



# Dynamic Test Cell Controller

## Test Cost Reduction at no compromise in Wafer Sort Operations

Rob Marcelis, Salland Engineering (Europe) BV  
rob.marcelis@salland.com

# Introduction

Dynamic Test Cell Controller

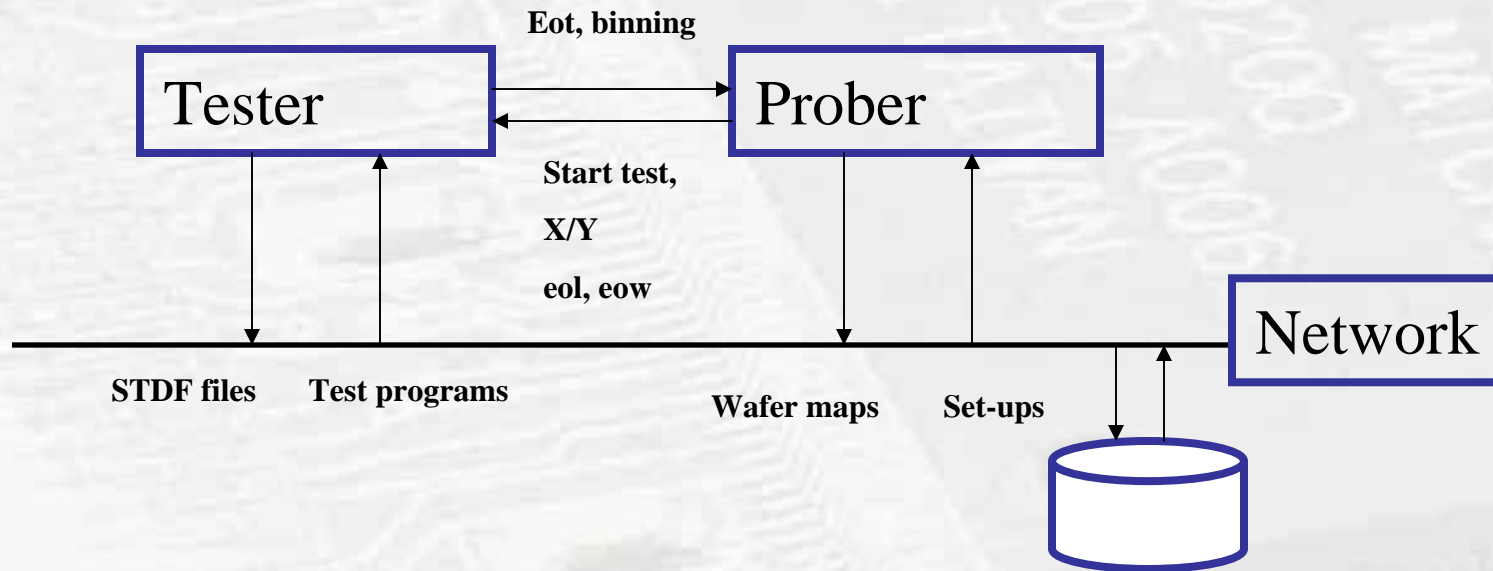
- Consistent increase of test cost over the years
- Manual entry of relevant data
  - Set-ups, test programs
  - Lot numbers, etc
- Information gathering done by Tester & prober independently
- Inefficiency in current process caused by lack of control
- No alternatives on individual equipment for improvements
- Integration need for “factory automation” environment
- More “bare die” to be delivered in the future

**The need for central control unit!!**

# Original set-up

- Traditional the prober and the tester form the test-cell
- Tester and prober communicate with each other
- Set-ups are loaded from the network to the prober
- Test programs are loaded from the network to the tester
- At the end wafer, maps are stored on the network by the prober
- STDF records are stored on the network by the tester
- No direct relation between wafer map and STDF

