
Introduction to IC Test

ECE 575 / 675

Chapter 2 & 3

Professor R. Daasch

Department of Electrical and Computer Engineering
Portland State University
Portland, OR 97207-0751
(daasch@ece.pdx.edu)

<http://ece.pdx.edu/~daasch/course/x75>

Outline

- ❑ Introduction to Component Test
- ❑ Automatic Test Equipment (ATE)
- ❑ Test Requirements and Specifications
- ❑ Device Under Test (DUT)
- ❑ ATE Program

Component Test

- ❑ Determines whether manufactured chip meets specs
- ❑ Must cover high % of modeled faults
- ❑ Must minimize test time (to control cost)
- ❑ No fault diagnosis
- ❑ Tests every device on chip
- ❑ Tests every device on wafer
- ❑ Test at speed of application or speed guaranteed by supplier

Automatic Test Equipment

- ❑ Functional testers (comparison testers) match DUT response to input to expected output
- ❑ Primary Goals of IC ATE (no surprises here)
 - Test at higher frequency
 - Test higher pin counts
 - Test at lower cost
- ❑ Wafer sort or probe test — done before wafer is scribed and cut into chips
- ❑ E-test site characterization — specific test devices are checked with specific patterns to measure:
 - Gate threshold
 - Polysilicon field threshold
 - Poly sheet resistance, etc.

Automatic Test Equipment

- Test Requirements and Specifications
 - Functional Characteristics
 - Type of Device Under Test (DUT)
 - Physical Constraints — Package, pin numbers, etc.
 - Environmental Characteristics — supply, temperature, humidity, etc.
 - Reliability — acceptance quality level (defects/million), failure rate, etc.
 - Test plan generated from specifications
 - Type of test equipment to use
 - Types of tests
 - Fault coverage requirement
- ATE should provide a practical (expected) working environment for DUT

Automatic Test Equipment

- ❑ ATE is generally limited to technology “less than” the DUT
- ❑ Board and Integrated Circuit (IC) Test Equipment differ on at least two points
 - Board test has more access to internal signals; IC tests are pad limited
 - Board test uses In-Circuit-Test (ICT) to identify bad components; IC tests identify circuits that fail to meet specifications
- ❑ Comparison Testers - compare “gold” device and Device Under Test (DUT) response
 - Functional test by comparison
 - Slower speed, lack flexible timing
- ❑ Stored Program Testers - compare DUT to stored patterns obtained from simulation or measured “gold”

Automatic Test Equipment

- Memory ICs use conventional, well-defined patterns which require less control
- Static tester — substantially a logic analyzer
- Dynamic test — aka ATE in this course
 - Higher initial cost - timing flexibility, speed, control and, data analysis
 - “Shared resource” testers multiplex voltage sources or timing generators on test channels (pins)
 - “Tester-per-pin” tester channels are identical and independent reducing programming demands (MegaTest 1983), 50-70% of ATE resources are per-pin
 - “Test Head” must be close to DUT reduce timing delay and timing mismatch
- Three economic factors for IC Test

Automatic Test Equipment

- Cost of Test — Test that is slow (eg long test time) or expensive (eg tooling) may price products too high
- Time to market — Test that slows product introduction may mean a missed market window
- Time to money — Test costs can be recovered by increasing yield or reliability
- Design for testability (DFT) generally relates to IC design, but it also affects ATE.
 - As much as 95% of new designs use full scan DFT
 - A new class of DFT testers may drastically reduce the cost of test.
 - DFT tester uses the SCAN port connections
 - Reduce or eliminate functional testing from IC's I/O pins
 - ICs with extensive use of DFT techniques, reduce or eliminate need for more than a DFT ATE

Automatic Test Equipment

- DFT-aware ATE integrate built-in self-test (BIST) resources into the architecture.
- Access to the on-chip BIST circuitry merged into the ATE software
- No operational distinction between software tools, instrumentation and software support in ATE.
- DFT-ATE reduce cost
 - Fewer tester pins are required because, only the scan I/O and control pins contacted
 - Relaxed scan data and clock rates can be satisfied at lower cost
 - Lower total number of DUT pins contacted, the probe card can be less expensive.
- Uses of ATE test data

Automatic Test Equipment

- Reject bad DUTS
- Fabrication process information
- Design weakness information
 - Devices that did not fail are good only if tests covered 100% of faults
- Failure mode analysis (FMA)
- Diagnose reasons for device failure, and find design and process weaknesses
- Allows improvement of logic & layout design rules

Building Blocks

- Each building block has requirements of accuracy, resolution and sensitivity
 - Accuracy — measurement uncertainty
$$\text{value} = \text{measured} \pm 0.1 \cdot \text{measured} \pm 0.05 \cdot \text{range}$$
 - Resolution — degree of observed change as trade-off with speed (averaging to reduce effects of random noise)
 - Sensitivity — absolute detection limit
- Device (Unit) Under Test - the DUT, aka Circuit Under Test (CUT)
- Test Head - enclosure(s) for front-end electronics close to DUT
- Mainframe - sheet metal for back-end tester electronics, power distribution

