

Multi Bit Flip-Flop Design with probability driven Clock Gating to Optimize Power Consumption

Prof.Vinay.K.Kolur, (Smt).Savitri R Sanadi

Department of Electrical & Electronics, V.P.

Dr. P.G.Halakatti college of Engineering and Technology, Vijayapur-586103

ABSTRACT:

Multi Bit Flip-Flops (MBFFs) and Data-Driven Clock-Gating (DDCG) in which number of flip-flops are grouped and use a common clock driver for a group of flip-flop. By using MBFF and DDCG techniques more power is saved and these are the effective low power design techniques. This project work increases the power savings by introducing a combined algorithm of DDCG and MBFF, based on Data to clock-toggling ratio and Multi-Bit Flip-Flops. To optimize the power consumption, several FFs are grouped in MBFFs to increase their activities. The combined algorithm of DDCG and MBFF based on probability driven data-to clock-toggling ratio of Flip-Flops. 17% to 23% power saving is achieved using this technique. This technique uses Xilinx ISE tool.

Keywords: Multi Bit Flip-Flop, clock-gating, low power, data toggling, power reduction.

I. INTRODUCTION

In digital system a bit of information is stored in 1 bit flip-flops (FFs), each has individual clock driver shown in fig1 and it consists of two master slave latches and an edge triggered clock driven by opposite clocks. Individual flip-flops consumes some amount of energy due to its internal clock drivers, these are main reasons in contributing power consumption in the circuit. To decrease power consumption number of flip-flops are combined and apply a common clock driver circuit for all FFs. Fig 1 shows grouping of two 1-bit FFs to get 2-bit MBFF. Similarly for, 4-MBFF and 8-MBFFs are also formed. So we can represent a MBFF by combining k-FFs

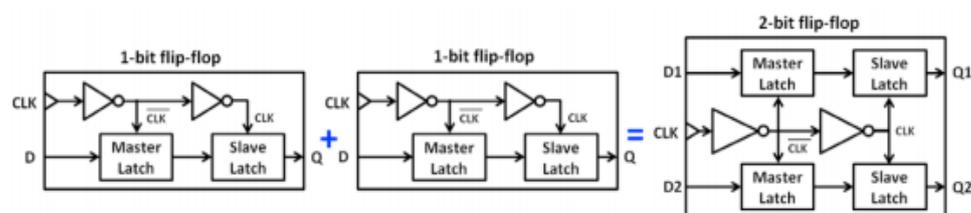


Fig1. 1-bit FF and 2-MBFF.

This paper discusses about the combining of flip-flops and front-end implementation of MBFF accumulation and measurement and effects of clock gating (CG)..

II. INTRODUCING CLOCK-GATING INTO MBFF

The Fig2 shows DDCG combined into k multi bit FFs. In the fig all the shaded blocks lie in a library cell. k group size, it increases power saving ability. Let us consider following equation.

$$(1-p)^k \ln(1-p) C_{FF} + \frac{C_{latch}}{k^2} = 0, \tag{1}$$

Where, C_{FF} is the clock input load of FF and C_{latch} is the clock input loads of latch. The solutions to equation (1) for number of activities are given in Table 3.1. For some values for C_{FF} and C_{latch} .

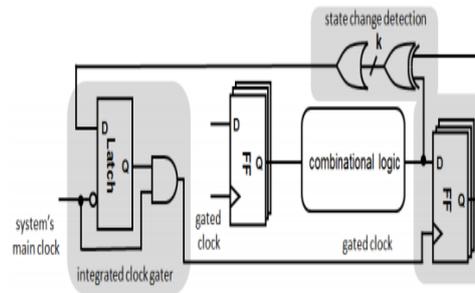


Fig2. DDCG combined into k -MBFF.

Table 1. MBFF multiplicity depends on toggling probability.

p	0.01	0.02	0.05	0.1
k	8	6	4	3

In this project work MBFFs discussed is DDCG. The optimal MBFF multiplicity dependency is shown in table1 on and DDCG combined into k MBFF as shown in fig 2 and simulated using spice simulator tool. Fig.3 indicates that power consumption 2bit flip flop versus 2 multi bit flip flop. There is no crossing point beyond 0.17 by using a 2-bit MBFF activity, at this point power starts lost.

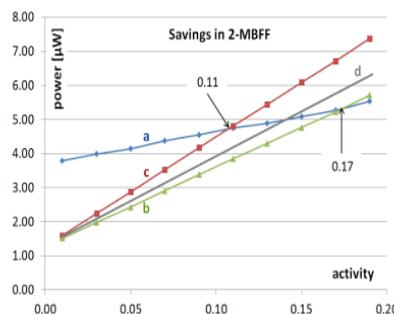


Fig 3. Power consumption of 2 –bit Flip Flops vs. 2- bit MBFF.

