IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

Reconfigured Low Power FPGA Architecture

Poornima.M[*1], Ramamoorthy.P[2]

[*1,2]Department of ECE, SNS College of Technology, Coimbatore, India

poorni2489@yahoo.co.in

Abstract

Field-programmable gate array (FPGA) based on lookup table level fine-grain power gating with small overheads is presented in this paper. A low power Asynchronous FPGA with LEDR encoding and 4-Phase dual Rail Encoding is designed in this paper. The power gating technique implemented in the proposed architecture can directly detect the activity of each look-up-table easily by exploiting features of asynchronous architectures. Moreover, detecting the data arrival in advance prevents the delay increase for waking-up and the power consumption of unnecessary power switching. 4-Phase Dual Rail encoding is to achieve small area and LEDR encoding is to get high throughput and low power. LEDR encoding is done at input and 4-phase dual rail encoding is done at the output, which reduces power. Power gating is done to each Logic Block, which shuts down power to the block which is ideal. Power reduction is achieved by selectively setting the functional units into a low leakage mode when they are inactive. The design is synthesized by using Xilinx and programmed using VHDL language and implemented using Xilinx tool. Since the power gating technique has small overheads, the granularity size of a power-gated domain is as fine as a single two-input and one-output lookup table.

Keywords: Asynchronous architecture, asynchronous field-programmable gate array (FPGA), level-encoded dual-rail (LEDR) encoding.

Introduction

Recently, high-performance VLSIs are required for real-time processing in intelligent systems such as mobile phones, digital cameras, televisions, robots and vehicles. The high performance requirements of these VLSIs cannot be achieved using general purpose processors which provide the flexibility to change applications by software approach. On the other hand, the special purpose processors such as Application-Specific Integrated Circuits (ASICs) can achieve high-performance through the hardware-level optimization for only a single target application. However, as the process technology advances and the circuit scale becomes large, the high mask and design costs of the ASICs become higher and higher. Reconfigurable VLSIs such as Field-Programmable Gate Arrays (FPGAs) are truly revolutionary devices that combine the benefits of the both general purpose processors and ASICs.

Field Programmable Gate Arrays (FPGAs) are widely used to implement special purpose processors. FPGAs are cost effective because interconnections of logic resources can be directly programmed by end users. Despite their design cost advantage, FPGAs impose large dynamic and standby power consumption overheads compared to custom silicon alternatives as in [1].

In FPGAs, the clock network occupies a large proportion of the dynamic power because it has significantly more registers than custom VLSIs. To solve the problem caused by clock, asynchronous FPGAs are proposed. Asynchronous FPGAs have mainly three advantages.

1. Low power consumption because of no dynamic power in inactive circuits.
2. Less emission of Electro Magnetic Interference (EMI) since each Logic Block (LB) operates at random points in time.
3. Robustness because there are automatically delay variations and there is no clock skew.

As the transistor feature sizes and threshold voltages decrease, the standby power due to leakage current becomes comparable to dynamic power. Power gating techniques are based on selectively setting the functional units into a low leakage mode when they are inactive. The power consumption of power gating circuitry is consumed by the sleep controller, the sleep signal distribution network, and the sleep transistors. The fundamental challenge for any power gating technique is to ensure that the saved standby power outweighs the power

overhead of the power gating. Power gating techniques are classified into two types: coarse-grain power gating and fine-grain power gating. In coarse-grain power gating, a large number of lookup tables (LUTs) share a single sleep controller so the area and power overheads of the sleep controller are relatively small. However, if any LUT within a coarse-grain power-gated domain is active, none of the LUTs which share the same sleep transistor can be set to the sleep mode. FPGAs with coarse-grain power gating also causes a large dynamic power and area overhead in the sleep signal distribution network since it is distributed to many LUTs through programmable interconnection resources. On the other hand, in fine-grain power gating, each LUT has its own sleep transistor and related sleep controller, so when any LUTs are inactive, they can be set to the sleep mode immediately. This results in much lower standby power compared to coarse-grain power gating. To reduce dynamic power consumption, we introduce an level-encoded dual-rail (LEDR)-based architecture. To reduce the standby power, a LUT-level power gating technique called autonomous fine-grain power gating is proposed.

Related Work

A. Asynchronous FPGAs

In asynchronous FPGAs, the clock and clock distribution network create several difficult challenges, namely dynamic power consumption and clock skew. References [2] and [3] proposed asynchronous FPGAs based on bundled-data encoding, the most common asynchronous encoding. In this encoding, delay elements are used for the control path. The worst-case minimum delay of delay elements must be larger than the worst-case maximum delay of the data-path. Thus, the use of delay elements limits the throughput. Especially, for FPGAs, since the data path is programmable, complex programmable delay elements are required.

