Emerging IC Testing Methods

Major changes are being driven by new Technologies and Quality & Cost requirements

Phil Nigh
phil.nigh@gLOBALFOUNDRIES.com
Purpose

• Overview of current testing methods
• Description of “emerging” methods that will change how we test over time.

Note that I’ll try to describe the most common industry methods (as I understand them).

There are many variations on Test implementations across the industry – so not all companies will follow the described details.
Overview of Today’s Testing Methods

• Example Test Flows & Test Coverage
  • Processor / SOC / ASIC Examples
  • Fault models & automatic test pattern generation
  • Automatic Test Equipment (ATE), Example Test Content

• Test Engineering Practices
  • Scan design & ATPG
  • Memory BIST & Repair
  • Test Data Compression (including support for diagnostics)
  • IO loopback testing
  • Structural vs. Functional Testing – use of “SLT” in chip production
  • Design Personalization & Process Monitoring
  • RF (mmWave) & Mixed-signal Testing
  • Fault Diagnostics & Yield Learning
  • What’s different about testing Automotive ICs?
Emerging Test Methods Changes

• Testing: the Role of a Foundry
• The changing “Value of Test”

What’s changing

• *Advanced Technology Failure Modes* – drives new testing, design & characterization methods
• *2.5D/3D & Heterogeneous Integration* – requires new industry test collaboration
• *Memory Testing*—changes driven by multichip applications, emerging technologies (e.g., MRAMs) and tuning for yield & reliability
• *“End-to-End” Data Integration* – enabling rapid Yield/Design/Test improvements
• *Adaptive Testing* – enables real-time, automated optimization
• *Product Reconfiguration & Adaptation at Test and in the Field* – which will improve yield, reliability and costs
Integrated Circuit (IC) Test Steps

Inline Test
During wafer fab -- test structures in kerf between chips ... optical inspections, etc.

Wafer Test
Test each die on all wafers

Package-level Test
Integrated Circuit (IC) Test Steps

During wafer fab -- test structures in kerf between chips ... optical inspections, etc.

This talk will predominantly be aimed at Wafer Probe Test & Final Test.
Purpose of Testing

• The primary role of Testing is to ensure all shipped components ("Integrated Circuits" == ICs) will work in the target application.
  • “Defect-free” (application perspective)
  • Including early life reliability defect acceleration

Emerging Testing Requirements

• Collect data that can be used to drive yield learning.
• Characterize the behavior of the product
• “Design Personalization” – modifying the details of the design to improve yield, reliability and power/performance. (more details later)
Miles of Wire, Billions of Connections
The Role of Testing

How to ensure every transistor operates as planned?

How to ensure each wire and the connections between wires are defect-free?

If there is a failure, how to localize the failure? (diagnostics)
Example Test Flow

**Wafer Test (Cold: 10°C)**
- Basic functionality check
- RAM test & repair
- Reliability voltage stress
- Process Performance Monitors

**Package-level Test (Hot: 90°C)**
- Full coverage tests (99%)
- Delay/performance Tests
- RAM test & repair
- IO parametric testing
Example Test Flow with Burn-in

**Wafer Test (85C)**
- Basic functionality check
- RAM test & repair
- Reliability voltage stress
- Ring oscillator measurements

**BURN-IN**
- High voltage (1.4X)
- High temp (140C)
- Exercise logic & RAMs

**Package-level Test (85C)**
- High Temperature RAM test & repair

**Package-level Test (10C)**
- Quick functionality check
- RAM test & repair

**Package-level Test (10C)**
- Full coverage tests
- Delay/performance Tests
- RAM test & repair
- IO parametric testing
Test Flow Options

- Some products use the same test content at each test step ... others use much different content.

