

Flip-Flop Grouping in Data-Driven Clock Gating

Varghese James A¹, Divya S², Seena George³

¹(Electronics and Communication Department, Sree Narayana Gurukulam College of Engineering, India)

²(Electronics and Communication Department, Sree Narayana Gurukulam College of Engineering, India)

³(Electronics and Communication Department, Sree Narayana Gurukulam College of Engineering, India)

Abstract : The major dynamic power consumers in computing and consumer electronics products is the system's clock signal, typically responsible for 30%–70% of the total dynamic power consumption. Clock gating is a predominant technique used for power saving. It is observed that the commonly used synthesis based gating still leaves a large amount of redundant clock pulses. Data-driven gating aims to disable these. To reduce the hardware overhead involved, flip-flops (FFs) are grouped so that they share a common clock enabling signal. Data-driven clock gating is employed for FFs at the gate level. The clock signal driving a FF is disabled when the FFs state is not subject to change in the next clock cycle. Data-driven gating is causing area and power overheads that must be considered. In an attempt to reduce the overhead, it is proposed to group several FFs to be driven by the same clock signal, generated by ORing the enabling signals of the individual FFs. Pseudo-random bit generators (PRBGs) are widely used in many electronic equipment, thus many researchers are proposing novel solution addressed to improve the inviolability performances required in cryptographic applications. LFSR is the most used topology to implement PRBG. In this paper, a method to reduce the power consumption of the popular linear feedback shift registers. The proposed scheme is based on the gated clock design approach and it can offer a significant power reduction.

Keywords – Data Driven clock gating, Linear feedback shift register, Pseudo random bit generator;

I. INTRODUCTION

The increasing demand for low power mobile computing and consumer electronics products has refocused VLSI design in the last two decades on lowering power and increasing energy efficiency. Power reduction is treated at all design levels of VLSI chips. From the architecture through block and logic levels, down to gate level circuit and physical implementation, one of the major dynamic power consumers in the system clock signal, typically responsible for up to 50% of the total dynamic power consumption. Clock network design is a delicate procedure, and is therefore done in a very conservative manner under worst case assumptions. It incorporates many diverse aspects such as selection of sequential elements, controlling the clock skew, the decision of the topology and physical implementation of the clock distribution network.

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 they will toggle in the next cycle.

Clock enabling signals are usually introduced by designers during the system and clock design phases, where the inter-dependencies 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 flip-flops 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.

Fig. 1.1 shows how a FF can find out that its clock can be disabled in the next cycle.

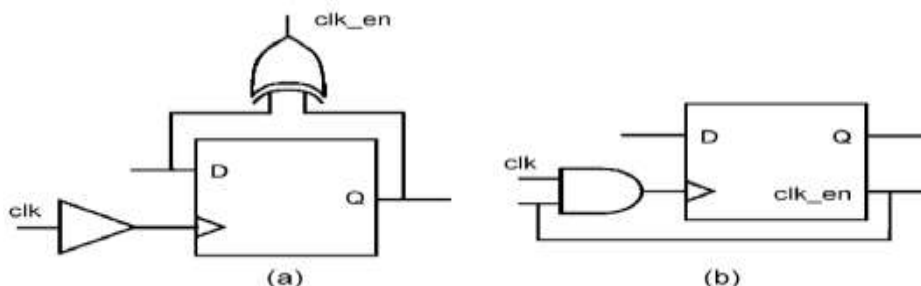


Fig.1.1. Enabling of the clock signal.

An XOR gate compares the FF's current output with the present data input that will appear at the output in the next cycle. The XOR's *clk_en* output indicates whether a clock signal will be required in the next cycle. The clock driver in Fig. 1.1(a) is then replaced by a 2-way AND gate called *clock gater*. We will use the symbol in Fig. 1.1(b) to represent FFs that incorporate generation of *clk_en*. In practice the XOR is connected to the output of FF's internal master rather than D, as it is guaranteed to be stable when the FF's slave is transparent.

