OPTIMAL+
Lifecycle analytics you can trust

Automatic scratch detection

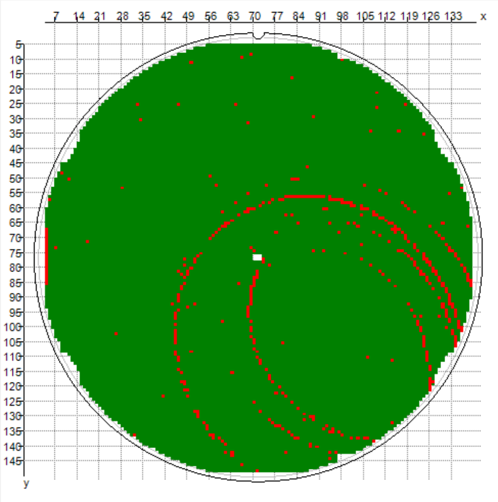
Improved product quality and reliability performance

Dan Sebban, Leonid Gurov & Gal Peled

IEEE International Workshop on Automotive Reliability & Test (ART)

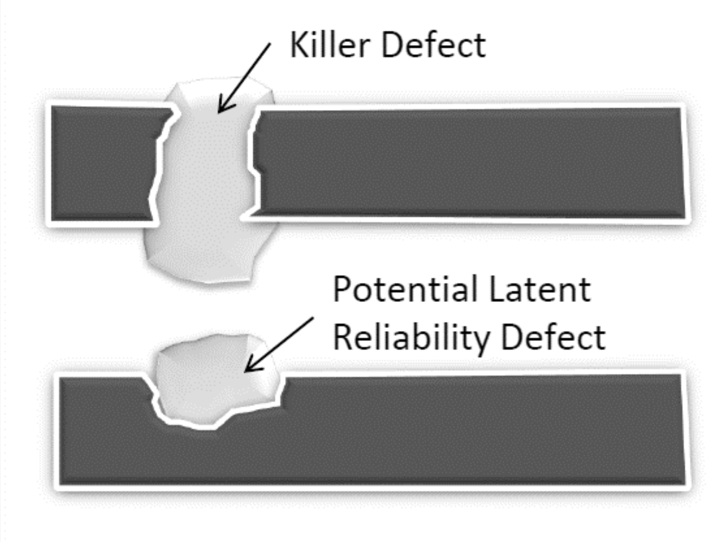
November 2018, Phoenix, AZ

Introduction



- Scratch patterns are clusters on a wafer having an elongated shape with a high aspect ratio.
- Scratches are usually caused by equipment alignment being out of tolerance or by mishandling by a human.
- Nearby chips can have hidden defects due to particles created from the scratching action but nevertheless pass test. These chips usually have lower reliability due to the latent defects.
- Having an automated way to detect and kill such unreliable dice is a must to meet aggressive quality and reliability requirements.

Latent defects



- The defect types that impact reliability are generally the same as those that impact yield.
- They are distinguished primarily by size and proximity to critical design features.

Source: KLA –Tencor Inline Defect Part Average Testing (I-PATTM) - 2018 Automotive Electronics Council Reliability Workshop

The root cause for scratches: CMP Example

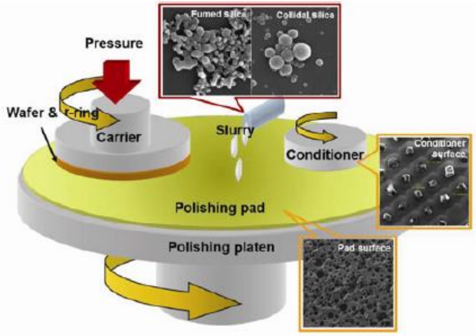


Fig. 2 Schematic diagram and consumables of CMP process.

Table 2 Potential causes of CMP defects and possible solutions [33].

Defect mode	Potential causes	Impact to device	Potential solutions
Particles	• Slurry/pad residue • Polish byproducts	• Shorting/opens • Pattern distortion	• Cleaner tooling • Clean chemistries
Macro scratches	• Large/hard foreign particles on polish pad	• Pattern removal over multiple die	• Pad conditioning • Pad cleaning • Environment
Micro scratches	• Slurry agglomeration • Pad asperities	• Shorting/opens	• Slurry filters • Pad/pad conditioning
Corrosion (metal CMP)	• Slurry chemistry • Clean chemistry	• Opens, Reliability	• Passivating films, • Chemistry optimization
Film delamination	• Weak adhesion • CMP shear force	• Shorting/opens • Device parametrics	• Improve adhesion • Low pressure CMP
Organic residue	• Inadequate cleaning • Residual slurry components	• Shorting/opens • Disturbed patterning of next layer	• Cleaner tooling • Slurry optimization • Clean chemistries

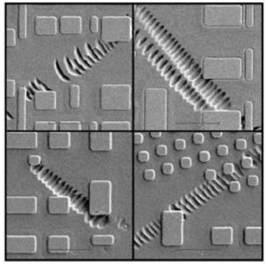


Fig. 5 Chatter mark scratches observed in STI CMP [33].

