ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Monthly Peer Reviewed Open Access International e-Journal

A Reconfigurable low power FPGA Design with Autonomous power Gating and LEDR Encoding

Mr. K. Anji Babu
M.Tech Student, VLSI Design,
Nova College of Engineering and Technology.

Mr.M.Purna Sekhar
Head of the Dept,
Nova College of Engineering and Technology.

Abstract:

Field Programmable Gate Arrays (FPGAs) are widely used to implement special purpose processors. FPGAs are economically cheaper for low quantity production because its function can be directly reprogrammed by end users. This paper investigates to design a reconfigurable low power Asynchronous FPGA cells. FPGAs consume high dynamic and standby power. In order to reduce the standby power by using autonomous fme grain power gating and reducing the dynamic power by using the level encoding dual rail (LEDR) architecture.

The proposed autonomous finegrain power gating method each lookup table has its own sleep transistor and related sleep controller. So when any lookup tables are inactive, they can be set to sleep mode immediately. LEDR encoding is used to data flow at input and output of FPGAs, it reduces the dynamic power. The circuit is simulated using Xilinx tool. Power reduction is achieved by selectively setting the functional units into a low leakage mode when they are inactive

Keywords:

Asynchronous Field Programmable Gate Array (FPGA), Power Gating, Level Encoded Dual Rail (LEDR) Encoding, Logic Block, Lookup Table, Sleep Controller.

I. INTRODUCTION:

In FPGA design, the clock gating and power gating is important work. To implement clock gating, circulation is employed. The idea of circulation is to retain the contents of the flip-flop in the sleep state. Circulation can reduce the dynamic power consumption of registers and the gates in the fan-out of the registers. However, the standby power consumption of the clock network cannot be reduced.

The standby power is a serious problem because it has an enormously large number of transistors to achieve its programmability. Low-cost FPGAs consume up to hundreds of milliwatts power. Power gating has emerged as the most effective design technique to achieve low standby power. Power gating techniques are based on selectively setting the functional units into a low leakage mode when they are inactive. Power gating technique designed by the following methods 1) Sleep controller 2) Sleep transistor 3) Sleep signal distribution network [1],[5]&[6]. Power gathering Techniques are two types.

Coarse Grain Power Gating :

Large number of lookup tables grouped and shared in single sleep controler. In FPGA all the lookup tables are not active state in same time. Depends on the progrose number of lookup tables are active state. If any of the lookup table active state in the group other lookup tabes does not goes to sleep state, so In this techniqe only consume little power consumetion designed reference[3].

Fine Grain Power Gating:

Over come the problems of coarse grain power gating we introduce the Fine grain power gating technique. In fine grain power gating technique each look up table having own sleep controller and related to sleep transistor, so any of the lookup table active states all other lookup table are goes to sleep state[I]. In this paper reduce the both standby power and dynamic power.

I I .Previous Work

A. Asynchronous Architecture Design

The asynchronous architecture it detects the activity of a power gated domain. The activities are:

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Monthly Peer Reviewed Open Access International e-Journal

1) To detennine when logic block is standby state, when sleep state & when active state. 2) It compares the phase of the input data and output data 3) It determine the function of lookup table. Dynamic power reducing purpose introduce dual rail encoding (existing)[2] and level encoding dual rail (proposed)architecture. Stand-by power reducing purpose introduced autonomous fine grain power gating technique.

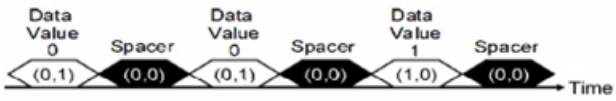

Fig. 1: DUAL RAIL ENCODING DATA Transmission

B. Dual Rail Encoding:

They use four-phase dual-rail encoding because of relatively small hardware cost. In four-phase dual-rail encoding, a spacer must be inserted between two consecutive valid data values. This results in low throughput and high dynamic power consumption because of the large number of signal transitions.

