Dynamic Fine-Grain Body Biasing of Caches with Latency and Leakage 3T1D-based Monitors

***Abstract*—In this paper, we propose a dynamically tunable fine-grain body biasing mechanism to reduce active & standby leakage power in caches under process variations. Front body biasing (FBB) is employed for active sub-arrays to speed up accesses while inactive sub-arrays are reverse body biased (RBB) to reduce standby leakage power. Cache sub-arrays are classified based on run-time leakage and latency distributions and applied bias voltage is updated to account for these changes. This ensures that under all scenarios, the cache will consume the lowest leakage power for the target access latency computed at design-time. The backbone of the hardware used for classification is the 3T1D DRAM cell embedded into conventional 6T based memory. By measuring the access and retention time of the 3T1D cell, we show that it is possible to classify cache arrays based on run-time latency/leakage measurements. The tremendous increase in energy in active mode by forward biasing (to meet latency requirements) is compensated by reverse biasing inactive arrays. Our technique reduces leakage energy consumption and access latency of the cache on average by 20% & 18% respectively when compared to other state-of-the-art proposals. Finally we show that our technique will improve parametric yield by a maximum of 38% for worst-case scenario.**

I. Introduction

Tremendous advancements in chip design have made possible billion-transistor integration over the last decade. This can be largely attributed to the improving capabilities of the manufacturing processes. However manufacturing has improved only to such extent that the problem of spatial variability of transistor parameters is inhibiting the power/performance gains achieved by scaling devices. Increasing power densities is another major cause of concern for high performance/low power designs. Power is dissipated in the form of heat leading to increased heat densities. With increase in temperature, the leakage power increases exponentially. Recent study has shown that operating temperature of chips can be as high as $90\,^{\circ}\mathrm{C}$ and in some cases as high as $120\,^{\circ}\mathrm{C}$ [1]. Frequent temperature shootups can result in functionality problems and cause permanent damage in the form of faults due to electromigration, thermal cycling & stress migration [2]. These faults can have a long lasting effect on the processor performance over the lifetime of the chip. While design-level techniques can be used to meet certain design constraints by improving tolerance to process variations, it is impossible to design considering worst-case temperature or power (in-turn leakage & delay) conditions owing to reduced yield and revenues.

As caches are a very important component from an area point of view, it becomes extremely challenging to optimize chip yield keeping in mind the effects of spatial variations of process parameters and temporal variations of temperature & power. Recent proposals [3], [4], [5] have suggested that post-silicon adaptivity can be used to improve SRAM yield and also reduce power consumption. Post-silicon adaptivity involves measuring leakage/latency of SRAM(s) and provide on-chip mechanisms to enforce circuit-level optimizations based on these measurements. Body biasing (BB) is one such technique. The threshold voltage of the transistor which is dependent on body source potential is modulated to improve performance or reduce leakage. In front body biasing (FBB), application of positive bias voltage reduces threshold voltage making transistors faster but at the cost of increasing leakage. In reverse body biasing (RBB), a negative voltage increases the threshold making transistors slower and also less leakier. Tolerance to process variations can be improved by utilizing both RBB and FBB and this is called adaptive body biasing (ABB). Based on latency/leakage measurements obtained at manufacturing time, bias voltages (either FB or RB) are set permanently for the lifetime of the chip. While this would greatly reduce the impact of spatial variations (die-to-die), susceptibility to temporal variations increases. Also, techniques for BB are targeted for an entire chip not accounting for the effects of within-die variations. Thus conventional techniques employing ABB result not only in heterogeneous performance across time but across functional blocks as well. In order to reap maximum benefits would require fine-grain control of functional blocks by measuring the latency/leakage local to that particular block at run-time and applying a optimal body bias that trades off leakage for latency. This is called dynamic fine-grain body biasing (DFGBB) [6]. In essence, this is a 2 step mechanism that requires a sensor like unit (to measure the latency/leakage) to be interfaced with a body bias control unit for generating an optimal bias voltage based on the measurements.

The paper makes the following contributions,

1.) As a first step, we present a novel three-transistor one-diode (3T1D) DRAM-based latency/leakage measurement hardware specially targeted towards memory structures such as register files & caches. By embedding a 3T1D into a 6T sub-array, we show that each read (or write) to the 3T1D cell will suffer almost the same variation on access power and latency when compared to any 6T cell in that sub-array (since it will use the same periphery circuits and the physical variations will be almost identical between the cells due to their proximity). The retention time and access time of the 3T1D are measured to determine the effects of process variation (spatial) on leakage and latency of the memory array. Because of the transient nature of both latency and leakage, the mechanism behaves well in tracking temporal changes.

2.) The above mechanism is interfaced with a modified version of the lookup table based adaptive FBB generator [7]. In addition, a hybrid charge pumping circuit is used for generating the negative bias required for RBB. By exploiting the unique access patterns that caches exhibit, active sub-arrays are forward biased while inactive/unused sub-arrays are reverse biased in a very speed effective manner. Not only does this offer enhanced access speeds (forward biasing), tremendous leakage power reduction is made possible by reverse biasing multiple unused sub-arrays.