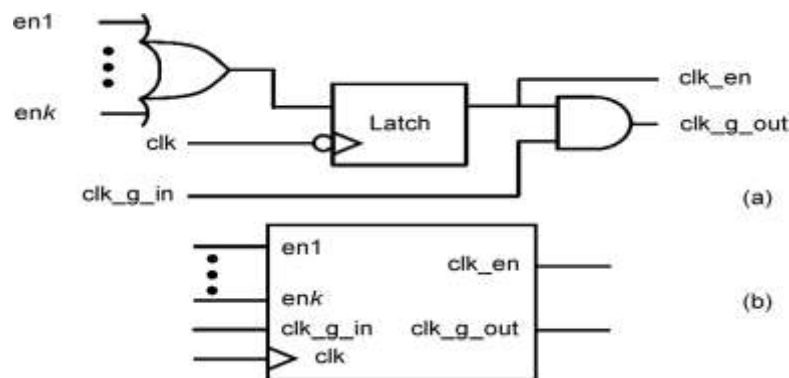


Fig.1.2. Joining k enabling signals generated by distinct flip-flops into one gating signal

Clock gating does not come for free. Extra logic and interconnects are required to generate the clock enabling signals, and the resulting area and power overhead must be considered. In the extreme case, each clock input of a flip-flop can be disabled individually, yielding maximum clock separation. This, however, results in high overhead. Thus, the clock disabling circuit is shared by a group of several flip-flops in an attempt to reduce the overhead. We could drive several FFs with a common gater if we knew that they are toggling simultaneously most of the time, thus achieving almost the same power reduction, but with fewer gaters. Fig.1.2 shows how to join k *clk_en* signals generated by distinct FFs into one gating signal.

II. LITERATURE SURVEY

Several dynamic power management techniques are adopted in vlsi circuits out of which the major one is clock gating.[5] uses multiple supply voltages to reduce clock tree power. The incoming, high voltage clock signal is down-scaled by means of a low-voltage buffer stage. The low-*Vdd* signal is then propagated throughout the circuit, and regenerating elements (e.g., buffers) are inserted into the tree structure to ensure the appropriate speed and slew rate of the transitions. Finally, the original high-voltage is restored through level-shifters before the clock signals feed the flip-flops.

In [3] Clock Distribution using Multiple Voltages in reduces the cost of buffering and voltage converters that is essential in the power reduction technique implemented using multiple supply voltages. The approach presented by Pangjung and Sapatnekar addresses this limitation by providing a more sophisticated algorithm for introducing buffers into the clock tree and for placing the low-to-high voltage shifters, which are now not necessarily located right in front of the flip-flops. The algorithm considers the possibility of buffer

insertion after every step of bottom-up sub tree merging. In the interest of keeping the skew very close to zero, the algorithm guarantees that the number of regenerating elements is equalized along any root-to-sink paths of the tree. However, in spite of the solid theoretical basis of this solution, experimental results showed very small differences with the clock trees generated by the approach using multiple supply voltages

In [2] focuses on Interconnect Power, i.e. energy dissipation due to the switching of interconnection capacitances, which are part of the total switched capacitance of each net– C_j . Applying wire capacitance reduction techniques to a small percentage of the wires can save the majority of the interconnect power. Capacitance can be reduced by interconnect reduction and increasing interconnect spacing thereby reducing capacitance and thereby reducing power dissipation.

In [4], a review of some existing techniques available for clock gating is presented. Also a new technique that provides more immunity to the existing problems in available techniques is discussed.

III. PROPOSED WORK

1. Data Driven Clock Gating

The principle of Data driven clock gating saves the individual clock gaters at the expense of an OR gate and a negative edge triggered latch that is required to avoid glitches of the enable signal. The combination of a latch with an AND gate is commonly used by commercial tools and is called integrated clock gate (ICG). Fig 3.1 illustrates the approach of Data driven clock gating.

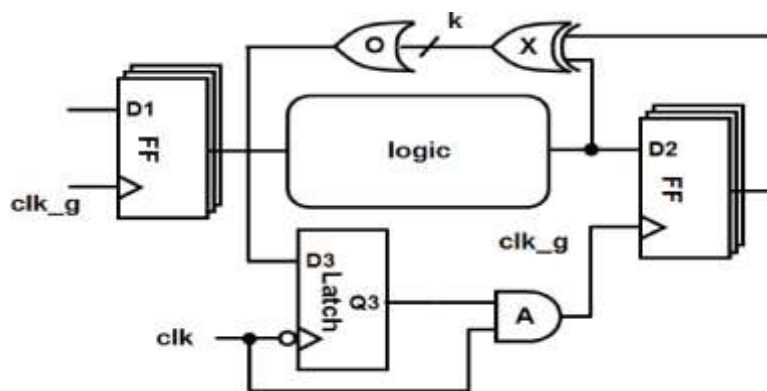


Fig.3.1 Practical data-driven clock gating. The latch and gater (AND gate) overheads are amortized over k FFs.

