Allowing Switching off Periphery Voltage Island Instead of Doing it per Instance Through Periphery VDD Collapse in SRAMs

Krashna Nand Mishra
Memory Compiler Group
Synopsys (India) Pvt Ltd
Noida, India
krashna@synopsys.com

Ruchin Jain
Memory Compiler Group
Synopsys (India) Pvt Ltd
Noida, India
ruchin@synopsys.com

Shailendra Sharad
Memory Compiler Group
Synopsys (India) Pvt Ltd
Noida, India
shai@synopsys.com

Ravindra Shrivastava
Memory Compiler Group
Synopsys (India) Pvt Ltd
Noida, India
rsrivast@synopsys.com

***Abstract*—In typical SoC Designs, having multiple SRAM cores on single chip has demanded switching off idle cores to attain lowest possible power down leakage because of SRAMs. This paper introduces a novel method called Periphery VDD Collapse to address this aspect by achieving nearly zero periphery leakage when in power down mode. By virtue of having switches outside memory, this feature allows designers to switch off supply for a voltage island containing multiple cores instead of doing it per memory. A novel level shifter interface is implemented to characteristically isolate periphery from bit cell array and allow periphery to be switched off while retaining data present in bit cell array. Experimental results show very minimal change in power and leakage compared to the case when power gates are embedded inside, but saves up to 10% area after being moved outside on chip.**

Keywords—Periphery VDD Collapse; Power gating; Level Shifter; Low leakage; Data Retention

I. INTRODUCTION

Along with the advent of technology scaling, low power VLSI technology has been an increasingly demanding and growing area in electronic chip design space. The possibility of creating low power designs made battery usage quite promising and led to create battery powered portable systems. With increasing use of battery powered portable applications coming to the fore with greater demand day by day, chip designers also had looked at various other possibilities and circuit innovations to further reduce power in Integrated circuits and chips. Low power Static Random Access Memories(SRAMs) is that major area of technology where there has been rapid progress in power reduction and ever growing electronics industry continue to demand more from SRAMs as they occupy a good space on chip.

Also, one of the negative side effects of technology scaling is that leakage power [1,2] of on-chip memory increases dramatically and forms one of the main challenges in future System-On-chip (SoC) designs. Therefore, an efficient memory leakage suppression scheme is critical for the success of ultra-low-power design.

Chip designers over the evolution of low power designs have perceived the idea of supply reduction as an easy solution to reduce both active power and leakage. And over the years it led to voltage scaling and using multiple power supplies [3] on a single chip. With having different supplies available, it became imminent for designers to segregate logic under different voltage islands to give themselves flexibility to switch off islands, which are not critical under power down scenario.

Thus, at the architectural level, leakage reduction can be achieved by gating off the supply voltage [4] of idle memory sections, or putting less frequently used sections into drowsy standby mode. This approach exploits the exponential reduction of leakage power with VDD, and achieves optimal power-performance tradeoffs with assistance of compiler-level cache activity analysis.

Under dual rail scenario for SRAMs, two voltage supplies are used, one for periphery section, which usually comprises of control circuitry, decoding logic and input/output IOs and the other, to bias bit cell array. This paper introduces the aspect of switching off primary periphery power supply, called VDDLOGIC of SRAMs to achieve almost zero leakage in periphery core and in turn support switching off entire periphery Voltage Island on chip.

A. Understanding Periphery VDD Collapse

To support low leakage/sleep modes in SRAMs, power gates are usually embedded inside the memory. When memory is to be put in sleep mode, these power switches from external supply VDDLOGIC to internal periphery supply VDDLOGIC_I are turned off, allowing VDDLOGIC_I to drop down to reduce leakage in periphery.

There are certain concerns associated with having power gates embedded inside. First, designers happen to use VDDLOGIC to bias sleep mode controls, which constrains VDDLOGIC to remain biased during power down modes.

