Subthreshold Leakage Control Techniques for Low Power Digital Circuits

by

James T. Kao

B.S. in Electrical Engineering and Computer Science, University of California at Berkeley (1993)

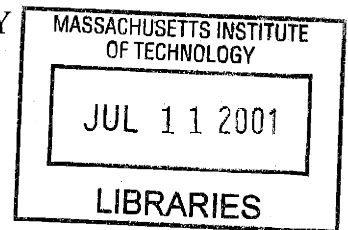
S.M. in Electrical Engineering and Computer Science, Massachusetts Institute of Technology (1995)

Submitted to the Department of Electrical Engineering and Computer Science in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

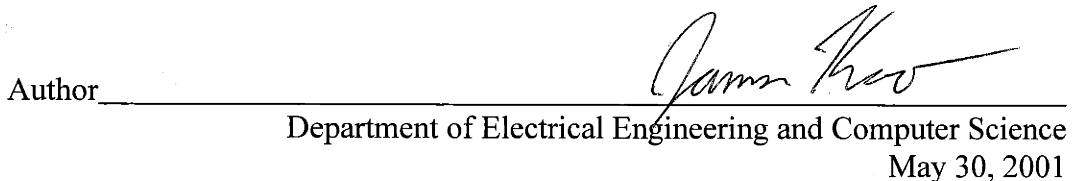


May 2001

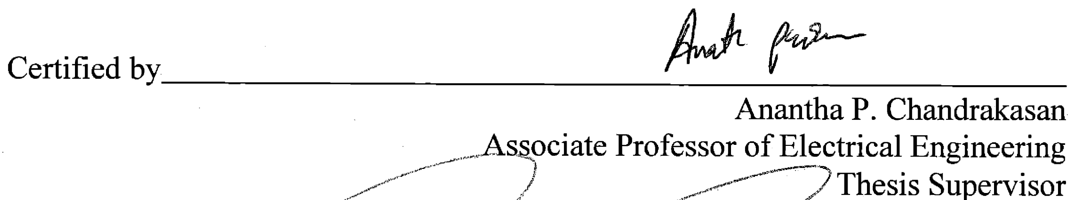
[June 2001]

© Massachusetts Institute of Technology, 2001. All rights reserved.

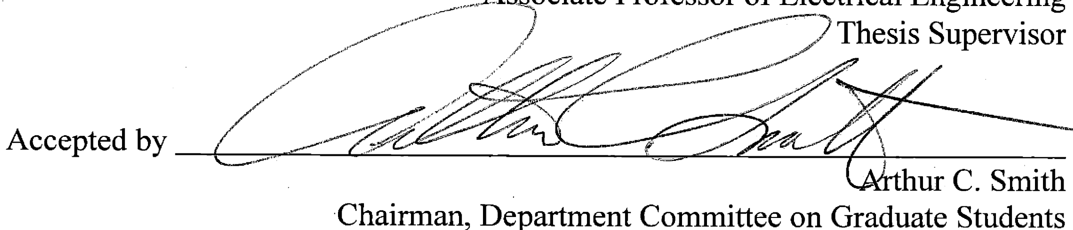
ARCHIVES



Author ______________________________
Department of Electrical Engineering and Computer Science
May 30, 2001



Certified by ______________________________
Anantha P. Chandrakasan
Associate Professor of Electrical Engineering
Thesis Supervisor



Accepted by ______________________________
Arthur C. Smith
Chairman, Department Committee on Graduate Students

Subthreshold Leakage Control Techniques for Low Power Digital Circuits

by

James T. Kao

Submitted to the Department of Electrical Engineering and Computer Science
on May 30, 2001, in partial fulfillment of the requirements for the degree of
Doctor of Philosophy in Electrical Engineering and Computer Science

Abstract

Scaling and power reduction trends in future technologies will cause subthreshold leakage currents to become an increasingly large component of total power dissipation. As a result, new techniques are needed in order to provide high performance and low power circuit operation. This dissertation develops new circuit techniques that exploit dual threshold voltages and body biasing in order to reduce subthreshold leakage currents in both standby and active modes. To address standby leakage currents, a novel sleep transistor sizing methodology for MTCMOS circuits was developed and new "imbedded" dual V_t techniques were described that could provide better performance and less area overhead by exploiting different logic styles. Work was also done to develop new MTCMOS sequential circuits, which include a completely novel way to hold state during standby modes. Body biasing circuit techniques were also explored to provide dynamic tuning of device threshold voltages to tune out parameter and temperature variations during the active state. This not only helps reduce active leakage currents but also improves process yields as well. A final research direction explored optimal V_{CC}/V_t tuning during the active modes as a function of varying workloads and temperatures so that a chip can automatically be configured to operate at the lowest energy level that balances subthreshold leakage power and dynamic switching power. Through novel circuit techniques and methodologies, this work illustrates how subthreshold leakage currents can be controlled from a circuit perspective, thereby helping to enable continued aggressive scaling of semiconductor technologies.

Thesis Supervisor: Anantha P. Chandrakasan
Title: Associate Professor of Electrical Engineering

Acknowledgments

First of all, I would to thank my advisor, Prof. Anantha Chandraksan, for his support over the years. His expertise and thorough, yet broad, grasp of the entire low power electronics field, from circuits to systems and algorithms, have been vital to my education during this period. He has kept me on track and helped direct this research into areas where we could make exciting and useful contributions to the field. I would also like to thank my other thesis committee members, Prof. Antoniadis and Prof. Troxel for their efforts and instruction, and especially for the fact that they were willing to schedule the thesis defense on very short notice so that I would be able to attend commencement. I would also like to thank Masayuki Miyazaki, visiting scientist from Hitachi, for his assistance with the development of the variable V_{CC}/V_T DSP testchip. In particular, Masa designed the board and spent many, many nights collecting data while I was toiling away with my thesis. I would also like to thank Shekhar Borkar, Vivek De, Siva Narendra, and Raj Nair from the Circuits Research Lab at Intel for providing their time and resources for designing and fabricating the adaptive body bias test chip. Finally, I would like to thank my fellow graduate students, too many to name, both in Anantha's group as well as in MTL as a whole. Interactions with other students play a vital role in the Ph.D. experience, and I have enjoyed being able to learn from the senior students when I first started out at MIT, and being able to pass on what I have learned to the newer generation.

I'd also like to thank my family for their unwavering support over the years. I have always been able to rely on them, and that one constant always gave me comfort during those long years that I have worked on this dissertation. In particular though, I'd like to thank my wife for her incredible devotion, understanding, patience, and support over the past 3 years that we have been married. I met Hideko while I was a graduate student, and she still agreed to marry me even knowing that we would be poor students for quite some time. She never complained when my research led us to Texas for a summer and Oregon for over a year, and she remained supportive even when I'd sometimes take my stress out on her unfairly. Throughout the course of this research, and the culmination of

this thesis, Hideko has helped keep me grounded and is a constant reminder of what is really important in my life.

Table of Contents

Chapter 1 Background 19
1.1 Sources of Power in Digital CMOS Circuits 19
1.2 Technology Scaling Impact on Subthreshold Leakage 21
1.3 V_{CC}-V_t Scaling Impact on Subthreshold Leakage Current 22
1.4 Total Power Reduction Philosophy 25
1.5 Burst Mode Circuits 27
1.5.1 Cell phone lifetime example 27
1.6 Source Biasing for Subthreshold Leakage Reduction 29
1.6.1 Switched-source-impedance leakage reduction technique 31
1.6.2 Self reverse biasing subthreshold leakage control technique 33
1.6.3 Stack effect for subthreshold leakage reduction 35
1.7 Dual V_t Methods for Subthreshold Leakage Control 35
1.7.1 Dual V_t Gate Partitioning 36
1.7.2 MTCMOS Technology 37
1.8 Body Biasing Techniques 37
1.9 Thesis Direction and Contributions 38

Chapter 2 Multi-Threshold Voltage CMOS Technology 41
2.1 MTCMOS Technology Overview 41
2.2 MTCMOS Sizing Impact On Performance 44
2.3 Inverter Tree Example Illustrating MTCMOS Delay 47
2.4 Vector Dependency on MTCMOS Sizing 49
2.4.1 8 bit carry save multiplier 50
2.5 MTCMOS Sleep Transistor 2nd Order Effects 52
2.6 MTCMOS Transistor Sizing Tools 54
2.7 Variable Breakpoint Switch Level Simulator 54
2.7.1 Variable breakpoint simulator for inverter tree 58
2.7.2 Variable breakpoint simulator for 3 bit adder 60
2.7.3 Simulator accuracy 61
2.8 Hierarchical Sizing Strategy Based on Mutual Exclusive Discharge Patterns 62
2.9 Example of Hierarchical Sizing 64
2.9.1 Sleep transistor merging step 66
2.9.2 Sleep transistor consolidation through parallel combination 67
2.10 Comparison with Optimal Sleep Transistor Size 69

2.11 Sleep Transistor Sizing Algorithm 69
2.12 Hierarchical Sizing Methodology 71
2.12.1 Parity checker example 72
2.12.2 Wallace tree multiplier example 73

Chapter 3 Dual V_t Domino Logic 77
3.1 Embedded Dual V_t 77
3.2 Dual Threshold Voltage Domino Special Case 80
3.2.1 Evaluate mode 81
3.2.2 Precharge mode 82
3.2.3 Standby mode 82
3.2.4 Clock delayed domino timing for single threshold voltage 83
3.2.5 Clock delayed domino timing for dual threshold voltages 85
3.3 Pipeline Standby Mode 87
3.4 Simulation Results 88
3.5 Dual V_t Domino Logic Issues 91

Chapter 4 MTCMOS Sequential Circuits 93
4.1 Previous MTCMOS Sequential Circuits 94
4.2 Improved MTCMOS Latch 98
4.2.1 Active state operation for the MTCMOS latch 99
4.2.2 Standby state 100
4.3 Sneak Paths in MTCMOS Sequential Circuits 101
4.3.1 Sneak leakage paths due to CMOS-MTCMOS parallel combinations 102
4.3.2 Sneak leakage paths due to CMOS-MTCMOS connection through low V_t pass-gates 103
4.3.3 Sneak leakage paths due to CMOS-MTCMOS reverse conduction paths 104
4.3.4 Techniques to eliminate sneak leakage paths 106
4.3.5 Short circuit currents due to MTCMOS-CMOS interfaces 108
4.4 Conventional MTCMOS Static Flip Flop 108
4.5 MTCMOS Flip Flop With CMOS Compatible Outputs 110
4.5.1 MTCMOS flip flop that stalls when clock is low 111
4.6 Impact of Parallel High V_t Circuits 112
4.7 MTCMOS Flip Flop Without Parallel High V_t Devices 113
4.8 Leakage Feedback Gates 114
4.8.1 Leakage feedback gate with floating inputs 117
4.8.2 Complex leakage feedback gates 123
4.8.3 Leakage feedback interface circuitry 124
4.9 Leakage Feedback Static Flip Flop 126
4.9.1 Active operation 127
4.9.2 Standby condition 129
4.9.3 Comparison of MTCMOS leakage feedback flip flop to conventional one 131
4.10 MTCMOS Dynamic Flip Flop with Standby Data Retention 132
4.10.1 Active operation 134
4.10.2 Standby operation 135
4.10.3 Exiting standby mode 136
4.11 MTCMOS Flip Flop Simulation Comparison 137

Chapter 5 Variable V_t Techniques - Body Biasing 141
5.1 Body Biasing Theory 142
5.2 Control of parameter variations 144

5.3 Adaptive Body Biasing ... 147
5.3.1 Implicit parameter variation tuning ... 148
5.3.2 2 dimensional matching considerations ... 149
5.3.3 Adaptive body biasing vs. DVS for compensating parameter variations ... 151
5.4 Within die adaptive body biasing ... 153
5.4.1 ABB for compensating intra die variations ... 154
5.4.2 Intra die variations and local matching criteria ... 155
5.5 Qualitative Benefit of Within Die Adaptive Body Biasing ... 157
5.5.1 Impact on yield and performance ... 159
5.5.2 Critical path systematic variations ... 160
5.5.3 Compensating for systematic variations ... 163
5.6 Forward and Reverse Body Biasing ... 164
5.7 Adaptive Body Biasing Test Chip ... 165
5.7.1 Adaptive body biasing generator implementation ... 166
5.7.2 Test chip simplifications ... 169
5.7.3 Phase detector implementation ... 169
5.7.4 Flip flop based phase detector performance ... 171
5.7.5 Alternative phase detector implementations ... 172
5.7.6 D/A Converter for PMOS Body Bias ... 175
5.7.7 NMOS Body Bias generator ... 179
5.7.8 Analog adaptive body biasing control alternatives ... 180
5.8 Test Chip Simulations ... 184
5.8.1 Limitations of the Modified Monte Carlo Simulations ... 187
5.8.2 Simulation results ... 190
5.9 Adaptive Body Biasing Effectiveness ... 206

Chapter 6 Optimal V_{CC} - V_t Circuit Operation ... 209
6.1 V_{CC}- V_t optimization overview ... 210
6.2 Theoretical V_{CC}-V_t Optimum ... 211
6.3 Optimal V_{CC}/V_t Scaling Trends Based on Theoretical Models ... 216
6.3.1 Role of switched capacitance on optimal V_{CC}-V_t scaling ... 217
6.3.2 Role of I_0 on optimal V_{CC}-V_t Scaling ... 218
6.3.3 Optimal V_{CC}-V_t positioning ... 219
6.3.4 Role of frequency target on optimal V_{CC}-V_t scaling ... 221
6.3.5 Comparison to dynamic voltage scaling ... 224
6.4 Triple Well Test Chip ... 228
6.4.1 Threshold voltage tuning limitations ... 230
6.4.2 Test chip block diagram ... 231
6.4.3 V_{CC}/V_t biasing procedure ... 233
6.4.4 LFSR for automatic vector generation ... 234
6.4.5 Critical path replica ... 235
6.5 Test Chip Simulations ... 237
6.5.1 Isoperformance V_{CC}-ΔV_{BB} simulation locus ... 237
6.5.2 Power vs. V_{CC} simulations for different frequencies ... 240
6.5.3 Process corner variation impact on minimum power point ... 242
6.5.4 Temperature impact on minimum power point ... 242
6.5.5 Optimal V_{CC}/V_t scaling comparison with DVS ... 244
6.6 Test Chip Measurements ... 245
6.6.1 Measured optimal V_{CC} / ΔV_{BB} operating points ... 250
6.7 Alternative Threshold Tuning Techniques ... 254

6.8 Automatic V_{CC}-V_t biasing 255
6.8.1 Open loop approach 256
6.8.2 Closed loop approach 257
6.8.3 Dual loop control 260
6.8.4 Degenerate optimal operating points 263
6.8.5 Test circuit for power estimation 264

Chapter 7 Conclusions 269

Appendix A Optimal V_{CC} / V_t Test Chip Operation 279
A.1 High Level Block Diagram 279
A.2 Chip Pinout 284
1.3 Board layout [Miyazaki] 285
A.4 Circuit Schematics 286

List of Figures

FIGURE 1-1. Constant delay curves for V_{CC}- V_t span. 24
FIGURE 1-2. Minimum Energy point as function of V_{CC} & V_t. 24
FIGURE 1-3. Graphical representation showing reduction of total active and standby leakage power. 26
FIGURE 1-4. Standard log I_{ds} - V_{gs} curve showing subthreshold slope and off leakage current values 29
FIGURE 1-5. Source biasing principle. 30
FIGURE 1-6. Source biasing example through switched source impedance technique. .31
FIGURE 1-7. Switched source impedance techniques for different circuit off scenarios 32
FIGURE 1-8. Self reverse biasing technique to reduce subthreshold leakage currents. ..33
FIGURE 1-9. Self reverse biasing technique applied to DRAM wordline drivers. 34
FIGURE 1-10. Dual V_t gate partitioning showing how critical paths can change. 37
FIGURE 2-1. MTCMOS circuit structure showing both polarity sleep devices. 42
FIGURE 2-2. MTSMOC block with purely combinational logic block can use a single polarity sleep device. 43
FIGURE 2-3. Sleep transistor modeled as resistor 45
FIGURE 2-4. Circuit model for MTCMOS delay 46
FIGURE 2-5. MTCMOS inverter tree. 47
FIGURE 2-6. Transient Response for 0->1 transition 48
FIGURE 2-7. Transient Response for 1->0 transition 49
FIGURE 2-8. Carry save multiplier diagram (4x4 bit shown). 50
FIGURE 2-9. 8x8 bit multiplier delay vs. W/L (shown as percentage of total NMOS pulldown W/L) for different input vectors (SPICE). 51
FIGURE 2-10. Reverse Conduction Paths 53
FIGURE 2-11. Variable break point switch level simulator function. 57
FIGURE 2-12. MTCMOS inverter tree used to compare variable breakpoint simulator with spice simulations. 58
FIGURE 2-13. Delay vs. W/L ratio for 0->1 and 1->0 inputs. 59

FIGURE 2-14. Virtual ground transient for 0->1 input for simulator and spice simulations. 60
FIGURE 2-15. 3 Bit adder to test variable breakpoint switching simulator......................60
FIGURE 2-16. 3 Bit adder spice versus variable breakpoint simulator comparison for two sample vectors.61
FIGURE 2-17. Different MTCMOS gate degradation scenarios that still satisfy overall delay value 63
FIGURE 2-18. Inverter chain example showing the 3 steps for merging sleep resistors. Simulation parameters: V_{CC}=1.0v, V_t=0.2v, C=50fF, lmin=0.7µm...........65
FIGURE 2-19. Circuit showing how sleep resistors can be combined in parallel. 67
FIGURE 2-20. Logic gates annotated with all possible transition times, so that sleep resistors can be merged. 70
FIGURE 2-21. 8 bit parity checker 72
FIGURE 2-22. 6x6 Wallace Multiplier..74
FIGURE 3-1. Embedded Dual V_t NOR gates with LVT devices shaded. Inputs shown to strongly turn off HVT devices for low standby leakage operation............78
FIGURE 3-2. High V_t device must be upsized by a factor f to equate current drives. 79
FIGURE 3-3. Dual V_t domino logic gate with low V_t devices shaded (with clock delayed methodology). 81
FIGURE 3-4. Dual V_t Domino gate in low leakage state. 83
FIGURE 3-5. Pipe stage showing clock delayed domino logic functionality. 84
FIGURE 3-6. Clocking methodology showing evaluate and precharge times.................84
FIGURE 3-7. Dual V_t pipe stage showing clock delayed domino logic functionality.....85
FIGURE 3-8. Clocking methodology showing evaluate and precharge times.................86
FIGURE 3-9. Clocking methodology showing evaluate and precharge times for non 50% duty cycle clocks..87
FIGURE 3-10. Pipeline sleep mode circuitry...88
FIGURE 3-11. Evaluation delay through pipeline stage..89
FIGURE 3-12. Precharge delay for pipeline stage..89
FIGURE 3-13. Leakage current for CLK=0...90
FIGURE 3-14. Leakage current for CLK=1...91
FIGURE 4-1. Conventional MTCMOS latch..95
FIGURE 4-2. MTCMOS balloon circuit schematic (A) and control signals (B).............96
FIGURE 4-3. MTCMOS balloon circuit schematic applied to master slave D flip flop. 97
FIGURE 4-4. MTCMOS latch that holds state during sleep...99
FIGURE 4-5. Latch during standby mode with data retention activated. 101
FIGURE 4-6. Leakage paths when only one polarity sleep transistor is used. 102
FIGURE 4-7. Leakage paths when only one polarity sleep transistor is used. 103
FIGURE 4-8. Leakage paths during standby state when only one polarity sleep transistor is used..104
FIGURE 4-9. Leakage paths during standby due to reverse conduction sneak path 105

FIGURE 4-10. MTCMOS latch with reduced sneak leakage paths. 107
FIGURE 4-11. MTCMOS Flip Flop with no sneak paths. 109
FIGURE 4-12. MTCMOS Flip Flop that has non floating output during standby state and sleep when clk high. 111
FIGURE 4-13. MTCMOS Flip Flop that can drive an output during sleep state, and sleeps when clk is low. 112
FIGURE 4-14. Inverter chain showing insertion of possible high V_t parallel device. 112
FIGURE 4-15. MTCMOS flip flop without parallel high V_t paths. 113
FIGURE 4-16. Low leakage but with driven outputs. 115
FIGURE 4-17. Leakage feedback gate principle. 116
FIGURE 4-18. Leakage Feedback output retains state regardless of changes to input. 118
FIGURE 4-19. Output state held by leakage currents. 119
FIGURE 4-20. Intersection showing DC operating point at logic "0". 120
FIGURE 4-21. Intersection showing DC operating point at logic "1". 121
FIGURE 4-22. Leakage feedback transient holding 0. 122
FIGURE 4-23. Leakage feedback transient holding 1. 123
FIGURE 4-24. Leakage Feedback gate as interface between MTCMOS and CMOS blocks (sharing CMOS output signal) 124
FIGURE 4-25. MTCMOS leakage feedback flip flop that has non floating output during standby state and sleeps when CLK is high. 127
FIGURE 4-26. MTCMOS Leakage Feedback flip flop in the standby mode 129
FIGURE 4-27. Sleep Mode timing diagram MTCMOS leakage feedback flip flop. 129
FIGURE 4-28. Basic MTCMOS dynamic flip flop. 133
FIGURE 4-29. MTCMOS leakage feedback dynamic flip flop. 134
FIGURE 4-30. Sleep/ active mode timing diagram for dynamic leakage feedback FF. 136
FIGURE 4-31. Flip flop delays (Tsetup + Tcq) for a)MTCMOS static FF, b)leakage feedback static FF, and c) leakage feedback dynamic FF. 138
FIGURE 4-32. Leakage current reduction during sleep modes for a)MTCMOS static FF, b)leakage feedback static FF, and c) leakage feedback dynamic FF. 139
FIGURE 5-1. Triple well technology 142
FIGURE 5-2. Body biasing to tighten distributions. 146
FIGURE 5-3. ABB block diagram. 147
FIGURE 5-4. Percent deviation in saturation currents as fuction of PMOS and NMOS body bias. 150
FIGURE 5-5. Within die body biasing scheme tightens overall distribution 155
FIGURE 5-6. Statistics of each local critical path of each island modeled with normal distribution 158
FIGURE 5-7. Statistics for local critical path frequencies with different distributions and systematic offsets. 162
FIGURE 5-8. Forward and reverse body bias ranges. 165
FIGURE 5-9. Adaptive body biasing test chip (multi-project chip) consisting of distinct ABB clusters comprised of multiple ABB generators. 166