Building Blocks

- ❑ Central Host Unit (CPU - Sparc Unix, WinTel)
- ❑ Pattern Generation
 - Store and Sequence control of pattern vectors
 - Special patterns from algorithms or scan
- ❑ Timing Generation
 - Precision timing edge generation
 - Driver format - response strobes
- ❑ Analog References - waveform amplitude, compare levels, load currents
- ❑ Pin Electronics
 - Controlled impedance for DUT
 - Drivers/receivers, compare levels and load currents

Building Blocks

- ❑ Precision Measurement Unit (PMU)
- ❑ Force voltage/current, measure current/voltage to any DUT pin
- ❑ Device Power Supplies - programmable voltage, measurable current
- ❑ Time Measurement Unit - clocks
- ❑ Picoammeter - low (eg device leakage) current measurements at DUT pin

Building Blocks

□ Example, ATE Costs

ATE	\$3M, \$300/yr
ATE Operator	\$17.50/hr
Test Cell Cost	\$35/hr
Programming Cost	\$173K/yr
Maintenance Cost	\$135K/yr

Run ATE for 10 years, 80hrs/wk, 52wks/yr → \$826K/yr operating costs

Set 80% yield and test cost \$0.10/part → 10.3M parts/yr

Allow 10% system down time → 1.3 sec/part

Allow 0.3 sec setup/part → 1 sec test time

Chip clock 200MHz → maximum 200 million test patterns

Exhaustive test → < 28 input bits of combinational logic

ATE Timing

- Timing sets drive or compare on a I/O pin
- Using SCx12 as an example - each pin has six timing edges
 - Drive edges - 2
 - Tristate edges - 2
 - Two compare edges - Window mode (level sensitive), Edge mode (strobe high/low)
- Drive is formatted by timing and pattern data
 - NRZ/NRZI, RZ/RZI, RO/ROI, SBC (surround by complement), RZX (force return to zero), ROX (force return to one), Fixed (high/low)
- Pattern Generation
 - Sequence Control
 - Waveform Selection

ATE Timing

- Pattern Mux Selects from
 - Memory Store (millions of vector bits);
 - Interleaved memory restricts real-time sequencing
 - Memory can be loop (thousands of vectors); search for trigger
 - Algorithmic Generator
 - Scan Memory

