BIST for FPGA Cores in SoCs

Master's Thesis Defense Presentation
Prepared by: Jonathan Harris
Presentation Outline

- Background Overview
  - Atmel FPGAs
  - Macro Generation Language (MGL)
  - Previous work in BIST for FPGAs
- BIST for Atmel FPGAs
  - Logic
  - Routing
- Summary & Conclusions
- Demo
Overview of Atmel FPGA

- **Array sizes**
  - AT40K Series
    - 16x16, 24x24, 32x32, & 48x48
  - AT94K Series
    - 16x16, 24x24, & 48x48

- **PLBs arranged in 4x4 arrays**
  - bounded by repeaters to buffer long or heavily loaded signals

- **1 RAM per 4x4 array**
  - Not considered in this thesis

FPGA Array
Programmable Logic Blocks

- Up to 4-input functions
  - 4 PLB inputs (W, X, Y, Z)
    - X & Y = local direct or global bus
    - W & Z = global bus
- Two 3-input LUTs
  - 4-input LUT function possible by combining LUTs
- 3 PLB outputs X, Y, L
  - Optional tri-state on L
**PIPs in FPGA Routing Resources**

- **Basic Structure**
  - Transmission Gate with configuration memory bit

- **Cross-point PIP**
  - Connects vertical and horizontal wire segments

- **Break-point PIP**
  - Connects two wire segments in same plane
**PIPs in FPGA Routing Resources**

- **Multiplexer PIP (MUX PIP)**
  - Choose one input to drive output
  - Decoded
    - \(n\) bits for \(2^n\) inputs
  - Non-decoded
    - \(n\) bits for \(n\) inputs

- **Switch-box PIP**
  - (compound Cross-point PIP)
    - Connections in various directions between wire segments
PLB Input Multiplexers

- Input MUX PIPs select from vertical & horizontal busses
- W and Z MUX PIPs feed directly into PLB
- X and Y MUX PIPs feed into local routing MUX PIPs
PLB Local Routing

- Local routing to adjacent PLBs
  - 4 orthogonal directs (Y)
    - North, South, East, & West
  - 4 diagonal directs (X)
    - Northwest, Northeast, Southwest, & Southeast

- Input MUX PIPs
  - Select from direct connections or from global routing resources
Global Routing Associated With the PLB

- Horizontal and Vertical Busses
  - 5 sets with 3 busses each
- Global Routing Cross-point PIPs
  - Connect vertical/horizontal x8 lines
- Local Routing Cross-Point PIPs
  - Make connections from PLB to x4 lines
  - Connect vertical/horizontal x4 lines
- Two notations for x8 lines
  - Abus x8 lines
  - Ebus x8 lines
Global Routing and Repeater Cells

- Global routing - Repeaters
  - Staggered repeaters
  - Alternate: x4 line and Abus line
  - x4 line and Ebus line

- MUX PIP repeaters
  - Located every 4 PLBs

- Connections made in both directions through repeaters
Macro Generation Language

- MGL included in Figaro IDS software
  - available free from Atmel’s website
- Aspects of programming and HDL languages
  - variables, constants, arrays
  - if-else statements
  - for and while loops
  - case statements
- Dynamic Macros
  - All combinations of:
    - Up to 4-input LUTs
    - Register, Feedback, Tri-state output

**MGL code example:**

```mgl
def 
for i in 0 to ARRAY_SIZE loop 
instance "ORAcell" + i of ORAcell is 
  location(2,i); 
  connections(
    "CLK" -> "CLK",
    "RST" -> "RST",
    "BL" -> "BL\{1\}_\{i\},
    "BR" -> "BR\{3\}_\{i\},
    "P/F" -> "P/F");
  placeports("P/F" -> "Y");
end instance;
end loop;
```
Three Basic Dynamic Macros

- **FGEN1**
  - Implements up to a four variable logic function
  - Registered or Combinational Feedback
  - Optional Tri-state on output
  - Outputs available directly from LUTs or from FF

- **FGEN2**
  - Implements up to a three variable logic function
  - Registered or Combinational Feedback
  - Outputs available directly from LUTs or from FF

- **MGEN**
  - Multiplier based macro with upstream AND gate
  - Implement up to four variable logic function with two inputs ANDed
  - Outputs available directly from LUTs or from FF
Previous Work In BIST for FPGAs (Logic BIST)