Now let's look at it from SoC design perspective where VDDLOGIC supply is shared by several other peripherals which collectively form VDDLOGIC voltage island, referring

2380-6923/19/$31.00 ©2019 IEEE
DOI 10.1109/VLSID.2019.00097

fig 1. Under a scenario when VDDLOGIC set of peripherals are supposed to be idle, so to achieve low leakage, designers cannot switch off VDDLOGIC voltage island as VDDLOGIC is being used by SRAM periphery in memories. They instead need to turn off power gates individually for all peripherals and memories to save leakage. And if there exists number of SRAM instances inside a single voltage island, it makes the process cumbersome as it must be done per memory.

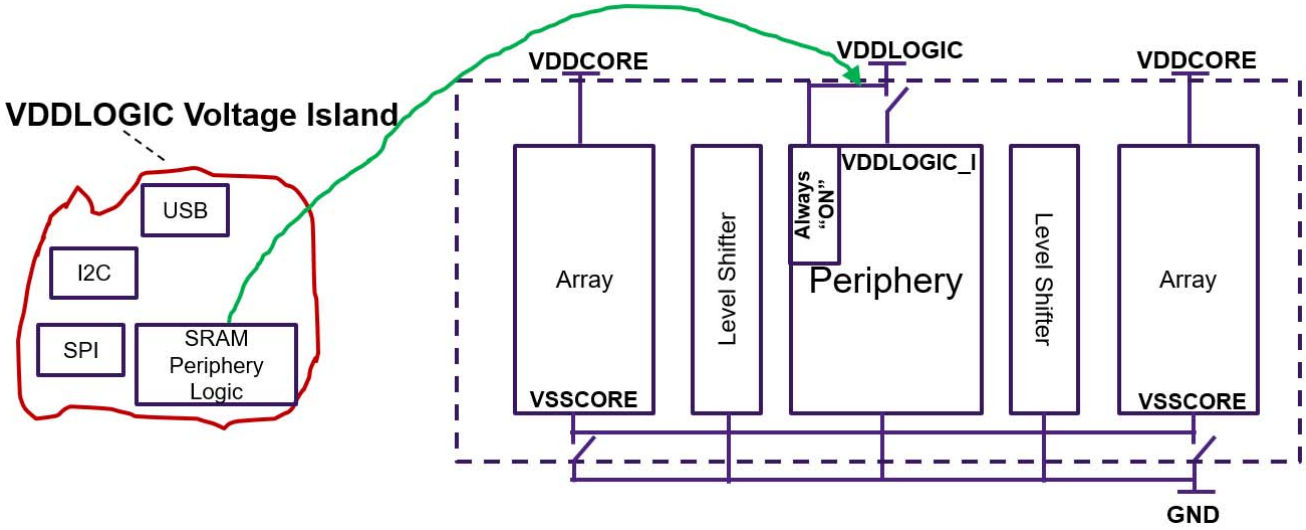

Fig 1 Understanding Periphery VDD Collapse

Secondly, since there is no external control over VDDLOGIC_I ramp-down profile, internal logic decides what voltage VDDLOGIC_I would settle down to during sleep modes and thus leakage saving is decided by internal memory design, and it's not in chip designer's control.

These thoughts give rise to possibility of having power gates outside memory, on chip to switch off entire voltage islands and let memory designers update circuits to enable SRAMs inside islands to function properly under such design requirements, and that gives birth to the concept of Periphery VDD Collapse in SRAMs.

The remainder of this paper is organized as follows: Section 2 explains typical low power SRAM architecture and briefly explains how power supplies are usually switched to achieve lower power down leakage, Sector 3 contains details of novel level shifter interface proposed in this paper to implement Periphery VDD collapse. Section 4 explains timing relationship among various primary signals to ensure proper removal/recovery of periphery collapse enable signal, PERICE. Section 5 shows leakage simulation results for few memory instances to show efficiency of implemented design. Finally, Section 6 concludes current work, and finally list of selected references.

II. LOW POWER SRAM ARCHITECTURE OVERVIEW

Fig1, the right part of it shows a typical low power SRAM architecture scheme.

It contains periphery core logic in the center with bit-cell arrays on either side in center decode fashion. Array on either side consists of memory cells capable of storing one bit each. Each cell in the array is connected to one of the word lines coming from periphery and a pair of column lines controlled by logic in the periphery for read/write operations. A specific cell is selected for reading or writing by activating its word line and bit lines.