ATE Timing

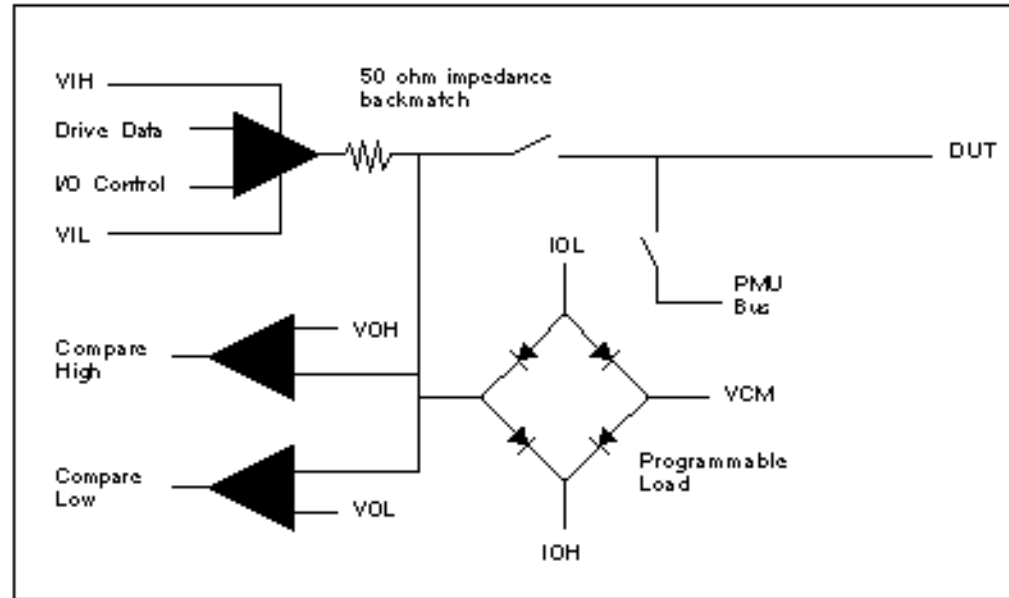


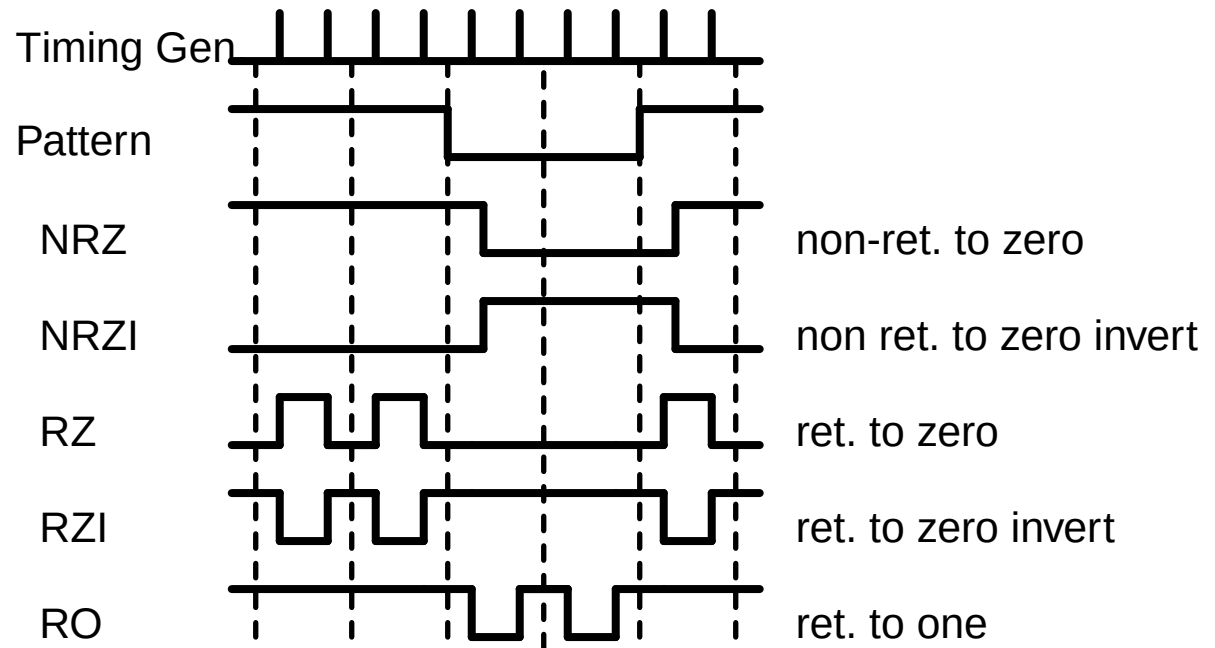
Figure 1-4. Pin Electronics Block Diagram

- Available drive formats largely determined by pin electronics
 - Format based on pattern data/timing generator
 - Markers are measured from T_0 (master clock) and placed based on drive and strobe requirements

ATE Timing

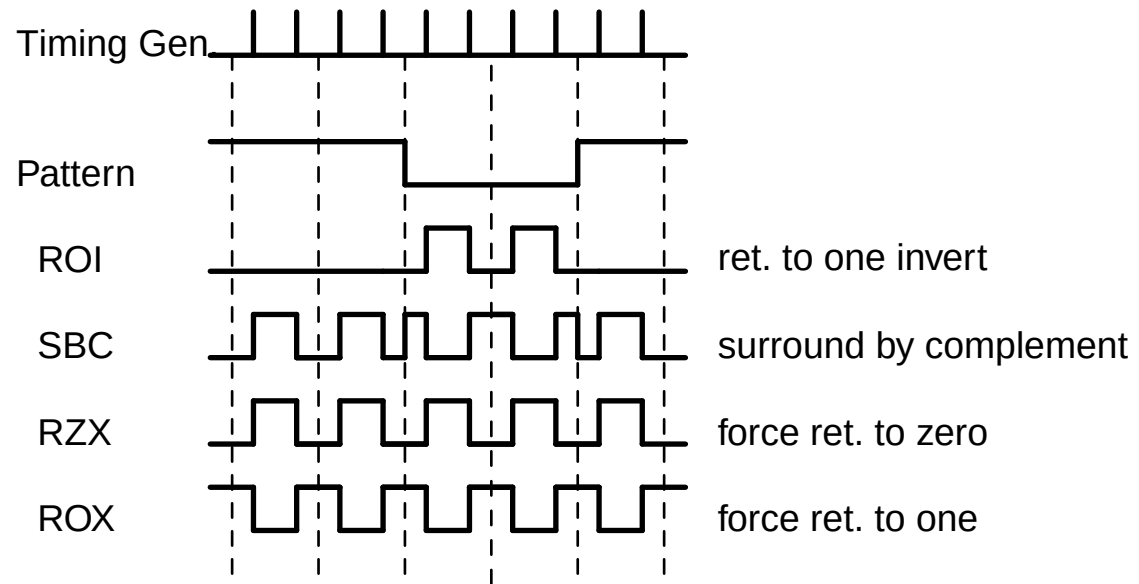
- More markers, more events from pattern
- Format events are controlled by rise(fall) of timing generator
- Output strobes define when DUT output are monitored/compared
 - Window - (open/close pairs) define a time slice
 - Edge - high or low comparison

ATE Timing



- RZ - Return to Zero and NRZ - Non-return to Zero - low demands on ATE resources but may take two or more executions

