



# International Journal for Innovative Engineering and Management Research

A Peer Reviewed Open Access International Journal

www.ijiemr.org

**COPY RIGHT**



**ELSEVIER**  
**SSRN**

**2019IJIEMR**. Personal use of this material is permitted. Permission from IJIEMR must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. No Reprint should be done to this paper, all copy right is authenticated to Paper Authors

IJIEMR Transactions, online available on 23<sup>rd</sup> Nov 2019. Link

[:http://www.ijiemr.org/downloads.php?vol=Volume-08&issue=ISSUE-11](http://www.ijiemr.org/downloads.php?vol=Volume-08&issue=ISSUE-11)

Title **EFFICIENT AND CONDITIONAL ALGORITHMS FOR REGISTER MODELING IN VERILOG**

Volume 08, Issue 11, Pages: 220–227.

Paper Authors

**CHOUDALLA SAILA RAM , K.PRASANNA**

KAKINADA INSTITUTE OF ENGINEERING AND TECHNOLOGY FOR  
WOMEN, KORANGI, ANDHRAPRADESH, INDIA, 533461



USE THIS BARCODE TO ACCESS YOUR ONLINE PAPER

To Secure Your Paper As Per **UGC Guidelines** We Are Providing A Electronic Bar Code

## EFFICIENT AND CONDITIONAL ALGORITHMS FOR REGISTER MODELING IN VERILOG

<sup>1</sup>CHOUDALLA SAILA RAM , <sup>2</sup>K.PRASANNA

<sup>1</sup>M.TECH VLES, DEPT OF E.C.E, KAKINADA INSTITUTE OF ENGINEERING AND TECHNOLOGY FOR WOMEN,KORANGI,ANDHRAPRADESH,INDIA,533461

<sup>2</sup>ASSISTANT PROFESSOR, KAKINADA INSTITUTE OF ENGINEERING AND TECHNOLOGY FOR WOMEN,KORANGI,ANDHRAPRADESH,INDIA,533461

### ABSTRACT:

Data-driven clock gated (DDCG) and multi bit flip-flops (MBFFs) are two low-power design techniques that are usually treated separately. Combining these techniques into a single grouping algorithm and design flow enables further power savings. We study MBFF multiplicity and its synergy with FF data-to-clock toggling probabilities. A probabilistic model is implemented to maximize the expected energy savings by grouping FFs in increasing order of their data-to-clock toggling probabilities. We present a front-end design flow, guided by physical layout considerations for a 65-nm 32-bit MIPS and a 28-nm industrial network processor. It is shown to achieve the power savings of 23% and 17%, respectively, compared with designs with ordinary FFs. About half of the savings was due to integrating the DDCG into the MBFFs. The proposed architecture of this paper analysis the logic size, area and power consumption using Tanner tool.

### INTRODUCTION:

recently published paper has emphasized the usage of Multi-Bit Flip-Flops (MBFFs) as a design technique delivering considerable power reduction of digital systems [1]. The data of digital systems is usually stored in Flip-Flops (FFs), each having its own internal clock driver. Shown in Fig. 1a, an edge-triggered 1-bit FF contains two cascaded master and slave latches, driven by opposite clocks CLK and CLK. It is shown in that most of the FF's energy is consumed by its internal clock drivers, which are significant contributors to the total power consumption. In an attempt to reduce the clock power, several FFs can be grouped in a module such that common clock drivers

are shared for all the FFs. Two 1-bit FFs grouped into 2-bit MBFF, called also dual-bit FF [1], is shown in Fig. 1a. In a similar manner, grouping of FFs in 4-bit and 8-bit MBFFs are possible too. We subsequently denote a k-bit MBFF by k-MBFF. MBFF is not only reducing the gate capacitance driven by a clock tree. The wiring capacitive load is also reduced because only a single clock wire is required for multiple FFs. It also reduces the depth and the buffer sizes of the clock tree and also the number of sub-trees. Beyond clock power savings those features also reduce the silicon area. Most distributed deals with MBFF have concentrated on physical

execution, driven basically by the postplacement format [4], [5], [7], [8], [13], [16]. In these works, FF exercises have a tendency to be overlooked. Each FF is related with time edges got from the design including 1-bit FFs. The wires associated with the information and yield of a FF are secured on their contrary side to whatever is left of the rationale, though the position of the FF is permitted to move around without abusing timing. This characterizes the district in the design where the FF can be dislodged and converged into the MBFF. The 2-MBFF combining is defined as an advancement issue that goes for boosting the quantity of blended FFs. Other works [9]–[11] have introduced clock-tree layout considerations as well. To further save power, [6] introduced CG, but the relationship among the CG strategy, the FF activities, and their grouping was not conclusive. Wang et al. [12] described another postplacement algorithm that accounted implicitly for switching data to estimate the expected power. Although [6] and [12] used switching data as a secondary criterion in postplacement FF grouping, our strategy is to use it as a primary clustering criterion, and do so at the preplacement RTL level.