Line (c) indicates that instance place Flip Flops need toggling disjoint. From this it is clear that most exceedingly bad case since, the clock driver meets accepted values to those two FFs, same time special case necessities it. With respect to line(b), in the event for disjoint toggling there will be no perspective for utilizing 2-bit MBFF whether the Flip Flops exercises would higher over 0. 11. Provided for an activity, those energy funds for 2-MBFF will be the separation the middle of line (b) alternately (c) with (a). Notice that to zero action those per-bit power saving is $(3.8-1.8)/2=1.0 \mu W$

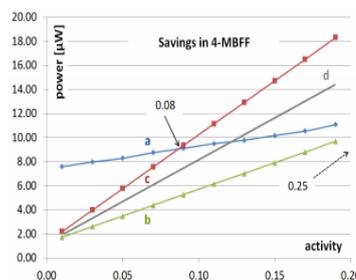


Fig 4. Power consumption of 4 –bit Flip Flops vs. 4-bit MBFF.

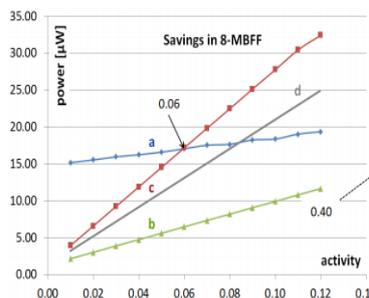


Fig 5. Power consumption of 8-bit FFs vs. 8-bit MBFF.

Fig4 indicates the power consumed by 4-bit MBFF, and Fig5 indicates power consumed by 8-bit flip flops. From these fig it is clear that the per bit power saving is $(15.3-2.5)/8=1.6 \mu W$.

III. FLIP-FLOPS ARE COMBINED IN A DDCG MBFF.

From the above discussions it clears that best grouping for FFs Might a chance to be attained for FFs whose toggling may be very nearly totally associated. That issue for Flip Flops combining yields maximum toggling correlation, and consequently maximal power savings, has been demonstrated as NP-hard. For large number of power simulations speaking to those common place of operation and application of the design. Such information might not exist in the early configuration phase. Additional regular majority of the data will be those Normal toggling greater part probabilities for each FF in the design; this That examination as such accepted that every last one of Flip-Flops combined in a Multi-Bit Flip-Flop bring same data toggling probability p.

Let us consider both FF_i and FF_j are toggling, when FF_i j is enabled then that time clock driver energy is fully utilized and there is no power waste. Power waste is happens when a Flip Flop is toggles. When

the clock pulse is enabled, then it drives both Flip-Flops, but one needs it. A waste W_{ij} half of the internal clock driver energy 2λ , and is given by

$$W_{(i,j)} = \frac{\lambda_2}{2} [p_j(1-p_i) + p_i(1-p_j)] = \frac{\lambda_2}{2} (p_i + p_j - 2p_i p_j). \quad (2)$$

2-bit MBFFs yields the energy waste

$$W_{(i,j)} + W_{(k,l)} = \frac{\lambda_2}{2} \left[\underbrace{p_i + p_j + p_k + p_l}_{(a)} - 2 \underbrace{(p_i p_j + p_k p_l)}_{(b)} \right]. \quad (3)$$

The following is the energy waste equation

$$W(\mathbf{P}) = \sum_{i=1}^{n/2} W_{(s_i, t_i)} = \frac{\lambda_2}{2} \left[\sum_{j=1}^n p_j - 2 \sum_{i=1}^{n/2} p_{s_i} p_{t_i} \right]. \quad (4)$$

The optimal pairing minimizing $W(\mathbf{P})$ is defined by the following theorem.

Theorem 1: Let n be even and let $FF_1, FF_2, FF_3, \dots, FF_n$ be ordered such that their toggling probabilities $P_1 \leq P_2 \leq \dots \leq P_n$

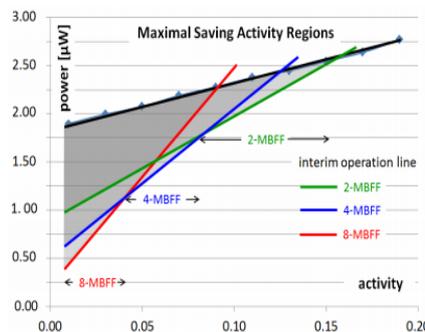


Fig 6. Dividing the activity into ranges of maximum power savings.

successive FFs is minimizing $W(\mathbf{P})$. 2-MBFFs summed upto grouping is generalized for k -MBFFs as follows.