FIGURE 5-10. Test chip architecture for adaptive body biasing generator 167
FIGURE 5-11. Flip flop phase detector operation 170
FIGURE 5-12. Alternative phase detector with up, down, and hold outputs 172
FIGURE 5-13. Conventional phase detector. 174
FIGURE 5-14. Converting a conventional phase detector to digital outputs levels....................174
FIGURE 5-15. Simple R-2R D/A converter. 177
FIGURE 5-16. Steering switches used for R-2R D/A converter. 178
FIGURE 5-17. Analog DLL approach using charge pump to implement adaptive body bias controller. 180
FIGURE 5-18. Dickson charge pump illustrating on chip generation of voltages higher than V_{CC} and lower than ground. 181
FIGURE 5-19. Example of how process skews can serve as anchor points to estimate probability distributions. 186
FIGURE 5-20. PMF for system consisting of N=1 blocks, standard deviation .121, targ frequency 0.967 191
FIGURE 5-21. CDF for system consisting of N=1 blocks, standard deviation .121, targ frequency 0.967 193
FIGURE 5-22. PMF for system consisting of N=1 blocks, standard deviation .0907, target frequency 1.0 194
FIGURE 5-23. CDF for system consisting of N=1 blocks, standard deviation .0907, target frequency 1.0 195
FIGURE 5-24. PMF for system consisting of N=10 blocks, standard deviation .121, target frequency 0.8378 199
FIGURE 5-25. CDF for system consisting of N=10 blocks, standard deviation .121, targ frequency 0.8378 199
FIGURE 5-26. PMF for system consisting of N=10 blocks, with standard deviation .0907, targ frequency 0.9024 200
FIGURE 5-27. CDF for system consisting of N=10 blocks, with standard deviation .0907, targ frequency 0.9024 201
FIGURE 5-28. PMF for system consisting of N=100 blocks, standard deviation .121, targ frequency 0.7338 203
FIGURE 5-29. CDF for system consisting of N=100 blocks, standard deviation .121, targ frequency 0.7338 203
FIGURE 5-30. PMF for system consisting of N=100 blocks, standard deviation .0907, targ frequency 0.8378 204
FIGURE 5-31. CDF for system consisting of N=100 blocks, standard deviation .0907, targ frequency 0.8378 205
FIGURE 6-1. Optimal V_{CC}/V_t biasing point trading off dynamic power with leakage power for constant performance. 212
FIGURE 6-2. Effect of effective switched capacitance variation on optimal V_{CC}/V_t biasing points. 217
FIGURE 6-3. Effect of I_0 variations on optimal V_{CC}/V_t biasing points. 218

FIGURE 6-4. Constant frequency curves showing isoperformance V_{CC}-V_t locus based on theoretical models. 221
FIGURE 6-5. Power vs. V_{CC} curves for different target frequencies based on theoretical models. 222
FIGURE 6-6. Power vs. V_t curves for different target frequencies based on theoretical models 223
FIGURE 6-7. Optimal V_{CC}/V_t as a function of frequency for theoretical models. 223
FIGURE 6-8. Optimal V_{CC}/V_t scaling power savings versus dynamic voltage scaling (V_t=.35 and V_t=.14) 226
FIGURE 6-9. Percent overhead in power by using dynamic voltage scaling instead of optimal V_{CC}/V_t scaling. 227
FIGURE 6-10. Layout and die photograph of the variable V_{CC}/V_t MAC chip. A) Ring Oscillator B) 4x4 MAC C) Adaptive body bias generator D) Die photo. 229
FIGURE 6-11. Test chip simplified global block diagram. 231
FIGURE 6-12. Block diagram of individual MAC operation. 232
FIGURE 6-13. Adaptive body bias generator version courtesy of Hitachi [Miyazaki]. ..234
FIGURE 6-14. Normalized ratio between delay of complex critical path and delay of 10 sequential inverters.236
FIGURE 6-15. V_{CC}-V_{BB} locus for fixed target frequencies. 238
FIGURE 6-16. Temperature and process variation impact on V_{CC}-ΔV_{BB} locus for fixed target frequencies (simulations). 239
FIGURE 6-17. Power as a function of supply voltage for different operating frequencies (from simulations) 240
FIGURE 6-18. Power vs. V_{CC} for different operating frequencies for nominal, slow process skew, and high temperature conditions (simulations). 241
FIGURE 6-19. Simulated power vs. frequency for dynamic voltage scaling versus optimal V_{CC}/V_t scaling. 244
FIGURE 6-20. Scope trace of critical path ring oscillator replica showing functionality at 0.1V operation 246
FIGURE 6-21. Scope trace showing DSP chip operation at 0.175V. 246
FIGURE 6-22. Critical path ring oscillator measurements of frequency versus forward and reverse body bias [Miyazaki]. 247
FIGURE 6-23. Forward bias currents that can arise in a Hitachi triple well process (shown for an inverter). 248
FIGURE 6-24. Supply and body current measurements for the test chip [Miyazaki] 249
FIGURE 6-25. Power versus supply voltage measurements for varying frequencies (forward bias current limited) [Miyazaki]. 251
FIGURE 6-26. Power versus threshold voltage measurements for varying frequencies (forward bias current limited) [Miyazaki]. 252

FIGURE 6-27. Measured power vs. frequency curves for dynamic voltage scaling versus optimal $V_{CC}/\Delta V_{BB}$ scaling. 253
FIGURE 6-28. Dual gated SOI (SOIAS) with active substrate. 254
FIGURE 6-29. Open loop V_{CC}/V_{BB} controller using lookup table. 256
FIGURE 6-30. Hybrid loop V_{CC}/V_{BB} controller using lookup table and adaptive body biasing circuit. 259
FIGURE 6-31. Dual loop V_{CC}/V_{BB} controller operation. 261
FIGURE 6-32. Possible use of test circuit to estimate total power consumption (dynamic + leakage) assuming body currents are negligible. 265
FIGURE 6-33. Test circuit with programmable taps to set ratio between dynamic and leakage power. 266
FIGURE A-1. High level block diagram of DSP test chip. 280
FIGURE A-2. Clock gating block. 282
FIGURE A-3. MAC block diagram. 282
FIGURE A-4. ABB generator block diagram[Miyazaki]. 283
FIGURE A-5. Substrate bias generator[Miyazaki]. 283
FIGURE A-6. Test chip orientation and pin mapping. 284
FIGURE A-7. Testing board configuration [Miyazaki]. 285
FIGURE A-8. Board I/O signals [Miyazaki]. 286
FIGURE A-9. Testchip global schematic. 287
FIGURE A-10. 4x4 MAC mesh network. 288
FIGURE A-11. MAC block with buffer stages shown. 289
FIGURE A-12. Multiply accumulate core. 290
FIGURE A-13. Linear feedback shift register implementation. 291
FIGURE A-14. Standard 8x8 multiplier architecture. 292
FIGURE A-15. Standard 24 bit adder. 293
FIGURE A-16. Standard register for accumulator. 294
FIGURE A-17. Clock gating block. 295
FIGURE A-18. Ring oscillator based on MAC critical path using worst case vector transitions values. 296

List of Tables

TABLE 1-1. Technology Scaling Trends 21
TABLE 1-2. HypotheticalDSPpowerandlifetimewith30%technologyscalingpergeneration. assumption 28
TABLE 2-1. CMOS delay, and % degradation for various W/L [shown as percentage of total circuit NMOS W/L]. 52
TABLE 2-2. Power/Energy considerations for MTCMOS multiplier. 52
TABLE 2-3. Parity generator performance as function of sleep transistor width for different input vectors 72
TABLE 2-4. Degradation of P6 delays in multiplier for different sleep resistances and vector choices 75
TABLE 2-5. Degradation of P1 delays in multiplier for different sleep resistances and vector choices 75
TABLE 5-1. Simulation results for system with N=1 blocks, and different sigmas. 196
TABLE 5-2. Simulation results for system with N=10 blocks, and sigma1 and 2 201
TABLE 5-3. Simulation results for system with N=100 blocks, and sigma 1 and 2 206
TABLE 6-1. Typical model parameters based on Hitachi SSH4 microprocessor. 216
TABLE 6-2. Detailed penalty of using DVS over V_{CC}/V_t scaling 227
TABLE 6-3. Simulation results of penalty of using DVS over V_{CC}/V_T scaling 244
TABLE A-1. Test chip pin descriptions 280

Chapter 1

Background

Low power circuit operation is becoming an increasingly important metric for future integrated circuits. As portable battery powered devices such as cell phones, pagers, PDA's, and portable computers become more complex and prevalent, the demand for increased battery life will require designers to seek out new technologies and circuit techniques to maintain high performance and long operational lifetimes. In non portable applications, reducing power dissipation is also becoming an increasingly important issue. In the past for example, high end microprocessors were engineered with performance being the primary goal, but in modern systems, power dissipation can be so large that heat removal becomes a problem.

1.1 Sources of Power in Digital CMOS Circuits

In modern digital CMOS integrated circuits, power consumption can be attributed to three different components: short circuit, leakage, and dynamic switching power. Short circuit currents occur in CMOS circuits during switching transients when both NMOS and PMOS devices are "on" but usually are small in well designed circuits[1]. Dynamic switching power is the dominant component of power consumption today and results from

the charging and discharging of gate capacitances during signal switching, as shown below:

$$P_{dynamic} = C_{switched} V_{CC}^2 f_{clk} \quad \text{(EQ 1-1)}$$

where $C_{switched}$ is the total effective switched capacitance, V_{CC} is the supply voltage, and f_{clk} is the switching frequency.

The third component of power consumption is leakage power. Although small compared to dynamic switching power, leakage power is becoming more important as scaling trends continue and efforts for ultra low power circuit operation are intensified. Leakage currents can be broken into various components such as PN junction reverse bias current, GIDL, oxide tunneling, and hot carrier injection, but the dominant component is subthreshold leakage currents given by

$$I_{leakage} = I_0 e^{\frac{V_G - V_S - V_{T0} - \gamma V_S + \eta V_{DS}}{n V_{th}}} \left(1 - e^{-\frac{V_{DS}}{V_{th}}}\right) \quad \text{(EQ 1-2)}$$

where V_{th} is the thermal voltage, n is the subthreshold swing coefficient constant, γ is the linearized body effect coefficient, η is the DIBL coefficient, and I_0 is given by

$$I_0 = \mu_0 C_{ox} \frac{W}{L} V_{th}^2 e^{1.8} \quad \text{(EQ 1-3)}$$

Neglecting body effect and DIBL, and assuming that $V_{ds} >> V_{th}$ then Eq 1-2 can be further simplified to the well known expression:

$$I_{leakage} = \frac{I_0}{W_0} W 10^{\frac{V_{GS} - V_T}{S}} \quad \text{(EQ 1-4)}$$

where $S = n V_{th} \ln 10$ is the subthreshold slope. For a typical technology with a subthreshold slope of 100 mV/decade, each 100mv decrease in V_t will cause an order magnitude increase in leakage currents. For extremely low V_{CC} - V_t operating points, leakage power can actually dominate dynamic switching power. Although subthreshold leakage currents

are limited in today's technologies, it will become an increasingly dominant component of overall power dissipation in future low power circuits[2].

1.2 Technology Scaling Impact on Subthreshold Leakage

Technology scaling is one of the driving forces behind the tremendous improvement in performance, functionality, and power in integrated circuits over the past several years. However, as scaling continues for future technologies, the impact of subthreshold leakage currents will become increasingly large. In industry, the standard scaling methodology has been constant field scaling with 30% reduction of all dimensions per generation as summarized below.

TABLE 1-1. Technology Scaling Trends

Scaling Parameter	1/S Constant Field Scaling	30% Scaling Field Scaling
W, L, T_{gox}, X_j	1 / S	0.7
Substrate doping	S	1.43
V_{CC}, V_{tn}, V_{tp}	1 / S	0.7
C_{gate}, I_{max}	1 / S	0.7
Propagation Delay	1 / S	0.7
Frequency	S	1.43
Chip Dimension	$1 / S^2$	0.5
Dynamic Power	$1 / S^2$	0.5
Leakage Power	exponential	exponential
Constant Die Assumption		
Chip Dimension	1	1
Functionality	S^2	1.43
Dynamic Power (Constant Die)	1	1
Leakage Power (Constant Die)	exponential	exponential

In general, using constant field scaling, physical dimensions (W, L, t_{gox}, X_j) all scale by a factor 1/S, substrate doping scales by S, and voltages (V_{CC}, V_{tn}, V_{tp}) scale by 1/ S, where S is greater than unity. Consequently, device currents scale by 1/S, gate capacitances scale by 1/S, and intrinsic gate delays scale by 1/S. Thus with 30% scaling of physical parameters, one can achieve close to a 50% improvement in frequency from

generation to generation, although this will be degraded by worsening interconnect dominated delays.

For 1/S constant field scaling, the resulting switching energy dissipated per event scales by $1/S^3$, where switching power dissipation scales by $1/S^2$ since the operating frequency increases with scaling. For a constant die size however, power dissipation due to dynamic switching currents remains relatively constant with scaling because the number of switching elements for the same die size will increase by a factor of S^2. On the otherhand, leakage currents increase exponentially with a reduction in V_t, and furthermore the total effective width of the devices will increase by a factor of S. For example, consider a technology with V_t of 400mv, and subthreshold slope of 80 mv/ dec that is to be scaled by 0.7 (corresponding to 30%). For a constant size die, we can see that scaling will provide almost a factor of 1.43 (close to 50%) improvement in frequency, while at the same time increasing the number of devices (functionality) by a factor of 2 (100%). The dynamic power dissipation scales by unity, but the leakage current will increase by a factor of $1.43*10^{(Vt/S\,(1-.7))}= 45$.

Although subthreshold leakage currents are not the dominant component of power dissipation in modern CMOS circuits, one can see that as a function of ordinary scaling the increase in leakage power can soon outpace dynamic switching power in future technologies. Unless circuit techniques or technology for controlling these leakage currents improve, one will be forced to reduce the scaling of V_t compared to the power supply in order to keep power dissipation in check. This will tend to reduce the amount of performance gain one can achieve through generational scaling[3].

1.3 V_{CC}-V_t Scaling Impact on Subthreshold Leakage Current

A major goal of the technology scaling trend described earlier is to reduce gate delay by 30% (thus increasing operating frequency by 43%), and to double transistor density through constant field scaling. This has been an underlining theme in microprocessor design, where high performance is the primary goal. A side benefit to constant field scaling was that a 50% reduction in power could be achieved during a basic shrink. However, in many systems, reducing power dissipation is becoming an important criteria to satisfy,

especially for battery powered applications. As a result, more aggressive V_{CC}-V_t scaling (which no longer obeys constant field scaling) can become important to minimize overall power dissipation. Fundamentally, it is possible to trade-off speed for power, and a simple way to do this is to scale the supply voltages.[4]

From an energy efficiency point of view, there is much potential to scale supply voltages to reduce power. Lowering the power supply is the most effective way to reduce power dissipation because the dynamic switching energy is proportional to the square of the supply voltage, as seen in Eq 1-1. In order to maintain performance during voltage scaling, one can employ parallelism to provide greater throughput at the expense of slower devices[5]. The quadratic reduction power dissipation due to voltage scaling is greater than the linear increase in switched capacitance due to the increased parallelism. Another effective approach to maintain performance while scaling supply voltages is to scale threshold voltages as well in order to provide a large enough gate overdrive to maintain high performance device operation. One can see how intrinsic gate speed can be maintained by scaling both V_{CC} and V_t, as shown in Eq 1-5

$$T_{pd} \propto \frac{CV_{CC}}{(V_{CC} - V_t)^{\alpha}} \quad \text{(EQ 1-5)}$$

where α models short channel effects[6].

To further illustrate this point, Figure 1-1 shows experimental measured data for a 101 stage ring oscillator consisting of iso-performance curves in V_{CC}-V_t space [7]. Clearly, it shows that a whole space of V_{CC}-V_t combinations will provide a fixed performance.

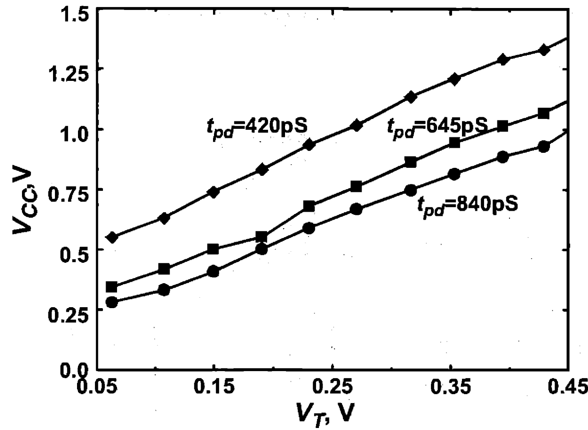


FIGURE 1-1. Constant delay curves for V_{CC}- V_t span.

With threshold voltage scaling however, subthreshold leakage currents will increase exponentially as quantified in Eq 1-2. Initially the increase in subthreshold leakage energy will be small compared to the quadratic reduction in dynamic power supply due to V_{CC} scaling for modern CMOS technologies. With further V_{CC}-V_t scaling however, the increase in leakage current can start to dominate the reduction in switching energies, indicating there must be an optimum V_{CC}-V_t point for a given target frequency. Figure 1-2 below shows experimental measurements for the 101 stage ring oscillator illustrating a minimum energy point as a function of V_t and the corresponding V_{CC} required to maintain performance.

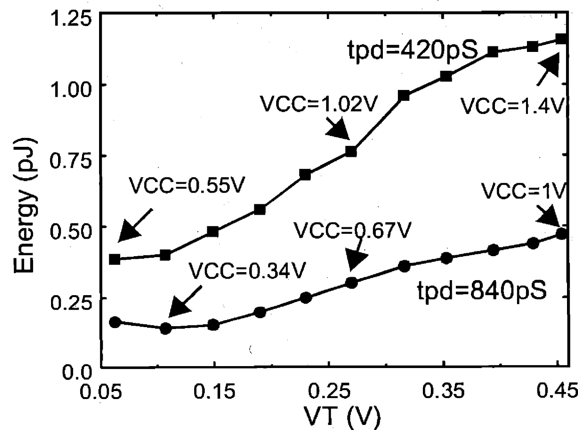


FIGURE 1-2. Minimum Energy point as function of V_{CC} & V_t.

For a given technology and V_{CC}/V_t ratio, the energy efficient V_t (and corresponding V_{CC}) point is significantly below the typical threshold levels of today's technologies. This excessive headroom indicates that there is still room for optimal V_{CC}-V_t scaling in today's technologies in order to lower overall power dissipation. Lowering threshold voltages has several undesired consequences however. Noise margins, short channel effects, and V_t variation all become worse with lower threshold voltages, and must be carefully balanced with any benefit one might gain in overall power dissipation.

1.4 Total Power Reduction Philosophy

As described in the previous section, subthreshold leakage currents will become a large component of total power dissipation in future technologies. Scaling theory alone dictates that subthreshold leakage currents will continue to become more important in overall power dissipation. Likewise, for low power scaling, the optimum energy point for V_{CC} and V_t will also correspond to a larger subthreshold leakage component. Although total power dissipation (dynamic + subthreshold leakage) during the active mode is reduced with scaling, further power gains can be achieved if subthreshold leakage currents are controlled wherever possible since these currents will make up a larger percentage of overall active power dissipation in future technologies. Furthermore, during the idle, or standby,

mode, where no computation is taking place, the overall idle power dissipation tends to increase since leakage currents are large as shown in Figure 1-3.

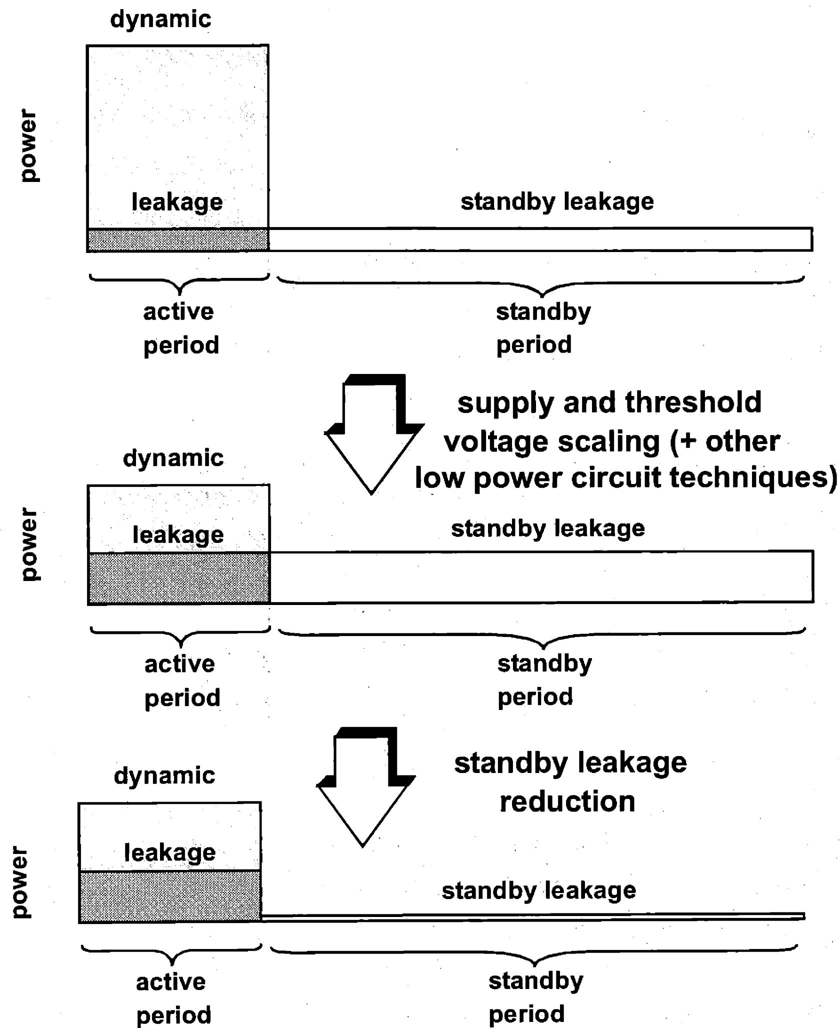


FIGURE 1-3. Graphical representation showing reduction of total active and standby leakage power.

In order to develop a power efficient system, both active and standby mode power reduction must be aggressively controlled such that the aggregate power consumption for the circuit in all modes of operation is minimized. In order to reduce the dynamic power component during the active period, traditional low power circuit techniques can be employed. For example, use of parallelization to lower supply voltages[5], scaling V_{CC}

and V_t[4], using multiple supply ranges[8] to power slow parts of a chip, or using circuit architectures that minimize effective switched capacitance can all be used to help reduce dynamic switching components during the active mode. However, new circuit techniques must be used to help control subthreshold leakage currents in both the active and standby modes because these components are becoming more of a problem in modern technologies and will tend to increase with technology and voltage scaling.

1.5 Burst Mode Circuits

Standby subthreshold leakage currents are especially detrimental in burst mode type circuits, where computation occurs only during short bursty periods and the system spends the majority of the time in an idle mode waiting for the next instruction. For example, a cell phone, pager, or even an X-terminal will all spend upwards of 90% of their time in a standby mode where the processor is waiting for a new instruction[7]. The X-terminal must wait for a user to input keyboard stimulus, while most of a cell phone's signal processing functions will remain idle until a call is made or received. For this class of burst mode type applications, it is extremely wasteful to have large leakage currents during the idle state because power is continuously drained during the standby mode with no useful work being done. The problem is especially severe for portable electronics, where battery power is drained needlessly during long idle times. As a result, subthreshold leakage reduction techniques during the standby mode can significantly reduce overall energy consumption for these burst mode applications.