This paper is organized as follows. In section 2, we discuss about the 3T1D cell and its performance in the presence of variations. In section 3, we propose a new hardware for classifying cache sub-arrays based on latency/leakage. Section 4 presents the fine grain body bias generator that is interfaced with the hardware presented in section 3. In section 5, leakage/latency improvements of the proposed scheme are analyzed. Section 6 makes a review of existing work in the literature. Section 7 presents the concluding remarks.

II. 3T1D Cell

Alternatives to 6T based SRAM have been researched diligently for want of increased memory density and lower vulnerability to variations. One such proposal is the 3T1D cell proposed by Luk *et al.* [8]. The capacitorless DRAM cell stores the data using a gated diode that is tied to the read-wordline as shown in Figure 1. The 3T1D unlike 1T DRAM memory provides non-destructive reads and access speeds comparable to that of standard 6T SRAM(s). When compared to 6T, the transistors of the 3T1D can be asymmetrical in strength. This has 2 fold advantages over 6T: Primarily, process variations causing device mismatch are likely to cause less failures to the cell [9]. Secondly, it improves the overall stability making it radiation hardened. Data is written into the cell by raising the write-wordline high and charging the bitline. Unlike a 6T, the voltage level at the storage node is degraded and roughly about $0.6V_{dd}$. A strong T1

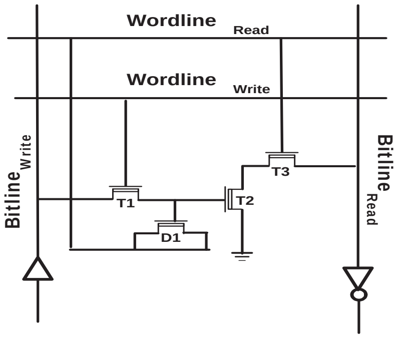


Fig. 1: Schematic of the 3T1D Cell

would further degrade the voltage resulting in lower retention time. This can be avoided by increasing the threshold of the write driver [8]. The read operation is initiated by precharging (to V_{dd}) the read-bitline and strobing the read-wordline. The retention time of the cell can further be increased by holding the read-wordline at a negative voltage during idle state. This has shown to increase the retention rate by as much as 40X [8]. Liang *et al.* have proposed a 3T1D-only cache architecture that offers 6T like performance but better robustness to process variations [10]. In this paper, we interleave the 3T1D cells in the memory arrays to use them as latency and leakage sensors as we will explain in short. The arrays keep the original 6T SRAM cells for program execution.

A. Retention & Access Time

For the sake of comparative study, a 3T1D and 6T are embedded into the same array sharing all the periphery and wordlines as shown in Figure 2. The SRAM cell is a single ported cell that is found in register files. With minor modification to the column multiplexer and write drivers, the 3T1D can be embedded into a conventional 6T (dual-ended) array found in caches. The access latency of both cells is measured independently and normalised to 6T's value at 30 °C. Looking at figure 3, only at high temperatures, 6T cell is more prone to performance loss when compared to the 3T1D. As opposed to regular 6T cells, the 3T1D are designed for single ended sensing. This combined with T2's (ref figure 1) boosting action provides very high read speeds even at high temperatures. This validates the fact that both 6T and 3T1D have similar access latency & in some sense mimic each other's functional behavior. While the 6T cell could already be used to measure access latency, 3T1D provides an extra measurable parameter called retention time. The retention time of the 3T1D is defined as the time taken for the voltage at the storage node to decay past $V_{dd}/4$. In [11], it is shown that the leakage through the cell is directly proportional to the retention (decay) time of the cell. In other words as leakage increases, the retention time decreases and vice-versa. Figure 4 shows the measured retention time for 500 caches simulated for spatial-temporal variability. The retention time is normalized to the lowest retention time at 110 °C. The samples are organized in order of reducing magnitude of retention time. Retention time with zero-variability at 30 °C is found to be 9.3μs. Under the presence of process variations, operating at 30 °C, the retention can be as high as 34.2μs or as low as 5μs. Because of the exponential relationship between leakage and temperature, the retention time can be as low 980ns at 110 °C under worst case process variations. It should clear from the above argument that both access & retention time of the 3T1D are an important figure of merit that can be measured to reflect the 6T's latency and leakage power variation under the effects of spatio-temporal variability.

