Testing and diagnosis of power switches in SOCs

Testing and Diagnosis of Power Switches in SOCs

Sandeep Kumar Goel Maurice Meijer Jose Pineda de Gyvez

Philips Research Laboratories
IC Design – Digital Design & Test
High Tech Campus 5, M/S WAY 41
5656 AE Eindhoven, The Netherlands
{sandeepkumar.goel, maurice.meijer, jose.pineda.de.gyvez}@philips.com

Abstract

The use of power switches in modern system chips (SOCs) is inevitable as they allow for efficient on-chip static power management. Leakage is today one of the main hurdles in low-power applications. Power switches enable power gating functionality, i.e., one or more parts of the SOC can be powered-off during standby mode leading in this way to savings in the overall SOC's power consumption. In this paper, we present a circuit and a method to test power switches. The proposed method allows testing of on/off functionality. In case of segmented power switches individual failing segments can be identified as well by using the proposed test strategy. The method requires only a small number of test patterns that are easy to generate. Furthermore, the proposed method is very scalable with the number of power switches and has a very small area-overhead.

1. Introduction

The ongoing miniaturization of the integrated circuit feature size has a significant impact on the chip's size, performance, and power consumption. With every technology node, the circuit performance is improved due to shorter transistor channel length, lower threshold voltage, and reduced gate-oxide thickness. This, however, will lead to increased leakage power due to increased sub-threshold leakage and gate oxide tunneling current. As feature sizes shrink below 100nm, leakage power becomes as important as dynamic switching power in many applications [1][2][3]. To minimize both dynamic switching power as well as leakage power dissipation, modern system chips (SOCs) require efficient power management. For that purpose, it is becoming common in practice to provide the whole SOC or some parts with power management modes. In general two modes can be distinguished: (1) active mode, and (2) standby mode. In active mode, the SOC (or part of it) is able to perform the function for which it has been designed, while in standby mode, SOC (or part of it) is idle.

To minimize standby power consumption, power supply gating is used. To enable power gating functionality, different parts of a SOC are equipped with one or more power switches [4][5][6]. For example, in a core-based SOC [7][8], every individual core can be equipped with one or more power switches. Based on the activity in the SOC and the data transactions between different cores, a core can be individually turned-off through the power switch. In this way, the leakage power is minimized for the core, which is turned-off, thus, leading to savings in the overall power consumption.

Figure 1(a) shows a conceptual example of a core-based SOC with four cores, which are connected to the power supply (VDD) via power switches. To minimize the risk of not working and ease of manufacturing, instead of a large power switch, segmented power switches [5] are fabricated in practice. A segment can contain one or more transistors. Therefore, each core in a SOC is connected to a number of small power switches. Figure 1(b) shows an example implementation of such a power switch.

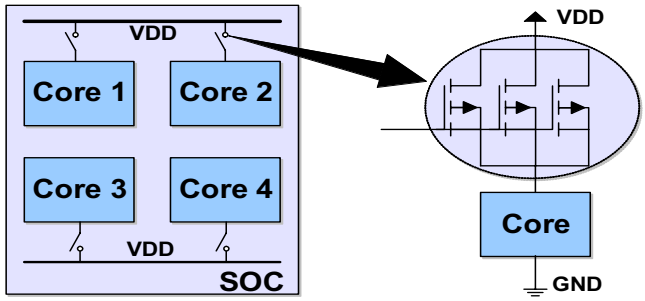

Figure 1: An example core-based SOC equipped with power switches.

Fault-free operation of power switches is very critical for the functional operation of the SOC. In case of faulty switches, the SOC can suffer from performance loss or may not be operational in the worst case. Therefore, testing of these switches for manufacturing defects is very important. In this paper, we present a simple yet effective Design-for-Test (DfT) circuitry together with a method to test power switches. The proposed method also enables identification of individual failing segments in case of segmented power switches. The area-overhead for the proposed DfT circuitry is very small.

The remainder of this paper is organized as follows. Section 2 reviews the prior work in this domain and motivates motivations for using power switches. Section 3 explains various types of power switches used in these SOCs. A formal definition of the problem addressed in this paper is described in Section 4. Section 5 presents our novel DfT approach for various types of power switches. Test pattern generation for the proposed approach is described in Section 5.4. Section 6 concludes this paper.

