A Low Cost Individual-Well Adaptive Body Bias (IWABB) Scheme for Leakage Power Reduction and Performance Enhancement In the Presence of Intra-Die Variations

Tom W. Chen
System VLSI Technology Organization
Hewlett Packard Co., Fort Collins, CO

Justin Gregg
Dept. of Electrical and Computer Engineering
Colorado State University, Fort Collins, CO

***Abstract*— This paper presents a new method of adapting body biasing on a chip during post-fabrication testing in order to mitigate the effects of process variations. Individual well biasing voltages can be changed to be connected either to a chip wide well bias or to a different bias voltage through a self-regulating mechanism, allowing biasing voltage adjustments on a per well basis. The scheme requires only one bias voltage distribution network, but allows for back biasing adjustments to more effectively mitigate die-to-die and within-die process variations. The biasing setting for each well is determined using a modified genetic algorithm. Our experimental results show that binning yields as low as 17% can be improved to greater than 90% after using the proposed IWABB method.**

I. Introduction

As modern CMOS technology is scaled down, the effects of die-to-die and within-die variations are becoming worse. Process variations can be categorized into four tiers, lot-to-lot variations, wafer-to-wafer variations, die-to-die variations, and within-die variations [1], [2]. For higher performance VLSI chips, die-to-die and within-die variations have a significant impact on their performance and power consumption [3], [4], [5]. Even though significant advances have been made to reduce process variations, silicon manufacturers have not been able to keep up with technology scaling. An existing statistical model [3], assuming a 3σ channel length deviation of 20% for the 50-nm technology generation, indicates that essentially a generation of performance gain can be lost due to systematic within-die fluctuations.

Small variations in spatial dimensions are becoming large relative to the critical dimensions in manufacturing processes. These large relative variations cause wide distributions of circuit operating frequencies and power dissipation. The distributions in frequency and power determine the percentage of circuits/chips which meet both a minimum frequency, f_t, and the power dissipation constraint, P_t. Given a fixed set of constraints, wider distributions make for lower binning yields after production.

Attempts have been made to adjust nfet and pfet body biases to affect the operating frequency and power consumption, thus, to improve product binning. The idea of adaptive body bias (ABB) was discussed in 1995 in [6] by Wann et. al. to reduce the transistor threshold voltage to retain the device performance. This technique was further explored by Kuroda [7] in the design of a DSP processor. Miyazaki [8], [9] extended the technique to a more complex microprocessor chip and proposed to control the substrate bias to adjust the delay of a circuit to improve yield. In [8], the substrate bias can be adjusted from -1.5V of reverse bias to 0.5V of forward bias to counter process variation as well as supply-voltage variation and operating-temperature variation. The results show that a 0.5V forward bias raises the maximum operating frequency of the processor by 10%. A recent work by Tschanz et. al. [12] described an adaptive biasing method which involved using an on-chip measuring circuit to determine the required back bias.

The results in [12] suggest that, while the simplest implementation of ABB was effective in mitigating the effects of die-to-die variation, its effect on within-die (WID) variation was limited. For this to be truly effective, V_{nb} needs to be adjusted separately for each section of the circuit which dictates using a triple-well process. The effectiveness of the method in [12] is further limited by the size of the sections used. Increasing the effectiveness requires adding another power grid section, along with a replica critical path, phase detector, counter, and R-2R ladder D/A converter. This proves to be enormously expensive in both die area and routing resources. Localized areas of high variations within a section were not addressed in [12].

We present an individual-well adaptive method of body bias control that mitigates the effects of die-to-die and WID process variations. We assume that an nwell process is the prevailing process available. Therefore, well biasing in this paper implies changing nwell bias voltage. However, the technique applies equally to pwell processes. Like the concept of sections in [12], we also assume that p-type transistors are grouped in certain way to form sections. The bodies of all the p-type transistors within a section are connected to a single nwell. We will address the way to group the transistors in later sections. However, unlike the concept of sections in [12], our section size can be small to provide fine-granular adjustments to the circuit without having any impact on area overhead. With only minimal additional circuitry and routing, individual well biases can be intelligently adjusted resulting in closely controlled chip power and performance. Our experimental results show that binning yields as low as 17% can be improved to greater than 90% after using the proposed IWABB method, far out-performing the method proposed in [12] under similar experimental conditions. In the remaining part of this paper, the general methodology of IWABB is presented in section 2, including a specific illustration of the method applied to

1530-1591/04 $20.00 (c) 2004 IEEE

an nwell process with pwell (substrate) biasing. Section 3 describes the experimental setup used to evaluate IWABB. Experimental results are summarized in section 4. Possible extensions to this work and conclusions are in the final two sections.