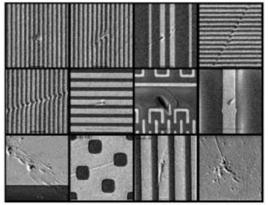
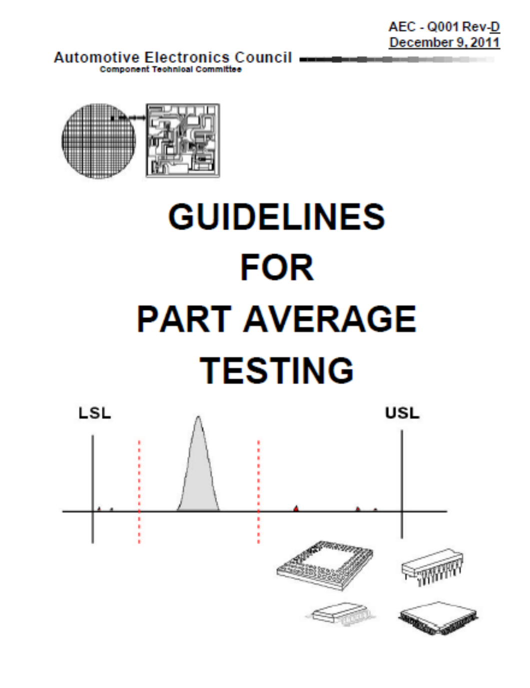


Fig. 6 Various scratches formed in Cu CMP [33].

Source: Scratch formation and its mechanism in chemical mechanical planarization (CMP)
Tae-Young KOWN, Manivannan RAMACHANDRAN, Jin-Goo Park

Requirements for outliers detection

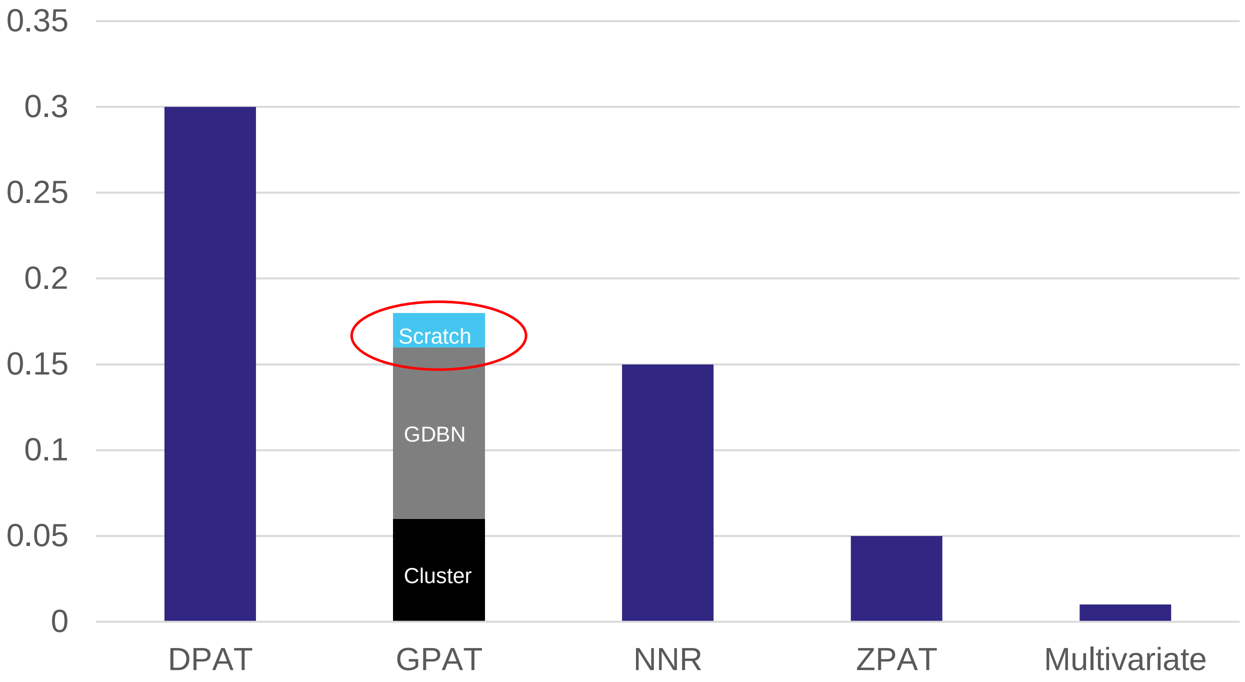


The AEC (Automotive Electronics Council) guidelines for Part Average Testing are not very specific in regards to Geographic PAT

