International Journal on Recent Researches in Science, Engineering & Technology (IJRRSET)
A Journal Established in early 2000 as National journal and upgraded to International journal in 2013 and is in existence for the last 10 years. It is run by Retired Professors from NIT, Trichy. Journal Indexed in JIR, DIIF and SJIF.
Research Paper
Available online at: www.jrrset.com

ISSN (Print) : 2347-6729
ISSN (Online) : 2348-3105
Volume 8, Issue 2
February 2020.
JIR IF : 2.54
SJIF IF : 4.334
Cosmos: 5.395

Received on: 03.02.2020 Published on: 26.02.2020 Pages : 13 - 17

DESIGN AND EXCUTION OF CORE LEVEL DFT ARCHITECTURE WITH GATE LEVEL APPROACH

R. Sivakumar [a*], J. Jayarajan [b]

Department of Electrical and Electronics Engineering, Dr. M G R Educational and Research Institute, Chennai-95, Tamilnadu, India

[a] e-mail: sivarmtech2006@yahoo.com, [b] e-mail: jraja140@gmail.com

Abstract

This work explains the concepts of core level design-for-test architecture with gate level approach, its flow of implementation on a design of wireless core-A and debugging of errors that are encountered at the time of implementation. The flow takes gate level net list and the same library finally generates I/O Wrapper cells, post stitched NL, SPF files for ATPG. Different tools like DFTC from Synopsys and Tessent-TK from Mentor are used throughout the flow.

Keywords: DFT Architecture, Testing, EDT Logic, Register

1. Introduction

So far, the quality and efficiency of scan test has been inhibited by issues such as less-than-optimum test coverage, high vector volume, slow shift speed, and excessive IR drop. One main reason for these issues is that the entire chip is being tested as a whole. This method of testing may create excessive amount of toggles during scan shift that impacts stability of PG mesh. The sheer size of a chip will also hamper the ATPG tool with long run time and inefficient and/or ineffective patterns reduce confidence of coverage obtained at block-level due to chip-level design-for-test (DFT) - unfriendly environment in which the block is placed, etc. With Core level DFT, the chip is carved into multiple sub-blocks, called cores, and each core is tested individually. The advantage of such an approach can be summarized in two-folds

Quality:

- Address IR drop issue by reducing overall switching activity during test
- Increases test quality by ensuring each core is tested with guaranteed coverage at top-level
- Allows different cores to be tested under different test conditions

Productivity:

- Lowers test cost by reducing overall vector count and potentially faster shift speed
- Provides better post-silicon debug resolution
- Provides framework to adopt different DFTsolutions for targeted cores.

2. General Features

- Test modes available for a core run through Core level DFT:
- CORETEST: Using scan in and scan out bus to test the core itself
- Compressed : Scan chains are compressed
- Uncompressed: Scan chains are not compressed
- EXTEST: flops on the input and output boundaries are made accessible onto a unique set of scan in and scan outs. These chains are made available for further stitching at top-level to test the interconnecting wires among cores and top-level logic outside the cores
- Support DC Scan, AC Scan, memory BIST, IDDQ and IO pad Characterization
- Support vector compression
- JTAG based control
- Support interconnect test between cores (EXTEST)
- Scan flops inside a core are grouped as wrapper chains and internal chains
- Clock domains are not mixed Chains are balanced to boost vector efficiency
- Support scan out port feed through to facilitate routing at top-level
- Each core has one compressor/decompressor
- Multiple Test Clocks are supported

3. DFT Core Level Architecture

As depicted by Figure 1, this approach uses the functional core as the outer boundary for DFT, and inserts the TEST CONTROL REGISTER inside the functional core at gate-level. The implementation provides the RTL description for the TEST CONTROL REGISTER and it can be synthesized into a gate-level verilog netlist (using minimum compile effort). Then, it inserts the TEST CONTROL REGISTER verilog description into a gate-level representation of the functional core during DFT scan stitching. It automatically makes the TEST CONTROL REGISTER instantiation, creates the necessary DFT ports and link them to the corresponding pins of the TEST CONTROL REGISTER (done through Dc_shell commands like create_net, connect_pin). The biggest advantage of this method is that it eliminates the need for an extra DFT wrapper hierarchy while preserving the ability to create a fully DFT-compliant core