Theorem 2: Let us consider n divisible by k , and gives FF_1, FF_2, \dots, FF_n be chance to be requested such-and-such their toggling probabilities satisfy P_1, P_2, \dots, P_n . The combining of successive Flip Flops reduces power waste incurred by the n/k -bit MBFFs. The above condition is for n divisible by k and n not divisible by k also addressed [8].

$$\mathbf{P} : \left\{ FF_{(k(i-1)+1, \dots, ki)} \right\}_{i=1}^{n/k} \quad (5)$$

Successive FFs is minimising the energy waste incurred by the n/k k -MBFF.

IV. CAPTURING EVERYTHING TOGETHER IN A DESIGN FLOW.

Now we have Figs.3, 4 and 5, are the power consumption in 2, 4 and 8-Multi-Bit Flip-Flops, respectively. The line (d) interim shows case of simultaneous and disjoint Flip-Flops toggling representation where, Flip-Flops toggles independently of each other. Knowing the activity of a Flip-Flop, the decision in what MB Flip-Flop size it should be grouped will follow the interim lines.

Fig. 6 represents how the range about Flip Flop action partitioned to obtain maximum power saving funds. The black line shows 1-bit Flip Flop un gated. Follows the power expended by an 1-bit ungated Flip-Flop. The triangular region covered between blue and red is per bit power utilization line.

In Flip-Flops have been combined sequentially as stated by their bit number in their register.. Table 3 reveals to every k –Multi-Bit Flip-Flop, 2, 4, 8 k those. By and large grouping toward monotonic movement may be favoured (colored for green), however previously, few situations it worsened. That might happen since combining of Flip-Flops is blind with toggling relationship.

Table 2. Average FF activity of pipeline registers in 32-bit MIPS.

pipeline stage activity for array sorting program												
grouping method	IF / ID			ID / EXE			EXE / MEM			MEM / WB		
	0.105			0.0856			0.0711			0.0473		
	2-MBFF	4-MBFF	8-MBFF	2-MBFF	4-MBFF	8-MBFF	2-MBFF	4-MBFF	8-MBFF	2-MBFF	4-MBFF	8-MBFF
by index	0.174	0.261	0.353	0.140	0.204	0.275	0.109	0.163	0.214	0.0793	0.117	0.173
by activity	0.169	0.261	0.353	0.134	0.189	0.231	0.116	0.155	0.190	0.0761	0.104	0.131
improve [%]	+2.9	0	0	+4.3	+7.4	+16.0	-6.4	+4.9	+11.2	+4.0	+11.1	+24.3
pipeline stage activity for array matrix multiplication												
grouping method	IF / ID			ID / EXE			EXE / MEM			MEM / WB		
	0.127			0.118			0.0799			0.0582		
	2-MBFF	4-MBFF	8-MBFF	2-MBFF	4-MBFF	8-MBFF	2-MBFF	4-MBFF	8-MBFF	2-MBFF	4-MBFF	8-MBFF
by index	0.203	0.311	0.422	0.169	0.241	0.300	0.128	0.180	0.262	0.0940	0.145	0.208
by activity	0.198	0.294	0.388	0.174	0.246	0.322	0.128	0.174	0.222	0.0938	0.127	0.162
improve [%]	+2.5	+5.5	+8.1	-3.0	-2.1	-7.3	0	+3.3	+15.3	+0.2	12.4	+22.1

Table 3 Power savings in the pipeline registers of a 32-bit MIPS.

	IF/ID	ID/EXE	EXE/MEM	MEM/WB	total
power [μW]	980	1056	952	916	3904
savings [μW]	284.4	344.0	332.0	388.4	1348
savings [%]	29.1	32.6	34.8	42.4	34.6

By using MBFFs monotonic order of their activities the pipeline registers are implemented. The result was measured with SpyGlass simulation. In this 34.6% power saving is achieved and The MIPS were operated in

1.1V and 200MHz. So the total power reduction was 23%. Table 2 shows power saving in the pipeline registers of a 32 bit-MIPS.

Table 4 shows to 8% net power savings the power measurements in this consists of both static and dynamic power components and also includes clock gating overheads. we can increase 8% power saving into 17% by changing from 1 bit flip-flops to ungated MBFF. By introducing clock gating the area penalty was 2.3%.The following table 4 shows power saving in 40nm network processor.