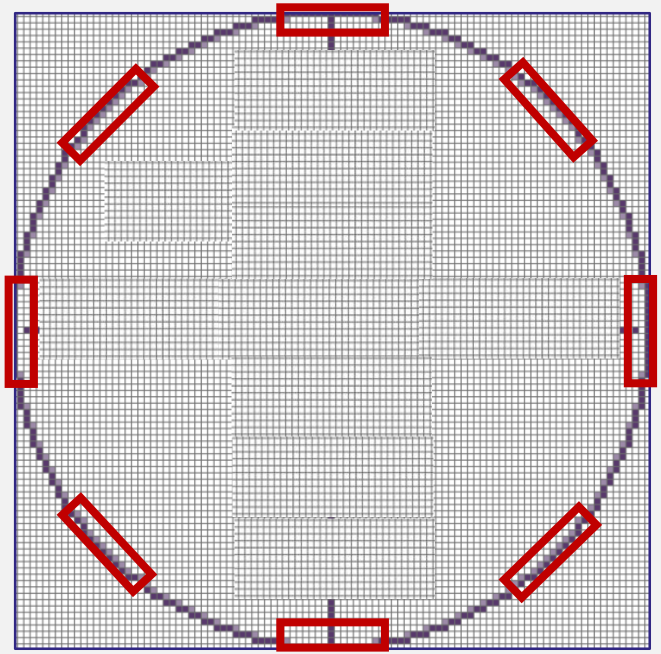
Mandatory techniques to identify good dice outliers within a wafer/lot:

- Adaptive Dynamic PAT
- Geographic PAT
- Wafer Map Stacking (or Z-PAT)

Typical yield loss due to outlier detection algorithms

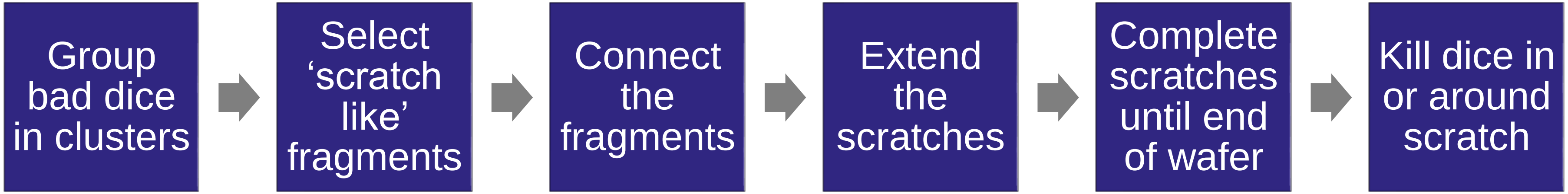


Concept for scratch detection



- When you look at a circle on a pixel map you can see that it consists of multiple straight lines.
- In the scratch algorithm, we look for these lines and check to see if there is a good arc that connects them together.

Scratch detection algorithm flow (simplified)



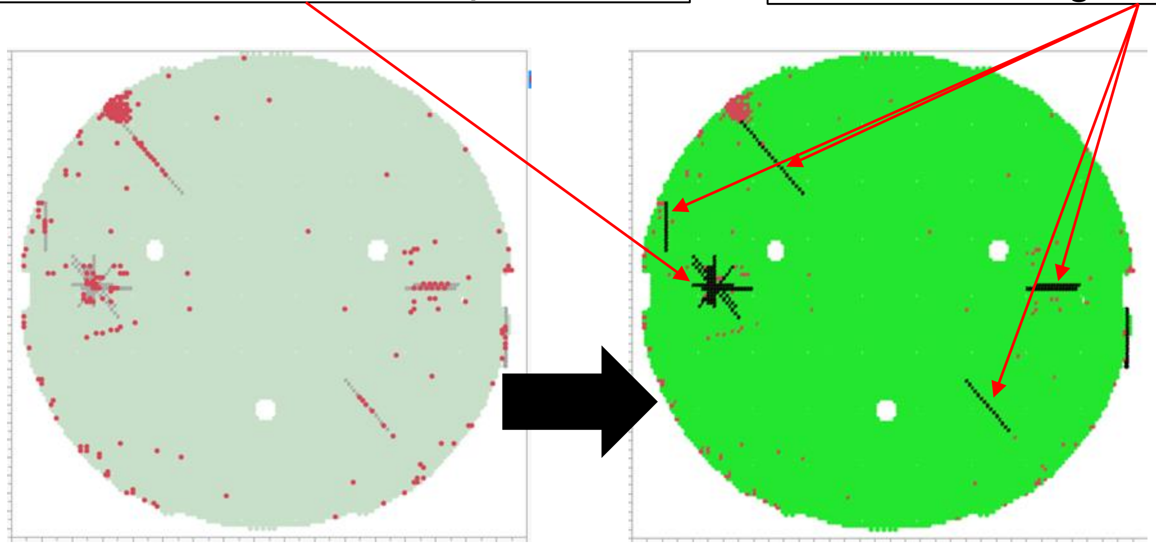
Patent pending

Algorithm flow step → Select scratch-like fragments

Group bad dice in clusters → Select ‘scratch like’ fragments → Connect the fragments → Extend the scratches → Complete scratches until end of wafer → Kill dice in or around scratch

