Standardized Access to Embedded Test and Debug Instruments
IEEE P1687 – The Next Step

Michael Laisne, mlaisne@qualcomm.com
Bill Eklow, beklow@cisco.com
Jeff Rearick, jeff_rearick@agilent.com,
Ken Posse, kepos@comcast.net
Al Crouch, al.crouch@innovys.com
Jason Doege, jdoege@da-test.com
Ben Bennetts, ben@dft.co.uk
Mike Ricchetti, mricchetti@ati.com
75 Different Instruments and counting under consideration!

Boundary Scan

MBIST

Clock Generator

Scan Configuration

Hub

1149.1 TAP

Alternate IDCode Register

Alternate IDCode Register

Core Instrument Interface

Configure Scan Wrappers

Configure Input PIN Isolators

Selection of Final vs Incremental Sig

Selection of Final vs Incremental Sig

Signal Tapping

Control Resetting

Selection of Final vs Incremental Sig

Configuration of Internal Memory BISTs

Configuration of Compactor Units

Configuration of External Memory BISTs

Configuration of Functional Units

Performance Monitors

O-Scope

IDBIST

IOWRAP

Current Control

Configuration of Internal Memory BISTs

Configure Scan Wrappers

Configure Clock Domain

Current Control

Configuration of Internal Memory BISTs

Selection of Output Comparator
**IJTAG (IEEE P1687) – Instruments under consideration**

<table>
<thead>
<tr>
<th>Instrument Interface</th>
<th>Configuration of Memory BISR</th>
<th>Selection of Final vs Incremental Sig</th>
</tr>
</thead>
<tbody>
<tr>
<td>Core Instrument Interface</td>
<td>Clock Tree Gating</td>
<td>Selection of Decompression Units</td>
</tr>
<tr>
<td>Alternate IDCode Register</td>
<td>Configuration of STUMPS Channels</td>
<td>Selection of Logic BIST Units</td>
</tr>
<tr>
<td>Analog Mux Network</td>
<td>Configure Clock Domains</td>
<td>Selection of Scan Mode</td>
</tr>
<tr>
<td>Assertion Checking</td>
<td>Configure Input Pin Isolators</td>
<td>Selection of Memory BIST Units</td>
</tr>
<tr>
<td>BERT</td>
<td>Configure Pullup/Pulldown Resistors</td>
<td>Selection of Output Compactor Units</td>
</tr>
<tr>
<td>Clock Divider and Chop Control</td>
<td>Configuring Scan Wrappers</td>
<td>Signal Tapping</td>
</tr>
<tr>
<td>Clock Mode Control for Delay Test</td>
<td>Control of ReSeeding</td>
<td>Single Step</td>
</tr>
<tr>
<td>Clock Tree Gating</td>
<td>Core Instrument Interface</td>
<td>State Dump</td>
</tr>
<tr>
<td>Configuration of Compactor Units</td>
<td>IOWRAP</td>
<td>Supplying Match Values</td>
</tr>
<tr>
<td>Configuration of Decomp Units</td>
<td>Power Control</td>
<td>Temperature Measurement</td>
</tr>
<tr>
<td>Configuration of external Memory BISTs</td>
<td>PLL and clock configuration</td>
<td>Test Executive</td>
</tr>
<tr>
<td>Configuration of Functional Units</td>
<td>PRBS</td>
<td>Test Scheduling of Memory BISTs</td>
</tr>
<tr>
<td>Configuration of internal Memory BISTs</td>
<td>Scanning in of Compare Signature</td>
<td>Voltage Control</td>
</tr>
<tr>
<td>Configuration of Memory BISR</td>
<td>Security</td>
<td>Voltage Measurement</td>
</tr>
<tr>
<td>Configuration of MISRs</td>
<td>Selecting Trace Buffers/Watch Buffers</td>
<td>Voltage Meter</td>
</tr>
<tr>
<td>Configuration of Polynomials</td>
<td>Enabling Comparators</td>
<td>Waveform Analysis</td>
</tr>
<tr>
<td>Configuration of STUMPS Channels</td>
<td>Enabling DMA</td>
<td>Waveform Generator</td>
</tr>
<tr>
<td>Configure Clock Domains</td>
<td>Enabling Scan Mode Configurations</td>
<td>Waveshape Generator</td>
</tr>
<tr>
<td>Configure Input Pin Isolators</td>
<td>Enabling Scan Safety Logic</td>
<td>Waveform Analyzer</td>
</tr>
<tr>
<td>Configure Pullup/Pulldown Resistors</td>
<td>Enabling Scan Wrappers (TAMs)</td>
<td>Waveform Generator</td>
</tr>
<tr>
<td>Configuring Scan Wrappers</td>
<td>Enabling Single-Step Operation</td>
<td>Waveform Generator</td>
</tr>
<tr>
<td>Control of ReSeeding</td>
<td>Enabling/Configuring Data Collection</td>
<td>Waveform Generator</td>
</tr>
<tr>
<td>Core Instrument Interface</td>
<td>Enabling/Configuring X-Masking</td>
<td>Waveform Generator</td>
</tr>
<tr>
<td></td>
<td>Generic</td>
<td>Waveform Generator</td>
</tr>
<tr>
<td></td>
<td>IBIIST</td>
<td>Waveform Generator</td>
</tr>
<tr>
<td></td>
<td>Inter-domain Synchronizers</td>
<td>Waveform Generator</td>
</tr>
<tr>
<td></td>
<td>IO Parametric Adjustment</td>
<td>Waveform Generator</td>
</tr>
<tr>
<td></td>
<td>IOBIST</td>
<td>Waveform Generator</td>
</tr>
<tr>
<td></td>
<td>IOWRAP</td>
<td>Waveform Generator</td>
</tr>
</tbody>
</table>