II. The IWABB method

In an nwell or triple-well CMOS process, nwells (pfet bodies) are normally connected directly to the power supply voltage, V_{dd}. Reducing the pfet body bias reduces their V_t, making the pfets switch faster and increasing their leakage current. Since pfets are inherently slower than nfets, their switching speed is usually one of the limiting factors in overall circuit performance. Increasing pfet speed can provide a significant speed up of the entire circuit. However, instead of using a separate power supply and power grid to control V_{pb} as previously experimented [12], one can use the capacitive coupling between drain and body of the pfets to provide V_{pb} for an entire nwell as illustrated in Fig. 1. By disconnecting the nwell from V_{dd} and allowing it to be regulated through the well-to-drain/source capacitance, the body voltage of all the pfets in the nwell will be determined collectively by their respective drain voltage. Assuming an nwell does not contain completely non-inverting logic[1], V_{pb} will always be maintained at somewhat below V_{dd}.

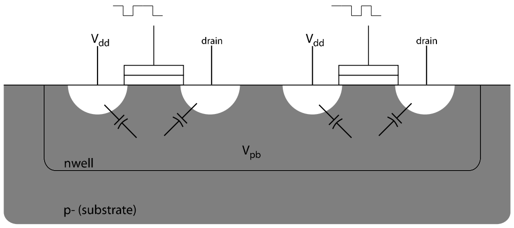

Fig. 1. Charge sharing in drain-body and source-body capacitances determines V_{pb} of the nwell.

The thought of letting the well voltages self-regulate by disconnecting them from any active bias voltage may sound scary at first. To better understand the impact of such a configuration, we need to understand the behavior of the floating bodies during switching events in terms of the performance gain achieved, added power consumption, and circuit reliability. Fig. 2 shows the amount of bounce the body voltage has in a simple inverter chain during a switching event. Fig. 3 shows the schematic of the inverter chain circuit. The bodies of both pfets in the circuit are connected to the same nwell. The distributed RC trees from the bodies to the V_{dd} contact in Fig. 3 model the parasitics within the well and between the well and the substrate. Furthermore, the bias voltage of the nwell, if left floating, is influenced by the relative size of the pfets that are always switching in the opposite direction. By varying the size of these two pfets, simulations of this circuit can help us understand

1) the amount of body bounce and its relationship with the relative sizes of transistors that switch in opposite direction, and
2) the impact of body bounce on overall gate delay.

Using a $0.1\mu m$ CMOS process and by propagating a switching event through the inverter chain while sweeping the width of the pfets in the inverters independently, we can look at their body bounce and the delay through the chain as a function of the individual inverter widths. This is equivalent to sweeping the number of transistors in the nwell that switch in each direction. It needs to be noted that sweeping the width of each of the inverters causes the well parasitics to be changed, and such changes are incorporated in the netlist during simulation. Fig. 4 shows the pfet body voltage bounce at the first inverter for both floated and normally biased cases when the well contact is at half of the maximum distance allowed by the process. To allow fair comparison, the pfet bodies were biased to the same voltage as the steady well biased voltage in the floated case. It is clear from Fig. 4 that there is an increase of about $40mV$ in body bounce when the nwell is floated as opposed to that of normally biased. This should not be of great concern in terms of reliability and latchup. Fig. 5 shows the percentage difference in delay between the floated and biased nwell. The delay of the floated well is slightly less across the sweep, though not significantly. On the left and bottom edges, the delay is improved by about 2.5% with the floating well. Furthermore, VLSI designs in 90nm and beyond require that the supply voltage to be close to 1V. With such a low supply voltage combined with small increase ($40mV$) in body bounce, the chances for latchup is very low. From this we can conclude that floating pfet bodies is not a concern to reliability compared to that of the normally forward biased wells. The floating well, however, doesn't require an additional power grid for the bias distribution. Any nwell can be controlled to be floated or connected to an active bias voltage. Such control requires only one scan-latch as shown in Fig. 3. Furthermore, to align the body bounce for floating wells better against that of the normally biased wells, one can group the transistors/gates into a single well in such a way that the ratio of the total transistor sizes for switching in one direction versus the other direction is balanced. In fact, Fig. 4 suggests that a ratio less than 2:1 or 3:1 is sufficient.