## **EXISTING SYSTEM:**

Clock gating:

Several techniques to reduce the dynamic power have been developed, of which clock gating is predominant. Ordinarily, when a logic unit is clock, its underlying sequential elements receive the clock signal regardless of whether or not they will toggle in the next cycle. Clock enabling signals are usually

introduced by designers during the system and clock design phases, where the interdependencies of the various functions are well understood. In contrast, it is very difficult to define such signals in the gate level, especially in control logic, since the inter-dependencies among the states of various flipflops depend on automatically synthesized logic. There is a big gap between block disabling that is driven from the HDL definitions, and what can be achieved with data knowledge regarding the flip-flops activities and how they are correlated with each other. The research presents an approach to maximize clock disabling at the gate level, where the clock signal driving a flip-flop is disabled (gated) when the flip-flop states is not subject to a change in the next clock cycle. Figure.1b shows enabling of the clock signal. On the other hand, such grouping may lower the disabling effectiveness, since the clock will disabled only when the inputs to all the flip-flops in a group don't change. It is, therefore beneficial to group flip-flops whose switching activities are highly correlated in derive a joined enabling signal.

The data of digital systems are usually stored in flip-flops (FFs), each of which has its own internal clock driver. In an attempt to reduce the clock power, several FFs can be grouped into a module called a multi bit FF (MBFF) that houses the clock drivers of all the underlying FFs. We denote the grouping of  $k$  FFs into an MBFF by a  $k$ -MBFF. Kapoor et al. reported a 15% reduction of the total dynamic power in a 90-nm processor design. Electronic design automation tools, such as Cadence Liberate,

support MBFF characterization. The benefits of MBFFs do not come for free. By sharing common drivers, the clock slew rate is degraded, thus causing a larger short-circuit current and a longer clock-to-Q propagation delay  $t_{pCQ}$ .

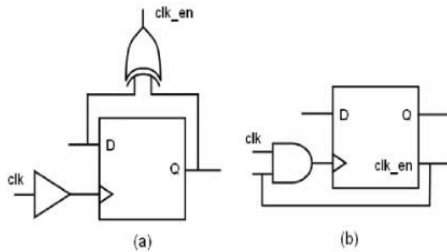


Figure : Enabling Clock signal

In a try to reduce the clock electricity, numerous FFs may be grouped in a module such that commonplace clock drivers are shared for all of the FFs. Two 1-bit FFs grouped into 2-bit MBFF, called additionally twin-bit FF [1], is proven in Fig. 1. In a similar way, grouping of FFs in 4-bit and eight-bit MBFFs are possible too. We in the end denote a  $ok$  -bit MBFF by means of  $k$  -MBFF. MBFF is not handiest lowering the gate capacitance pushed by means of a clock tree. The wiring capacitive load is likewise decreased because handiest a unmarried clock twine is required for more than one FFs. It additionally reduces the depth and the buffer sizes of the clock tree and also the number of sub-timber. Beyond clock energy financial savings the ones functions also lessen the silicon location.