Proceedings of the Eleventh IEEE European Test Symposium (ETS'06)
0-7695-2566-0/06 $20.00 © 2006 IEEE

2. Prior Work

Multiple supply voltages together with power switches are increasingly used to minimize both dynamic and static power consumption. Lackey et al. proposed a system architecture called *voltage islands* in [9]. In a voltage islands architecture, the complete chip is divided among different islands, where every island contains one or more logic blocks and has a unique power characteristic. In other words, different islands use different on-chip voltage supplies. Furthermore, every island can be equipped with one or more power switches to enable power gating functionality. Similarly, [10][11] describe techniques to minimize the power consumption by using different supply and threshold voltages. Power Wise Interface (PWI) [12] open standard defines a serial interface between a SOC power supply controller and an external voltage regulation system. The external voltage regulation system allows dynamic adjustment of the on-chip supply voltage and the threshold voltages.

Most of these papers assume implicit testing of power switches when the chip is powered-on. However, this type of testing is not sufficient and one needs to find out whether all power switches are working correctly or not. To understand this, let us consider an example chip with one switch partitioned in n segments out of which, j are defective. Let's also assume that during power-on the chip still works. Observe, however, that the correct functional operation of the chip does not necessary mean that all n segments are functional. In fact, the j faulty segments will escape fail detection. If this is the case, the remaining n-j switches will draw more current and it is possible that with time, some of them will breakdown and cause the chip to malfunction. Moreover, the SOC's performance can be hampered during active mode operation because of the additional voltage drop across the switch due to the failing segments. Therefore, it is very important to identify individual failing power switches. To the best of our knowledge, there are no papers available in literature that describe the testing and diagnosis of power switches in SOCs. In this paper, we present a simple yet effective circuit and method to test and diagnose power switches.

3. Types of Power Switches

Several techniques for power switch realization are presented in [4][5][6]. These techniques describe the use of one or more PMOS or NMOS transistors as power switches. Based on the requirement, these transistors can be controlled individually or by a common input signal.

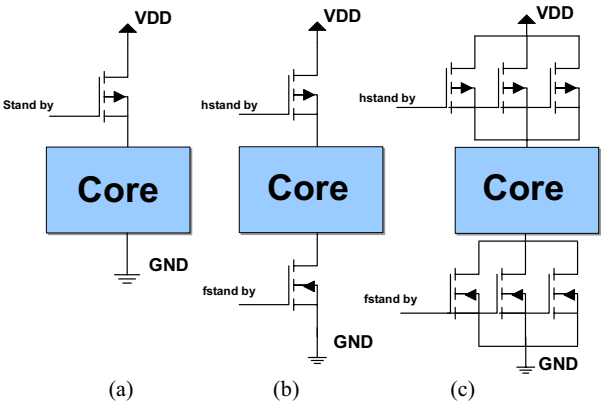

Figure 2: Examples of power switches.

Figure 2 shows the three types of power switch configurations that are widely used. Figure 2(a) shows an example of a power switch known as *Header switch*. A header switch is a PMOS transistor that is controlled by a dedicated control signal *standby*. When the *standby* signal is active, e.g. logic '1', the header switch does not conduct. In this situation, power supply to the core is gated and the IP does not operate functionally. When the *standby* signal is inactive, e.g. logic '0', the header switch conducts and the IP is able to operate. An alternate to the *Header switch* is the so-called *Footer switch*, which is an NMOS transistor and is used to control the ground line (GND). Figure 2(b) shows an example of a symmetric *Header and Footer switch.* In this case, there are two control signals *hstandby* and *fstandby*. However, the control signal *fstandby* is usually the inverse of the signal *hstandby*. The operation of this switch can be easily derived from the operation of the header switch.