- Temperature requirements?
  - *Typically try to cover high & low temperatures*

- Goal is to limit the number of test steps to minimize costs.
  - *Most products will require at least 2 test steps ... some will require more.*

- Burn-in is used to detect early life reliability defects by raising temperature & voltage for a few hours
  - *Burn-in is typically only required on high reliability ICs*
  - *May require a test before & after burn-in*
Types of Tests

• **Patterns**: … 0s/1s in & out of the chip
  • Logic & embedded memories
  • Applied at different “corners” (VDD, timings, …)
  • Includes RAM “repair” – replace defective cells with spares

• **Parametric tests**: measure voltage, current, speed

• **Core-specific tests**: PLLs, ADC/DACs, High-speed serial IOs, …

• **Reliability screens**: Artificially age chips … try to break weak chips (e.g., burn-in)
Automatic Test Equipment (ATE) Types

**Typical ATE costs $500K -- $2M**

ATE vendors: Teradyne, Advantest, Xcerra, ...

**Burn-in oven**
List of Typical Tests (SOCs, Processors)

- IO contact tests (open / shorts)
- Chip leakage (IDDQ)
- Logic testing (0s/1s) -- scan, functional
  - Slow (stuck-at), fast/delay
- Process Monitor structures (Ring Oscillators, etc)
- Electronic chip ID (ECID) tests
- Memory tests (mostly built-in self-test / BIST)
- IO parametric testing
- High-speed serial (HSS) IO testing
- PLL tests, Analog core tests
- High-voltage stress tests (reliability)
Defects / Faults

- Historically, production testing has mainly focused on defecting “spot defects”.

Additional material / shorts

- Gate oxide short
- Missing via
- Bad implant
- Missing metal/open
Fault Models

To automatically generate test patterns, the software must have a model representing defect behavior – called a “fault model”.

**Stuck-at fault model**

Each gate input & output can be permanently stuck at logical ‘0’ or ‘1’

(6 faults for NAND gate below)

**Transition fault model**

Used for AC/delay test generation. Each gate input & output can have a slow transition (rising or falling).
Automatic Test Pattern Generation (ATPG)

- **Structural Testing** – gate-level test of logic circuits by detecting a high number of modeled faults.
  - Stuck-at fault coverage: 97% -- 99.9%
  - Transition fault coverage: 80% -- 95%

- Patterns are automatically generated
  - 5,000 – 50,000 patterns

- **Scan design** converts the sequential circuits into a combinational one for ATPG by adding a ‘scan port’ to all flip-flops.
  - ATPG for sequential circuits has been found to be “computationally impossible” for today’s circuits (e.g., >50K logic gates)
Design-for-Test (DFT)

- DFT is used extensively on all modern logic ICs (processors, ASICs).
- DFT dramatically reduces test cost, shortens test time, reduces tester cost and improves diagnose-ability.
- DFT Examples:
  - Scan design
  - On-chip clock generation
  - IO boundary scan (JTAG)
  - On-chip loopback
  - Test data compression
  - Built-in self test (BIST)
  - Logic BIST, Memory BIST
  - On-chip RAM repair
  - Partial good (multi-core) reconfiguration
DFT Architecture

Scan Inputs

Chip Boundary

Boundary Scan Latch

Chip Logic

MBIST

RAM

Scan Chains

Register Array

FlipFlops

Scan Outputs

Test Inputs CLOCKS

Functional I/O

Functional I/O Test Inputs     CLOCKS

Scan Inputs
DFT Architecture

Scan Inputs

Chip Boundary

Boundary Scan Latch

All internal flip-flops are connected to a scan chain

Functional I/O Test Inputs

CLOCKS

Test Inputs

Scan Outputs

Functional I/O

FlipFlops

Scan Chains

RAM

MBIST

Register Array

Scan Inputs
DFT Architecture

Scan Inputs

Chip Boundary

Scan Outputs

Boundary scan is include on all chip inputs & outputs.
DFT Architecture

Boundary scan is included on all chip inputs & outputs.

This boundary scan structures enables almost all internal circuits to be tested without contacting these “Functional I/Os”. (enabling “Reduced Pincount Test”)
**Steps to apply a test pattern to test internal logic:**

1) **Scan data into the chip**
2) **Pulse Clocks to capture data**
3) **Scan out data**
DFT Architecture

Memory Built-in Self-Test (BIST) engine is used to test internally memories.
Memory BIST

- Embedded memories (SRAMs, TCAMs, DRAMs) all are tested with “memory BIST”.
  - Many different types of patterns … could be more than 20 types of patterns
  - Many different timing/voltage combinations … including VDD bumps up/down

- Virtually all embedded RAMs have spare bits (rows and/or columns) that can replace defective locations.