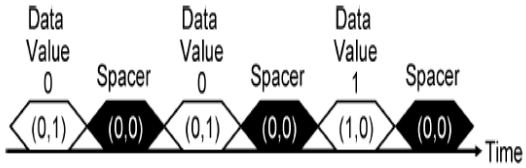

Fig. 1. Example of four-phase dual-rail encoding.

TABLE I
CODE TABLE OF LEDR ENCODING

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

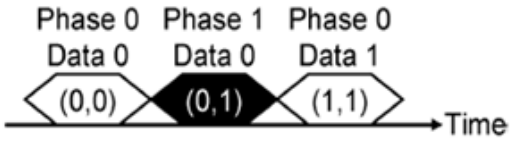

Fig. 2. Example of LEDR encoding.

Reference [4] proposed asynchronous FPGAs based on dual-rail encoding which requires no delay insertion. They use four-phase dual-rail encoding because of relatively small hardware cost. However, as shown in Fig. 1, 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. References [3] proposed asynchronous FPGAs based on LEDR encoding. LEDR is one of several two-phase dual-rail encodings. In LEDR encoding, no spacer is required. Table I shows the code table of LEDR encoding. In LEDR encoding, each data value has two types of code words with different phases. Fig. 2 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 0 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.

Architecture

A. Overview

Architecture of the proposed FPGA which has a mesh-connected cellular array based on a bit-serial architecture is shown in the Fig3. Each logic block (LB) has a sleep controller which controls the sleep transistor of the LUT. As the asynchronous protocol,

we employ LEDR encoding which is suitable for FPGAs. The proposed architecture requires four wires: two for a data, one for acknowledge (ACK), and one for wake-up. The wake-up signal is used to wake up the next LB in advance. Since the next LB has already been woken up before the data arrives, there is no penalty of the wake-up time.

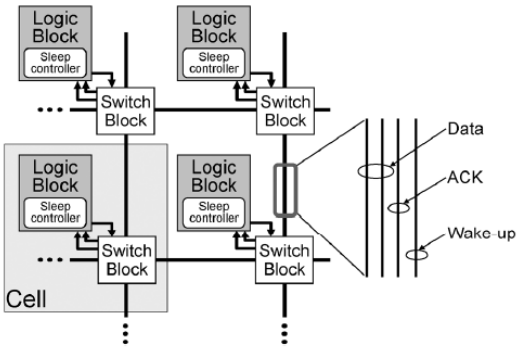

Fig 3. Architecture of FPGA

The switch block shown in Fig 4 consists of pass-switch blocks. In a switch block, a wire-set consists of four wires: two for data (V and R), 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. The four pass switches are used for the four wires of the wire-set, respectively. The pass switches are controlled by the same memory bit.

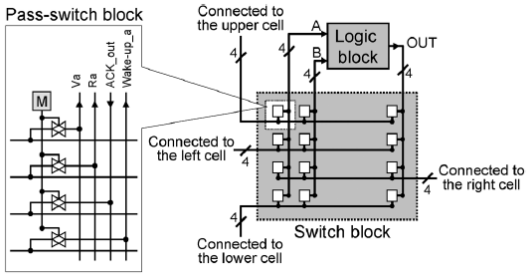

Fig 4 Structure of CELL

B. Logic block

Each LB shown in fig 5 mainly consists of a LUT, an output register, a sleep controller, and a C-element. The LUT operates arbitrary two-input and one output logic functions. The C element is a state-holding element for handshake protocol. The grey region is the sleep controller. The Wake-up signals from previous LBs are used to wake up the LB before the new input data arrives. The Data-arrive signal is used to wake up the next LB when the data arrives. The phase comparator is used to detect the data arrivals. Two latches retain the Wake-up signals from previous LBs until all the input data arrive at the LB.

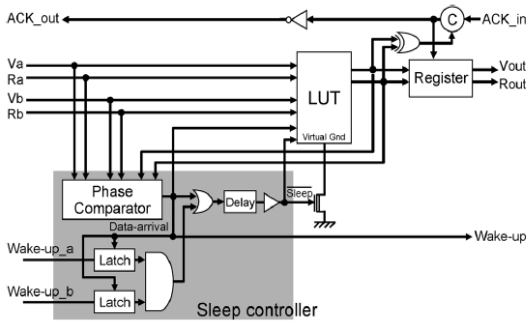

Fig 5 Logic block

In asynchronous architecture activity detection of logic blocks are detected by the principle shown in the Fig 6. The main problem in this principle is logic block operate in one-input and one-output due to this idle state occurs in the logic block.

