

MAGMA DESIGN AUTOMATION

Failure Analysis Using Camelot K-Bitmap



Application
Overview

 **MAGMA**®



Copyright © 2011 Magma Design Automation, Inc.
All rights reserved.

Magma is a registered trademark and Camelot is a trademark of Magma Design Automation. All other product and company names are trademarks or registered trademarks of their respective companies.

Magma Design Automation

1650 Technology Drive
San Jose, CA 95110

Tel 408-565-7500

Fax 408-565-7501

www.magma-da.com

Introduction

This document introduces the Camelot™ K-Bitmap tool and discusses the need, benefits and system requirements of the tool.

Converting memory bit failures from logical to physical addresses is a time-consuming and tedious process, which becomes more complex with challenging designs. Failure Analysis (FA) teams manually manipulate the data to create inline-defect-data-scan-overlays for failed bit patterns. But again, this is a time-consuming process. Compounding this problem, manual FA on these memory devices is becoming prohibitive due to increasing metal levels and decreasing geometries. Manual calculation of X and Y coordinates of bit cells is obsolete and ineffective. Therefore an automated method is required to translate logical bit failures to physical bit failures and automatically drive FA tools to the X and Y coordinates.

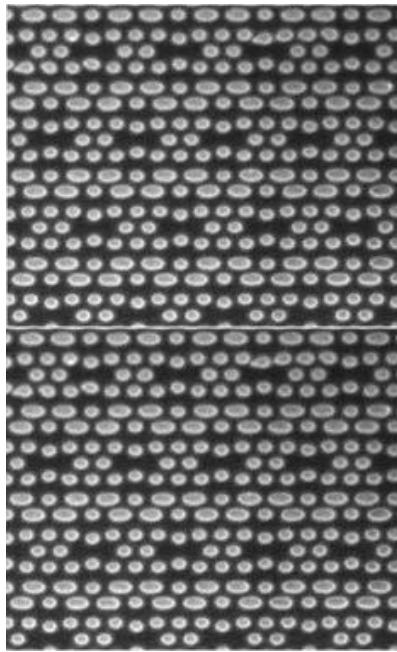


Figure 1. View of Memory Bitmap Cells

The Solution

To solve this problem, an approach is required that can automate the test flow to isolate defect localization within memory devices. This approach requires a quick and easy-to-use setup that comprehends multiple devices that can come online very quickly. The solution should allow sharing of information across functional groups, so that this expertise does not reside with any one person. The solution should leverage the skills of many engineers who work together as a cohesive group to perform root cause analysis. Additionally, the solution should provide a seamless approach that directly communicates the results to the automated stages of any FA tool to automate fault localization and verification.

The Camelot K-Bitmap Tool

The Camelot K-Bitmap tool provides an integrated solution. Figure 2 displays how to set up and create a memory specification file. This file contains the size and location information of memory cells, arrays and blocks of the device, and defines the physical attributes of the memory arrays that set the framework for logical-to-topological and topological-to-physical data conversions.

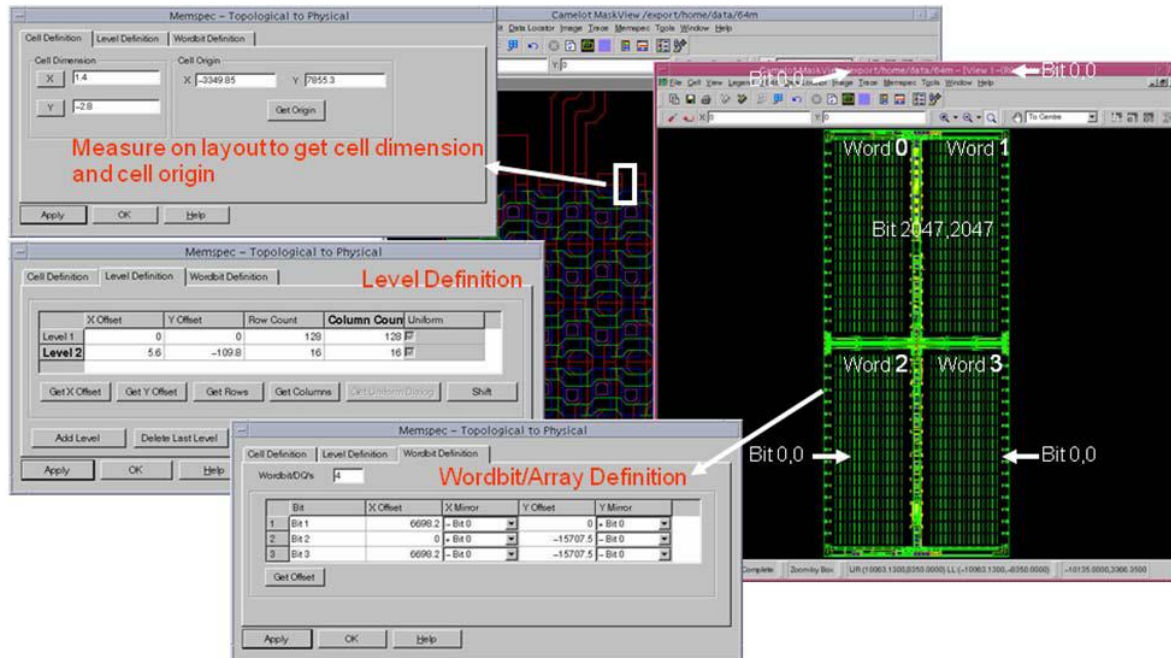


Figure 2. Defect Information Displayed by the K-Bitmap Tool

The base Camelot CAD Navigation software with the K-Bitmap tool allows FA teams to perform automated conversion of logical address to physical address. The tool enables faster identification and categorization of bit failure types. It provides an overlay of defects to bit failures to determine critical defects, hit rates and the root cause. It performs failure analysis for fast analysis of root cause failures due to non-correlated defects and provides a feedback loop for defect scan tools on sensitivity issues. It also integrates the ability to discover defects that are not detected by the scan tool and to scan new levels that are not detected by the scan tool.

Benefits of the Camelot K-Bitmap Tool

The Camelot K-Bitmap tool, in conjunction with the Camelot CAD Navigation software, provides various benefits for failure analysis. The tool converts bitmap logical failures to physical locations of the specified array for defining hard and soft failures. It allows users to define bitmap defect types through bitmap classification. It allows users to perform additional defect analysis by permitting overlay of inline process defects to bit failures. This isolates the root cause of the defect. The tool can perform FA on bit fails or logic maps that do not have a defect associated with them. This helps further characterize the scan tool's sensitivity and detect defects that occur at levels that are not scanned. Users can then use this data to create a list of additional defect types that affect yield.

Summary

With its powerful features, the K-Bitmap tool enables users to gather more data in less time than manual methods. Unlike a manual process which requires certain skills, the Camelot K-bitmap tool automates the process and can be easily used by various members of the team.

System Requirements

The system requirements for the Camelot Defect Wafer Map and K-bitmap option are as follows:

- System: SunFire or Linux system with one or more CPUs
- Processor: UltraSPARC IIIi for Sun Fire or Intel/AMD Dual/Quad Core for Linux
- OS: Solaris 9,10, Redhat Enterprise or Suse Linux
- Memory: > 8GB
- Speed: > 1GHz
- Disk space: > 250GB