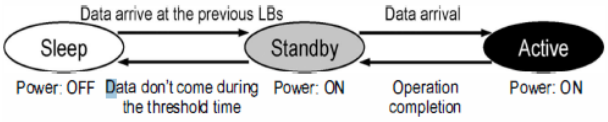

Fig 2. :CONTROL STRATEGY OF THE PROPOSED POWER GATING METHOD

Level Encoding Dual Rail:

Proposed asynchronous FPGAs based on LEDR encoding. LEDR is one of several two-phase dual-rail encodings. In LEDR encoding, no spacer is required. This results in high throughput and low dynamic power consumption because of the number of signal transitions reduced by half [3],[4].

Autonomous Fine Grain Power Gating:

The proposed autonomous fine grain power gating shown To solve this problem, we propose an efficient control strategy of the autonomous fine-grain power gating. The standby state is used to do the following:
1. wake up the LB before the data arrives
2. power OFF the LB only when the data does not come for quite a while.

Wave-Pipeliningfor Bit-Serial FPGAs :

In FPGAs, area for routing is dominant. To reduce the routing area without performance degradation, wave-pipe lining is combined with bit-serial architecture. To reduce the area in FPGAs, reducing the complexity of interconnect using bit-serial architecture is efficient. However, bit-serial data transmission decreases the throughput. To achieve a both a simple interconnect and high throughput, bit-serial wavepipe lining FPGA architectures have been proposed.

III. ARCHITECTURE DESIGN

The overall architecture proposed in reference [3]. The block diagram of the proposed FPGA shown below Figure 3. Mesh controlled cellular array based on bit serial architecture used. Each logic block mainly consists of a look up table, an output register and a sleep controllers. The look up table operates arbitrary of two input and one output logic function. The registers store the data value and produce the output to switch block. Sleep controller monitor wake up the successive block when it gets data.

The switch block consists of pass transistor switches. In a switch block, a wire-set consists of four wires: two for data lines (Vout and Rout), one for the acknowledge signal and one for the wake-up signal. A pass-switch block consists of four pass switches and a single memory bit. This FPGA architecture logic block can be connected to the switch block. In the switch block there are four signals 1) data signal (first bit) 2) data signal (second bit) 3) acknowledgement signal 4) data arrival signal. The above four signal acknowledgement signal and data arrival signal connected to pervious logic block.

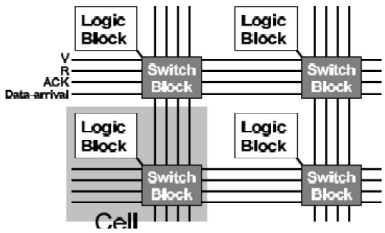

FIG. 3: BLOCK DIAGRAM OF THE PROPOSED FPGA

The two pass switches are used for the four wires of the wire-set, one Va, Ra, ack and wakeup signal wires respectively. The pass switches are controlled by the same memory bits.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Monthly Peer Reviewed Open Access International e-Journal

A.LEDR Encoding Design:

The design of LEDR encoding proposed in reference [3],[4].

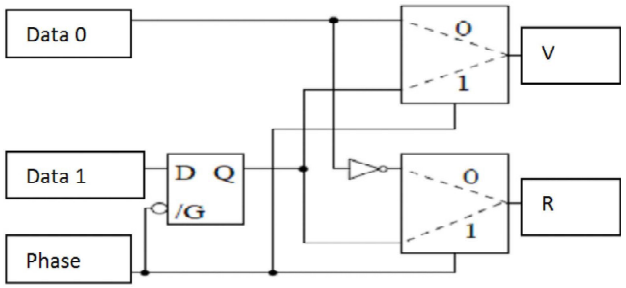

FIG. 4: LEVEL ENCODING DUAL RAIL ARCHITECTURE

This architecture operation based on data's and phase signal. The D latch signal given to selectors and selectors produce the output of level encoding signal.