To control power down currents, periphery is powered by switched supply VDDLOGIC_I during power down and array leakage is managed by disconnecting internal VSSCORE node from ideal GND and elevating it through source biasing. VDDLOGIC is biasing only "always-on" control logic to determine control signals for VSSCORE/VDDLOGIC_I switches.

One of the perceived concerns with above architecture is that primary periphery supply VDDLOGIC should be kept up during power down modes to bias "always-on" logic, which constrains it from completely switching off VDDLOGIC supply of memory from outside and in turn switching off entire island on VDDLOGIC supply.

III. NOVEL LEVEL SHIFTER INTERFACE

Now consider the situation when VDDLOGIC is collapsed from outside using power switches moved on chip, the places it's going to create issues are the ones where VDDLOGIC interfaces with other supply domains inside memory. The level shifter is one such interface. To put word lines in VDDCORE domain, level shifters are usually inserted right before the word line drivers. Fig 2 shows conventional voltage level shifter [5] design.

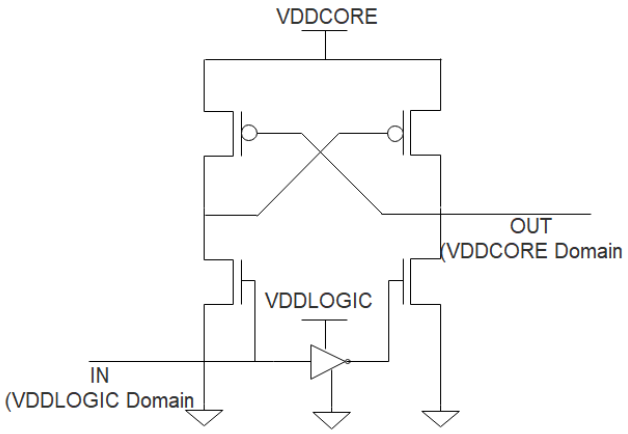

Fig 2 Conventional Voltage Level Shifter Design

In Periphery collapse case, signals powered at VDDLOGIC would go floating, and so does IN, input to level shifter.

Since IN/INB go floating, it creates DC current/sneaking leakage paths between VDDCORE and GND and, also may send bit-cell interface signals like word-line to go in in-deterministic state and cause data retention loss.

This paper introduces a novel level shifter interface which characteristically isolates VDDLOGIC and allows it to collapse to achieve min leakage possibility while keeping the bit cell data retained by ensuring proper levels for bit-cell interface signals. We use single pin PERICE (Active high Periphery Collapse Enable) to device a scheme which would avoid any possible leakage paths. Fig 3 shows level shifter with updated interface signals.

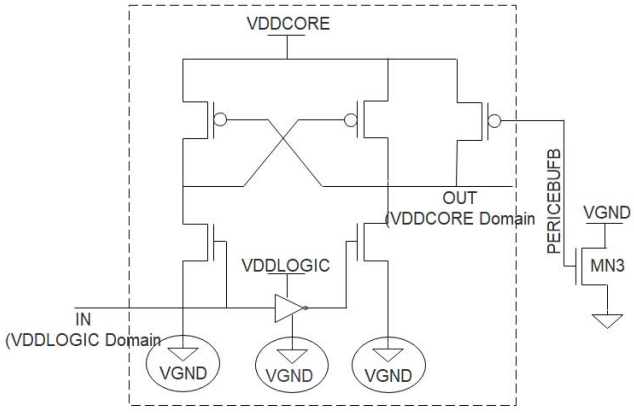

Fig 3 Voltage level shifter design including updated interface signals

When PERICE goes high, virtual ground is established for the ground nodes of level shifter by switching off NMOS device MN3. PERICEBUFB is buffered inverted version of PERICE signal, which is used as control, shown in fig 3 to make VGND floating so that signal coming in as input to level shifter won't be able to create any leakage paths from VDDCORE as ideal current sink is removed from all NMOS source terminals. MN3 is a relatively bigger size switch, which is used per bank to connect VGND nodes of all level shifters corresponding to word lines.