# Original set-up

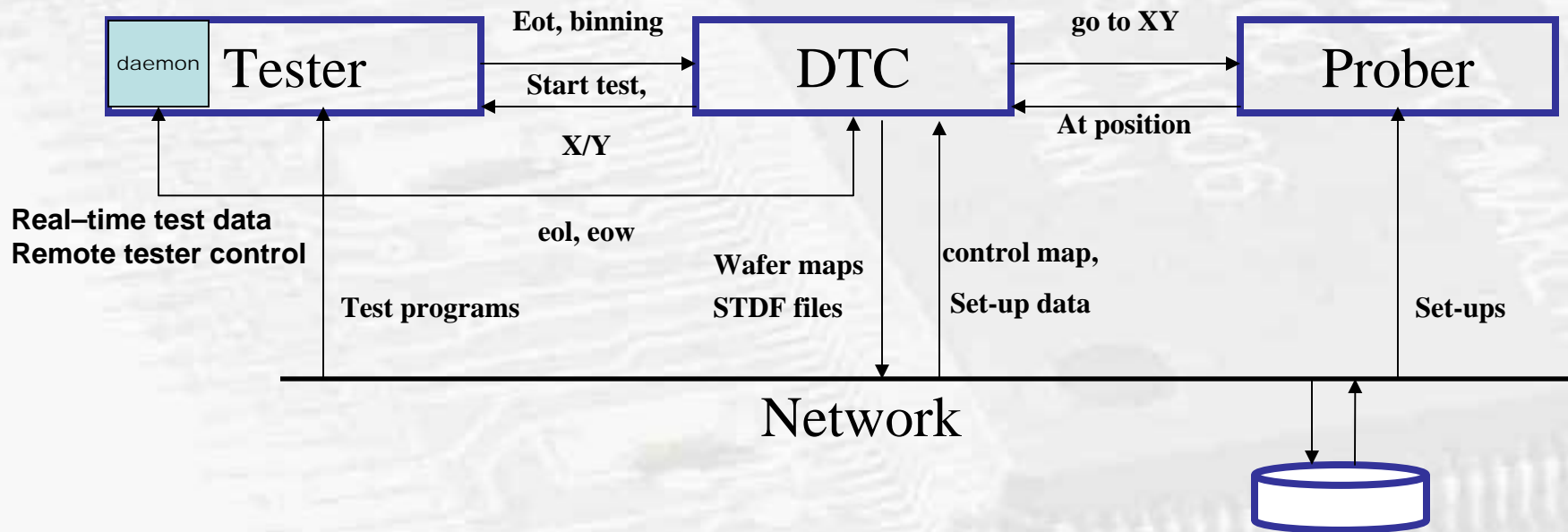


# DTC set-up

- DTC in between tester and prober
- No results to the prober
- Installation of tester daemon on the tester workstation
- Prober will be under remote control as well as the tester
- Events will be handled and initiated by the DTC
- All advantages offered by the DTC can now be used!!

# DTC set-up

Dynamic Test Cell Controller





# Process Optimization

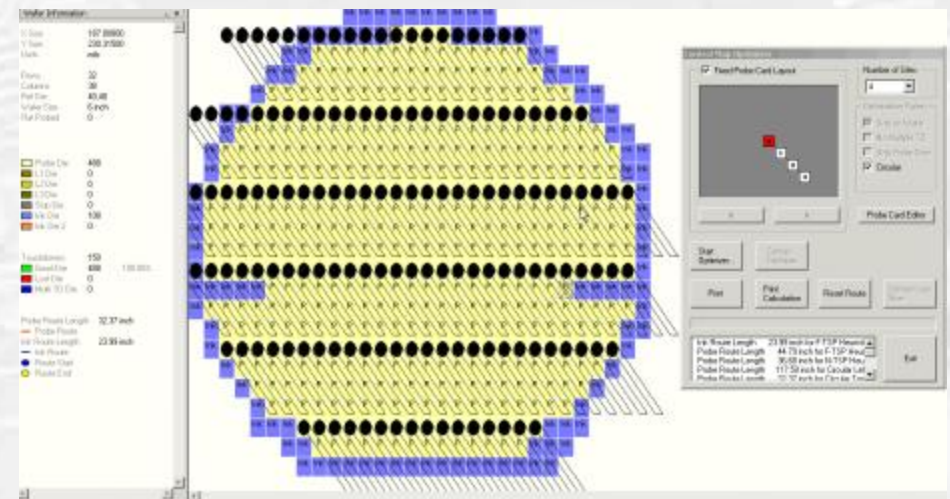
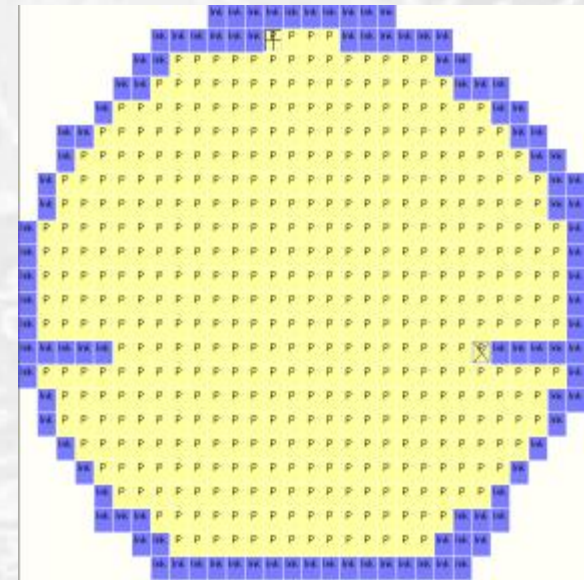
Dynamic Test Cell Controller