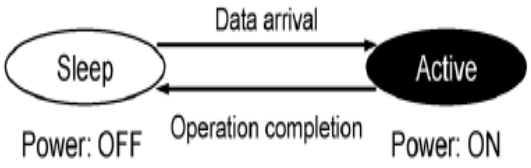

Fig 6 Control strategy of autonomous power gating

To solve this problem, we propose an efficient control strategy of autonomous fine-grain power gating. As shown in Fig.7, the standby state is used to,

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.

The use of the standby state has two major advantages. First, the wake-up time can be hidden since the LB has already been woken up when the data arrivals. Second, the dynamic power can be saved since the number of the unnecessary switching of the sleep transistor is reduced.

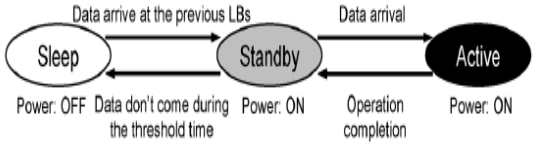

Fig.7 Control strategy of autonomous fine grain power gating

Fig.8 shows an example to explain the proposed power gating method using two LBs: LB1 and LB2 where LB1 is the “previous LB” of LB2. As shown in Fig.8 (a), LB1 and LB2 are respectively in the standby and sleep state as the initial state. To avoid the penalty of the wake-up time, LB1 which is the first LB of the pipeline chain is not powered OFF.

In other words, LB1 is in either the standby state or the active state. As shown in Fig. 8(b), when the new data arrives at "the previous LB" (LB1), a wake-up signal from LB1 is sent to LB2 to wake it up. Then, LB2 turns to the standby state. As shown in Fig. 8(c), when the data arrives at LB2, LB2 turns to the active state. In this state, the operation is executed immediately because the sleep transistor is woken up in the standby state. As shown in Fig. 8(d), LB2 turns to the standby state since the operation of LB2 is complete. As shown in Fig. 8(e), if no data arrives at LB2 during the threshold time, LB2 predicts that the data does not arrive for quite a while. Then, LB2 turns to the sleep state and is powered OFF. The threshold time is determined such that the LB is not powered OFF in a busy condition where data arrives frequently. The LB is woken up before the data arrival and powered OFF only while the LB is idle.

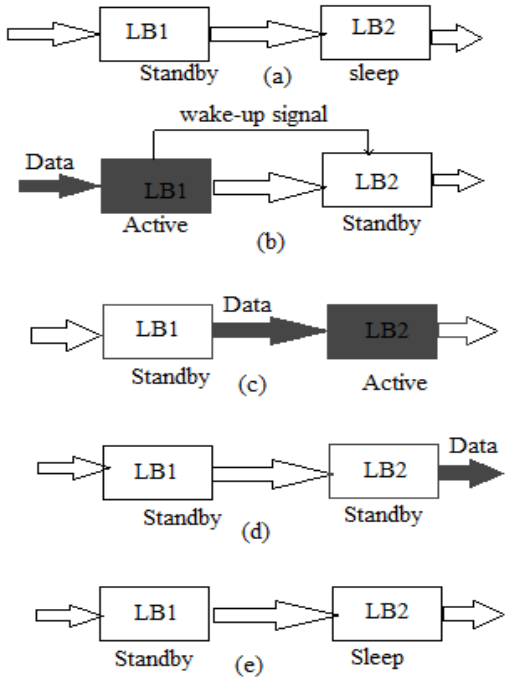

Fig 8 Example of fine grain power gating

C. Phasse Comparator

The phase comparator shown in fig 9 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 Phase-out, it means that all new data has arrived. In that case, the LB is active, and the output is "1." Otherwise, it means that some data has not yet arrived and that the LB cannot start the operation. In that case, the LB is inactive, and the output is "0."

