Clock Gating Based on Auto-Gated Flip-Flops

V Muralidharan | M Vignesh | M Varatharaj

[1](Assistant professor, ECE PG department, Christ the King Engineering College, India, muralivlsi5@gmail.com)
[2](ECE PG department, Christ the King Engineering College, Coimbatore, INDIA,vignesh2k9@gmail.com)
[3](Head of the Department of ECE, Christ the King Engineering College, India, varatharaj_ms80@rediffmail.com)

Abstract— *Clock gating is very useful for reducing the power consumed by digital systems. Three gating methods are known. The most popular is synthesis-based, deriving clock enabling signals based on the logic of the underlying system. It unfortunately leaves the majority of the clock pulses driving the flip-flops (FFs) redundant. A data-driven method stops most of those and yields higher power savings, but its implementation is complex and application dependent. A third method called auto-gated FFs (AGFF) is simple but yields relatively small power savings. This paper presents a novel method called Look-Ahead Clock Gating (LACG), which combines all the three. LACG computes the clock enabling signals of each FF one cycle ahead of time, based on the present cycle data of those FFs on which it depends. It avoids the tight timing constraints of AGFF and data-driven by allotting a full clock cycle for the computation of the enabling signals and their propagation. A closed-form model characterizing the power saving per FF is presented. It is based on data-to-clock toggling probabilities, capacitance parameters and FFs' fan-in. The model implies a breakeven curve, dividing the FFs space into two regions of positive and negative gating return on investment. While the majority of the FFs fall in the positive region and hence should be gated, those falling in the negative region should not. Experimentation on industry-scale data showed 22.6% reduction of the clock power, translated to 12.5% power reduction of the entire system.*

Keywords—Clock gating, clock networks, dynamic power reduction.

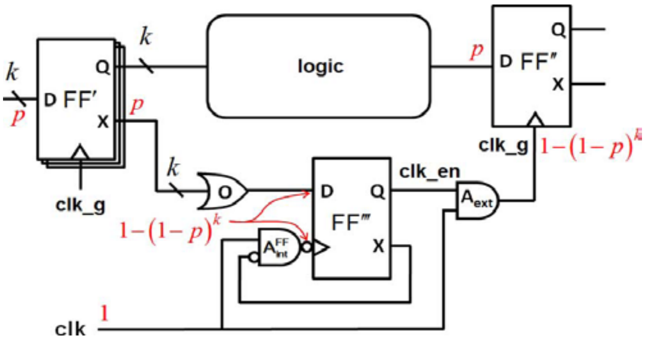
1. Introduction

One of the major dynamic power consumers in computing and consumer electronics products is the system's clock signal, typically responsible for 30% to 70% of the total dynamic (switching) power consumption . Several techniques to reduce the dynamic power have been developed, of which clock gating is predominant. Ordinarily, when a logic unit is clocked, its underlying sequential elements receive the clock signal regardless of whether or not their data will toggle in the next cycle.With clock gating, the clock signals are ANDed with explicitly predefined enabling signals. Clock gating is employed at all levels: system architecture, block design, logic design and gates . Several methods to take advantage of this technique are described in, with all of them relying on various heuristics in an attempt to increase clock gating opportunities. We call the above methods *synthesis-based*. Synthesis-based clock gating is the most widely used method by EDA tools . The utilization of the clock pulses, measured by data-to-clock toggling ratio, left after the employment of synthesis- based gating may still be very low.it depicts the average data-to-clock toggling ratio, obtained by extensive power simulations of 61 blocks comprising 200 k FFs, taken from a 32 nm high-end 64-bit microprocessor. Those are mostly control blocks of the data-path, register-file and memory management units of the processor. The technology parameters used throughout the papers are of 22 nm low-leakage process technology.

2. Auto-Gated Flip-Flops

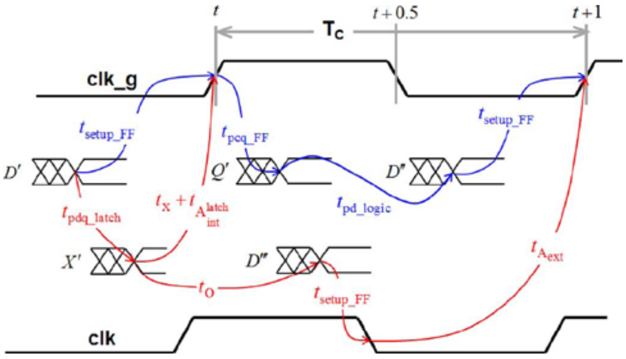
A. Introduction.

