

Tracking the Pipelining-Power Rule along the FPGA Technical Literature

Eduardo Boemo
Escuela Politécnica Superior
Universidad Autónoma de Madrid
Spain
Tel: 34 914976213
eduardo.boemo@uam.es

Juan P. Oliver
Facultad de Ingeniería
Universidad de la República
Uruguay
Tel: 598 27110974
jpo@fing.edu.uy

Gabriel Caffarena
Escuela Politécnica Superior
Universidad CEU San Pablo
Spain
Tel: 34 913726435
gabriel.caffarena@ceu.es

ABSTRACT

This work reviews the contributions of power-oriented pipelining over the last two decades, and adds up-to-date results on 65 nm and 45 nm FPGAs. The data show that power consumption can be reduced by a factor between 0.1 and 0.8 using different levels of pipelining. More than 34 experiments, developed in 12 laboratories in 8 countries during 17 years are summarized. This paper also contributes to this research topic adding updated results for Altera 65 nm Cyclone III and Xilinx 45 nm Spartan-6 devices.

Categories and Subject Descriptors

B.5.1 [Hardware]: Register-Transfer-Level Implementation – styles, pipeline.

General Terms

Algorithms, Measurement, Performance, Design, Experimentation.

Keywords

Pipeline, power consumption, glitches, logic depth, FPGA, look-up table, LUT.

1. INTRODUCTION

The use of pipelining to increase throughput is a well-known concept, derived from the Henry Ford T-Model assembly organization, dated from 1913. With a delay of more than 60 years, the Ford ideas were adapted to electronics circuits by Leonard Cotton in 1965 [1].

In the decade of the '90, the pipeline technique surfaced again in a different field: Low-Power Design. The original idea was to minimize power by lowering the power supply and –

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from Permissions@acm.org.

FPGAWorld '13, September 10 - 12 2013, Stockholm, Sweden
Copyright 2013 ACM 978-1-4503-2496-0/13/09 ...\$15.00.
<http://dx.doi.org/10.1145/2513683.2513692>

simultaneously – by applying pipelining to reach again the original speed of the circuit. Thus, the worse commutation speed associated to a lower voltage could be compensated. However, there is different straightforward effect of pipelining to minimize power: It is based on the elimination of glitches or spurious transitions.

Glitches occur because logic signals do not arrive to gates – or look-up tables in FPGAs – synchronically, since they traverse different levels of combinational logic, or the combinational paths' delays are not equalized. Even though they do not add errors to well-designed synchronous circuits, they contribute significantly to dynamic power consumption (from 20% to 70%) [2] due to they effectively increase the original switching activity of signals [3]. Pipelining acts twofold: first, it reduces the logic depth of combinational paths. Thus, it decreases the probability of glitches. And second, it prevents the generated spurious transitions to be propagated from one pipeline stage to the next.

The importance of glitching is also revealed when obtaining high-level models of the power consumption of arithmetic components. Neglecting hazard activity leads to serious underestimation of the power dissipated. When high-level models include the effect of spurious transitions, the estimation accuracy is dramatically increased: the error in the power consumption estimation for LUT-based non-pipelined multipliers can be reduced from 60% to 15% when glitching is accounted for [4]. Moreover, if the models are applied to DSP algorithms that make use of several arithmetic components, the error in the estimation can reach up to 97% when glitches are overlooked, while if the intra- and inter-routing spurious transitions are considered, the error drops to 30% [5].

In this paper, we present a synopsis of the several published results over the last two decades regarding the glitch power optimization through pipelining. The starting point is difficult to determine. A. Chandrakasan, S. Sheng, and R. Brodersen clearly state the principle of the rule in 1992 [6]: “*As an added bonus, increasing the level of pipelining also has the effect of reducing logic depth and hence power contributed due to hazards and critical races*”. The idea is first quantified by J. Leijten, J. van Meerbergen, and J. Jess in 1993 [7], and then summarized in [8]. They simulate the activity and estimate the power of four pipeline versions of a direction detector circuit, concluding that “*a retiming frequency which is optimal for power can be found ... running it at the original clock frequency may result in lower power dissipation*”.