ATE Timing



- ❑ SBC - Surround by Complement - used for setup and hold time validation
- ❑ Note that an extra transition (at cycle boundary) can be generated by SBC
- ❑ Running \$2.56M ATE for 99% of the time for 10 years costs \$0.008/sec

ATE Timing

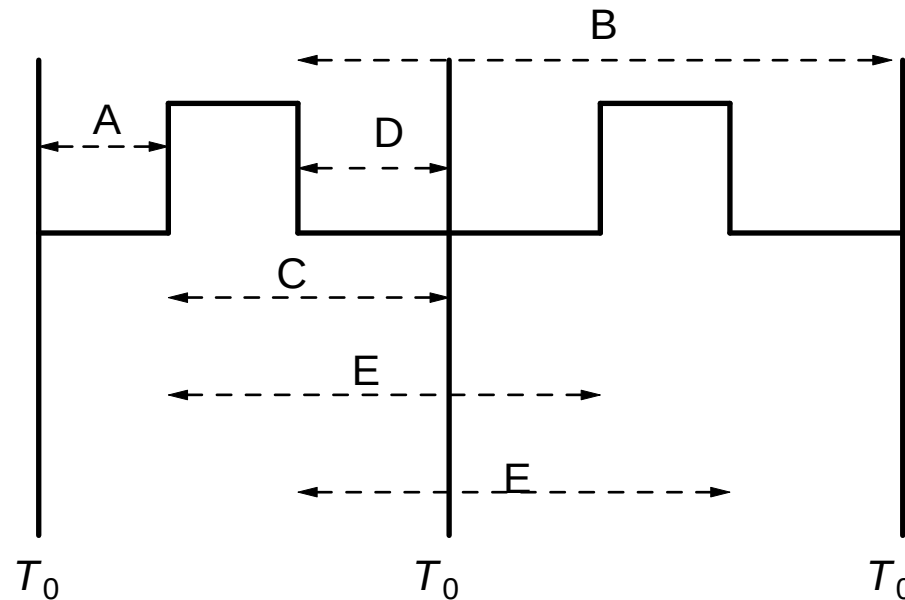
- ❑ Worst test time reduces throughput 3% which with \$50 parts and 30% margins decreases profits by \$450K/1M units/year
- ❑ Best to worst test ordering results in \$14K/1M units/year test cost
- ❑ Slower tests such as I_{ddQ} need fastest test applications (ie use functional patterns vs scan for I_{ddQ})
- ❑ Where to put Built-In-Self-Test (BIST) in a low latency test flow? Self-Test Using MISR/Parallel (STUMPS)
- ❑ Drive formats are the templates
- ❑ A non-type format not defined from T_0 to t_1
- ❑ They are combined with pattern bits to generate realistic conditions during test
- ❑ Markers are referenced to T_0 , the cycle boundary

ATE Timing

- Each resource is scheduled by the test program
- Some ATE replace timesets and formats with event, time pairs

ATE Timing

□ Timing Specifications



- Using SC specifications as an example (all times in nanoseconds, 10^{-9} s)
 - Each timing edge has a range of where in the timeset it can be placed

ATE Timing

- $A \equiv \text{Leading edge Normal} = 0 \text{ to } (2 \times \text{period} - T_{PW}) - 5$
- $C \equiv \text{Leading edge Match (ie trigger)}$
 $= 0 \text{ to } (\text{period} - T_{PW}) - 5$
- $B \equiv \text{Trailing edge Normal} = 0 \text{ to } (2 \times \text{period}) - 5$
- $D \equiv \text{Trailing edge Match (ie trigger)} = \text{Lead} + T_{PW} \text{ to period}$
- $E \equiv \text{Retrigger} = 14$

ATE Program

- ❑ Many different formats for test information
 - Format for the interface between CAD and IC vendors
 - Format for the interface between IC vendors and ATE
- ❑ Consortium formed to address growing concerns with large volumes of digital test data
- ❑ STIL common language for transfer density digital test patterns from simulation, automatic test pattern generation (ATPG), built-in-self-test (BIST), and ATE
- ❑ From http://grouper.ieee.org/groups/1450/Flyer_frame.html

With a standard between EDA, IC, and ATE environments, the generation, movement and processing of test data is greatly facilitated. This facilitation addresses a key issue with the handling of large digital test programs by defining a common format to contain this data.