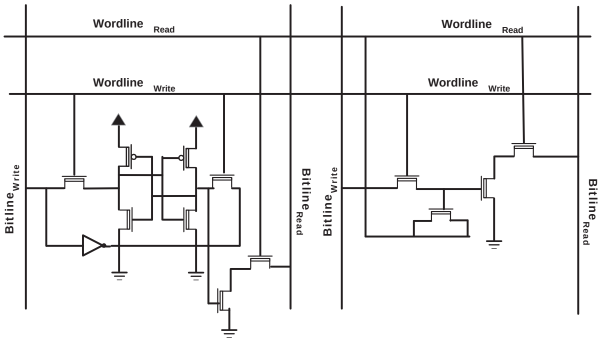


Fig. 2: 3T1D embedded with a 6T SRAM

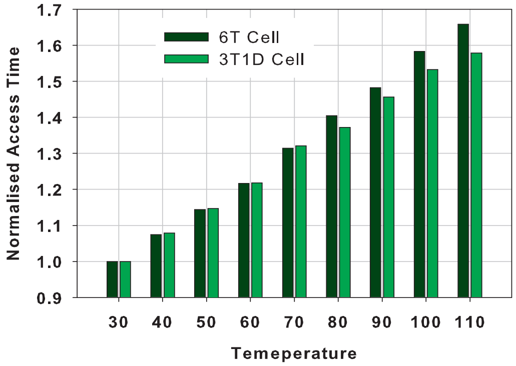


Fig. 3: Access Time Vs Temperature

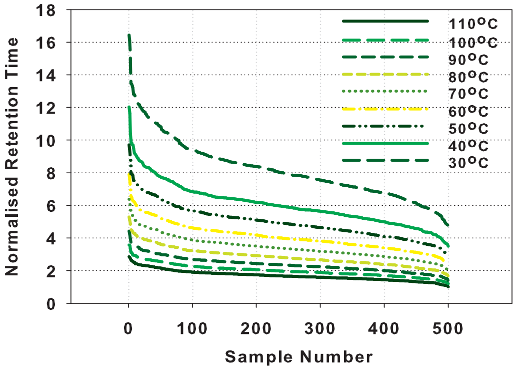


Fig. 4: Retention Time Vs Temperature

B. Simulation Setup

The simulated cache is 32KB in size with multiple 1KB sub-arrays. Each sub-array is organized into 128 columns by 64 rows with a 32 bit read-out. Due to area constraints, the decoders are designed with dynamic cmos and column multiplexer is tree-like design. Because random variations are known to affect 6T cells more than systematic variations and there is no definite way of tracking random variations, 1 3T1D per sub-array is more than sufficient[12]. We modify the column multiplexer and write drivers in order to accommodate the 3T1D in a regular 6T array. The associated area & energy overhead was estimated to be 0.31% and 0.78% respectively. For modeling process variations, we adopt the quadtree based multi-level partition scheme [13]. The σ for systematic and random variation of V_{th} is 6.4%. It is assumed that variances of intra-die systematic and random variation to be equal. Due to the strong correlation between parameter values, it is assumed that variance of L_{eff} to be half of that of V_{th} [12]. The systematic and random variations of L_{eff} is derived as 3.2%. Inter-die variations of both parameters is set to an offset value of 3%. 500 samples of the cache are simulated on HSPICE with 45nm PTM [14].

III. Latency/Leakage Measurement

A. Run-Time Classification of Cache Sub-Arrays

The purpose of classifying cache sub-arrays individually at run-time is very much alike calibration. Calibration enables to rectify any deviations that arise out of manufacturing or during the lifetime of the chip (i.e. degradation). Most often run-time circuit level optimization like body & source biasing, supply voltage minimization that have been proposed for leakage minimization are enforced without this available information [15], [16]. Such optimizations resulting from holistic procedures have been enforced across varying chips yielding non-uniform benefits. For any optimization that needs to extract maximum benefits using the available leakage/latency measurements, the granularity of the classification have to be very fine as shown in Figure 5(classification & measurement are used in a interchangeable fashion). A very high access time and low retention translates directly to high access latency and significantly high leakage power. This is the one of the non-ideal cases that we would like to avoid at any cost. For the sake of simplicity, we would like to call each discrete combination of leakage and latency a bin. The nomenclature used (min,low,high,max) is specific to our scheme and is not representative of the actual degree of separation. It is well established that both latency and leakage are transient and by generating this table-based data, on-die registers can be frequently updated with this information to be made available for cross-layer optimizations. By making the latency/leakage bounds more tight during classification, circuit optimizations can have more fine-grain control by having better cognizance of the power/performance status of each sub-array.