The popularization of field-programmable gate arrays (FPGA)

detonates the experiments about pipelining for glitch reduction [9-22]. FPGAs are composed of a large collection of different wiring networks that are able to bond different elements of the chips. The large area reserved for these programmable interconnections leads to high wiring capacitances, as well as abundant presence of glitches. These two factors, together with the abundance of flip-flops in current FPGAs, turned pipelining into an ideal technique for this technology. The ideas did not reach manufacturers databooks and application notes until year 2000. For example, Actel states in [14] that “added registers not only speed-up the design but also help to reduce the switching activity”. Xilinx recommends in [21] “By using more pipelining between layers of logic, you can block a glitch from propagating to other structures, which will reduce dynamic power”. Finally, Altera indicates that “reducing inadvertent glitching of logic within a glitch-prone design significantly reduces dynamic power ... A second method is introducing pipelining to reduce the combinatorial logic depth. Pipelining reduces design glitches by inserting flipflops into long combinational paths” [18].

The rest of the paper is organized as follows: In section 2, we present an overview of pipelining in the low-power design arena. An analysis and classification of the most relevant results are presented. Updated results on 45 nm FPGAs are summarized in Section 3. Finally, the last section adds new results corresponding to a Spartan 6 FPGA and a Cyclone-III FPGA device.

2. PIPELINING AS LOW-POWER DESIGN TECHNIQUE TO MINIMIZE GLITCHES ALONG 20 YEARS

The designers of digital systems manage empirical rules and principles rather than universal and invariant laws. These rules do not guarantee to achieve the best solution in any situation, but, in general, they can contribute to reduce the number of iterations in the design cycle. The relationship between pipelining and power reduction might as well be considered one of these rules. It has been verified during nearly 20 years in at least 35 published experiments using 8 different microelectronic fabrications (1, 0.8, 0.42, 0.35, 0.22, 0.13, and 0.18 μm , as well as 90 nm), developed by 24 researchers of 13 laboratories situated in 8 countries (Argentina, Canada, Holland, Portugal, Spain, UK, USA and Uruguay). Naturally, “the exception that proves the rule” also exists: It is only necessary to run into a technology where the synchronization power overhead produced by the pipelining surpasses the datapath glitch power reduction. An example in 1 μm Standard Cells is [13].

The following values, even though have been picked-up from actual measurements, must be considered in a qualitative way. That is, only to illustrate the efficacy of the rule. Most of the experiments use primarily binary multipliers as benchmark circuits, but other valid blocks have been tested. However, the problem resides in a lack of uniformity in the measurement criteria.

The power of these circuits can be completely described separating the total consumption in the datapath, synchronization, static, and off-chip power components. Some researchers consider that the tested block will be embedded in a complete system; therefore, they do not compute the off-chip power. This is done by simply not measuring the current in the power supply rails that separately feeds the pads of the FPGA. Other researchers, interested in the isolation of the pipeline effect, subtract the values

of both off-chip and static power to the measured results. Thus, the effect of pipelining results highlighted with respect to those that report total power. Finally, some studies present simulation results based on the output of power estimators.

These diverse results do not invalidate the central idea of this paper: pipeline has been proved as an efficient low-power design technique along nearly 20 years. However, the readers must avoid the temptation of cross comparisons among different research groups, years, technologies, or FPGA manufacturers.

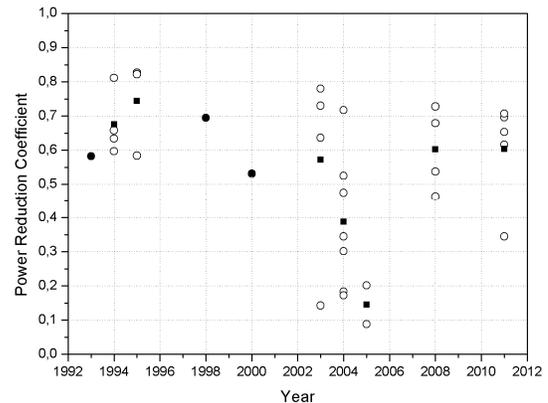


Figure 1. Summary of power reduction by pipeline in 35 published results.

Figure 1 shows a summary of experiments about pipelining and power. The Y-axis represents the power reduction factor, expressed by the ratio between the power consumption of the best pipeline version and the power consumption of the original combinational circuit. Dark square points represent the average value for each year.

It is interesting to note the 8-year gap between the first few applications of power-oriented pipeline and the second resurgence of the topic, related to its adoption by major commercial companies.

