Impact of different power reduction techniques at architectural level on modern FPGAs

Henryk Blasinski
Department of Microelectronics and Computer Science
Technical University of Lodz
Poland
hblasinski@dmcs.pl

Frederic Amiel, Thomas Ea
Institut Superieur d'Electronique de Paris
France
frederic.amiel@isep.fr, thomas.ea@isep.fr

***Abstract*—In this work, the impact of different power reduction techniques at architectural level on modern FPGAs is evaluated. Starting from the 65 nm technology node, manufacturers have implemented a number of technologies to reduce overall power consumption. In our study we are performing real life measurements and proposing several architecture level power reduction techniques. Our solutions are then implemented and evaluated in an image processing intellectual property block. Obtained results demonstrate that certain techniques such as clock gating may result in up to 90% power savings.**

***Index Terms*—FPGA, low-power, IP**

I. Introduction

With the reduction of the transistor size, electronic components are supposed to dissipate less and less power. On the other hand they become more and more complex, and contain an increased number of transistors in order to support more and more complex applications. In embedded systems for example, while the technology should allow to decrease the power consumption of electronic boards, on the contrary, we notice its increase. With the growth in complexity also came the increased static power consumption of each component. Negligible for a long time, this static part of the dissipated power of electronic devices can now be quite high, having the same order of magnitude as the dynamic power. At last FPGAs, dissipating by nature far more than ASICs yet remain indispensable for most electronic designs, start to use various techniques to reduce their dissipated power [1]–[3].

The purpose of this article is to estimate, measure and test the impact of the FPGA end user techniques which permit to reduce the power consumption on FPGA components. We are going to demonstrate the interest in their applications as well as their limits. Current research in the low power FPGA design focuses rather on silicon level improvements [4] or variable power supply voltage techniques [5]. As opposed to [4] we are providing real life measurement results rather than simulations.

This article is organised as follows: Section 2 presents theoretical elements which explain the power consumption of electronic devices, and exposes briefly the power reduction techniques used by the FPGA manufacturers. Then an enumeration of the possible power reduction approaches at architectural level is presented. Section 3 presents platforms we used, as well as the methodology for the various measurements we made in order to estimate the influence of each of the previously explained power reduction techniques. The final paragraph concludes on these approaches and proposes some low-power system design strategies.

II. Power consumption & reduction techniques in FPGAs

A. Power consumption in FPGAs

The two sources of power consumption in FPGAs are static and dynamic dissipation. Most of power is lost in the static part as leakage current and it is mainly due to transistor size shrinking, low supply voltage and regular architecture of the FPGA. The dynamic power dissipation is due to parasitic capacitance charging and discharging across conduction channel resistances. At 65 nm node, static power (resp. dynamic power) represents 60% (resp. 40%) of total FPGA dissipation. Typically, the total power consumption among FPGA blocks can be attributed to different blocks as shown in Fig. 1. User logic and signal routing constitute 70% of the FPGA power consumption.

B. Technology & micro-architecture for power reduction in FPGAs

FPGA vendors and their manufacturers use various techniques to optimize static and dynamic power dissipation.

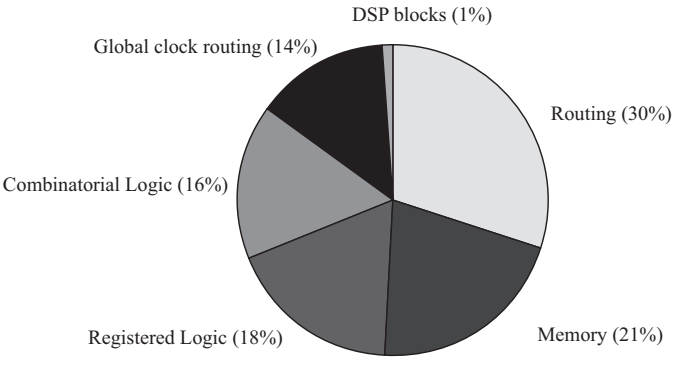


Fig. 1. Average core dynamic power dissipation by block type in Stratix III devices at a 12.5% toogle rate [6].