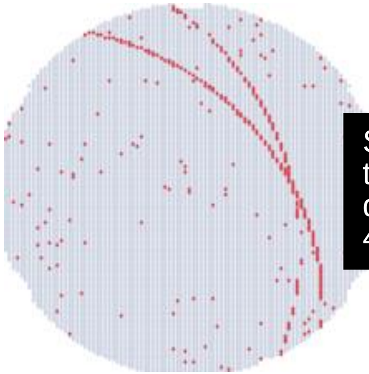
Cluster where segments of different directions intersect has low aspect ratio

Separated segments represent well-defined fragments of scratch

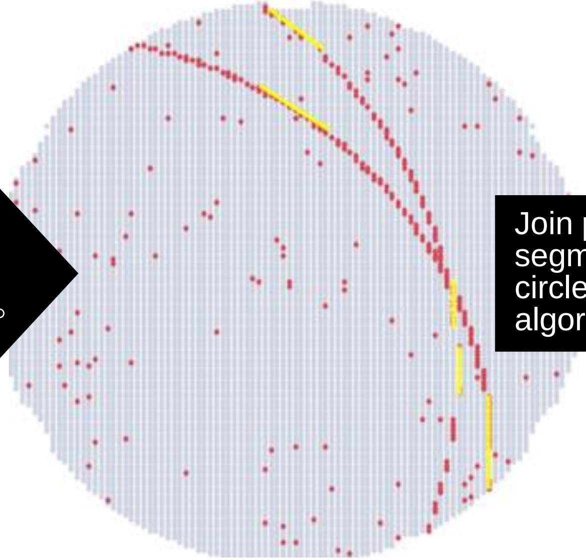


Algorithm flow step → Connect the fragments

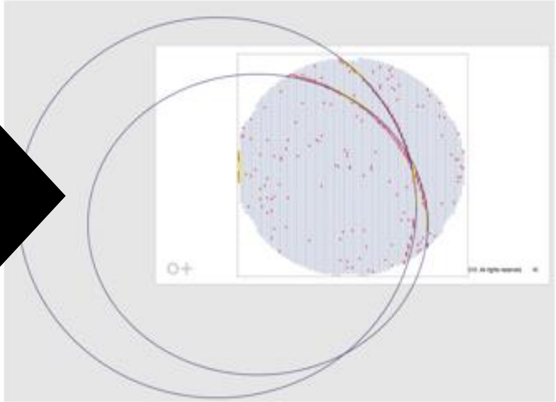
Group bad dice in clusters → Select 'scratch like' fragments → **Connect the fragments** → Extend the scratches → Complete scratches until end of wafer → Kill dice in or around scratch



Segments in the four main directions: 0°, 45°, 90°, 135°



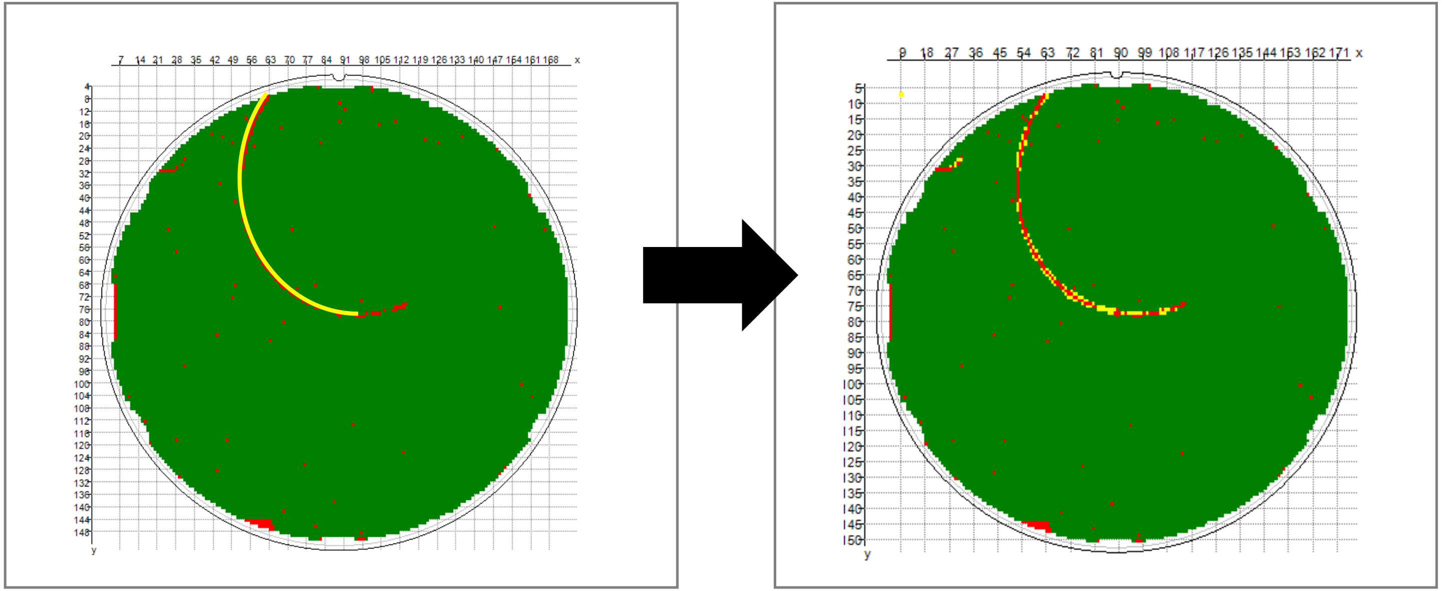
Join pair of segments by circle fitting algorithm



