

## **Design and Simulation of Data-Driven Clock Gating Technique for Sensor Network**



**Pydipeddigari Ganesh**  
M.Tech, (Vlsi Design),  
Dept Of ECE,  
SVCET, Chittoor.



**L. Narayana Rao**  
Assistant professor,  
Dept Of ECE,  
SVCET, Chittoor.

### **Abstract:**

Clock gating is very useful for reducing the power consumed by digital systems. Three gating methods are known. The most popular is data driven-based, latch based and And gate based. It unfortunately leaves the majority of the clock pulses driving the flipflops (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 and-gate based is simple but yields relatively small power savings. This paper presents a novel method called data driven-based, which combines all the three. Latch based FFs 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 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. Simulation process is using modelsim 6.5e and synthesis process is using Xilinx 12.1i. Experimentation on industry-scale data showed 22.6% reduction of the clock power, translated to 12.5% power reduction of the entire system.

### **Key words:**

Clock gating, clock networks, dynamic power reduction.

### **I.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, with all of them relying on various heuristics in an attempt to increase clock gating opportunities. We call the above methods data driven 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. Fig. 1 depicts the average data-to-clock toggling ratio, obtained by extensive power simulations of 61 blocks comprising 200k FFs, taken from a 32 nm high-end 64-bit microprocessor. Those are mostly control blocks of the datapath, register file and memory management units of the processor.

The technology parameters used throughout the papers are of 22 nm low-leakage process technology. Their clock enabling signals were derived by a mix of logic synthesis and manual definitions. The clock capacitive load is 70% of their total load. The blocks are increasingly ordered by their data-to-clock activity ratio. It is clearly shown that the data toggles in a very low rate compared to the gated clocks. Point (a) shows that in 87% of the blocks (53/61) the data toggles less than 6% compared to the gated clock, where the average shown by the horizontal dashed line is 3%. Also plots the corresponding cumulative clock capacitive load.

Point (b) shows that the above 87% blocks are responsible for 95% of the total clock load. Consequently, the switching of a significant portion of the system's clock load is redundant, but consumes most of its power. This calls for other than synthesis-based methods to address the above redundancy, a method called data-driven clock gating was proposed for flip-flops (FFs). There, the clock signal driving a FF, is disabled (gated) when the FF's state is not subject to change in the next clock cycle [9]. In an attempt to reduce the overhead of the gating logic, several FFs are driven by the same clock signal, generated by ORing the enabling signals of the individual FFs [8]. Based on the data-to-clock toggling probability, a model to derive the group size maximizing the power savings was developed.

A comparison between the synthesis-based and data-driven gating methods showed that the latter outperforms for control and arithmetic circuits, while the former outperforms for register-file based circuits. Data-driven gating is illustrated in Fig. 2. A FF finds out that its clock can be disabled in the next cycle by XORing its output with the present input data that will appear at its output in the next cycle. The outputs of XOR gates are ORed to generate a joint gating signal for FFs, which is then latched to avoid glitches. The combination of a latch with AND gate is used by commercial tools and is called Integrated Clock Gate (ICG). It is beneficial to group FFs whose switching activities are highly correlated. The work in [10] addressed the questions of which FFs should be placed in a group to maximize the power reduction, and how to find those groups. Data-driven gating suffers from a very short time-window where the gating circuitry can properly work.

This is illustrated in Fig. 3. The cumulative delay of the XOR, OR, latch and the AND gater must not exceed the setup time of the FF. Such constraints may exclude 5%-10% of the FFs from being gated due to their presence on timing critical paths [10]. The exclusion percentage increases with the increase of critical paths, a situation occurring by downsizing or turning transistors of non-critical path to high threshold voltage (HVT) for further power savings.