Another popular way to implement power switches is via a number of transistor segments, where every segment can contain one or more transistors. All transistors in a segment share the drain, gate, source and bulk material. Again, they can be implemented as a header, or a header + footer switch. Figure 2(c) shows an example of a segmented symmetric header and footer switch. In this figure, there is only one segment and it contains three transistors. Here, the IP is able to operate if at-least a pre-determined number of segments are turned on simultaneously. One of the reasons behind such an implementation is to improve on Design-for-Manufacturability (DfM). Other reasons can be seen as the gradual on/off switching of the power switch to minimize the VDD/GND bounce [5], or the physical placement of the segmented power switches around the core.

4. Problem Definition

Power gating functionality is enabled by power switches using two distinct operation modes, e.g. (1) the switch is conducting, and (2) the switch is not conducting. From a test point-of-view, power gating functionality has to be verified to prevent failures due to manufacturing defects. For example, a short between the source and the drain of a header switch will cause the switch to be permanently on. Similarly, an open between a header switch and the core to connected to it, will cause the core to be permanently off. Therefore, it is very important to test the power gating functionality. Furthermore, if there are a number of segments in a power switch, then it is important to ensure that all segments are working correctly. This relates to the life-time of the switch. In this paper we present a test methodology and a Design-for-Test (DfT) circuit required to allow the testing of power switches. The problem addressed in the paper can be formally defined as follows.

Problem [Testing of Power Switches]
Instance: Given is a power switch with a number of segments m, and for each segment s, the number of transistors t_s and the segment on-resistance R_s are given. Furthermore are given the supply voltage *VDD* and a number k, which represents the minimum number of segments that needs to be turned on simultaneously such that the core connected to the power switch can operate functionally.
Objective: Determine a test circuitry and a pattern set P_m, such that all segments are tested for on/off functionality and each individual failing segment can be identified.

For a single header switch, $m = 1$, $t_s = 1$, and $k = 1$. For multi-segment power switches, the number of segments k that needs to be turned on simultaneously is determined at design time and

should be selected such that they are sufficient to provide a value of at least VDD_{min} at V_{core}, where V_{core} is the voltage at the core's power supply pin.

Selection of Number of Segments

Although in the problem definition as described above, we assume that k is given, it is important to know how the value of k is selected for a given power switch. The number of segments k is selected such that the on-resistance of the k segments does not exceed the resistance of the core, i.e.,

$$\frac{\sigma}{\sum_{s=1}^{k} \frac{1}{R_s}} \le R_{core} \quad \text{for } 1 \le k \le m$$

where R_s is the on-resistance of a segment s, R_{core} is the equivalent core resistance ($\sim 1/\alpha Cf$), and σ ($0 < \sigma \le 1$) is a constant. When this condition is met, V_{core} will be at least VDD_{min}. A value of VDD_{min} at V_{core}, will ensure that the core can still operate functionally. The value of σ depends on the value of VDD_{min} (for example $\sigma = 1$ for $VDD_{min} = VDD/2$). Consequently, the value of k can be determined by the following expression:

$$k \ge \frac{m \times \sigma}{R_{core} \times \sum_{s=1}^{m} \frac{1}{R_s}}$$

Please note that k should be an integer. Consider the example of a logic block with a switching circuit capacitance of 0.6nF, a maximum operating frequency of 200MHz, a nominal power supply (VDD) of 1.2V and VDD_{min} as 0.6V ($\sigma = 1$). The power switch is sized such that the voltage drop across the switch is less than 0.01VDD when the block is running at its maximum performance.

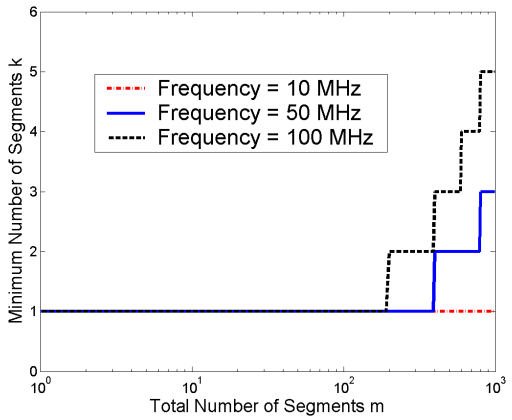

Figure 3: Selection of minimum number of segments.

Figure 3 shows k as a function of m and different test clock frequencies. From the figure, one can see that the value of k does not increase linearly with m. Depending on the operating frequency, the value of k varies between 1 and 5 even for a power switch with 1000 segments. Therefore, for a large set of practical power switches, the value of k should not exceed five.