- Prober control
  - Control map optimization
  - Communication overhead reduction
  - API functionality
  - Retest recovery analysis
- Tester control
  - Tester daemons

# Process Optimization

## Dynamic Test Cell Controller

- Prober control
  - Control map optimization
  - Circular top/down versus optimized
    - Circular top/down 150 touchdowns
    - Optimized 138 touchdowns
    - 8% reduction!! On touchdowns
    - 12.5% reduction on travel route





# Process Optimization

- Prober control
  - Communication overhead reduction
    - Standard communication flow:
      - For Electroglas probers:
        - Prober; Start test together with X/Y coordinates and site code
        - Tester; End of test with binning
        - Prober; Map update, internal house keeping, move to next die
      - With DTC to Electroglas:
        - DTC to prober; Move X/Y
        - Prober to DTC; At position
        - DTC to tester; Start test together with X/Y info and site code
        - Tester to DTC; End test with binning
        - DTC to prober; Move X/Y

Saving: Internal house keeping and map update on prober

# Process Optimization

- Prober control
  - Communication overhead reduction
    - Standard communication flow:
      - For TSK Probers:

Prober;	SRQ to inform at position
Tester;	SRQ to acknowledge
Tester;	Ask for site code information
Prober;	Answers with site code info
Tester;	Ask for X/Y coordinates
Prober;	Answers with X/Y information for site "0"
Tester;	Receipt of X/Y info is trigger to start test
Tester;	At end of test send binning.
Prober;	Map update, internal house keeping, move to next die
Prober;	SRQ to inform at position
      - With DTC to TSK:

DTC to Prober;	Move X/Y
DTC to Tester;	During Prober index send site code
Prober to DTC;	At position DTC to Tester; X/Y info to Start test
Tester to DTC;	end test with binning
DTC to Prober;	Move X/Y

Saving: Internal prober house keeping and map update plus part of tester communication during indexing of prober

# Process Optimization

- Prober control
  - API functionality
    - On each event hook a separate program can be launched. New functionality can be added by this mechanism.
  - Retest recovery analysis
    - Only retest those bins which have proven to be recoverable. In practice this saves 50-70% of the retest time.
    - Combine this with direct retest instead of retest at the end, saving is also on the travel time.