## II. DATA-DRIVEN CLOCK GATING:

Clock enabling signals are very well understood at the system level and thus can effectively be defined and capture the periods where functional blocks and modules do not need to be clocked. Those are later being automatically synthesized into clock enabling signals at the gate level. In many cases, clock enabling signals are manually added for every FF as a part of a design methodology. Still, when modules at a high and gate level are clocked, the state transitions of their underlying FFs depend on the data being processed. It is important to note that the entire dynamic power consumed by a system stems from the periods where modules' clock signals are enabled.

Therefore, regardless of how relatively small this period is, assessing the effectiveness of clock gating requires extensive simulations and statistical analysis of FFs toggling activity, as presented subsequently. Fig. 1 shows the FFs' toggling activity in an arithmetic block comprising 22K FFs, designed in 40-nm technology, taken from Ceva's X1643 DSP core for multimedia and wireless baseband applications [21]. The statistics is obtained from extensive simulations of typical modes of operation, consisting of 240-K clock cycles.

The average time window when the FFs clock signal is enabled is only 10%, which is still responsible for the entire dynamic power consumed by that block. The clock enabling signals are obtained by RTL synthesis and manual insertions. As Fig. 1 shows, a FF toggled its state only 2.9% of the clock enabled time period, on the average, thus more than 97% of the clock pulses driving FFs are useless. Such a low toggling rate (of non-clock signals) is very common [12]. Another example of a 40-nm control block comprising 37-K FFs (part of Mellanox ConnectX network processor [23]) has also been examined.

There, the clock signal is enabled 20% of the time and within that window the average FF toggling is only 1.3%, and here too more than 98% of the clock pulses driving FFs are useless. It follows from the above examples that no matter what RTL and gate levels clock enabling signals are followed, there are still many opportunities to gate the clock signal at the FF level.

The data-driven gating proposed in [9] is illustrated in Fig. 2. A FF finds out that its clock can be disabled in the next cycle by XORing its output with the present data input that will appear at its output in the next cycle. The outputs of K XOR gates are ORed to generate a joint gating signal for k FFs, which is then latched to avoid glitches. The combination of a latch with AND gate is commonly used by commercial tools and is called integrated clock gate (ICG) [13]. Such data-driven gating is used for a digital filter in an ultralow-power design [24]. A single ICG is amortized over k FFs. There is

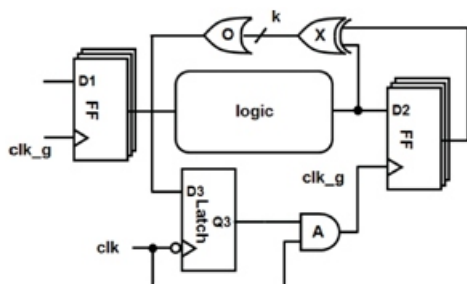


Fig: Data driven based clock gating technique

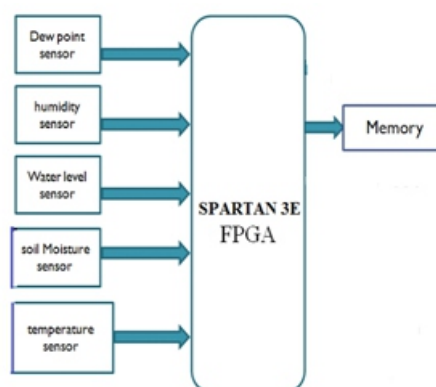
a clear tradeoff between the number of saved (disabled) clock pulses and the hardware overhead. With an increase in k, the hardware overhead decreases but so does the probability of disabling, obtained by ORing the k enable signals. Let the average toggling probability of a FF (also called activity factor) be denoted by p ( $0 < p < 1$ ). Under the worstcase assumption of independent FF toggling, and assuming a uniform physical clock tree structure, it is shown in [9] that the number k of jointly gated FFs for which the power savings are maximized is the solution of

$$(1 - p)^k \ln(1 - p)(c_{FF} + c_w) + c_{latch}/k^2 = 0 \quad (1)$$

where  $c_{FF}$  is the FFs clock input capacitance,  $c_w$  is the unit-size wire capacitance, and  $c_{latch}$  is the latch capacitance including the wire capacitance of its clk input. Table I shows how the optimal k depends on p. Such a gating scheme has considerable timing implications, which are discussed in [9].