- Comparison-based Logic BIST
  - ORCA 2C & 2CA
  - Xilinx 4000 & Spartan
- Configure PLBs as TPGs, BUTs, and ORAs
  - Run 1\textsuperscript{st} test session
  - Flip architecture, run 2\textsuperscript{nd} test session

\[=\text{TPG}\
\[=\text{BUT}\
\[=\text{ORA}\

Test Session 1

Test Session 2
Previous Work In BIST for FPGAs (ORA Structure for Logic BIST)

- ORA compares 4 corresponding BUT outputs
- Feedback latches any mismatches
  - FF holds Pass/Fail indication
  - Configuration memory read back (Xilinx)
    - Read Pass/Fail results in ORA flip-flops
  - Shift results out via shift register (ORCA)
Previous Work In BIST for FPGAs (Routing BIST)

- **Routing BIST**
  - **Comparison-based**
    - Configure PLBs as TPGs and ORAs – select wire segments and PIPs as WUTs
      - ORCA 2C & 2CA
      - Xilinx 4000 & Spartan
      - Xilinx Virtex

- **BIST results**
  - Configuration memory readback (Xilinx)
  - Shift register (ORCA)
Previous Work In BIST for FPGAs (Routing BIST)

- Routing BIST
  - Parity-based
    - Configure PLBs as TPGs and ORAs – select wire segments and PIPs as WUTs
      - Xilinx 4000 (switch boxes only)
    - Parity over separate routing
Architectural Impacts on Logic BIST for the Atmel FPGA

- Banks of 4 PLBs
  - Set/Reset Polarity
  - Clock Edge Triggering
- Individual PLB Programmable Set or Reset
Architectural Impacts on Logic BIST for the Atmel FPGA

- PLB X and Y Inputs
- Directly connect to adjacent PLBs
  - Orthogonally (Y)
  - Diagonally (X)
- Impose constraints on logic BIST architecture
Logic BIST for the Atmel FPGA

Routing Scheme 1

Routing Scheme 2

Test Session 1

Test Session 2
ORA Structure for Atmel Logic BIST

- ORA compares ‘X’ and ‘Y’
- Comparison of only 2 inputs w/ FB to latch mismatch
- Shift signal needed
Reconfigured ORA Structure

- Partial Reconfiguration used to create shift register
- Routed shift signal used to retrieve results
  - Also used for additional testing

Reconfigured ORA Connections

Reconfigured ORA Structure
ORA Alternatives for Logic BIST

- **2-PLB ORA**
  - 1 PLB = comparison
  - 1 PLB = shift function
- Uses global routing
- Limits Diagnostic Resolution
Generation of Logic BIST Configurations

- **5 MGL BUT Configurations**
  - Chosen from dynamic macros
  - Generated through MGL program

- **3 BUT Configurations – Theoretical Best Case**
  - Chosen assuming control over all PLB configuration bits
  - Assuming observability of all PLB outputs

- **4 Manually Produced BUT Configurations**
  - Generated through combination of MGL and C programs
    - MGL program used to generate template bitstreams
    - C program used to perform bit manipulation
MGL – 5 BUT Configurations

**BUT1 – FGEN1R**

- Inputs: X, W, Y, Z
- Outputs: X, W, Y, Z
- Logic gates: AND, OR, NOT
- Clock (clk), Reset

**BUT2 – FGEN1**

- Inputs: X, W, Y, Z
- Outputs: X, W, Y, Z
- Logic gates: AND, OR, NOT
- Clock (clk), Set

**BUT3 – FGEN1RF**

- Inputs: X, W, Y, Z
- Outputs: X, W, Y, Z
- Logic gates: AND, OR, NOT
- Clock (clk), Reset, Set
MGL – 5 BUT Configurations

BUT4 – MGEN

BUT5 – FGEN2F
Logic BIST Fault Coverage – 5 BUT Configurations

- Individual FC (Left Edge PLBs)
- Individual FC (Middle PLBs)
- Individual FC (Right Edge PLBs)
- Cumulative FC (Middle PLBs)
- Cumulative FC (Left Edge PLBs)
- Cumulative FC (Right Edge PLBs)

BUT Configurations:
1. 10
2. 20
3. 30
4. 40
5. 50

Fault Coverage:
- BUT Configuration 1: 60
- BUT Configuration 2: 50
- BUT Configuration 3: 40
- BUT Configuration 4: 30
- BUT Configuration 5: 20
Theoretical Limit – 3 BUT Configurations