ATE Program

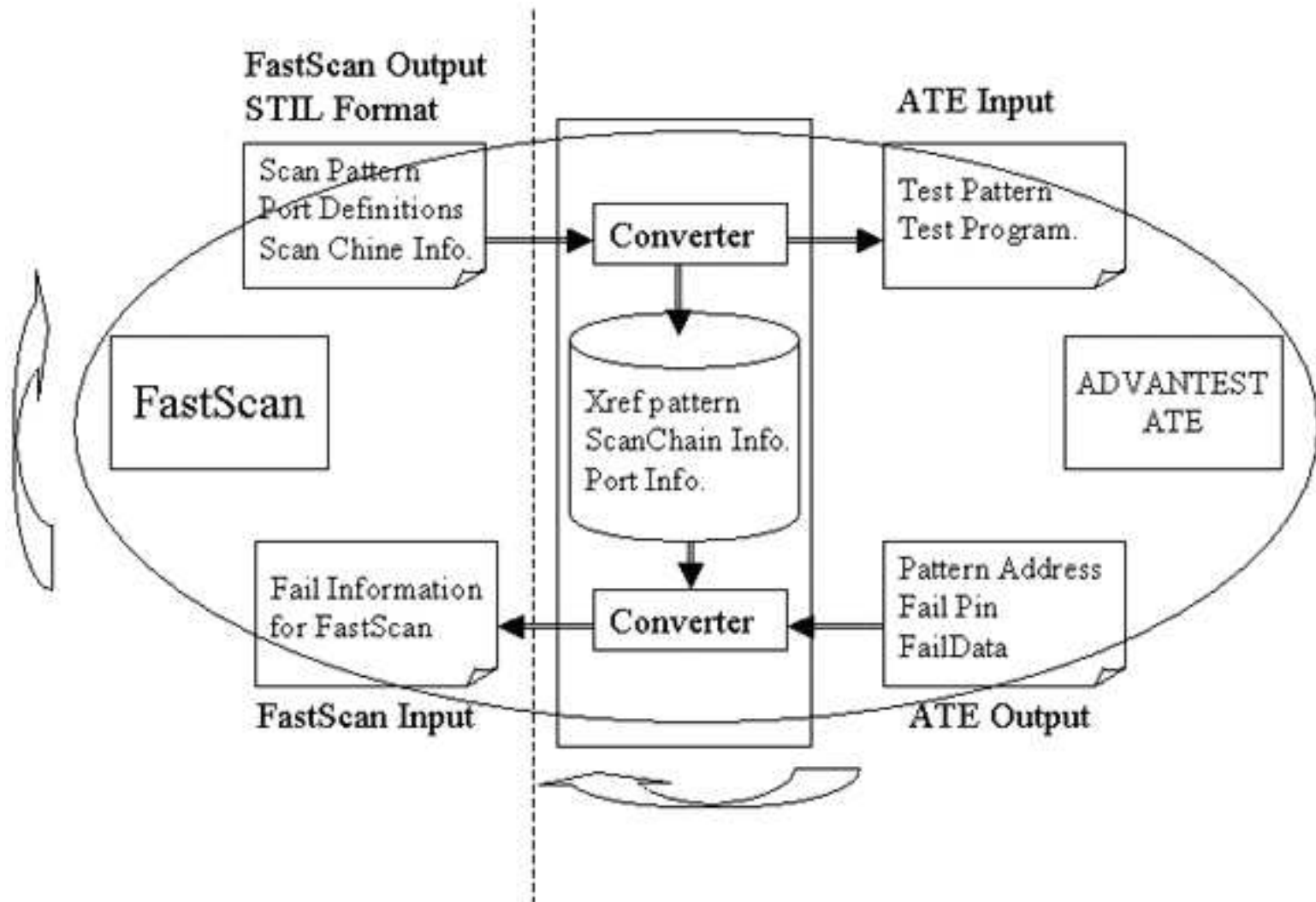
A standard also allows for immediate access to test equipment supporting this standard. It benefits both ATE equipment manufacturers and IC vendors, removing the need for internal test interfaces supporting proprietary languages, and eliminating the inertia present when these interfaces need to be modified to support different equipment.

Finally, a cohesive standard allows for the creation of shared toolkits, generation environments, and identification of commonality in the test flow between vendors and in the test environment overall.

Ultimately, this will benefit the IC designer as vendor-specific flows are replaced with common tool sets, and designers can operate on common data rather than data formatted for specific vendors.

- A sample of how STIL is positioned in the data flow is between Advantest (ATE) and Mentor Graphics (EE CAD)

ATE Program



ATE Program

- Test programs generally (not just STIL) similar to VHDL Testbench
 - Define I/O, power and reference voltages
 - Define labels for groups of I/O
 - Set working voltages
 - Set clock period
 - Set drive formats/output strobes
 - Load pattern data
- Loops and algorithm controls

ATE Program

- Basic (Never) Rules for Test programming
 - 1) Never test inputs as outputs
 - 2) Never float an input (see 1) (CMOS latchup)
 - 3) Pins voltages between power and reference (CMOS latchup again)
 - 4) Limit current on voltage force (drive)
 - 5) Limit voltage on current force (drive)

ATE Program

- Common Test Sequence
 - Probe test (wafer sort) — catches gross defects
 - Contact electrical test
 - Functional & layout-related test
 - DC parametric test
 - AC parametric test
 - Unacceptable voltage/current/delay at pin
 - Unacceptable device operation limits