Patent pending

Algorithm flow step → Extend the scratches

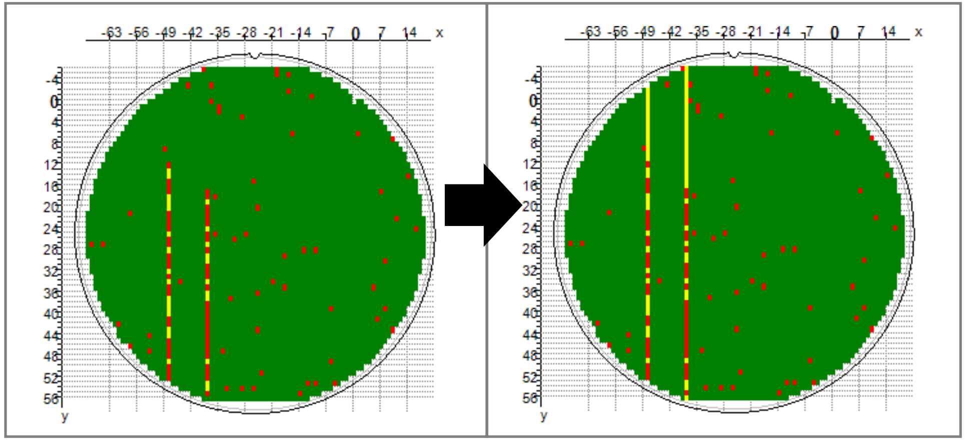
Group bad dice in clusters → Select ‘scratch like’ fragments → Connect the fragments → **Extend the scratches** → Complete scratches until end of wafer → Kill dice in or around scratch