The FF's master latch becomes transparent on the falling edge of the clock, where its output must stabilize no later than a setup time prior to the arrival of the clock's rising edge, when the master latch becomes opaque and the XOR gate indicates whether or not the slave latch should change its state. If it does not, its clock pulse is stopped and otherwise it is passed. In a significant power reduction was reported for register-based small circuits, such as counters, where the input of each FF depends on the output of its predecessor in the register. AGFF can also be used for general logic, but with two major drawbacks. Firstly, only the slave latches are gated, leaving half of the clock load not gated. Secondly, serious timing constraints are imposed on those FFs residing on critical paths, which avoid their gating.



LACG takes AGFF a leap forward, addressing three goals; stopping the clock pulse also in the master latch, making it applicable for large and general designs and avoiding the tight timing constraints. LACG is based on using the XOR output in to generate clock enabling signals of other FFs in the system, whose data depend on that FF.

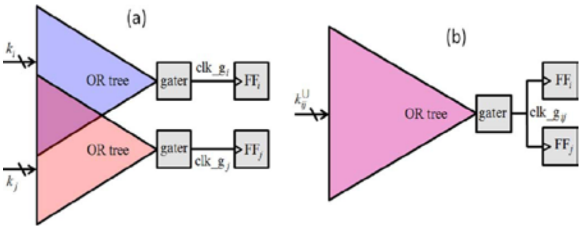
B. *Timing sequencing of LACG clock gating.*



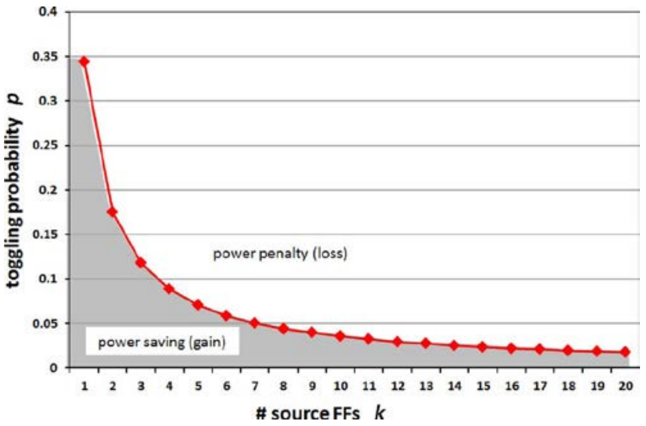
Using a FF for gating is a considerable overhead that will consume power of its own. This can significantly be reduced by gating FF'' Notice that since is oppositely clocked and its data is sampled at the clock's falling edge, its clock enabling signal must be negated. Also, is an ordinary FF where the internal XOR gate is connected between and D" and Q" .

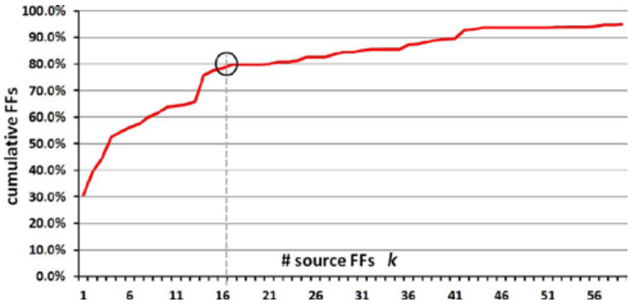
3. Modeling the Power Savings

The dynamic power overhead of LACG has been considered in the above breakeven analysis. There is also static power overhead. It should be noted that due to the full cycle allotted for the derivation of the enabling signals, the logic involved uses high threshold voltage and smallest devices. Moreover, as shown in the next section, the gating logic can be shared among several target FFs, which further reduces the overhead. We decided on LACG for a FF if it falls some safeguard margin apart the curve to compensate for leakage overhead. The detailed discussion of the design methodology is beyond the scope of this work.



It is not difficult to verify from that is decreasing with the increase of and . Clearly, large values of those may result in power loss rather than savings. We subsequently characterize the breakeven point. Substitute in implies a dependency between and , where the values of the various capacitances are known from the characterization of the cell library in use and by estimating the interconnecting wires. The dependency is shown by the Shmoo plot in Fig. 8 for the parameters in Table I, taken from a 22 nm process technology cell library.