We will return to those when discussing the implementation of data-driven gating as a part of a complete design flow. Latch For the scheme proposed in Fig. 2 to be beneficial, the clock enabling signals of the grouped FFs should preferably be highly correlated. Data-driven clock gating is shown to achieve savings of more than 10% of the total dynamic power consumed by the clock tree [15]. Reference [24] reported 20% power savings. It took advantage of the very low dynamic range of the data in a digital filter. The gating logic is tailored to the structure of the filter, whereas the approach discussed in this paper is more general and applies to large scale and a wide range of designs. The experiments described in Section VI.

### III. BLOCK DIAGRAM:



#### Temperature sensors:

Temperature sensors are devices used to measure the temperature of a medium. There are 2 kinds on temperature sensors: 1) contact sensors and 2) noncontact sensors. these sensors measure a physical property which changes as a function of temperature. In these project a digital temperature is being used. The digital temperature sensors features low power consumption, up to 12 bit resolution and can operate over a temperature range as wide as -55 to +125°C.

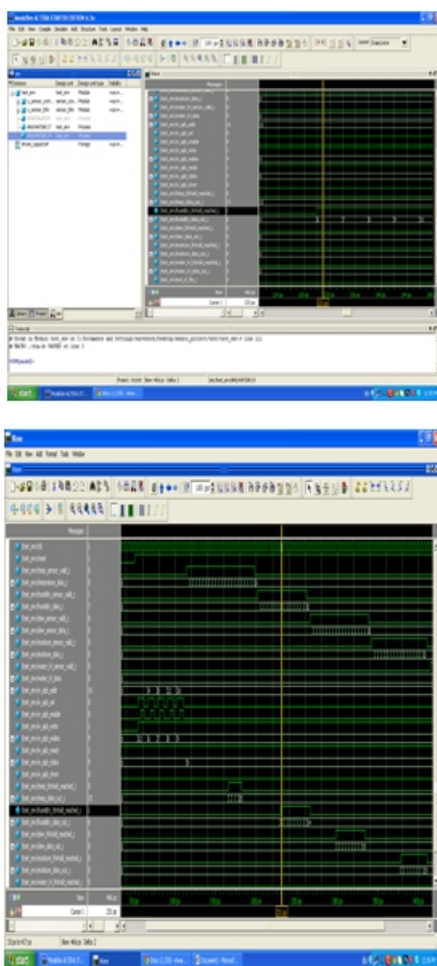
#### Humidity sensors:

A humidity sensor measures the relative humidity and expressed as a percent (RH %). It is the ratio of actual moisture in the air to the highest amount of moisture in air can hold at that temperature. The most common type of humidity sensor used is the “capacitive sensor.” This sensor is based on electrical capacitance . The sensor is composed of two metal plates with a

non-conductive polymer film between them. The film collects moisture from the air, and the moisture causes minute changes in the voltage between the two plates. The changes in voltage are used to know the amount of moisture in the air. Humidity sensors are of three types. Resistive, Capacitive, and Thermal Conductivity sensing. Resistive sensors are useful in remote locations. Capacitive sensors are useful for wide RH range and condensation tolerance. Thermal conductivity sensors are beneficial in corrosive environments that have high temperatures.