More on Drive Formats

- DUT Test Specifications
 - Pin and package requirements (input/output, power)
 - Working voltages
 - Input conditions (V_{IL} , V_{IH})
 - Output conditions (V_{OL} , V_{OH}) (I_{OL} , I_{OH})
 - System and internal timings

ATE Hardware

□ Pin Electronics

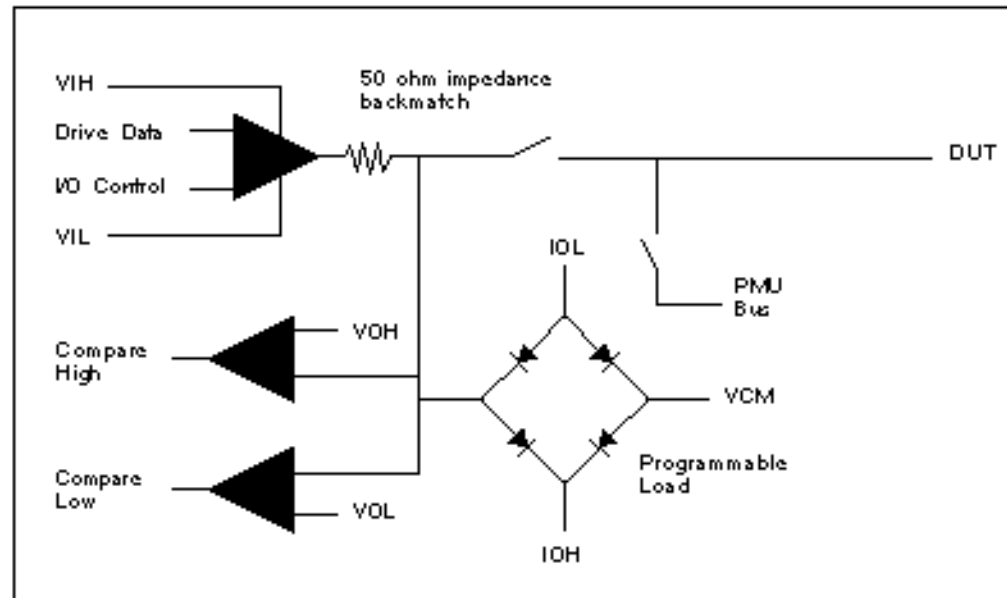


Figure 1-4. Pin Electronics Block Diagram

ATE Hardware

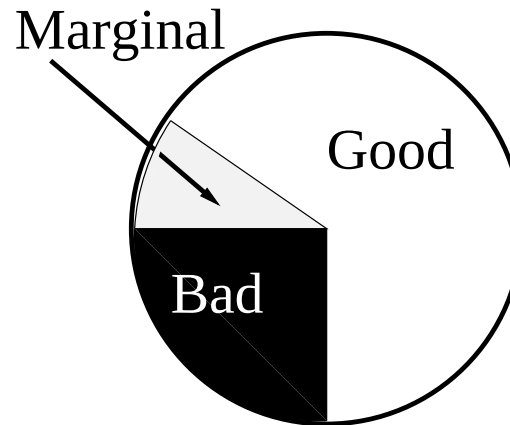
- ❑ Analog to Digital Converter (ADC) - Provides precision conversion (16 bits)
- ❑ Completes Measurements for PMU
- ❑ Measurements for Calibration
- ❑ Device Power Supply (DPS)
 - Force voltage or current
 - Current measured as one from several ranges
 - DPS connects via pogo pins
 - DPS connect to ADC via backplane for calibration
 - DPS connects to ADC via analog bus for A to D
- ❑ Clock provides reference oscillator (system clock)
- ❑ Clock (SC) provides a free-running clock and reference power supplies (for loadboard peripherals)

ATE Hardware

- ❑ Precision (parametric) Measurement Unit (PMU)
 - Force voltage and current
 - Measure voltage and current
 - PMU connected to PMU bus relay connection to DUT pins
 - PMU connects to pogo pins
 - PMU connects to ADC for calibration/analog bus for A to D
- ❑ Test Flows
- ❑ Defect Based Test
- ❑ Functional Test
- ❑ Scan Test
- ❑ Voltage Tests
- ❑ I_{ddQ} Test

Test Flows

- ICs are Good (65%), Bad (25%), and Marginal (10%).