- For some implementations, BIST automatically calculates the repair solution and configures/blows the electronic fuses. (all done on-chip)
Embedded RAM BIST & Repair

- For some implementations, memory BIST & repair can be applied at each test step.
  - Some implementations only can repair once.

- RAM repair can be done at many different conditions (VDD, speed, patterns) at wafer and package test.

- Some memory BIST implementations are “programmable” (e.g., pattern content can be selected through on-chip programming) … some are fixed.
Memory Testing Patterns (examples)

- Blanket 0s/1s
- Checkerboard, Walking 1s/0s,
- Marching 0s/1s/X/C-/A/Y/B
- GALPAT
- WordLineStripe
- BitLineStripe, Repeating Read/Write
- Diagonal, Sliding Diagonal
- Butterfly
- Stability Patterns … Bitline / Wordline
- Neighborhood Pattern Sensitive
- Data Retention (write … wait ................. read)
- Toggle Pattern
- N-square (not practical for larger memories … too many patterns)
Logic Testing – Test Data Compression

• A difficulty of scan-based logic testing is the large amount of data volume needed. (and test time)
  • *Scan data volume* = number of test patterns $\times$ number of scan latches

• The industry has moved to “test data compression” – encoding test data.
  • *This reduces both test data volume and test time*
  • *Today: 50X—100X reduction in test data volume and test time is common.*

• Logic BIST is a form of test data compression – patterns are generated on-chip and output data is fully compressed on-chip.
  • *May require many “test points” to achieve reasonable coverage.*
On-Chip Data Compression Options

**Input Side**
- “Scan-input Fanout” – with one scan-in pin – distribute data to multiple scan chains.
- Pseudo-random pattern generation (PRPG) – generate input data on-chip using series of flip-flops with XORs and feedback paths.

**Output Side**
- Multiple input shift register (MISR) – similar structure to PRPG – but also takes inputs from each scan chain to create a unique signature.
- XOR gates – multiple scan chains drive XOR scanouts to reduce number of scan output pins.
On-chip multi-input shift registers (MISR) compress output data into a relatively short signature.
XOR Output Compression

XORs can take multiple scan chain outputs and condense to 1 scan output pin.
On-Chip MISR + Scan-in Fanout

Scan Inputs

Fanout out a single scan input to many internal scan chains

Functional I/O

Test Inputs

CLOCKS

Functional I/O

Chip Logic

RAM

ABIST CNTRL

SRLs

Scan Chains

GRA

MISR or XOR

Fanout network: 24 ->48 -> 384

Fanout out a single scan input to many internal scan chains
On-chip pseudo-random pattern generators (PRPGs) generate input data. (data looks like stream of random 0s & 1s)
Logic BIST (LBIST)

After initializing PRPG & MISRs – LBIST can generate many test patterns – and compress all outputs – by just applying millions/billions of clock cycles.
Logic BIST (LBIST) with weighting

Basic PRPGs produce a random sequence of 0/1 patterns.

Weighting circuits can be added to produce a higher probability of 0s or 1s.
“Weighting” can improve fault coverage and reduce the amount of switching. (reducing dynamic power)
Logic BIST

- LBIST is becoming much more important since it is being used in automotive ICs.
  - ISO26262 requires some form of “online testing”. LBIST can be run at engine startup … and when auto is turned off. In addition, a short LBIST (e.g., 20ms) can be regularly run when the auto is in normal operation. (IC pauses normal function for 20ms … runs LBIST … then restores/continues)

- LBIST offers the benefit of virtually no test data volume.