Same PERICEBUFB is used to ensure that output of level shifter is clamped to VDDCORE which in turn clamp word line at '0' ensuring bit-cell content not getting disturbed because of non-zero state of word lines. Design ensures that VGND goes floating first, followed by level shifter output going high to avoid any short circuit paths.

Moreover, given the memory architecture shown in fig 1, apart from disabling word lines, we need to take care of other signals which could affect data retention. Fig 4 shows the extension of fig 1 containing details of source biasing logic connected to VSSCORE node. It introduces two enables-TDCE and DDCE. When the memory is put into sleep mode, TDCE goes low to disconnect VSSCORE from GND, and DDCE goes high, which connects VSSCORE node to source biasing circuitry to elevate VSSCORE higher than GND to reduce bit cell array leakage even as ensuring that VSSCORE only rises appropriately enough to keep ensuring data retention [6].

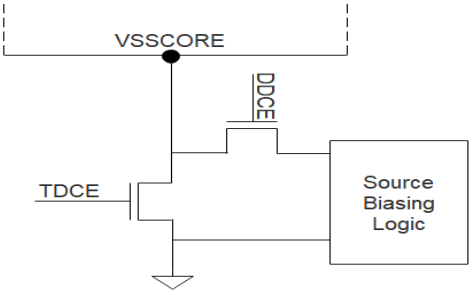

Fig 4 Bit cell array source biasing

Since TDCE/DDCE controls are VDDLOGIC biased, they would also go floating during periphery collapse mode. Hence level shifter used for word lines, explained in fig 3 can also be used to ensure that TDCE/DDCE are asserted properly during periphery collapse.

A simple way to achieve this is to insert level shifter just before TDCE/DDCE drivers and move driving logic to VDDCORE (always-on bit cell array voltage), as shown in fig 5.

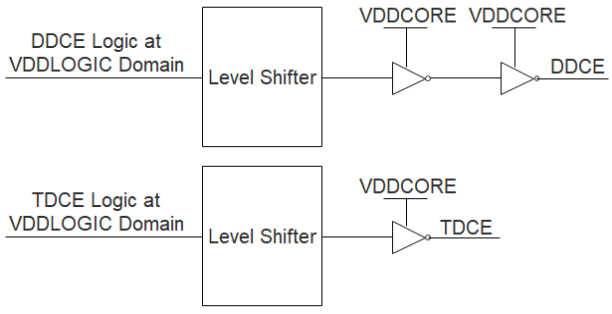

Fig 5 TDCE/DDCE logic update to support Periphery Collapse

Thus, properly isolating logic at voltage interfaces ensures that we can continue to support data retention power down modes even in absence of periphery supply VDDLOGIC.

IV. TIMING DIAGRAM AND CHARACTERIZATION PARAMETERS

To ensure proper functional operation and periphery VDD collapse, PERICE timing relationship must be maintained with other relevant signals. Fig 6 shows timing diagram which indicates timing requirements. Below is the definition of timing parameters displayed in the figure.

- *perice_hold* is the time from CLK rise to PERICE rise. It is to ensure no normal read/write operation is tried while collapse is taking place. For simplicity, this could be equated to cycle time of CLK.
- *perice_removal* is defined as PERICE Removal time, PERICE rise to VDDLOGIC Collapse. VDDLOGIC can collapse only after this delay. This time is needed for creating isolation in level shifters.
- *perice_recovery* is PERICE Recovery time after VDDLOGIC wake-up. This is to make sure periphery circuits are set properly before PERICE pin falls.
- *perice_setup* is defined as PERICE recovery time from PERICE fall to the next CLK for valid read/write operation. It's time for source biased internal node VSSCORE to reach normal ground potential.