From a technological point of view, manufacturers use the same technology as in the ASIC field. For example, they use multi-threshold voltage V_t transistors (low V_t for performance and high V_t for reducing leakage current), variable transistor length, various oxide thickness (triple gate oxide, super-thin oxide), low-k dielectric, strained silicon, copper routing. That lead to some specific FPGAs called 'low power FPGA families', optimized for power dissipation at the cost of lower performance (for example, Cyclone III from Altera). Certain high density, high performance FPGAs implement programmable power technology allowing a high speed or low power modes by adjusting back bias voltage of the transistors and also selectable core voltage (0.9 V or 1.1 V on Stratix III from Altera).

From vendors point of view it is important to design original micro-architectures and to develop 'intelligent' tools targeting power consumption according to those architectures. Increasing local interconnect, modifying the number of LUT inputs, setting up new dedicated blocks such as multipliers, DSP and RAM, programmable I/O, idle mode, having optimized synthesis, place and route tools reduce significantly power dissipation [6].

C. Power reduction techniques at architecture level

Similar techniques for power reduction used in the ASIC world are applied in FPGAs [7] but their application is not automatic and a trade-off has to be made between performance, extra logic cost and power dissipation. Techniques such as clock gating, state machine encoding, word encoding, retiming, pipelining, computational blocks sharing, operand isolation, pre-computation, logic rearrangement, etc. have a significant impact on power dissipation [8].

III. Experiments & Results

A. Platforms

For the purpose of our study we have selected two different 65 nm FPGA circuits belonging to two distinct families. Cyclone III is a low cost, low power solution offering a balance between performance and cost. Stratix III on the other hand is the high end chip optimised solely for speed and computational power.

The first of the selected chips is Cyclone III EP3C25, having approximately 25k logic elements and about 200 pins. It is available on a development board which provides tools to measure power consumption by means of probe resistances built into the board. The user has to provide his own measurement equipment to read out power consumption values.

Out of the Stratix III family the EP3SL150F1152 was chosen. It has about 113k logic elements and 730 pins. The biggest difference lies in the architecture of logic modules (6-input instead of 4-input LUTs) and operating frequency, which for the Stratix III may reach 550 MHz. Stratix III comes on a development board with various peripherals (memories, buttons, LEDs, displays, etc.) and more importantly dedicated circuitry for power consumption measurements. It consists of a MAX II CPLD device connected to a 24 bit ADC measuring voltage across probe resistances and displaying those values on a separate numeric display. Measurement systems are completely embedded within development boards and allow to measure current drawn by the core and various I/O banks of the FPGA.

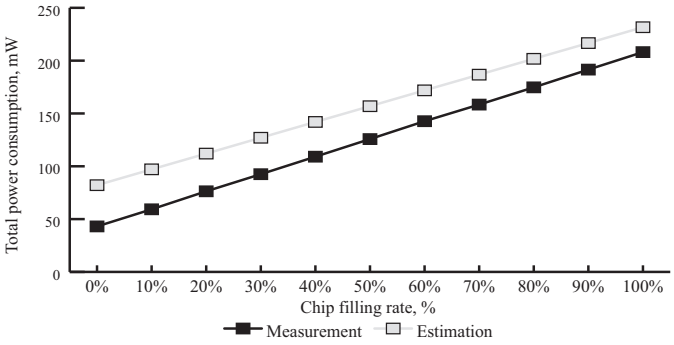


Fig. 2. Comparison between Cyclone III EP3C25 power measurement and estimation.

B. Power consumption evaluation

In order to verify standard power reduction techniques on the FPGA core only, we filled a Cyclone III EP3C25 chip up to 90% with counters and adders supplied with random number generators. Fig. 2 confirms the liability of the manufacturer power estimator. The offset between the two curves is most likely due to the temperature difference between the simulation and the real measurement. In practice it is very hard to assure a known and constant temperature of the FPGA core, it can be only achieved in an isolated test chamber. The measured power consumption versus frequency (Fig. 3) seems to demonstrate a non linear relation. Its origin may also be of thermal phenomena, which is partially confirmed by EDA tools. In our case Quartus II handles the package type and the variation of parameters with respect to the temperature. As it appears an increased frequency also causes the increase of the core temperature which in turn results in an even larger power consumption. This non-linear trend is also present in Fig. 4, this time however in the form of non even distribution of directional coefficients of straight lines. The test-bench described in Fig. 4 also confirms the interest of clock gating, if implemented it allows to 'turn off' parts of the FPGA real estate and consequently reduce the active area.