- BUT … LBIST takes more patterns to achieve the equivalent fault coverage.
Fault Coverage vs. Patterns

Fault coverage

Deterministic patterns

100%

Number of patterns
Fault Coverage vs. Patterns

Fault coverage

Deterministic patterns

LBIST patterns

Number of patterns
Test Background – Functional vs. Structural Testing

**Functional Testing**

- A data
- B data
- Traffic
- Trash
- Boot Op Sys

**Structural Testing**

- Scan Tests
- Logic BIST (LBIST)
- JTAG
Evolution of Test Methods

Structural vs. Functional Testing

• “Many years ago” functional test was the standard for production testing.
  • BUT ... no ATPG or diagnostics ... coverage too low ...
    ATE requirements expensive/unpractical.

• Then “structural testing” dominated. (for most digital applications ... ASICs/SOCs/DSPs/GPUs)
  • sometimes supplemented with a little bit of functional test

• Now functional testing is making a comeback – e.g., System-level IC test (SLT).
  • Plug Holes in Structural test coverage
  • Better performance/power verification testing
  • Lower risk during bring-up (no time for learning)
  • Quicker path for closing test holes (found in system testing & application)
Fault Diagnostics

• Diagnosis is the process for localizing the defect (or failing location) to a relatively small area.
  • small enough for PFA to find the defect

• For high-volume products, companies typically do “volume diagnostics” – e.g., localize the failing circuit for a ‘few hundred’ … or maybe a ‘few thousand attempts / day.

• Emerging Diagnostic Methods: Cell Aware fault model, layout-driven diagnostics, bridging fault model, merging with fab data … broading support.
Volume Diagnostics

- Fabs today require diagnosing a large number of logic failures to drive yield learning (e.g., 1000 / day)

- Global Foundries (among others) are moving toward “layout-aware” diagnostics
  - Previously, failures were only localized to gate inputs/outputs.
  - Cell Aware fault model

- Ideally, volume diagnostic data collection would be based on fast fail data collection on the tester using the same patterns as used for “pass/fail” testing.
  - OPMISR+ … industry view is that it is not possible to perform diagnosis based only on MISR fail data collection
  - Today, some in industry do volume diagnostics based on a small set of full scan in/out, stuck-fault test patterns (e.g., 200-500).
  - Diagnostics based on XOR test data compression fail data – emerging in industry as good opportunity
RF, Analog and mmWave Testing

- **Microwave & mmWave Capability**
  - Requires 3d Modeling, DSP/FPGA Processing
  - Sockets, Test Boards, Connectors
  - 86GHz – Tester Requirements
  - High Frequency Calibration

- **Microwave Test Solutions**
  - 12GHz – 86GHz Backhaul in MFG
  - Wafer and Module Solutions

- **Custom Test Development**
On-Chip Process Monitors

- Most ICs today contain additional structures not required by the application to monitor the fabrication process. *(some of these structures may also be in the area between dies on the wafer)*

- **Example Structures:**
  - Ring Oscillator for transistor performance
  - Wire / via structures to monitor the “back end” processing
  - Example circuit paths – not used by the application – only for monitoring performance degradation during life. *(e.g., NBTI)*
• Odd number of inverters in a ‘ring’ will cause continuous waveform that is easy to measure.

• Multiple configurations: transistor $V_t$, wiring/load, chip location, …
Reliability Defect Screening

- Reliability defects are ones that occur during the IC life.
- These defects can be accelerated by raising VDD and temperature.
- Burn-in operation raises voltage & temperature to accelerate these defects.
  - Burn-in durations may be 1 hour to 24 hours.
  - Many devices in parallel in burn-in oven.
- A smaller amount of acceleration can be achieved by just raising VDD during production testing.
  1. Raise VDD and apply patterns (e.g., Memory BIST and logic stuck-fault patterns)
  2. Static VDD bumps (no patterns applied)
Product Personalization

- Set VDD per IP block (core)
  - e.g., 220 cores .. each with tunable VDD

- Set FMAX per IP block

- Partial good configuration
  - This could be much more complex than shown previously – e.g., combinations of embedded PUs & embedded memories may only support limited combinations.
  - Also, an option is to characterize cores on-the-fly and select the ones with the most margin. (reliability)

- Well-biasing per IP block (e.g., FDSOI)

- On-chip monitors (thermal, VDD) can drive real-time adjustments (both at Test and in application) to FMAX, VDD, Well-bias.
Emerging Test Methods
Historically, the “fabless” partner has had responsibility for Test. (test programs & development)

As technology advances, the foundry will have a bigger role in Testing (development & execution):

• Foundry should best understand failure modes and the preferred test methods for detecting these faults.
• The foundry plays a key role in yield learning – and should help guide test developers in best methods for data collection. (including unique methods)
• Some fabless customers will want test development help and Post-fab supply chain management
  • Particularly if foundry has unique expertise in advanced technologies & testing (e.g., 2.5D, SiliconPhotonics, mmWave)
Some advanced technology failure modes can be really difficult to drive to root cause.