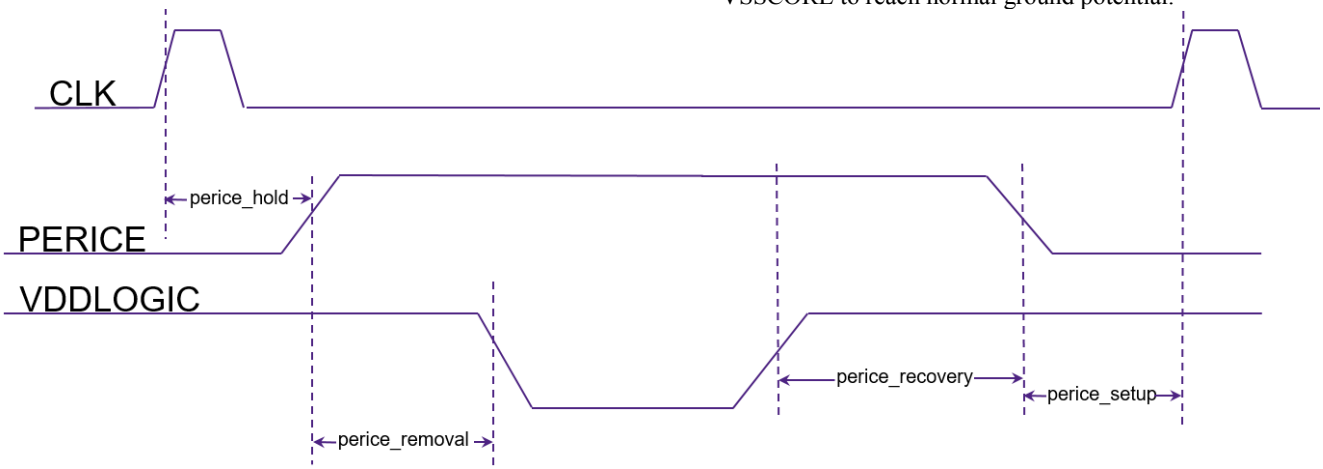

Fig 6 Timing diagram and characterization parameters

A. Characterizing periphery VDD leakage

Since VDDLOGIC is an external supply to memory being collapsed from outside during periphery VDD collapse mode, profile of collapsing it would largely be driven by chip level specifications and requirements. Thus it's important to analyze how VDDLOGIC leakage would be characterized.

Let's consider a system wherein there're stream of VDDLOGIC switches powering the VDDLOGIC island containing memory and other peripherals, as shown in fig 7.

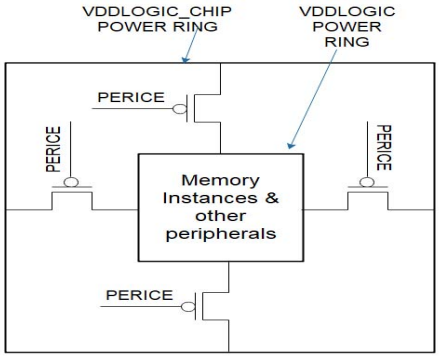

Fig 7 Switch Array around periphery at SoC

VDDLOGIC_CHIP is external chip regulator supply and VDDLOGIC is the switched chip supply dedicated to this island. PERICE is switched HIGH when SRAM is put into periphery collapse mode.

Total estimated leakage power, P_{EST} of the system would be

$$P_{EST} = I_{LEAK} *(VDDLOGIC\_CHIP-VDDLOGIC) + I_{MEM} *(VDDLOGIC)$$

Considering I_{LEAK} is the leakage current through switch when PERICE Control is high and I_{MEM} is the leakage through SRAM in collapse mode. Since $I_{LEAK} = I_{MEM}$,

$$P_{EST} = I_{LEAK} *VDDLOGIC\_CHIP$$

Thus even if VDDLOGIC settles at some intermediate potential depending on leakage through switch and how much leakage SRAM can sink in, system leakage would always be determined by leakage through switch. So VDDLOGIC leakage for memory can be put zero in liberty files as system leakage would always come from external switch at SoC.

V. SIMULATION RESULTS

To evaluate the cost of added Collapse feature, post layout simulations have been carried out on corner compiler memory instances. Leakage simulations are performed using HSIM on TSMC16FFC process, on FF/125C dual rail corner with

VDDLOGIC 0.77V and VDDCORE 0.88V, and are captured below in fig.8. Table I shows the instance configurations used for analysis and Table II shows % area impact due to addition of collapse feature.