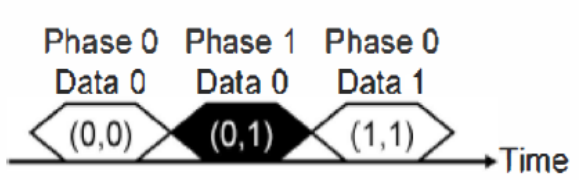

Fig. 5: LEDR ENCODING DATA TRANSMISSION

In LEDR encoding, no spacer is required. Table 1 shows the code table of LEDR encoding. In LEDR encoding, each data value has two types of code words with different phases. Figure 7 shows the example where data values "0," "0," and " 1" are transferred. The main feature is that the sender sends data values alternately in phase ° and phase 1. Because no spacer is required, the number of signal transitions is half of four-phase dual-rail encoding. As a result, the throughput is high and the power consumption is small. Based on this observation, in the proposed FPGA, LEDR encoding is employed for implementing the asynchronous architecture to reduce the dynamic power.

Table 1 : CODE TABLE OF LEDR ENCODING:

		Code word (V, R)
Phase 0	Data 0	(0,0)
	Data 1	(1,1)
Phase 1	Data 0	(0,1)
	Data 1	(1,0)

B. Circuit Implementation:

The circuit can be implemented by using below the architectures. The logic block can be connected to the switch oblock. Each switch block connected to the other switch block. This logic block and switch block described below the architecture diagrams.

C. Logic Block Design:

The overall structure of logic block shown Figure 6.

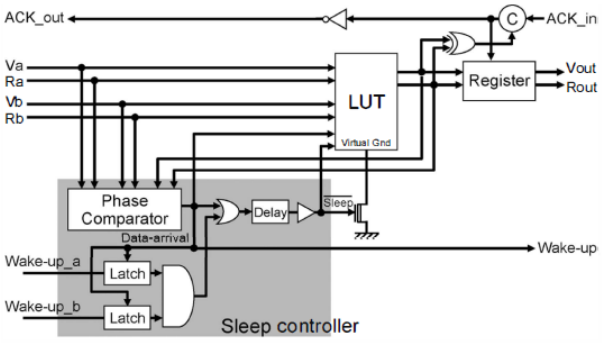

Fig. 6 : BLOCK DIAGRAM OF LOGIC BLOCK

In this logic block contains lookup table, sleep controller, registers and programmable delay elements presented. The description of the logic block described below.

(a) Lookup Table Design:

This architecture contains four sub modules. Each sub modules consist of a decoder, a multiplexer and memory bits. The decoder designed by two four input AND gates. The output of the decoder is given to the multiplexer. The multiplexer is designed by four pass transistor logic and one inverter logic. The decoders exclude invalid input patterns with different phases. The valid data are fed to the multiplexer.

As a result, the numbers of multiplexers are reduced and the transistor count is reduced compared to the multiplexer type LOOKUP TABLE. If the combination of inputs are invalid (i.e., if the two inputs have the different phases) all passtransistors turn OFF according to the output of the decoder. The decoder and multiplexer based lookup table shown Figure 7. The designed reference [4] .

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Monthly Peer Reviewed Open Access International e-Journal

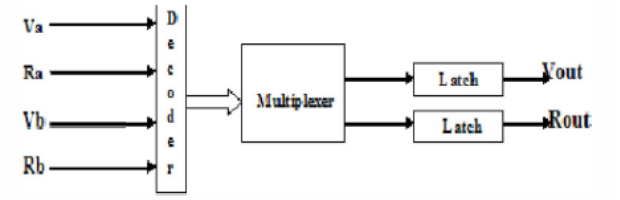

Fig. 7 : BLOCK DIAGRAM OF THE PRO

The detailed structures of the decoder and multiplexer based lookup table showed below Figure 8. In that diagram a AND gates are used for decoder and pass transistor used for multiplexer logic.

(Va,Ra)=(1,1) (Data1 ,PhaseD) (Vb,Rb)=(1, 1) (Data1 ,PhaseD)

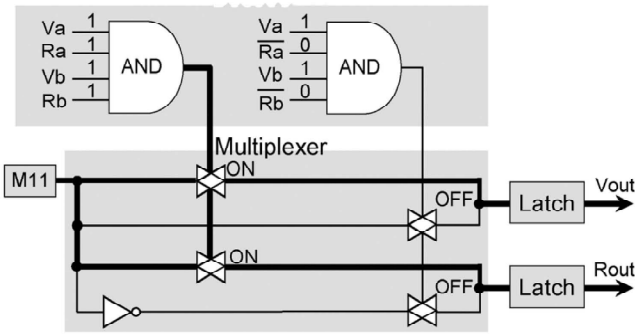

FIG. 8: THE DETAIL STRUCTURE OF LOOK UP TABLE

The previous outputs stored in latch, if input patterns are valid (i.e., if the two inputs have the same phase), according to the corresponding pass-transistors turn ON. The value of the memory bit is selected as outputs; the outputs are stored in the latches. (b) Sleep Controller Design It contains phase comparator, latch and programmable delay design. It is discussed below Figure 9. The sleep controller design designed reference [2].