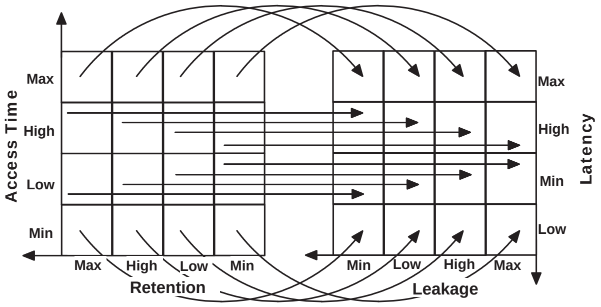


Fig. 5: Discrete Classification based on Latency/Leakage

B. Discretization Architecture

As temperature has a more observable effect on the retention time when compared to access time, we begin with the classification based on leakage. Theoretically to measure the retention time, we could use a simple delay-to-pulse circuitry to count the number of cycles the voltage corresponding to a ’1’ at the storage node of the 3T1D takes to decay by sending a continuous stream read requests one after another and waiting till the voltage degrades completely. This has 2 fold disadvantages. It was earlier shown that retention time is of the order of μs. This means that it would require hundreds of thousands of cycles for a counter with a low pulse-width clock to complete the operation. Response mechanisms typically are expected to have very fast response in the order of 1000 of cycles. As a simple rule of thumb, faster the response, greater the benefits. Further, lower the leakage, higher the retention and so longer the time it takes it complete the operation. This is furthered by the decaying of the voltage at the storage node which makes subsequent read accesses slower. It was found that by strobing the read-wordline high continuously instead of sending read requests one after another, this problem can be alleviated as a result retention time reducing by nearly 20X to around hundreds of nanoseconds [8]. The decaying behavior of the storage node is replicated at the output of the sense amplifier. It is this window of few hundred nanoseconds that is deeply impacted by parameter variation. The measuring hardware just has to convert the time the output of the sense amplifier is high into something measurable on-chip. Any scheme that involves a delay-to-pulse circuitry can generate a clock cycle for every period that the output of the sense amplifier is held high [17]. Our proposed leakage-bin classification architecture is shown in Figure 6. The output of the cache array is linked to an adder which has the feedback of a clocked register. The register is clocked at a frequency bounded by the pulse width of the minimum difference between any 2 adjacent bins. The total number of bins to be used for classification can be decided at design-time. With minimum number of bins, those sub-arrays that have very different leakage profiles have every chance of being placed in the same bin. With reducing number of bins, the bounds of leakage within which a sub-array is placed into a bin is very loose. The bin selection procedure is initiated by writing a 1 to a 3T1D cell and signaling a read access and constantly holding the read-wordline high. The output of the sense-amplifier after a given period begins to decay. As long as the output of the sense amp is high enough to signal a 1, the adder increments the value of register by a 1 at the clock rate. The register is incremented at a predetermined frequency whose clock period is low enough to make sure adjacent bins exhibit a difference of at least 1 cycle as shown in Figure 7. It is clearly observable from Figure 7 that the cycle count increases with the bin number. This is in direct relation to the fact that reducing leakage along bin number corresponds to increasing retention times which is reflected by the increase in cycle count along the x-axis. It can be seen that the clock frequency used for 64-bin classification is 16X higher than the frequency used for 4-bin classification. This exhibits a linear relationship between the number of bins to be used and the frequency of the clock. Some researchers may argue that it is not a viable option to have a frequency divider for bin-classification purposes. The simplest solution to the problem would be to design for a frequency that would cater to the maximum number of bins, for instance 64. If the user intends to classify based on lesser number of bins, say 8, classification is performed by grouping into bins which are multiples of 8. This way in an 8-bin classification using 64 bins, bin 8 would represent 1 and bin 16 represents 2 and so on. This can be seen in Figure 7 where for a 4 bin classification, all the bins less than equal to 16 have 1 output cycle, all those between 16-32 have 2 output cycles and so on. This is made possible by using a modulo counter.

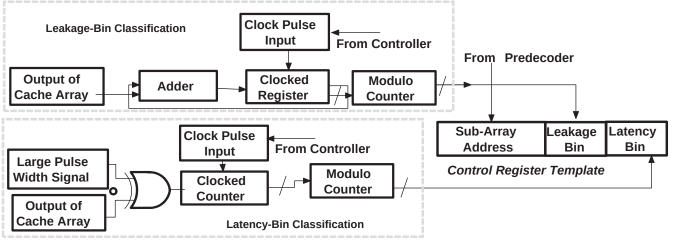


Fig. 6: Hardware Based Leakage/Latency Bin Classification based on Retention/Access Time

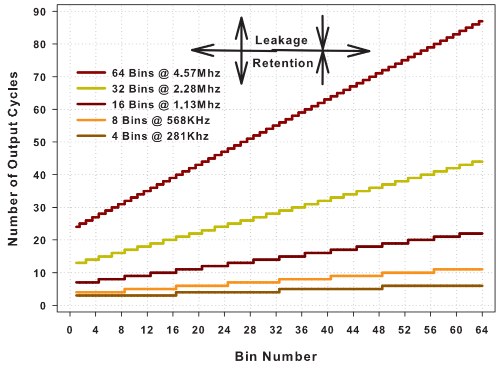
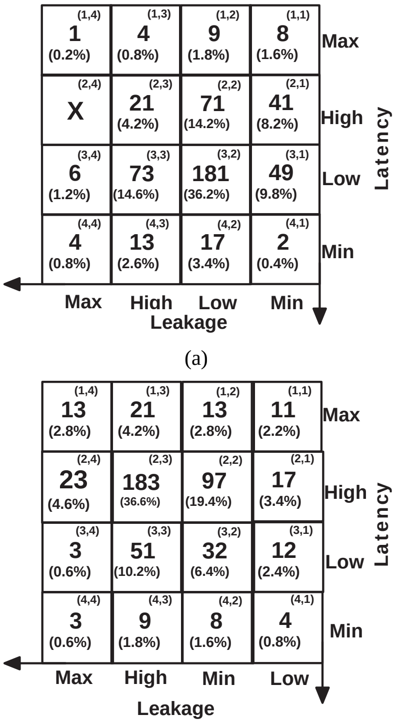


Fig. 7: Number of Output Cycles Vs Power Bin Number. (The number of output cycles (modulo) target number of bins) is the value that is written into the control register

In order to classify based on latency dependent on critical path delay, we determine the time to read access the 3T1D cell. Under the impact of spatial variability, for a set of 500 samples, the access times have been found to vary between 14-18% when maintained at a fixed ambient-temperature. This translates to a difference of about 400ps between the slowest and fastest arrays. In effect, the separation between adjacent bins can be as low as 6ps. As a result, for 64-bin classification, even multi-Ghz frequencies cannot produce a clock whose period is 6ps. Thus for multi-Mhz frequencies, the maximum target-number of bins is 4. The procedure to measure delay in terms of cycle count is similar to the one proposed in [17]. A signal with a very large pulse width is XOR'ed with the output of the cache array. The clocked counter starts incrementing on enabling the control signal to initiate reading a '1' from the 3T1D. As long as the output of the sense-amplifier is 0 and large-pulse width signal high, the counter is incremented for every cycle of the input clock. As soon as the output of the sense-amplifier reaches a high, the counting stops.