## IV. SIMULATION RESULTS:

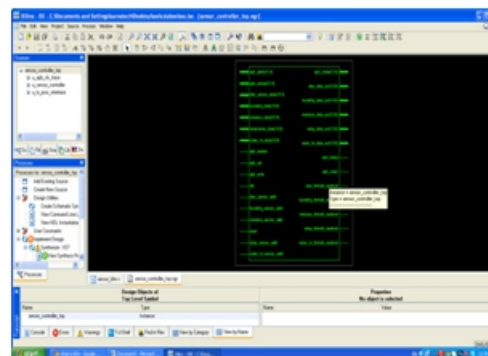
Simulation Results using Modelsim tool



## V. SYNTHESIS RESULTS:

Synthesis Results using Xilinx

## Block Diagram:



## VI. 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. This is a matter of a further research.

## REFERENCES:

- [1] V. G. Oklobdzija, Digital System Clocking – High-Performance and Low-Power Aspects. New York, NY, USA: Wiley, 2003.
- [2] L. Benini, A. Bogliolo, and G. De Micheli, “A survey on design techniques for system-level dynamic power management,” IEEE Trans. VLSI Syst., vol. 8, no. 3, pp. 299–316, Jun. 2000.
- [3] M. S. Hosny and W. Yuejian, “Low power clocking strategies in deep submicron technologies,” in Proc. IEEE Int. Conf. Integr. Circuit Design Technol., ICICDT 2008, pp. 143–146.
- [4] C. Chunhong, K. Changjun, and S. Majid, “Activity-sensitive clock tree construction for low power,” in Proc. ISLPED, 2002, pp. 279–282.

- [5] A. Farrahi, C. Chen, A. Srivastava, G. Tellez, and M. Sarrafzadeh, "Activity-driven clock design," *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.*, vol. 20, no. 6, pp. 705–714, Jun. 2001.
- [6] W. Shen, Y. Cai, X. Hong, and J. Hu, "Activity and register placement aware gated clock network design," in *Proc. ISPD*, 2008, pp. 182–189.
- [7] Synopsys Design Compiler, Version E-2010.12-SP2.
- [8] S. Wimer and I. Koren, "The Optimal fan-out of clock network for power minimization by adaptive gating," *IEEE Trans. VLSI Syst.*, vol. 20, no. 10, pp. 1772–1780, Oct. 2012.
- [9] M. Donno, E. Macii, and L. Mazzoni, "Power-aware clock tree planning," in *Proc. ISPD*, 2004, pp. 138–147.
- [10] S. Wimer and I. Koren, "Design flow for flip-flop grouping in data-driven clock gating," *IEEE Trans. VLSI Syst.*, to be published.
- [11] M. Muller, S. Simon, H. Gryska, A. Wortmann, and S. Buch, "Low power synthesizable register files for processor and IP cores," *INTEGRATION, The VLSI J.*, vol. 39, pp. 131–155, 2006.
- [12] A. G. M. Strollo and D. De Caro, "Low power flip-flop with clock gating on master and slave latches," *Electron. Lett.*, vol. 36, no. 4, pp. 294–295, Feb. 2000.
- [13] C. E. Stroud, R. R. Munoz, and D. A. Pierce, "Behavioral model synthesis with Cones," *IEEE Design Test Comput.*, vol. 5, no. 3, pp. 22–30, Jun. 1988.
- [14] J. A. Bondy and U. S. R. Murty, *Graph Theory*. : Springer, 2008.
- [15] V. Kolmogorov, "Blossom V: A new implementation of a minimum cost perfect matching algorithm," *Math. Prog. Comp.*, pp. 43–67, 2009.
- [16] J. Kathuria, M. Ayoub, M. Khan, and A. Noor, "A review of Clock Gating Techniques," *MIT Int. J. Electron. and Commun. Engin.*, vol.1, no. 2, pp. 106–114, Aug. 2011.
- [17] S. Wimer, "On optimal flip-flop grouping for VLSI power minimization," *Oper. Res. Lett.*, vol. 41, no. 5, pp. 486–489, Sep. 2013.
- [18] "A Comparison of Intel's 32 nm and 22 nm Core i5 CPUs: Power, Voltage, Temperature, and Frequency," Oct. 2012 [Online]. Available: <http://blog.stuffedcow.net/2012/10/intel32-nm-22-nm-core-i5-comparison/>