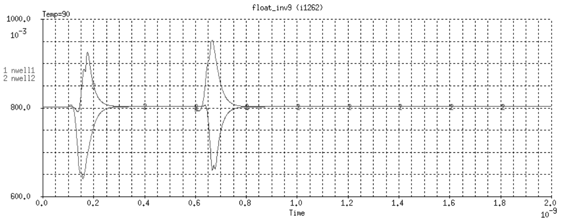

Fig. 2. Body voltage, V_{pb}, bounce in inverter chain shown in Fig. 3 with nwell biased to $0.8V$, $W_{p1} = 16\mu m$, $W_{p2} = 16\mu m$ and the nfets appropriately sized to match.

Since floating wells can only increase P_{op} (due to increased leakage current), changing the biasing of connected wells is

[1] i.e. a significant number of pfets in each nwell are conducting at anytime.

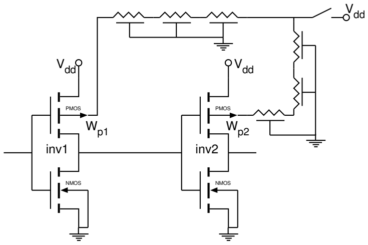

Fig. 3. Schematic of a test circuit to investigate body voltage bounce including parasitic well resistance and capacitance.

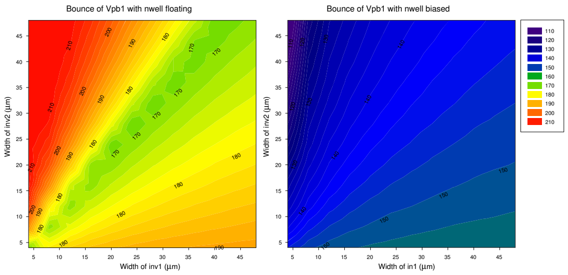

Fig. 4. Comparison of body voltage bounce in mV when nwell is floated and biased to $0.8V$.

needed in order to reduce overall power dissipation. This can be done in both a single- and triple-well process in three ways: the voltage of connected pwells (substrate in an nwell process) voltage can be lowered, the voltage of connected nwells can be increased, or both.

During circuit testing, the operating frequency, f_{op}, and power, P_{op}, measurements used to bin the chip can first be used to control well connections (i.e. floating or biasing). In order to improve the binning yield, one needs to move the chip into an acceptable region where $f_{op} \geq f_t$ and $P_{op} \leq P_{max}$, where f_t and P_t are target frequency and power. Of course, even for the relatively easy case of allowing floating nwells and pwell biasing the search space is enormous. For a circuit with n nwells, there are 2^n possible configurations of floating nwells. Combining this with the range of allowable pwell biases (based on a finite power supply resolution and range) makes an exhaustive search infeasible. However, determining which wells to float can be intelligently done with a genetic algorithm. Each well is assigned to a single bit in a binary chromosome, and the genetic algorithm searches for good combinations of floating and connected wells based on an objective function using f_{op}, P_{op}, f_t, and P_t.

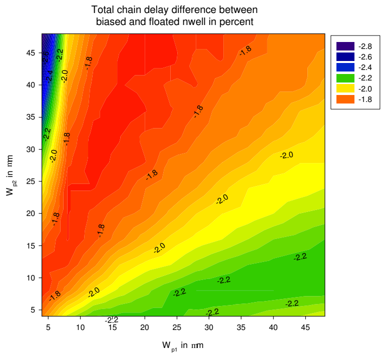

Fig. 5. Percentage delay difference in inverter chain when floated and biased to $0.8V$.