Latency \ Leakage	Max	High	Low	Min
Max	(1,4) 1 (0.2%)	(1,3) 4 (0.8%)	(1,2) 9 (1.8%)	(1,1) 8 (1.6%)
High	(2,4) X	(2,3) 21 (4.2%)	(2,2) 71 (14.2%)	(2,1) 41 (8.2%)
Low	(3,4) 6 (1.2%)	(3,3) 73 (14.6%)	(3,2) 181 (36.2%)	(3,1) 49 (9.8%)
Min	(4,4) 4 (0.8%)	(4,3) 13 (2.6%)	(4,2) 17 (3.4%)	(4,1) 2 (0.4%)

(a)

Latency \ Leakage	Max	High	Min	Low
Max	(1,4) 13 (2.8%)	(1,3) 21 (4.2%)	(1,2) 13 (2.8%)	(1,1) 11 (2.2%)
High	(2,4) 23 (4.6%)	(2,3) 183 (36.6%)	(2,2) 97 (19.4%)	(2,1) 17 (3.4%)
Low	(3,4) 3 (0.6%)	(3,3) 51 (10.2%)	(3,2) 32 (6.4%)	(3,1) 12 (2.4%)
Min	(4,4) 3 (0.6%)	(4,3) 9 (1.8%)	(4,2) 8 (1.6%)	(4,1) 4 (0.8%)

(b)

Fig. 8: (a) Runtime Classification of whole cache at ambient temperature. Numbers in each box indicate total & percentage number of caches in each bin. (Inset) (x,y) represents x output cycles obtained using leakage-bin classification hardware and y output cycles using latency-bin classification hardware. (b) Runtime Power/Performance binning of whole cache at $110\,^{\circ}\mathrm{C}$.

For a fixed supply voltage of 1V and 500 cache samples, the classification was performed for 4 binning levels of latency and leakage. A cache is placed into a respective bin after measuring the retention and access time of the 3T1D in each sub-array and considering the slowest sub-array. No caches have been placed in the high latency, low leakage bin as shown in Figure 8a. This phenomenon is characteristic to our single frequency grouped-levels binning methodology. As the 4-bins have been approximated by scaling the 64-bin classification, the bounds of each bin are loose resulting in misplacement of high latency, low leakage caches across the 3 immediate neighboring bins along its Cartesian co-ordinates. By re-running the simulations adjusting the input-pulse frequency specific to 4-bin classification, a considerable number of caches were categorized into the high latency, low leakage bin. Assuming we consider all caches that have latency and leakage greater than *high* as yield loss, hardly 50% of caches are accepted. Further the presented yield estimates in Figure 8a hold true only when the cache is operating at nominal temperatures. Common phenomena such as sudden temperature shootups can result in the performance going from high to low and in some cases to minimum. The problem is compounded by increasing leakage with temperature. It is clearly observable from Figure 8b that number of chips placed in the high-performance low-power bin at $30\,^{\circ}\mathrm{C}$ shifts diametrically to the high-power low-performance bin at $110\,^{\circ}\mathrm{C}$. This results in yield going from bad to worse. From the above results it is clear that, we have been successful in translating the logical relation presented in figure 5 to a hardware based methodology. In the next section, we will discuss as to how we can exploit this available measurements to improve the overall yield in terms latency/leakage profiles.

IV. Applying Fine Grain Body Biasing

It was shown in [18] that reduction in leakage power is possible by optimizing the 6T cell at design-time for high V_{th} and applying a large forward body bias at run-time to compensate for the increased latency. In other words, the array is purposefully designed for lower speed (lower leakage) and made to run faster during operation. This would mean that a large FBB is applied irrespective of whether the array meets the required timing or not. From a statistical standpoint, both latency/leakage can be on either side of target design value as a result of process variations. Hence no forward biasing is required for those arrays that already meet both leakage and latency targets. It is this very non-determinism that we would like to exploit in order to generate an optimal bias voltages dependent on the actual latency/leakage measured.

We propose to use a modified version of the lookup table-based adaptive forward body biasing mechanism[7]. A predecoder inside a cache receives the address of the sub-array to be accessed. As shown in figure 9, the latency bin of the sub-array is sent to the LUT by comparing the address received from the predecoder to the address field of all control registers. The latency bin is referenced inside the LUT to obtain a codeword that is sent to the FBB generator. This codeword corresponds to the lowest forward bias voltage that satisfies the latency constraint. The codewords in the figure are only indicative and do not represent the actual bias voltages. This lookup table is defined at design time. The FBB generator consists of four components - decoder, level shifter, demux & resistor tree. The resistor tree is used for generating the forward bias voltages. As it requires a voltage (VDDH) higher than VDD and lower (VDDL) than VSS, a level shifter is employed. The demux receives the address of the sub-array to be accessed and routes the generated forward bias to the correct sub-array. The resistor tree consists of a series of transistors connected together acting as a potential divider. The number of transistors divide the range (VDDH-VDDL) into intermediate voltages. In our case we assume a maximum range of 500mV. We have used 20 transistors in our design, and connected switches to the 4th,8th,12th and 16th transistors to generate intermediate voltages of 0.1,0.2,0.3 & 0.4V respectively. The decoders are used to select the correct combination of switches to generate the appropriate FBB.