Table 4. Power savings in 40nm network processor.

unit	FF CLK power [mW]	total CLK power [mW]	total unit power [mW]	FF CLK power save [mW]	FF CLK power save [%]	total CLK power save [%]	total unit power save [%]	area penalty [%]
A	80	1112	1,802	44	57.6	4.09	2.52	1.7
B	304	316	1,638	104	33.4	32.5	6.22	2.8
C	184	268	760	76	41.9	28.6	10.1	2.7
D	72	172	294	32	45.2	19.2	11.2	2.3
E	162	368	884	88	53.4	23.8	9.90	4.3
F	112	204	252	80	69.7	38.2	31.0	1.3
G	124	368	556	72	57.4	19.7	13.0	1.9
total	1,040	2,804	6,186	496	47.5	17.7	8.00	2.3

The latch and gate (AND gate) overheads are amortized over k Flip Flops. Let us consider activity factor p and also called as toggling probability of a Flip-Flop, p is $0 < p < 1$. Let us assume Flip-Flop is independent of toggling and uniform physical clock tree structure k is jointly gated Flip-Flops for which the power saving are increased is the solution of

$$(1-p)^k \ln(1-p) C_{FF} + \frac{C_{latch}}{k^2} = 0, \quad (6)$$

Where C_{FF} is FFs clock input capacitance, C_w will be the unit-size wire capacitance, and C_{latch} may be that clock capacitance including that wire capacitance of its clock input. Table 1 demonstrates how the ideal k depends on p. We will returns on the individuals when examining the execution from claiming data-driven gating as a part of complete design flow.

V. IMPLEMENTATION AND INTEGRATION IN A DESIGN FLOW.

Backend design flow of Data-driven clock gating implementation has the following design step procedure.

- 1) Find out Flip-Flops toggling probabilities includes running an far reaching test bench representing to typical operation modes of the design with figure out the size k of a gated Flip-Flop group by solving equation number (1).
- 2) In order to get prepared locations of FFs in the layout, run the placement tool.
- 3) Using a Flip-Flops grouping tool the proposed model and algorithms presented in this work is implemented, utilizing data toggling information got from step 1 and Flip-flops locations' information got in step 2. That

Conclusion from claiming this step may be k-size Flip-Flop sets, to place the FFs in every set will make clocked by a common gate.

4) By using hardware description language like Verilog HDL to present the data-driven clock gating logic into the hardware description. This is completed naturally by a programming tool; including suitable Verilog code on actualize all the logic shown in fig. 2. Those FFs need aid associated as stated by those grouping acquired to step 3. Practically addressing presents the gating logic into Register Transfer Logic (gate-level description). This has been depends on configuration technique being used and its discussion is past the growth for this work. Finally we need to put gating logic into RTL depiction.

5) Once again run the test bench of step 1 on confirms that full identity of Flip-Flops' outputs previously, then following the introduction of gating logic. Despite data driven clock gating, Eventually should not change its exact definition, ought to not progress those rationale of signals, and Subsequently FFs toggling ought further bolstering remain identical, An strong configuration stream must execute in this step.

6) Common backend design flow completion proceeds by applying place and route tool backend design flow executed.

VI. RESULTS



Fig 7. The output waveform for data driven clock gating

The result is obtained by using xilinx software tool. when reset is equal to 1 no output when reset equal to 0 and clock signal is also enable at that time Flip-Flops are enabled and we are getting output. in this work to show power reduction we have used D-FF hence what is input 32 bit data we are going to input that same will be output we are getting.

VII. CONCLUSION

Clock gating is used in first in first out concept to optimize the energy utilization. For further energy saving DDCG and MBFF are used in sequential circuits. In order to optimize power consumption Clock gating is used. But it leaves more amounts of redundant bits of clock pulses. In order to reduce redundant bits DDCG is used. Multi-bit flip-flop is used to optimize power consumption by eliminating redundant inverters in the

circuit. Combination of Multi-bit Flip-Flop with Data driven clock gating will increase the further power optimization. Xilinx software tool is used for implementing this proposed system. The combination of DDCG with Multi-Bit Flip-Flop is used to reduce power consumption.

REFERENCES

- [1]. Kapoor, Ajay, Cas Groot, Gerard Villar Pique, Hamed Fatemi, Juan Echeverri, Leo Sevat, Maarten Vertregt et al. "Digital systems power management for high performance mixed signal platforms." *Circuits and Systems I: Regular Papers, IEEE Transactions on* 61, no. 4 (2014): 961-975.
- [2]. Wimer, Shmuel, and Israel Koren. "The optimal fan-out of clock network for power minimization by adaptive gating." *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on* 20, no. 10 (2012): 1772-1780.
- [3]. Santos, Cristiano, Ricardo Reis, Guilherme Godoi, Marcos Barros, and Fabio Duarte. "Multi-bit flip-flop usage impact on physical synthesis." In *Integrated Circuits and Systems Design (SBCCI), 2012 25th Symposium on*, pp. 1-6. IEEE, 2012.
- [4]. Yan, Jin-Tai, and Zhi-Wei Chen. "Construction of constrained multi-bit flip-flops for clock power reduction." In *Green Circuits and Systems (ICGCS), 2010 International Conference on*, pp. 675-678. IEEE, 2010. 15
- [5]. Jiang, IH-R., Chih-Long Chang, and Yu-Ming Yang. "INTEGRA: Fast multibit flip-flop clustering for clock power saving." *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on* 31, no. 2 (2012): 192-204.