In the physical implementation, the XOR gate is integrated into the FF, while the OR gate, AND gates and the latch are integrated into the clock gater. There are two distinct clock signals: clk_g is the ordinary gated signal driving the registers, while clk is driving the latches of the clock gaters. Clearly, the hardware savings increases with k , but the number of disabled clock pulses is decreasing. Thus, for the scheme proposed in Fig. 3.1 to be beneficial, the clock enabling signals of the grouped FFs must be highly correlated.

1.1 Joint Gating And Gater's Optimal Fan-Out

Assume that a circuit contains $n=2^N$ FFs whose clock signals are driven by the clock tree. Its leaves are connected to the FFs and gaters fan-out is $k=2^K$. We assume that $N=\alpha K$, where α is the number of levels of the clock tree. A leaf gater has unit size (driving strength). The gater at the first level is connected to the leaf by a wire of unit length and unit width. We now introduce the following notations to quantify and analyze the power savings achieved by joint clock enabling: C_{ff} —FF's clock input capacitance; C_{latch} —latch capacitance; including the wire capacitance of its input; C_w —unit wire capacitance; C_{gater} —unit drive gater capacitance; C_{OR} —OR gate capacitance.

To assess the clock gating impact on power we consider the toggling of FF as an independent random variable. A FF has probability to change state and $q=1-p$ to stay unchanged. The probability of a group of FFs to stay unchanged (as a group) is therefore q^k . The probability p is sometimes called *activity factor*. It is well known that the average activity factor of non clock signals is very low, since a typical signal toggles very infrequently. Clock gating incurs certain power and area costs. As shown in Figs. 1.1–3.1, FFs need additional XOR gates and every gater requires a k -way OR gate and a latch. Moreover, there is a wiring penalty resulting from the separation of clk_g and clk . The interconnections realizing clk_g are switching only when the clock is

required for FF toggling. These are the real functional clock wires with the full sizing required to deliver high quality clock signal.

The interconnections propagating clk are needed for the latches residing at the gaters and are used at each cycle. Notice that clk exists only at gaters in the first level of the tree and above, but does not exist at the leaves (FFs). There are also the clk_en signals, feeding back the activity of k children gaters (or FFs at leaves) to the OR gate at their parent. These wires can be of a minimum width, subject to delay constraints. A reasonable assumption for the subsequent analysis is that their length is similar to that of clk_g since they connect the same elements as clk_g does.

The calculation of the power consumed by the shadow tree with its logic overhead is based on toggling probabilities. An enabling signal informs the gater at level j whether its child gater at level j-1 needs the clock pulse in the next cycle. The toggling independence is a worst case assumption since toggling correlation increases power savings as it reduces the probability of a gater to send a clock signal to a FF when it does not need it. We calculate the net power savings, denoted by C_{net_saving} , $1 \leq j \leq \alpha$ for a single branch of the tree and then sum over all branches. At the leaves where FFs are connected ($j=1$), the net power savings per branch satisfies,

$$C_{net_saving} \geq q^k (C_{ff} + C_w) - [(Clatch/k) + (1-q) (C_w + COR)] \quad (1)$$

The term $q^k (C_{ff} + C_w)$ in (1) is the savings due to the disabling of clk_g. The term $Clatch/k$ is the overhead due to the latch at the parent gater being always clocked by the clk signal. The division by stems from the fact that the latch overhead is amortized among the branches connected to the gater. The overhead $(1-q) (C_w + COR)$ is due to the switching of clk_en. Notice that if the probability of a FF to toggle is $p=1-q$, then $Pr (clk_en=1) = 1-q$ and hence its switching probability cannot exceed $1-q$. Equating to zero the derivative of (1) with respect to k yields the following implicit equation for the optimal k:

$$q^k \ln q (C_{ff} + C_w) + (Clatch/k^2) = 0$$

The lower the toggling probability of FF is, the higher the optimal k is. The resolution of gating proposed in this paper is of individual FFs at individual clock cycles. Gating at that resolution has been proposed for regularly structured circuits such as Linear Feedback Shift Register (LFSR) and counters, where the amount of power savings can be predicted from the circuit's structure. An implementation of the concept of clock gating described in this paper is done in traditional Linear Feedback shift register circuit.