BUT1 – FGEN1R

BUT2

BUT3
Logic BIST Fault Coverage – 3 BUT Configurations

- Cumulative Fault Coverage
- Individual Fault Coverage

Fault Coverage (%)

BUT Configuration

1

2

3

Fault Coverage (%)
Manually Generated – 4 BUT Configurations

BUT1 – FGEN1R

BUT2

BUT3

BUT4
Logic BIST Fault Coverage – 4 BUT Configurations

- Individual FC (Lft. Edge Cells)
- Individual FC (Mid. Cells)
- Individual FC (Rt. Edge Cells)
- Cumulative FC (Lft. Edge Cells)
- Cumulative FC (Mid. Cells)
- Cumulative FC (Rt. Edge Cells)
Optimum Choice of Logic BIST Configurations

- 4 Manually Produced BUT Configurations
  - Advantages of 4 Configurations
    - Minimum number of configurations achievable
    - Best case realizable fault coverage (99.7%)
  - Disadvantages of 4 Configurations
    - Loss of fault coverage of PLBs along edge of array
      - Can be overcome by rotation of logic BIST architecture
Logic BIST Architecture Rotation

West

TPG 1

TPG 2

North

TPG 1

TPG 2

East

TPG 1

TPG 2

South
Implications of Logic BIST Rotation

- Most local interconnect tested
- Improved diagnostic resolution
- All except 8 PLBs have 99.7% FC
  - 8 PLBs left with < 90% FC

- Good FC (99.7% - PLB)
- (95.81% - PLB + Intentt.)
- XYXY FC (82.23% - PLB)
- (87.75% - PLB + Intentt.)
- YXYX FC (87.95% - PLB)
- (84.4% - PLB + Intentt.)
Faults Left Undetected

- Local routing Cross-point PIPs
  - Most detected in routing BIST
- L output and tri-state buffer
  - Detected in routing BIST
- Some X and Y direct connections
  - Most detected in routing BIST
Comparison to Previous Work

- Smaller PLB facilitates fewer configurations
  - Less number of modes of operation in Atmel vs. previous
- Local interconnect testing simultaneously for Atmel

<table>
<thead>
<tr>
<th>FPGA</th>
<th>Number of BIST Configurations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORCA 2C</td>
<td>9</td>
</tr>
<tr>
<td>ORCA 2CA</td>
<td>14</td>
</tr>
<tr>
<td>Xilinx 4000</td>
<td>12</td>
</tr>
<tr>
<td>Xilinx Spartan</td>
<td>12</td>
</tr>
<tr>
<td>Atmel AT40K</td>
<td>4</td>
</tr>
<tr>
<td>Atmel AT94K</td>
<td>4</td>
</tr>
</tbody>
</table>
Routing BIST for the Atmel FPGA

3 Basic Fault Models

- Faults in PIPs
  - stuck-on, stuck-off
- Wire segment faults
  - stuck-at 0, stuck-at 1, open wire
- Faults between wire segments (bridging faults)
  - wires shorted together
Routing BIST for the Atmel FPGA

- Modified version of parity approach proposed by Sun, et al.
- 2-bit binary counter with parity generation
  - up counter with even parity
  - down counter with odd parity
- Parity used as test pattern
- Three wire segments observed simultaneously
Routing BIST for the Atmel FPGA

- Opposite logic values between:
  - Different bits of individual test patterns
  - Corresponding bits of the two test patterns
- Useful for testing:
  - Bridging faults between wire segments
  - Stuck-at faults and opens in wire segments
  - Stuck-at and stuck-on/stuck-off faults in PIPs

<table>
<thead>
<tr>
<th></th>
<th>Up-count with Even Parity (C1, C0, Parity)</th>
<th>Down-count with Odd Parity (C1, C0, Parity)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>000</td>
<td>111</td>
</tr>
<tr>
<td></td>
<td>011</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td>101</td>
<td>010</td>
</tr>
<tr>
<td></td>
<td>110</td>
<td>001</td>
</tr>
</tbody>
</table>
Routing BIST for the Atmel FPGA

- Routing BIST configurations generated with combination of MGL and C programs
- ORA can not be implemented in MGL
  - C program required for bit manipulation
- Configurations target:
  - Cross-point PIPs
  - Vertical and Horizontal Repeaters
  - L output and tri-state buffer
  - X direct connections
- Wire segments tested with cross-point PIPs and repeaters
  - Wire segments stuck-at 0, stuck-at 1, open, and some bridging faults
- Self Test AReas (STARs)
  - Subset of routing resources configured as WUTs
Cross-Point Routing BIST Phases

