Renesas’ Strategy to apply Tessent MemoryBIST Soft Programmable

Oct. 11, 2023
Tomonori Sasaki
DFT platform Technology Section
Digital Design Technology Department
Renesas Electronics Corporation
OUTLINE

Company
- Renesas Introduction

Background
- BACKGROUND
  - Tessent MemoryBIST Hard programmable
  - Tessent MemoryBIST Soft programmable Algorithms

Solution
- New function Soft Programmable Operation set

Evaluation&Results
- EVALUATION for Soft operation set function

Summary
- Conclusion
SOLUTION OFFERINGS

Our mission is to develop a safer, healthier, greener, and smarter world by providing intelligence to our four focus growth segments: Automotive, Industrial, Infrastructure, and IoT that are all vital to our daily lives, meaning our products and solutions are embedded everywhere.

Automotive
Highly reliable vehicle control, safe and secure autonomous driving, Eco-friendly electric vehicles

Industrial
Lean, flexible and smart industry

Infrastructure
Robust infrastructure, enabling safety and efficiency

IoT
Comfortable, safe and healthy lifestyles through IoT
BACKGROUND

◆ There are cases which new algorithm is required for quality unstable memory under new process and miniaturization.

◆ When new algorithm is required for a detected memory fail, it is not possible to add it as hard logic since the MBIST implementation has already been completed.

◆ Tessent MemoryBIST has soft programmable algorithm function. But it didn’t have a soft operation set function to control memory enable signals.

◆ In this time, Tessent MemoryBIST provides complete soft programmable function. So, we evaluated this to determine product applicability.
Tessent MemoryBIST Hard Programmable

- Tessent MemoryBIST has two methods as the implementation for MBIST algorithms.

  - Hard programmable
  - Soft programmable  (Next page.)

- Hard programmable

  - Main use: mass production test
  - Implement the algorithms as hard logic.

    - For Mass production pattern
    - For using memory fail analysis in past

```
Algorithm (MARCH) {
  TestRegisterSetup {
    OperationSetSelect : SyncRamOps ;
    …
  }
  MictoProgram {
    Instruction (Write0) { … }
    Instruction (Read0_Write1) {
      OperationSelect: ReadModifyWrite
      Instruction (Read1) { … }
    }
  }
}
```

```
OperationSet (SyncRamOps) {
  Operation (NoOperation) { … }
  Operation (Write) { … }
  Operation (Read) { … }
  Operation (ReadModifyWrite) { … }
}
```

<table>
<thead>
<tr>
<th>MBIST Controller</th>
</tr>
</thead>
<tbody>
<tr>
<td>Xaddress up</td>
</tr>
<tr>
<td>Xaddress down</td>
</tr>
<tr>
<td>Yaddress up</td>
</tr>
<tr>
<td>Yaddress down</td>
</tr>
<tr>
<td>Bank-addr up</td>
</tr>
<tr>
<td>Bank-addr down</td>
</tr>
<tr>
<td>Data_input</td>
</tr>
<tr>
<td>Expect_data</td>
</tr>
<tr>
<td>Chip_select</td>
</tr>
<tr>
<td>Write_enable</td>
</tr>
<tr>
<td>Read_enable</td>
</tr>
</tbody>
</table>
Tessent MemoryBIST Soft Programmable Algorithms

- Soft Programmable Algorithm (Existing func.)
  - Main use: Memory Fail analysis
  - Can add new algorithm after implementation
  - However, the algorithms with OperationSet implemented as Hard logic.
    ➞ Cannot add new enable control pattern which isn’t defined in OperationSet.
  - Must prepare instruction reg. for using soft algorithm
    - Specify with “soft_instruction_count” parameter
    - Renesas strategy: prepare it for three instructions

```
MBIST Controller

Soft Programable

Soft Algorithm (Existing func.)

Algorithm (MARCH) {
  TestRegisterSetup {
    OperationSetSelect : SyncRamOps ;
    ...
  }
  MicroProgram {
    Instruction ( Write0 ) { … }
    Instruction ( Read0_Write1 ) {
      OperationSelect : ReadModifyWrite
    }
    Instruction ( Read1 ) { … }
  }
}
```

Instruction Reg.