For example, failures caused by circuit marginalities, process variation ... plus test sensitivities.
Testing: The Role of the Foundry?

• Some advanced technology failure modes can be really difficult to drive to root cause.

• Foundries MUST work with customers to ensure that testing can provide data to get to root cause as quickly as possible.
Test at GLOBALFOUNDRIES

- Burlington, VT
- Dresden
- Malta, NY
- Fishkill, NY
- Singapore
### Collaboration Drives GLOBALFOUNDRIES Packaging

*Dave McCann, SWTW 2015*

#### Design Capabilities
- Co-Design Reference Flows
- DFx support and tools
- Validated Interposer reference flow
- DFT enabled interposer flows

#### OSAT Partnerships
- JDAs at leading edge nodes
- Innovative business models
- 14LPP 2D, 2.5D & 3D Package qualifications at multiple OSATs
- 2.5D package qualified with GF interposer
- 3D TSV package qualified
- Seamless supply chain models in setup for 2.5D products

#### Internal Capabilities
- Bump & probe
- Turnkey solutions
- Investment in emerging technologies
- >15 years experience providing turnkey solutions
- 2.5D and 3D TSV enabled at Leading Edge

#### Memory Partnerships
- Logic-Memory Demonstrators
- Innovative business models
- 2.5D HBM integration demonstrator
- Seamless supply chain model in setup

---

Korean Test Conference

June 2016
Test industry drivers

Evolution of the International Test Conference

Up through the late 1990s

Test Equipment
(Teradyne, Advantest, Xcerra, ...)

~Year 2000 to ~2013

Design-for-Test EDA Companies
(Synopsys, Cadence, Mentor Graphics)
The Changing Value of Proposition of Test

... it is no longer just to identify good/bad die

- Design personalization
  - Power/performance optimization, tuning, configuration for yield/reliability, die ID

- Data collection at Test drives:
  - Adaptive Test, Adaptive back-end flows
  - Yield Learning, product bring-up/verification
  - Optimal build/supply chain for multichip products
  - Production test optimization: test time, delivery, data collection, equipment utilization, retest/equipment monitoring & maintenance

- Advanced diagnostic/yield learning methods
  - including on-tester, adaptive data collection
Changing Failure Modes

• If the predominant failure modes change from *Mostly physical spot defects* \(\rightarrow\) "soft" failures

• then, card & system-level verification should change

1. Characterize margins across process region during debug/development

2. Closely check guardbands (*VDD, speed, temperature*) when defining production flows

3. During volume production – routinely characterize margins (*both passing & failing parts* … *and parts which fail downstream tests*)
Many Embedded Cores (many identical)

66 identical PUs ... 64 must be good at IC test

152 identical PUs ... 148 must be good at IC Test

PU = Processor Unit
Testing Multicore Designs

• Test data reuse ... generate data for a single embedded core ... reuse for each instance

• “Complete” test reuse configuration for each core instance – full configurability on the tester (e.g., test one-at-a-time, test all in parallel, option to test all possible options)
  • On-the-fly characterization/fail flows
  • Power/thermal/VDDdroop/noise analysis

• Bookkeeping / configuration for partial good support (communication out thru the field)

• Diagnostics/yield analysis capability that understands cores & configuration.
  • Intelligent, on-the-fly determination of data dump options
  • Ideally, no additional test passes for diag. data collection
Yield Analysis of Multicore Designs

Virtual Yield Learning Structures  (CMU, Maly)

66 identical PUs … 64 must be good at IC test

152 identical PUs … 148 must be good at IC Test

Good
Average
BAD
Embedded Circuitry for Tuning, Characterization, and Diagnostics

Emerging chip designs have “hundreds” of these circuits today … in the near future, there will be “thousands”.