A very similar analysis was performed for a design with a general purpose processor and intellectual property (IP) blocks. Comparable results have been obtained.

C. Intellectual Property blocks

In order to verify power reduction techniques in practice an intellectual property (IP) block was created and implemented in the Stratix III platform. The IP was realising in hardware an image processing function related to the iris segmentation and identification algorithm [9] [10]. Typical architectures with IP blocks involve a general purpose processor - gpp (for example NIOS II) connected to the IP and other peripherals (memories,

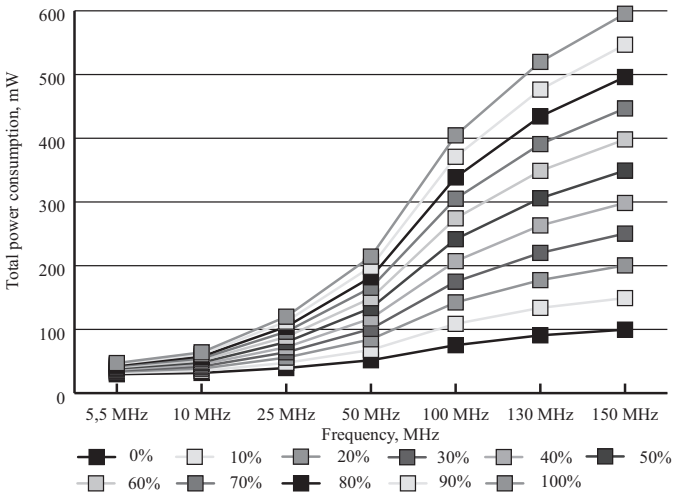


Fig. 3. Cyclone III EP3C25 power consumption vs. frequency for various chip filling rate.

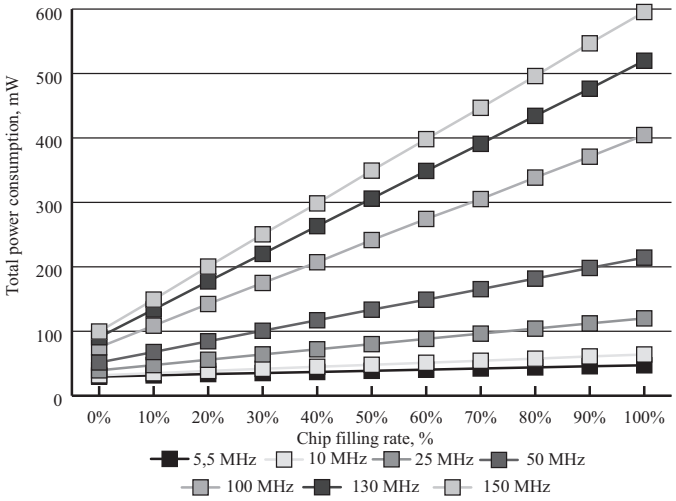


Fig. 4. Cyclone III EP3C25 power consumption vs. chip filling rate for various frequencies.

DMA, etc.) via a bus. The gpp is responsible for configuration of IPs, which in turn realize given functions and usually store results in the memory for further blocks to process.

The IP was realising the basic morphological operations on an image: erosion and dilation. Their thorough description is in [11]. However, in very simple terms the function consists in selecting the minimal (erosion) or maximal (dilation) value from a neighbourhood, specified with a mask, of an image pixel and assignment of that value to this pixel. This operation can be decomposed into a number of comparisons and memory read/write operations, making its hardware implementation relatively easy. However with an increasing image and mask sizes the amount of data needed to be processed grows very rapidly. For example for a 200 x 200 pixel image and a 23 bit mask one needs to process approximately 21 megabytes of data. The details of the internal architecture of the IP can be found in [12].

TABLE I
EROSION EXECUTION TIME COMPARISON (IN SECONDS).

