# Low Power testing by don't care bit filling technique

Chetan Sharma
(M.Tech-VLSI Department, JSS Academy of Technical Education, Noida, India)

**Abstract:** Test power is major issue of recent scenario of VLSI testing. There are many test pattern generation techniques for testing of combinational circuits with different tradeoffs. The don't care bit filling method can be used for effective test data compression as well as reduction in scan power. This paper gives a new advancement in automatic test pattern generation method by feeling don't care bit of the test vector to optimize the switching activities. Finally this concept produces low power testing.

**Keywords:** ATPG test vector generation, Huffman code, Parity bit generation, Switching activity.

**Introduction:** Chip designing is mainly divided into two steps firstly is frontend part secondly backend industry. Firstly RTL is design by keeping into account the testing aspects for minimizing technical effort in test vector generation and producing low cost testing. RTL is designed by the specification prepared by architecture team according to customer requirement. Then verification of functionality is done by previously made test bench.
In the part of backend, synthesis is done for producing schematic and layout of the desired chip. Here all power and timing analysis is done. Optimization is also occurred in this step. Now it is send to fabrication team.
**Test power:** It is very essential to minimize the test power and test cost of the test procedure of any previously designed RTL. If test power or test cost is more than a normal value then RTL of that chip is useless. For the generation test vector anyone can use manual method of test pattern generation in which CAD tool can be used. A specified method is used for self generation of test vectors for testing the CUT (circuit under test).There are various method like as D Roth's algorithm technique, Boolean difference method.

These technique may also have don't care bit "x" at the test vector. This don't care should replaced by particular defined bit. We can choose either 0 or 1.Normally it is seen that design engineer choose randomly this don't care bit. But it should choose such that there will be minimum switching activity in the test vector because on reducing switching activity, desirable power reduction will produce. Thus we can divide the power dissipation during scan testing into: (1) scan-in power - due to transitions in scan test vectors. (2) scan-out power - due to transitions in the output response being scanned out.

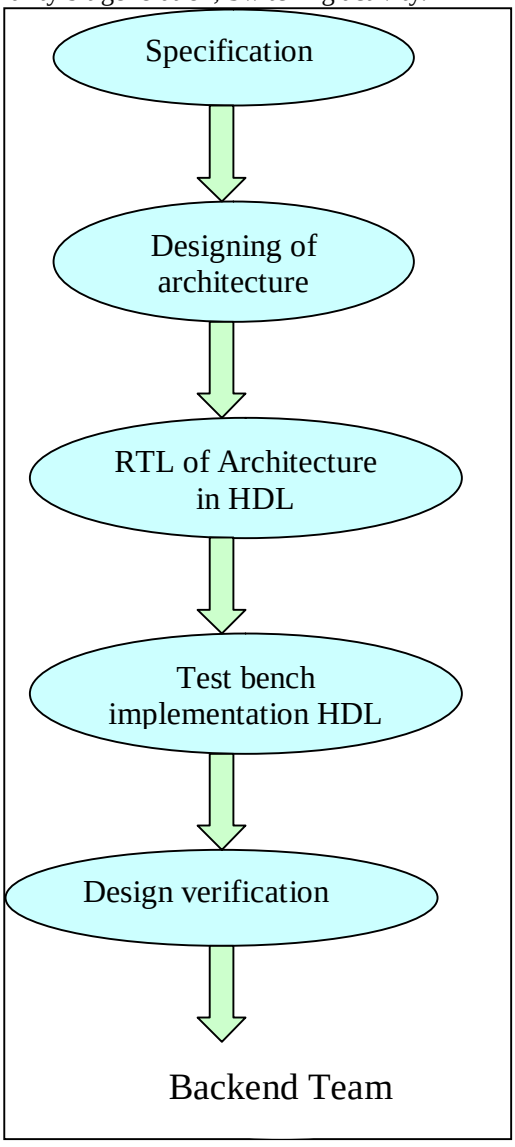


Fig (a) steps in Frontend VLSI Design

In this paper, a widely used weighed transitions metric (WTM) introduced, is used to estimate the average and peak power consumption. Test data $T=\{T_1,T_2,.....,T_m\}$ has m patterns and the length