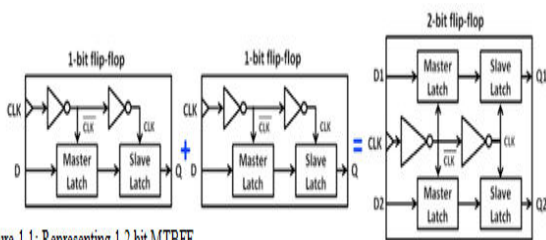


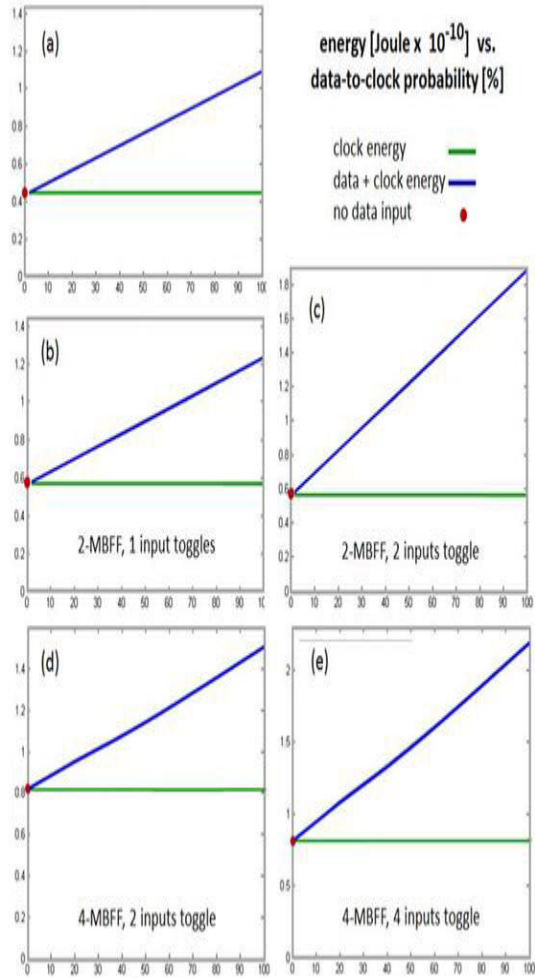
Figure 1.1: Representing 1, 2 bit MTBFF

To remedy this, the MBFF internal drivers can be strengthened at the cost of some extra power. It is therefore recommended to apply the MBFF at the RTL design level to avoid the timing closure hurdles caused by the introduction of the MBFF at the backend design stage. Due to the fact that the average data-to-clock toggling ratio of FFs is very small, which usually ranges from 0.01 to 0.1, the clock power savings always outweigh the short-circuit power penalty of the data toggling. An MBFF grouping should be driven by logical, structural, and FF activity considerations. While FFs grouping at the layout level have been studied thoroughly, the front-end implications of MBFF group size and how it affects clock gating (CG) has attracted little attention. This brief responds to two questions. The first is what the optimal bit multiplicity  $k$  of data-driven clock-gated (DDCG) MBFFs should be. The second is how to maximize the power savings based on data-to-clock toggling ratio (also termed activity and data toggling probability).

### **PROPOSED SYSTEM:**

Clearly, the best grouping of FFs that minimizes the energy consumption can be achieved for FFs whose toggling is highly correlated. Using toggling correlations for MBFF grouping has the drawback of requiring early knowledge of the value change dump vectors of a typical workload. Such data may not exist in the early design stage. More commonly available information is the average toggling bulk probability of each FF in the design, which can be estimated from earlier designs or the functional knowledge of modules. FFs'

toggling probabilities are usually different from each other. An important question is therefore how they affect their grouping. We show below that data-to-clock toggling probabilities matter and should be considered for energy minimization.



### Capturing everything in a design flow:

In the following paragraphs, we combine the activity  $p$  and the MBFF multiplicity  $k$  in a design flow aimed at minimizing the expected wasted energy. Fig. 2(a)–(c) illustrates that the power savings of the 2-MBFF, 4-MBFF, and 8-MBFF, respectively, are used. Knowing the activity  $p$  of an FF, the decision as to which MBFF size kit best

fits follows the interim lines, lines (d). To obtain the per-bitpower consumption, lines (d) in Fig. 2(a)–(c), representing an MBFF realistic operation, were divided by their respective multiplicity. The result is shown in Fig. 3.

Figure 2: Power consumption of  $k$ -bit FFs compared to  $k$ -MBFF: 2-MBFF (a), 4-MBFF (b) and 8-MBFF (c). Line (a) is the power consumed by  $k$ -bit FFs driven independently of each other. Line (b) is the ideal case of simultaneous (identical) toggling. Line (c) is the worst case of exclusive (disjoint) toggling. Line (d) is an example of realistic toggling.

To maximize the power savings, Fig. 3 divides the range of FF activity into regions. The black line follows the power consumed by a 1-bit un gated FF. The triangular areas bounded by the black line and each of the green, blue, and red per-bit lines show the amount of power savings per activity obtained by grouping an FF in the 2-MBFF, 4-MBFF, and 8-MBFF, respectively. It shows that for a very low activity, it pays to group FFs into an 8-MBFF. As activity increases, there will be some point where the 4-MBFF overtakes and pays off more than the 8-MBFF. At some higher activity, the 2-MBFF overtakes and pays off more than the 4-MBFF, up to an activity where the power savings stops. The remaining FFs can be grouped into un gated MBFFs, simply to reduce the number of internal.