Sensor types

- Voltage monitors
- Thermal measurements
- Vdroop measurements
- Process monitors (e.g., ring oscillators)
- HSS eye measurements
- Clock tuning buffers
- Memory margin tuning
- Noise injectors
Embedded Circuitry for Tuning, Characterization, and Diagnostics

- A key direction is to use all of this circuitry at each level of testing – including card/system/field.

- At most test steps, only a sample of measurements will be taken.

- Store this data … “forever”.

- For characterization (bring-up, RMAs), one should be much more thorough about collecting all data.
Embedded Circuitry for Tuning, Characterization, and Diagnostics

- This circuitry will also be used in the field – for monitoring, data collection, diagnostics and field adjustment.

Test data from each device at each step will be stored ‘forever’.

This enables statistical analysis of data whenever there are Yield or Quality or Reliability issues.
Field IC Adjustments / Reliability

- On power-up ... run BIST and online tests ... check VDD/FMAX margins ... adjust tuning circuitry.
- Real-time thermal & VDD monitoring ... on-going adjustments of VDD, FMAX, software codes & well-bias.
  - Given a thermal limit ... maximize performance (VDD, FMAX)
- Lifetime performance monitoring ... adjustments.
  - e.g., on-chip circuitry measures functional speed ... NBTI degradation ... then adjust VDD to compensate
- “Phone home” data collection. (e.g., margin char.) Repair components before they fail. (Obviously, this is only for some applications.)
- Logic & Memory reconfiguration around failures.
  - e.g., PU redundancy, remove/replace Memory on fail
Test Program Verification

• Given all this personalization & real-time adaptability – how can we verify all modes during production testing?

• And how to verify test program correctness?

For example, mobile processor ramps in 3 months – how can we verify test program is “perfect” in first 2 weeks?

This is a really hard problem
Test Program Verification

• Test must exercise all modes and verify all adjustments done at Test.  (at least for now …)

• Good test program development practices
  • Test engineers are mostly software engineers today
  • Agile Methods for Test Development

• Test code reuse, test reuse, test reuse, …

• System-level Verification
  • Not just “does it work” – but check functionality margins  (test to fail)
  • Close collaboration with System Verification/Qual

• SLT in Verification  (and often in production)
Adaptive Test on Steroids
Adaptive Test Flow -- Single Test Step

Feedforward data
Data from previous test steps, inline test, historical test results, Off-line analysis results, Production/Supply Chain data

Real-time analysis
Yield, fallout/bins, parametric data, Production results

Test

Post-Test
Analysis results (outliers), dispositioning, product flows, target applications, future test optimization
“RT A/O” stands for “Real-Time Analysis & Optimization”

“PTAD” is “Post-Test Analysis & Dispositioning”

**Adaptive Test Flow [ITRS]**

- **ETEST**
  - Fab data

- **PTAD**

- **Wafer Probe**

- **Burn-in**

- **Final Test**

- **Card/System Test**

- **Field Operation**

**Database & Automated Data Analysis**

(including post-test statistical analysis, dynamic routings and feedforward data)

- **Assembly Ops.**
  - Includes build operations at any level of assembly

- **Fab data**
- **Design data**
- **Business data**
- **Customer specs**
Adaptive Test Applications

• There are 100s of Adaptive Test Applications that have been published in the literature.
  • *International Technology Roadmap for Semiconductors* ([www.itrs.net](http://www.itrs.net)) has many examples.

• These applications can:
  • *Lower production costs*
  • *Improve product Quality & Reliability*
  • *Enable unparalleled opportunities for test data collection to drive yield learning*

• I’m not going to try to summarize all applications here.
“Statistically more Reliable ICs”

Red chips are fails …

black & green chips are passes.