Mask	IP 100 MHz	NIOS II 100 MHz	DSP 1.2 GHz
Square, 5	0.083	0.427	0.031
Square, 11	0.142	1.486	0.141
Square, 23	0.247	5.343	0.610
Disc, R=3	0.093	0.283	0.047
Disc, R=6	0.161	1.217	0.167
Disc, R=11	0.247	6.020	0.530

TABLE II
ENERGY PER OPERATION (IN JOULES).

Device	Operation		50 MHz	100 MHz
Stratix III	Erosion	Hardware	0.72	0.45
		Software	15.7	8.43

D. Measurement results

The main reason behind hardware implementation is the improvement of the execution time of a given operation. The same could be observed for the tested IP. It is interesting to note that the same function realised in software on a gpp such as NIOS II is executed many times slower than its hardware counterpart. The hardware execution time is of the same order of magnitude as for the software realisation on much more advanced and faster circuits such as DSPs (TI6455 running at 1.2 GHz) or even personal computers (Core 2 Duo 2.0 GHz) (Table I). It is important to note that this architecture was not designed to deliver the lowest power consumption possible, but to be competitive in terms of algorithm execution time.

Since the execution time has been drastically reduced due to the hardware implementation it balances out the slight increase in power consumption caused by a larger overall system size. Hence the energy required to perform a single operation is much lower for the hardware implementation (Table II). Since static power consumption is more or less constant it appears to be beneficial to increase the operating frequency. The above conclusion holds for both families of devices, Cyclone III and Stratix III . In this approach, even more energy could be saved by switching the gpp into the 'idle' state. Fig. 5 shows the interest of the idle mode, in which the processor clock is switched off. A comparison is made between such 'idle' state and running an infinite 'while(1)' loop.

Similarly the efficiency of power reduction techniques, especially clock gating was tested. Clock gating was implemented into the IP block as a VHDL circuit description (Fig. 7). It was then up to the EDA tools to properly synthesize the entire design. Clock gating technique offers very promising results and in the most extreme case, in which the IP is in the 'idle' state, can reduce its power consumption even by 90% (Fig. 6(a)). Of course power reduction will be less impressive, when the IP is executing its function. The nature of erosion/dilation is such that most of clock domains had to

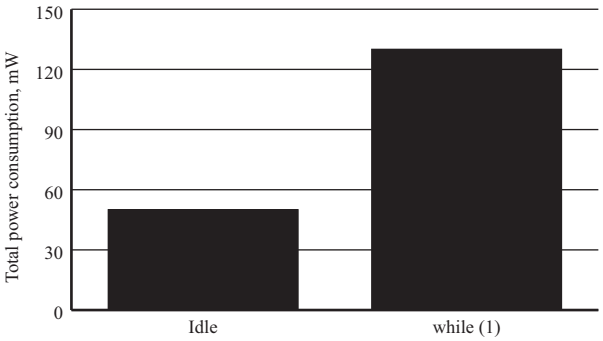
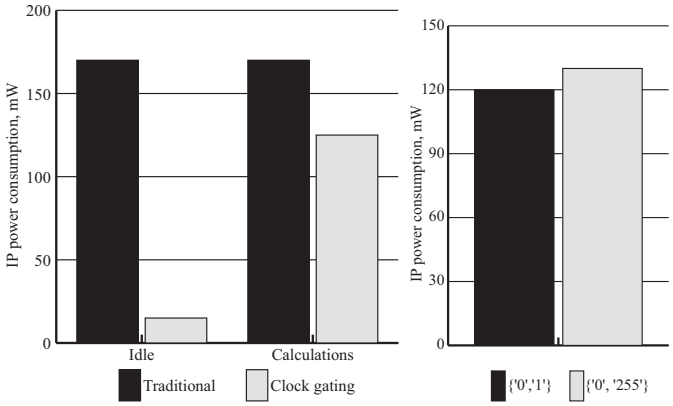


Fig. 5. General purpose processor power consumption.



(a) Clock gating (b) Signal activity

Fig. 6. Erosion/dilation IP power consumption reduction analysis.

remain on.