2.Linear Feedback Shift Register

A **linear-feedback shift register** (LFSR) is a shift register whose input bit is a linear function of its previous state. The most commonly used linear function of single bits is XOR. Thus, an LFSR is most often a shift register whose input bit is driven by the exclusive-or (XOR) of some bits of the overall shift register value.

The initial value of the LFSR is called the seed, and because the operation of the register is deterministic, the stream of values produced by the register is completely determined by its current (or previous) state. Likewise, because the register has a finite number of possible states, it must eventually enter a repeating cycle. However, an LFSR with a well-chosen feedback function can produce a sequence of bits which appears random and which has a very long cycle. Applications of LFSRs include generating pseudo-random numbers, pseudo-noise sequences, fast digital counters, and whitening sequences. Both hardware and software implementations of LFSRs are common. Fig.3.2 shows a 4 bit Fibonacci LFSR and its state diagram

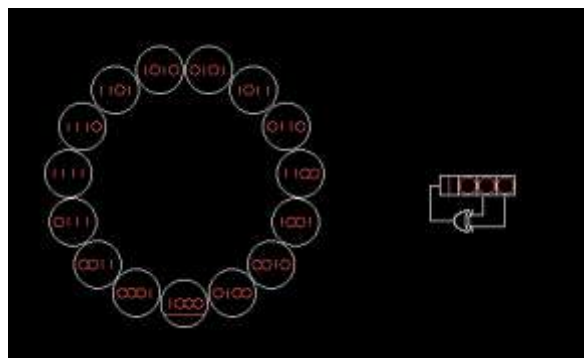


Fig: 3.2. A 4-bit Fibonacci LFSR with its state diagram

The XOR gate provides feedback to the register that shifts bits from left to right. The maximal sequence consists of every possible state except the "0000" state.

2.1 Fibonacci Lfsrs

The bit positions that affect the next state are called the taps. In the diagram the taps are [16, 14, 13, and 11]. The rightmost bit of the LFSR is called the output bit. The taps are XOR'd sequentially with the output bit and then fed back into the leftmost bit. The sequence of bits in the rightmost position is called the output stream. The bits in the LFSR state which influence the input are called *taps* (white in the diagram). Fig.3.3 shows a 16-bit Fibonacci LFSR with its taps chosen.

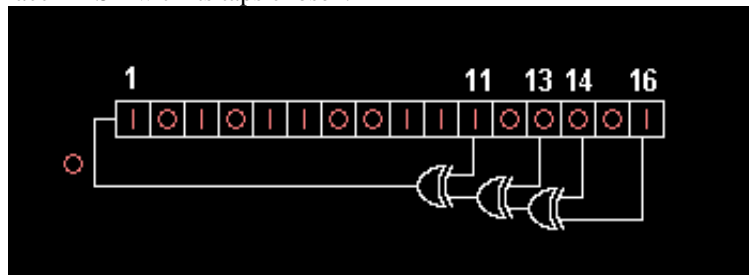


Fig: 3.3. A 16-bit Fibonacci LFSR.

The feedback tap numbers in white correspond to a primitive polynomial in the table so the register cycles through the maximum number of 65535 states excluding the all-zeroes state. A maximum-length LFSR produces an m-sequence (i.e. it cycles through all possible $2^n - 1$ states within the shift register except the state where all bits are zero), unless it contains all zeros, in which case it will never change.

2.2.Gated-Clock Design of Linear-Feedback Shift Registers

In this paper, a method to reduce the power consumption of the popular linear feedback shift register is presented. The proposed scheme is based on the gated clock design approach and it can offer a significant power reduction, depending on technological characteristics of the employed gates. A traditional LFSR with ordinary clock gating is first implemented using Vhdl coding and simulated in Xilinx 14.2. The Output is then compared with an LFSR with data driven clock gating introduced into the clock circuitry. As shown in Fig. 3.4, an LFSR is obtained with an array of FFs with a linear feedback performed by several XOR gates.