1.5.1 Cell phone lifetime example

As an example of how technology scaling and leakage currents can become a problem for burst mode circuits, it is useful to estimate how a hypothetical DSP for a cell phone is affected with technology scaling and increased leakage currents. A cell phone spends the majority of the time in a waiting period, and a much shorter period in the active talk mode. Typically during the waiting period, the cell phone must turn on for short bursts of communication where the $t_{on}/(t_{on} + t_{off})$ ratio is approximately 0.01 during which the phone might have to do some bookkeeping work and communicate with a basestation.

As a rough approximation, one can assume a technology with V_{CC}=1.5 and V_t= 0.5V with the DSP leakage power normalized to 1 and the dynamic power normalized to 200. In this scenario, the leakage power is only a small fraction of the total power. Furthermore, one can assume that the battery has enough energy to supply 1.99 hours of talk time and 133.3 hours of standby time for the DSP (simplification that only the DSP is consuming power, and neglecting the effects of analog transceivers and amplifiers in the phone). During the talk period, both active and leakage power are present. During the standby time, leakage power and a fraction of the dynamic power (.01 from above) are present to account for the fact the phone must periodically perform computations.

Table 1-2 below shows typical values for how the chip power dissipation and battery lifetimes could vary with standard constant field scaling of 30% per generation. The table also shows a case where a leakage reduction technique (reduction of 3 orders of magnitude) is employed during the standby state. As described later, techniques such as MTCMOS can easily provide this energy savings[9].

TABLE 1-2. Hypothetical DSP power and lifetime with 30% technology scaling per generation.

Parameter	V_t=0.5	V_t=.35	V_t=.245
Active power	200	68.6	23.5
Subthreshold Leakage power	1	22.1	173.9
Subthreshold Leakage power (MTCMOS)	.001	.0221	.1739
Talk lifetime	1.99 hours	4.4 hours	2.03 hours
Standby lifetime	133.3 hours	17.5 hours	2.3 hours
Standby lifetime (MTCMOS)	199.9 hours	564.9 hours	977.6 hours

The table thus shows the impact of standard 30% constant field scaling per generation on system parameters. The numbers assume that each generational change is a simple shrink of the DSP, and the chip clock frequency stays the same. As a result, active power dissipation (CV^2f) scales by $(0.7)^3$ for each generation since voltage and switch capacitance both scale by 30%. Leakage power on the otherhand increases exponentially as threshold voltage drop by 30% per generation. With these representative numbers, one can see that the DSP talk lifetime increase from 2 hours to 4.4 hours and back to 2 hours as technology scales. This is due to the fact that active power continues to drop with scaling, but leakage power tends to increase. The effect on standby lifetime is quite severe, as it drops from

200 hours to 17.5 hours to 2.3 hours following these simple calculations. Clearly subthreshold leakage currents become increasingly problematic for burst mode circuits as scaling continues. However, by using a subthreshold leakage reduction technique during the standby state, standby times can be increased significantly.

1.6 Source Biasing for Subthreshold Leakage Reduction

Several different techniques have been explored in the literature to help reduce subthreshold leakage currents during the standby mode. "Source biasing" techniques can be thought of as a whole class of circuit techniques that have been described in the literature to reduce standby subthreshold leakage currents. For example, the literature includes techniques such as switched source impedance[10], self reverse biasing[11], and stack effect[12] which all rely on an underlining source biasing principle to reduce subthreshold leakage currents.

In source biasing, the main idea is to simply bias the transistor source of an off device in order to reduce the leakage current exponentially. This would be very effective, for example, to place a circuit block in an low leakage state during the standby mode. Figure 1-4 shows the log I_{ds} vs. V_{gs} curve for a standard off NMOS, where the drain is at V_{CC} and source at ground.

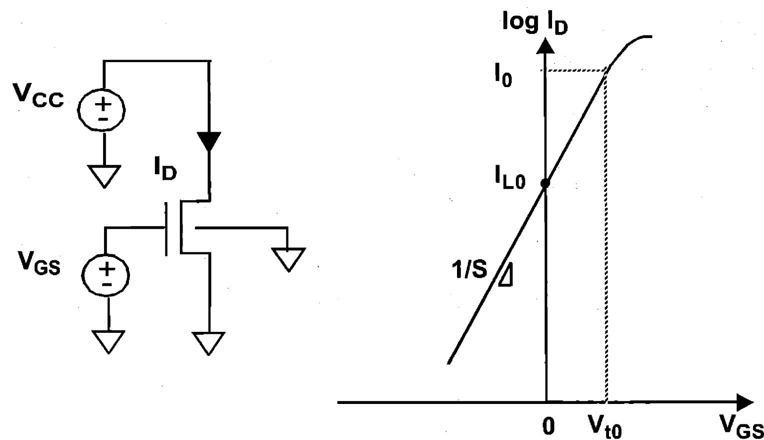


FIGURE 1-4. Standard log I_{ds} - V_{gs} curve showing subthreshold slope and off leakage current.

With V_{gs}=0, this becomes the typical leakage condition of an inverter with the inputs driven low, where the PMOS is strongly turned on and the leakage of the logic gate is set by the leakage of the off NMOS. Since V_{ds} is large for this device, the leakage current can be approximated very closely as

$$I_{DS} = \frac{I_0}{W_0} W 10^{\frac{-V_{T0}}{S}} \qquad \text{(EQ 1-6)}$$

However, if a bias voltage is applied to the source, the leakage current can be reduced dramatically as shown in Figure 1-5.

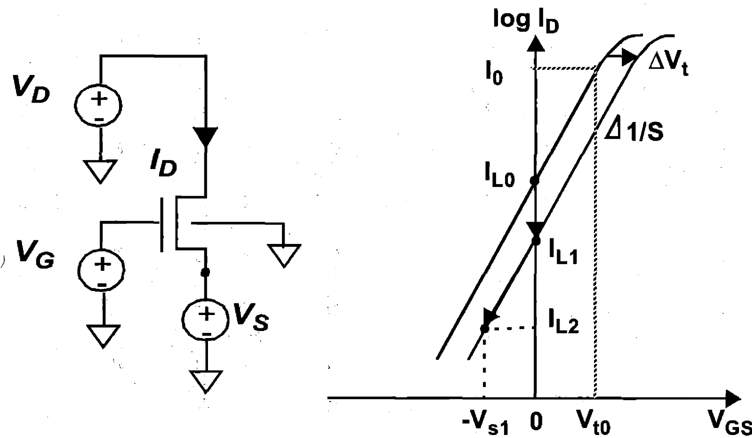


FIGURE 1-5. Source biasing principle.

The effect of the positive bias voltage results in two main mechanisms which tend to lower the leakage currents. First, since V_S is greater than zero, body effect tends to raise the threshold voltage of the device, which shifts the curve to the right such that the V_{GS}=0 leakage value would move from I_{leak0} to I_{leak1}.

$$I_{leak1} = \frac{I_0}{W_0} W 10^{-\frac{V_{t0} + K(\sqrt{V_S + 2\psi} - \sqrt{2\psi})}{S}} \qquad \text{(EQ 1-7)}$$

Second, the gate-source voltage becomes reversed biased such that V_G=0, V_S is positive, so V_{GS} is negative, further reducing leakage currents to I_{leak2}.

$$I_{leak2} = \frac{I_0}{W_0} W 10^{-\frac{V_S + V_{t0} + K(\sqrt{V_S + 2\psi} - \sqrt{2\psi})}{S}} \qquad \text{(EQ 1-8)}$$

In reality, it is not feasible to bias an off transistor source terminal with an ideal voltage source in order to reduce subthreshold leakage currents during the standby state. Instead, a variety of circuit elements can be used to provide this voltage offset.

1.6.1 Switched-source-impedance leakage reduction technique

One early technique which exploited this source biasing principle can be identified in [10], where researchers used switch-source-impedance CMOS to reduce standby leakage currents in standby modes.

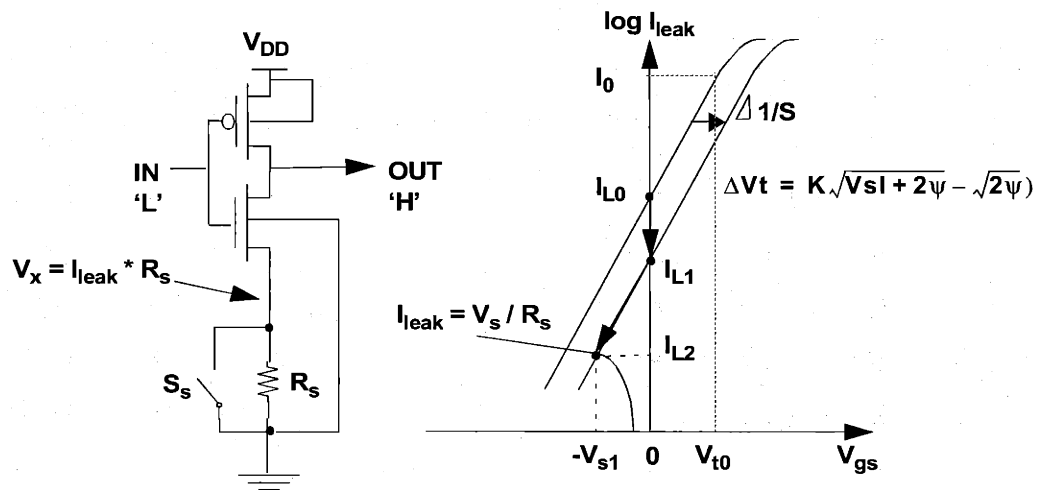


FIGURE 1-6. Source biasing example through switched source impedance technique.

This technique utilizes a large switched impedance that can be placed in series with an off low V_t transistor during the standby (and bypassed during the active mode), which will cause the standby leakage currents to decrease, but still maintain high performance during the active regime. Because the leakage currents flow through the large resistor during the standby state, a nonzero steady state voltage develops on node V_x, which is simply the

equilibrium operating point where the leakage current through a source biased device matches the i=v/r current through the switched resistance.

To effectively utilize this technique however, the off state of each gate must be known, such that an appropriate source impedance can be switched. For example, for an inverter with the PMOS off, a switched impedance to V_{CC} would be needed, but for the NMOS off condition, a switched impedance to V_{SS} would be required. Figure 1-7 illustrates these allowable configurations.

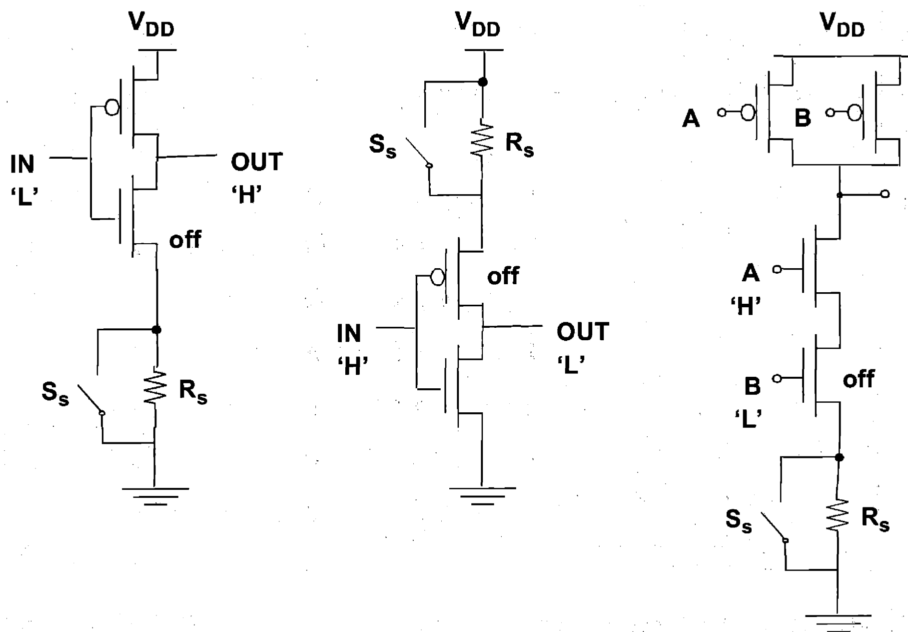


FIGURE 1-7. Switched source impedance techniques for different circuit off scenarios.

Although this technique can be used to reduce leakage currents by orders of magnitude, they require extremely large resistances, which are difficult to implement in LSI integrated circuits. It is possible to share the switched source impedance however among several different gates, which can help reduce the area constraint.

1.6.2 Self reverse biasing subthreshold leakage control technique

Another variation of this theme was presented in [11] known as self reverse biasing. This technique simply replaces the switched source impedance with another off transistor, so that the equilibrium value is set through the series of off devices. This technique was applied to decoded driver circuits where there are a large number of simple repeated logic blocks, but only a few of them operate at the same time. An extra series device is then placed in series with the decoded driver blocks such that the overall leakage currents of the previous drivers gets reduced due to the induced self reverse bias.

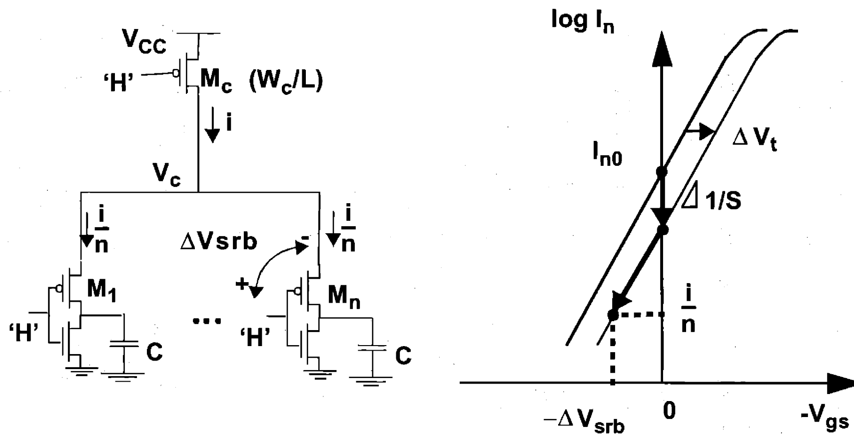


FIGURE 1-8. Self reverse biasing technique to reduce subthreshold leakage currents.

As illustrated in Figure 1-8, PMOS device M_C is placed in series with an array of off drivers (driven with a logic high) such that the PMOS devices are turned off. When M_C is turned on, it can supply enough current to the array because only a few drivers will turn on at any one time, so W_C need not be very large. However, when M_C is turned off, to the first order the leakage current gets set by the single device M_C if one neglects the drain voltage of M_C. As a result, V_{CC} decreases until a self reverse bias develops over the PMOS driver transistors until the driver leakages become i/n. This funnel effect can significantly reduce leakage currents because one has replaced several parallel leakage paths with a single one. This was further explored in applications to multi gigabit drams in

[13][14] and is illustrated in Figure 1-9. In large memory arrays, the actual number of switching gates is very small, so subthreshold leakage currents can be very large component of total power[15]. For example wordline drivers, traditionally sized quite large to drive the large wordline capacitances, are especially susceptible to large leakage currents.

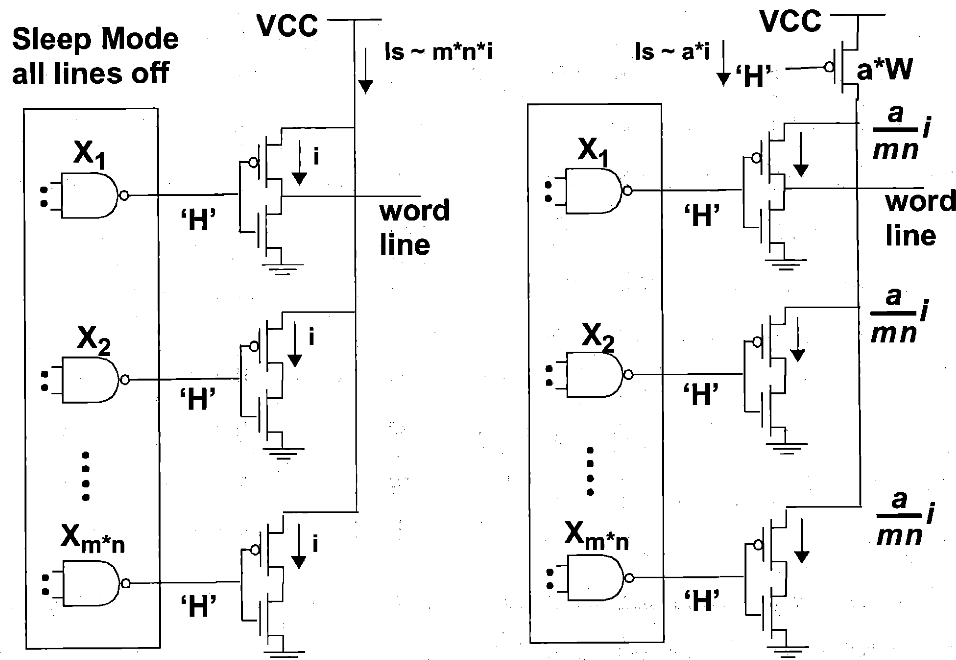


FIGURE 1-9. Self reverse biasing technique applied to DRAM wordline drivers.

In work presented in [11][13][14], the mechanism used to explain reduction in leakage currents was a "funnel" type effect where the leakage of n drivers was reduced to the leakage of a single stand alone series device. In actuality, the leakage reduction due to the funnel effect is only one component of the overall leakage reduction because the drain voltage of M_C can not be ignored. Because M_C and M_n PMOS devices are in series and are both turned off, the leakage amount will decrease even more compared to the case where M_C is off and the drain voltage is at V_{SS}. As described before, the leakage of the

M_n devices are reduced through self reverse biasing plus body effect. On the otherhand the leakage of M_C is reduced because of the reduced V_{ds} impact on the leakage, and also the fact that with smaller drain voltages, DIBL reduces, which causes leakage currents to reduce.

1.6.3 Stack effect for subthreshold leakage reduction

The stack effect was further explored in [12], where one could take advantage of the reduced leakage characteristics of series "off" devices in CMOS circuits. If a logic block can be fed with an appropriate vector during the standby mode to place stacked devices in the off state, then leakage can again be reduced. This approach is beneficial because no added source impedance circuitry is required, since the existing series connected devices in the circuit are exploited with appropriate choice of input vector. However, only limited amounts of leakage reduction are achievable with this approach.

1.7 Dual V_t Methods for Subthreshold Leakage Control

A more effective method of dealing with subthreshold leakage currents is to employ dual V_t technology, where the process has both high and low threshold voltage devices. By having two different flavors of devices, one can utilize specialized circuit topologies to take advantage of the speed benefits of low V_t as well as the leakage reduction benefits of high V_t devices. For example, in a technology with a subthreshold slope of 100mV/Decade, a 300mV change in V_t will produce 3 orders of magnitude reduction in subthreshold leakage currents. As a result, dual V_t techniques can be very effective as one continues to scale threshold voltages since multiple flavors of transistors are available. On the otherhand, source biasing techniques are much less effective than dual V_t techniques inherently, and will also become less effective with future scaling because the intrinsic leakage will continue to worsen, body effect will be less effective with increased short channel effects, and the biasing ranges will be reduced at low supply voltages. Dual V_t techniques on the otherhand will provide inherently fast and non-leaky devices that can be engineered through the process to provide the desired performance characteristics.

Dual V_t technology is becoming increasingly attractive in modern advanced CMOS technologies because the cost of an additional threshold voltage is relatively inexpensive for a process technology, requiring only an extra implant step. Yet the added flexibility of multiple threshold voltages for circuit designers is very valuable and can be used effectively to help provide high performance circuit operation while reducing subthreshold leakage currents during both active and standby circuit operation.

1.7.1 Dual V_t Gate Partitioning

One of the most straightforward applications of dual V_t technology is simply to partition a circuit into critical and non critical regions, and to only use fast low V_t devices when necessary to meet performance goals. This approach will reduce subthreshold leakage currents both in the active and standby modes since low V_t devices are only sparingly used[16]. In general, high performance device operation and low subthreshold leakage currents are mutually exclusive properties because each tends to push threshold voltage in opposite directions. For example, during the active mode, if circuits need to be fast then inherently they will be more leaky as well. As a result, it makes sense to only use fast low V_t devices where necessary in order to maintain high performance operation, and slow high Vt devices in peripheral or I/O circuitry, for example, that is not speed critical. However, in aggressive high performance low power circuit topologies that have many balanced critical paths, many of the gates cannot be slowed down, and there is only limited leakage reduction that can be achieved.

One dual V_t partitioning scheme that can be applied to random combinational logic can be to first implement the logic with all low V_t devices (to ensure fastest performance), and to selectively implant non critical gates to be high V_t. However, the difficulty is that non critical gates which were made high V_t can then become critical gates, as illus-

bias values as another control knob to tune threshold voltages dynamically. For example, application of maximum reverse bias, subject to device reliability and on chip voltage generation limits, can increase device threshold by several hundred mV, resulting in exponential reductions in leakage currents[19][20][21][22]. A variant of the body biasing technique, where the source terminals rather than the body terminals are biased was shown in[23]. Body biasing techniques, including dynamic tuning applications, are explored in more detail in later chapters.

1.9 Thesis Direction and Contributions

This thesis is geared towards exploring novel circuit techniques to provide subthreshold leakage control by exploiting the use of dual threshold voltages and body biasing. These basic principles for limiting subthreshold leakage currents have been explored previously in the literature, but circuit solutions have not been complete or robust enough to be applied to real VLSI systems. This thesis attempts to explore the main difficulties with existing solutions, and also explores new circuit structures and architectures that are very effective at limiting subthreshold leakage currents in both active and standby modes.

Chapter 2 for example describes in more detail MTCMOS techniques for reducing leakage currents during the standby state. Of particular difficulty with this technology is properly sizing the high V_t sleep devices to ensure that performance is maintained for all input vectors. These transistor sizing issues are explored in detail in this thesis, and a novel hierarchical sleep transistor sizing methodology was developed to ensure that MTCMOS will be fully functional even in a large system. Chapter 3 on the otherhand explores a new type of dual V_t approach that was developed, imbedded dual V_t, which eliminates the need for using series high V_t power switches associated with MTCMOS. A special case of the imbedded dual V_t principle was applied to domino circuits, which illustrates how subthreshold leakage currents can be drastically reduced during standby states without effecting overall performance.

Chapter 4 then explores several novel sequential circuits that can retain state during low leakage standby modes, while still providing high speed active operation. In the literature, several sequential circuit approaches have been proposed, but many of these

trated in Figure 1-10. This difficulty highlights the need for improved CAD tools to help designers handle the increased design complexity of dual V_t devices[17].

