2, pp.279-302, April 2005.
24. F. Li, D. Chen, L. He, and J. Cong, “Architecture evaluation for power-efficient FPGAs,” Proc. ACM Int. Symp. on Field Programmable Gate Arrays, pp.175-184, 2003.
25. D. Chen, J. Cong, and Y. Fan, “Low-power high-level synthesis for FPGA architecture,” Low Power Electronics and Design, pp. 134-139, 2003.
26. D. Chen, J. Cong, Y. Fan, and Z. Zhang, “High-level power estimation and low-power design space exploration for FPGAs,” Proc. Asia South Pacific Design Automation Conf., pp. 529-534, 2007.
27. D. Chen, J. Cong, F. Li, and L. He, “Low-power technology mapping for FPGA architectures with dual supply voltages,” Proc. ACM Int. Symp. on Field-Prog. Gate Arrays, pp. 109-117, 2004.