Computing, Information Systems
& Development Informatics Journal

Volume 3. No. 3. July, 2012

A Self Learning based Diagnosis of Faulty Configurable Logic Blocks (CLBs) in Field Programmable Gate Arrays (FPGA) Using Reconfiguration

B.K.V. Prasad
Dept. of ECE, JNTUH
Hyderabad, AP, India
Kali_197i@kluniversity.in

P. SatishKumar
Dept. of ECE, MLRIT
Hyderabad, AP, India

B. S. Charles
Dept. of ECE, SSCET
Kurnool, AP, India

T.Madhu
Dept. of ECE, SIET
Narsapur, A.P, India

Mohammad Shafivullah Khan
Associate Professor,
Dept. of ECE,
KL University
Vaddeswaram, A. P, India
khanklu@hotmail.com

Reference Format: Prasad. B.K.V., SatishKumar, P., Charles, B.S., Madhu, T. & Mohammad Shafivullah Khan. (2012). A Self Learning based Diagnosis of Faulty Configurable Logic Blocks (CLBs) in Field Programmable Gate Arrays (FPGA) Using Reconfiguration. Computing, Information Systems & Development Informatics Journal. Vol 3, No.3. pp 1-8

A Self Learning based Diagnosis of Faulty Configurable Logic Blocks (CLBs) in Field Programmable Gate Arrays (FPGA) Using Reconfiguration

Prasad, B.K.V., Satishkumar, P., Charles, B.S., Madhu, T. & Mohammad Shafivullah Khan

ABSTRACT

In many areas of digital systems Field programmable gate arrays (FPGAs) are most important for designing. The main uses of FPGAs are, these are programmable, and faults can be easily diagnosed, once faulty locations are identified. The location and identification of faults in FPGA has not yet been explored much. A methodology for the testing and diagnosis of faults in FPGAs is presented based on automatic circuit reconfiguration. The proposed method imposes no hardware overhead. This method can also be used in fault-tolerant systems, in which a good functional circuit can be still mapped to a FPGA with faulty elements, as long as the fault sites are known. The logic synthesis software assigns the Configurable Logic Block (CLB) resources without system designer intervention. It is very advantageous for the designer to understand certain CLB details, including the varying capabilities of the look-up tables (LUTs), the physical direction of the carry propagation, the number and distribution of the available flip-flops. FPGA consists of 25 Configurable Logic Blocks (CLB). Each CLB is assigned with an application. The inputs for CLB are applied from a file. There is also a fault file in which error CLBs are present. If there is error CLBs, those CLBs are replaced by the spare CLBs. Finally, the errors CLBs are corrected with proper inputs and modified bits are displayed. So efficiency is not reduced and configurability is done without replacing the faulty components. This FPGA can tolerate not only single faults but also for multiple faults. The power analysis results provided for fault free, stuck-at-1, stuck-at-0 faults in digital circuits validate the point that faulty circuits dissipates more and hence draw more power.

Key words: Configurable Logic Block (CLB), Power Dissipation, Fault Tolerance, Fault Diagnosis, Faults, Full adder (FA).

1. INTRODUCTION

In recent years, there has been a rapid increase in the use of computer-based systems in areas such as railway traffic control, air traffic control and telecommunications. This trend has led to a growing interest in the validation of fault tolerance properties and evaluation of their reliability .Further the continuous increase in the integration level of electronic systems is making increasingly difficult to guarantee an acceptable degree of reliability. Fault diagnosis includes both fault detection and fault location. Identifying the fault location is an important step in FPGA to increase its performance.

In FPGA, a portion of the device resources is set aside to perform the fault handling. A few system resources are consumed than with an external fault monitor. An internal fault monitor can also run on FPGA that support partial configuration. Such a fault monitor could run continuously without disturbing device functionality. This internal fault monitor conserves limited-pin resources and avoids the relatively slow process of transferring information off-chip through the pins as shown in figure.1. Testing faults in general FPGAs has been proposed by many researchers. In these methods, the FPGA under test is not mapped to a specific logic function. This approach allows for rapid detection of both hard and soft faults. The resources needed to perform fault testing can be kept to minimum by using fault-scanning methodology.