5. Testing of Power Switches

We first start with a test solution for a single header power switch. The solution also applies to a single footer power switch. Later, the proposed circuitry is extended for the symmetric header plus footer switch and the segmented power switches.

5.1 Header Switch

For a core with a header power switch shown in Figure 4(a), on/off functionality of the power switch can be tested by means of the additional circuitry shown in Figure 4(b).

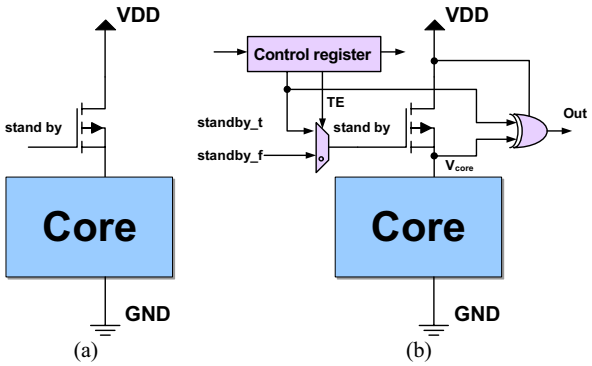

Figure 4: Test circuitry for a header switch.

The basic idea is to use a comparator to compare the logic level value of the core's power supply V_{core} with the logic value of the *standby* signal. The comparator can be implemented as an EX-OR gate as shown in Figure 4(b). The comparator and the other added circuitry are powered from the undisrupted power supply *VDD* to allow operation when the controlled power supply V_{core} is turned-off. The signal *standby_f* is the functional signal to control the operation of the power switch, while the signal *standby_t* is the test signal for the same operation. The signal *standby_t* can be provided by means of a shift register, which can be programmed and controlled via the IC level IEEE 11491.1 TAP controller.

To observe the value of the comparator output (*Out*), the output can be connected to a scan-able flip flop, which can be scanned out. This also makes the proposed circuitry compliant with existing Design-for-Test (DfT) approaches such as scan test for logic circuits.

5.1.1 Power Gating Functionality

To test the power switch with the proposed circuitry, the following two patterns are applied. In the test mode, the signal *TE* is set to 1 so that the multiplexer in front of the power switch selects the signal *standby_t*.

Pattern 1: *TE* = 1, and *standby_t* = '1'

In this case, the power switch should be turned-off and hence V_{core} should be much lower than *VDD*. Please note that the leakage should be constrained by the power switch. Ideally V_{core} should be zero. Therefore, the EX-OR output should be equal to '1', i.e., *Out* = '1' for a correct operating power switch. If *Out* is equal to logic '0' instead of '1', this indicates that the power switch is not working correctly and there may be a short between V_{core} and *VDD*.

Pattern 2: *TE* = 1, and *standby_t* = '0'

In this case, the power switch should be turned-on and hence V_{core} should be equal to *VDD*. Therefore, the EX-OR output should be equal to '1', i.e., *Out* = '1' for a correct operating power switch. If *Out* is equal to logic '0' instead of '1', this indicates that the power switch is not working correctly and that

there may either be an open in the power switch or there exists a short between V_{core} and *GND*.

Therefore by observing the value of the *Out* signal, one can check whether the switch is working correctly or not.

5.1.2 Ordering of Test Patterns

It is important to note here that the order in which the two test patterns are applied is very important for the correct testing of the power switch. If pattern 2 is applied first, node V_{core} will be charged to *VDD* and hence sufficient time must be allowed for the complete discharge of node V_{core} before applying pattern 1. Otherwise, the output of the EX-OR gate will be '0' irrespective of the power switch functionality, which can also mean a false faulty behavior.

5.1.3 Testability of the Proposed Structure

In the proposed test circuitry, the output of the EX-OR gate is always tested for logic level '1'. A close look at the circuitry also reveals that it is not possible to drive a logic level '0' at the *Out* signal. Therefore, a stuck-at-1 fault at the *Out* signal cannot be detected. Furthermore, as the proposed test method requires *Out* to be at logic level '1', both test patterns will pass irrespective of the defects in the power switch. To circumvent such a situation, a test point can be added to the circuit so that the output of the EX-OR gate can be tested for stuck-at-1 fault. One example implementation of such a test point is shown in Figure 5.