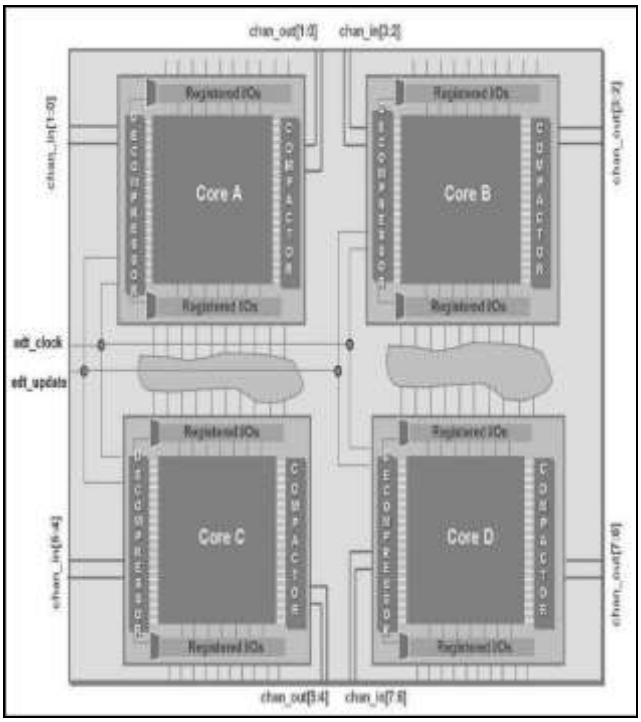

Figure 1 DFT Core Level Architecture

3.1 Test Control Register Details

The TEST CONTROL REGISTER provides all mode controls for the core that instantiates it. Its main content is a programmable JTAG register chain that controls scan-test access of the pertinent core. As well, it houses various gating logic based on the register bits. The Figure 2 shows JTAG interfacing with TEST CONTROL REGISTER.

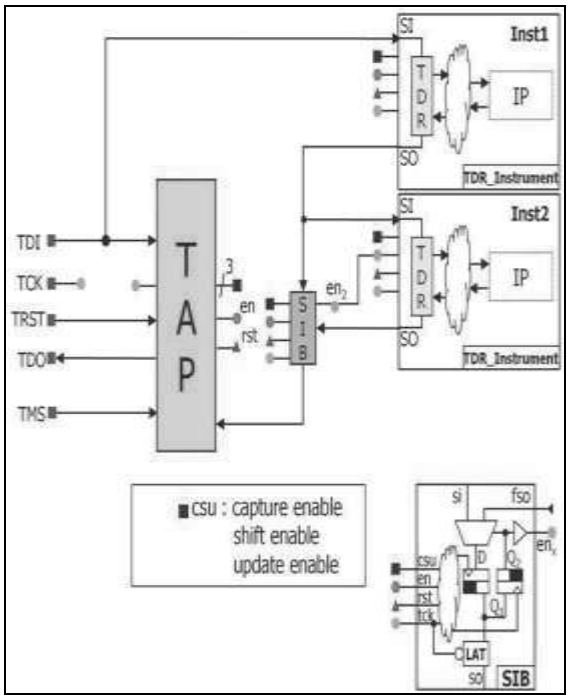

Figure 2 JTAG Interface with Test Control Register

3.2 EDT Logic

Embedded Deterministic Testing (EDT) is the technology used by Tessent Test Kompress and it is shown in Figure 3. EDT technology is based on traditional, deterministic ATPG and uses the same fault models to obtain similar test coverage using a familiar flow. EDT extends ATPG with improved compression of scan test data and a reduction in test time.

Embedded logic - EDT logic is generated and embedded in the IC to:

- Receive the compressed test patterns from the ATE and decompress them.
- Deliver the uncompressed test patterns to the core design for testing.
- Receive and compress the test results and return them to the ATE

4. Core Level DFT Implementation

This section provides a high-level flow/process description of how a given core marches through DFT implementation. The final outcome of this flow is delivery of post-DFT net list, with acceptable DFT quality, and supporting files into Physical Design (PD). Figure 4 shows the flow. It is divided into sections, each of which will be described below.

4.1 Step1

The main scan implementation is broken down into 2 DFTC/Design Compiler runs. It is implemented by a set of automation scripts. The steps are as follows:

1. The Design Compiler (DC) script will insert the TEST CONTROL REGISTER into the functional core and create the necessary ports and connections at the core level.

2. Then, all functional I/Os of the core will be checked for isolation. The flow will insert an I/O wrapper cell on any I/O that is not properly registered.
3. The final step in this section is to perform a trial stitch to look at how best to hook chains for balancing. This will dump out a scan ordering instruction for the Wrapper interface chains to be used in the next step.