Only a small section of FPGA is tested at a time, but testing can scan across the FPGA assuring that the entire FPGA will be tested eventually. In FPGA, faults mainly occur in interconnects contain horizontal and vertical single and double length lines intersecting at a box called programmable switch box. Each switch matrix consists of programmable pass transistors and is used to establish connection between the lines. All the interconnections are composed of metal segments with programmable switch points and matrices to implement the desired routing. Field programmable gate arrays (FPGAs) can be programmed in the field to implement any logic circuit. They are widely used in rapid system prototyping because of their reprogram ability, and they have also been used in many practical circuits.

An FPGA usually consists of an array of identical configurable logic blocks (CLBs), programmable I/O blocks (IOBs), and programmable interconnects. Many methods have been proposed to test FPGAs. In some works, the circuits under consideration are programmed FPGAs, in which logic circuits have been implemented**.** Since an FPGA can be programmed in many different ways, this method is not applicable to manufacturing time testing, as we do not know the final configuration. Testing faults in general FPGAs has been proposed by many researchers.

In these methods, the FPGA under test is not mapped to a specific logic function. As a result, multiple test sessions are usually required, with each session dealing with one configuration. Traditional chip-level testing usually deals with fault detection only, while fault diagnosis is often conducted at the system level. This is because components in the chip cannot be repaired. However, faults in FPGAs can be easily tolerated by not including faulty elements in the final circuit. Therefore, FPGA chips with faults can still be used if we can identify the fault sites. In this paper, we propose a reconfiguration of configurable logic blocks methodology for faults in CLBs.

Our method is also based on the automatic circuit reconfiguration technique, which means that the testing process is conducted by replacing the faulty CLB with spare CLB but a nearby one CLB and the requirement for external ATE support is limited. The testing time is affected by the number of faults on the chip only, and is independent of the chip size. An application circuit must be mapped into an FPGA with adequate resources. While the number of CLBs/LABs and I/Os required is easily determined from the design, the number of routing tracks needed may vary considerably even among designs with the same amount of logic.