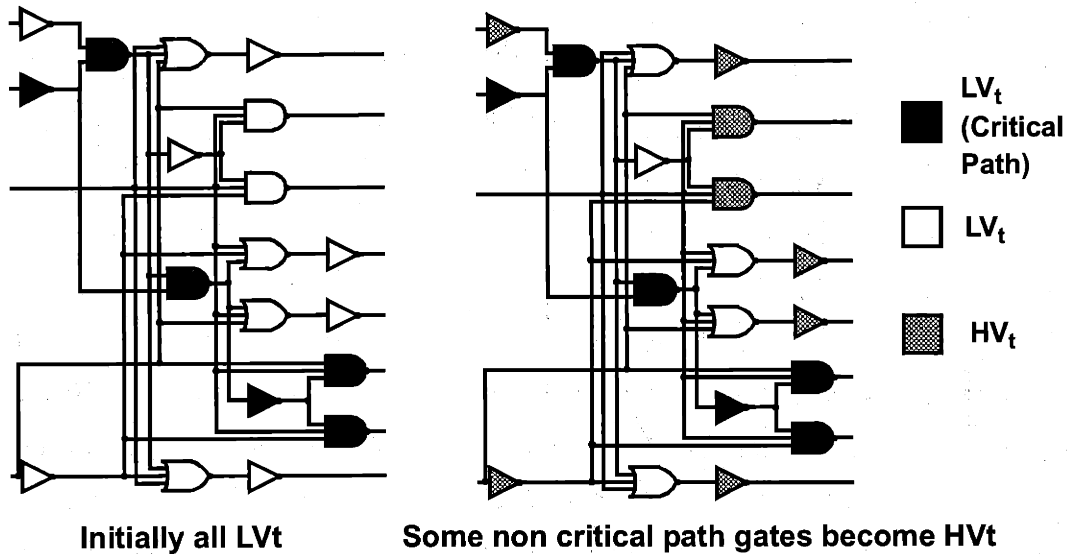


FIGURE 1-10. Dual V_t gate partitioning showing how critical paths can change.

1.7.2 MTCMOS Technology

A specialized case of dual V_t technology that is more effective at reducing leakage currents in the standby mode is MTCMOS (Multi-Threshold CMOS) first described in [18]. This technique involves using high V_t transistors to gate the power supplies of a low V_t logic block, and is described in more detail in the next chapter. MTCMOS gates are especially effective at reducing standby leakage currents because the leakage currents can be shut off with high V_t gates during the standby state, yet during the active state the internal logic can operate at high speeds through low V_t devices.

1.8 Body Biasing Techniques

A final technique of controlling subthreshold leakage currents is to adjust device body biases (in a triple technology for example) to tune device threshold voltages directly. Unlike dual V_t approaches, where explicit high V_t and low V_t devices are used to provide low leakage and high performance devices, body biasing techniques use the body terminal

had poor performance or added too much control overhead to be useful. This thesis takes a more fundamental approach at addressing sequential circuit architectures, and begins with an in depth analysis of potential sneak leakage paths that can arise in sequential circuits (this analysis has been missing, and sometime incorrect in prior art). By fully understanding these limitations involved with sequential circuit design, new latch and flip flop architectures have been developed that have better performance and are more efficient than existing approaches. Furthermore, a novel MTCMOS leakage feedback gate structure is also introduced, which provides a variant on MTCMOS circuit techniques where gates no longer have to float during standby states. This novel circuit idea has many important applications including improved flip flop structures and CMOS-MTCMOS logic interfaces.

The second part of the thesis explores new ways to utilize body biasing mechanisms for improving overall power dissipation. Previous work showed that straightforward application of reverse body bias during the standby state can reduce standby subthreshold leakage currents very effectively without any transistor sizing issues or state retention problems associated with MTCMOS. Because this standby application of body biasing is straightforward there was not too much new to add to the existing research. However, a large contribution of this thesis was to develop body biasing techniques that dynamically tune threshold voltages during the active mode. These techniques are very effective at helping to compensate the effects of parameter variations, which will increase as technology continues to scale. By tightening parameter variations through adaptive body biasing, yields can be improved, and subthreshold leakage currents during the active mode can be reduced by ensuring that circuits operate only as fast as necessary. The final chapter of the thesis pursues an extension to the body biasing idea where both supply and threshold voltages are tuned for minimum power during the active mode. For a given circuit operating at a fixed operating frequency there is an optimal V_{CC}/V_t scaling ratio that balances between dynamic and leakage power. Although this realization is not new, this research improves upon the state of the art by providing a theoretical background for how operating parameters, especially changing workloads, effect the optimal V_{CC}/V_t biasing point. This final chapter also concludes with a novel circuit architecture that automatically biases a chip to operate at the minimum power operating point. This is a significant

improvement over other low power techniques such as frequency scaling or dynamic voltage scaling to help lower power consumption for varying workload operating conditions.

Chapter 2

Multi-Threshold Voltage CMOS Technology

MTCMOS is a very attractive technique for reducing subthreshold leakage currents during standby modes by utilizing high V_t power switches. This technology is straightforward to use because existing designs can easily be modified into MTCMOS blocks simply by adding high V_t power supply switches, yet circuits can easily be placed in low leakage states at a fine grain level of control. MTCMOS technology processing is also straightforward, and requires only an additional implant processing step to provide multiple threshold voltage ranges. Although MTCMOS circuit techniques have been introduced in earlier work[9], MTCMOS still presents many challenges because of difficulties with optimally sizing the sleep transistors. This chapter will explore issues with sleep transistor sizing on MTCMOS circuit performance, and techniques for properly sizing sleep devices to ensure circuit functionality.

2.1 MTCMOS Technology Overview

The basic MTCMOS structure is shown in Figure 2-1, where a low V_t computation block is gated with high V_t power switches. When the high V_t transistors are turned on, the low V_t logic gates are connected to virtual ground and power, and switching is performed

through fast devices. However, by introducing an extra series device to the power supplies, MTCMOS circuits will incur a performance penalty compared to CMOS circuits, which worsens if the devices are not sized large enough. When the circuit enters the sleep mode, the high V_t gating transistors are turned off, resulting in a very low subthreshold leakage current from V_{CC} to ground. MTCMOS is only effective at reducing standby leakage currents and therefore is most effective in burst mode type application, where reducing standby power is a major benefit.

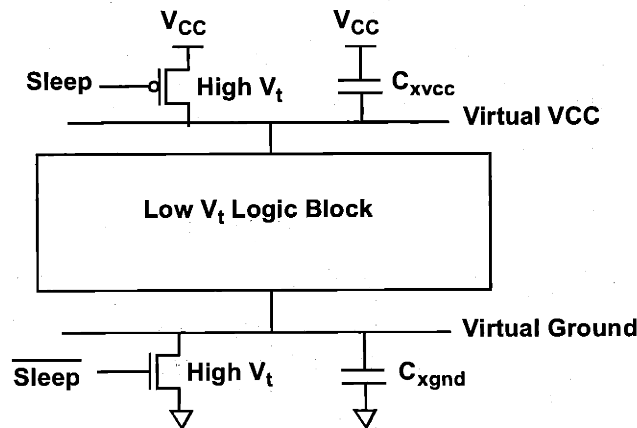


FIGURE 2-1. MTCMOS circuit structure showing both polarity sleep devices.

Although both PMOS and NMOS gating transistors are shown in Figure 2-1, only one polarity sleep device is actually required to reduce leakage if the logic block is purely combinational. NMOS sleep transistors typically are more effective because they have lower on resistances than a corresponding PMOS device, and subsequently can be made smaller for the same current drive. One possible advantage of using high Vt PMOS sleep devices though is that the body terminals can be tied to virtual V_{CC} instead of the actual power supply, which may simplify the logic structures because libraries PMOS cells may not need to be modified[25]. In any case, Figure 2-2 shows a simplified MTCMOS block that can be implemented with only an NMOS sleep device if the low V_t logic block is purely combinational.

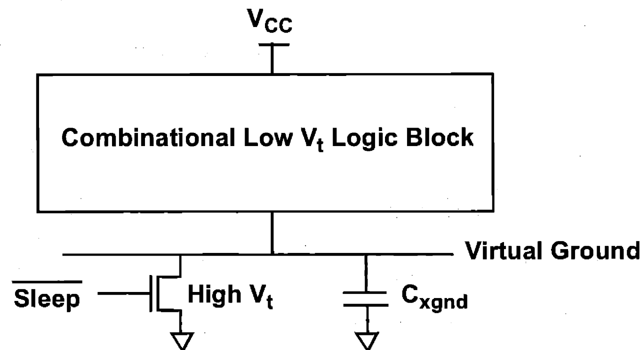


FIGURE 2-2. MTSMOC block with purely combinational logic block can use a single polarity sleep device.

It is important for the sleep transistor to have a high enough conductance during the active mode because the series resistance acts to degrade performance. One simple way to improve the conductance of the sleep device in the active mode is to overdrive the gate voltage (i.e. above V_{CC} for the NMOS device), and similarly the leakage reduction can be improved by under-driving the gate (i.e. below ground for the NMOS device) during the standby mode. Cutting off leakage currents by underdriving the gate is actually more useful for low V_t sleep devices, although reliability issues may become a concern.[24][25]. Even without under-driving though, MTCMOS circuits can achieve several orders of magnitude reduction in leakage currents, which results from two effects. First, the total effective leakage width of the original CMOS circuit is reduced to the width of a single off high V_t NMOS transistor (provided it is smaller than the original pulldown width), and second the increased threshold voltage results in an exponential reduction in leakage currents. To first order, the leakage behavior of the sleep device is characterized entirely by the threshold voltage of the NMOS sleep transistor (neglecting the drain voltage impact on leakage). A small reduction can be further achieved if the internal logic gates are configured such that all the NMOS core devices are off during the standby state (thereby creating a source biasing scenario), but the savings are negligible compared to the leakage reduction already achievable with a high V_t NMOS device.

MTCMOS circuits are thus very effective at solving the subthreshold leakage problem during standby modes in future technologies. The sleep state mode operation is very straightforward, simply involving turning off the power switches, and will produce guaranteed leakage reduction of several orders of magnitude. On the otherhand, the active mode circuit operation behaves theoretically just like an ordinary CMOS implementation, so existing architectures and designs can easily be ported to an MTCMOS implementation. One serious drawback to the widespread use of MTCMOS techniques though is that appropriate sleep transistor sizing becomes difficult for large circuit blocks[26]. Another problem (to be addressed in a later chapter) is that sequential circuits will lose data when the power transistors are turned off.

2.2 MTCMOS Sizing Impact On Performance

To understand the effect of sleep transistor on active circuit performance, one can model the effect of a high V_t NMOS sleep transistor as a linear resistor R to ground as illustrated in Eq 2-1 and Eq 2-2 using well known long channel MOS equations. During normal circuit operation, virtual ground should be close to real ground, so the V_{ds} of the high V_t switch will be very small and the device will be in the triode, or linear, region of operation, thus making a resistive approximation very accurate. As sleep transistor width increases, the effective resistance decreases, which improves circuit performance.

$$I_{DS} = \mu_n C_{ox}\frac{W}{2L}(2(V_{GS}-V_t)V_{DS}-V_{DS}^2) \qquad \text{(EQ 2-1)}$$

$$R_{eff} = \left(\frac{\partial I_{DS}}{\partial V_{DS}}\right)^{-1} = \left(\mu_n C_{ox}\frac{W}{L}(V_{GS}-V_t)\right)^{-1} \qquad \text{(EQ 2-2)}$$

Correct high V_t sleep transistor sizing is a key design parameter that affects the performance of MTCMOS circuits. If sized too large, then valuable silicon area would be wasted

and switching energy overhead would be increased. On the otherhand if sized too small, then the circuit would be too slow because of the increased resistance to ground.

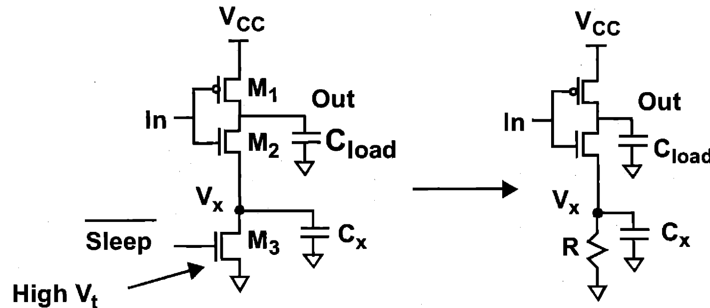


FIGURE 2-3. Sleep transistor modeled as resistor

Figure 2-3 shows a simple example of a single inverter gated with a high V_t NMOS device. During the active mode, one can see that only the high to low transition is degraded by the series switch, whereas the low to high transition is unaffected. This asymmetry can be exploited by selectively using PMOS and NMOS sleep transistors for individual gates depending on whether one transition direction is more critical than the other in a design. When the inverter is discharging, and neglecting the parasitic capacitance C_X, any charge flowing out of the source of M_2 will flow through the series device, inducing a voltage drop V_X. This voltage drop has two effects: first it reduces the gate drive from V_{CC} to V_{CC}-V_X, and second it causes the threshold voltage of the pulldown NMOS to increase due to the body effect. Both changes result in a decrease in the discharging current, which slows the output high to low transition. To maximize performance, the series transistor should be made as large as possible, subject to area and switching overhead constraints. As one continues to scale V_{CC} to lower voltages, the effective resistance of the high V_t sleep transistors will continue to increase due to reduced V_{gs}-V_t, and thus even larger size series devices will be required to provide a small enough resistance.

Figure 2-3 showed a scenario where a single inverter was gated with a series high V_t device. More realistically, in an MTCMOS circuit, many gates will be switching simultaneously through a shared common series device, so the sleep transistor sizing needs to

be even larger to take this into account. Figure 2-4 shows an example where N inverters simultaneously switch through a common series effective resistance.

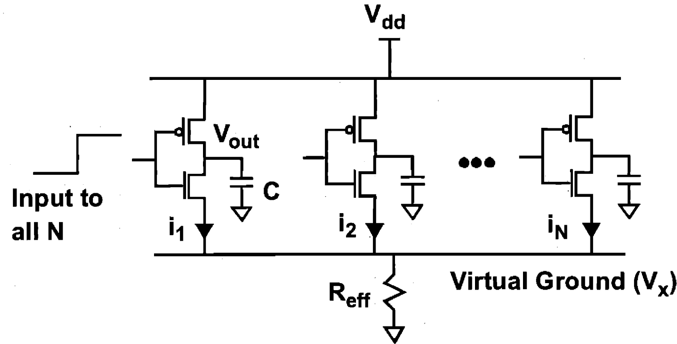


FIGURE 2-4. Circuit model for MTCMOS delay

In Figure 2-4, V_x is the equilibrium voltage where the current V_x/R_{eff} is equivalent to the sum of the saturation currents that are set by the reduced gate drive of each gate. Assuming the discharge current is constant and all gates are switching continuously during the period, the propagation delay for the jth gate can be modeled as:

$$T_{pdhl} \approx \frac{CV_{CC}}{2I_j} \quad \text{(EQ 2-3)}$$

where I_j is the saturation current that needs to be solved for explicitly shown in Eq 2-5 below. By summing the total MOSFET gain factors for each discharging gate, where $\beta_j = \mu_n * C_{ox} * (W/L)$ and $\beta_{total} = \beta_1 + ... \beta_n$, and equating V_x to the voltage drop across the sleep resistor, we have:

$$V_x = \frac{1}{2}\beta_{total}(V_{CC} - V_x - V_t)^2 R_{eff} \quad \text{(EQ 2-4)}$$

This can easily be solved for V_x, which can be used to compute the saturation current from the jth gate.

$$I_j = \frac{1}{2}\beta_j(V_{CC} - V_t - V_x)^2 \quad \text{(EQ 2-5)}$$

Clearly, as more devices switch through the same sleep transistors, currents will increase, resulting in more virtual ground bounce, which subsequently degrades perfor-

mance. Especially for larger circuits, sleep transistor sizing must be large enough to sink these currents during worst case conditions effectively without degrading performance. Fortunately, for a large circuit not all gates are switching at the same time, so the sleep devices can be shared among several different gates.

2.3 Inverter Tree Example Illustrating MTCMOS Delay

The following figure is a typical inverter tree structure implemented in an MTCMOS technology where an NMOS sleep transistor lies between virtual ground and ground. This circuit structure very clearly demonstrates how several gates can switch simultaneously and create large time varying voltage drops across the sleep transistor that slow down the circuits at different rates during signal propagation.

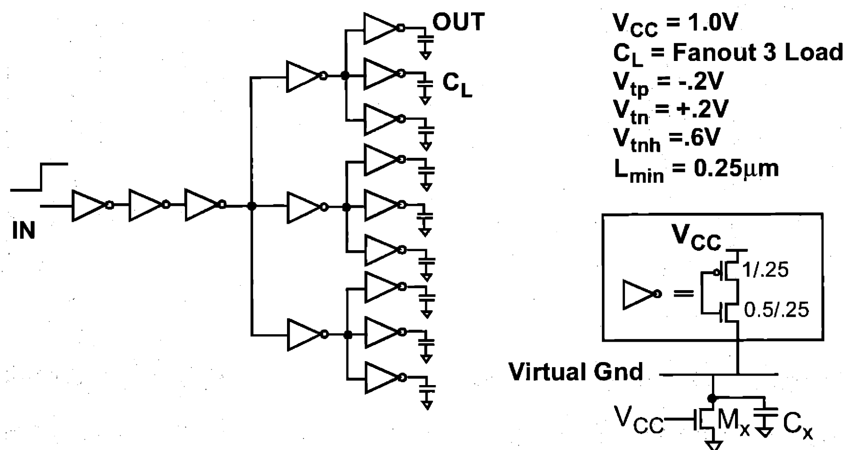


FIGURE 2-5. MTCMOS inverter tree.

Simulations were performed using an aggressive model suitable for ultra low voltage operation at 0.25 microns to illustrate the switching impact on MTCMOS circuits. The MTCMOS inverter tree shown in Figure 2-5 is implemented with a NMOS type sleep transistor, so high to low transitions are susceptible to ground bounce and performance degradation. As a result, the input 0->1 transition is especially slow because in the final stage, all nine inverters are discharging simultaneously, which causes the virtual ground line to bounce according to eqns 8-10. Figure 2-6 shows simulation results for the virtual ground transient and output waveform for different W/L sleep transistor sizes quantified

as a percentage of the total inverter tree NMOS pulldown width for this 0->1 input transition. The virtual ground transient reveals a gradual rise when the first inverter begins discharging and a sharper "bump" when the final stage is reached. The figure also shows how the output waveform slows down when the sleep transistor width is too small.

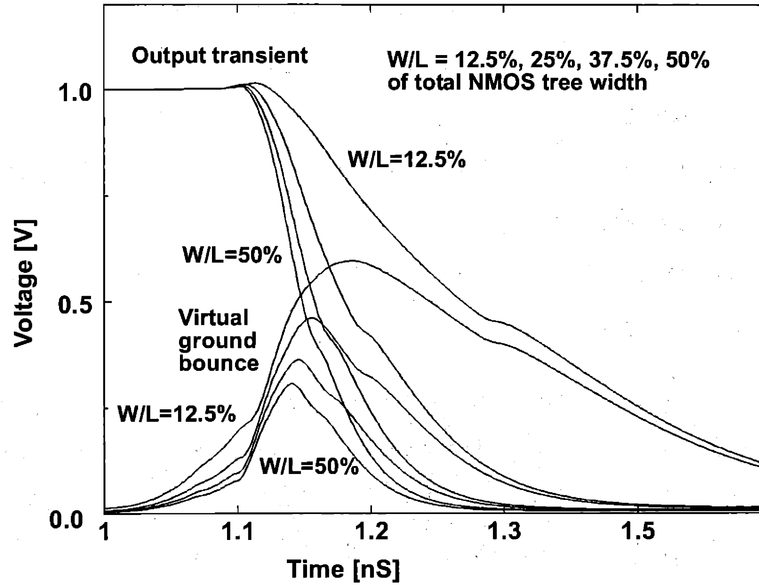


FIGURE 2-6. Transient Response for 0->1 transition

Conversely, in Figure 2-7, transients for the input 1->0 transition are tabulated. Because the final 9 simultaneous gates transition from low to high, large currents no longer flow through the NMOS sleep transistor. Instead, a maximum of only 3 gates from the previous stage actually transitions from high to low. Subsequently, the ground bounce and resulting degradation in performance is much less compared to the previous scenario.

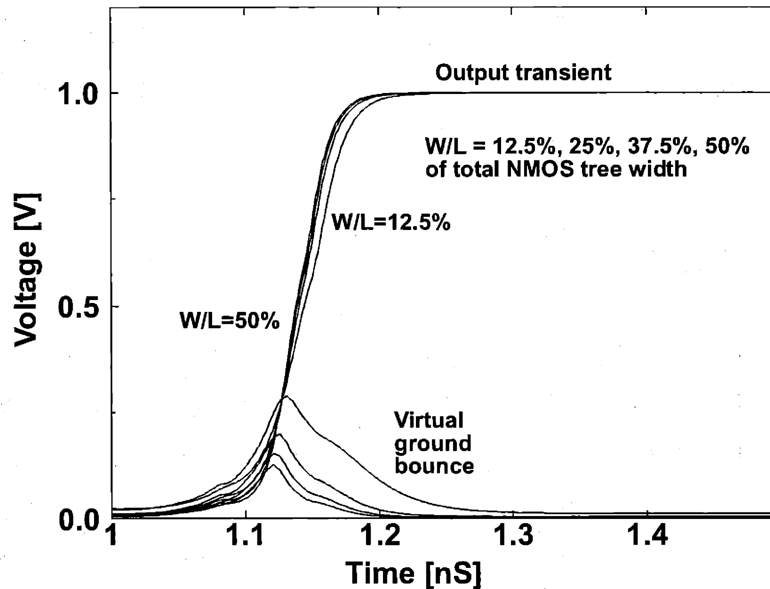


FIGURE 2-7. Transient Response for 1->0 transition

2.4 Vector Dependency on MTCMOS Sizing

For more complex MTCMOS circuits, the input vector and resulting circuit discharge pattern plays a very important role in determining worst case circuit performance. For example, the worst case pattern for a base CMOS design will not typically translate to the worst case pattern for an MTCMOS implementation because the MTCMOS circuit will be slowed down due to virtual ground bounce. Thus MTCMOS circuits will be more susceptible to input vectors that will cause large currents to flow through the sleep transistors, whereas ordinary CMOS circuits will not be affected. When analyzing MTCMOS circuits, one cannot simply examine a critical paths in the circuit, but must also consider all other accompanying gates that are switching. Because the worst case delay is strongly affected by different input vectors and glitching behavior, it is very difficult to correctly size the sleep transistor. In fact, to optimally size the sleep transistor, one would need to exhaustively simulate the entire circuit for all possible input vectors and all sleep transistor sizes. An exhaustive approach is unreasonable for anything but the smallest circuit blocks, so

alternative sizing strategies must be used. Approximations such as sizing sleep transistors based on total circuit width, estimating peak current spikes, and hierarchical sizing strategies based on partitioning circuits into smaller more manageable pieces have been explored in the literature.

2.4.1 8 bit carry save multiplier

A larger MTCMOS circuit like an 8x8 bit carry save multiplier demonstrates the impact of input vector on circuit performance. Because of size limitations, Figure 2-8 shows only a 4x4 version with a worst case delay path highlighted. Because of the regularity of this implementation, it is easy to see that one critical path (many others exist) lies along the diagonal and bottom row.