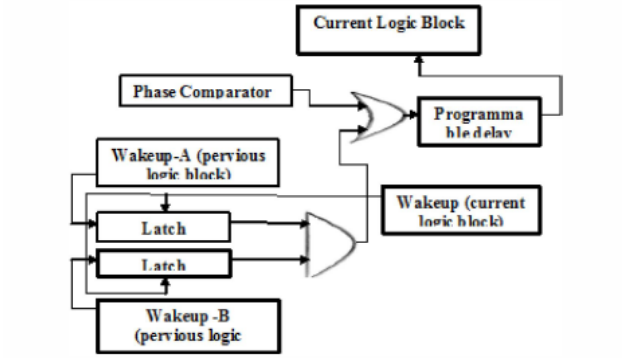

FIG. 9: BLOCK DIAGRAM OF THE SLEEP CONTROLLER DESIGN

The each block in the sleep controller described in below designs.

1. Phase Commparatear Design:

The block diagram of a phase comparator for a two-input and one-output LOGIC BLOCK shown Figure 10. The phase comparator is used to detect the data arrival. Phases of each data are extracted by XOR gates. If PHASE-A and PHASE-B are different from PHASEOUT, it means that all new data has arrived. In that case, the LOGIC BLOCK is active, and the output is ' 1'. Otherwise, it means that some data has not yet arrived and that the LOGIC BLOCK cannot start the operation. In that case, the LOGIC BLOCK is inactive, and the output is '0'.

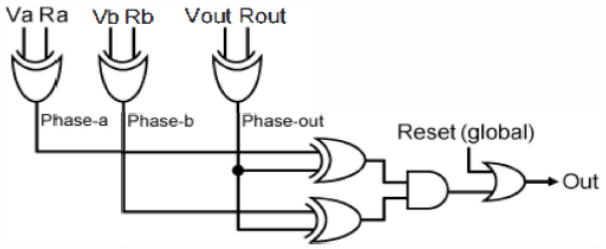

FIG. 10: BLOCK DIAGRAM OF THE PHASE COMMPARATEAR DESIGN

The phase comparator output and pervious logic block wakeup signal given to the programmable delay.

2. Latch Design:

The latch design truth table showed Table 2. If the Wake-up signal once goes to ' 1', the latch retains the signal until all data arrive at the LOGIC BLOCK. When all data arrive at the LOGIC BLOCK and no data arrives at the previous Logic Blocks, the output of the latch is reset to '0'.

TABLE 2: TRUTH TABLE OF THE LATCH FOR THE WAKE-UP SIGNAL

Wake-up	*Data-arrival*	*Out*
0	0	No change
0	1	0
1	0	1
1	1	1

3. Programmable Delay Design:

The function of the programmable delay is to delay the sleep signal by the predetermined threshold time. The programmable delay consists of a series of OR gates and several memory bits. The memory bits are used to program the delay time. The designed reference [3].

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Monthly Peer Reviewed Open Access International e-Journal

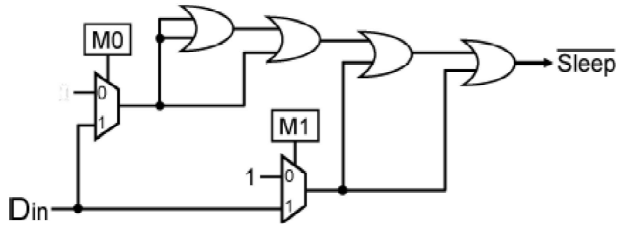

FIG_ 11: BLOCK DIAGRAM OF THE PROPOSED PROGRAMMABLE DELA y DESIGN

TABLE 3: TRUTH TABLE OF THE PROGRAMMABLE DELAY DESIGN:

M0	M1	Normalized threshold time
0	0	Not use power gating
0	1	0.5
1	0	Not defined
1	1	1

Truth table shows the relationship between the memory configuration and the threshold time. Din turns from ‘0’ to , 1’. Since it used as an input of the last OR gate, of ‘ 1’ makes the output of the last OR gate to ‘ 1’ immediately. In powering OFF the LB, Din turns from ‘ I’ to ‘0’. The value ‘0’ of propagates through the series of OR gates so that the sleep signal is delayed. The use of more OR gates and more memory bits make it possible to increase the number of choices of the delay time. The programmable delay does not delay the sleep signal in powering ON.

IV. FPGA WITH FULL ADDER OPERATION

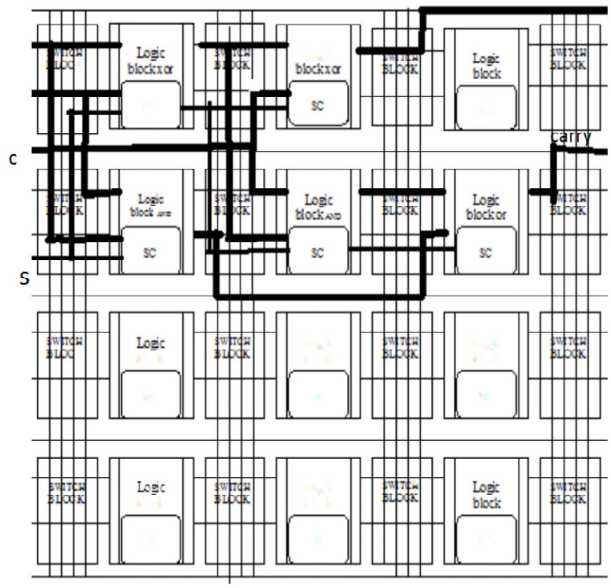

FIG. 12: BLOCK DIAGRAM OF THE LOGIC BLOCK WITH FULL ADDER OPERATION

The full adder circuit having there are two Xor gate and two, gate and one or gate. The inputs are a, b & c, the outputs are sum and carry. The above FPGA IC two perform the full adder operation means to take the five logic gates, on the time the current five logic blocks are active state all other logic blocks are sleep state.

The sleep signals connected to the succeed logic block of sleep controller. The each logic block having own sleep controller, so the logic block arrive at the input the sleep controller also start work and output is given to the next logic block. The next logic
blocks also active state. This method power easily minimized.

V. RESULT AND DISCUSSION:

The developed models can be analyzed for functional correctness using a top-down design methodology and starting from a high level description at the system/algorithm level. The detailed models can be generated by increasing the description details considering the hardware implementation aspects. The RTL (Register Transfer Level) code is expected to provide better model for synthesis.

The functionally correct code, describing the Entities and Architectures, may then be simulated for verification and synthesized into actual hardware. There are various software tools that support design of individual components and then integration into the system to verify the design using simulation.

The synthesis involves analyzing the VHDL code, synthesizing for the target architecture, optimizing subject to design constraints such as placement directives or delay specifications, and generating an optimized FPGA net list.

Placement and routing tools generate an optimal placement subject to delay constraints and then interconnect the logic using the available routing resources on the particular FPGA. The simulation and synthesis of the model has been carryout using and Xilinx ISE foundation series 10. 1 as per the design environment discussed. The designed model has been simulated and synthesized

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Monthly Peer Reviewed Open Access International e-Journal

B.Power Analysis:

The power required to implement the Logic block after incorporating the developed fine-grain power gating LUT using the target device has been computed and listed in Table 4 for individual components.