A. IWABB with nwell floating and substrate biasing

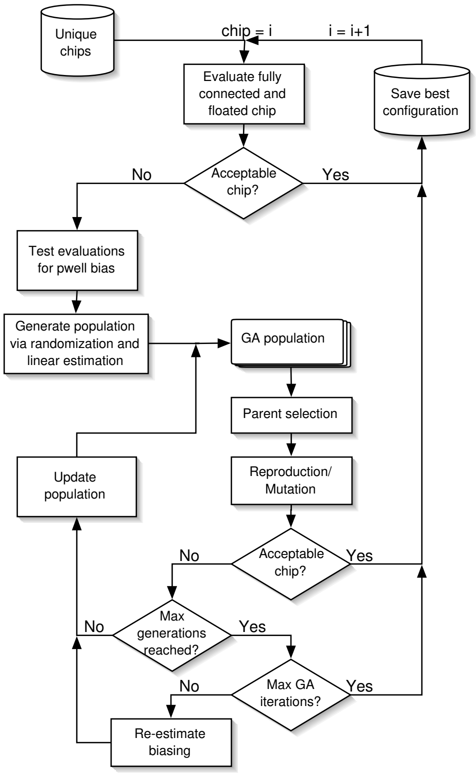

Fig. 6. Flow chart of the intelligent adaptive body biasing algorithm, IWABB.

Fig. 6 gives a graphical representation of the IWABB algorithm. Given a set of chips with process variations, IWABB optimizes each chip based on it's specific variations. If the given chip has n nwells, IWABB is run with an n-bit chromosome where each bit represents a single nwell. Initially, each chip is evaluated twice: once with all nwells connected to V_{dd} and once with all floated. If either of these configurations is acceptable, it is saved and the next chip is started. If an acceptable configuration is not found in the initial tests, an evaluation is run to determine the effectiveness of substrate biasing. Using these three evaluations, the $\Delta P_{op}/\Delta f_{op}$ slope can be determined for both nwell floating and substrate biasing. Using a simple linear estimation, the approximate number of floating nwells and approximate pwell bias can be determined. This step is not true to the nature of genetic algorithms, but can significantly reduce the number of evaluations needed. Based on the linear estimate of the number of floating nwells and the amount of substrate bias, a random population of chromosomes is generated and evaluated. A basic Genitor[11] style genetic algorithm is run with this initial population. Tournament selection is used to select two parent chromosomes from the population. These two parents beget one child chromosome via the reproduction function. The child's floating nwells are generated by favoring the more fit parent in a HUX-style crossover[11]. The child substrate bias is determined by the average of the parental substrate biasing. The child is then mutated both randomly and based on the average of the two parents. If the average P_{op} of the parents is greater than P_t, a decrease in substrate bias is favored. If the average f_{op} of the parents is less than f_t, the number of floating nwells is at least the average number of floating parental nwells. If the average f_{op} of the parents is greater than f_t, the number of floating nwells is at most the average number of floating parental nwells. This sort of directed mutation is not true to the nature of genetic algorithms, but helps improve the speed of convergence. The child is then evaluated. If it is not acceptable, and the maximum number of generations has not been reached, the population is updated by replacing the least fit chromosome with the child. The next generation of the genetic algorithm then starts. If the maximum number of generations have been completed, all of the substrate biases are updated based on a linear estimation. The genetic algorithm is then restarted. At the end of the genetic algorithm iterations, the best chromosome is recorded.

III. Experimental setup

A 32-bit static ripple adder was used as a test circuit. The circuit contained 896 transistors (448 of each pfet and nfet). P_{op} and f_{op} measurements were made by modeling the circuit in SPICE using a $0.10\mu m$ commercial nwell process with $V_{dd} = 1.1V$. Pfets contained in the same adder bitslice were grouped into a single nwell.

A. Modeling of manufacturing variability

In a real manufacturing process every dimension and parameter defined in the design process becomes a random variable. To simplify, only the variability of transistor gate length was modeled since it has the most pronounced effect on circuit operating properties. Due to the short-channel effect, variations on channel length will also cause the threshold voltage, V_t, to fluctuate. The overall distribution of transistor lengths upon variation should be a p-variate normal distribution $N_p(\mu = L, \Sigma)$ with a mean $\mu = L \in \Re^p$ and a covariance $\Sigma > 0 \in \Re^{p \times p}$, where p is the number of transistors and L is the transistor drawn length. Transistors in the same adder bit were set to have very highly correlated variations, while variations of widely separated transistors were not correlated. Successively neighboring adder bits had correlation coefficients of 0.85, 0.25, 0.10, 0.05 and 0.00. Transistors in bits further than three bits apart have only random correlation in their variation. These correlations were used to construct a 896×896 four-diagonal 28×28 block matrix representing the correlation between every pair of transistors as illustrated in Fig. 7. This correlation matrix ρ has an equal structure to the covariance matrix Σ since the variance of all the transistors were assumed to be equal. By partitioning the normal along this structure and applying a linear transformation to each part, this distribution can be rewritten as:

$$\rho = \begin{array}{|c|c|c|c|c|c|c} \hline A & B & C & D & E & 0 & \dots \\ \hline B & A & B & C & D & E & \ddots \\ \hline C & B & A & B & C & D & \ddots \\ \hline D & C & B & A & B & C & \ddots \\ \hline E & D & C & B & A & B & \ddots \\ \hline 0 & E & D & C & B & A & \ddots \\ \hline \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \end{array}$$

$$A = \begin{array}{|c|c|c} \hline 1.0 & 0.85 & \dots \\ \hline 0.85 & 1.0 & \ddots \\ \hline \vdots & \ddots & \ddots \end{array}$$

$$\{B, C, D, E, 0\} = \begin{array}{|c|c} \hline \{0.25, 0.10, 0.05, 0\} & \dots \\ \hline \vdots & \ddots \end{array}$$

Fig. 7. Correlation matrix used in Monte Carlo simulation.

$$L' = N_p(\mu, \Sigma) = L + N_p(\vec{0}, \rho)\sigma$$

where $\rho > 0 \in \Re^{p \times p}$ is the correlation matrix of $N_p(\mu, \Sigma)$, and σ is the standard deviation of all the transistors. The probability density function of this new distribution can be expressed as:

$$pdf(N_p(\tilde{0}, \rho)) = |2\pi^p \rho|^{-1/2} \exp\left\{-\frac{1}{2} x^\top \rho^{-1} x\right\}$$

where $x \in \Re^p$ [13]. The standard deviation σ used to generate the new lengths was 3.33nm. This is consistent with the 10% variation at 3σ used in [14]. Fig. 8 shows the pre-IWABB characteristics of one hundred chips generated by this simulation.

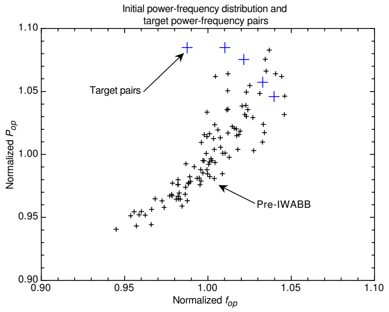

Fig. 8. Pre-IWABB power and frequency characteristics from SPICE of 100 chips generated by the Monte Carlo simulation shown with the target power-frequency pairs (f_t, P_t) used in IWABB testing.

B. Target dependence

IWABB's effectiveness is dependent on the target power-frequency pair. This dependence is due to the relation between speed and power dissipation when varying well floating and biasing. Because of this dependence, it becomes important to investigate the effectiveness of IWABB with different target points (f_t, P_t). Each target pair equates to a initial yield from the simulated manufacturing process. Comparing this initial yield to the yield after IWABB can give the designer an insight into how much improvement can be achieved. The red crosses in Fig. 8 shows the points used to test the effectiveness of IWABB relative to the initial power-frequency distribution of the test circuits.

IV. Experimental results

Initial experiments with IWABB using a simulated triple-well process did not show a significant advantage over using an nwell process. Due to space limitation, we will not give details of the triple well experiments in this paper. This and the high cost of implementing a triple-well process pushed our focus on to the standard nwell process. Three methods were tested:

Floating nwells with dual well biasing (NWF+DWB): This method involved adjusting the configuration of floating nwells and uniformly rebiasing the connected nwells and the substrate in order to meet f_t and P_t. This option provided the most flexibility to IWABB, but was also the most expensive. It would need two additional power supplies and one power grid in addition to V_{dd} and ground grids. It also represents the largest search space for IWABB due to the two discrete biases.

Floating nwells with nwell biasing (NWF+NWB): This method involved adjusting the configuration of the floating nwells and uniformly rebiasing the connected nwells. This method reduced the search space significantly compared to NWF+DWB and required one less power supply. It also reduced IWABB's options in controlling the operating frequency and power of the circuit since it only only addressed pfets.

Floating nwells with substrate biasing (NWF+PWB): This is the simplest IWABB method, involving adjusting the configuration of floating nwells and uniformly rebiasing the substrate. It only required one additional power supply.