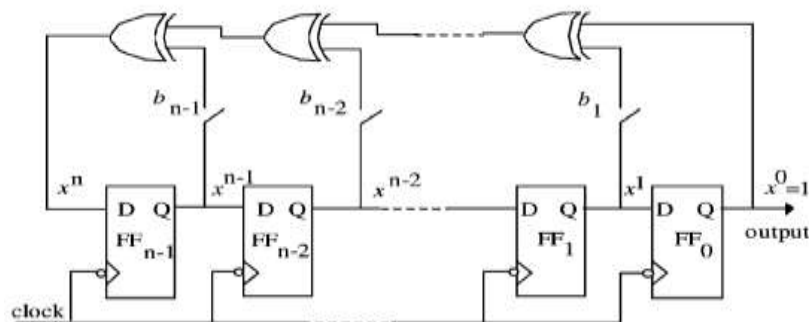


Fig.3.4. Simplified circuit of a generic n-bit LFSR

Although LFSRs are very simple to implement, they are based on a rather complex mathematical theory. However, they can be efficiently describes through the nth-order polynomial,

$$P_n(x) = x^n + b_{n-1}x^{n-1} + \dots + b_1x + 1$$

where the binary coefficients b_i define the well-known polynomial characteristic which the generator main properties depends on. As it is well known, LFSRs exhibit a high-speed bit generation and they have very good statistical properties. The main drawback for these generators is the high power consumption given as,

$$PTR = nPFF + t.PXOR$$

where n is the register's length (i.e., the order of the generator), t is the number of the inner taps (i.e., the number of the terms of the polynomial characteristic except and 1). The terms PFF and PXOR are the

dynamic power consumption of D-FF and XOR gates respectively. Both PFF and PXOR are proportional to $V_{dd}^2 f_{ck}$, where V_{dd} is the supply voltage and f_{ck} the clock frequency.

Here a method to reduce the power consumption of the popular linear feedback shift register is presented by applying the Data Driven clock gating technique and it can offer a significant power reduction, depending on technological characteristics of the employed gates. A traditional LFSR with ordinary clock gating is first implemented using VHDL coding and simulated with ISim simulator in Xilinx 14.2. The Output is then compared with an LFSR using Data driven clock gating.

The clock path toggles at every clock cycle, thus dissipating a significant amount of power especially at high clock rate. Vice versa, power consumption of the D-path and the XOR gates depend on the switching activity at the inner node. The clock domain power is found to be very high for various chosen clock frequencies as given in table 1.

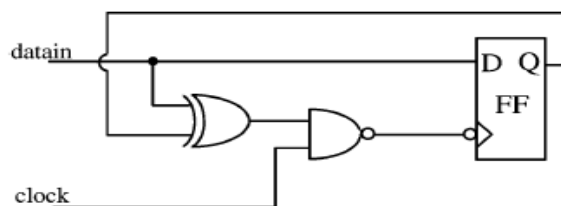


Fig. 3.5 Gated clock based on synchronization respect to input and output of the FF.

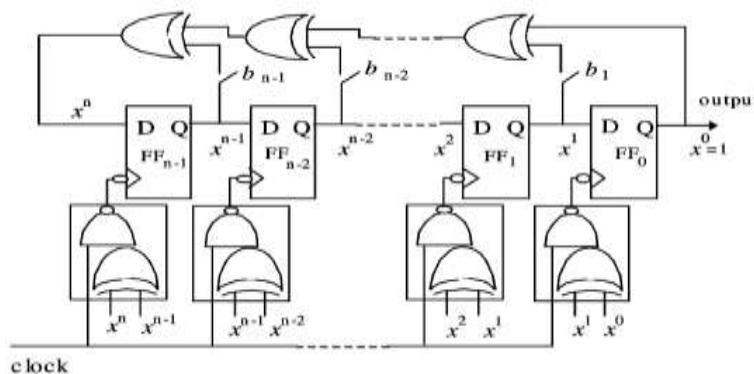


Fig.3.6. Gated-clock n-bit LFSR

3 LFSR USING DATA DRIVEN CLOCK GATING

In data driven clock gating approach flip-flops are organized in to different groups based on the toggling probability and different clocks are given to each group. So we need not have to give so supply master clock all the time. 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 that the number k of jointly gated FFs for which the power savings are maximized is the solution of

$$(1 - p)k \ln(1 - p) (c_{FF} + c_W) + c_{latch}/k = 0$$

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.

p	0.01	0.02	0.05	0.1
k	8	6	4	3

Table 1. Dependency of optimal FF group size on toggling probability

With an increase in k , the hardware overhead decreases and the toggling probability also decrease. Hence in the design of a 32 bit LFSR circuit, we may group four FFs combined to form single group which means $k=8$. So here, we have eight groups ($8 \times 4 = 32$). Fig.3.7 shows a 32 bit LFSR using Data driven clock gating technique. The LFSR uses a gated clock for each of the 8 groups so that the each clock may drive four FFs simultaneously.