A few practical comments are in order. The grouping should not cross clock domains. The clock enable signals introduced by the RTL synthesis and manually by designers are untouched. Groupings should also

consider logical relations and practical layout concerns. One example is the pipeline registers of a microprocessor, which are natural candidates for MBFF implementation (see Section V). It is expected that the place and route tool will locate bits belonging to the same register close to each other, whereas FF clusters of registers belonging to distinct pipeline stages will be placed away from each other. FFs belonging to different pipeline registers should therefore not be mixed in an MBFF. Similar arguments hold for other system buses and registers such as those storing data, addresses, counters, statuses, and the like. Another example is the FFs of finite-state machines, whose MBFF grouping should not cross control logic borders. Finally, the aforementioned post-placement MBFF clustering must consider the timing constraints, which are built into their algorithms. By contrast, the MBFF grouping algorithm does not require explicit timing constraints since it works at the RTL design level. In order to bridge the gap between the RTL grouping and the grouping driven by backend timing-closure considerations, we suggested appropriate DDCG design flow. The main idea involves providing “natural” physical layout directives for FF grouping by employing a prior placement. clock drivers.

## RESULTS AND DISCUSSIONS:

pipeline stage activity for array sorting program												
grouping method	IF / ID			ID / EXE			EXE / MEM			MEM / WB		
	2-MBFF	4-MBFF	8-MBFF	2-MBFF	4-MBFF	8-MBFF	2-MBFF	4-MBFF	8-MBFF	2-MBFF	4-MBFF	8-MBFF
	0.105			0.0856			0.0711			0.0473		
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		
	2-MBFF	4-MBFF	8-MBFF	2-MBFF	4-MBFF	8-MBFF	2-MBFF	4-MBFF	8-MBFF	2-MBFF	4-MBFF	8-MBFF
	0.127			0.118			0.0799			0.0582		
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 1. Average FF activity of pipeline registers in 32-bit MIPS.

The proposed MBFF layout waft has been used for a 32-bit pipelined MIPS processor, applied in TSMC 65nm process technology. Workload of two applications has been used, shown in Table 3. For every check the average interest of a FF in the pipelined register is proven in blue colour below the name of the pipeline stage. Notice the activity lower with the development of the pipeline stage from preparation fetch (IF) to jot down-again (WB).

Two MBFF grouping techniques are examined. In the primary, FFs were grouped sequentially in step with their bit wide variety in their sign in. The second approach grouped FFs in growing order of their sports, shown in Section 4 to be optimal whilst FFs are assumed to toggle independently of every other. Both grouping techniques adhered to the constraints of now not crossing clock area limitations and no longer blending FFs of unrelated logic entities. Table three suggests for each okay - MBFF, ok □ 2, 4, 8 the average hobby. In

most cases grouping by monotonic interest is preferred (coloured in green), even though in few cases it worsened (coloured in crimson). That can take place since the grouping is ignorant of toggling correlation. The pipeline registers have been then applied with MBFFs grouped with the aid of monotonic order of their hobby. As shown in Fig. Nine, the grouping starts offevolved with eight-MBFFs for the low activities, after which it is progressing to 4-MBFFs and 2-MBFFs with the FFs activities boom, up to the zero benefit factor wherein grouping stops and the rest FFs live alone and un-gated. Those should of course be grouped in un-gated MBFFs, just to lessen the quantity of internal clock drivers. Table four shows the strength financial savings completed at every of the pipeline registers for the kind and matrix multiplication weighted workload. The consequences have been measured with SpyGlass [11] simulation in which the MIPS changed into operated in 1.1V and 200MHz. 34.6% savings turned into performed. The pipelined registers consumed sixty five% of the whole MIPS electricity (memory now not protected), so the full energy reduction of the entire strength (CG HW overhead protected) changed into 23%.

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

**Table 2. Power savings in the pipeline registers of a 32-bit MIPS.**

We subsequently display the electricity savings performed by way of the grouping

set of rules for an entire commercial network processor designed in 28nm TSMC technique technology, operating in 800MHz. The processor is split into seven units, named A to G, shown in Table 6. It consumes a complete of 6.2 Watts, in which forty five% is charged to the clock community with its underlying FFs. The unique design incorporates un-gated MBFFs, so the electricity financial savings is purely due to the addition of the clock gating in Fig. 5, on pinnacle of the financial savings obtained by using much less drivers within the un-gated MBFFs that existed in the unique design. Furthermore, the unique layout includes giant clock permit common sense signals, described by way of each RTL compiler and guide insertions.

The sports of the FFs have been profiled first and then looked after. Table five shows a complete of eight% internet strength financial savings, in which the energy measurements include both dynamic and static components and all the CG HW overheads. The eight% strength savings changed into obtained on top of 9% savings that were performed through converting from 1-bit FFs to un-gated MBFFs, yielding a 17% combined financial savings. Such financial savings is fantastically preferred by using the industrial VLSI layout community. The vicinity penalty due to the creation of clock-gating circuitry was 2.3%.