Instruction Reg.

Instruction Reg.

Micro code

Xaddress up

Xaddress down

Yaddress up

Yaddress down

Bank-addr up

Bank-addr down

Data_input

Expect_data

Chip_select

Write_enable

Read_enable

Instruction Reg.

Instruction Reg.

Instruction Reg.

Micro code

OperationSet ( SyncRamOps ) {
  Operation (NoOperation) { … }
  Operation (Write) { … }
  Operation (Read) { … }
  Operation (ReadModifyWrite) { … }
}
SOLUTION : New function Soft Programmable Operation set

Want to test memory / fail analysis greater flexibility in post-silicon.

- Soft Programmable Operation set (New func.)
  - Main use: Memory Fail analysis
  - Can add new algorithm including operation set after implementation
  - Must prepare operation reg. for using soft algorithm
  - Specify the following parameters
    - “soft_operation_count” parameter
    - “soft_operation_length” parameter
EVALUATION for Soft Programable operation set function

For greater flexible memory test / fail analysis in post-silicon.

New Soft Programable Operation set function need to add the registers for storing micro codes.

Therefore, it becomes a trade-off between quality (algorithm flexibility) and MBIST area.

◆ Evaluate the MBIST area impact for applying Soft Programable Operation set.
  ● Assume that the allowable area increase range is 10%
    (Although the actual conditions depend on the product specifications.)
EVALUATION & RESULT (1)

✓ Check an area increase rate when changing the number of “soft_operation_count”
  - “soft_operation_count” parameter : 3 → 6 → 9

→ By preparing more operations, the setting of enable signals can be made flexibly.
  - Target 1 controller of ASIC product : with 16 Mem-IF (including {Max 1024word x 16bit} )

<table>
<thead>
<tr>
<th>soft_operation_count</th>
<th>Area [0.63/gate]</th>
</tr>
</thead>
<tbody>
<tr>
<td>No use</td>
<td>79,292 [gate]</td>
</tr>
<tr>
<td>3 operation</td>
<td>88,540 [gate]</td>
</tr>
<tr>
<td>6 operation</td>
<td>93,356 [gate]</td>
</tr>
<tr>
<td>9 operation</td>
<td>93,895 [gate]</td>
</tr>
</tbody>
</table>

(soft_instruction_count = 13)
(soft_operation_length = 2)

The impact on MBIST area becomes heavy, over 10%, when setting 3 operations or more.
EVALUATION & RESULT (2)

✓ Check an area increase rate when changing the number of “soft_operation_length”
  • “soft_operation_length” parameter : 2 ➔ 3 ➔ 4 ➔ 5
  ➔ By preparing more length, Allows more operations for one address.
  • Target 1 controller of ASIC product : with 16 Mem-IF (including {Max 1024word x 16bit} )

<table>
<thead>
<tr>
<th>soft_operation_length</th>
<th>Area [0.63/gate]</th>
</tr>
</thead>
<tbody>
<tr>
<td>No use</td>
<td>69,532 [gate]</td>
</tr>
<tr>
<td>Operation length 2</td>
<td>72,103 [gate]</td>
</tr>
<tr>
<td>Operation length 3</td>
<td>78,171 [gate]</td>
</tr>
<tr>
<td>Operation length 4</td>
<td>79,917 [gate]</td>
</tr>
<tr>
<td>Operation length 5</td>
<td>82,511 [gate]</td>
</tr>
</tbody>
</table>

(soft_instruction_count = 3) (soft_operation_count = 3)

The impact on MBIST area becomes heavy, when setting 3 length or more per 1 operation.
CONCLUSION

◆ Evaluation result:

- Based on 10% area increase, can apply the soft operation set with the following setting.
  - The number of operation is 3 and the number of operation length is 2.

→ We can use this function as follows under the above setting.
  - Can test with more flexible enable control.
  - Complex algorithms that use many operations become trade-off with area increase.

→ If the product specifications allow for a MBIST area increase, we consider to implement soft programmable under the following purpose.
  - Aiming for 100% cause identification in failure analysis.

◆ Expectation for Siemens EDA
  - Reduce an area for soft programmable, especially for operation length.