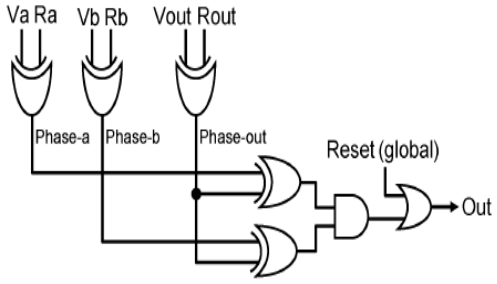

Fig 9 Phase comparator

D. Look Up Table

LUT based on a hybrid of decoders and multiplexers is extended to power gating to reduce the number of multiplexers for correct output from invalid combination of input. Fig.10 shows the block diagram of LUT, which consists of four sub-modules. Each sub-module consists of a decoder, a multiplexer and a memory bit. The decoders exclude invalid input patterns with different phases. Then, only valid data are fed to the multiplexer. As a result, the numbers of multiplexers are reduced. In order to prevent indefinite values from occurring, the sleep signal is input to the decoders. In the sleep mode, all pass-transistors turn OFF according to the outputs of the decoders; the outputs of multiplexers are in the high impedance condition; the latches keep the previous operation result.

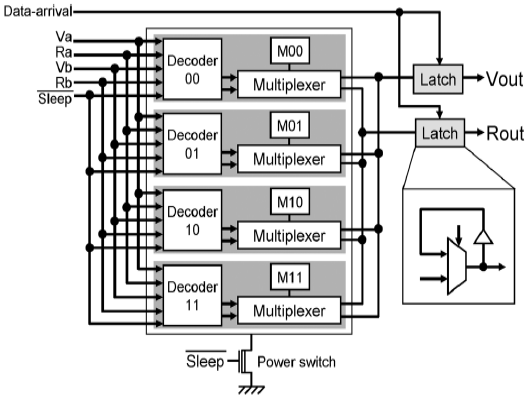

Fig 10 Block diagram of LUT

Proposed Structure

When applying the proposed fine-grain power gating method to complex data path architectures, the major problem is that the area of an LEDR based LUT increases rapidly as the number of inputs increases although the LEDR encoding is useful to enhance the throughput and reduce the power consumption of switch blocks. To solve this problem, four-phase dual-rail encoding can be efficiently combined with the LEDR encoding. Four

phase dual-rail encoding is suitable for a LUT because of its small area, while LEDR encoding is suitable for switch block because of its high throughput and low power.

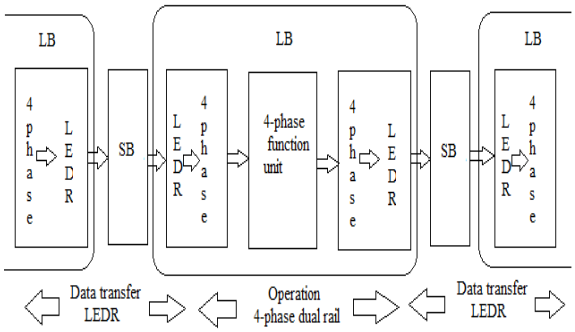

Fig.11 Proposed logic block

As 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, based on this observation, the proposed FPGA combines 4-phase dual rail encoding for implementing function units and LEDR encoding for the data transfer using programmable interconnection resources as shown in the Fig 11.

Simulation Results

A. XILINX ENVIRONMENT

Fig 12 shows the RTL Schematic of Fine grain FPGA with number of cells. Each cell has logic block, which contains sleep controller and switch block. Each cell is connected with four wires: two for data, one for acknowledge (ACK) and one for wake up.

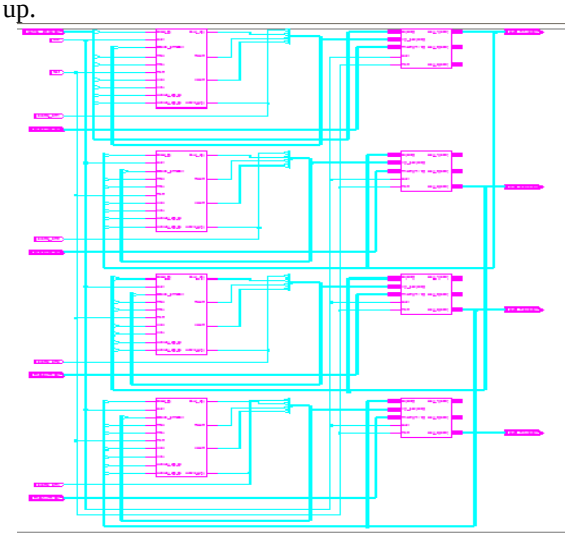

Fig 12 RTL Schematic of Fine grain FPGA

B.MODELSIM ENVIRONMENT

The design has been simulated in MODELSIM for the further study of the design.