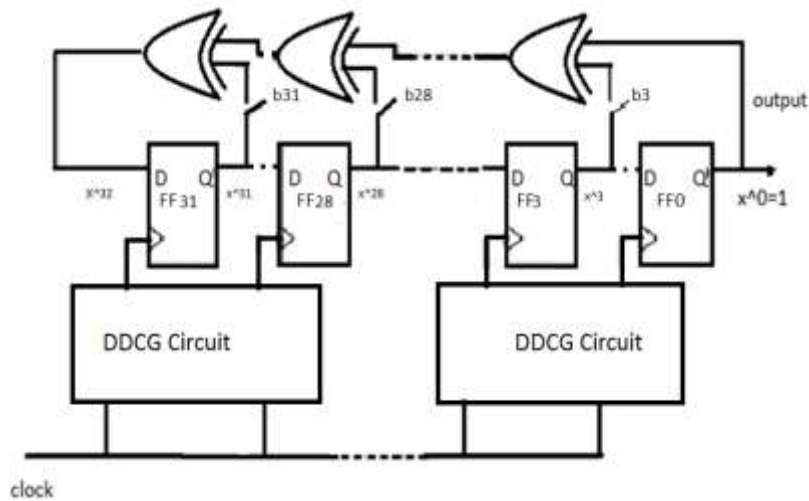


Fig.3.7 LFSR circuit with data driven clock gating.

Clock utilization is analyzed from the simulation run in Xilinx platform. Also the power report is generated to find the power consumptions. The clock domain power is reduced to 0.38 mW from 1.27mW as in the case of an LFSR using ordinary clock gating approach for a clock frequency of 50 MHz. For higher frequencies say, 100 MHz and 1000 MHz there is further significant power reduction in clock domain power. The results are tabulated in table 2.

The clock gating circuitry as shown in Fig.3.8 is used for generating eight child clocks from the master clock signal for triggering each set of flip flops. For reducing complexity of the figure only the first and last groups are shown in the figure.

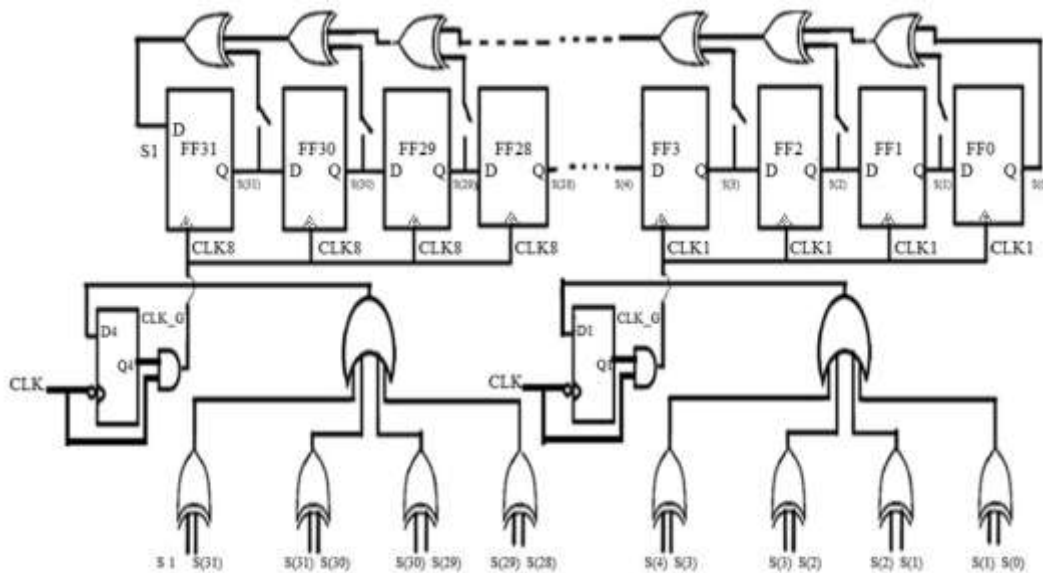


Fig 3.8 Detailed circuitry of the LFSR

IV. RESULT AND ANALYSIS

1 .LFSR with ordinary Clock Gating

While in ordinary Clock Gating 32 different clocks are given to 32 flip-flops. Here the number of clocks is large. More power will be consumed in this large number of clocks.