During laboratory tests several other power reduction techniques were also evaluated. One approach involves designing such architectures that limit unnecessary signal transitions. This methodology was tested for binary input images. Image pixel data was encoded in two ways: {'0', '1'} and {'0', '255'}. In the latter encoding style, when encoding image pixel as '255' a huge redundancy of information occurs, since it would be enough to use a single signal instead of eight . Experimental results reveal that the influence of signal activity is non-negligeable (Fig. 6(b)), in this particular case it accounted for about 7% of the IP power consumption.

IV. Conclusions

The goal of our study was to propose and evaluate several low-power FPGA design strategies. According to our experimental results power consumption increases linearly with chip filling rate and with frequency. In order to minimize the effect of static power consumption the component should be maximally filled. Also because of the static power consumption it is beneficial to increase the operating frequency of the system. Moreover, constant low temperature should be assured, otherwise power consumption may be significantly increased, by even more than 100%. Finally designers should avoid any software solutions whenever hardware implementations can be made. Not only such algorithms are faster, but in our particular

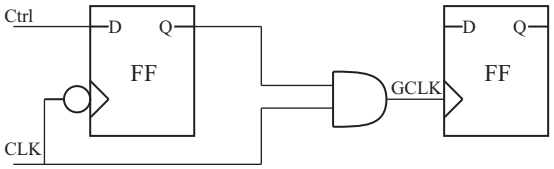


Fig. 7. Clock gating diagram. FF - flip-flop; Ctrl - clock enable signal; CLK - clock signal; GCLK - gated clock signal.

case we managed to reduce the energy required by operation by a factor of 20. Still if a gpp is needed it should be used as little as possible, preferably left in the idle state.

Out of all the low power techniques tested, clock gating is the most promising one, offering a serious reduction of IP power consumption of up to 90%. However the number of clock domains is limited by the architecture of the FPGA (number of clock trees). The designer should also take into account low signal transition data encoding in order to minimize internal signal activity. In our particular design we gained about 10%.

In future works we would like to evaluate the influence of various power reduction techniques on FPGA I/Os. Moreover up to now the temperature effect has been estimated on the base of simulations, real life experiments with temperature probes would provide more detailed results.

References

[1] "Power optimization in Stratix III FPGAs," Altera Corporation, application note, 2007.
[2] "Achieving low power in 65-nm Cyclone III FPGAs," Altera Corporation, white paper, 2007.
[3] M. Klein, "Power consumption at 40 and 45 nm," Xilinx Corporation, Tech. Rep., 2009.
[4] T. Tuan, A. Rahman, S. Das, S. Trimberger, and S. Kao, "A 90-nm low-power fpga for battery-powered applications," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 26, no. 2, pp. 296–300, Feb. 2007.
[5] Y. Lin, Y. Hu, L. He, and V. Raghunat, "An efficient chip-level time slack allocation algorithm for dual-vdd fpga power reduction," in *Low Power Electronics and Design, 2006. ISLPED'06. Proceedings of the 2006 International Symposium on*, Oct. 2006, pp. 168–173.
[6] *Quartus II Handbook, v.9.1, chapter 11. Power Optimization.* Altera Corporation, 2009.
[7] A. Amara, F. Amiel, and T. Ea, "FPGA vs. ASIC for low power applications," *Microelectronics Journal*, vol. 37, no. 8, pp. 669–677, August 2006.
[8] C. Piguet, *Low-power Electronics Design*. CRC Press, 2005.
[9] E. Rydgren, T. Ea, F. Amiel, F. Rossant, and A. Amara, "Iris features extraction using wavelet packets," in *Proc. IEEE International Conference on Image Processing ICIP'04*, vol. 2, October 2004, pp. 861–684.
[10] F. Rossant, M. T. Eslava, T. Ea, F. Amiel, and A. Amara, "Iris identification and robustness evaluation of a wavelet packets based algorithm," in *Proc. IEEE International Conference on Image Processing ICIP'05*, vol. 3, September 2005, pp. 257–260.
[11] R. Gonzales and R. Woods, *Digital Image Processing*. Prentice Hall, 2008.
[12] H. Blasinski, F. Amiel, and T. Ea, "Comparative study of power reduction of various solutions of an image processing IP on modern FPGAs," in *Proc. Faible Tension Faible Consommation Conference FTFC'08*, May 2008.