Each of the N sub-arrays require 3 amplifiers (one each for FBB & RBB) to boost the body voltage to a level sufficient to bias the entire array and to enable sleep mode. A hybrid charge pump is used to generate negative bias for RB biasing inactive sub-arrays. The amplifiers for routing the RBB voltage are enabled based on the address of the sub-array to be accessed. The address of the to-be accessed sub-array is decoded and all the output lines of the decoder act as the enable signal for the RBB amplifiers. Only one of output lines is high (corresponding to the sub-array to be accessed) and the remaining are low. Thus by inverting these lines, the RBB amplifiers are enabled. It was shown in [18] that if a sub-array is accessed in a given cycle then it is likely to be accessed in the immediate next cycle and those that are idle are expected to remain idle for a considerable amount of time. This phenomena called temporal locality of reference, can be exploited to forward bias those sub-arrays that are currently being accessed and reverse bias those that are idle. This eliminates the need to regenerate the same FBB voltage on per-cycle basis by constantly referencing the lookup table. As a result, RBB generator needs to be aware of the inactive sub-arrays for a large number of cycles. Because it receives the address of the active array only once, the state of inactive arrays needs to be

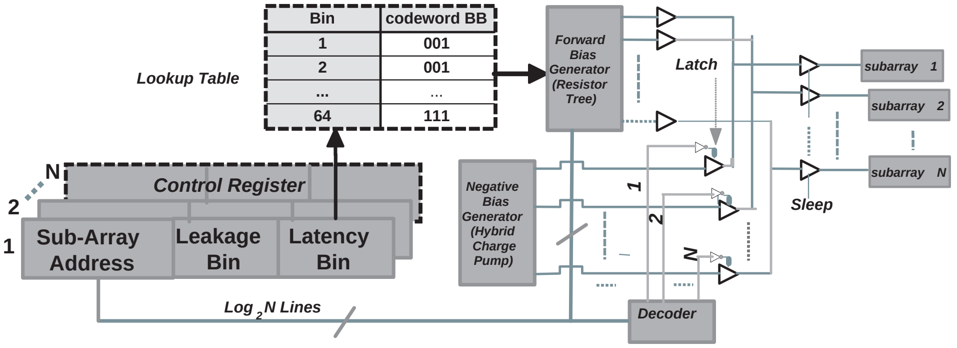
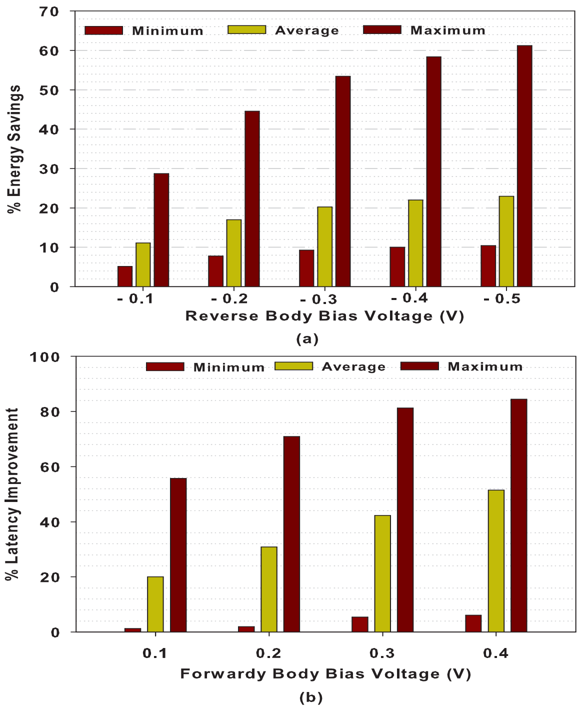


Fig. 9: Fine-Grain Body Bias Generator for Caches

stored. An extra latch is provided to store the state of the inactive array for enabling/disabling RBB mode. In addition to hiding the transition latency involved in switching from RBB to FBB and vice-versa in a very time-effective manner, the transition energy involved in switching between RBB and FBB is reduced significantly.

V. Experimental Results

In accordance with the analysis presented in [6], BB voltages range from a maximum RBB of -500mV to a maximum FBB of 400mV. On a per-access basis, only one sub-array is active and the remaining are inactive. This is the closest representation of the actual architectural state of the entire cache.