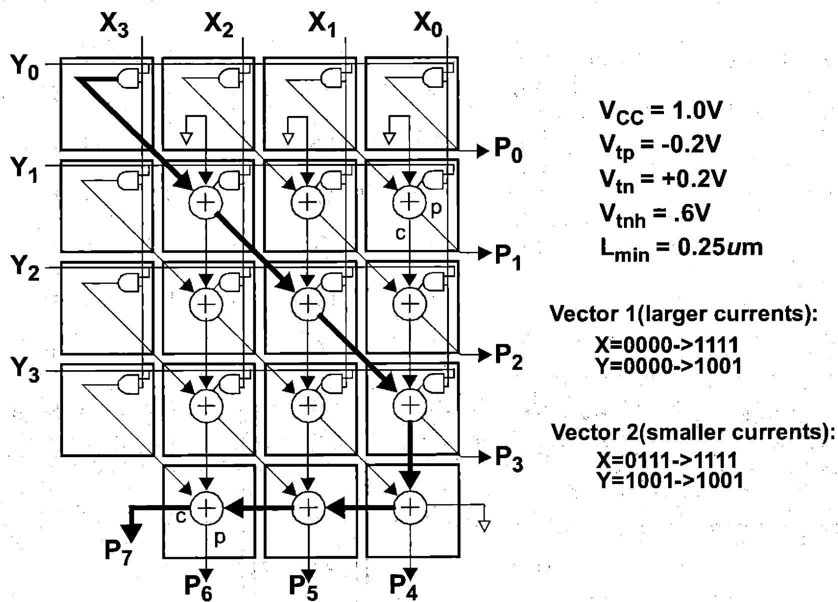


FIGURE 2-8. Carry save multiplier diagram (4x4 bit shown).

However, two distinct input vectors that give the same delay in a CMOS implementation can give very different results in a MTCMOS circuit. The transition from (x:00,y:00) -> (x:FF,y:81) for example causes many more internal transitions in adjacent cells and thus is more susceptible to ground bounce than the (x:7F, y:81) -> (x:FF, y:81) transition. The second input causes a rippling effect through the multiplier, where only a

few blocks are discharging current at the same time. Figure 2-9 shows how delay varies with the W/L ratio (expressed as a percentage of the total multiplier NMOS pulldown width) of the sleep transistor for these two cases.

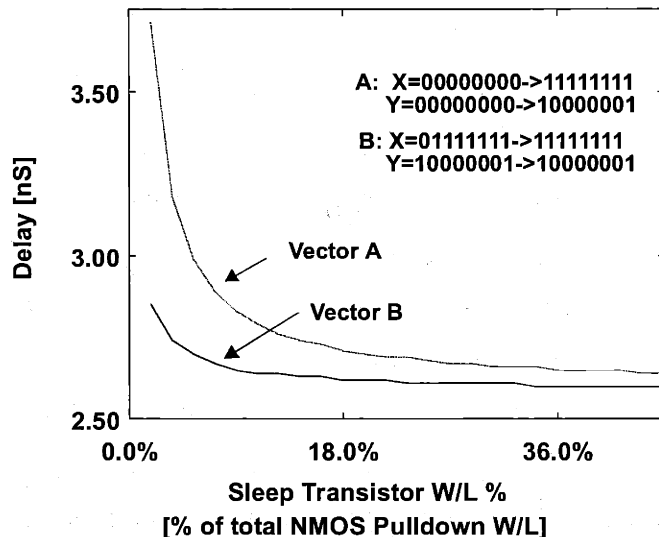


FIGURE 2-9. 8x8 bit multiplier delay vs. W/L (shown as percentage of total NMOS pulldown W/L) for different input vectors (SPICE).

Table 2-1 summarizes some key values from the plot. For example, if one wished to size the sleep transistor to provide less than 5% speed penalty for vector A, then one must size the sleep transistor W/L to be greater than 18% of the total effective W/L of the NMOS pulldown network for the multiplier. On the otherhand, if one were to examine the vector B, the same analysis could lead one to erroneously size the sleep transistor width to be only 5.4% of the total multiplier NMOS W/L, which would actually correspond to a 15% degradation in speed for the previous case. Since input vector strongly influences delays in MTCMOS, it is very important to determine the worst case input vector for opti-

mally sizing sleep transistors. To optimally size the sleep transistor would require exhaustive simulation of all possible input vectors, a task which is unrealistic for large systems.

TABLE 2-1. CMOS delay, and % degradation for various W/L [shown as percentage of total circuit NMOS W/L].

Initial XY	Final XY	Delay CMOS	% Degradation with W/L = 5.4%	% Degradation with W/L = 18%
0x 00 00	0x FF 81	2.59 ns	15.4%	4.6%
0x 7F 81	0x FF 81	2.58 ns	4.7%	1.6%

The energy characteristics of the 8 bit multiplier are also summarized in Table 2-2. For a standard low V_t CMOS implementation, leakage power is a significant component of total power dissipation, but can be reduced almost five orders of magnitude by using high V_t gating devices (sized 18% of the total W/L) during the standby mode. The switching energy required to go from sleep to active mode is small compared to the energy savings one could achieve during the low leakage standby state. For example, the sleep mode switching overhead energy would have been dissipated in only 200pS during the high leakage condition. As a result, in this example it makes sense to place the multiplier in sleep mode even at fine grain idle periods.

TABLE 2-2. Power/Energy considerations for MTCMOS multiplier.

Circuit Approach	Dynamic Switching Energy per Event	Leakage Power Active Mode [Watts]	Leakage Power Sleep Mode [Watts]	Sleep Switching Energy per Event	Sleep Switching Breakeven Time
CMOS	4e-12 J	1.5e-4 W	1.5e-4 W	NA	NA
MTCMOS W/L 18%	~4e-12 J	1.45e-4 W	2.2e-9 W	3.1e-14 J	2e-10 S

2.5 MTCMOS Sleep Transistor 2nd Order Effects

The parasitic capacitances due to wiring and junction capacitances on the virtual ground line, as shown in Figure 2-3, actually helps reduce the virtual ground line bounce by serving as a local charge sink or reservoir for current. However, having a large capacitance in itself does not offset the effects of a poorly sized sleep transistor. Since current is con-

stantly switching through the sleep resistance of a complicated logic block, the parasitic capacitance would have to be prohibitively large to prevent a large IR drop from developing. With a large time constant, it will also take longer for the virtual ground node to discharge back to ground if it does reach a large value. While capacitance on the virtual power does help reduce transient spikes in MTCMOS circuits, proper sleep transistor sizing is still of utmost importance.

MTCMOS logic blocks can also suffer from a reverse conduction phenomenon where current flows backwards from the virtual ground through the low V_t NMOS transistor and charges up the output capacitance, as shown in Figure 2-10. Conversely, for a PMOS sleep transistor the output capacitance partially discharges as current flows up towards a virtual V_{CC} line. To be more specific, in the NMOS case, the virtual ground node can rise above 0V so that another gate, which is supposed to be low, can experience reverse conduction as the output voltage rises from 0V to V_x. This charging current comes from the discharging current of other gates transitioning from high to low, where a fraction of the discharge current is actually bypassing the sleep transistor. As a result, the MTCMOS circuit is slightly faster because the V_x voltage drop is not quite as large as one would expect if all current flowed through the sleep transistor to ground. Another effect of the reverse conduction, which pins output low voltages to V_x, is that a gate charging from low to high would be faster since it is already precharged to V_x. The drawback is that the noise margins in the circuits are reduced, and in the worst case the circuit can fail logically.

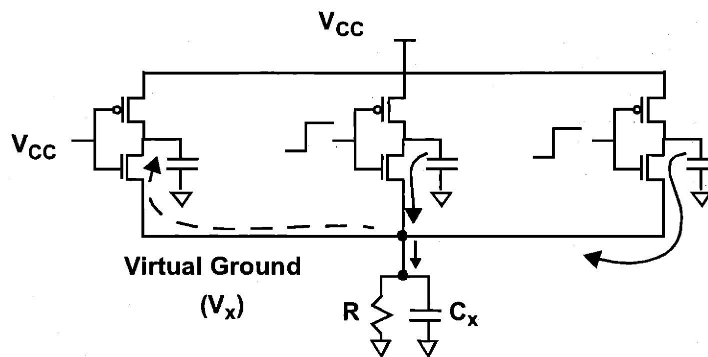


FIGURE 2-10. Reverse Conduction Paths

2.6 MTCMOS Transistor Sizing Tools

As described in the previous section, sleep transistor sizing is the most critical design issue in MTCMOS circuits. The sleep transistor should be sized large enough to ensure that performance requirements are met, yet it should not be overly conservative because large sleep transistors take up valuable silicon area, and result in larger switching capacitance when going between active and standby modes. Sleep transistor sizing is especially difficult in complex MTCMOS circuits because the worst case input vector is highly dependent on the discharge patterns of the core logic since vectors which cause more transitions through the high Vt device will cause more ground bounce and performance degradation. As a result, in MTCMOS one can no longer simply examine a critical path (as in a CMOS circuit) to determine circuit performance, but instead must consider all discharge patterns of non critical regions of the circuit. Worst case vectors in MTCMOS also vary as sleep transistor sizes change. For example, an MTCMOS sleep transistor sized very large such that there is very little performance degradation compared to CMOS might have a critical path that is long, but does not cause many extra transitions. However, for that same circuit with a smaller sleep device, where there is more impact due to ground bounce, a different vector which exercises a shorter path might become critical because of the added degradation component due to ground bounce. Thus, in order to accurately size the sleep transistor for a complex circuit block, one must simulate all possible input vectors to determine the worst case MTCMOS input vector for a given sleep transistor size. Because of the exponential explosion in complexity for exhaustive simulations, this approach can only work for small circuits.

2.7 Variable Breakpoint Switch Level Simulator

To help address the problem of exhaustive vector simulations for MTCMOS sleep transistor sizing, a tool to provide switch level simulations for MTCMOS was developed. The motivation behind building this tool was to develop a much faster simulator than currently available so that exhaustive simulations could be performed several orders of magnitude faster than with a differential equation solver tool like SPICE. A traditional switch level simulator, although fast, would not be able to handle the degradation effect of the MTCMOS sleep device, so the variable breakpoint simulator was developed so that reasonable

first order simulations that take into account sleep transistor size can be used to help determine worst case vectors for sleep transistors[27].

To help analyze worst case input vector patterns, a switch level variable breakpoint simulator was developed to rapidly compute delay as a function of sleep transistor size. The advantage of this simulator is that first order timing information can be gathered very quickly for very large input vector spaces compared to a slow differential equation solver like SPICE. A traditional switch level simulator, although fast, would not be able to handle the degradation effect of the MTCMOS sleep device, so the variable breakpoint simulator was developed so that reasonable first order simulations that take into account sleep transistor size can be used to help determine worst case vectors for sleep transistors. Rather than using the timing information as is, the tool is more useful for identifying potential vectors that will cause large variations in an MTCMOS circuit and can be used to narrow down the vector space to be analyzed with a more detailed simulator like SPICE.

The underlining algorithm behind this simulation tool is to dynamically adjust each gate's propagation delay based on the total number of gates switching, since different amounts of currents will produce different voltage drops across the sleep transistor. If each gate is modeled as an equivalent inverter with an effective load capacitance C_L, then the delay model derived in Eq 2-4 and Eq 2-5 for N inverters discharging simultaneously can be applied directly to more complex logic circuits.

The input and output voltage waveforms for each gate are treated as piecewise linear, and gates are assumed to begin switching exactly when the input voltage exceeds $V_{CC}/2$. In the case of an ordinary CMOS implementation (with sleep resistance equal to 0), the simulation tool simply models each gate as a constant current source that discharges a load capacitance. When a finite sleep resistance is introduced in the circuit, the gates are modeled as time varying (stepwise) current sources discharging their respective load capacitances, which results in a piecewise linear output voltage whose slopes can vary in time. These breakpoints occur whenever a gate in the logic block starts or stops switching because delays must be recomputed when the total current flowing through the sleep transistor changes. With each gate modeled as a first order dynamic system, one only

needs to keep track of the current output voltage (state) and input stimulus to predict the delay behavior.

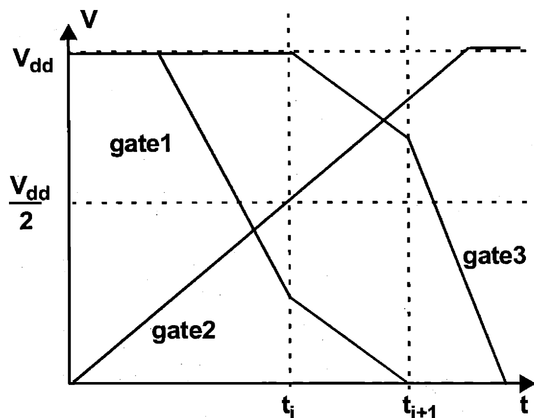
In order to process these breakpoints, the simulator computes an associated "best guess" for time to reach the switching threshold and time to finish switching for each gate. The simulator time steps to the nearest breakpoint, determines if any new elements are switching and then recomputes the "best guess" for these breakpoints by taking into account slower or faster gate transitions. The breakpoint times for individual gates are not fixed because if another gate switches first, then the speed of the subsequent gate will change, requiring a new delay calculation. For a simulation time of T_{sim}, current drive of I_j, and load capacitance C_L, a discharging gate who's output voltage is currently > $V_{CC}/2$ would have it's expected switching threshold breakpoint calculated as

$$T_{dnext} = T_{sim} + \frac{C_L}{I_j}\left(V_{out}(T_{sim}) - \frac{V_{CC}}{2}\right) \qquad V_{out} > \frac{V}{2} \tag{EQ 2-6}$$

Conversely, the simulation time breakpoint corresponding to when the gate finishes transitioning is represented by:

$$T_{end} = T_{sim} + \frac{C_L}{I_j}V_{out}(T_{sim}) \qquad V_{out} > 0 \tag{EQ 2-7}$$

Figure 2-11 shows the output waveforms as functions of time for three different gates in a larger MTCMOS circuit. One breakpoint is labeled as t_i, corresponding to the switching threshold of gate 2, and another is shown as t_{i+1}, corresponding to the time gate 1 finishes discharging. The other six breakpoints are not labeled.



1. gate 2 charges up
2. gate 2 crosses $V_{dd}/2$ at t_i and causes gate 3 to switch
3. gate 1 slope reduces due to added discharge current
4. gate 3 slope increases at t_{i+1} since gate 1 ends

FIGURE 2-11. Variable break point switch level simulator function.

Immediately before time t_i, gate 1 is discharging at a constant slope and gate 2 is transitioning from low to high. However, at the breakpoint t_i, gate 2 passes the threshold voltage and causes gate 3 to begin discharging. This increased current causes the virtual ground to bounce, and consequently both gate 1 and gate 3 slow down. At this point subsequent breakpoints will have to be updated to reflect slower circuits, so that the next breakpoint, t_{i+1}, is actually later in time than what was predicted earlier. When gate 1 finishes switching, gate 3 will speed up because less current needs to be sunk through the sleep transistor. Again, the breakpoints are recomputed at this point to reflect different operating conditions. The variable breakpoint simulator thus only needs to simulate the circuit at breakpoints which are variable in time and computed from the current operating conditions.

The delay model used in the variable breakpoint switch level simulator has several limitations. First of all, the assumption that the output capacitance is discharged by a current source equal to the saturation current I_j is simply false, since the transistor does spend time in the triode, or linear region of operation. Second, we neglect the effect of parasitic capacitances on the virtual ground line, but this effect becomes important only for large resistances or large capacitances. Also, the effect of the input slope on output delay time is

ignored, and only a very simplistic first order MOSFET model (neglecting body effect, channel length modulation, velocity saturation) is used. Another important limitation is that complicated gates are modeled as a simple inverter, which can also lead to timing inaccuracies. By addressing these issues in future work, the simulator accuracy can be improved significantly. However, since the simulator is most useful for qualitative analysis in determining potential vectors that are sensitive to MTCMOS, complete timing accuracy is not mandatory.

2.7.1 Variable breakpoint simulator for inverter tree

To verify the accuracy of the variable breakpoint simulator, it was applied to the clock distribution network shown in Figure 2-12 and compared with SPICE results.

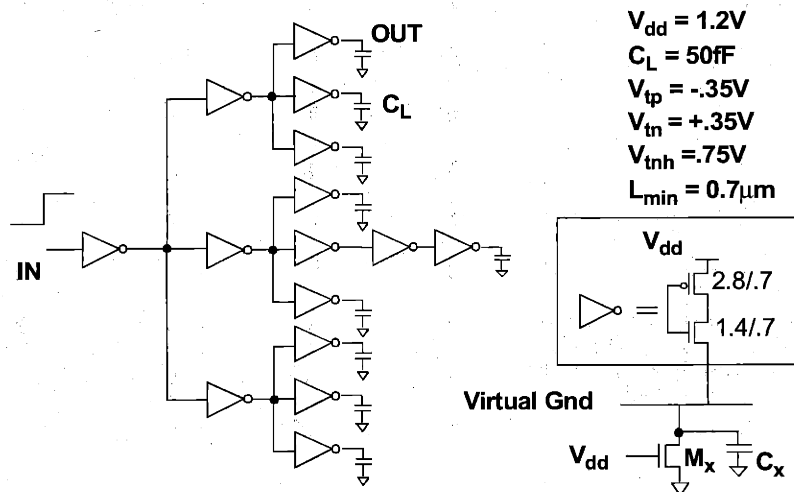


FIGURE 2-12. MTCMOS inverter tree used to compare variable breakpoint simulator with spice simulations.

Figure 2-13 shows delay measurements versus sleep transistor width for both SPICE and from the switch level simulator. The simulator captures the basic effect of sleep transistor sizing on propagation delay, and even though it is based solely on a first

order delay model, still manages to track the switching variations of this MTCMOS circuit.

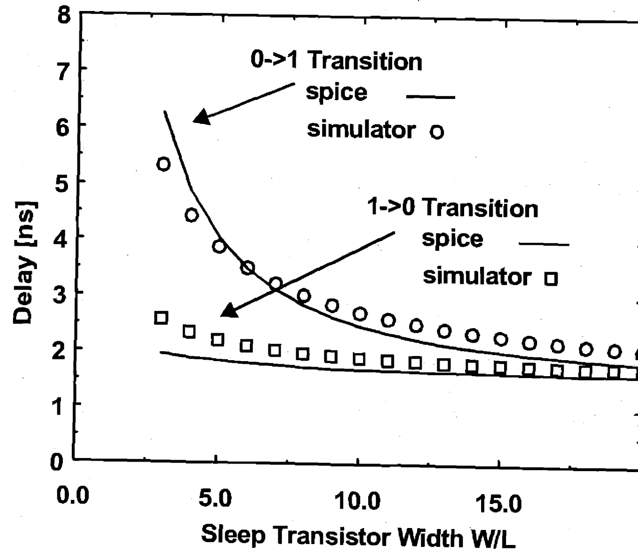


FIGURE 2-13. Delay vs. W/L ratio for 0->1 and 1->0 inputs.

Figure 2-14 shows the virtual ground bounce transients in the inverter tree during the transition as computed from SPICE as well as the simulator. Since the simulator models discharging gates as constant current sources and neglects the effects of capacitance in parallel with the sleep transistor, the ground bounce transient is simply a stepwise function. For the very high resistance case (unrealistic/ undesirable in actual circuits), the virtual ground is very slow in discharging due to a larger RC time constant.

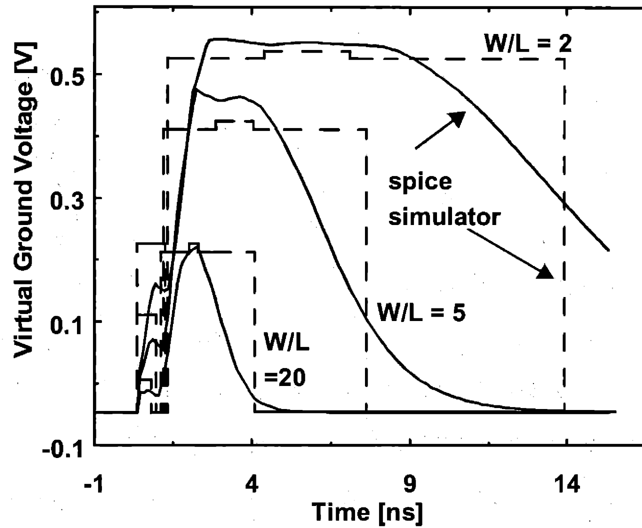


FIGURE 2-14. Virtual ground transient for 0->1 input for simulator and spice simulations.

2.7.2 Variable breakpoint simulator for 3 bit adder

A 3 bit ripple carry adder was also exhaustively simulated both with SPICE and with the variable breakpoint switch level simulator. The adder is a standard mirror adder implemented with 3x28 transistors, and the circuit was simulated with the initial carry bit grounded, but using every possible pair of 6 bit input vectors. This resulted in 26 * 26 = 4096 possible vectors.

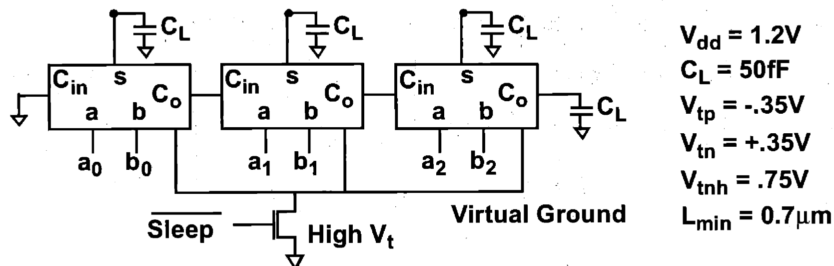


FIGURE 2-15. 3 Bit adder to test variable breakpoint switching simulator.

The variable breakpoint switching simulator was over 1000x faster than spice simulations when this circuit was exhaustively simulated for all input combinations. Figure 13 shows a comparison between the propagation delay on the 3 bit ripple carry adder as a function of W/L between SPICE and the variable breakpoint switch level simulator. Two different vector pairs are shown: (x:000 y:100)->(x:011 y:101) is the worst case delay vector pattern, while (x:011 y:100) -> (x:011 y:101) is a vector less susceptible to MTCMOS delay. From the plot, we can see that the simulator gives extremely good results for delay.

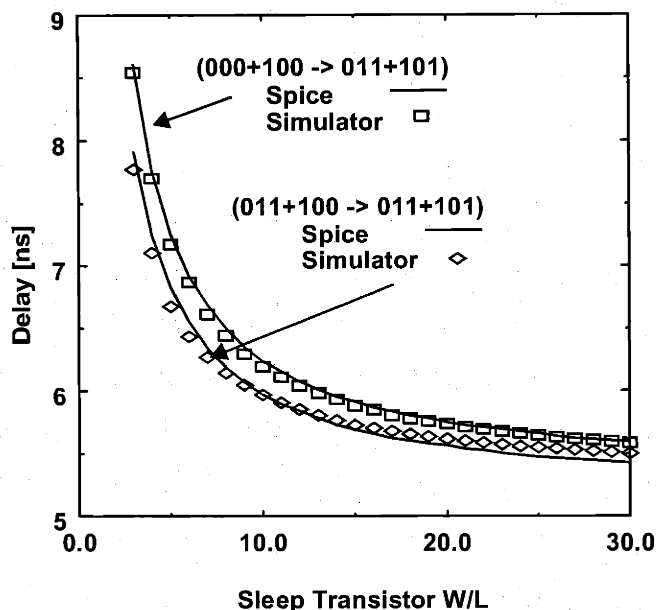


FIGURE 2-16. 3 Bit adder spice versus variable breakpoint simulator comparison for two sample vectors.