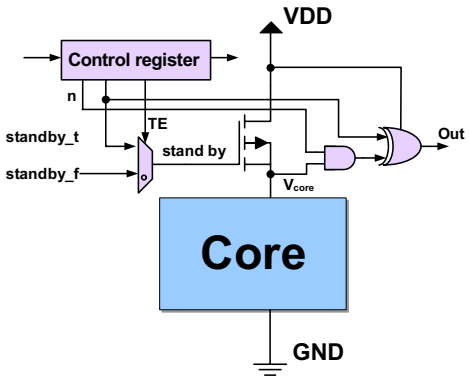

Figure 5: Adding test point to the proposed circuitry.

Here, we use a simple AND gate as a test point, however a transparent scan flip flop as well as other types of test points can also be used. The input *n* of the AND gate can be programmed via the control register. Therefore, by selecting signals as TE = 1, *n = '0',* and *standby_t ='0',* we can force a value '0' at *Out*. Now by observing the *Out* signal, we can check whether there is a stuck-at-1 fault at *Out* or not. It should be noted that all added test circuitry is powered from the undisrupted power supply *VDD*.

5.2 Header and Footer Switch

The above proposed circuitry and test method can be easily extended for a symmetrical power switch (header + footer switch). In this case, the proposed circuitry is added separately to both the header switch as well as to the footer switch. The resulted circuitry is shown in Figure 6.

Header and footer switches are tested sequentially, e.g. first the header switch is tested followed by the footer switch. The *select* signal at the output multiplexer selects the desired output node.

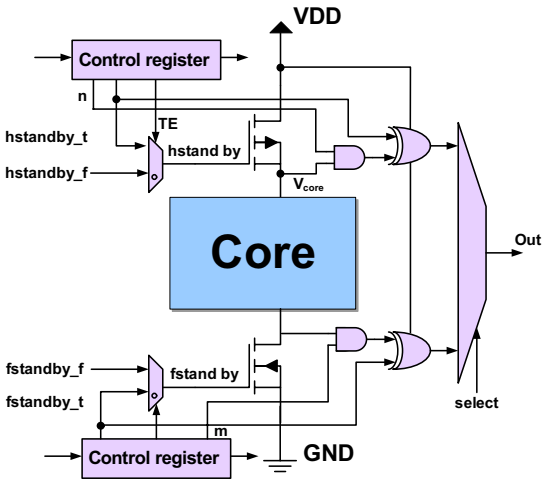

Figure 6: Test circuitry for a symmetrical power switch.

It is important to note that for clarity, two separate control registers and two TE signals are shown in Figure 6. However, a single control register can be shared between the header and the footer switch.

5.3 Segmented Power Switch

As described earlier, one of the most popular ways of implementing power switches is via a number of transistor segments. A segment can contain one or more transistors. The number of segments *m* and the number of transistors t_s in each segment *s* are determined at design time (see Section 4). Different segments can have different number of transistors. The above described test circuitry and method also enables testing of segmented power switches.

Now let us consider a general case. To test and identify individual failing segments, we use the concept of a *sliding window*. For every test pattern, a window on w ($k \leq w \leq m$) segments is selected. For any two consecutive patterns, the two respective windows have *w-1* common segments. In other words, there is an overlap q of *w-1* segments between two subsequent windows. The complete operation is repeated in a sliding window fashion until all segments are turned-on and off at least once; hence tested.

Figure 7 shows the test circuitry required for a two segment header power switch. The segments shown in the figure contain four transistors each. Please note that the positions of test bits (s1, s2…) in the segment control register can be optimized in order to minimize the register programming time. During the testing of segments, both *TE* and *n* are set to '1'. Let's assume $k = 1$. If we take window size $w = 1$, there cannot be any overlap between windows, i.e., $q = w\text{-}1 = 0$. For $w = 1$, three test patterns as shown in Table 1 are required to test the segments,. The values of the fault free as well as faulty responses for these patterns are also listed in the table.