Dual well biasing (DWB): Provided as a reference, this method is similar to that proposed in [12]. It requires a supply network for nwell bias and a power supply for the substrate bias.

Each method was tested with biasing resolutions of $20mV$ and $100mV$. The yield improvement results are shown in Fig. 9 and 10. NWF+DWB had the best increase in yield, being able to improve a yield as low as 17% to almost 90%. The larger search space of this method effected the results, especially in the $20mV$ bias resolution runs. At the lowest initial yields, NWF+DWB consistently ran into the time limit set on the algorithm. An example of the final power-frequency distribution from this method is shown in Fig. 11.

NWF+PWB was very effective, and more efficient than NWF+DWB due to the smaller search space. It's improved yields were comparable to NWF+DWB for all yields when the biasing resolution was $20mV$. With $100mV$ resolution, this method didn't have enough control over the forward bias on the nfets to increase their speed without violation the power constraint.

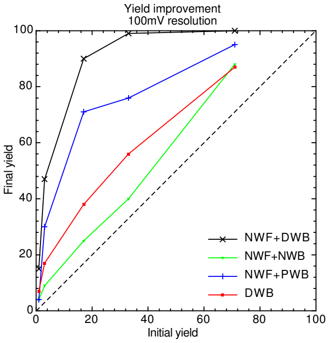

Fig. 9. IWABB yield improvement using floating nwells with dual well biasing, nwell biasing and pwell biasing and 100mV bias resolution.

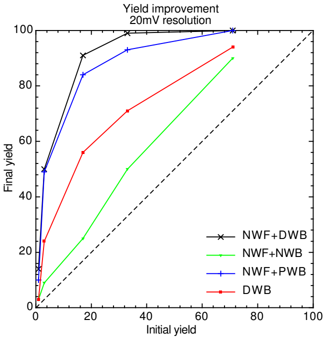

Fig. 10. IWABB yield improvement using floating nwells with dual well biasing, nwell biasing and pwell biasing and 20mV bias resolution.

NWF+NWB was not nearly as effective as NWF+DWB. It's improved yield was well below that for NWF+DWB for all target pairs. This is mainly caused by the method only having control over half of the circuits (pfets only). Slower circuits weren't able to be sped up beyond the speed of the nfets, and high power circuits couldn't be changed without a significant frequency reduction.

The reference method of DWB was reasonably effective. Its local gradient search is much more simple than any of the IWABB genetic algorithm search methods. At high initial yields, this algorithm's improved yields were comparable to IWABB, being able to improve a 33% yield to over 70%. However, at moderate and low yields, NWF+DWB and NWF+PWB far outperformed it's 17% to 56% improvement.

The different biasing methods produced require different biasing resolutions to be effective. NWF+DWB was equally effective regardless of the biasing resolution. Even though it requires two power supplies, the supplies can be simple. NWF+PWB requires sufficient biasing resolution to closely control the nfets. The $100mV$ resolution handicapped this method. In more difficult cases where the initial yield was lower than 10%, the smaller biasing resolution actually performed worse. This difference can be accounted for again by the limited search time and the larger search space.

Overall, the different methods all showed an increase in yield for all initial yields and target power-frequency pairs. NWF+DWB and NWF+PWB far outperformed DWB for all initial yields. Since search time in a real implementation could be increased dramatically,[2] the size of the search space is only of minor consideration. Considering all the costs and the relative effectiveness of the different methods, NWF+PWB with $20mV$ bias resolution was the best.

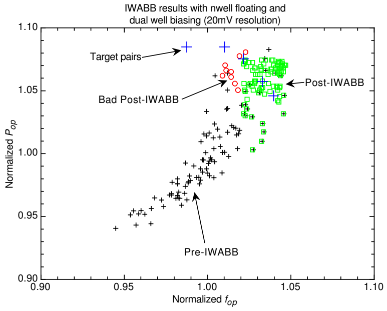

Fig. 11. Post-IWABB chip properties compared to pre-IWABB properties for nwell floating with 20mV resolution nwell and pwell (substrate) biasing.