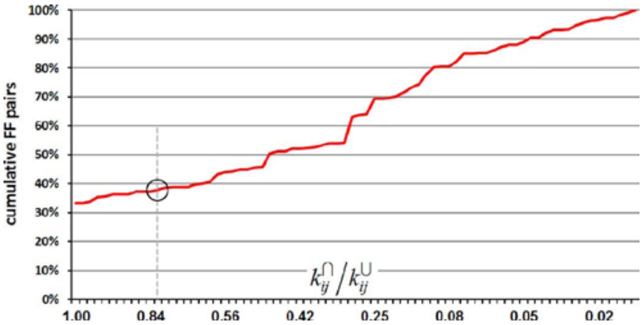




A. *Minimizing the gating logic*

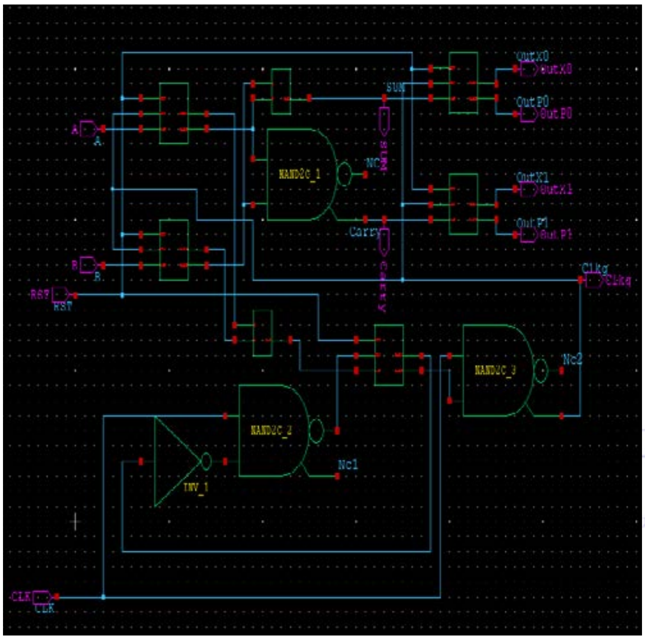
The savings expression in (7) assumed a separate gating logic for each target FF. This consumes a considerable power and area, and the gating logic should therefore be minimized. There are many cases where few target FFs depend on similar source FFs. In such cases there is no point in generating separate clock gating signals.

We subsequently develop logic sharing model to minimize the gating cost. showing two target FFs, and ,with their corresponding OR trees, driven by and source FFs, respectively. and have common source FFs, shown pictorially by the overlap of the trees. A different implementation were the OR logic is merged and a single gater is used for the two FFs. The larger the overlap is, the more desirable is the merge. In addition to logic reduction, the number of clock drivers and gaters will also be reduced.