2.7.3 Simulator accuracy

The variable breakpoint switch level simulator was shown to reasonably track MTCMOS delays as a function of sleep transistor sizings. As a result, a simulator approach can be used as a first order tool to help analyze a large number of circuit vector patterns to determine worst case input vectors in order to properly size the high V_t sleep switches. Although the above examples show vector samples that match very well with SPICE simulations, there are still many scenarios where mismatches may arise. For example, input

patterns that cause large amounts of internal node glitches are not modeled accurately with this simulator, and second order circuit effects like velocity saturation, body effect, and parasitic capacitances are not taken into account. Thus more work can be done to improve the simulator to give more accurate results. In effect, there is a trade-off between the simulator accuracy and the simulator speed (for example a switch level simulator versus an RC simulator versus a full blown SPICE simulator).

Even if the simulator were improved to the point where it could accurately and reliably predict worst case input vectors in MTCMOS circuits, it would still take too long to compute for large circuits. An approach that exhaustively simulates all possible input vectors increases in complexity as $O(N^2)$ for large circuit blocks, so even if the simulator can be made exceedingly fast, the problem could easily become intractable for larger systems.

2.8 Hierarchical Sizing Strategy Based on Mutual Exclusive Discharge Patterns

Another approach to sizing sleep transistors is a hierarchical sizing approach based on mutual exclusive discharge patterns[28]. In this approach, one synthesizes an appropriate sleep transistor size based on mutual exclusive discharge patterns to guarantee a fixed performance requirement, which is approximately an O(N) procedure. However, the drawback for this synthesis approach is that the sleep transistor will be larger than optimum. As a result, one will trade-off larger area than necessary for a more tractable approach to sizing MTCMOS sleep transistors. In contrast, the brute force method of sizing MTCMOS sleep devices involves exhaustively simulating all possible input vectors to find the worst case delay for a given sleep transistor size. For different sleep transistor sizes, the worst case input vector could also change, requiring several iterations to ensure that a target performance level is met.

These two sleep transistor sizing methodologies have very different fundamental goals. One of the characteristics of choosing an optimal sleep transistor size to meet a fixed performance specification is that while the worst case delay through the circuit degrades by no more than a fixed percentage, individual gate delays may degrade by more

or less than that fixed percentage. For example, if some circuit portions are especially fast, they can make up for slower regions, and the net result still satisfies our performance goal.

The hierarchical synthesis technique on the otherhand uses a different approach. The new sizing methodology uses a bottom up approach to synthesize the sleep transistor by ensuring that every individual gate will not degrade by more than a fixed percentage. By guaranteeing that each individual gate meets or exceeds a local performance constraint, then any combination of gates in a path will also meet or exceed the performance constraint, and thus the macro performance measure will be satisfied.

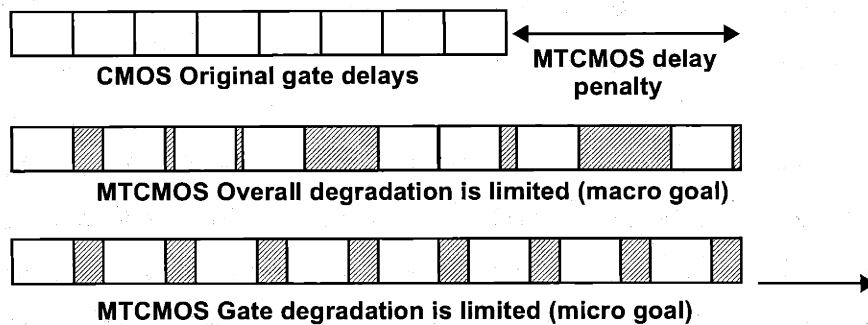


FIGURE 2-17. Different MTCMOS gate degradation scenarios that still satisfy overall delay.

Figure 2-17 shows an example where an original CMOS critical path is degraded in an MTCMOS implementation (scale is exaggerated) for the case where internal gates can degrade more or less than a fixed percentage, and for the case where internal gates each degrades by no more than a fixed percentage. Both cases will meet the worst case delay characteristics, although the last case will be a much more conservative estimate. For example, if the sleep transistor is sized large enough so that no gate degrades by more than a fixed percentage, it is likely that some gates will degrade by less than this limit. As a result, the cumulative delay most likely would degrade by less (but never more) than this limit. If an MTCMOS implementation using a single polarity sleep device is used, then roughly only half of the individual MTCMOS gates(the pull down paths) will be degraded and for well balanced circuits the performance degradation would be half as well. For

example, if one were to ensure that all elements degrade by no more than 5% during an MTCMOS implementation, then one can guarantee that any interconnection of the these elements will degrade by no more than 5% from it's original CMOS counterpoint. Furthermore, with a single polarity sleep transistor roughly only half of the individual MTCMOS gates will be degraded, resulting in an overall degradation of only 2.5% in performance for a balanced circuit.

Forcing every single gate to meet a nominal performance measure is a much more demanding criteria than simply constraining the cumulative delay. However, in the context of MTCMOS circuits, it is much easier to implement this sizing strategy because one does not need to determine the worst case input vector pattern for the whole circuit. Instead, each individual gate can be assigned it's own high V_t sleep transistor, whose size will be locally determined through exhaustive simulations. Once the MTCMOS circuit is sized with individual sleep transistors then one can systematically merge the sleep transistors together because they can be shared among mutually exclusive gates, where no two gates can be discharging current at the same time. Finally, these sets of sleep transistors can then be combined to make a single sleep transistor for the whole circuit that guarantees that for any input vector, the MTCMOS circuit performance will be within the specified range of the corresponding CMOS circuit.

2.9 Example of Hierarchical Sizing

A good way to describe the sleep transistor sizing and merging technique is through an example. Figure 2-18 shows how an MTCMOS circuit can initially be sized using individual sleep transistors that can be merged together at later steps into a common power switch for the larger block. The circuit consists of three chains of five low V_t transistors, and measurements are made for the input to output delay, the delay for inverter I5, and the virtual ground bounce transients.

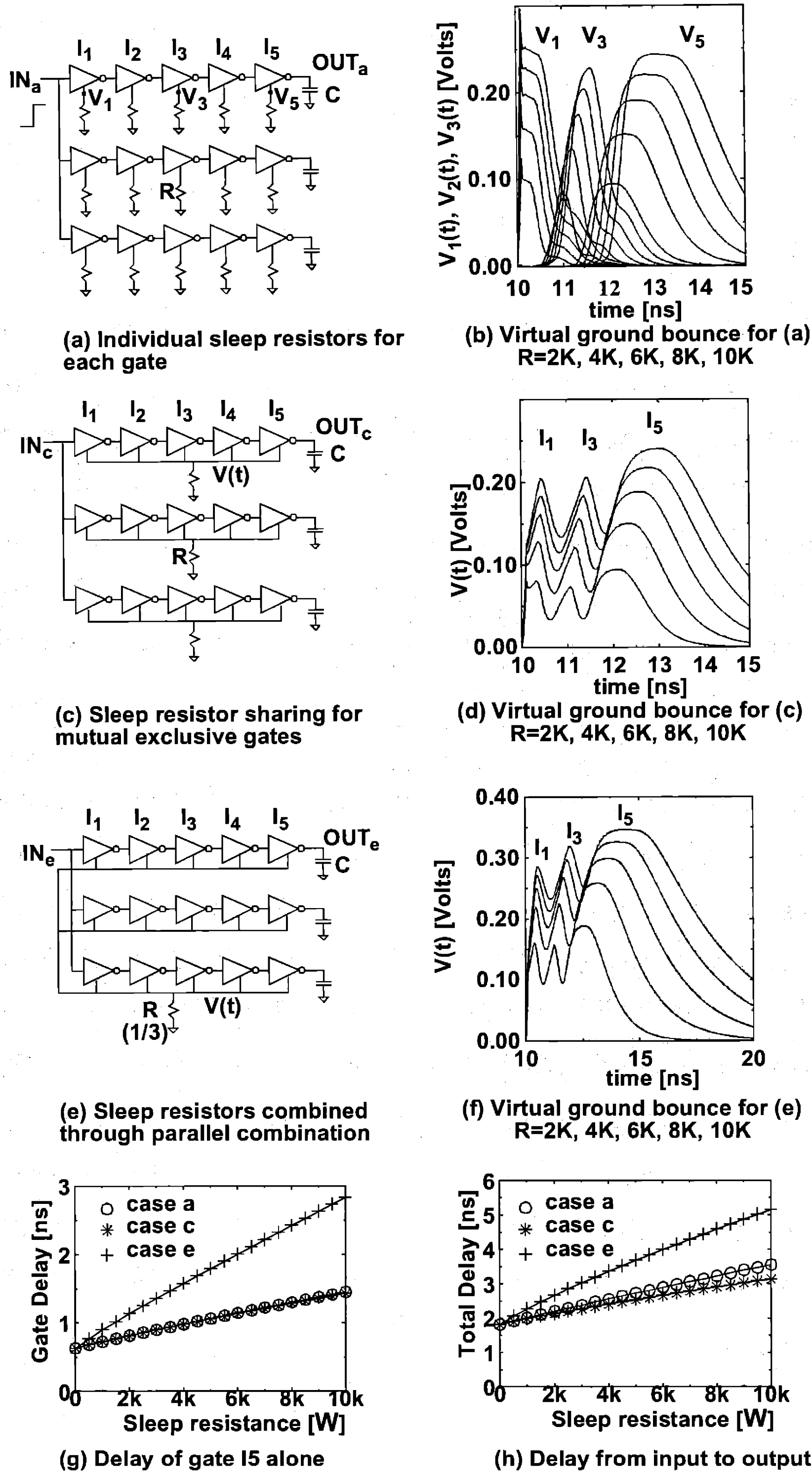


FIGURE 2-18. Inverter chain example showing the 3 steps for merging sleep resistors. Simulation parameters: V_{CC}=1.0v, V_t=0.2v, C=50fF, l_{min}=0.7μm.

Figure 2-18a shows the first step in the transistor sizing procedure, where individual sleep resistors (which model sleep transistors in the "on" state) are sized to ensure that no gate degrades by more than a fixed percentage. As can be seen in columns 2 and 3 of table x, the overall performance of the inverter chain will be satisfied if the internal gates meet the required speed. (i.e. the % delay in column 3 is always less than or equal to that of column2).

However, the overall delay of the series degenerated gates will be less than the individual gate delays because the low to high transitions of I_2 and I_4 are not degraded by the NMOS sleep transistor. Figure 2-18b shows how the virtual ground lines (V_1, V_3, and V_5) for this circuit will fluctuate as a result of a rising step function applied to the input.

2.9.1 Sleep transistor merging step

Although it is relatively simple to develop an MTCMOS sizing strategy by individually adding high V_t transistors to each gate in a circuit, this can result in large overestimates in sleep transistor area and large overheads in wiring area. However, since not all gates in the circuit will switch at the same times, it is possible to merge sleep transistors together from mutual exclusive gates and thereby reduce circuit complexity. For a set of n such gates with equivalent sleep resistances r_1, r_2,... r_n, the sleep resistors can be combined and replaced by a single $r_{eff} = \min(r_1, r_2, \ldots r_n)$. These mutually exclusive gates will discharge currents through the sleep transistor at different times so that the virtual ground bounce that each transitioning gate experiences will still be the same or smaller than before. As a result, the delay of each gate sharing the common sleep transistor should also be the same or smaller than in the original circuit. An added benefit of replacing n sleep resistors with a single one is that the subthreshold leakage current will decrease by a factor of n, and also the increased parasitic capacitance on the virtual ground line can improve performance.

Figure 2-18c shows how the original inverter tree's sleep resistors can be replaced by only 3 resistors by utilizing the same high V_t switch for mutual exclusive gates. Inverters I_1, I_2, I_3, I_4, and I_5 for example will never transition from high to low at the same times, and as a result can share a common sleep transistor. Examining the delay of inverter I_5 alone, one can see that both cases a and b have virtually identical delays, as

illustrated in Figure 2-18g. Furthermore, the total path delay in the merged scenario meets or exceeds the performance of the individually sized devices. The slight performance improvement seen in Figure 2-18h is due to the larger parasitic capacitance on the virtual ground line for the merged, which tends to low pass filter the virtual ground bounce. As a result, inverter I1 will be faster in the merged case because the virtual ground bounce rises more slowly. As the parasitic capacitances charges up through, later stage gates will not see these beneficial effects since the capacitance does not have time to discharge again.

2.9.2 Sleep transistor consolidation through parallel combination

Having separate sleep resistors for different groups of mutually exclusive gates can be cumbersome for the circuit layout. In many cases, it is possible to lump these sleep transistors together as a parallel combination, and performance will still be maintained. Although total transistor area will be the same, wiring and layout area can be reduced. To quantify this point, consider the circuit in.

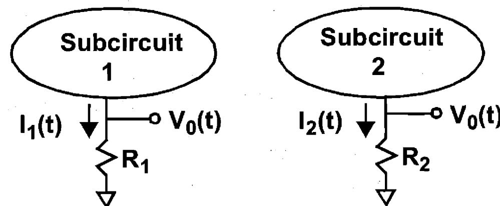


FIGURE 2-19. Circuit showing how sleep resistors can be combined in parallel.

If the virtual ground voltages for two different subcircuits is similar, then they can be modeled as two current sources, $i_1(t)$ and $i_2(t)$, connected to resistors r_1 and r_2 to give a voltage waveform $v_0(t)$ for both cases. However, if $i_1(t)$ and $i_2(t)$ are summed together and r_1 and r_2 are placed in parallel, then the new voltage over the resistor is:

$$v(t)= (i_1(t) + i_2(t)) * (r_1 // r_2)$$

$$= (i_1(t) * r_1 * r_2 + i_2(t) * r_1 * r_2) * (1 / r_1 + r_2)$$

$$= (v_0(t) * r_2 + v_0(t) * r_1) * (1 / r_1 + r_2)$$

$= v_0(t)$

which is the same as before. Thus, for two subcircuits with very similar virtual ground transient behaviors, combining the two systems together will result in unchanged virtual ground characteristics, so the overall performance should be unchanged. In general, if voltages $v_1(t)$ and $v_2(t)$ are very different, then the resistors should be combined such that $v(t)$ will not exceed the minimum of $v_1(t)$ or $v_2(t)$. In this case, $r_{eq} = \min(v_1(t), v_2(t)) / (v_1(t)/r_1 + v_2(t)/r_2)$.

In Figure 2-18e, the three separate sleep resistors from Figure 2-18c can be replaced by a single resistor with three times the conductance that now gates the entire circuit. Figure 2-18g and Figure 2-18h show comparisons of the delay vs. sleep resistor size for these two cases, and illustrates how the resistance must be lowered by one third in order to achieve the same performance. Another way to appreciate this relationship is to examine the virtual ground transient response shown in Figure 2-18d and Figure 2-18f. By scaling the resistance by 1/3 for the case with a single global sleep transistor, the virtual ground bounce shown in Figure 2-18f can be matched to the that of Figure 2-18g, which would give the same delay behavior.

In general, combining separate sleep transistors into a single common one will be beneficial. The increased parasitic capacitances will tend to speed up the circuit during the capacitor charging stage. More important, the worst case scenario where the subcircuits will all discharge simultaneously is not common. Because the larger resistances used in the original subcircuits are replaced by a smaller resistance applied to the combined circuit, in many cases individual gates will be faster than before. In some degenerate examples using pure parallel combination, it may be possible that two subcircuits with separate sleep transistors might have very different virtual ground transient responses. In such a case, combining sleep transistors by a simple parallel combination will speed up one case, but could possibly slow down the other (the one with a much smaller virtual ground bounce). However, this is most likely not going to affect the overall performance of the circuit as a whole.

2.10 Comparison with Optimal Sleep Transistor Size

As a concrete example, we simulated the MTCMOS inverter network where the sleep transistor was designed to provide only a 5% degradation in performance over a conventional CMOS implementation. By simulating a single inverter with a sleep resistor in SPICE, we discovered that a sleep transistor with an equivalent resistance of less than 340Ω was required for less than 5% individual degradation. When applied to the inverter chain network and merged together, the sleep transistor equivalent resistance was 113Ω, with a 3.13% degradation in delay. The predicted sleep transistor required was actually an overestimate, because direct simulation shows that one only needs a resistance of 180Ω in order to achieve a 5% degradation in performance. By using this transistor sizing methodology, the transistor width was overestimated by 60%. One major cause for this discrepancy is that in MTCMOS circuits with NMOS sleep transistors, typically only half the gates, those switching from high to low, are actually degraded. Thus even if the high to low transition degrades by 5%, the overall chain will degrade on average by only 2.5% if pulldown and pull up transitions are balanced. Although this inverter chain circuit is easy enough to size through brute simulation, the resistor synthesis approach can be applied to more complicated circuits where exhaustive simulation is not possible.

2.11 Sleep Transistor Sizing Algorithm

The previous example demonstrated how MTCMOS sleep transistors can be sized individually for each gate and then shared among mutually exclusive gates, where no two gates can be discharging current at the same time. The primary value of this technique is in the sleep transistor reduction step, because area of the sleep transistor is of primary concern in MTCMOS circuits. One approach to develop a mutual exclusive set of gates in a circuit is to use a criteria based on the structural interconnections in the network graph. Assuming a unit delay model for each gate, one can tabulate all the possible times that any particular gate can switch. Mutually exclusive gates can then be grouped together whenever there is no intersection between the corresponding sets of times. In order to minimize total sleep transistor sizes, the number of these groupings of mutually exclusive gates

should be minimized, and the sleep transistors chosen to be the largest transistors in each respective group.

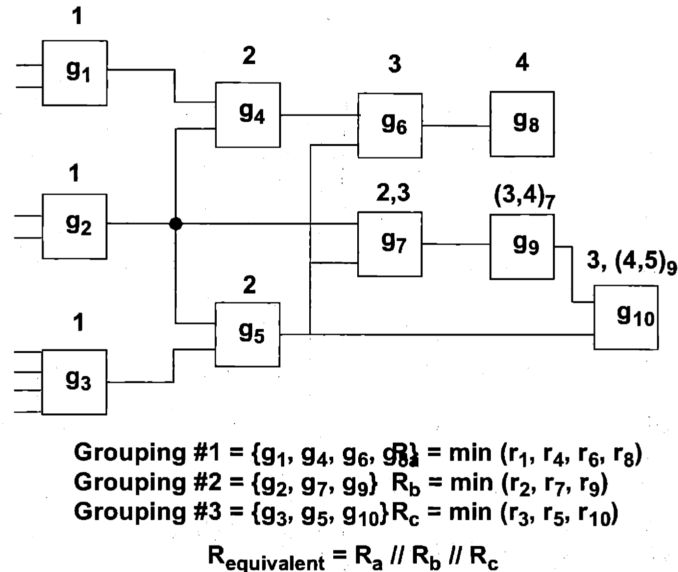


FIGURE 2-20. Logic gates annotated with all possible transition times, so that sleep resistors can be merged.

Figure 2-20 illustrates this procedure for a random logic circuit with arbitrary gate interconnections, where it is assumed that each gate has a corresponding sleep transistor (modeled as a resistor). Each gate is annotated using a unit delay model with all possible time slots that a transition can occur. Gates that do not have a time period in common will thus be mutually exclusive, and can be grouped together with a common sleep transistor. In cases where a gate can switch at multiple times, we further annotate the set of transition times by a subscript indicating the reference gate, because these two gates are also mutually exclusive even though they share a time slot. For example, gates g7 and g9 both show possible transitions at time 3, but this will never happen simultaneously because g9 is always one time unit behind g7. Ideally, the groupings should be selected to minimize the overall sleep transistor widths such that gates with very large sleep transistors should be lumped together.

This merging technique based on mutual exclusive gate discharge patterns is most effective for balanced circuits with minimal glitching. Fortunately, a large class of circuits

fall into this category, especially since less glitching is attractive from a low power point of view. For circuits with more complicated interconnections and glitching, the merging technique can still be used, although the compression ratio would probably be lower. To further improve the sleep transistor reduction, more rigorous CAD tools can be developed to determine mutual exclusivity that is based on logic rather than the structural connections in a circuit.

2.12 Hierarchical Sizing Methodology

Although the MTCMOS transistor sizing algorithm has been presented at the gate level, in fact it can be applied at many hierarchical levels of a circuit. The algorithm simply operates on generic circuit blocks that are elements within a larger module, and each block is assumed to have a local high V_t sleep transistor that is used for gating the power supply rails. The algorithm is applied to the network by combining the sleep transistors for mutual exclusive blocks. Thus, the blocks that the algorithm operates on can represent individual gates, cells within an array (like an adder cell in a multiplier), or even a module within a chip (like an ALU). In all these cases, a gating sleep transistor can be shared among several different blocks if those blocks have activity patterns that do not overlap in time.

In order to achieve the best results, one should initially use a detailed simulator like SPICE to simulate as large a block as possible and to exhaustively determine the optimal sleep transistor size. Next, the hierarchical merging technique can then be applied to these existing blocks to synthesize an overall sleep transistor for a larger module, where determining a worst case input vector would have been exceedingly difficult. Utilizing a hierarchical approach to sizing the sleep transistors is very attractive because detailed circuit complexity can be abstracted away at the expense of accuracy.

One limitation of sharing a single sleep transistor among several distinct blocks is that one must also take into account the increased interconnect resistance for blocks that are far away from the sleep transistor. As a result, one may need to size sleep transistors larger than expected to compensate for the added interconnect resistances and may also need to widen the virtual ground wires to maintain performance.

2.12.1 Parity checker example

As a practical example, the hierarchical sizing methodology was applied to a 32 bit MTC-MOS parity checker circuit. The circuit consists of 31 XOR gates which are connected as a tree with 5 levels.Figure 2-21 shows a smaller 8 bit version of this circuit.

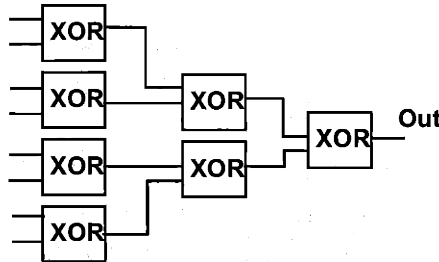


FIGURE 2-21. 8 bit parity checker

The XOR gate was simulated by itself to determine the local sleep resistance needed for a single gate to meet performance requirements. For 20% degradation, the sleep transistor needs a resistance less than 4800Ω, and for 10% degradation the resistance must be less than 2400Ω. With an application of the merging algorithm based on mutual exclusive discharging gates, the total number of sleep transistors required could be reduced from 31 (one for each gate) to only 16. The resulting sleep transistor for the entire 32 bit parity checker was then calculated to be less than 300Ω for 20% degradation and less than 150Ω for 10% degradation.

Since there are too many vector pairs (2^{64}) to test exhaustively, Table 2 below shows simulation results for a subset of 5 input vectors. Each of these vectors was chosen to exercise a critical path through the top row of the parity checker. Furthermore the critical 2-input XOR gates each transition with the worst case inputs (x=0->1,y=0->0).