The first pattern *s1* = '0', *s2* = '1', enables the first segment, which is controlled by *s1*, while the second segment is disabled. As it is assumed that turning on one segment is sufficient to provide a value of at least VDD_{min} at V_{core}, the *Out* signal should be '0'. Therefore, this pattern checks for the presence of a complete open in the first segment.

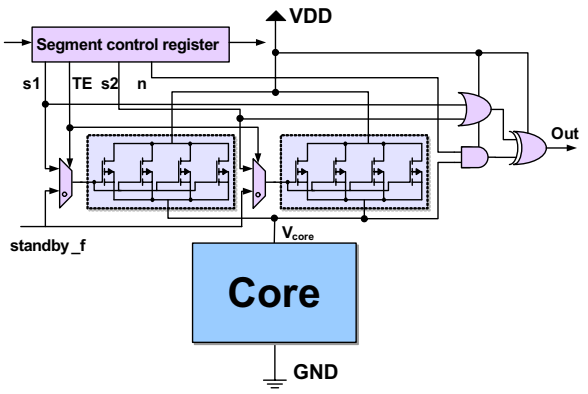

Figure 7: Test circuitry for segmented power switch.

Similarly the second pattern *s1* = '1', *s2* = '0', enables second segments and disables the first segment. In this case also, the *Out* signal should be '0'. This pattern checks for an open in segment two. The third pattern disables both segments, and hence V_{core} should be ideally at ground; and *Out* should be '1'. If this is not the case, then either of the two segments or both of them have a short to *VDD*. Therefore, by applying these three patterns, the two segments can be tested for on/off functionality as well as for possible manufacturing defects.

Pattern	*w* = 1			
	Inputs		Output	
	s1	*s2*	*Out*	
			Faulty free	Faulty
1	0	1	0	1
2	1	0	0	1
3	1	1	1	0

Table 1: Test patterns for testing a two segment power switch.

For a more generic case with *m* = 5, and *k* = 3, the sliding window concept is shown in Figure 8. In the figure, the window size is 3, i.e., *w* = *k* =3. For clarity, the core connected to the power switches is not shown and configurations for only few patterns are shown. In this case, the overlap between any two consecutive windows is of two segments.

If multiple segments are tested at a time, the sliding window concept is useful to identify in which segment the fault occurred; although it is not possible to pinpoint segment's faulty transistor. The concept of sliding window is also very useful when there are a large number of segments and testing them individually might not be very economical.

It is important to note that the resistance of the core R_{core} should be constant during the testing of all segments, since the value of *k* is selected for a particular R_{core}. The resistance of a core can be kept constant by configuring the core in a transport mode, while shifting a sequence of all 1's or all 0's through the scan chains in the core. Figure 9 shows an example of scan chains configured in transport mode. In the figure, signal *global_se* is the top level scan enable signal, while *se* is the local scan enable signal connected to the scan chains. By using the *transport* signal, scan chains can be configured in shift mode irrespective of the state of the signal *global_se*. Here, we can safely assume that during shift cycles, states of the scan chains are not propagated through the combination logic of the core.

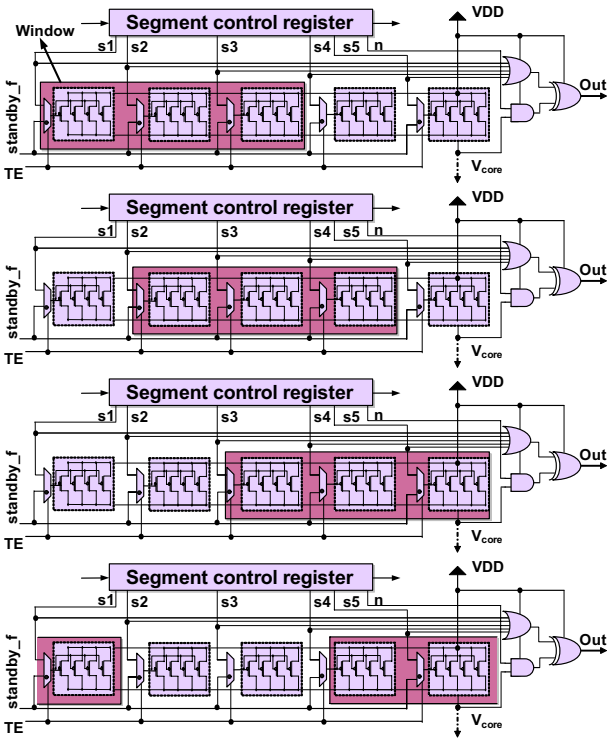

Figure 8: Testing of segmented switches by sliding overlapping windows.