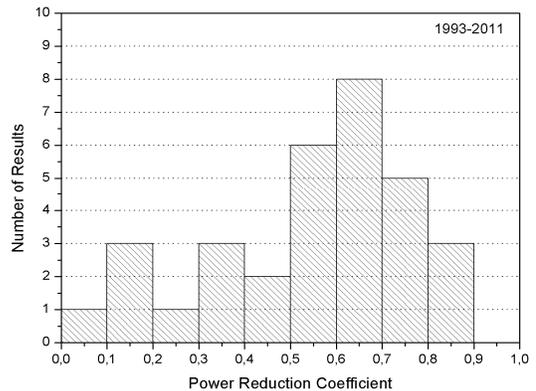


Figure 2. Histogram of the best results reported about the effect of pipeline

(All papers: simulation and measurement results)

The distribution of power reduction achieved is depicted in

Figure 2. It can be seen that extreme results are not frequent, while power reductions around 50% tend to be obtained.

In Figure 3, it is shown the area penalty of the pipelined versions, measured as the ratio between the number of flip-flops of the pipeline version and the corresponding value of the original circuit. The results show a significant increase of FF use, but this does not necessarily imply an additional economic cost in FPGAs implementations. On one hand, these chips have plenty of registers. And, on the other hand, the configuration shift-registers associated to each LUT are also available for pipeline implementations.

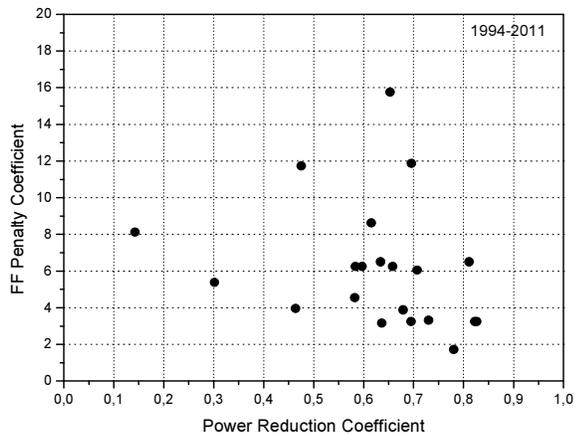


Figure 3. Area penalty measured as extra FFs with respect to the original version. Only experiments with moderate pipelining have been included

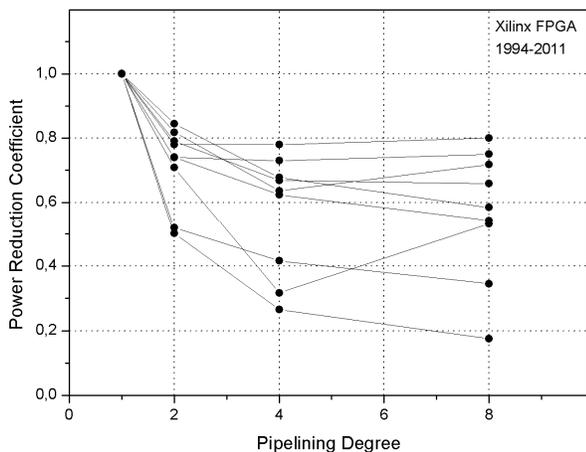


Figure 4. Power reduction factor vs. different levels of moderated pipelining degree (Xilinx FPGAs)

The effect of moderated pipeline for Xilinx, Altera and Actel FPGAs is depicted in Figures 4 and 5. A degree of pipelining of 2 or 4 always produces an important power reduction with no significant cost in latency and extra flip-flops. It can be noted that the curves are non-monotonic, and extending the pipeline degree leads to a power increment. The reason for that is that the glitching power reduction due to pipelining does not make up for

the clock power increment produced due to the increase in FF usage. Hence, the importance of finding the optimal pipeline degree of a circuit.

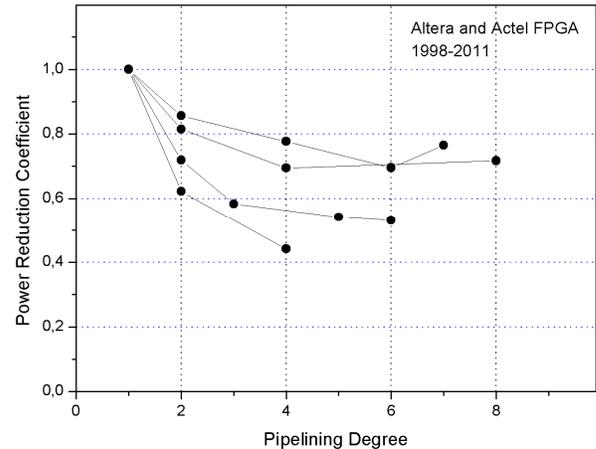


Fig. 5. Power reduction factor versus different levels of moderated pipelining degree (Altera and Actel FPGAs)

In summary, the benefits of pipelining as a low-power design technique have been clearly proven throughout the last two decades, but important efforts must be put into getting to a consensus on the power measurement setup as well as on the way to assess the power reductions obtained.