# Process Optimization

- Tester control
  - Tester daemons
    - Via the tester daemon, remote control is offered over the tester
    - Combine this with direct retest instead of retest at the end, saving is also on the travel time.
      - Real time test data is available
      - Instruction to load and run programs can be given

# System Utilization

- Real time monitoring
  - Count down to next assist moment
  - Alarm pole support
  - Differentiate down time between tester and prober
- Reporting
  - Web report to show floor status in one view
  - Ability to show down to wafer map level
  - Graphical and textual reports showing Utilization
    - Per selected period
    - Per selected machine
    - Entire sort floor



# Real time data grabbing

Dynamic Test Cell Controller

- Tester daemon generates real-time tester data
- DTC could decide for pass/fail
- No need to switch-on data log function on tester
- STDF records generated by the DTC
- Ideal place for wafer map to “meet” STDF record
- 100% data integrity
- Abort resume without corruptions
- One data log format regardless the connected tester

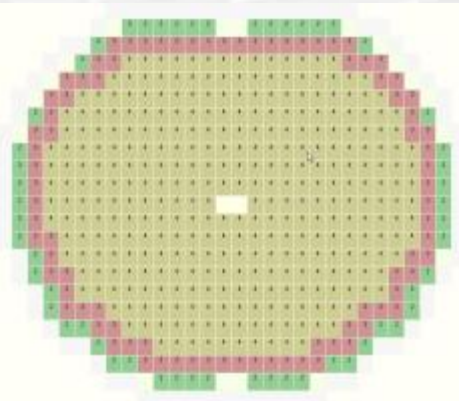
# Real time analysis

## Dynamic Test Cell Controller

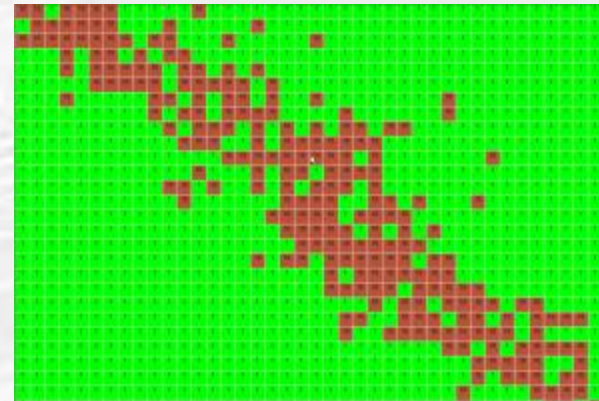
- Trend watching and responding
- Adaptive testing
- Dynamic sampling, result driven
- Part Average Testing
- Cluster detection
- Smart Sample Probe
- Dynamic lot composite, dynamic control map
- Drift map generation

# Cluster detection

- Cluster detection is done according the Intel investigation:  
“Reliability Improvement and Burn-in Optimization Through  
The Use of Die Level Predictive Modeling”  
First shown is implementation of the Intel document,  
followed by cluster map calculation



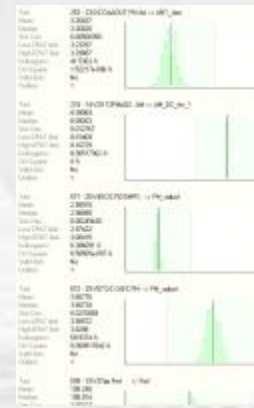
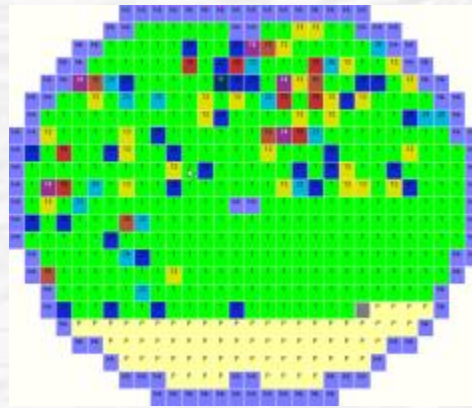
SWTW - 2006