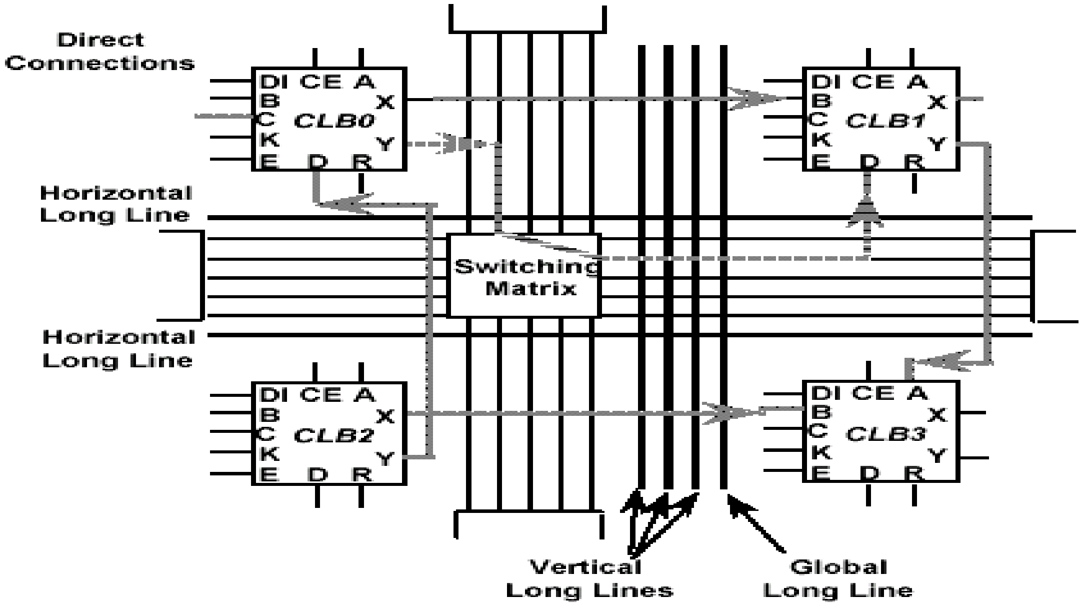


Fig 1. FPGA Interconnect

2. FPGA ARCHITECTURE

The most common FPGA architecture-consists of an array of logic blocks (called Configurable Logic Block, CLB, or Logic Array Block, LAB, depending on vendor), I/O pads, and routing channels. Generally, all the routing channels have the same width (number of wires). Multiple I/O pads may fit into the height of one row or the width of one column in the array. In general, a logic block (CLB or LAB) consists of a few logical cells (called ALM, LE, Slice etc). A typical cell consists of a 4-input lookup table (LUT), a full-adder (FA) and a D-type flip-flop, as shown in Figure.2.. The LUTs are split into two 3-input LUTs. In normal mode those are combined into a 4-input LUT through the left mux. In arithmetic mode, their outputs are fed to the FA. The selection of mode is programmed into the middle mux. The output can be either synchronous or asynchronous, depending on the programming of the mux to the right. In practice, entire or parts of the FA are put as functions into the LUTs in order to save space

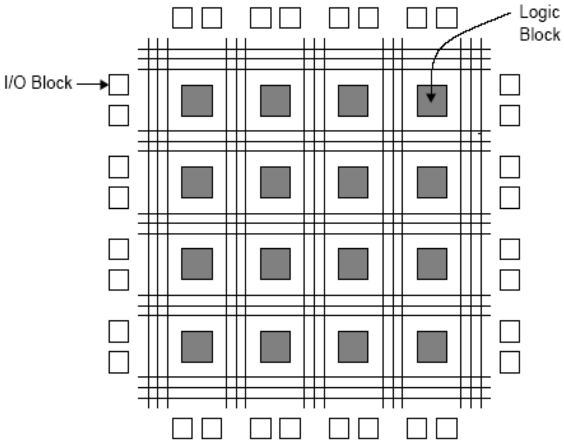


Fig 2. Structure of an FPGA with a logic cell

3. FAULTS IN DIGITAL CIRCUITS

A. Failure and Faults

Failure is said to have occurred in a circuit or system if it deviates from its specified behaviour. Fault on the other hand is a physical defect which may or may not cause a failure. The nature of fault can be classified as logical and non- logical. A logical fault causes the logic value at point in a circuit to become opposite to the specified value. Non logical faults include the rest of faults such of malfunction of clock signal, power failure.

B. Modeling of faults

Faults in circuits occur due to defective components, break in signal lines, lines shorted to ground or power supply short circuiting of signal lines, excessive delays etc. Besides errors or ambiguities in design specification design rule violations etc also results in faults. In general the effect of a fault is represented by means of model, which represents the change the fault produces in circuit signals. The fault models are stuck-at–fault, Stuck-open –fault, Bridging fault.

C. Stuck-at-fault

The most common model used for logical faults is the "single-stuck-at fault" it assumes that a fault in a logic gate results in one of its inputs or the output being fixed referred to as "classical fault model". The "suck-at" model is also used to represents multiple faults in circuits. In multiple "stuck-at fault" it is assumed that more than one signal line in the circuit are stuck-at logic'1' or logic '0'. In other words a group of stuck-at faults exit in the circuit at the same time. A multiple faults is unidirectional if all its constituent faults or either stuck-at-0 or stuck-at-1, but not both simultaneously.

D. Bridging Fault

Bridging faults form an important class of permanent fault which cannot be modeled as stuck-at faults. A bridging fault occurs when two leads in a logic network are connected accidentally and wired- logic is performed at the connection. Depending on whether positive or negative logic is being used the faults have affected respectively ANDing or ORing the signals involved. Fabrication of an FPGA is a complicated which in turn may introduce a fault during fabrication may prevent a transistor from working and may break or join interconnections. Two common types of defects occur in metallization either, under-etching the metal, this arises between long closely spaced lines, which results in bridge or short circuit between adjacent lines or over-etching the metal and causing breaks or open circuits. So it is quite obvious that faults are likely to occupy an area in interconnects as larger as the logic resources.