3. UPDATED RESULTS ON 65 nm AND 45 nm FPGAS

Here, we present a new experimental set of measurements. The devices used were an Altera 65 nm Cyclone III EP3C16F484 and a Xilinx 45 nm Spartan-6 XC6SLX16-2. Unsigned integer multipliers with different word sizes were utilized as benchmark circuits (e.g. 32×32 , 54×54 and 64×64).

The only external input to the FPGA was the clock signal. The inputs of the multipliers were generated internally in the FPGA by means of a linear feedback shift register that was used in all the studied cases. Also, all the inputs and outputs of the multipliers were registered.

The power consumption was obtained by measuring the current from the internal core power supply by means of a serial shunt resistor, and, multiplying the acquired value by 1.2 V. The measurement was repeated several times and averaged. The maximum observed variation was ± 1 in the third significant digit and the relative error in the current measures was smaller than 1.5%.

Tables 1 and 2 show the results. The best values obtained in the experiments have been already included in the figures and analysis from the previous section. The results yield that power reduction factor down to 0.34 are obtained with moderate area increase (i.e. FFs). As in figures 4 and 5, the increment in pipeline degree does not always lead to power reductions – but there is always an improvement with respect to the original case. Important benefits can be obtained for small number of pipeline stages (i.e. from 2 to 5), highlighting the effectiveness of this low-power design technique.

Table 1. Pipelining results for 65 nm FPGA (Cyclone III).

mult 32×32 50 MHz			
Pipeline stages	mA	FF	Power reduction factor
1	52.9	128	1.00
2	39.2	541	0.74
3	37.4	822	0.71
4	40.1	1001	0.76
5	40.5	1300	0.77
6	41.9	1458	0.79
7	54.2	1524	1.02

mult 54×54 50 MHz			
Pipeline stages	mA	FF	Power reduction factor
1	119.6	188	1.00
2	102.4	953	0.86
3	93.0	1509	0.78
4	92.9	1776	0.78
5	90.3	2338	0.76
6	83.2	2672	0.70
7	91.5	3456	0.77

mult 64×64 50 MHz			
Pipeline stages	mA	FF	Power reduction factor
1	164.3	256	1.00
2	149.6	1181	0.91
3	116.0	1718	0.71
4	102.3	2401	0.62
5	110.8	3144	0.67
6	110.7	3773	0.67
7	111.5	4208	0.68

Table 2. Pipeline results for 45 nm FPGA (Spartan-6).

mult 32×32 50 MHz			
Pipeline stages	mA	FF	Power reduction factor
1	43.2	128	1.00
3	34.1	289	0.79
5	26.6	1103	0.62
7	26.4	1182	0.61
9	---	---	---

mult 54×54 50 MHz			
Pipeline stages	mA	FF	Power reduction factor
1	67.2	216	1.00
3	34.9	666	0.52
5	27.9	2346	0.42
7	---	---	---
9	23.1	3206	0.34

4. CONCLUSIONS

In this paper, we have reviewed twenty years of pipelining as a low-power design technique. The results obtained by variated research groups in the area have been analyzed. Up-to-date results on 65 nm and 45 nm FPGA devices have also been included.

The immediate conclusion of the work is that pipelining is an effective low-power technique. The power consumption reductions obtained by simply applying a few levels of pipelines are staggering with minimum cost increase.

Also, a lack of uniformity in the power measurement setup (i.e. simulation vs. actual measurement), as well as in the way to assess the achieved power reductions (i.e. total power vs. dynamic power), showed up while collecting and organizing the data used for this review. This situation makes difficult the comparison of the reported results.

5. ACKNOWLEDGMENTS

This work has been supported by the Spanish Ministry of Science and Innovation under contract TEC2007-68074-C02-02, the Uruguayan Agency for Research and Innovation (ANII) under grant PR-POS-2008-003, and project Banco Santander - University CEU San Pablo USP-BS PPC05/2010. The authors wish to thank Juanjo Noguera from Xilinx and David Karchmer from Altera for donation of FPGA boards. We also want to recognize the valuable and fast technical support of Gabriel Cutillas at Avnet Iberia.