TABLE 2-3. Parity generator performance as function of sleep transistor width for different input vectors

Input Vector	CMOS [ns]	R =150Ω [ns], %degr	R = 300Ω [ns], %degr
1	9.08ns	9.14ns, 0.7%	9.21ns, 1.4%
2	9.07ns	9.34ns, 3.0%	9.60ns, 5.5%

TABLE 2-3. Parity generator performance as function of sleep transistor width for different input vectors

Input Vector	CMOS [ns]	R =150Ω [ns], %degr	R = 300Ω [ns], %degr
3	9.07ns	9.46ns, 4.3%	9.87ns, 8.8%
4	9.08ns	9.44ns, 4.0%	9.81ns, 8.0%
5	9.08ns	9.34ns, 2.9%	9.60ns, 5.7%

The SPICE simulation shows how the sleep transistor sizes (150Ω and 300Ω) ensure performance within 10% (9.99ns) and 20% (10.90ns) of the CMOS critical delay of 9.08ns. Vector #1 does not cause large currents to flow in adjacent gates, so its degradation in performance is not large (0.7% and 1.4%). However, vector #3 creates significant currents through adjacent gates, and as a result is more susceptible to degradation (4.3% and 8.8%). In all cases however, the delays are significantly faster than predicted. Although there are other vector combinations that will result in larger delays, typically the sleep transistor sizing from the algorithm will still be a conservative overestimate of the required sleep transistor size. This is due mainly to three factors. First, only one half of the gates, those switching from high to low, are actually degraded as described in section 3.4. As a result, ensuring all NMOS transistors degrade by no more than 20%, will likely cause only a 10% degradation in overall performance. Second, our gate partitioning can be further improved by using more sophisticated algorithms to determine mutual exclusivity, as only a structural logic independent grouping algorithm was used. Finally the requirement that each gate's high to low transition degrade by no more than a fixed amount is overly stringent, and also contributes to a conservative estimate of sleep transistor size.

2.12.2 Wallace tree multiplier example

As another example, the hierarchical sizing methodology was applied to a 6x6 bit Wallace tree multiplier circuit shown in Figure 7. This is a type of circuit is well suited for this

algorithm because there are many mutually exclusive gates that cannot transition at the same time[29][30].

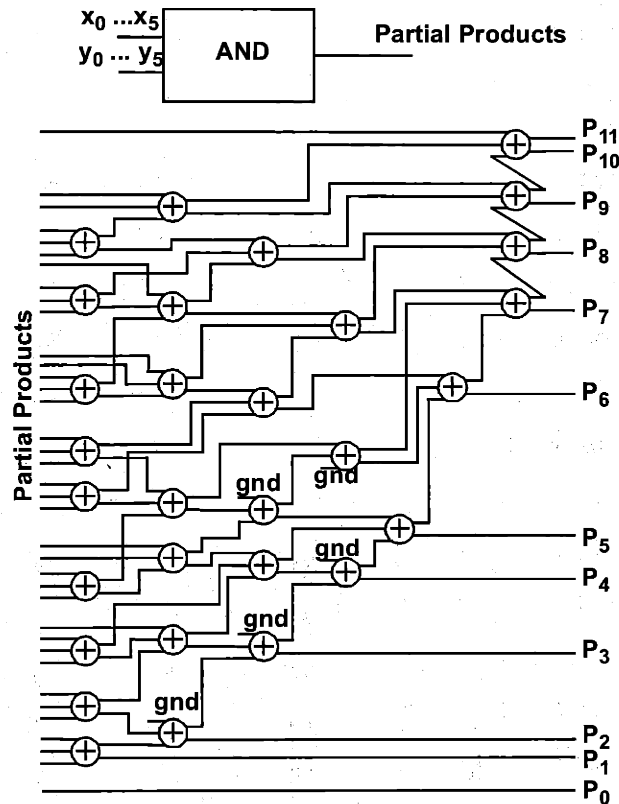


FIGURE 2-22. 6x6 Wallace Multiplier.

Initially, the AND gates and the carry save adder units (with representative loadings) were simulated in SPICE to determine optimal high V_t sleep transistor sizes (actually equivalent resistances) for each unit to give rise to a fixed degradation in performance. To achieve a degradation of 20% and 10%, the CSA required sleep transistors with equivalent resistances of 1600Ω and 800Ω, respectively. Likewise 20% and 10% degradation of the AND gates requires equivalent resistances of 2700Ω and 1350Ω, respectively.

Next, the sleep transistor reduction and merging steps were performed to give rise to an equivalent resistor that could gate the entire multiplier. By tabulating all possible time periods that each cell can transition, we were able to reduce the 36 AND cell and 30 adder cell sleep resistors into 21 AND cell and 15 adder cell sleep resistors. The total

equivalent resistance for the multiplier could then be written as (R_{add}/15) // (R_{and}/21), corresponding to 30Ω and 60Ω for 10% and 20% maximum degradation. The merged resistance is a factor of two greater than the case where no merging takes place, which would correspond to a factor of two decrease in sleep transistor sizing. The branches of this wallace tree structure were not completely balanced because adder cells at inner levels of the tree could actually receive inputs from two levels before. As a result, this implementation has fewer mutual exclusive gates because a fair amount of glitching can occur. Another implementation that balances the paths more carefully would result in larger compression results from the merging algorithm.

For a 6x6 bit multiplier, there are 2^{24} possible input vector pairs, so again it was not possible to exhaustively verify the circuit. However 6 representative vectors were simulated for output nodes P6 and P1 as shown in Table 2-4 and Table 2-5.

TABLE 2-4. Degradation of P6 delays in multiplier for different sleep resistances and vectors

Input Vector	CMOS [ns]	R =30Ω [ns], %degr	R = 60Ω [ns], %degr
1	8.79ns	9.01ns, 2.6%	9.26ns, 5.4%
2	8.46ns	8.87ns, 4.9%	9.16ns, 8.3%
3	8.72ns	8.92ns, 2.2%	9.111ns, 4.4%
4	11.17ns	11.28ns, 1.0%	11.40ns, 2.1%
5	12.31ns	12.43ns, 1.0%	12.76ns, 3.7%
6	6.55ns	6.79ns, 3.6%	7.07ns, 8.0%

TABLE 2-5. Degradation of P1 delays in multiplier for different sleep resistances and vectors

Input Vector	CMOS [ns]	R =30Ω [ns], %degr	R = 60Ω [ns], %degr
1	3.19ns	3.24ns, 1.9%	3.40ns, 6.8%
2	3.19ns	3.25ns, 2.1%	3.41ns, 6.9%
3	2.98ns	3.09ns, 3.7%	3.17ns, 6.8%
4	2.98ns	3.05ns, 2.3%	3.10ns, 3.8%

TABLE 2-5. Degradation of P1 delays in multiplier for different sleep resistances and vectors

Input Vector	CMOS [ns]	R =30Ω [ns], %degr	R = 60Ω [ns], %degr
5	2.98ns	3.12ns, 4.8%	3.21ns, 7.8%
6	3.16ns	3.31ns, 5.1%	3.46ns, 10.1%

By using the hierarchical sizing algorithm, the degradation in any path within the multiplier should degrade by no more than the nominal amount (10% and 20%). As shown in above tables, two very different paths, ending at P6 (which can include 6 cells), and P2 (which always includes 2 cells), are ensured to meet these performance requirements. Again, since only an NMOS sleep transistor was used, typically only 1/2 the transitions will actually be affected and total degradations will be limited to near 5% and 10%. Also, as described earlier, the restriction that all paths meet the same performance constraint will yield an overly conservative estimate of sleep transistor sizing. For example, an inherently slow path like P2 should be allowed to degrade more since it will unlikely be the worst case delay. By relaxing the degradation requirements for non critical gates, then the sleep transistor can be reduced in size. Nonetheless, the hierarchical sizing strategy provides at least an upper bound on the size of the sleep transistor needed to ensure performance.

Chapter 3

Dual V_t Domino Logic

MTCMOS circuits require the insertion of extra series high V_t devices which have no other purpose but to limit leakage currents during the standby mode. These sleep transistors are difficult to size correctly, and being in series with the pulldown/ pullup path will always degrade performance. Another style of dual threshold voltage design is embedded dual V_t logic, of which dual V_t domino is a special case. Dual V_t domino utilizes both high V_t and Low V_t transistors to create a circuit with extremely low leakage in the sleep mode, yet suffer no reduction in performance[31].

3.1 Embedded Dual V_t

In embedded dual V_t logic, high and low threshold voltages are assigned to the devices already existing within each logic gate, thereby eliminating the need for extra series switches. Potential leakage paths for each gate can be implemented with high V_t devices, and can be turned off with proper vector activation during the standby mode. This fine

grain application of dual V_t also is beneficial because high V_t device sizing is no longer impacted by the discharge pattern of other circuits, as was the case for MTCMOS.

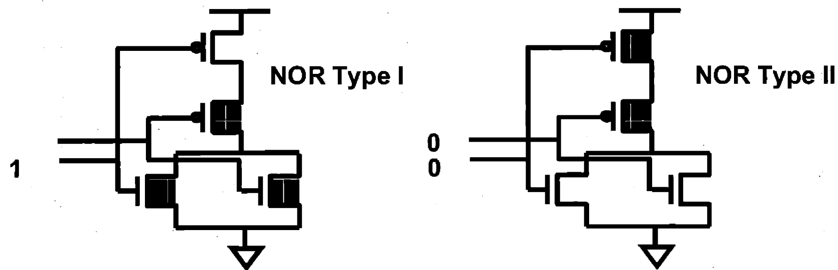


FIGURE 3-1. Embedded Dual V_t NOR gates with LVT devices shaded. Inputs are shown to strongly turn off HVT devices for low standby leakage operation.

Figure 3-1 shows two types of NOR gates in the embedded dual V_t circuit style, where existing devices are chosen to be high or low V_t, and gates can be placed in low leakage states. In the first NOR circuit, activating the gate with one input high will place it in a low leakage condition because of the off high V_t device. Likewise for the second NOR circuit, activating the gate with both inputs low will place that configuration in a low leakage state. Unlike MTCMOS techniques which have floating outputs during the sleep state, embedded dual V_t circuits will have actively driven outputs, which could make mixing different circuit types together more attractive.

The embedded dual V_t principle is straightforward to understand; one simply needs to ensure that for each leakage path from V_{CC} to ground, one must have an off high V_t device. For any arbitrary input vector, one can choose all the "off" devices to be high V_t, which will place that gate in a low leakage condition. Unlike the stack effect techniques presented by previous researchers to limit standby leakage currents, the flexibility of using embedded high V_t devices allows one to apply this technique arbitrarily to any standby vector choice because there is no need to ensure that source biasing occurs. The leakage reduction is purely a result of the imbedded high V_t devices turning off. In actuality, not all the off devices need to be high V_t either- series devices do not all have to be high V_t, as long as there is one high V_t "stopper" in each chain. Furthermore, during the standby state, internal nodes are still actively driven with this technique (and do not float),

so it possible to directly drive CMOS gates with logic blocks implemented using the imbedded dual V_t style.

The use of dual V_t devices in a gate tends to trade-off between speed and leakage currents. For example, if the dual V_t gate's low leakage condition yields a "low" output state, then any low-to-high transition will be slowed down (compared to an all low V_t implementation) because the transition must pass through at least one high V_t device. On the other hand, any high-to-low transition would only pass through low V_t devices, and thus would not be degraded. Unfortunately, this behavior is not exactly desirable because the transition when leaving a standby condition often times is critical. For example, a word line driver in a memory would spend most of its time in a waiting period (low leakage configuration), and would be asserted when a read or write on that line is required. However, the transition for the line assertion would be slow, and the transition back to the standby state would be fast. This presents a fundamental property of the embedded dual V_t approach. As a result, if one transition direction is faster than another for a gate, then one must preset the circuit into the correct output state (opposite state of the standby case) such that the transition will be fast and through low V_t devices. On the otherhand, MTCMOS does not suffer from the same constraint in skewing transition directions because the sleep state of the gate is not dependent on the gate input vectors. As a result, the sleep transistor can be placed on either the power supply or the ground line depending on whether the rising of falling transition of the gate is more critical.

To compensate for the reduced speed of high V_t devices in embedded dual V_t circuit techniques, one can upsize the high V_t devices to ensure performance as illustrated in Figure 3-2.

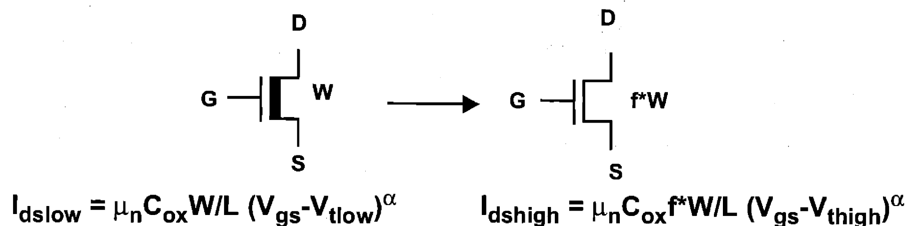


FIGURE 3-2. High V_t device must be upsized by a factor f to equate current drives.

To equilibrate the current drives of these two devices, the scaling factor f can be computed to be

$$f = \frac{(V_{CC} - V_{tlow})^{\alpha}}{(V_{CC} - V_{t(lowhigh)})^{\alpha}} \quad \text{(EQ 3-1)}$$

The drawbacks to upsizing the high V_t devices are twofold. First, the area is increased, which results in lower integration, and second, the upsized high V_t devices will cause loading problems for previous stages, requiring them to be driven using stronger devices or with a lower fanout. Because the high V_t devices are increased by a factor f to maintain gate performance, the leakage current of the high V_t device will also increase by the same factor. Nonetheless, the upsized high V_t device leakage currents are still much smaller than the low V_t device because of the exponential decrease in leakage.

3.2 Dual Threshold Voltage Domino Special Case

Dual threshold voltage domino logic is a special case of the embedded dual V_t principle, and provides the performance equivalent of a purely low V_t design with the standby leakage characteristic of a purely high V_t implementation. Because of the fixed transition directions in domino logic, it is very straightforward to uniformly configure the input vectors to place a domino logic block in a low leakage state, and furthermore because only one transition direction is critical in domino logic, one can configure the embedded dual V_t devices such that only non critical paths are high V_t. In effect, the dual V_t domino gate allows one to trade-off reduced precharge time for lower standby leakage currents. Dual V_t domino methodology utilizes low threshold voltages for all transistors that can switch during the evaluate mode and utilizes high threshold voltages for all transistors that can switch during the precharge modes. However, since precharge modes are not in the critical path, this can be performed more slowly without the need for upsizing the high V_t devices.

Figure 3-3 shows a typical dual V_t domino stage that can be used in a clock delayed timing scheme, which consists of a pull down network, inverter (I_1), leaker device (P_1), and clock drivers (I_2, I_3), with the low V_t devices shaded.

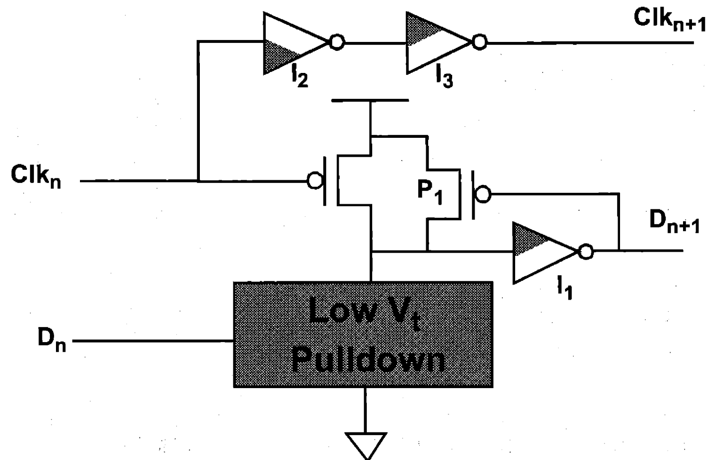


FIGURE 3-3. Dual V_t domino logic gate with low V_t devices shaded (clock delayed

3.2.1 Evaluate mode

Before the domino gate enters the evaluate stage, the internal node is precharged high, while D_n, D_{n+1}, Clk_n, and Clk_{n+1} are all low. When Clk_n goes from low to high and data arrives on D_n, the domino gate will quickly evaluate through the low V_t NMOS devices in the logic network and the low V_t PMOS of I_1. Likewise the rising Clk_n signal will also pass through I_2 (fast pull down) and I_3 (fast pull up) to supply the clocking signal to the next level of domino logic. The delay through I_2 and I_3 are matched to the delay through the logic and inverting stages such that the next data arrival is timed with the next evaluate clock. Finally, to maintain a high internal node voltage during evaluation, the P_1 transistor needs to supply enough current to satisfy the leakage from the low V_t NMOS block. The main benefit of this dual V_t domino approach however, is that during the evaluate phase, all transitions in the domino gate pass through low V_t devices.

3.2.2 Precharge mode

During precharge, the behavior of the circuit is the exact opposite, where the charging and discharging paths must pass through high V_t devices. By balancing the clock drivers I_2, I_3 with the precharge time and I_1 delay, the data zeroing and clock precharge signal for the next stage will also be closely aligned to avoid contention. Because high V_t devices perform the precharge functions, the precharge time is longer than for the case where all low V_t devices are used. This is acceptable because precharge time is not the critical transition in domino logic. During the precharge time, there will be significant leakage currents because only the low V_t devices will be off, but this active leakage current will be small compared to the switching energies required during operation. By precharging in the large leakage condition, the critical switching devices can then be fast by utilizing low V_t transistors.

3.2.3 Standby mode

During the standby mode, it is important to be able to place the domino logic gate in a low leakage condition. By forcing the clock to be high (evaluate), and forcing all inputs to the domino gate to be high, then all high V_t devices will strongly be turned off. This standby condition is unconventional because traditional domino logic design stalls the circuit in the precharge mode, while dual V_t domino must stall in the evaluate mode. This is a consequence of the behavior of imbedded dual V_t circuits- if one is stalled in the standby low leakage state, then the transition out of the leakage state must be slow.

Figure 3-4 shows the dual Vt domino gate placed in the standby mode where Clk_n and D_n are both driven high. As a result, the precharge PMOS device, the P_1 leakage

device, the I_2 PMOS, the I_3 NMOS, and the I_1 NMOS all are strongly turned off and leakage currents are significantly reduced.

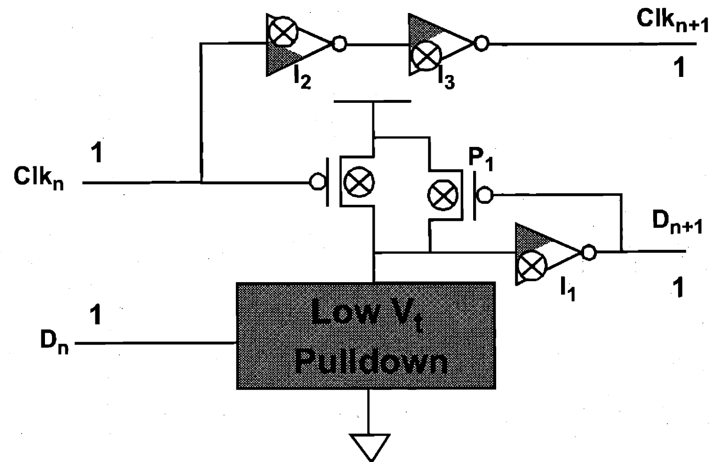


FIGURE 3-4. Dual V_t Domino gate in low leakage state.

It is very important to ensure that the input to the dual V_t domino gate is driven high to prevent the internal dynamic node from floating. A floating dynamic node can create large short circuit currents in inverter I_1. By driving the dynamic node to be a solid 0, the I_1 NMOS will drive a strong 1 on the output, which will be necessary to turn off the leaker device as well.

3.2.4 Clock delayed domino timing for single threshold voltage

The domino logic style shown in Figure 3-3 and Figure 3-4 is clock delayed domino logic, where the clock to downstream gates is delayed with the flow of data so that downstream foot switches can be removed for faster performance. This is illustrated in Figure 3-5 for a single V_t implementation showing a typical pipe stage that would be used in a two phase

clocking methodology. The pipe stage shows a logic depth of 8 gate delays, consisting of 4 domino gates and 4 inverters using a fixed V_t technology.

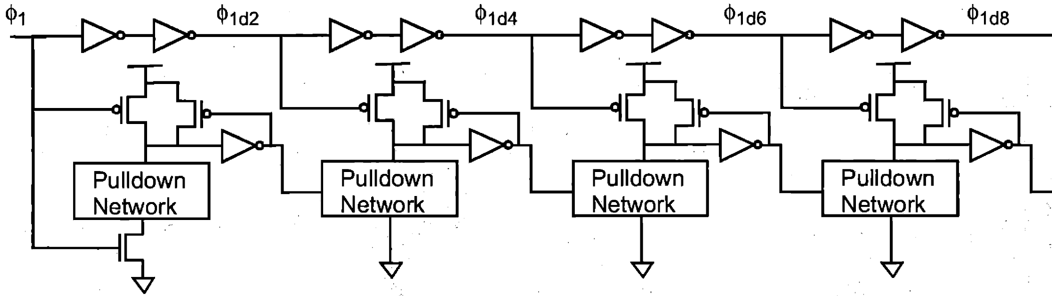


FIGURE 3-5. Pipe stage showing clock delayed domino logic functionality.

If the clock delay of each stage is matched to the domino gate delay, then the evaluate signal will arrive in conjunction with the gate input signal. Furthermore, the precharge signal will also arrive in conjunction with the input zeroing signal. As a result, this timing constraint eliminates the short circuit condition of footless domino where inputs can be high at the same time the precharge signal is asserted[32].

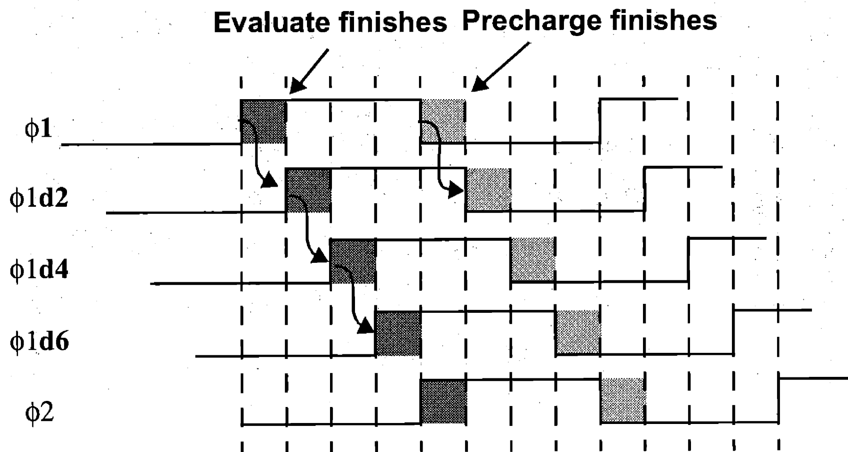


FIGURE 3-6. Clocking methodology showing evaluate and precharge times.

