Configurable BISR Chain For Fast Repair Data Loading

Wei Zou
Siemens Digital Industries Software
Wilsonville, Oregon, United States
zou@siemens.com

Benoit Nadeau-Dostie
Siemens Digital Industries Software
Ottawa, Ontario, Canada
benoit.nadeau-dostie@siemens.com

***Abstract*—Embedded memories take a significant portion of current system-on-chip (SOC) designs. Built-in self-repair (BISR) is widely used to improve the design yield by replacing the faulty elements of a memory with spare ones. BISR programs the repair data obtained by a memory built-in self-test (BIST) circuit to a fuse box and transfers the repair data to the memory when the chip is powered up. However, loading the repair data to the memories using a serial scan chain is very time consuming due to the increased number of memories. To reduce the long repair data loading time, we propose a configurable BISR chain repair system. In the proposed solution, the serial scan chain is partitioned into an optimal number of segments. Only the segments connected to defective memories are scanned. Experimental results show that the proposed method can significantly reduce the repair data loading time compared to two baseline methods.**

Keywords—BISR, BIRA, BIST, Embedded Memory, Yield

I. INTRODUCTION

Current high-density system-on-chip (SOC) designs include a large number of embedded memories. The number of memories can be over thousands and even tens of thousands in some applications. For example, in [1], a Graphcore MK2 IPU has 1472 powerful processor cores, and each core has an unprecedented 900MB memory. Designed tightly to the technology limits, memories are more prone to failures than other circuits, which can affect not only manufacture yield but also circuit reliability adversely. Built-in self-test (BIST) techniques [2] are typically employed to identify defects and problems in the memories. Moreover, a circuit having embedded memories usually includes built-in self-repair (BISR) circuitry [3] for performing a repair analysis (built-in repair analysis or BIRA) and for replacing faulty elements with spare ones. Repair information obtained by the built-in self-repair (BISR) circuitry can be stored in a non-volatile storage device such as a fuse box. When the circuit is powered up, the repair information can be retrieved and loaded. The transportation of the repair information between the memories and the fuse box involves a scan network, which is often referred to as a BISR chain. Data delivery via a single scan network is serial in nature. A circuit can have hundreds or thousands of memories. Using a conventional scan network can take too long time for manufacture and in-system test and repair.

To reduce the repair data loading time when a chip is powered up, two methods were proposed in [4] and [5]. In [4], the failure information coming from different memories or segments of memories tested is merged to calculate a common repair solution which can be broadcasted. Although the proposed method can reduce both the repair hardware overhead and the repair data loading time, it is difficult to achieve more than an order of magnitude due to routing issues. In [5], bypass logic is added to exclude a memory from the BISR chain if a memory is fault free and doesn't need a repair. There is one data bit which controls the bypass logic. If the data bit is '0', the memory will be included in the BISR chain, otherwise it will be excluded from the BISR chain. All the data bits controlling the bypass logic are concatenated into a scan chain dedicated for the chain configuration data shifting. The chain configuration data is programmed to the fuse box during the repair data programming, and it is loaded to the configuration chain before the repair data loading. A similar method is also proposed in [6]. For a mature process, only a very small number of memories need repair. Most of the memories are bypassed during the repair data loading. The repair data loading time can only be reduced by an order of magnitude with this method as the majority of memories only use one or two spares and the length of each repair register associated to a memory is about 10 bits on average. In this paper, we propose a new method which divides the BISR chain into an optimal number of segments and each segment contains a number of repair registers which may or may not be shared by several memories. A segment selection circuit is added to each segment to exclude the segment from the BISR chain if the memories of the segment are fault free. Different from the method of [5], the data bits controlling the segment selection circuits are part of the BISR chain, which reduces the routing overhead by not introducing a new scan chain. The proposed method can further reduce the repair data loading time compared to the method in [5].

The rest of this paper is organized as follows. Section II describes the general repair system widely used in the industry and the prior work on fast repair data loading. Section III describes the proposed configurable BISR chain repair system. Section IV summarizes experimental results. Section V concludes this paper.

II. THE GENERIC REPAIR SYSTEM AND PRIOR WORK OF FAST REPAIR DATA LOADING

The generic repair system widely used in the industry is shown in Fig. 1. For illustration purpose and easy understanding, we only show six memories which are tested by two memory BIST controllers. For each memory, there is a BISR register connected to it. The data stored in the BISR register is used for memory repair. The size of the BISR register is determined by the memory size and the number of spare elements that the memory has. Assume all the memories in Fig. 1 have 128 rows and only one spare row for repair. Any failing row detected during the test can be replaced with the spare row during the repair. For this memory, the BISR

2022 IEEE International Test Conference (ITC) | 978-1-6654-6270-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/ITC50671.2022.00012

2378-2250/22/$31.00 ©2022 IEEE
DOI 10.1109/ITC50671.2022.00012

Regular Paper