6. REFERENCES

- [1] L. W. Cotton: Circuit implementation of high-speed pipeline systems. In: *Fall Joint Computer Conference AFIPS* (1965).
- [2] A. Shen, A. Kaviani, and K. Bathala: On average power dissipation and random pattern testability of CMOS combinational logic networks. In: *IEEE International Conference on Computer-Aided Design*, pp. 402–407 (1992)
- [3] L. Shang, A. S. Kaviani, and K. Bethala: Dynamic power consumption in Virtex II FPGA family. In: *International Symposium on Field-Programmable Gate Arrays FPGA*, pp. 157–164 (2002)
- [4] R. Jetvic, C. Carreras and G. Caffarena: Fast and Accurate Power Estimation of FPGA DSP Components based on High-Level Switching Activity Models. *International Journal of Electronics*, vol. 95(7), pp. 653-668 (2008)
- [5] A.A. Gaffar, J.A. Clarke and G.A. Constantinides: Modeling of glitch effects in FPGA based arithmetic circuits. In: *IEEE International Conference on Field Programmable Technology*, pp. 349 – 352 (2006)
- [6] A. Chandrakasan, S. Sheng, and R. Broderse: Low-Power CMOS Digital Design. *IEEE J. of Solid-State Circuits*, vol. 27(4) (1992)
- [7] J. Leijten: *Analysis of Transition Activity and Power Dissipation in Synchronous Logic Circuits*. Nat. Lab. Technical Note, no. 339/93, Philips Electronics N.V. (1993)
- [8] J. Leijten, J. van Meerbergen and J. Jess: Analysis and reduction of glitches in synchronous networks. In: *European Design and Test Conference* (1995)
- [9] E. Boemo, G. González de Rivera, S. Lopez-Buedo and J. Meneses: Un estudio sobre circuitos segmentados en FPGAs. In: *Design of Circuits and Integrated Systems*, pp. 549-554 (1994)
- [10] E. Boemo, G. Gonzalez de Rivera, S. Lopez-Buedo and J. Meneses: Some Notes on Power Management on FPGAs. In: *Field-Programmable Logic and Applications FPL'05*, LNCS, vol. 975, pp.149-157, Springer-Verlag (1995)
- [11] E. Boemo, S. Lopez-Buedo, G. González de Rivera, and J. Meneses: On the Usefulness of Pipelining and Wave Pipelining as Low-Power Design Technique. In: *Int. Workshop Power and Timing Modelling for Performance of Integrated Circuits*, pp.252-263 (1995).
- [12] E. Mussol and J. Cortadella: Low-Power Array Multiplier with Transition-Retaining Barriers. In: *Int. Workshop Power and Timing Modelling for Performance of Integrated Circuits*, pp.252-263 (1995).
- [13] E. Boemo, S. Lopez-Buedo, C. Santos, J. Jauregui and J. Meneses: Logic Depth and Power Consumption: A Comparative Study between Standard Cells and FPGAs. In: *Design of Circuit and Integrated Systems Conference* (1998)
- [14] Actel Corp.: Pipelining and Re-timing techniques and their effect on Power Dissipation (2000)
- [15] G. Sutter, E. Todorovich, S. López-Buedo, and E. Boemo: Logic Depth, Power, and Pipeline Granularity: Updated Results on XC4K and Virtex FPGAs. In: *Computación Reconfigurable & FPGAs (JCRA Workshop)*, pp.201-207 (2003)
- [16] S. Wilton, S. Ang, and W. Luk: The Impact of Pipelining on Energy per Operation in Field-Programmable Gate Arrays. In: *Field-Programmable Logic and Application*, LNCS, vol. 3203, pp. 719–728, Springer-Verlag (2004)
- [17] M. Khan: Power Optimization in FPGA Designs. Altera Corp. (2006)
- [18] N. Rollins: *Reducing Power in FPGA Designs Through Glitch Reduction*. Master of Science Thesis, Dep. of Electrical and Computer Engineering, Brigham Young University (2007)
- [19] S. Bard and N. Rafla: Reducing Power Consumption in FPGAs by Pipelining. In: *Midwest Symposium on Circuits and Systems*, IEEE Press (2008)
- [20] J. Ramos-Meixedo: *Metodologias de projecto de baixo consumo para implementações em FPGA*. Master Thesis, Universidade do Porto (2008).
- [21] M. Klein: Optimizing Xilinx FPGAs for Power. *Xcell Journal*, Q1 (2009).
- [22] J.P. Oliver and E. Boemo: Power Estimations vs. Power Measurements in Cyclone III Devices. In *Southern Conference on Programmable Logic*, pp.87-90, IEEE Press (2011)