Rob Marcelis

# Part Average Testing

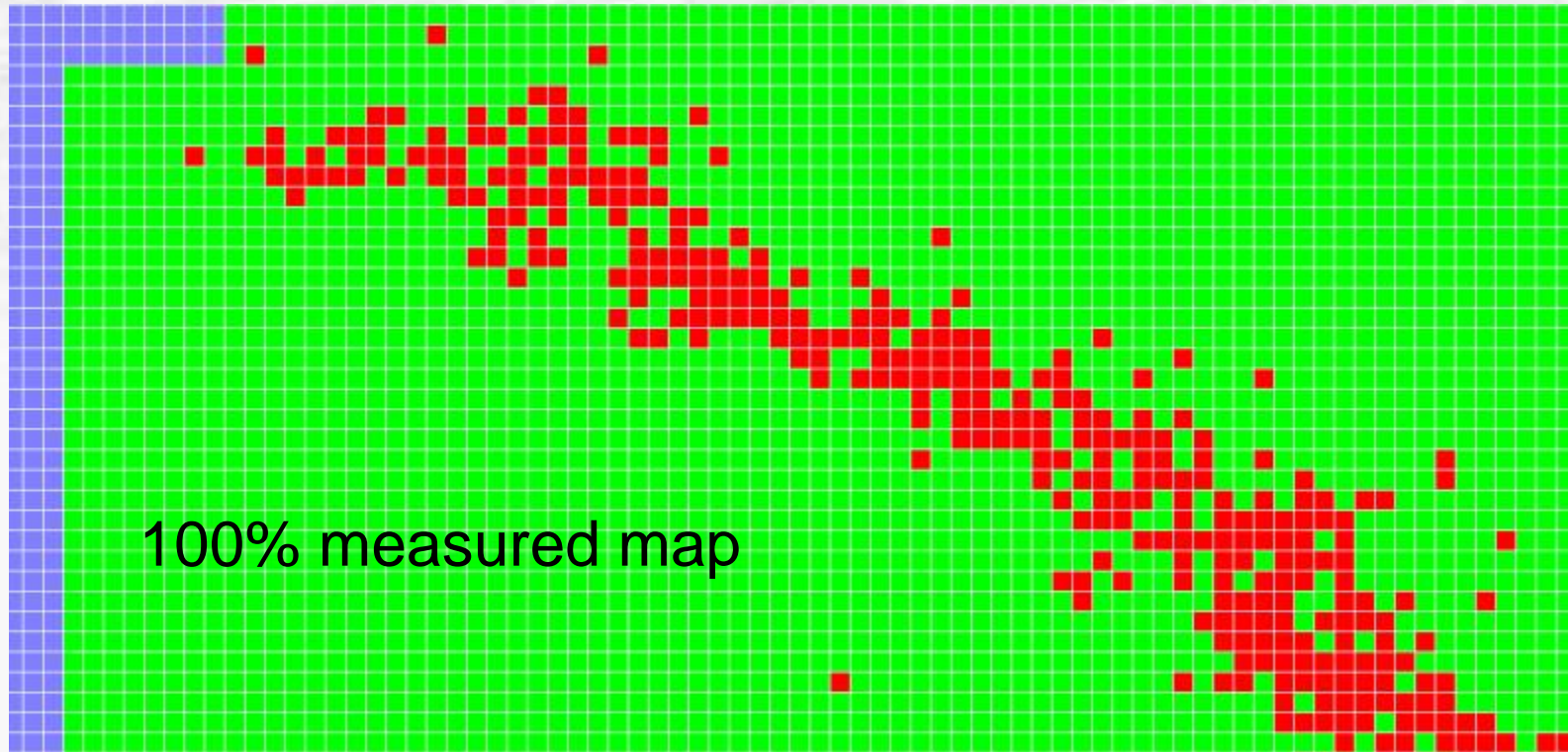
- PAT example on the DTC
  - Real time data is stored in “STDF”
  - At end of wafer, first cluster detection takes place
  - Followed by PAT analysis or outlier detection
  - Cluster bin is 31, PAT bin is 30





# Dynamic sampling

- Smart Sample probe example





# History tracking

- Full traceability of all events during wafer/lot
- Overview of all decisions when and why
- In file format and/or report format