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

**Table 3. Power savings in a 40nm network processor.**

### CONCLUSIONS:

This brief suggests combining MBFFs and probability-driven CG to increase their power savings. A model utilizing the relationship between the optimal MBFF multiplicities to FF data-to-clock toggling probabilities is used in a practical design flow, achieving 17% and 23% power savings, compared with designs with ordinary FFs. About half of these savings can be attributed to the integration of DDCG into MBFFs

### REFERENCES

[1] A. Kapoor et al., “Digital systems power management for high performance mixed signal platforms,” *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 61, no. 4, pp. 961–975, Apr. 2014.

[2] 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.

[3] C. Santos, R. Reis, G. Godoi, M. Barros, and F. Duarte, “Multi-bit flipflop usage impact on physical synthesis,” in *Proc. 25th IEEE Symp. Integr. Circuits Syst. Design (SBCCI)*, Sep. 2012, pp. 1–6.

[4] J.-T. Yan and Z.-W. Chen, “Construction of constrained multi-bit flipflops for clock power reduction,” in *Proc. IEEE Int. Conf. Green Circuits Syst. (ICGCS)*, 2010, pp. 675–678.

[5] IH-R. Jiang, C-L. Chang, and Y-M. Yang, “INTEGRA: Fast multibit flip-flop clustering for clock power saving,” *IEEE Trans. CAD Integr. Circuits Syst.*, vol. 31, no. 2, pp. 192–204, Feb. 2012.

[6] C. L. Chang and I. H. R. Jiang, “Pulsed-latch replacement using concurrent time borrowing and clock gating,” *IEEE Trans. Comput.-Aided Design Integr.*, vol. 32, no. 2, pp. 242–246, Feb. 2013.

[7] M. P.-H. Lin, C-C. Hsu, and Y-T. Chang, “Post-placement power optimization with multi-bit flip-flops,” *IEEE Trans. CAD Integr. Circuits Syst.*, vol. 30, no. 12, pp. 1870–1882, Dec. 2011.

[8] Y.-T. Shyu, J.-M. Lin, C.-P. Huang, C.-W. Lin, Y.-Z. Lin, and S.-J. Chang, “Effective and efficient approach for power reduction by using multi-bit flip-flops,” *IEEE Trans. VLSI Syst.*, vol. 21, no. 4, pp. 624–635, Apr. 2013.

[9] S. Liu, W.-T. Lo, C.-J. Lee, and H.-M. Chen, “Agglomerative-based flip-flop merging and relocation for signal wirelength and clock tree optimization,” *ACM Trans. Design Autom. Electron. Syst.*, vol. 18, no. 3, article no. 40, Jul. 2013.

[10] M. P. H. Lin, C. C. Hsu, and Y. C. Chen, “Clock-tree aware multibit flip-flop generation during placement for power optimization,” *IEEE Trans. Comput.-Aided Design Integr.*, vol. 34, no. 2, pp. 280–292, Feb. 2015.



- [11] C. Xu, P. Li, G. Luo, Y. Shi, and IH-R. Jiang, “Analytical clustering score with application to post-placement multi-bit flipflop merging,” in Proc. ACM Int. Symp. Phys. Design, 2015, pp. 93–100.
- [12] S.-H. Wang, Y.-Y. Liang, T.-Y. Kuo, and W.-K. Mak, “Power-driven flipflop merging and relocation,” IEEE Trans. CAD Integr. Circuits Syst., vol. 31, no. 2, pp. 180–191, Feb. 2012.
- [13] S.-C. Lo, C.-C. Hsu, and MP-H. Lin, “Power optimization for clock network with clock gate cloning and flip-flop merging,” in Proc. ACM Int. Symp. Phys. Design, 2014, pp. 77–84.
- [14] S. Wimer, D. Gluzer, and U. Wimer, “Using well-solvable minimum cost exact covering for VLSI clock energy minimization,” Operations Res. Lett., vol. 42, no. 5, pp. 332–336, Jul. 2014.
- [15] SpyGlass Power, accessed on 2016. [Online]. Available: <http://www.atrenta.com/solutions/spyglass-family/spyglass-power.htm>
- [16] Y.-T. Chang, C.-C. Hsu, M. P.-H. Lin, Y.-W. Tsai, and S.-F. Chen, “Postplacement power optimization with multi-bit flip-flops,” in Proc. IEEE Int. Conf. (CAD), Nov. 2010, pp. 218–223.
- [17] Hsu, Chih-Cheng, Mark Po-Hung Lin, and Yao-Tsung Chang, “Crosstalk-aware multi-bit flip-flop generation for power optimization,” Integr. VLSI J. vol. 48, pp. 146–157, 2015.