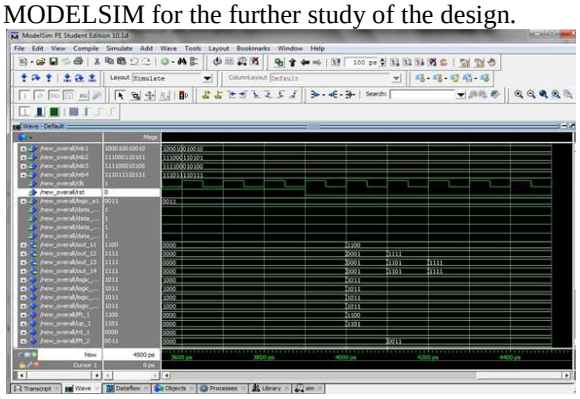

Fig 13(a) Input and output waveform for Fine Grain FPGA

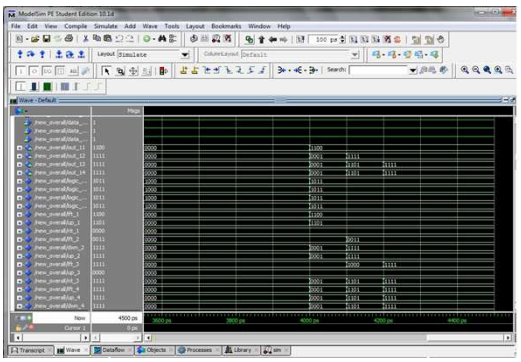

Fig 13(b) Input and output waveform for fine grain FPGA

The above Fig.13 (a) & (b) show the input and output waveform of LUT. The input of the logic block is directly connected to the output of another logic block which result in complexity of interconnection network between the logic block is reduced.

Conclusion and Future Enhancements

In asynchronous architecture, the activity of an LB is easily detected only by comparing the phases of the input and the output data. To implement the autonomous fine grain power gating efficiently, the standby state is used to wake up the LB before the data arrives and power OFF the LB only when the data doesn't come for quite a while. As a result, the wake-up time can be hidden and the dynamic power of unnecessary switching of the sleep transistor can be saved.

When applying the proposed fine-grain power gating method to complex data path architectures, the major problem is that the area of an LEDR based LUT increases rapidly as the number of inputs increases although the LEDR encoding is useful to enhance the throughput and reduce the

power consumption of switch blocks. To solve this problem, four-phase dual-rail encoding can be efficiently combined with the LEDR encoding in future work. Four phase dual-rail encoding is suitable for a LUT because of its small area, while LEDR encoding is suitable for switch block because of its high throughput and low power.

References

[1] George.H. Z. V and Rabaey. J(1999), "The design of a low energy FPGA,"in Proc. Int. Symp. Low Power Electron. Des., CA, Aug., pp.188–193.

[2] K. Maheswaran and V. Akella, "PGA-STC: Programmable gate array for implementing self-timed circuits," *Int. J. Electron.*, vol. 84, no. 3, pp. 255–267, 1998.

[3] R. Payne, "Self-timed FPGA systems," in *Proc. Int. Workshop Field Program. Logic Appl.*, 1995, pp. 21–35.

[4] T. Tuan, S. Kao, A. Rahman, S. Das, and S. Trimberger, "A 90 nm low-power FPGA for battery-powered applications," in Proc. FPGA, Feb. 2006, pp. 22–24.

[5] Ishihara.S, Hariyama.M, and Kameyama. M, "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. 119–120.

[6] Teifel. J and Manohar. R, "An asynchronous dataflow FPGA architecture," IEEE Trans. Computers, vol. 53, no. 11, pp. 1376–1392, Nov.2004.

[7] Ahmed. Y, Mohab. A, and Mohamed. E, "A comparative study between static and dynamic sleep signal generation techniques for leakage tolerant designs," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 16, no. 9, pp. 1114–1126, Sep. 2008.

[8] Mui. M, Banerjee. M, and Mehrotra. A, "Supply and power optimization in leakage-dominant technologies," IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst., vol. 24, no. 9, pp. 1362–1371, Sep. 2005.

[9] Xilinx Inc., San Jose, CA, "Synthesis and simulation design guide,"2008.[Online].Available http://www.xilinx.com/itp/xilinx10/books/docs/im/sim.pdf

[10] Hu.Z, Buyuktosunoglu.A, Srinivasan.V, Zyuban.V, H. Jacobson, and P. Bose, "Microarchitectural techniques for power gating of execution units," in Proc. Int. Symp. Low Power Electron. Des., 2004, pp. 32–37.

[11] Youssef. A, Anis.M, and Elmasry.M, "Dynamic standby prediction for leakage tolerant microprocessor functional units," in Proc. Ann. IEEE/ACM Int. Symp. Microarch., Washington, DC, 2006, pp. 371–384.