of the pattern is n bits .Each test pattern $T_i=\{t_{i1},t_{i2},...t_{in}\}$,$1\leq i\leq m$, $1\leq j\leq n$ denotes the $j^{th}$ bit in $i^{th}$ pattern. weighed transitions metric $WTM_j$ for $T_j$ the average test power $P_{avg}$ and peak power $P_{peak}$ are estimated as per the formula [4]

$$WTMj = \sum_{i=1}^{n-1} (n-i) * (tj.i \oplus tj.i+1)$$

$$Pavg = \frac{\sum_{j-1}^{m} WTMj}{m}$$

$$Ppeak = \max_{1\leq j\leq m} WTMj$$

**Methods of Filling Don't Care Bits**

Automatic test pattern generated test data contains a huge amount of don't care bits. Such don't care bits in test data can be manipulated to enhance the test data compression and to reduce test power. For the statistical codes, test data is divided into equal size blocks of B bits. To improve the test data compression, the no. of distinct blocks in a given test set should be reduced and frequency of occurrence for each distinct block should be increased. In this technique of filling don't care bits which have less computational complexity compared to other proposed algorithms.

**(a) Minimum Transition bit filling technique:**

Consider a test vector matrix that has 0, 1 and X entries, where each row of the matrix corresponds to a test vector for the circuit. X is an unspecified value and can be filled with either 0 or 1. The conventional approach for filling the X's in the test cube is to do random fill (R-fill) in which the X's are randomly replaced by 0's or 1's. In Rfill, the idea is that it increases the chance that a single test cube would detect additional faults and hopefully the other test cubes would not be required and can be eliminated during reverse fault-simulation. However, since we are considering power, which involves the number of weighted transitions in the test vector, it is best to consider Minimum Transition Fill (MT-fill).

In MT-fill, a series of X entries in the test vector are filled with the same value as the first non-X entry on the right side of this series. This minimizes the number of transitions in the test vector when it is scanned in.

For example, consider the test vector: 100**XX**010**X**1**X**0. This vector, after MT-fill, would become 100**00**010**11**0**0**. If the test vector has a string of X bits that is not terminated by a non-X bit on the right side, then it should be filled by the bit value to the left of the sequence. For example: 1000001011**XX** should be 100000101111 after MT-fill.

Let us understand this technique by following example

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| X | 0 | 1 | 1 | 1 | X | X | X | X | 0 | 1 | 0 | 0 | 1 | X | X |
| 1 | 1 | 0 | 1 | 1 | 1 | X | X | 0 | 0 | 0 | 0 | 1 | 0 | X | X |
| 1 | 1 | 0 | 1 | X | 1 | X | X | 1 | 1 | 0 | X | 1 | 1 | X | X |
| 0 | 1 | 0 | 1 | 1 | 0 | X | X | X | 0 | 1 | X | 1 | 0 | X | X |

Fig (b): Test Set for Algorithm Demonstration

| **Test vectors (size=8)** | **Applying MT-fill** | **WTM** |
|---|---|---|
| T1=X0111XXX | 00111111 | 6 |
| T2=X01001XX | 00100111 | 14 |
| T3=110111XX | 11011111 | 11 |
| T4=000010XX | 00001000 | 7 |
| T5=1101X1XX | 11011111 | 11 |
| T6=110X11XX | 11011111 | 10 |
| T7=010110XX | 01011000 | 21 |
| T8=X01X10XX | 00111000 | 9 |

Fig (c): Applying MT-fill algorithm and WTM

For Block size =4 and 3-encoded distinct block using selective Huffman ,optimal Huffman and modified selective Huffman technique gives compression ratios are 12.5%, 12.5% and 34.37%.

**(b) Bit filling in ATPG method:**

For example test vector generated by automatic test pattern generation technique is " 1X10X " shown in fig(c). It has two X bits i.e. second bit and fifth bit.