- The smallest segment, ie the marginals, are the focus of most of the effort
- Remove good ICs to “be sure” marginals are left on the factory floor (yield loss)
- Leave with good ICs; source of reliability fails (early failures)
- Test techniques; functional test (at-speed or other), voltage stress (high and low), I_{ddQ} , scan (full/partial/AC)

Test Flows

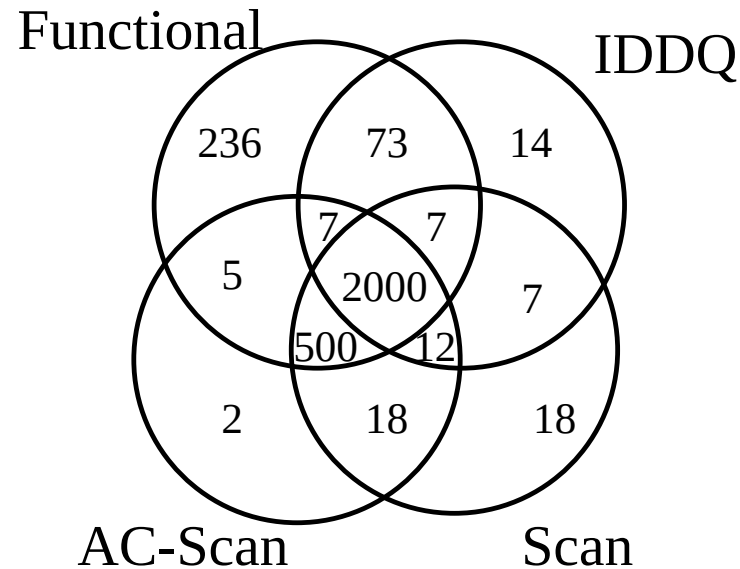
- Repeated studies still show each technique detects unique defects
- Burn-in remains an essential test practice
- Tests with a Boolean output, for example, functional or delay testing contrast with continuous test signals, for example, I_{ddQ} .
- Setting threshold limits for continuous signals is art not science
- Area of interest is reduce or eliminate functional test
- Test effectiveness has two components
 - The number and type of defects detected by a specific test type
 - The rate and latency of test type
- Many studies summarize detection with Venn Diagrams

Test Flows

- Each cell represents one or combination of tests
- The result of unique cell detects all faults is easily visualized
- Combinations of tests are quickly evaluated

Test Flows

- Sample from 2000 ITC Proceedings by Maxwell et. al.



- Test rate or latency shifts focus to test time
 - Stop-on-first-fail means DUT testing is stopped on a fail before entire test suite is completed
 - Order of testing may change effectiveness of a specific test
 - Stop-on-first-fail shifts burden to early detection to reduce average DUT test time

Test Flows

Time consumption for Test Sequence (18K units)†

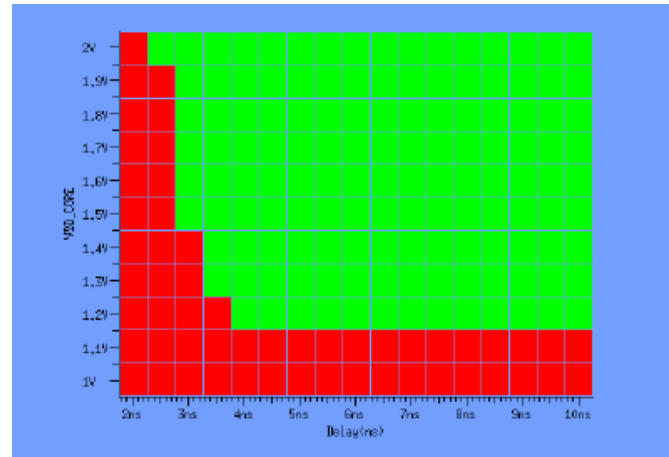
Test1	Test2	Test3	Test4	Time(s)
Func	I_{ddQ}	Delay	Stuck	62.6K
I_{ddQ}	Func	Delay	Stuck	62.9K
Func	I_{ddQ}	Stuck	Delay	63.1K
I_{ddQ}	Func	Stuck	Delay	63.5K
...				
Delay	Stuck	I_{ddQ}	Func	68.3K

†From Butler et. al., Proc. ITC 2000

- Other issues for test latency
 - Scan Rate lower than system clock rates
 - Scan fail is embedded in scan chain (random number shift clocks before fail detected on Test Output)
 - I_{ddQ} test time longer for current to “settle”

Test Flows

- Shmoo Test
 - The hunt for electrical bugs begins with the Shmoo plot
 - Delay strobe measurement (x-axis)
 - Supply voltage VDD (y-axis)



Defect levels

- General relation

$$DL = \frac{\text{fraction faulty chips passing test}}{\text{chips passing test}}$$

- Defect levels are a customer measure (*ppm* part per million)

- Part fails acceptance test
- Part fails system test
- Part fails maintenance test

- For a specific fault coverage T (fraction of faults detected by test)

- $T = 0 \rightarrow$ tests detect no possible faults in the circuit
- $T = 1 \rightarrow$ tests detect all possible faults in the circuit

$$DL(T) = \frac{Y(T) - Y(1)}{Y(T)}$$

Defect levels

$$DL = \left\{ 1 - Y^{1-T} \right\}$$

[aka Williams and Brown defect yield model]