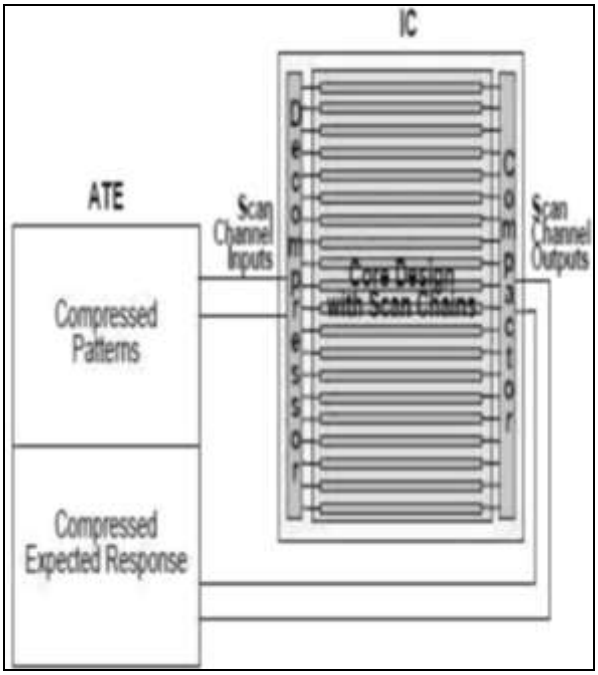

Figure 3 Tester Connected to a Design with EDT

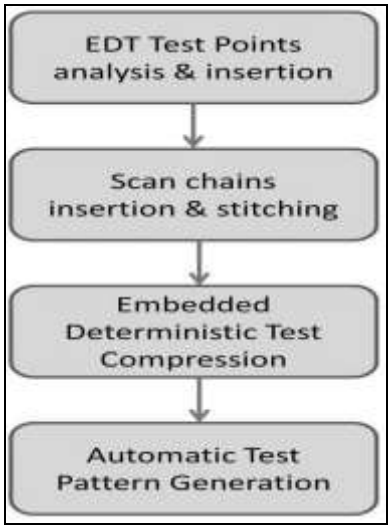

Figure 4 DFT Core-Level Implementation Flow

4.2 Step 2

It takes the net list and Wrapper interface chain ordering information from step 1 to perform the actual scan stitching. As mentioned, stitching is done in 3 modes simultaneously, Compressed, Uncompressed and EXTEST. The result of this pass is a post-stitched net list. As well, some intermediate report files are dumped out. These are for the generation of final CTL and SCANDEF files. The scan chains are stitching with clock and edge mixing turned off and terminal lockup latch disabled.

5. Errors Debugged

Errors like D1,D2,D3,D9,T3 which are violating scan cell data rules and scan chain trace rules were encountered and were debugged using DFT Visualiser by setting appropriate test clocks, proper definition of reset signals, adding specific design constants.

6. Results

I/O Wrapper cells were inserted with trial stitch during step1. Scan stitching is done and EDT logic is inserted and post-stitched net list is obtained with generation of EDT and libcomp files in step2. Post DFT NL with scan def files and CTL models were generated for further uses in the PD, SOC level testing.

7. Conclusion

With the above implementation test quality increases by ensuring each core is tested with guaranteed coverage at top-level, IR drop issue is also addressed by reducing overall switching activity during test. Compressed test vectors reduce ATE memory and channel requirements and reduced data volume results in shorter test application times and higher tester throughput than with traditional ATPG.

8. References

[1]. Y. Juan, H. Huang and S. Lai, A distortion cancelation technique with the recursive DFT method for successive approximation analog-to-digital converters, IEEE Transactions on Circuits and Systems II, 2015, 63(2), 146-150.
[2]. S. C. Lai ,Y.S. Lee and S. F. Lei, Low-power and optimized VLSI implementation of compact recursive discrete fourier transform (RDFT) processor for the computations of DFT and inverse modified cosine transform (IMDCT) in a digital radio mondiale (DRM) and DRM+ receiver, Journal of Low Power Electronics and Applications,2013, 3, 99-113.
[3]. Mandal and R. Mishra, Reconfigurable design of pipelined CORDIC processor for digital sine-cosine, Journal of Signal Processing Theory and Applications, 2012, 1, 27-35.
[4]. Lai, S.C., Juang, W.H., Lin C., Luo, C.H. and Lei, S.F., High-throughput, power efficient, coefficient-free and reconfigurable green design for recursive DFT in a portable DRM receiver, Int. J. Electr. Eng., 2011, 5(6), 137–145.
[5]. Shaik, U. Yedukondalu, A lower power, low complexity and memory free implementation of novel recursive DFT and IDFT algorithms, IJERA, 2013, 3(4), 662-666.
[6]. Erik Jan Marinissen et al., A DFT architecture for 3D-SICs based on a standardizable die wrapper, Journal of Electronic Testing: Theory and Applications, 2012, 28(1), 73-92.
[7]. Anshuman Chandra and Krishnendu Chakrabarty, Low-power scan testing and test data compressionfor system-on-a-chip, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2002, 21(5), 597-604.
[8]. S. Dey, E. J. Marinissen and Y. Zorian, Test access methodology for system-on-chip testing, Testing System Chips: Methodologies and Experiences, 1999, 5(8), 36–48.
[9]. F. M. Bufler, Y. Asahi, H. Yoshimura, C. Zechner, A. Schenk and W. Fichtner, Monte Carlo simulation and measurement of nano scale n-MOSFETs, IEEE Trans. Electron Devices, 2003, 50(2), 418–424.
[10]. J. C. Hamann, J. W. Pierre, S. F. Legowski, and F. M. Long, Using monte carlo simulation to introduce tolerance design to undergraduates, IEEE Trans. Educ., 1999, 42(1), 1–7.
[11]. S. Basu, I. Sengupta, D. R. Chowdhury and S. Bhowmik, An integrated approach to testing embedded core and interconnects using test access mechanism (TAM) switch, J. Electron. Test.: Theory Appl. (JETTA), Special Issue on Plug-and-Play Test Automation for System-on-a-Chip, 2002, 18(4/5), 475–485.