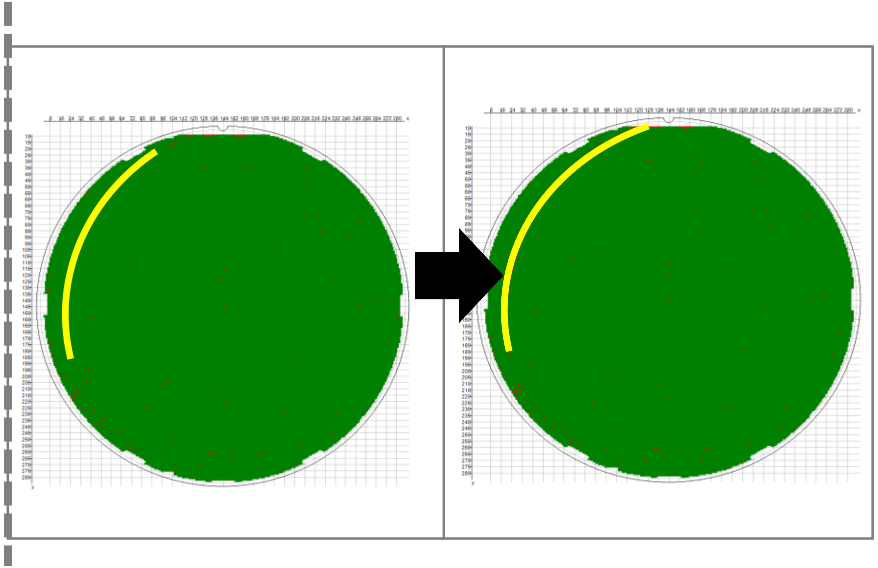


Patent pending

Algorithm flow step → Complete scratches until end of wafer

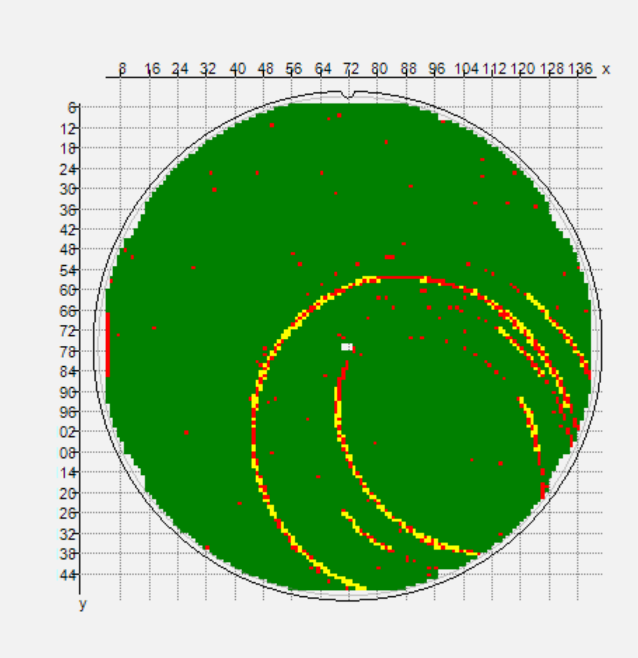
Group bad dice in clusters → Select ‘scratch like’ fragments → Connect the fragments → Extend the scratches → **Complete scratches until end of wafer** → Kill dice in or around scratch