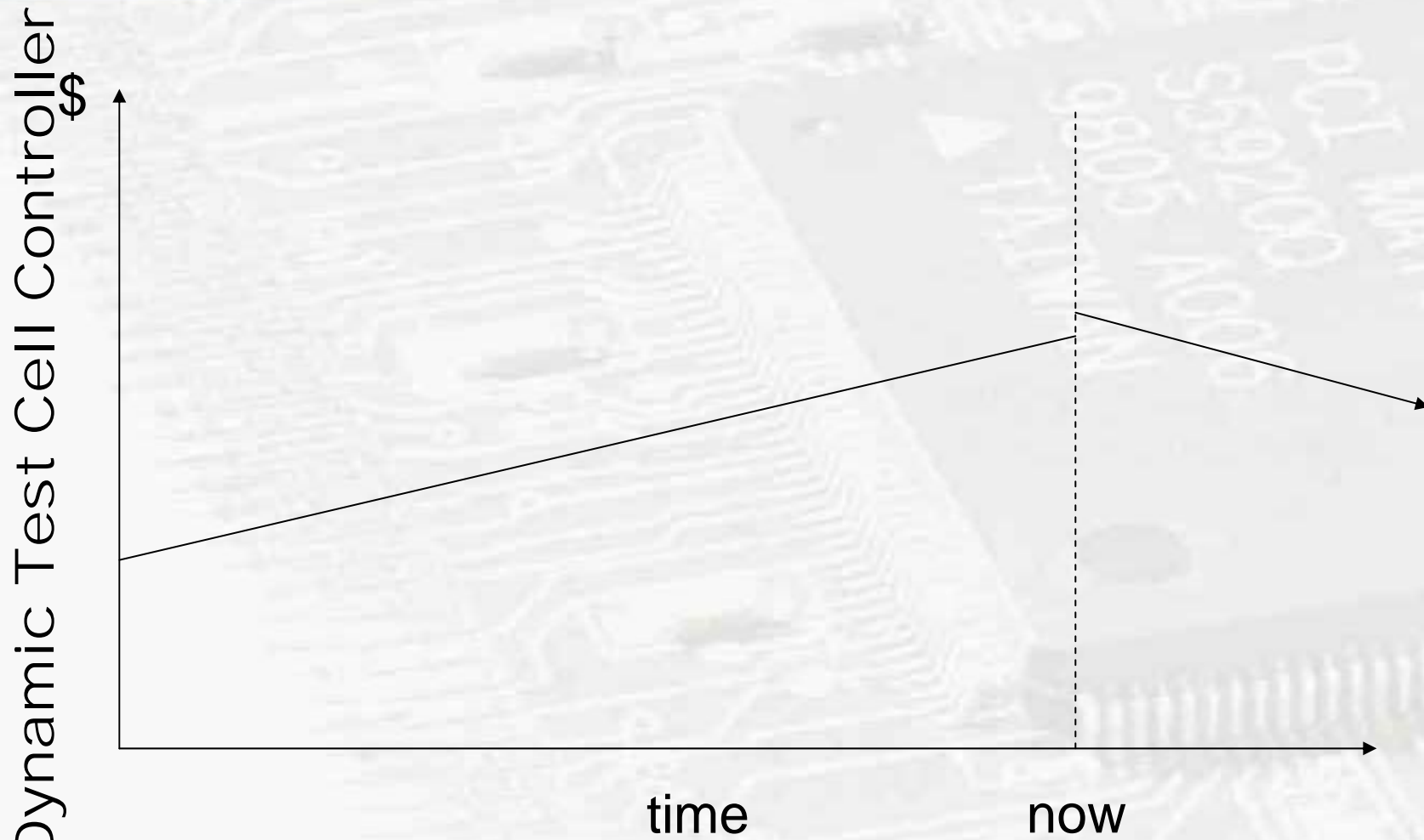
# Quality

- Touchdown monitoring
  - Per die amount of touchdowns in the map
  - Auto rebinning after exceeding of max. number of TD
  - Touchdown display map and report
- Probe card database
  - Keep track of touchdowns per card
  - Yield trend
  - Yield per site for multi-site probe cards
- PAT; outlier detection

# Epilogue

- By understanding the concept it becomes clear that the DTC will offer;
  - Efficiency improvements
  - Better test-cell utilization
  - Easily to migrate with “work stream” environments
    - Inkless & paperless wafer sort
  - Higher quality standard
  - Overall test cost reduction
  - A tool that introduce a new look on wafer sort
  - Future proof!!

# Cost of test



● **Thank you for your attention**