

MAGMA DESIGN AUTOMATION

Automated Fault Localization and Verification Using Camelot LogicMap



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 discusses the process of closing the automated fault localization and verification loop using Camelot™ LogicMap.

Converting failing scan and scan pattern data from logical to physical addresses is a time-consuming and nearly impossible task. It becomes more complicated with challenging logic designs and complex patterns. A fault simulator program is needed to diagnose scan and scan pattern failures on a wafer-level basis due to the amount of die fallouts associated with this failure mechanism. Inline defect scan data overlaid to failed logic patterns can be performed through manual data manipulation, but it is a very time-consuming process. To compound this problem, manual failure analysis (FA) on logic devices is prohibitive due to increased metal levels, length of failure patterns at different levels and ever-shrinking geometries. An automated method is required to translate the logical to physical logic failures, overlay defects to these failures and drive FA tools to the X and Y coordinates automatically. Overlaying defects to these failures speeds identification. The patterns that do not have defects associated with them can be identified through FA to complete all defect failing categories that affect yield.

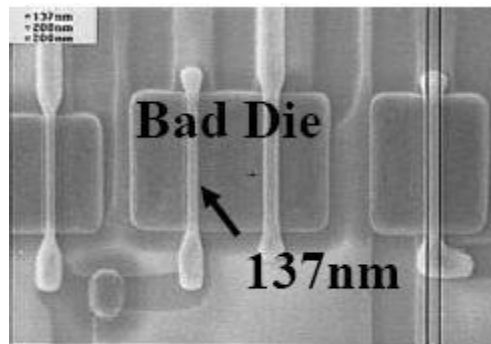


Figure 1. Manufacturing Weak Spot

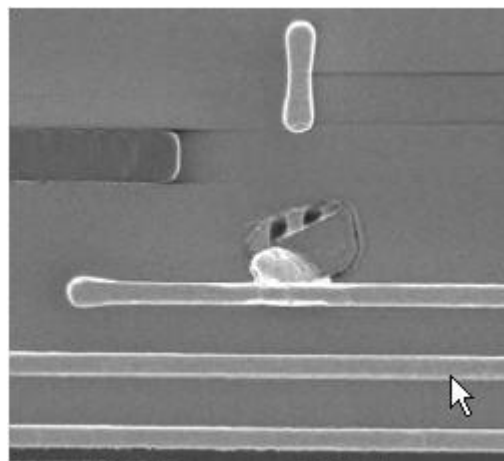


Figure 2. ILD Rip Out at CMP4

The Solution

To solve this problem, an approach is required that can automate the test flow diagnostics for defect localization with logic devices. This approach should have a quick and easy set up that comprehends multiple devices that can come online very quickly. It should enable expertise to be shared across different functional groups and should leverage the skills of many engineers working together as a cohesive group to perform root cause analysis. Additionally, it should communicate the results to automated stages of any FA tool to enable automated fault localization and verification.

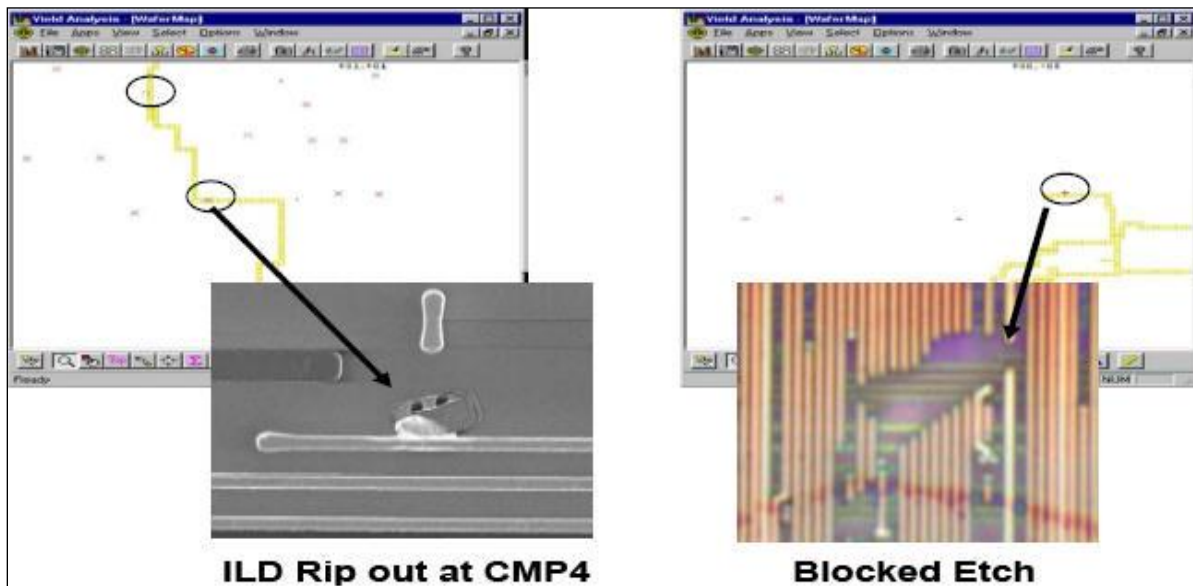


Figure 3. Logic Map Failed Electrical Nets Overlaid With Inline Defect Data Displayed in YieldManager™

The Camelot LogicMap Solution

The Camelot LogicMap solution has the required capabilities that allow users to efficiently close the loop for automated fault localization and verification. It provides automated conversion of logical to physical logic scans and patterns. It enables faster identification and categorical accounting of logic failure types or patterns. It provides an overlay of defects to logic failures. Failure analysis can be performed on bit fails or logic maps that do not have a defect associated with them.