(a)

(b)

Fig. 10: (a) Percentage Energy Savings as a function of the reverse body voltage & (b) Percentage Latency Improvement as a function of forward body voltage. The bars represent savings when compared to ZBB

A. Leakage & Latency Reduction

Looking at figures 10(a) & (b), both leakage & latency are a very strong function of the bias voltages. Energy is calculated after accounting for the energy consumed by bias generators and the energy lost due to active mode forward biasing. The minimum and maximum values correspond to the lowest & highest improvements obtained for one sub-array among all sub-arrays among all caches (D2D). The average is the lowest of the arithmetic mean of the improvements obtained for all sub-arrays in a given cache (WID) among all caches (D2D). It can be seen that the minimum average savings in energy is 12% (-0.1V) and the maximum is 24% (-0.5V. For -0.3V it is 20% and the improvement in energy savings is minimal for voltages above. This is because process variations are known affect multiple transistor parameters (threshold, oxide thickness, effective channel length) which in-turn affect leakage and threshold voltage is the only parameter that can be dynamically altered with body biasing. By providing an adaptive RBB generator, the leakage bin field can be used to determine appropriate reverse bias voltages for further energy reduction. Looking at the results of latency improvements in figure 10(b), it can be seen that there is a large discrepancy between minimum and maximum values. This is because, we body bias only the SRAM array and the latency is calculated for the entire access path constituting predecoders, row decoder, column multiplexer, sense amplifiers, wordline and write drivers that are not body biased. Techniques like dual V_{dd} & dual V_{th} can then be employed to reduce the impact of process variations on periphery [19], [20]. Because our mechanism can alter the forward body voltage based on the measured latency, we can expect maximum latency reduction even under worst-case process variations.

B. Evaluating Yield

Heuristics for estimating parametric yield suggest that caches which fail to meet the latency constraint (maximum allowed access latency under process variations) can be considered an yield loss. In sub-90nm designs as leakage can play a very important role, it was shown that in addition to considering latency cutoff, caches that consume leakage power greater than 3μ should also be rejected [21].

Adopting the above heuristics, we determine the parametric yield for three different cases - No body biasing, only forward body biasing active sub-arrays with single voltage [18] & our proposal - fine-grain body biasing each sub-array. We assume that all inactive sub-arrays are reverse biased at -0.3V in our proposal. The yield is determined for multiple latency constraints and for one leakage constraint of 3μ. A cache is considered yield loss if more than 3 sub-arrays fail to meet the constraints. It can be seen from Figure **??** that under no body biasing (ZBB) the yield reduces from 82% to 60% for tighter latency constraints. The yield loss is only as a result of sub-arrays failing to meet latency constraints and not because of leakage constraints. The yield for forward biasing with one voltage is constant at 60% in all cases. While all sub-arrays clear the latency cutoff because of lowering the threshold, sub-arrays fail to meet the leakage cutoff resulting in yield loss. For all cases of latency constraint, the yield is 100% in our case. This is mainly because of 2 factors. Unlike adaptive body biasing where we decide to either use RBB or FBB, here we use both in a time shared manner. The selected forward bias voltage is the minimum voltage for which the latency cutoff is met as shown in Figure 12. This is to ensure that active leakage power is further reduced. For the case when latency constraint is μ+σ, 82% of caches do not need any bias. By providing a bias generator with just 1 voltage of 0.1V, the yield can be significantly improved to 95%. With further increase in the number of available bias voltages, the increase in yield is minimal. The yield is actually not a function

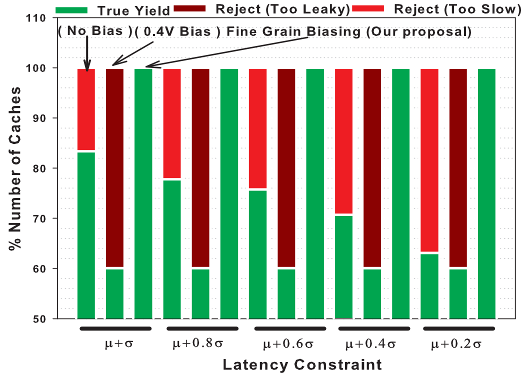


Fig. 11: Yield estimated for ZBB, Constant FBB and DFGBB as a function of Latency Constraints

of the number of bias voltages but a function of the minimum & maximum bias voltage that ensure all sub-arrays meet both latency & leakage targets. By increasing the number of available voltages (by reducing the steps), more fine-grain control can be achieved. For high performance designs, the latency constraints are around μ+0.2σ & μ+0.4σ and it is clearly evident that there are caches that require both 0.3V and 0.4V FB voltages.

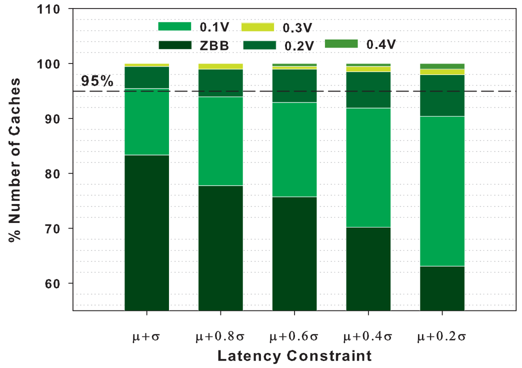


Fig. 12: Number and amount of FB Voltages required for 100% Yield

VI. Related Work

There have been several studies on latency/leakage measurement and post-silicon adaptivity independently. To the best of our knowledge this is the first work that offers a comprehensive solution by combining both techniques efficiently.

In [22], [23], [24] temperature sensors that exploit one or more of transistor characteristics dependent on temperature have been proposed. By measuring temperature as a variable parameter, response mechanisms can be enforced for energy-delay optimization. Because they require special interface hardware, it becomes impossible to integrate these sensors into caches. In [11], a 4T SRAM cell based temperature/leakage sensor has been envisaged. In terms of measuring delay as a variable parameter, it does not offer a straight forward solution.