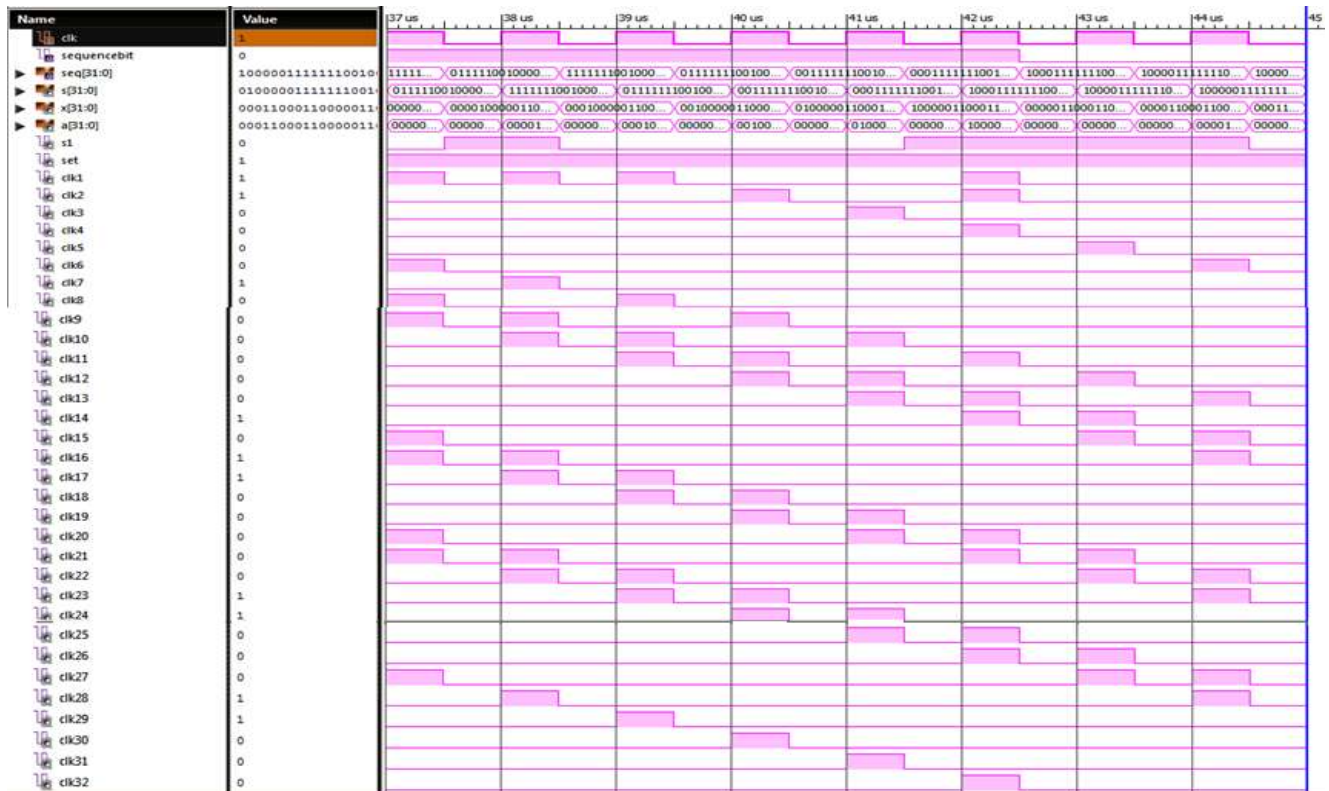


Fig.4.1. Simulation waveforms of an LFSR with ordinary clock gating

2. LFSR using Data Driven Clock Gating

In data driven Clock gating the number of clocks are reduced from 32 to 8. Hence a large amount of power and area can be consumed.