Now on the place of X state, defined bit 0 or 1 is used according to previous and next bit of this don't care bit. As shown in example second don't care is replaced by bit 1 because there is no switching among first three bits of test vector. If we replace it by 0 then it'll increase 2 switching activities. Finally affect the power. So by this method second bit is replaces as defined bit 1.

In the case of fifth bit generated by ATPG method. We'll check only fourth bit because bit vector length is 5.As fourth bit is 0 so this don't care should replaced by 0 for no switching activity involved in fourth and fifth bit position. So new modified test vector is " 11100 " This modified test vector is shown in fig (d).

| Bit no. | Assigned Bit | Power reduced or not |
|---|---|---|
| First bit of test vector | 1 | NA |
| Second bit of test vector | **1** | Reduced |
| Third bit of test vector | 1 | NA |
| Fourth bit of test vector | 0 | NA |
| Fifth bit of test vector | **0** | Reduced |

Fig (d): Modified test vectors for switching activity reduction

**(c) Hamming distance based technique:**

The Hamming distance of block B1 with highest frequency of occurrence will be calculated from the B2 with the second highest frequency. The Hamming distance is 1 if the bits on the same position of two blocks are opposite, i.e. '1' and '0'. The Hamming distance between two blocks is summation of such bits with opposite values.

The Hamming distance between 10X1 and 010X is 2 as its first and second bits have opposite values. If the Hamming distance between B1 and B2 is more than 0, the Hamming distance with next block with descending order of frequency will be calculated. Two blocks for which the Hamming distance is 0, will be merged and a new block M1 will come into existence. The next block in the sequence will be than compared with merged block M1. This process is repeated until further merging is not possible. The process is repeated with the next highest frequently occurring still unmerged block. The merging has increased the number of specified bits. Still there is a chance that few bits are unspecified. Such bits are replaced with zeroes.

**EXPERIMENTAL RESULTS**

In this paper describe don't care bit filling algorithm and calculate Weighted Transition Matrix, peak power, average power and compression ratio using different Huffman coding methods , implementation MT-fill technique and Hamming distance based technique algorithm and Selective Huffman ,Optimal Selective Huffman and Modified Selective -Huffman code were MATLAB7.0 language. The experiments are conducted on a workstation with a 3.0 GHz Pentium IV processor and 1GB of memory.

**Conclusion:** For any chip, Power is calculated by $P = \alpha CV^2f$. It means the switching activity is directly proportional to power dissipation. In this paper power dissipation of testing process is decreased by minimizing switching activities of test vector. This test vector is generated by Automatic test pattern generation method. It is done by replacing don't care bit to a defined bit as discussed in this paper.

**References:**


[1] P.Girard Survey of Low –Power Testing of VLSI Circuits: proceeding IEEE Design & Test -2002 pp.82-92

[2] N.Nicola and B.M.Al-Hashimi Power – Costrained Testing of VLSI Circuits: proceeding in Kluwer Academic Publishers-2003

[3] P.Girard, C. Landrault, S. Pravossoudovitch and D.Severac Reducing Power Consumption During Test Application by Test Vector Ordering: proceeding in ISCAS-1998 pp.296-299

[4] R.Sankaralingam, R. Oruganti and N. Touba Static Compaction Techniques to Control Scan vector Power Dissipation :Proceeding in IEEE VLSI Test Symposium-2000,pp. 35-42

[5] N.A.Tauba Survey of Test Vector Compression Techniques :proceeding IEEE transcaction Design & Test of Computers-2006

[6] Mehta U, Dasgupta K, Devashrayee N Modified Selective Huffman Coding for Optimization of Test Data Compression,Test Application Time and Area Overhead :Proceeding in Journal of Electronic Testing Theory and Applications-2010,vol.26

[7] K.A.Bhavsar Mehta, Analysis of Test Data Compression Techniques Emphazing Statistical Coding Schemes: proceeding in ACM Digital Library USA,2011