**April 2006**
75 Different Instruments and counting under consideration!
These instruments are needed for testing, debugging, yield improvement

- IP companies do not want to be
  - Devoting huge resources to generic test structures
  - Writing programs to enable these test structures
  - Having to redesign the wheel
  - Debugging test structure that can be bought off the shelf
IJTAG (IEEE P1687)

- Common, standardized internal test structures
  - Cost less to implement
  - Are less risky on 1st Silicon
  - Enable debugging tests
  - Result in better quality products
  - Enable faster yield learning
    - Result in even better quality products
  - Result in more defect-free systems
    - Create happier customers
IEEE P1687 is the draft standard for access and control of instrumentation embedded within a semiconductor device. It provides a standardized process that allows test systems to communicate with embedded instruments.
IEEE 1149.1 (JTAG) defines BSDL. BSDL describes boundary scan in a chip.
IEEE P1687 (IJTAG) defines a language that describes internal test and debug structures.
TAP controller must have
  1687 host (1687-ifier)
Each instrument must have
  1687 client
Each instrument may contain
  Instrument
  Pipe
  1687 host
  Any or all
Each instrument may communicate one level up (interrupt) or one level down if it has both a 1687 client and 1687 host
IJTAG (IEEE P1687)

Asynchronous I/F (sending instructions and configurations, receiving results and acknowledgment)

Synchronous I/F (sending clock, instructions and receiving results)
Scan path location of registers is contained in a file based 1687 description language
IJTAG (IEEE P1687)

Too Slow!!
Functional diagnosis – run functionals
IJTAG (IEEE P1687)

Functional diagnosis – capture results
Functional diagnosis – scan out results

IJTAG (IEEE P1687)
Functional diagnosis

- Set watch variables
- Set breakpoints
- Set cycle step mode

1. Functional simulation or pre-existing database

2. Send functional cycles to tester through standardized API (EDA and Tester)

3. Send chain flush command

4. Receive test results from tester API

5. Vary conditions through EDA API
• Using this standardized approach foundries can purchase off the shelf tools for memory yield improvement.
• Similar structures can be created for ATPG yield improvement, for functional vector debug, and for functional to test mode timing correlation.
90 nm Memory

- FA result on 90 nm single bit speed failure

6-T SRAM Schematic

90 nm failure SEM Image

Bright spot showed electrical path from poly contact to vdd or “1”
IJTAG (IEEE P1687) - Considerations

- Determine requirements – what makes sense and what is doable
- Requires adoption of standardized TAP features and architecture that supports IJTAG (specialized instructions)
- Creation of standardized mechanism for high bandwidth data
  - External Instrumentation and synchronization
- Language based Instrument description
- Establishment of Language based Protocols for use
- Relationship to other standards
Questions?