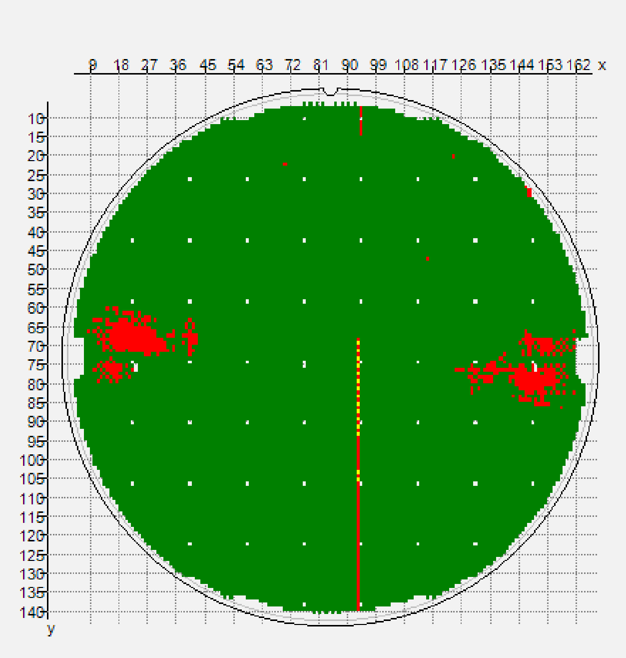


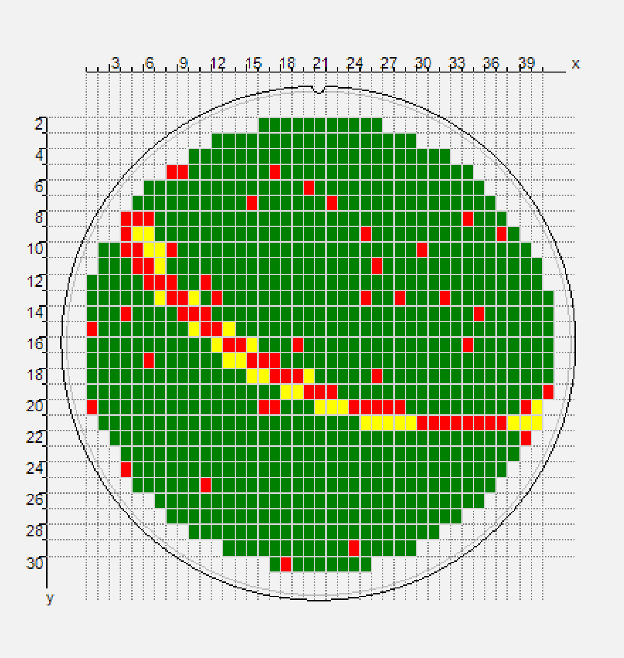


Patent pending

Scratch detection examples

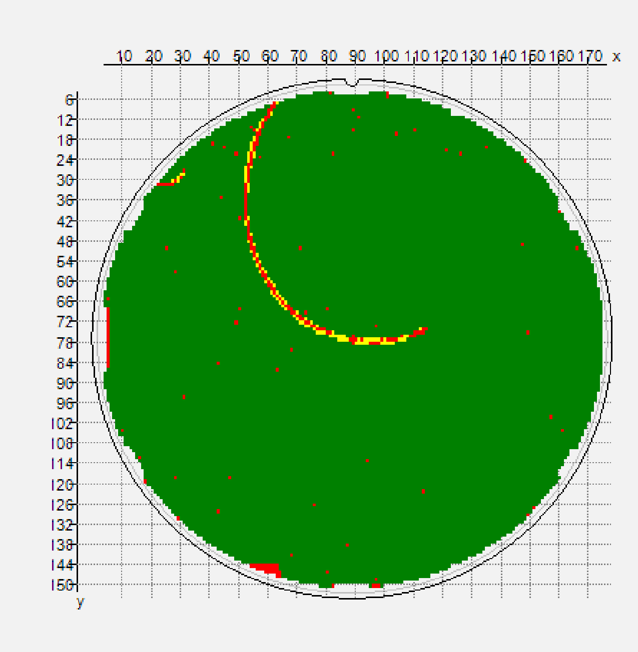


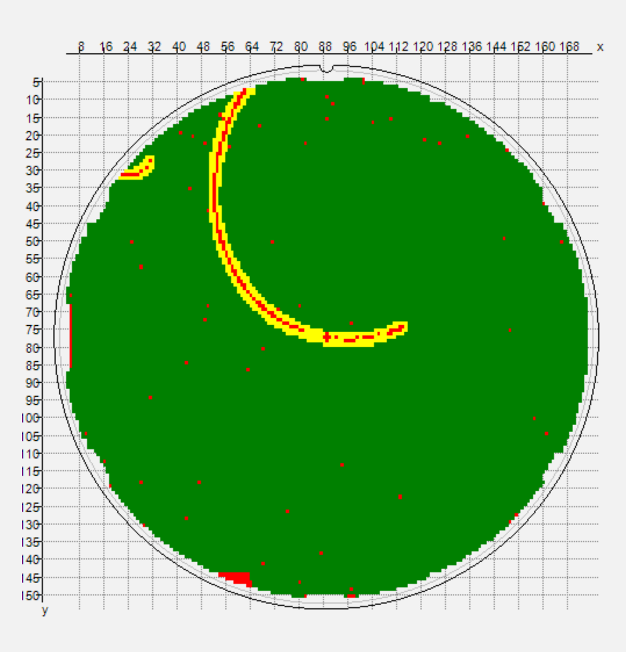


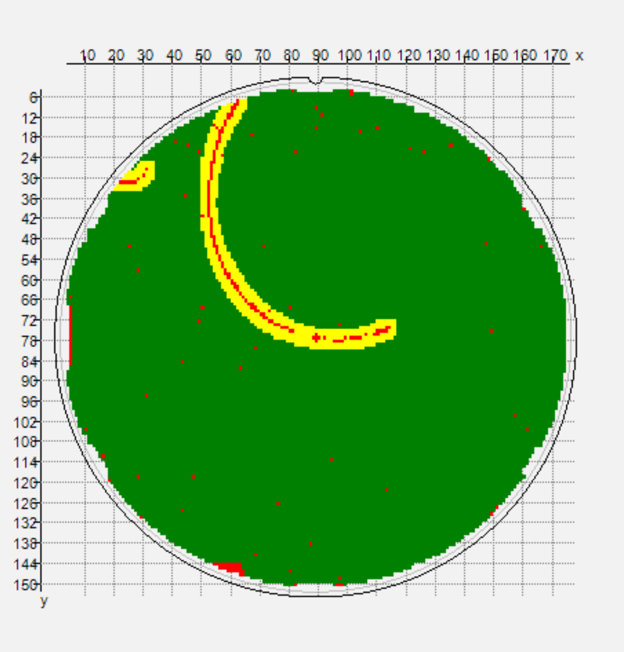


Ink inside vs. ink around

Ink inside	One ring	Two rings







Patent pending

Scratch detection confusion matrix

Confusion Matrix			
		Actual	
		scratch	not scratch
Predict	scratch	True positive	False positive
	not scratch	False negative	True negative

Precision
likelihood a prediction of a scratch is correct

$$\frac{\text{True positive}}{\text{Predicted scratch}}$$

Sensitivity
likelihood a scratch will be detected

$$\frac{\text{True positive}}{\text{Actual scratch}}$$

Accuracy
likelihood the overall prediction is correct

$$\frac{\text{True positive + True negative}}{\text{Total population}}$$

Patent pending

Algorithm fine tuning – Performance

Wafer Level					
		Actual			
		scratch	not scratch	precision	Accuracy
Predict	scratch	65 (2.31%)	6 (0.21%)	91.55%	99.75%
	not scratch	1 (0.04%)	2739 (97.44%)		
	sensitivity	98.48%			

