The Interconnect Verification Challenge

François Cerisier and Mike Bartley
Test and Verification Solutions
IP-SOC 2012
Grenoble, 5 Dec 2012
The Interconnect Verification Challenge

- What’s an interconnect
- Interconnect’s characteristics
- Topologies
- Transaction’s Paths
- Verification Goals
- Verification Environment
- Protocol conversion
- Scoreboard features
- Scoreboard architecture
SoC Interconnect Characteristics

- **Master/Slave communications**
  - Protocols (AXI, AHB, APB, OCP, PLB, OPB, DCR, Wishbone, company corporate bus, ...)
  - Bus Widths (16/32/64/128)

- **Memory Maps**
  - Shared memory map for all masters ?
  - Memory map clusters ?
  - One Memory map per master ?

- **Address Space**
  - Physical Address Space
  - System Virtual Memory Address Space (System MMU)
  - Virtual Address Space

- **SoC specific features**
  - Error management → invalid requests
  - Security → invalid request depending on security attributes
  - Power management
    - Invalid requests
    - Wake-Up
Interconnect Topologies

- **Shared Bus**
  - Chip select, arbiter

- **Cross Bar**

- **Muxes/Switch/Routers**

- **Network on a Chip**
Interconnect Routes

Slaves Memory Space

- **Slave 5**
  - AXI
  - Address: 0x7000_3FFF
  - 0x7000_2000

- **Slave 4**
  - APB
  - Address: 0x7000_1FFF
  - 0x7000_0000

- **Slave 3**
  - AHB
  - Address: 0x5FFF_FFFF
  - 0x5000_0000

- **Slave 2**
  - OCP
  - Address: 0x401F_FFFF
  - 0x4000_0000

- **Slave 1**
  - OCP
  - Address: 0x2FFF_FFFF
  - 0x2000_0000

- **Slave 0**
  - AXI
  - Address: 0x1FFF_FFFF
  - 0x1000_0000
Verification Goals

- **Address Map**
  - Are all masters able to access all possible slaves?
  - … under virtual address mode?
  - Errors on invalid addresses

- **Protocol Sanity**
  - Are all kinds of transactions supported on each route?
  - Are bursts/locks supported on each route?
  - Protocol not broken under stress conditions

- **SoC features**
  - Security
    - Can secure transactions access to all slaves?
    - Are unsecured transactions getting errors from secured slaves?
  - Power Management Use cases
    - Are we getting error from power off slaves?
    - Are we able to wake up a slave?

- **Use cases**
- **Performance Analysis**
- (Interconnect integration)
Interconnect Verification Environment
Interconnect Verification Environment

Virtual Sequences

SoC Interconnect

Scoreboard
Choosing the right sequence

FIFO

// fast master
Keep req_delay in [1..10];
// average is 5 cycles

// slow slave
Keep resp_delay in [1..20];
// average is 10 cycles

Probability to fill the FIFO is very high
Probability to empty the FIFO is very low

// slow master
Keep req_delay in [1..20];
// average is 10 cycles

// fast slave
Keep resp_delay in [1..10];
// average is 5 cycles

Probability to fill the FIFO is very low
Probability to empty the FIFO is very high
Choosing the right sequence

• Dynamic constraints
• Scenarios vary over time

 ➔ Make Interconnect reaching further traffic congestions
**Protocol conversion issues**

**AXI transfer**

Request transfer

AXI burst len=3  
Size = WORD  
Address = 0x3  
Kind = WRAPPED  
LOAD

Response transfer

<table>
<thead>
<tr>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>F</td>
<td>E</td>
<td>D</td>
<td>C</td>
<td>B</td>
<td>A</td>
<td>9</td>
<td>8</td>
</tr>
<tr>
<td>17</td>
<td>16</td>
<td>15</td>
<td>14</td>
<td>13</td>
<td>12</td>
<td>11</td>
<td>10</td>
</tr>
</tbody>
</table>

**Converted transfers**

Request transfer

LD16  
Addr = 0

Request transfer

LD8  
Addr = 0x10

Response transfer

<table>
<thead>
<tr>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>F</td>
<td>E</td>
<td>D</td>
<td>C</td>
<td>B</td>
<td>A</td>
<td>9</td>
<td>8</td>
</tr>
<tr>
<td>17</td>
<td>16</td>
<td>15</td>
<td>14</td>
<td>13</td>
<td>12</td>
<td>11</td>
<td>10</td>
</tr>
</tbody>
</table>

Response transfer

| 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 |
Scoreboard Requirements

• Connect to any bus protocol VIP
• End to End transaction checking
  – Data, direction, attributes, response, atomicity

• Support for:
  – Multiple address maps, Virtual address space
  – Address map reconfiguration, MMU
  – Security, Power management
  – User defined security/filtering (DRM, …)

• Comparison policies
  – Strict:
    • one to one transaction comparison
  – Permissive:
    • Allow transaction address realignment, dummy reads, nops
  – Per checker configuration
    • User switch on/off each checker (per path)
User’s experience

• **3 derivatives of a SoC Interconnect**
  – 40 masters, 60 slaves with over 200 paths
  – 5 protocols, 3 different bus sizes
  – Security Management
  – Power Management features
  – Dynamic address translations

• **Scoreboard Developments**
  – Right architecture choice is key
  – Generic features / Generic Adapters
  – Search and comparison algorithms

• **Verification results**
  – Address map specification
  – Wrong protocol translations of AXI FIXED from 64 to 32 bit buses
  – Deadlock in some traffic congestions involving bursts
  – Deadlock in power management
Conclusion

• SoC Interconnect needs to be verified from end to end

• Verification Environment should address
  – Complex scenarios
  – Stress/congestion conditions

• Interconnect SoC scoreboard should be generic & highly configurable

• Scoreboard can also provide:
  – Functional coverage metrics
  – Performance information