TABLE 4: POWER REPORT FOR LOGIC BLOCK

Power summary	*I(mA)*	*P(mW)*
Total estimated power consumption		7
Vccint 1.80V	0	0
Vcco33 3.30V	2	7
Clocks	0	0
Inputs	0	0
Logic	0	0
Outputs		
Vcco33	0	0
Signals	0	0
Quiescent Vcco33 3.30V	2	7

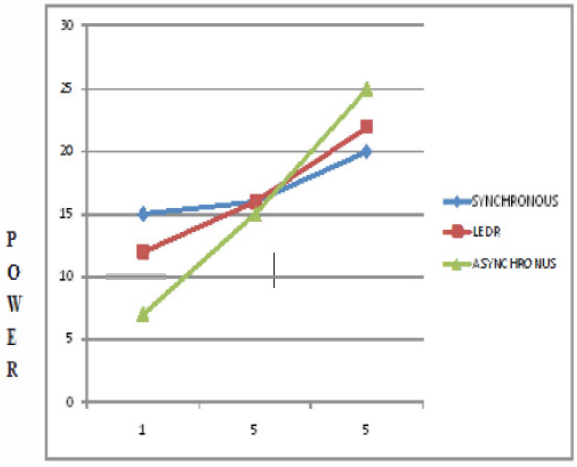

FIG. 15: POWER REPORT FOR SINGLE LOGIC BLOCK AND FULL ADDER LOGICS

The power analysis chart compare the three types power reduction techniques for synchronous FPGA based techniques, simple LEDR encoding FPGA based techniques and asynchronous FPGA based techniques. Compare to existing techniques the proposed techniques is efficient techniques for power gating.

VI. FUTURE WORK:

In this paper the standby power reduction purpose used only for LEDR encoding but both LEDR and dual rail encoding used in same FPGA architecture means again reduce the power and increase the through put since 4-phase dual rail encoding achieves small area and low power for function unit, while LEDR encoding achieves high throughput and low power for data transfer.

VII. CONCLUSION:

In this paper, an asynchronous FPGA architecture based on autonomous fine-grain power gating with small overheads has been developed and analyzed. The implementation of the autonomous fme-grain power gating has been done efficiently using the standby state to wake up the Logic block before the data arrives and power OFF the Logic block only when the data does not come for quite a while. As a result, the wake-up time has been reduced. Since their data paths change dynamically and frequently, it is more difficult than FPGAs to determine the control parameters for each Logic block using offline analysis. The power analysis also has been carried out and it has been found that using the proposed fine-grain power gating method, the FPGA consumes 7 m W powers.

REFERENCES:

[I] Masanori Hariyama, "A Low Power FPGA Based on FineGrain Power Gating Shota Ishihara, Student Member," IEEE Trans. (VLSI) Systems, Vol. 1 9, No. 8, August 20 11 .

[2] J. Teifel and R . Manohar, "An asynchronous dataflow FPGA architecture," IEEE Trans. Computers, vol. 53, no. 11 , pp . 1376-1 392, Nov. 2004.

[3] M. Hariyama, S. Ishihara, C.c. Wei and M. Kameyama, "A field programmable VLSI based on an asynchronous bitserial architecture," in Proc. IEEE Asian Solid-State Circuits Conf. (A- SSCC), Jeju, Korea, Nov. 2007, pp. 380-383.

[4] M.Hariyama, S. Ishihara, and M. Kameyama, "Evaluation of a fieldprogrammable VLSI based on anasynchronous bitserial architecture," IEICE Trans. Electron, vol. E9 1 -C, no.9,pp. 1 4 1 9-1 426,2008.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Monthly Peer Reviewed Open Access International e-Journal

[5] M.Hariyama, S. Ishihara, and M. Kameyama, "A low-power field-programmable VLSI based on a fine-grained powergating scheme," in Proc. IEEE Int. Midw. Symp. Circuits Syst. (MWSCAS), Knoxville, Aug. 2008, pp. 430-433.

[6] S. Ishihara, M. Hariyama, and M. Kameyama, "A low-power FPGA Based on autonomous fine-grain power-gating," in Proc. Asia South Pacific Des. Autom. Conf. (ASP-DAC), Yokohama, Japan, Jan. 2009, pp. 1 1 9-1 20.

[7] A. Youssef, M. Anis, and M. Elmasry, "Dynamic standby prediction for leakage tolerant microproces-sor functional units," in Proc. Ann. IEEE/AC,M Int. Symp. Microarch., Washington, DC, 2006, pp. 37 1