- 8x8 STARs
- Global Routing
  - Cross-point PIPs
    - stuck-off faults
    - stuck-on faults
- Wire segments
  - opens
  - bridging faults
- Local Routing
  - Cross-Point PIPs
    - stuck-on faults
**Cross-Point PIP Abus STARs**

- Checkerboard Pattern created in array
  - allows opposite logic values from adjacent STARS
  - tile very evenly into any array

![Checkerboard Patterns](image)

- a) 16x16
- b) 24x24
- c) 32x32
- d) 48x48
Cross-Point PIP Ebus STARS

- Checkerboard Pattern created in array
  - allows opposite logic values from adjacent STARS
  - does NOT tile evenly into array
  - 8x8 STARs, 8x4 STARs, 4x8 STARs, & 4x4 STARs

a) 16x16
b) 24x24
c) 32x32
d) 48x48
Repeater Routing BIST Phases

- Repeaters comprised of four MUX PIPs
- Connections made in both directions
- Same TPGs and ORAs as in Cross-point PIP tests
- 3 Sets of Repeater Configurations
  - Connections between x8 and x4 lines on same side (loop backs)
  - Connections between x8 and x4 lines on different sides (diagonals)
  - Connections between two x8 lines and between two x4 lines (straight-through)
Repeater Routing BIST Set 1
Configuration

- Alternating STARs
  - up-count w/ even parity & down-count w/ odd parity
- Loop-back connections made in repeaters
- TPGs and ORAs swapped to test opposite directions
- Rotate CCW 90° to obtain vertical tests
  - 1x8 STARs (horizontal tests) or 8x1 STARs (vertical tests)
Repeater Routing BIST Set 2
Configuration

- Alternating overlapping STARs
  - up-count w/ even parity & down-count w/ odd parity
- Diagonal connections made in repeaters
- TPGs and ORAs swapped to test opposite directions
- Rotate CCW 90° to obtain vertical tests
  - 1x16 STARs (horizontal tests) or 16x1 STARs (vertical tests)
Repeater Routing BIST Set 3
Configuration

- No alternating STARs
  - 4xarray (horizontal tests) or arrayx4 (vertical tests)

- Straight-through connections in repeaters
  - no need for opposite logic values

- TPGs and ORAs swapped to test opposite directions

- Rotate CCW 90° to obtain vertical tests
  - 4xarray STARs (horizontal tests) or arrayx4 STARs (vertical tests)

Up-count w/ Even Parity
**Abus Repeater Set 1 Configuration**

- Applies regularly to Abus lines
  - Repeater boundaries line up with array boundaries
  - 1x8 STARs match with array sizes
- Repeaters at array boundaries
  - Not fully tested, only inputs from one side of repeater

**Diagram:**
- **Up-count w/ Even Parity**
- **Down-count w/ Odd Parity**
Ebus Repeater Set 1 Configuration

- **Ebus repeater boundaries offset**
  - STARs line up in middle of array
  - Problems at edges
    - Developed scheme to test edge Ebus repeaters
    - Reduced diagnostic resolution – 1 of 2 repeaters
Abus Repeater Set 2 Configuration

- Diagonal connections have issues with boundaries
- $1 \times 8$ overlapping STARs to create diagonal connections
  - STARs do not exactly line up with array boundaries
  - No diagonal connections tested at array edges
- Loop around connections made at the edges

Up-count w/ Even Parity  Down-count w/ Odd Parity
EBus Repeater Set 2 Configuration

- STARs are 2x16 due to boundary mismatches
  - Different STAR architecture for different array sizes
  - STARs fit exactly for 16x16 array, overlapped for 32x32 and 48x48 arrays
  - All Ebus repeaters tested for desired faults

16x16, 32x32, and 48x48 arrays
Ebus Repeater Set 2 Configuration

- STARs for 24x24 – 2xarray w/ different architecture
  - 24x24 is odd multiple of 8, unlike other array sizes
  - Requires different STAR architecture to test repeaters
- All repeaters tested for desired faults
Abus & Ebus Repeater Set 3
Configuration