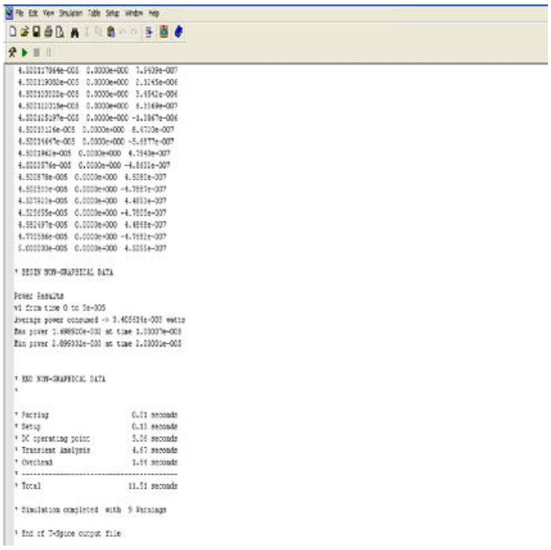


4. Experimental Results

A. LAGC Logic.

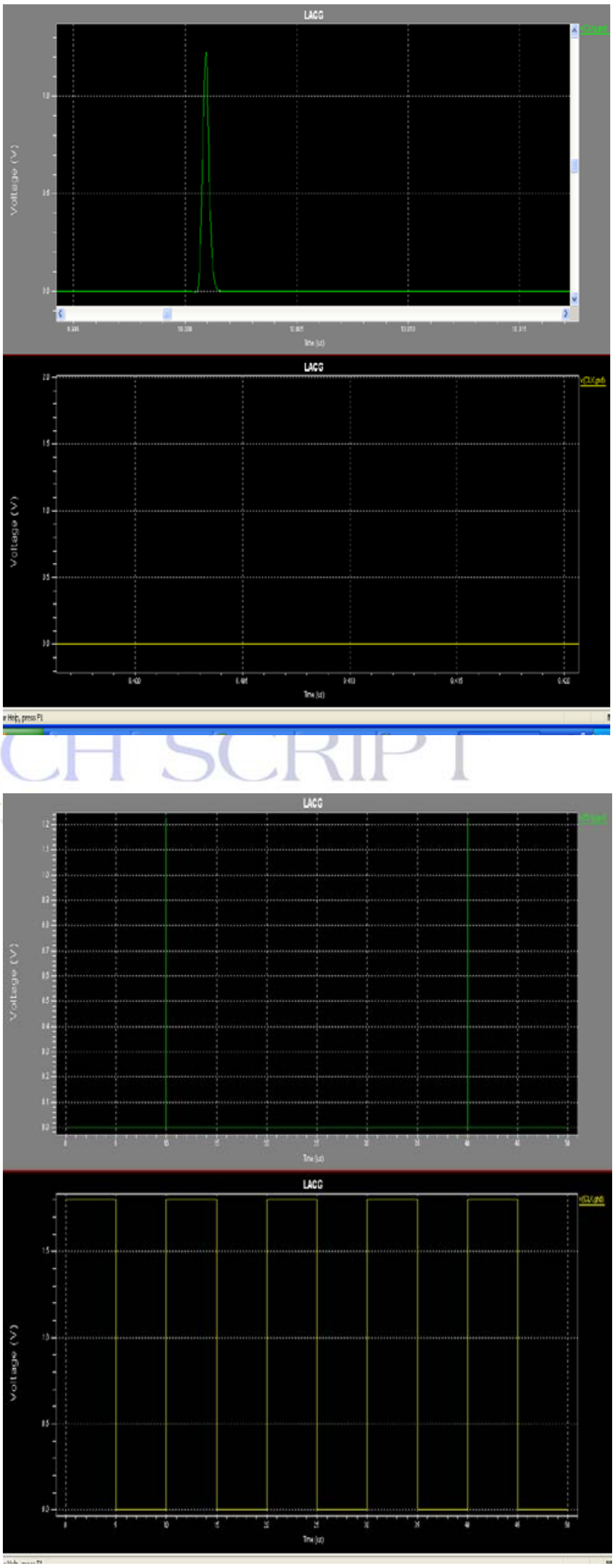


B. Avearage Power Cosumption.



```
 4.500117064e-005  0.0000e+000  7.5409e-007
 4.500119082e-005  0.0000e+000  2.3145e-006
 4.500120322e-005  0.0000e+000  3.4542e-006
 4.500122015e-005  0.0000e+000  6.3169e-007
 4.500125197e-005  0.0000e+000 -1.2067e-006
 4.50013126e-005  0.0000e+000  8.4720e-007
 4.50014647e-005  0.0000e+000 -5.6877e-007
 4.5001962e-005  0.0000e+000  4.7540e-007
 4.5003576e-005  0.0000e+000 -4.8631e-007
 4.500878e-005  0.0000e+000  4.5080e-007
 4.502533e-005  0.0000e+000 -4.7587e-007
 4.507923e-005  0.0000e+000  4.4853e-007
 4.525655e-005  0.0000e+000 -4.7805e-007
 4.582497e-005  0.0000e+000  4.4858e-007
 4.772586e-005  0.0000e+000 -4.7582e-007
 5.000000e-005  0.0000e+000  4.5055e-007

* BEGIN NON-GRAPHICAL DATA

Power Results
v1 from time 0 to 5e-005
Average power consumed -> 3.405634e-003 watts
Max power 1.698920e-002 at time 1.00007e-005
Min power 2.099302e-003 at time 2.00001e-005


* END NON-GRAPHICAL DATA
*

* Parsing                         0.03 seconds
* Setup                           0.33 seconds
* DC operating point              5.06 seconds
* Transient Analysis              4.67 seconds
* Overhead                        1.44 seconds
* -----------------------------------------
* Total                          11.53 seconds

* Simulation completed  with  9 Warnings

* End of T-Spice output file
```

C. Output Wave forms.



5. CONCLUSION

Look-ahead clock gating has been shown to be very useful in reducing the clock switching power. The computation of the clock enabling signals one cycle ahead of time avoids the tight timing constraints existing in other gating methods. A closed form model characterizing the power saving was presented and used in the implementation of the gating logic. The gating logic can be further optimized by matching target FFs for joint gating which may significantly reduce the hardware overheads. While this paper discussed the case of merging two target FFs for joint gating, clustering target FFs in larger groups may yield higher power savings.

REFERENCES

[1] V. G. Oklobdzija, *Digital System Clocking – High-Performance and Low-Power Aspects*. New York, NY, USA: Wiley, 2003.J.

[2] M. S. Hosny and W. Yuejian, "Low power clocking strategies in deep submicron technologies," in Proc. IEEE Int. Conf. Integr. Circuit Design

[3] C. Chunhong, K. Changjun, and S. Majid, "Activity-sensitive clock tree construction for low power," in *Proc. ISLPED*, 2002, pp. 279–282.