This device

... is statistically more reliable than these three devices
Analysis drives test flow Adaptation

*Skip burn-in or skip cold socket or add SLT -- based on automated analysis*

### MULTIPLE FLOWS

- **Wafer Probe Test**
- **Die-level dispo & routing**
  - **Test**
  - **BI**
  - **HOT**
- **Post-test dispo**
- **Stock**
- **Demand-targeted fusing**
- **SLT**
  - **Ship**

*System screeners, sample only*

[Partly based on Intel article, ITC06]
End-to-End Data Integration & Continuous Analysis

There is broad agreement that we need to integrate all IC data ... design ... fab process ... test data ... through card & system test ... then field operation.

BUT ... existing solutions are mostly based proprietary databases and/or only supports a subset of available data.
Will we all be using the Cloud to store & access the End-to-End Data?
End-to-End Data Integration & Continuous Analysis

- Note that a number of different parties will be collecting this data and putting it into the database.

- Also, no one company will own all data ... or even have access to all of it. (and a number of different companies will need access to a portion of the data)

The purpose of this data integration is to be able to analyze all data – enabling unprecedented analysis leading more efficient manufacturing, better product quality & reliability and faster root cause analysis.
End-to-End Correlation & Optimization
automated ... continuously adapting & optimizing

Common Database

Automated Analysis & Actions

Repeated daily
How will test equipment be changing over the next few years?
What does ATE look like?

- I don’t see major changes in ATE architecture
  (.. but reduced pincount and performance resources)
  - Simple/cheap digital pins (configurable)
  - **on-chip IO parametrics ... no external loopback for very high speed IOs** (on-chip only)
  - Limited pincount Test Port (flexible)
  - Flexible power supplies (possibly many / DUT)
  - Data collection capability (digital fail capture, scan data dump ... fast/off-line data dump to storage)
  - More on-the-fly analysis, **diagnostic/digital data tweaking**
  - **ATE-offline analysis engine real-time co-Analysis**
  - RF/high speed options if needed
  - Integration options of System-level test (SLT) plug-in cards

- **Fixtures:** fine pitch probing options (cost), die-level options, 2.5D/multichip handling
Heterogeneous Integration (HI)

- HI is the integration of separately manufactured components into a higher level assembly that in the aggregate provides enhanced functionality and improved operation characteristics.
  (and ideally lower costs)

- Also ... ideally ... enable “plug & play” ... any component can be replace with newer technology version of that die.

IRDS contact for HI:
dave.armstrong@advantest.com

- CPU
- NVRAM
- DRAM
- RF/mmWave IC
- ASIC / SOC
Heterogeneous Integration

- Known Good Die – silicon providers supply bare die with same quality as packaged die.
- At Package Test – we can test each part with “(almost) full content” from wafer probe test.
- Silicon providers will share all required Test information (patterns, conditions, BIST, etc.) to enable the “Integrator” to fully test their silicon at Package Test.

Who is the “Integrator”? Who is primary interface with all silicon providers … including guaranteeing Reliability & Quality … and manage yield across all silicon suppliers.
3D Packaging

• True “3 dimensional” packaging is coming.
  • Transistors at each level

• Future ICs will have 10s of 1000s of connections between active layers.

• Power management, Test & yield management are critical issues.

• Do we test at each layer?
• How?
When does Test become easier?

• Designs have a lot of adjustment capability & adaptability … and the ICs can reconfigure around many failures …

• So at some point if we trust this capability … we can start to test less thoroughly. (after product verification which demonstrates these functions are correct)

For example:

• Cost of an embedded core failing in the field is much less if there are spares (lower coverage requirements)

• Low Voltage margin … automatically is adjusted in the field to avoid low voltage failures (less guardband needed at Test)
Ideal System

- On-line testing
- Fault-tolerant
- Self-healing
- Graceful degradation
- Redundant systems
- Avoid ‘unrecoverable faults’
- Field adjustable
- Automated diagnostics

[Mark Barber]
Summary – Changes in the next few Years

• Test will drive design adaptability & reconfiguration for power/performance & reliability improvements.

• Data collection at Test will enable Adaptive Test, Test Optimization, End-to-End Correlations improving Costs & Quality & Reliability & Yield.

• Multichip technologies are coming.