- The true process yield $Y(1)$ also denoted by just Y
 - $Y(1)=0 \rightarrow$ all circuits faulty
 - $Y(1) = 1 \rightarrow$ all circuit fault free
- Simple example
 - Suppose $Y=0.5$ (50%) and no testing $T=0$ then 50% of all parts are faulty

$$DL = \left\{ 1 - 0.5^{1-0} \right\} = 0.5$$

- Suppose T is increased to 0.8

Defect levels

$$DL = \left\{ 1 - 0.5^{1-0.8} \right\} = 1 - 0.87 = .13$$

reducing the number of faulty devices by 75%

- A considerable improvement but $DL=0.13$ is not good enough
 - $DL = 13\% \rightarrow DL = 130,000 \text{ DPM}$
 - Typical DPM goal < 1000
- Increasing Y reduces the requirement of testing, permitting a reduction in test requirement (ie smaller T)
- Increasing T reduces the process yield requirements, permitting a reduction in yield (ie smaller Y)
- Other models

$$DL = \frac{(1 - T)(1 - Y) \exp \{-(n_0 - 1)T\}}{Y + (1 - T)(1 - Y) \exp \{-(n_0 - 1)T\}}$$

where n_0 is the average number of faults on a faulty chip

Defect levels

- With n_0 defect level model explicitly uses yield, test detection improvements and defect clustering
- Higher defect clustering translates to lower fault coverage requirements
- Simple example
 - Suppose a yield of 0.8 (80%), defect level is (DL=0.5%)
 $n_0 = 2 \rightarrow T=0.95$
 $n_0 = 10 \rightarrow T=0.38$
- Central question of reliability “Will all bad parts tested good fail?”
 - Short one word answer is no
 - Failure depends on the location of the missed defect

$$\text{Critical Area Ratio} = \frac{A_r}{A_c}$$

where A_r is the reliability area and A_c is the yield area

Defect levels

$$Reliability = Y^{A_r/A_c}$$

- More generally the device age t is required

$$R(t) = Y^{c(t)}$$

where $c(t)$ is a function of time, Critical Area Ratio, clustering etc.

Defect Based Test

- Traditional approach to test uses functional test with some (less emphasis on) scan tests

Comparison Traditional and Structured Tests†

Faults Detected	Traditional Method	Structural Method
stuck-at delay I_{ddQ} memory	functional/scan at-speed functional sampled functional functional/BIST	scan/BIST AC scan/BIST scan BIST

†From Maxwell et. al., Proc. ITC 2000

- Structural tests leverage Design for Test (DfT) DUT circuitry to reduce ATE requirements

Defect Based Test

□ Functional Test

- System-on-Chip (SOC) and Intellectual Property cores (IP cores) reduce system fault coverage from functional test
- Functional vectors demand largest share of ATE resources
- Functional vectors are derived from design vectors
 - Converting from simulation to ATE expensive; timing chief among many conflicting vector requirements vs. ATE resources
- Functional vector debug and result interpretation beyond Pass/Fail slower than other forms of testing

Defect Based Test

- High-Voltage Stress
 - Restricted (selected) set of functional vectors
 - Break-down weak (thin) oxides
 - Further testing with I_{ddQ} or Functional
 - Voltage sets are higher than burn-in
 - Requires high pattern coverage to verify all parts of chip
 - Analog and memory functions are most difficult to stress with HV

Defect Based Test

- Low-Voltage Test
 - Restricted (selected) set of functional vectors
 - Temperature activated (and also device aging) not affected by voltage stress
 - Bridging Faults (eg resistive vias) may not fail a nominal functional test
 - Requires high pattern coverage to verify all parts of chip
 - At lowered operating voltage recast as a stuck-at fault or delay
 - Delay fault difficult to diagnose because low-voltage implies reduced clock

Defect Based Test

□ I_{ddQ} Test

- I_{ddQ} has a long tradition and remains a defect oriented test
- Smaller set of test vectors compared to functional test
- PLL, on-chip clocks limit application of I_{ddQ}
- Screens for defects that increase DC current levels
- Slow test because transient effects have to decay out of signal
- Diminishing with increased spread of transistor (circuit, system) characteristics
- Measure before and after voltage tests to show changes in current

Defect Based Test

□ Scan Structural Tests

- Reduces ATE timing and pin-out requirements and increases DUT pin memory requirements
- Level-Sensitive Scan Design (LSSD) converts single clock latches to two-phase clock flip-flops
- One, typically more, scan chains course through DUT
- Other test points inserted by additional node multiplexing
- Generate scan vectors with ATPG (Automatic Test Program Generation)
- Scan with single-step clocking increases controllability and observability of DUT
- AC Scan (descendant of LSSD) test the transitions of the DUT
 - Apply two vectors; vector V1 is a setup and vector V2 is a stimulus

Defect Based Test

- Permits delay path testing
- Two sources of vector pair V1/V2
 - Single shift of V1 (V2 limited to single cycle shift codes of V1)
 - V1 is computed from V2; V1 applied and two clocks execute to yield V2 as test output
 - V1 and V2 muxed as separate master latches in scan flip-flop
- Typical Scan Single Stuck-At Fault (SSF) coverage, 90 – 95%