Floating Touchdown - New methodology for high parallelism testing driving test time reduction

Doron Avidar

Roy Cohen

F12 Test Engineering

IEEE SW Test Workshop

Semiconductor Wafer Test Workshop

June 10 - 13, 2012 | San Diego, California

Introduction

- High parallelism Flash testing on small densities (high DPW) requires multiple touchdowns to cover the wafer (due to limited tester resources)
- At each touchdown the tester waits till all sites complete testing before sending command to the prober to move to the next touchdown
- The “slowest” tester site will determine the test time at each touchdown
- Test time of a wafer is determined by the sum of the slowest site at each touchdown

Introduction – Cont.

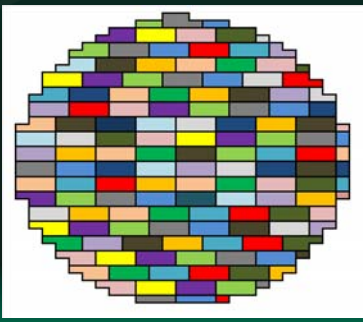
- If we can eliminate the need to wait for the slowest site at each touchdown we will be able to save test time
- How can we break down this Tester – Prober dependency?
- Lets try and save some test time by using Floating Touchdown

Floating TD concept

- **Manufacture a FWC (Full wafer Contact) probe card**
  - covering all DUTs with a single touchdown

- **Built-in design of tester site sharing schematic that will enable to cover the wafer in 1 touchdown using the inherent max. tester parallelism**
  - example : a wafer with 800 DPW and a tester with max. parallelism of //144 → each tester site will be split into 6 probe card sites (800/144 = 5.5 → rounded to 6)
  - This is instead of using a //144 card that will cover the wafer in 6 touchdowns

//144 Tester sites shared into 6 probe card site each

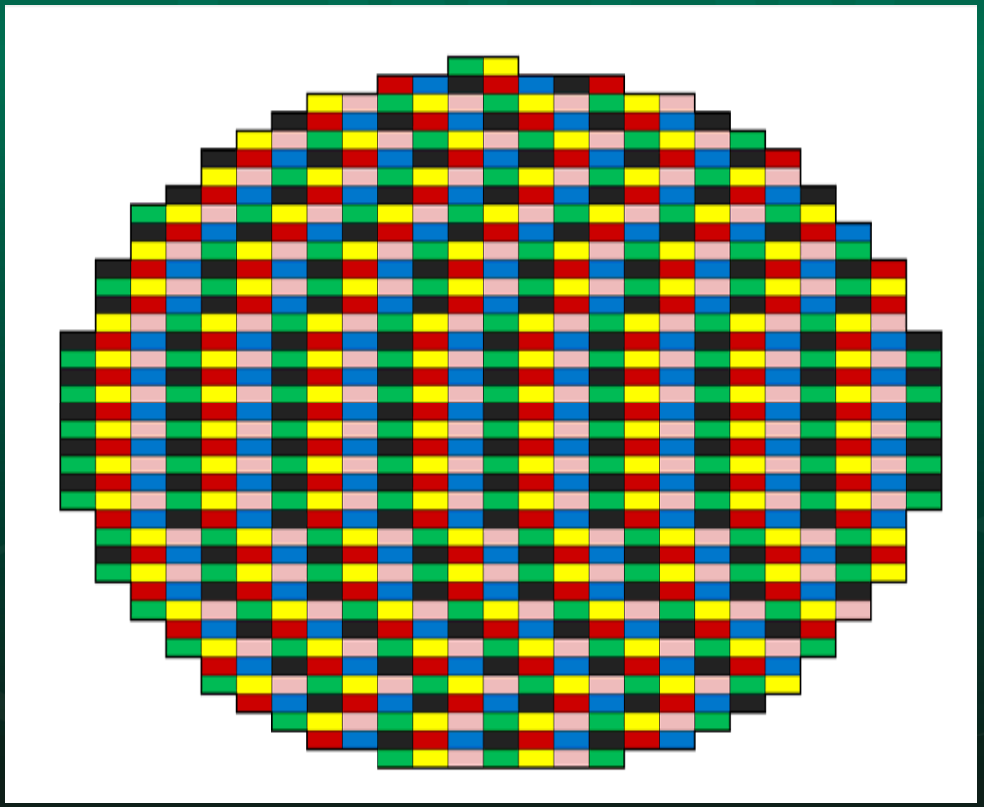


Floating TD concept – cont.

