Adapting an Industrial Memory BIST solution for testing CAMs

Jais Abraham, Uttam Garg
Intel, Bangalore, India

Glenn Colon-Bonet,
Intel, Fort Collins, USA

Ramesh Sharma, Chennian Di
Intel, Santa Clara, USA

{jais.abraham, uttam.k.garg, glenn.t.colon-bonet, ramesh.sharma, chennain.di}@intel.com

Benoit Nadeau-Dostie, Etienne Racine
Mentor Graphics, Canada

Martin Keim
Mentor Graphics, USA

{benoit_nadeau-dostie, etienne_racine, martin_keim}@mentor.com

Abstract

Content Addressable Memories (CAMs) have found widespread use in applications that require high speed search capabilities. Each cell in the CAM array is associated with a storage unit and a comparator logic. Due to the various customized features in the CAM implementations, creation of an automated BIST solution for testing them has presented unique challenges. This paper shows that, with suitable modifications to the CAM test collar, an existing BIST solution and flow traditionally used to test embedded SRAMs can be used to test the CAMs.

Keywords: Binary CAM, Ternary CAM, BIST

1. Introduction

Content Addressable Memories (CAMs) have been employed widely for high speed search operations in applications such as networking and cache look up [1][2][3][4][5][6]. Compared to RAMs, which return the data based on the address provided to it, the CAMs compare the incoming data with the keys present in the array and return the address(es) of the matching entries in the array. A "hit" or "match" is said to occur when the incoming compare data can be found in the CAM array. Thus, the CAM cells have a comparator logic in addition to the storage element. CAMs are usually followed by a priority encoder and a payload RAM. The priority encoder resolves multiple matching entries in the array to the highest (or lowest) matching address and in many high speed implementations, the match lines serve as fully decoded address to the payload RAM.

CAMs are usually classified as Binary CAMs (BCAM) and Ternary CAMs (TCAM). Quaternary CAMs have recently been reported [7]. BCAMs generate a hit if the complete entry is matched. On the other hand, the Ternary CAMs (TCAM) provide the capability to match a part of the entry, while treating the remaining part as don't cares, which are specified using additional mask bits. TCAMs need an extra storage array to store the mask bits. In addition, different TCAM implementations provide the capability to mask specific bits of an entry or parts of the entries or the complete entry itself.

Several CAM BIST solutions have been presented in the past. Representative examples can be found in [8-11]. In our experience, CAM designs often include new features or features that are implemented differently in a way that would affect the BIST implementation and its automation. The contribution of this work is to demonstrate how an existing memory BIST solution for embedded SRAMs can be used to test CAMs in that context. Although the solution in this paper describes a particular CAM design, it will become clear that the solution can easily be extended to other implementations as well. The paper explains two exemplary CAM algorithms, providing insight into how these algorithms differ from those for SRAMs. In practice, variations of these algorithms should be considered, based on the presence or absence of specific CAM features. However, the treatment on the algorithms is not exhaustive and other works on CAM test [8-11] can be referred for this purpose. The only requirements for the existing BIST tool is to have a powerful algorithm programming capability used in combination with flexible operation definition of user-defined control signals.

The paper is organized as follows. Section 2 discusses the architecture of CAMs and provides a description of the generic ports that are available on the CAMs. Section 3 provides an overview of the fault models considered for testing the CAM comparator logic and the algorithms used to test them. Section 4 discusses the details of an example custom CAM collar. Section 5 discusses the differences in the design flow compared to the one used for RAM BIST. Section 6 concludes the paper.

2. CAM Architecture

The logical view of a Binary CAM array is shown in Fig. 1. The Binary CAM can be conceptualized as having two functions: storage and compare. The storage function behaves like a RAM and needs to be tested for the faults in the address decoder, storage cells and the read/ write circuitry. Associated with every bit is also a comparison logic. The result of the comparison is calculated across the complete entry to generate a match/hit.

Authorized licensed use limited to: Siemens AG GBS B&S SOL GMS. Downloaded on September 13,2024 at 14:01:21 UTC from IEEE Xplore. Restrictions apply.