E. Stuck-open–fault

A certain type of logic fault associated with digital circuits depend on the particular semiconductor technology chosen to realize the circuits stuck-open (S-op) faults are peculiarly in CMOS digital integrated circuits. They are not equivalent to classical stuck-at-faults. The major difference between the classical and s-op faults is that the former leave the faulty gate as a combinational circuit, but the S-op faults turn it in to a sequential circuit. Stuck–open fault causes the output to be connected neither to VSS nor to VDD. In FPGA, the nets in the routing resources consist of two parts. The connection nets that run between the logic. The second is Resources and the control nets that control the switches between the interconnections nets. Each connection net depends on the setting of the control nets. A test generation algorithm for FPGA must generate not only the test vectors for the connection nets, but also the configuration date for the controlling switches and controlling nets. Interconnects are usually an analyzed as a set of nets with different faults such as stuck-at, bridge and stuck-open. Fault detection means the discovery of something wrong in digital circuits or circuits or interconnects lines or in controlling switches and controlling nets. Fault location means the identification of fault with components, interconnecting lines or in controlling switches or controlling nets. Fault diagnosis includes both fault detection and fault location. The fault detection is carried out by applying a sequence test inputs and observing resultant outputs at response analyzer.

4. AUTOMATIC CIRCUIT RECONFIGURATION

An FPGA in which there are 25 Configuration Logic Blocks (CLB).Each CLB is applied with an application such that 16 CLBS are fitted with an application. The remaining CLBs are spare CLBs. The input to the CLB is applied from a file. There is also a fault file in which error CLBs are present and the fault can be single fault or multiple fault. If there is error CLBs those CLBs are replaced by the spare CLBs. Finally the error CLBs are corrected with proper inputs and modified bits are displayed. Reconfiguration time is not wasted to get a new configuration and accuracy is also not reduced.

5. BLOCK DIAGRAM OF CLB

The block diagram of the CLB is shown in figure.3.The description of each block is given in this section

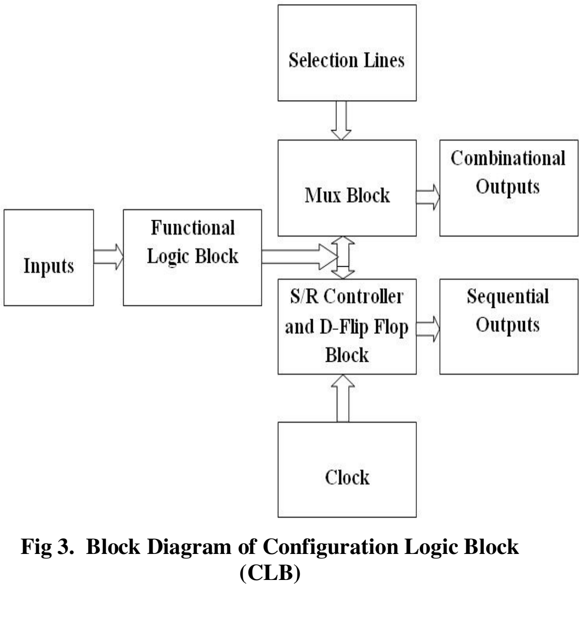


Fig 3. Block Diagram of Configuration Logic Block (CLB)

A. Functional Logic Block

In this block we specify which type of function has to configure into configurable logic block. We use DSCH to design CLB and we assign AND function in configurable logic block .In figure 4 three function logic blocks are present, represented as F, G and H. We assign four input AND function for both F and G function logic blocks and three input AND function for H function logic block.

B. MUX block

In this block we use several types of 2:1 and 4:1 multiplexers. This multiplexer drive the configurable logic blocks and helps to give both combinational and sequential outputs. These MUXes gives an output based upon their selection lines.