- **The testing of each tester site is done independently**
- **Each tester site will test it's shared probe card sites in a sequential mode (no parallel testing occur)**
  - Floating TD is Electronic stepping (vs. Mechanical stepping)
- **What happens to the test time?**
- **Remember current test time : SUM (MAX (TT/TD))**
- **Floating touchdown test time : MAX (SUM (TT/SITE))**
  - In order for the test time of the Floating TD to be equal to the Current test time , need to have one tester site that will probe the 6 slowest DUTs on the wafer.
    - Chance for that in the 800 DPW ~ 1/26M

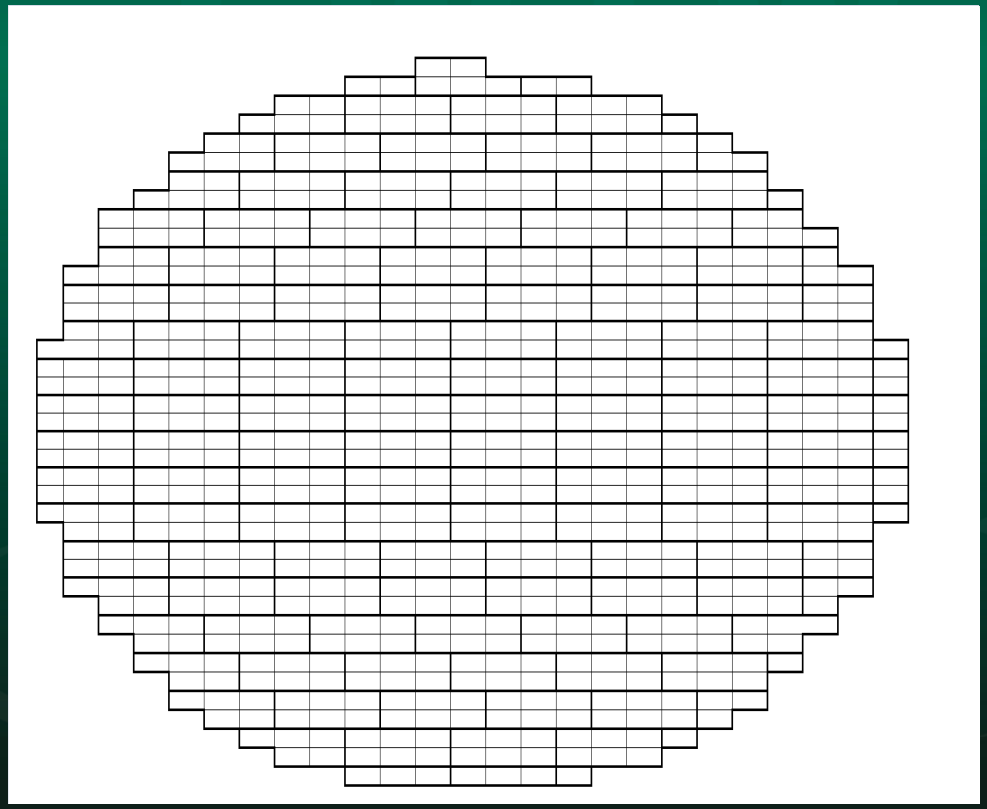
Current testing (FWA)