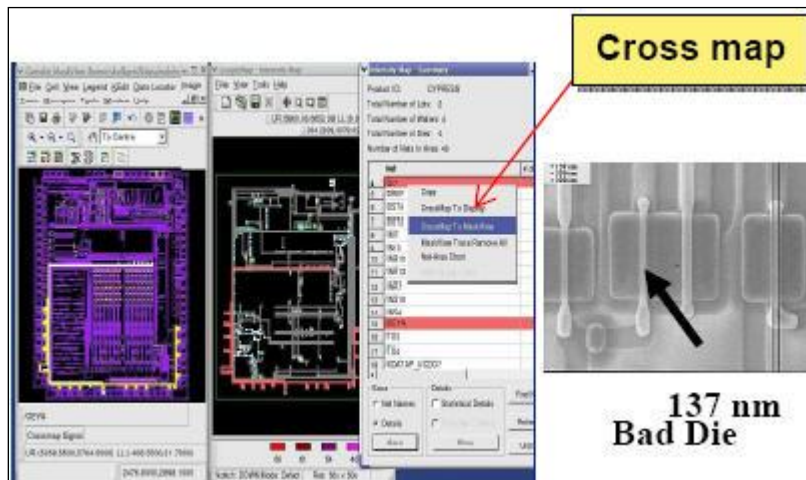


Figure 4: Cross Map

Benefits of the Camelot LogicMap Solution

Implementing the Camelot LogicMap solution provides various benefits. It allows users to convert logic map logical fails to physical locations within the logic section of the die for defining repetitive or hotspot fails. It defines types through logic map classification of the root cell. It also performs additional analysis with defect-scan-data-overlay to logic fails to identify root cause. Camelot LogicMap performs FA on logic fails that do not have a defect associated with them. This helps to further characterize the scan tool's sensitivity or to 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.

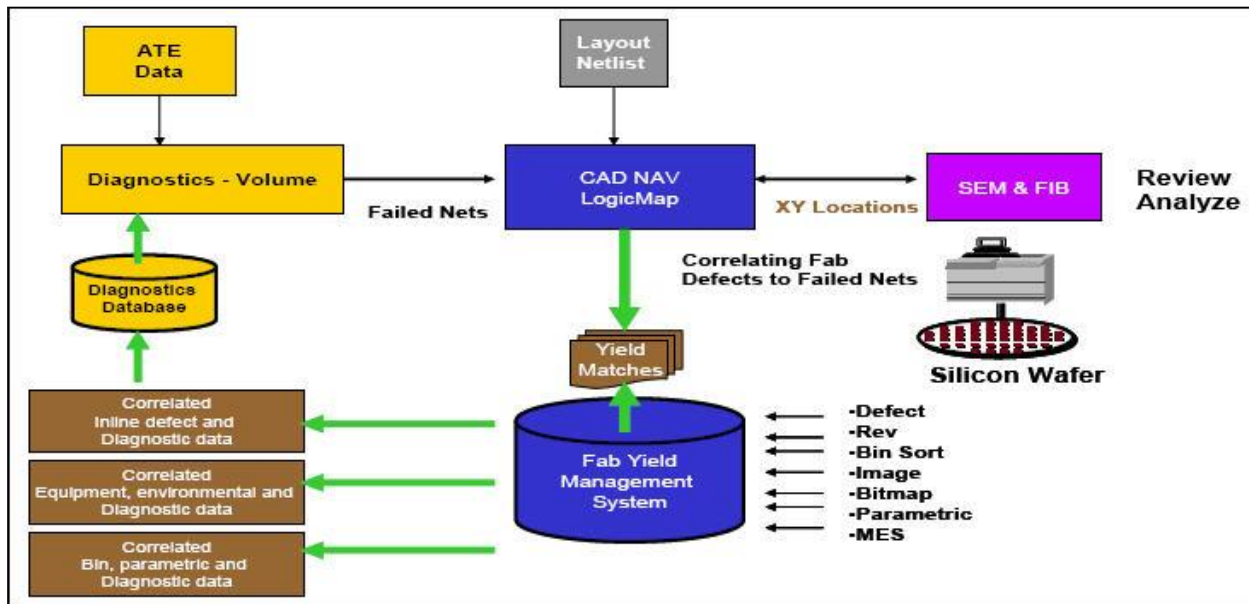


Figure 1. Camelot LogicMap Architecture

Summary

The Camelot LogicMap solution enables users to gather more data in less time than manual methods. Unlike a manual process which requires certain skills, the Camelot LogicMap tool automates the process and can be easily used by various members of the team.

System Requirements

This solution requires both the Camelot CAD Navigation software and the LogicMap option.

The system requirements for the LogicMap 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