In [25], Das *et al.* have proposed a novel framework for analyzing cache yield that is aware of both process variations and revenue. Also a new cache redundancy scheme called substitute cache that replicates data from cache lines affected by process variation has been proposed. The only priority for replicating cache words in the redundant cache is lines with very high latency. However, the technique does not assume leakage to have equal priority as this can change by orders of magnitude under the effects of process variations. Singh *et al.* have partitioned the SRAM array into blocks of different voltage groups to account for intra-array variations. While partitions that are very slow have very high voltage, the remaining have lower voltage levels for power saving. This technique characterizes the spread of spatial variability using empirical results that may/may not correlate with on-chip measurements. In [26], latency of failing wordlines is improved by boosting wordline voltage. Failing wordlines are tested during manufacturing and using an EEPROM, failure information is stored. For the lowest area overhead & boosting by 1V this technique enhances yield by 12% under worst-case process variations. However, this results in 37% increase in dynamic power.

VII. Conclusion

In this paper we propose a combination of latency/leakage monitoring and dynamically tunable fine-grain body-biasing techniques to maximize active and standby leakage reduction in caches. The latency/leakage are monitored using the proposed hardware that is interfaced with a 3T1D cell embedded into the 6T sub-array. By measuring the time to access & retention time of the 3T1D, it was shown that the sub-arrays can be classified based on run-time leakage/latency measurements. Then a lookup table based adaptive fine grain body biasing mechanism utilizes this measurement, to generate an optimal bias. While active sub-arrays are forward biased to improve performance, inactive sub-arrays are reverse bias to reduce leakage. The experimental results show that our technique on average improves access latency & reduces leakage energy by 18% & 23% respectively. The adaptability to temporal changes ensures cache performance & power consumption over the lifetime of the chip is constant.

References

[1] W. Liao *et al.*, "Microarchitecture level power and thermal simulation considering temperature dependent leakage model," in *ISLPED'03*.
[2] D. Brooks *et al.*, "Power, thermal, and reliability modeling in nanometer-scale microprocessors," in *IEEE Micro'07*.
[3] A. K. Singh *et al.*, "Mitigation of intra-array sram variability using adaptive voltage architecture," in *ICCAD'09*.
[4] M. Cho *et al.*, "Postsilicon adaptation for low-power sram under process variation," in *IEEE Design & Test'10*.
[5] J. Gregg *et al.*, "Post silicon power/performance optimization in the presence of process variations using individual well-adaptive body biasing," in *IEEE TVLSI07*.
[6] R. Teodorescu *et al.*, "Mitigating parameter variation with dynamic fine-grain body biasing," in *MICRO07*.
[7] B. Choi *et al.*, "Lookup table-based adaptive body biasing of multiple macros," in *ISQED07*.
[8] W. Luk *et al.*, "A 3-transistor dram cell with gated diode for enhanced speed and retention time," in *VLSI'06*.
[9] K. Lovin *et al.*, "Empirical performance models for 3t1d memories," in *ICCD'09*.
[10] L. Xiaoyao *et al.*, "Process variation tolerant 3t1d-based cache architectures," in *MICRO'07*.
[11] S. Kaxiras *et al.*, "4t-decay sensors: a new class of small, fast, robust, and low-power, temperature/leakage sensors," in *ISLPED '04*.
[12] Sarangi *et al.*, "VARIUS: A model of process variation and resulting timing errors for microarchitects." *TSM*, 2008.
[13] Agarwal *et al.*, "Statistical timing analysis for intra-die process variations with spatial correlations." *ICCAD*, 2003.
[14] "Predictive Technology Models, http://www.eas.asu.edu/ ptm."
[15] Liu *et al.*, "Leakage power reduction by dual-vth designs under probabilistic analysis of vth variation." *ISPLED*, 2004.
[16] S. Borkar *et al.*, "Parameter variations and impact on circuits and microarchitecture," in *DAC'03*.
[17] C. Poki *et al.*, "A time-to-digital-converter-based cmos smart temperature sensor."
[18] C. H. Kim *et al.*, "A forward body-biased low-leakage sram cache: device and architecture considerations," in *ISLPED03*.
[19] S. Ganapathy *et al.*, "Modest: a model for energy estimation under spatio-temporal variability," in *ISLPED '10*.
[20] H. Homayoun and A. Veidenbaum, "Reducing leakage power in peripheral circuits of l2 caches," in *ICCD'07*.
[21] S. Ozdemir *et al.*, "Yield-aware cache architectures," in *MICRO'06*.
[22] P. Ituero *et al.*, "Leakage-based on-chip thermal sensor for cmos technology," in *ISCAS'07*.
[23] S. Remarsu *et al.*, "On process variation tolerant low cost thermal sensor design in 32nm cmos technology," in *GLSVLSI '09*.
[24] Q. Chen *et al.*, "A cmos thermal sensor and its applications in temperature adaptive design," in *ISQED '06*.
[25] A. Das *et al.*, "Evaluating the effects of cache redundancy on profit," in *MICRO'08*.
[26] Y. Pan *et al.*, "Selective wordline voltage boosting for caches to manage yield under process variations."