TABLE I. MEMORY INSTANCE CONFIGURATIONS USED FOR ANALYSIS

Instance	No. Of Words	No. Of Bits	Column Mux	No. Of Bank	Rows Per Bank	No. Of Column
Big	2048	128	4	2	256	256
Tall	2048	8	4	2	256	16
Wide	16	512	1	1	16	256
Small	16	16	1	1	16	16

Fig 8 shows clearly that leakage in collapse mode compared to power down retention mode when power gates were inside, is almost same. It confirms that adding collapse mode and moving some logic to always on array supply hasn't added extra leakage.

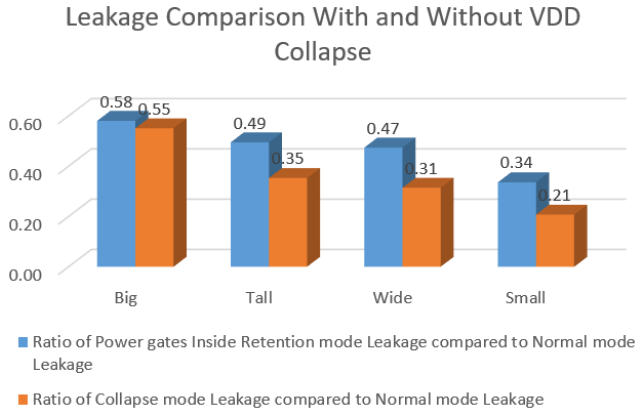

Fig 8 Leakage results with and without collapse (in µA)

TABLE II. AREA COMPARISON

Instance size	% Area Saving (Area with Retention switches outside Memory (Periphery Collapse Mode) Vs Area with Retention switches embedded inside the memory
Big	1.65%
Tall	1.65%
Wide	9.44%
Small	9.44%

As shown in table II, collapse logic though takes additional area in level shifter tile, but moving power gates outside effectively saves area, more for wider (more IOs) instances where number of power switches used could be more.

VI. CONCLUSIONS

Periphery VDD collapse is a design feature which allows periphery voltage to be brought down to zero and enables switching off periphery Voltage Island using power gates on chip, instead of doing it per memory using switches embedded inside memory.

The solution proposed in this paper is of great concern mainly due to the flexibility it gives to chip designers while trying to squeeze down leakage as much as possible and because of very little extra cost associated with this feature. Simulation results show almost no change in dynamic power with very minimal power and speed overhead.

REFERENCES

[1] S. Borkar, "Design challenges of technology scaling," IEEE Micro, Vol. 19, No. 4, pp. 23-29, 1999.

[2] K. Bernstein, C. T. Chuang, R. Joshi and R. Puri, "Design and CAD challenges in sub-90 nm CMOS technologies," ICCAD, pp 129-136, Nov. 2003.

[3] N. Chabini, I. Chabini, E. M. Aboulhamid, and Y. Savaria, Methods for minimizing dynamic power consumption in synchronous designs with multiple supply voltages. IEEE Transactions on ComputerAided Design of Integrated Circuits and Systems 22, 346 (2003).

[4] Hailin Jiang; Marek-Sadowska, M.; Nassif, S.R,"Benefits and costs of power-gating technique" in ICCD,2005 Page(s):559 – 566

[5] Bert Serneels, Michiel Steyaert and Wim Dehaene, "A high speed, low voltage to high voltage level shifter in standard 1.2V 0.13µm CMOS," Analog Integr Circ Sig Process Vol. 55, pp. 85-91, 2008.

[6] K. Anami, M. Yoshimoto, H. Shinohara, Y. Hirata, and T. Nakano, " Design considerations of a static memory cell," IEEE J, Solid-State Circuirs, vol. SC-18, no. 4, pp 414-418, Aug. 1983

[7] N. Kim, "Drowsy instruction caches: leakage power reduction using dynamic voltage scaling and cache sub-bank prediction," Proceedings of the 35th Annual Int'l Symposium Microarchitecture (MICRO-35), IEEE CS Press, 2002. pp. 219–230.

[8] H. Qin et al, "SRAM leakage suppression by minimizing standby supply voltage," in Proc. of ISQED, 2004, pp.56-60.

[9] Michael Keating, David Flynn, Robert Aitken, Ala Gibsons and Kaijian Shi, "Low Power Methodology Manual for System on Chip Design", Springer Publications, New York, 2007