C. S/R and D-FF block

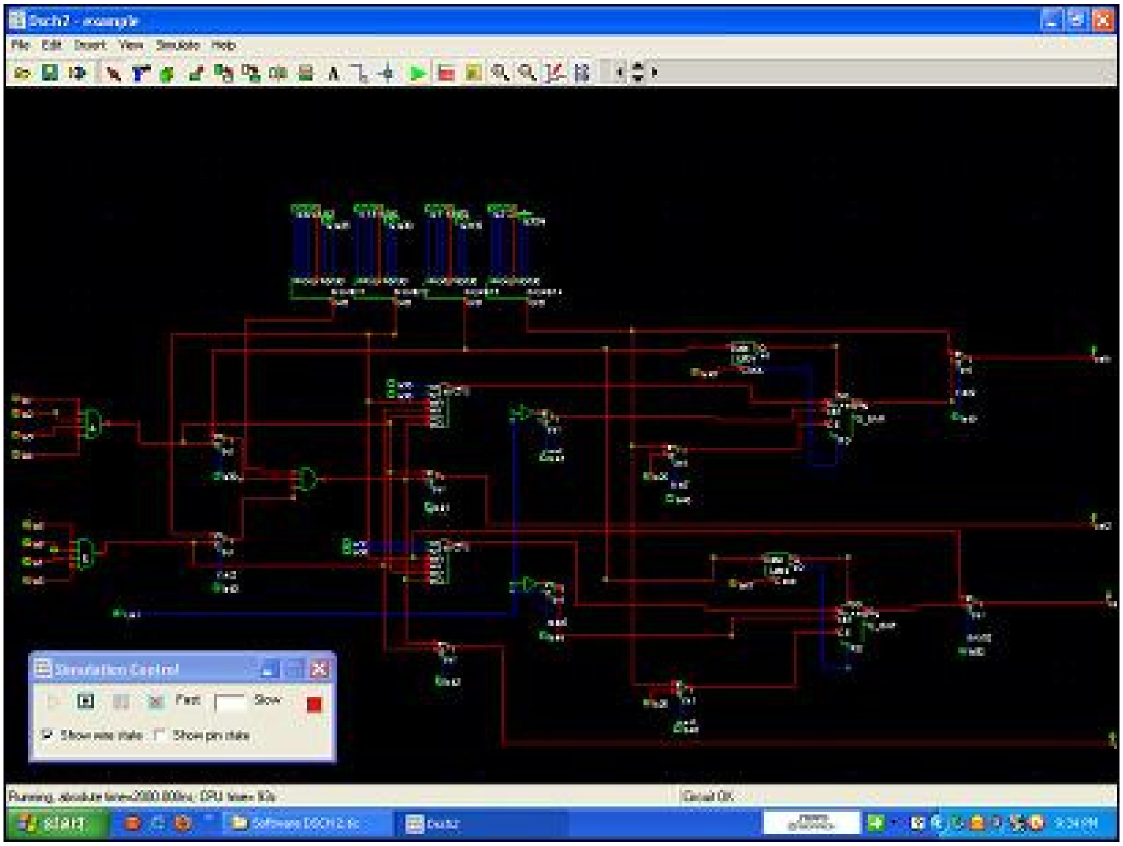
In this block instead of S/R controller we use D- latch for set and reset operation for D flip flop. In D flip flop block we use FDCPE- D Flip-Flop with Clock Enable and Asynchronous Preset and Clear. This S/R controller and D-Flip flop gives sequential outputs. Clock is essential for this block so we give asynchronous clock to this block.

D. Outputs

In CLB circuit there are four outputs, two are sequential and two are combinational. Sequential inputs, S/R controller and D-Flip flop drive the sequential output and functional logic block and multiplexers drives the combinational outputs.

E. Testing a CLB

A simplified diagram of figure 4 is the CLB by name XC4000 Family. This CLB has 13 inputs (I= 13), in which one is the clock input (K), nine signals are input to the combinational part (F1 to F4, G1 to G4, and Hl), and the other three are for the sequential part (DIN, S/R, and EC). There are four outputs in a CLB (0 to 3), in which two are combinational outputs and the other two are for sequential circuits. The combinational part consists of three look-up tables (LUTs), and three multiplexers (MUXs), whose outputs are H1, X, and Y, respectively. The sequential part is made up of the remaining components: two D flip-flops (F = 2), the SIR control, and the remaining MUXs. The LUTs consist of any logic gates. To test the gates, each bit has to be set to both 0 and 1. Therefore, at least two phases are required to exercise all possible faults in the LUTs. There are 4-to-1 MUXs in a CLB. As a result, we need at least four test phases so that each input to- output connection of these MUXs can be exercised.


Dsch2 - example
File Edit Insert View Simulate Help
Simulation Control
Fast
Slow
☑ Show wire state □ Show pin state
start
Software DSCH2.lit
Dsch2

Running, absolute time=2080.000ns, CPU time 10s

Circuit OK

Fig 4. XC4000 Configurable Logic Block