Therefore, if the length of the longest scan chain in the core is l_{max}, then after l_{max} clock cycles, all flip flops in the scan chains will contain either a logic '1' or a logic '0' depending up on the input sequence (in Figure 9, a sequence of all 1's is shown). After this, the activity in the core will be just the clock activity. As the resistance of the core is directly proportional to the activity inside the core, by keeping the activity constant, the resistance is kept constant as well.

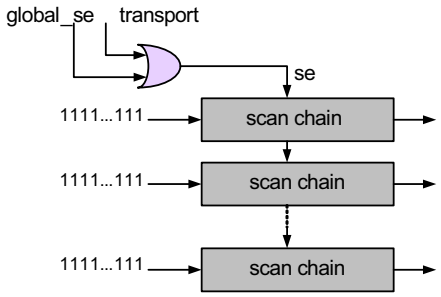

Figure 9: Configuring scan chains in transport mode.

5.4 Test Pattern Generation and Diagnosis

In the previous section, we showed how the sliding window concept can be used to test a given (*m, k*) segmented power switch. Just to test on/off functionality of a power switch, only two patterns, with all 1's and all'0 are sufficient. However, our objective is not just to the test power switches for on/off functionality but also to identify individual failing segments as many as possible. Therefore, for maximal diagnosis of individual failing segments, the window size *w* should be equal to *k*. Based on this, the num-

ber of test patterns $|P_m|$ required for a given (m, k) power switch can be calculated as follows.

$$|P_m| = m + 1, \quad \forall\ k = 1, \text{or } k = m$$
$$|P_m| = 2m, \quad \forall\ 1 < k < m$$

It is important to note that the number of patterns does not depend on the overlap q between the windows for two subsequent patterns. Basically, $p1$ distinct patterns with k 0's and (m-k) 1's are required to check whether there is a complete open in any of the segments, while $p2$ distinct patterns with (k-1) 0's and (m-k+1) 1's are required to check a complete short in any of the segments. As we want to test the segments using the sliding window concept, the patterns should contain uninterrupted runs of required 0's or 1's. To generate the required pattern set for a given (m, k), one needs to start with a m-bit vector with the first k bits as '0's and (m-k) bits as '1's. To obtain the next vector, one needs to circularly shift the sequence of k '0' bits to the right by one position. The shift operation needs to be carried out m-1 times to get $p1$ distinct test patterns. Similarly, for $p2$ patterns, one needs to start with an m-bit vector with the first k-1 bits as '0's and (m-k+1) bits as '1's.

For the case with k = 1, $p1$ is m, while $p2$ is 1 as it corresponds to a pattern with all 1's. Similarly, for the case with k = m, $p1$ is 1, while $p2$ is m. Therefore, the total number of patterns for these two boundary cases is m+1. For all other cases with $1 < k < m$, both $p1$ and $p2$ are m, therefore, $2m$ patterns are required to test power switches. Some examples of the generated pattern set for different values of (m, k) are given Figure 10.

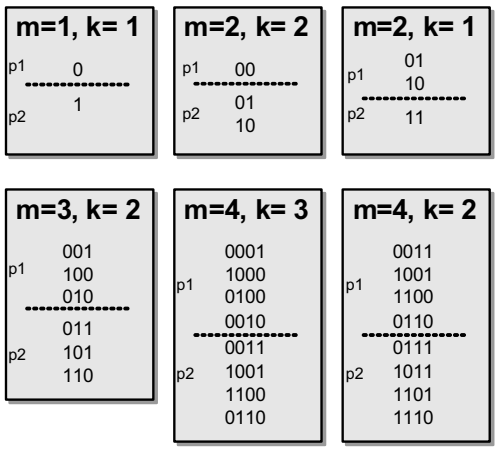

Figure 10: Test pattern set for a range of m and k

It is important to note that based on the value of k and m, it might be impossible to pinpoint at individual failing segments. For example, in cases with k = 1 and m > 1, it is not possible to detect which of the m segments has a possible short. Similarly, for the cases with $k = m$, segments cannot be diagnosed individually for a possible open. For maximum diagnosis of shorts in the segments, the value of k should be m-1; while for maximum diagnosis of opens in segments, the value of k should be 1. The proofs for these statements are omitted due to limited space.