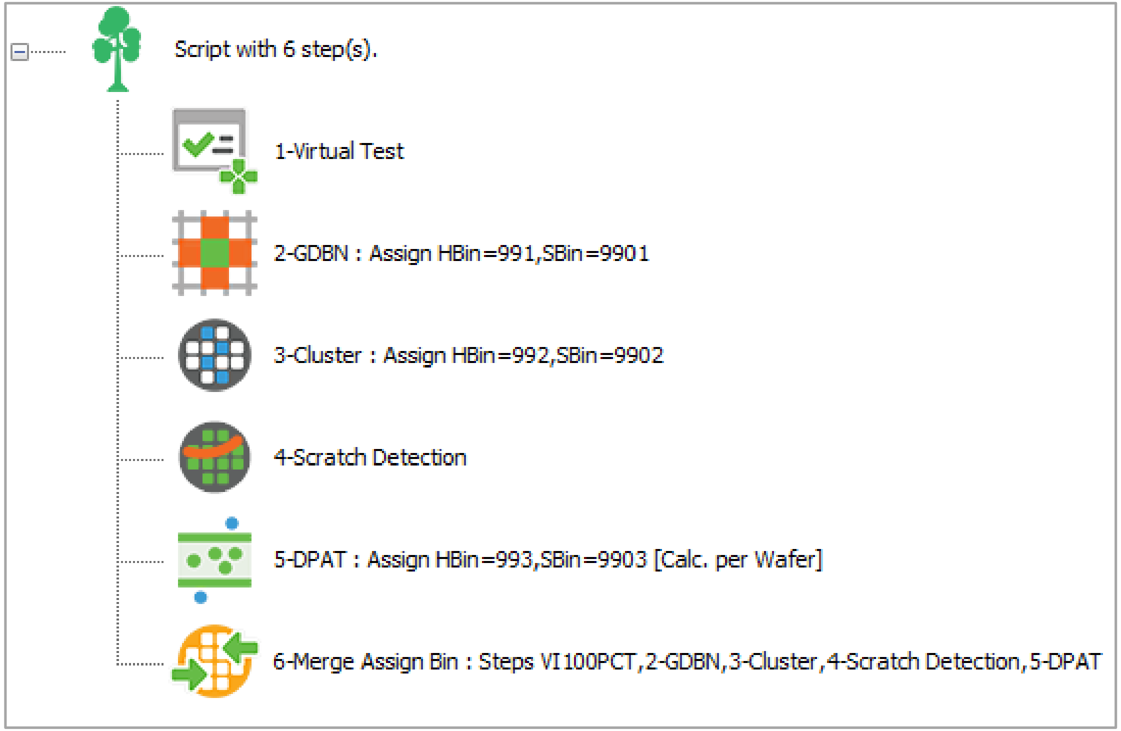
Scratch Level [numbers are # of scratches]					
		Actual			
		scratch	not scratch	precision	Accuracy
Predict	scratch	96 (90.57%)	8 (7.55%)	92.31%	
	not scratch	2 (1.89%)			
	sensitivity	97.96%			

Dice Level [numbers are # of good dice]					
		Actual			
		scratch	not scratch	precision	Accuracy
Predict	scratch	1557 (0.04%)	338 (0.01%)	82.16%	99.99%
	not scratch	20 (0.001%)	3957155 (99.95%)		
	sensitivity	98.73%			

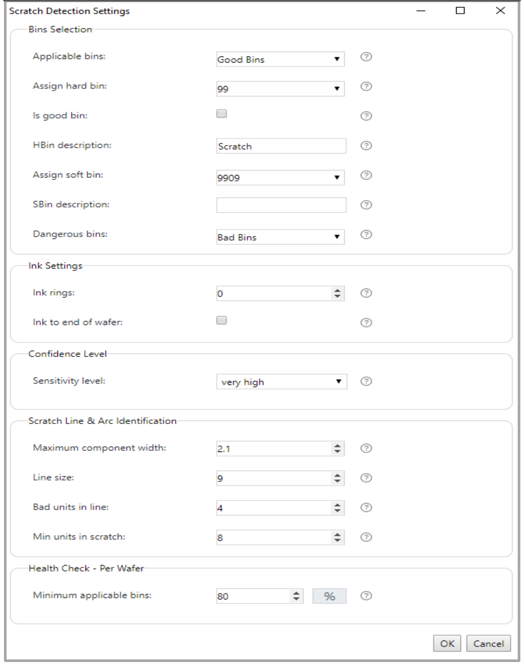
- Tests were done on 2,811 wafers from different products and dice sizes. The algorithm ran in Automatic Mode (same default settings)
- **Wafer level:** shows how good the algorithm is for detecting wafers with suspected scratches
- **Scratch level:** shows how reliable the algorithm is in distinguishing between different scratches on the same wafer
- **Die level:** shows how reliable the algorithm is in inking (killing) individual dice and predicts the expected yield loss and DPPM

Patent pending

Scratch detection step within a full outlier detection rule



Scratch step settings



Patent pending

Scratch detection specific settings

Sensitivity level – controls the sensitivity of the algorithm

Ink rings – the number of rings that will be inked around the scratch

Ink to end of wafer – the scratch will be inked to the edge of the wafer

Maximum component width – aspect ratio for the scratch

Line size – the minimum length of the line pattern

Bad units in line – the number of bad units in a line pattern to qualify as a scratch candidate

Min # of units in a scratch – the minimum number of bad units to be considered a scratch

Patent pending

Takeaways

A new algorithm to automatically identify scratch patterns on a wafer and kill potential unreliable dice was developed

It detects scratch patterns with different shapes, such as:

- Straight lines
- Curvy lines
- Single and multiple lines including discontinued shapes

It is possible to also use inline defectivity data (on top of test data) as input to the algorithm

The algorithm can be used in conjunction with additional parametric and geographic outlier detection steps, and improves reliability for screened devices

Thank you

OPTIMAL+

Lifecycle analytics you can trust