- Configuration similar for Abus and Ebus
  - Straight-through connections in MUX PIPs tested
  - All Ebus repeaters tested
  - All Abus repeaters but at array boundaries tested
  - Test only for stuck-off faults

Up-count w/ Even Parity
Detected Repeater MUX PIP Faults

- All stuck-on, stuck-off faults detected
- Wire segments stuck-at 0, stuck-at 1, and open
- Most bridging faults
  - Between x8 lines
  - Between x4 lines
  - Between x8 lines and x4 lines

<table>
<thead>
<tr>
<th>MUX PIP</th>
<th>R8 Input</th>
<th>R4 Input</th>
<th>L8 Input</th>
<th>L4 Input</th>
</tr>
</thead>
<tbody>
<tr>
<td>R8</td>
<td>S-On</td>
<td>S-Off</td>
<td>S-On</td>
<td>S-Off</td>
</tr>
<tr>
<td></td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>L8</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>R4</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>L4</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

Legend:
- Green: S-Off
- Orange: S-On

[Diagram of MUX PIP faults with connections]
L Output Tri-State TPG Connections and Timing Analysis

C2 delay
C2
C1
C0

C0
C1
C2
C2d
To ORA
L Output Tri-State BIST Architecture

- Same architecture as logic BIST
  - Similar TPG – 3-bit binary counter with delayed 3rd bit
  - Comparison-based ORAs
  - Rotated to form East, West, North, and South sessions
  - Closely coordinated with X direct configurations
  - Ensures proper testing of local routing cross-point PIPs associated with L output
X-Direct Routing BIST Phases

- Targets unobserved X connections
  - Architectures flipped horizontally – 4 test sessions
  - Connections not used in logic BIST
- Also test some local routing cross-point PIPs
  - Closely coordinated with L output configurations
    - Ensures testing of most cross-point PIPs associated with L output
Comparison to Previous Work

- # Configurations between ORCA and Xilinx
- Differing routing architectures
  - Staggering of routing resources
    - Xilinx – busses
    - Atmel – repeaters
  - Rotational symmetry
    - ORCA, Atmel
- Most significant impact
  - Limited access to all routing resources

<table>
<thead>
<tr>
<th>FPGA</th>
<th>Number of BIST Configurations</th>
</tr>
</thead>
<tbody>
<tr>
<td>ORCA 2C</td>
<td>27,44</td>
</tr>
<tr>
<td>ORCA 2CA</td>
<td>27,44</td>
</tr>
<tr>
<td>Xilinx 4000</td>
<td>128</td>
</tr>
<tr>
<td>Xilinx Spartan</td>
<td>206</td>
</tr>
<tr>
<td>Atmel AT40K</td>
<td>48</td>
</tr>
<tr>
<td>Atmel AT94K</td>
<td>48</td>
</tr>
</tbody>
</table>
MGL’s Effect on BIST Development and Application

- Imposes PLB testing issues
  - MGL alone not enough to test PLBs
    - C program required for bitstream post-processing

- Causes ORA issues in routing BIST
  - Can not implement ORA through MGL alone
    - C program again required for bitstream post-processing

- Major problem with software versions!!!
  - Newer versions of CAD tools create problems
    - Statements in MGL not compiled properly in new software versions
Summary of BIST Configurations

- 4 logic BIST and 48 routing BIST Configurations
  - 4 Logic BIST
    - North, South, East, West Sessions
  - 48 Routing BIST
    - 16 Cross-point PIP configurations
    - 24 Repeater configurations
    - 4 L output and tri-state configurations
    - 4 X direct connection configurations

- Most resources tested during logic and routing BIST
  - some X and Y direct connections untested
    - at edges of array – direct connects to I/O cells
  - some L output cross-point PIPs
    - at edges of array
  - Abus lines 8 PLBs from the edges incompletely tested
  - Abus repeaters at edges incompletely tested
  - All can be tested in BIST for the I/O cells
Conclusions

- Small PLB has large impact on BIST
  - affects logic and routing BIST
    - BIST results retrieval
    - ORA implementations (logic and routing BIST)
    - logic BIST architecture (BUT-to-ORA connections)
    - routing BIST architecture (modified parity-based approach used)
- MGL has implications on BIST
  - can not use MGL alone
    - C programs required for post processing of bitstreams
      - used to set up proper test conditions in logic BIST
      - used to create ORAs for routing BIST
    - Software version problems!!
- Need better way to test the FPGA
  - AVR can be used to test FPGA core in AT94K SoC