Wafer capture every 30sec



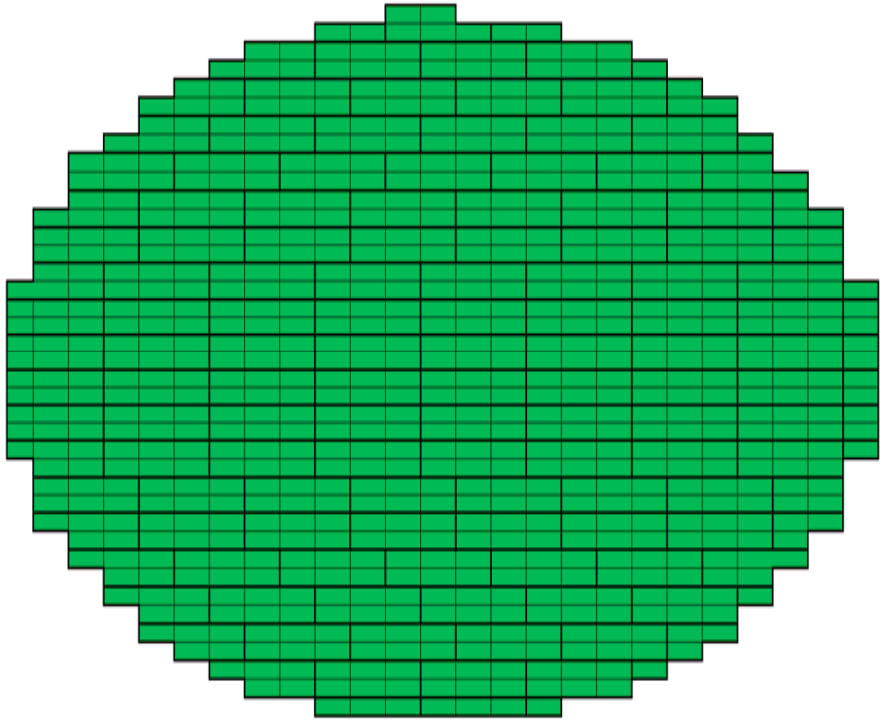
Floating TD testing

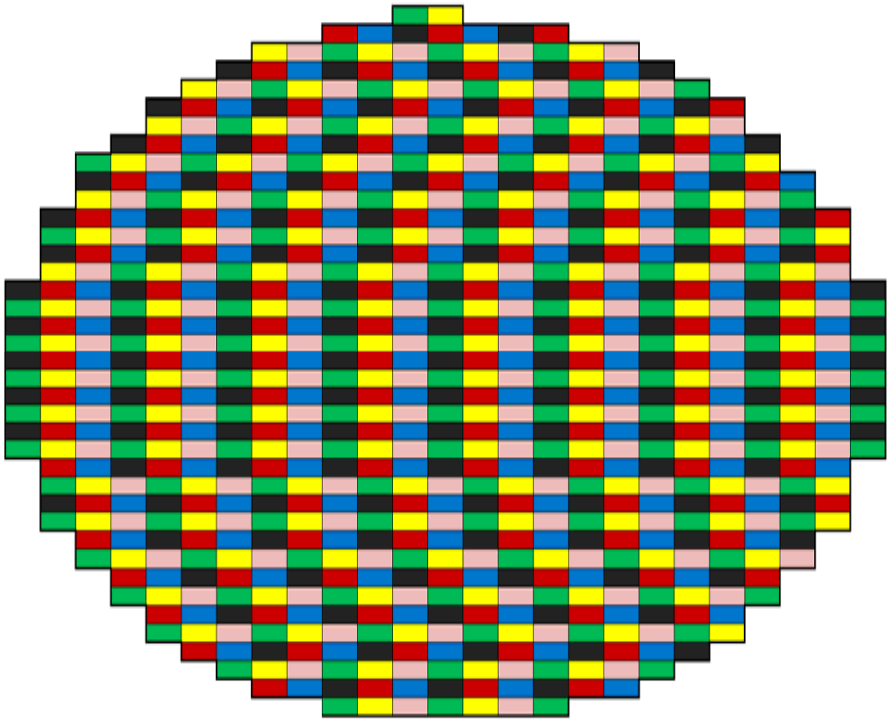
Wafer capture every 30sec



Test time comparison

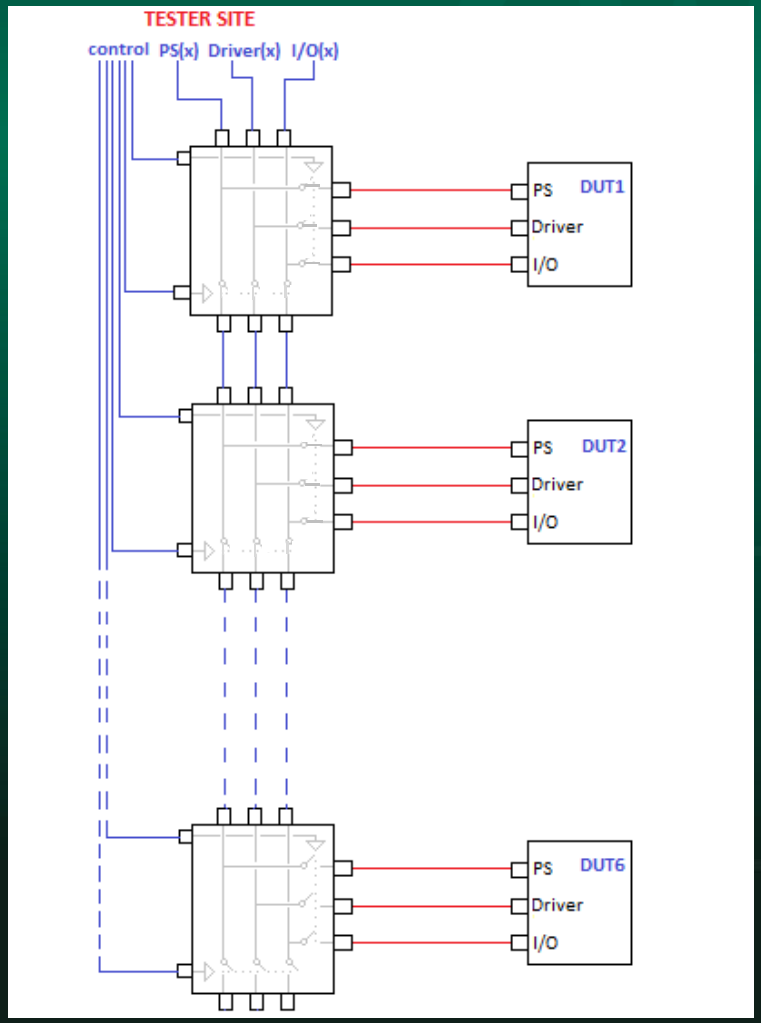
Wafer capture every 30sec





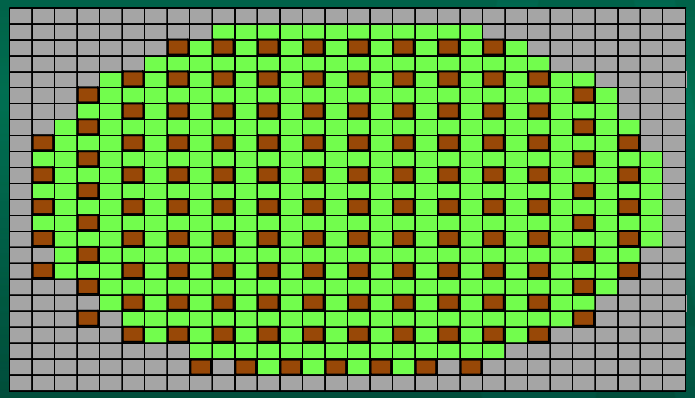
Floating TD **23% TTR** Current

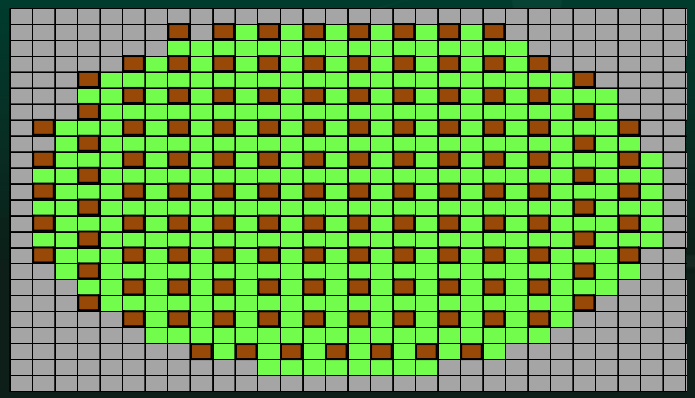
implementation



- **Sequential testing along the shared probe card sites**
- **Switch required to**
  - Switch PS + Drivers + I/O
  - Each control will open/close all channels at the same time
  - Minimum number of controls (resource limitation)

Implementation





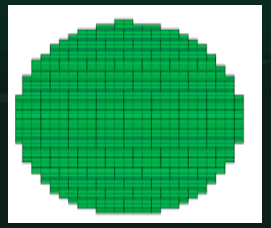
- **Current methodology**
  - Prober waits for tester command that all sites completed testing before moving to next TD
  - DUT X/Y set by the prober
  - Tester creates SiteLogs
- **Floating TD methodology**
  - Prober has 1 TD (initial X/Y)
  - Require MultiDUT environment
  - Test program handles all X/Y and SiteLog creation

Challenges on the way to FTD

- **Component complexity**
  - Component is required to handle high number of inputs/outputs (PS, Driver, IO)
  - Minimum switch controls
  - Number of components on the probe card
- **Signal integrity on switched channels**
  - Signal from tester to DUT runs through multiple components
- **Test program complexity**
  - Require to handle SiteLog and X/Y creation (instead of the tester)

Next steps

- **Benchmark components for sequential testing**
  - Input → PS, Driver, I/O
  - Output → (1) DUT (2) next component
  - Min. control needed for switching
- **Test program development**
  - Handle sequential testing
  - Create X/Y
  - Create SiteLog independently
- **Prototype**
- **Optimize test time reduction**
  - What impacts the TTR?
  - How to achieve Max. TTR?



Summary

- Floating TD is a testing methodology aimed to save test time by overcoming the dependency of the Tester – Prober during the testing process
- Using your current test cell with a unique probe card site sharing of this methodology might save you up to 25% test time
- Floating TD is beneficial for designs with multiple touchdown count
- Complexity of implementation depends on the testing environment existing at your site

Thank You

Q?