V. Future work and extensions

- Given such a powerful tool for optimizing well biasing, there needs to be a way to group fets into wells during the layout phase of a chip in order to maximize the effectiveness of IWABB.
- In order to minimize the amount of body bounce in floating wells, an intelligent transistor grouping algorithm is needed that balances the transistor switching polarity in a well and also takes layout constraints into account.
- Balancing switching polarity within a well is difficult. If a large number of pfets in a floated well switch in the same direction simultaneously, the droop in well voltage can be significant. Theoretically, this bounce can act as a pre-charge for the switching event. The effects of this bounce needs to be further quantified.

VI. Conclusions

The variations introduced during the chip manufacturing process have an enormous impact on the characteristics of the chips produced and, therefore, the yield of the production line. We have introduced a low-cost adaptive body bias solution and an algorithm to intelligently adjust body biasing of transistors on a chip in order to mitigate the effects of die-to-die as well as within-die process variations to meet frequency and power targets. We have shown IWABB using nwell floating and various biasing techniques to be very effective in improving binning yields. Initial binning yields as low as 17% can be improved to greater than 90%. Considering implementation cost and effectiveness, nwell floating with $20mV$ resolution pwell biasing was the best. Although the test circuit is simple, we expect the algorithm would perform similarly on larger circuits since there is no circuit specific limitations in the algorithm or unique characteristics in our test circuit.

References

[1] D. Boning and J. Chung, "Statistical Metrology: Understanding Spatial Variation in Semiconductor Manufacturing", *Proc. of the SPIE*, pp.16–26, Dec. 1996.

[2] D. Boning and J. Chung, "Statistical Metrology: Measurement and Modeling of Variation for Advanced Process Development and Design Rule Generation", *Proc. of Int. Conf. On Characterization of Metrology for ULSI Technology*, pp.395–404, March, 1998.

[3] K.A. Bowman, S.G. Duvall, and J.D. Meindl, "Impact of Die-to-Die and Within-Die Parameter Fluctuations on the Maximum Clock Frequency Distribution", *2001 IEEE ISSCC*, pp.278–279, Feb. 2001.

[4] W.C. Riodan, R. Miller, J.M. Sherman, and J. Hicks, "Microprocessor Reliability Performance as a Function of Die Location for a 0.25um Five Layer Metal CMOS Logic Process", *1999 IEEE Int. Reliability Physics Symp.*, pp.1–11, March, 1999.

[5] S. Nassif,"Delay Variability: Sources, Impacts, and Trends", *2000 IEEE ISSCC, Digest of Technical Papers*, pp.368–369, Feb. 2000.

[6] H.C. Wann et. al., "Channel Doping Engineering of MOSFET with Adaptable Threshold Voltage Using Body Effect for Low Voltage and Low Power Applications", *1995 Int. Symp. on VLSI Tech., Systems, and Applications*, pp.159–163, 1995.

[7] T. Kuroda, et. al., "A 0.9V 150MHz 10-mW 2-D Discrete Cosine Transform Core Processor With Variable Threshold-Voltage Scheme", *1996 ISSCC Digest of Technical Papers*, pp.166–167, 1996.

[8] M. Miyazaki, G. Ono, and K.A. Ishibashi, "1.2-GIPS/W Microprocessor Using Speed-Adaptive Threshold-Voltage CMOS With Forward Bias", *IEEE JSSC*, Vol.37, No.2, pp.210–217, Feb. 2002.

[9] M. Miyazaki, H. Mizuno, and K. Ishibashi, "A Delay Distribution Squeezing Scheme With Speed-Adaptive Threshold-Voltage CMOS (SA-Vt CMOS) for Low Voltage LSls", *Proc. of 1998 ISLPED*, pp.48–53, Aug., 1998.

[10] Kerry Bernstein and Norman J. Rohrer, *SOI Circuit Design Concepts*, Kluwer Academic Publishers, 2002.

[11] David Goldberg *Genetic Algorithms*, Addison-Wesley, 1989.

[12] James Tschanz, et al., *Adaptive body bias for reducing impacts of die-to-die and within-die parameter variations on microprocessor frequency and leakage*, IEEE ISSCC 25.7, Feb 2002.

[13] J.D. Jobson, *Applied Multivariate Data Analysis*, Springer-Verlag, 1992.

[14] Ashish Srivastava, et al., *Modeling and analysis of leakage power considering within-die process variations*, ISLPED, 2002.

[2] Since power and frequency measurements would be nearly instantaneous, as opposed to SPICE simulation time.