6. Conclusion

To minimize dynamic and leakage power consumption, advanced power-aware design techniques are required for modern system chips. A very effective way to minimize leakage power consumption is to turn-off the parts of a chip, which are not active. For this purpose, power switches are used. Power switches can be implemented in various ways such as header switch, footer switch or a segment of transistors. In this paper, we presented a structured method to test power switches. The proposed method requires a simple circuitry and works well for all types of existing power switches. The proposed method requires a few test patterns and also enables diagnosis of faulty segments in case of segmented power switches. As a future work, we are planning to put the proposed structure in silicon.

Acknowledgements

We thank our colleagues Erik Jan Marinissen and Luis Elvira Villagra for their useful comments on an earlier draft version of this paper.

References

[1] Semiconductor Industry Association (SIA), *International Technology Roadmap for Semiconductors, 2005.*

[2] K. Roy, S.Mukhopadhyay, and H. Mahmoodi-Meimand, *Leakage Reduction Mechanism and Leakage Reduction Techniques in Deep-Submicrometer CMOS Circuits,* Proceedings IEEE, volume 91, pp.305-327, February 2003.

[3] M. Meijer, F. Pessolano, and J. Pineda de Gyvez, *Technology Exploration for Adaptive Power and Frequency Scaling in 90nm CMOS*, Proceeding IEEE ISLPED, pp.14-19, 2004.

[4] J. Kao, S. Narendra, and A. Chandrakasan, *MTCMOS Hierarchical Sizing Based on Mutual Exclusive Discharge Patterns*, Proceedings IEEE/DAC Design Automation Conference (DAC), pp.495-500, 1998.

[5] S. Kim, S. Kosonocky, and D. Knebel, *Understanding and Minimizing Ground Bounce During Mode Transition of Power Gating Structures,* Proceedings IEEE ISLPED, pp.22-25, 2003.

[6] S. Kosonocky, M. Immediato, P. Cottrell, T. Hook, R. Mann, and J. Brown, *Enhanced Multi-Threshold (MTCMOS) Circuits using Variable Well Bias.* Proceedings IEEE ISLPED, pp.165-169, 2001.

[7] Y. Zorian, E.J. Marinissen, and S. Dey*, Testing Embedded-Core Based System Chips*, Proceedings IEEE International Test Conference (ITC), pp. 130-143, 1998.

[8] S.K. Goel, K. Chiu, E.J. Marinissen, T. Nguyen and S. Oostdijk, *Test Infrastructure Design for the Nexperia™ Home Platform PNX8550 System Chip*, Proceedings IEEE Design Automation and Test in Europe (DATE), pp. 108-113, 2004.

[9] D. Lackey, P. Zuchowski, T. Bednar, D. Stout, S. Gould and J. Cohn, *Managing Power and Performance for System-on-Chip Designs using Voltage Islands,* Proceedings IEEE International Conference on Computer-Aided Design (ICCAD), pp. 195-202, 2002.

[10] R. Puri, L. Stok, J. Cohn, D. Kung, D. Pan, D. Sylvester, A. Srivastava and S. Kulkarni *Pushing ASIC performance in a power envelope*. Proceedings IEEE/ACM Design Automation Conference (DAC), pp. 788-793, 2003.

[11] R. Puri, D. Kung and L. Stok, *Minimizing Power with Flexible Voltage Islands*, Proceedings IEEE International Symposium on Circuit and Systems (ISCAS), pp. 21-24, 2005

[12] Power Wise Interface (PWI) Open Standard, *http://www.pwistandard.org.*