Fig.4.2. Simulation waveforms of an LFSR with Data-Driven clock gating

3. Comparison of Power Consumption of LFSR

As the frequency increases the power consumption also increases. while we are using LFSR with ordinary clock gating the power consumption will be large. when LFSR with Data Driven Clock Gating used

Power consumption (mV)	F=50 MHz		F=100 MHz		F=1000 MHz	
	LFSR with ordinary clock gating	LFSR with DDCG	LFSR with ordinary clock gating	LFSR with DDCG	LFSR with ordinary clock gating	LFSR with DDCG
Clock power	1.27	.38	2.54	0.76	25.41	7.59
Dynamic power	7	12	27	25	1894	452
Total power	50	54	69	68	1942	496

number of clocks be reduced and there by power consumption will also be reduced

Table 2: Compares the power dissipation of a LFSR with a DDCG LFSR and an LFSR with ordinary clock gating for varying clock frequencies.

V. CONCLUSION

This paper studied the problem of grouping FFs for joint clocking by a common gater to yield maximal dynamic power savings in a 32 bit LFSR circuit. In order to evaluate the power reduction obtained by applying DDCG in LFSR, we have evaluated the power consumption in 32 bit LFSR with traditional clock gating and power consumption in 32 bit LFSR with DDCG for same input vector and same clock cycles. VHDL code of former LFSR was simulated in Xilinx 14.2 ISE Navigator and then VHDL code of latter was simulated and synthesized and the Xpower was obtained using Xilinx XPower Analyzer. The proposed LFSR is found to be power efficient than an ordinary LFSR. As an extended work the proposed LFSR can be applied to other circuits such as pseudo random number generators as well for obtaining better performance results. Dynamic power reduction can be done by reducing the clock capacitance instead of focusing on the clock frequency as done in this paper since power consumption has a linear relationship with clock capacitance also.

Acknowledgements

My profound and sincere expression of thanks and gratitude to mentors Ms. Divya. S and Ms. Seena George for their unstinted support, help and guidance. My sincere thanks to one and all who may not find a mention but have always wished only success.

REFERENCES

Transaction Papers:

- [1] Shmuel Wimer, , and Israel Koren, "Design Flow for Flip-Flop Grouping in Data-Driven Clock Gating", IEEE Trans. Very Large Scale Integr. (VLSI) Syst.
- [2] H. Zhang, J. Rabaey, "Low-Swing Interconnect Interface Circuit"s, ISLPED-98: ACM/IEEE International Symposium on Low-Power Electronics and Design, pp. 161-166, Monterey, CA, August 1998.
- [3] J. Pangjun, S. S. Sapatnekar, "Clock Distribution using Multiple Voltages," ISLPED-99: ACM/IEEE International Symposium on Low-Power Electronics and Design, pp. 145-150, San Diego, CA, August 1999.
- [4] Jagrit Kathuria, M. Ayoubkhan, Arti Noor. "A Review of Clock Gating Techniques", MIT International Journal of Electronics and Communication Engineering Vol. 1 No. 2 Aug 2011 pp 106-114.

Proceedings Papers:

- [5] L. Benini, A. Bogliolo, and G. De Micheli, "A survey on design techniques for system-level dynamic power management," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 8, no. 3, pp. 299–316, Jun. 2000.
- [6] M. Igarashi, K. Usami, K. Nogami, F. Minami, Y. Kawasaki, T. Aoki, M. Takano, C. Misuno, T. Ishikawa, M. Kanazawa, S. Sonoda, M. Ichida, N. Hatanaka, "A Low-Power Design Method using Multiple Supply Voltages", ISLPED-97: ACM/IEEE International Symposium on Low-Power Electronics and Design, pp. 36-41, Monterey, CA, August 1997.
- [7] N. Magen, A. Kolodny, U. Weiser, and N. Shamir, "Interconnect-power dissipation in a microprocessor," in Proc. Int. Workshop Syst. Level Int. Predict., 2004, pp. 7–13.
- [8] S. Wimer and I. Koren, "The Optimal fan-out of clock network for power minimization by adaptive gating", IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 20, no. 10, pp. 1772–1780, Oct. 2012
- [9] W. Aloisi and R. Mita, "Gated-clock design of linear-feedback shift registers," IEEE Trans. Circuits Syst., II, Brief Papers, vol. 55, no. 5, pp. 546–550, Jun. 2008.
- [10] Y.-T. Chang, C.-C. Hsu, M. P.-H. Lin, Y.-W. Tsai, and S.-F. Chen, "Post-placement power optimization with multi-bit flip-flops" in Proc. IEEE/ACM Int. Conf. Comput., Aided Design, Nov. 2010, pp. 218–22.