One of the drawbacks of this clock delayed methodology is that there is a tighter constraint in the precharge time for these circuits than for the traditional domino logic

style where all gates use a clocked footswitch. In traditional domino, all gates begin the precharge cycle at the same time, and thus can take the entire 1/2 cycle to precharge. On the otherhand in the clock delayed domino shown above with the clock buffers balanced, both evaluate and precharge for each gate must complete in the allotted 2 buffer delay slot because the precharge of each gate proceeds serially. In other words, the precharge requirement arises because gate N+1 can start precharging only after gate N has already preacharged and driven its outputs to 0. However, in the dual V_t clocked domino delay, the buffer delays are not uniform, which allows one to capture more time during the clock cycle to perform the precharge.

It is important to also retain the footswitch in the first stage of logic. This is because the phase1 logic block would most likely be driven by a phase 2 logic block, but the results from the phase 2 block could become valid when the G1 is still precharging. At very low clock frequencies for example, phase 2 signals can complete earlier, and would remain asserted on G1 for most of the clock cycle during which phase 1 clock is precharging.

3.2.5 Clock delayed domino timing for dual threshold voltages

The dual V_t clock delayed domino circuit structure is very similar to that of Figure 3-5 except that low threshold voltage and high threshold voltages are now used throughout the domino gate.

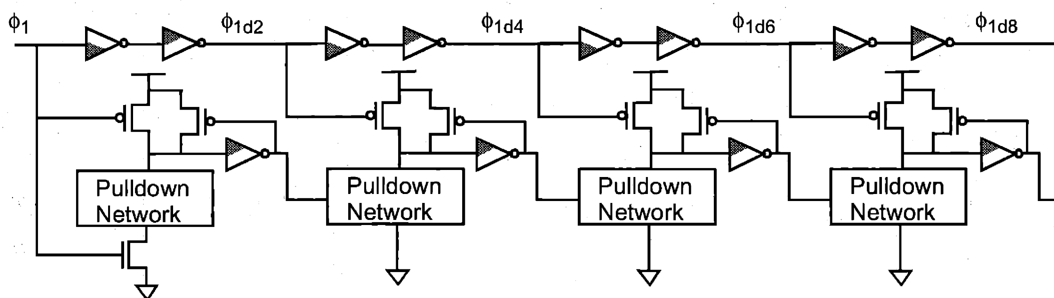


FIGURE 3-7. Dual Vt pipe stage showing clock delayed domino logic functionality.

The operation of the dual V_t clock delayed domino is more complicated than the single V_t case because non critical transitions are delayed while the critical transitions are kept fast. This results in a non uniform stretching of the clock pulses, to ensure matching of both evaluate and precharge clock signals with data propagation. This clock stretching is extremely important because it also provides a mechanism for increasing available precharge time, which makes it possible to use high V_t precharge devices. A timing diagram for this architecture is shown below.

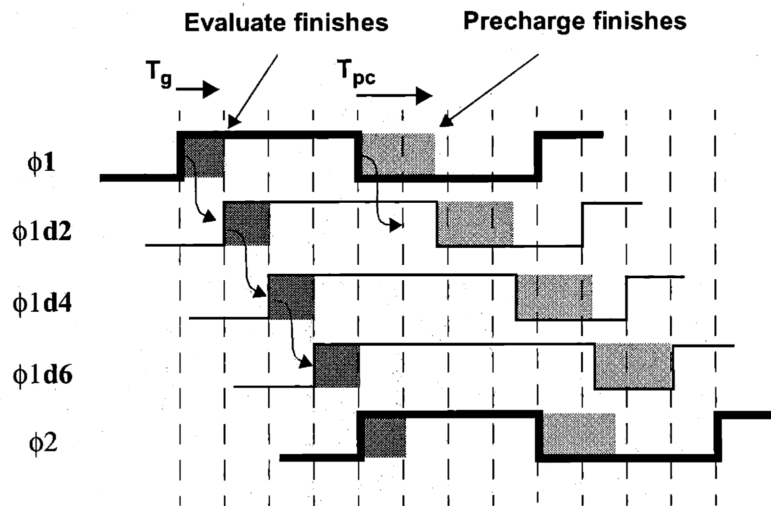


FIGURE 3-8. Clocking methodology showing evaluate and precharge times.

As can be seen in Figure 3-8 above, the falling edge of the clock for downstream gates continues to be slowed down through high V_t devices, while the rising edge of the clock (the evaluate signal) proceeds downstream through low V_t devices unimpeded. For each stage, the rising edge is delayed by an amount T_g, but the falling edge is delayed by an amount T_{pc}, resulting in the clock pulse stretching by T_{pc}-T_g for each stage. The slower high to low transitions thus provides a time T_{pc} to perform the precharge. For the above scenario of a two phase clocking methodology with 50% duty cycle and 4 logic gates per phase, then the amount of time available to perform a precharge is $T_{pc} = 7/4 * T_g$, where T_g is the delay per domino stage (dynamic gate plus inverter delay). In the case where the evaluate clocks are not 50% duty cycle, then even more time can be allotted for precharge. In general, if the total evaluation clock is $N*T_g$, the precharge clock is $M*T_g$

(so total clock period is $(N+M) * T_g$), then the time available per gate for precharge is $(M+N-1)T_g/N$. In the traditional domino style where all domino gates are clocked with the same clock (and NMOS evaluate switches are used to prevent contention), then the entire precharge clock can be utilized for each gate.

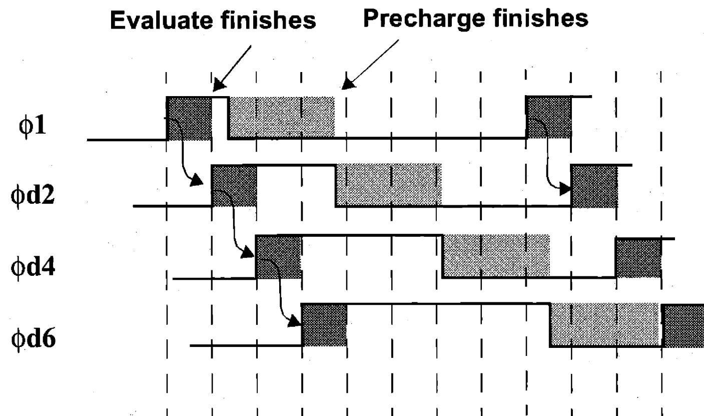


FIGURE 3-9. Clocking methodology showing evaluate and precharge times for non 50% duty cycle clocks.

3.3 Pipeline Standby Mode

Figure 3-10 illustrates how to place a more complicated datapath consisting of several pipeline stages into standby mode. The first step is for the control circuitry to finish computing any instruction in the pipeline so that no data is lost. Next, both phases of the domino pipeline are placed in sleep mode by gating the clocks to a logic "1" so that all gates are in the evaluate mode. Lastly, the first level inputs to the beginning of the pipeline must also be gated to a logic "1", which will cause all subsequent gates in the pipeline

to evaluate in a cascaded fashion. The resultant datapath will thus be in a low leakage state where all high V_t devices are strongly turned off.

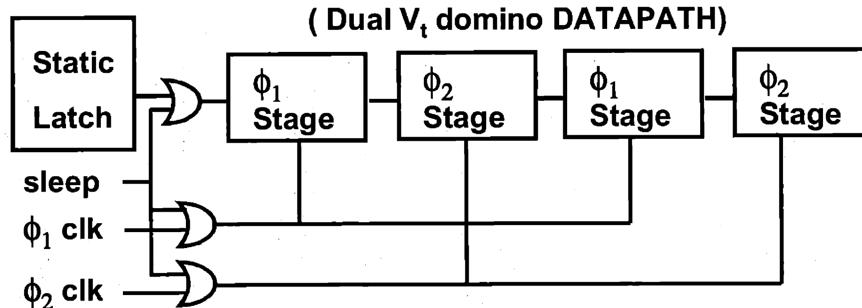


FIGURE 3-10. Pipeline sleep mode circuitry.

3.4 Simulation Results

To verify the functionality and benefit of dual V_t domino logic, simulations were performed on a representative pipeline stage modeled as an inverter chain with 4 domino NOR gates and 4 accompanying static inverters. The NOR gate has 8 inputs, each driving a fanout of 3 load. These wide gates are a good representative of domino circuits because domino technology is most effective for gates with wide, rather than deep, pull down networks. The experimental circuit has the exact same structure as shown in Figure 3-3 and figure xxx. Simulations were performed on three circuit variants with the exact same transistor sizings: an all low V_t design, an all high V_t design, and a dual V_t design. As predicted, the LVT delay is significantly faster than the HVT one. However, the DVT has a fast evaluate time on par with the all LVT design, but has a slow precharge delay on par with the all HVT design.

The performance benefit of low V_t domino is most apparent at lower voltages, where V_t is on par with V_{CC}. Figure 3-11 shows a comparison of LVT, HVT, and DVT delays as a function of the operating voltage, shown in the graph as a percent deviation from the nominal V_{CC} operating point. Clearly, the trend shows how LVT and DVT benefits are most effective at low voltages. For example, at -40% deviation (low V_{CC}), the reduction in delay over a HVT implementation is 44.5%, while it is only 24.1% at +20%

deviation (high V_{CC}). Interestingly, the DVT circuit delay is actually faster than the all LVT device in all cases, and this can be attributed to the fact that during switching, the pull down network has less leakage contention from the off PMOS device in the DVT case.

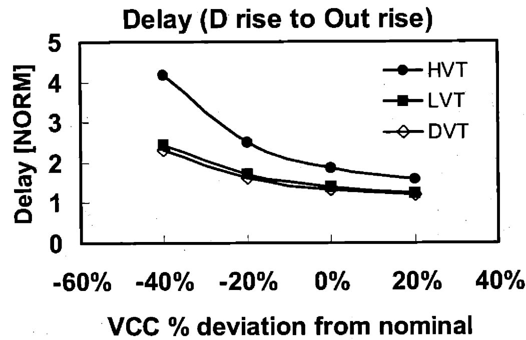


FIGURE 3-11. Evaluation delay through pipeline stage.

Figure 3-12 on the otherhand shows a plot of precharge delay as function of operating voltage. Precharge delay was measured as the delay between the falling Clk line at the input of the block to the falling edge (precharged state) of the final block output. As can be seen in the figure, the LVT implementation has a fast precharge delay, while the HVT and DVT circuits have virtually identical but much larger delay times. Again, since the precharge delay is not in the critical path, this will not effect the overall circuit speed.

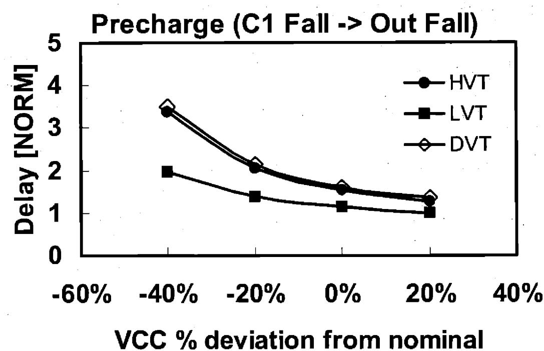


FIGURE 3-12. Precharge delay for pipeline stage.

Simulations were also performed to verify the leakage benefits in the DVT design. Two scenarios are explored: one where the circuit is stalled in the precharge mode with the

data input zeroed, while the second scenario is where the circuit is stalled in the evaluate mode with all data inputs activated. As described earlier, the proper DVT standby mode is the latter case.

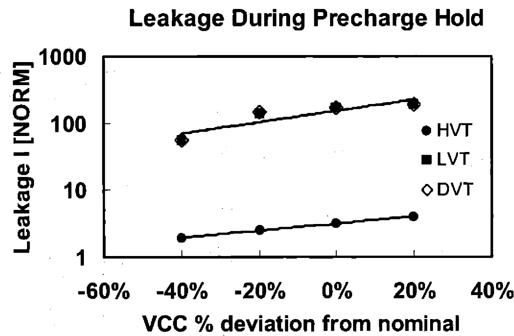


FIGURE 3-13. Leakage current for CLK=0.

Figure 3-13 and Figure 3-14 illustrate two different components of leakage reduction seen in the DVT standby mode case. First of all, by holding the circuit in the evaluate mode rather than the precharge mode, the leakage will be reduced because the leakage path in each gate is through a single off PMOS, rather than 8 off NMOS transistors in parallel. Thus leakage currents are reduced slightly in all three cases. For the DVT case, the greatest benefit of holding the circuit in the evaluate mode is that the leakage path will be through a high V_t PMOS device. As can be seen in Figure 3-14, the DVT implementation leakage is comparable to the leakage of the HVT implementation, both of which are an order of magnitude less than the LVT case. Another interesting phenomenon shown in the figures is the trend showing how low V_t device leakage increases more rapidly than high V_t devices with supply voltage. This scaling trend is due to worst short channel effect on the lower V_t devices, making their leakage more susceptible to supply voltage scaling

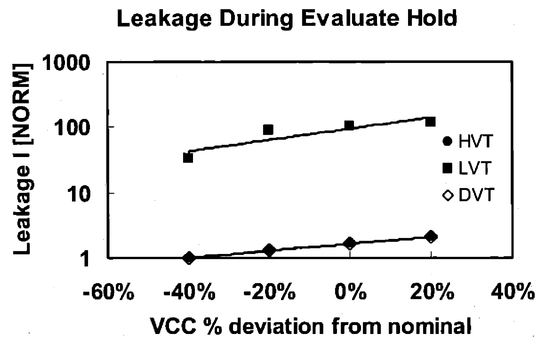


FIGURE 3-14. Leakage current for CLK=1.

3.5 Dual V_t Domino Logic Issues

Dynamic logic's superior performance over static CMOS can be directly attributed to it's lower noise margin. The trade off between higher performance and lower noise margin is fundamental in VLSI circuits. In domino circuits, the noise margin is directly related to the threshold voltage of the NMOS pull down tree, so there is definitely a limit to how low Vt's can scale. Furthermore, active leakage in large fan in gates, if large enough, can effect functionality when a domino gate tries to hold an internal node high. A large keeper device helps, but this will directly effect performance, and active leakage power dissipation still remains a problem.

However, research has shown that domino gates can be made to function at low voltages and low V_t's. With careful attention to noise, the use of keeper devices, and improved device characteristics, domino logic is still likely be used in future technologies. As long as low V_t and low V_{CC} dynamic logic can be made to work, then it will be beneficial to use the dual V_t domino methodology described in this paper. Although it has little effect on active leakage power, dual V_t domino significantly reduces standby leakage, which can play an important role in many applications where waiting times are long. Furthermore, switching to standby mode using this methodology has low overhead because one only needs to gate the clocks and then assert the initial inputs into the pipeline. As a result, this power down mode can also be effective at fine grain control such as for inactive modules within a chip like a multiplier or divider.

Chapter 4

MTCMOS Sequential Circuits

MTCMOS techniques are very effective at reducing standby subthreshold leakage currents in combinational logic blocks. However, these dual V_t techniques cannot be directly applied to sequential circuits that must retain data during standby modes. For example, an MTCMOS circuit that uses both PMOS and NMOS high V_t power switches will disconnect the internal nodes from the power supplies and cause them to float and corrupt standard memory circuits. In the case of a single polarity device, some gate output nodes driven to the opposite rail might be strongly driven, but other output nodes would still float. Imbedded dual V_t techniques are also not applicable to sequential circuits because these leakage reduction techniques rely on feeding a known input vector into each gate which is stalled in a predefined configuration. Sequential circuits, on the otherhand must be able to hold either high or low data during the standby state.

Memory circuits such as registers and latches are often times speed critical elements that make up a large fraction of the overall circuitry in modern digital designs, so it is important to utilize fast low V_t devices in these circuits, yet still reduce subthreshold leakage while retaining state during the standby mode. Preserving state during the standby mode is important because it allows circuits to be placed in a low leakage idle mode during the standby period, but can easily be woken up to finish computation during

the active period. Without this ability, only coarse level power down schemes are applicable, where a large block can be disconnected from power or ground, but data computation must be completely flushed out, and system state is saved in external circuitry. However, when MTCMOS circuits preserve state during the standby state, then core circuitry can be placed in low leakage modes at a much finer resolution both spatially and temporally. By combining clock gating and MTCMOS sleep transistor gating techniques, both dynamic and subthreshold leakage currents can be reduced whenever a circuit block is idle. For example, if a circuit block has not been used very often during a microprocessor algorithm, or the system is interfacing with a slow input source (keyboard), then the appropriate circuitry can be placed in standby and can easily be turned back on when necessary. For some architectures, it might be possible to stall a pipeline during the middle of a computation, and to resume where it left off after the system is activated again. However, in order to seamlessly transition between standby and active modes at such a fine grain resolution it is essential to efficiently preserve state in MTCMOS memory elements. Even for standby schemes that operate at a higher hierarchical level, (for example computations must be completely finished or flushed out before a block is turned off), there is still need for static storage elements that save the state of the system during the idle mode. As a result, efficient MTCMOS sequential circuits that retain state during the standby condition are vitally important circuit components necessary for low power systems.

4.1 Previous MTCMOS Sequential Circuits

Several MTCMOS sequential circuits that can preserve state during the standby mode have been proposed in the literature. One of the first and most straightforward ways to preserve state is to utilize parallel high V_t CMOS inverters to provide a static recirculation

path during the standby state. Figure 4-1 below shows an MTCMOS latch that was presented in [9], which can retain state during the standby state.

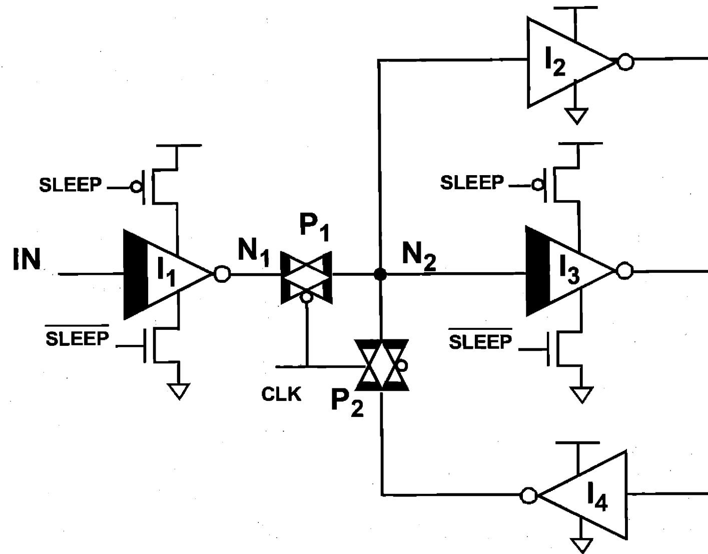
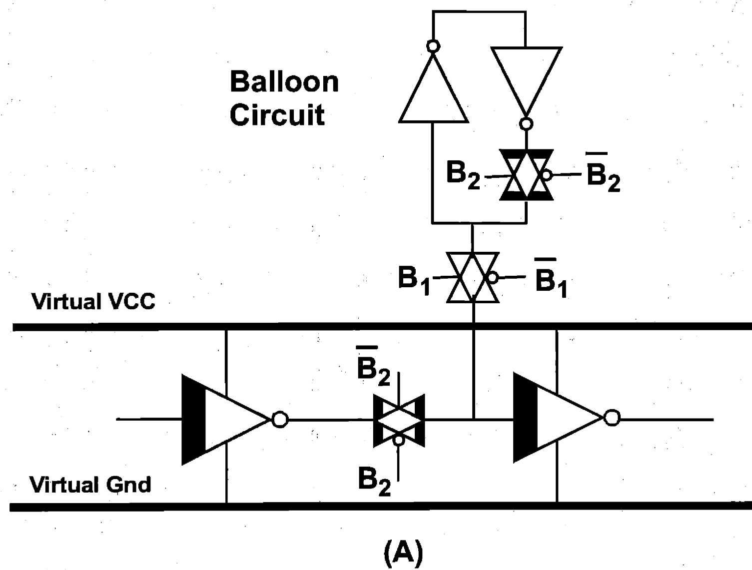


FIGURE 4-1. Conventional MTCMOS latch.

During the active period, the above latch functions like an ordinary CMOS latch. The high V_t power switches are turned on, and the latch is transparent when clock is low, and is opaque when the clock is high. During the standby mode, CLK is kept low and high V_t inverters I_2 and I_4 are used to hold the state of the latch even though I_1 and I_3 are disconnected from the power supply.

In [9], the latch was shown having both transmission gates low V_t, while the authors later modify the latch in[33] and assert that both transmission gates must actually be high V_t to "eliminate leakage currents". However, it turns out that the low V_t transmission gate implementation can eliminate leakage currents if the proper sleep configuration is used. However, a latch implementation that utilizes low V_t critical path passgates and high V_t recirculation passgates is actually better than both previous implementations, and is described in more detail later in the chapter. Another area where the authors seem to have made a mistake with this latch is their analysis of sneak leakage paths. This chapter will also provide a more thorough analysis of sneak leakage paths and techniques to eliminate them.

Another MTCMOS sequential circuit that holds state during the idle modes are "balloon" circuits described in [33]. Instead of using parallel high V_t inverters to maintain recirculation paths during the sleep state, this approach uses a completely autonomous balloon circuit that is used to explicitly write in stored data during the standby state, and can be read out when returning to the active mode. These balloon circuits can be made minimum sized because they simply hold data and do not need to be fast. The other benefit is that all MTCMOS gates can share common virtual V_{CC} and virtual ground lines since MTMOCS gates are completely decoupled from the high V_t CMOS balloon elements. The function of an MTCMOS balloon circuit is shown below.



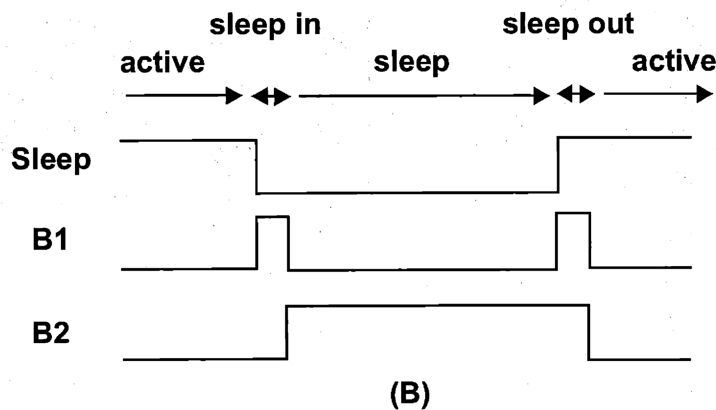


FIGURE 4-2. MTCMOS balloon circuit schematic (A) and control signals (B).

The balloon circuit operates in four distinct phases. During the active mode, the balloon circuit is disconnected from the internal MTCMOS logic through a high V_t pass-gate. During the sleep-in stage, the data on the internal MTCMOS node is stored into the balloon circuit. During the sleep state, the balloon circuit is again disconnected from the MTCMOS logic, and the data is recirculated in the high V_t balloon circuit. Finally, during the sleep-out state, the MTCMOS logic path is broken and the stored data is written into the MTCMOS node. Returning back to the active mode completes the cycle. An edge triggered flip flop that utilizes a balloon storage mechanism is shown below.

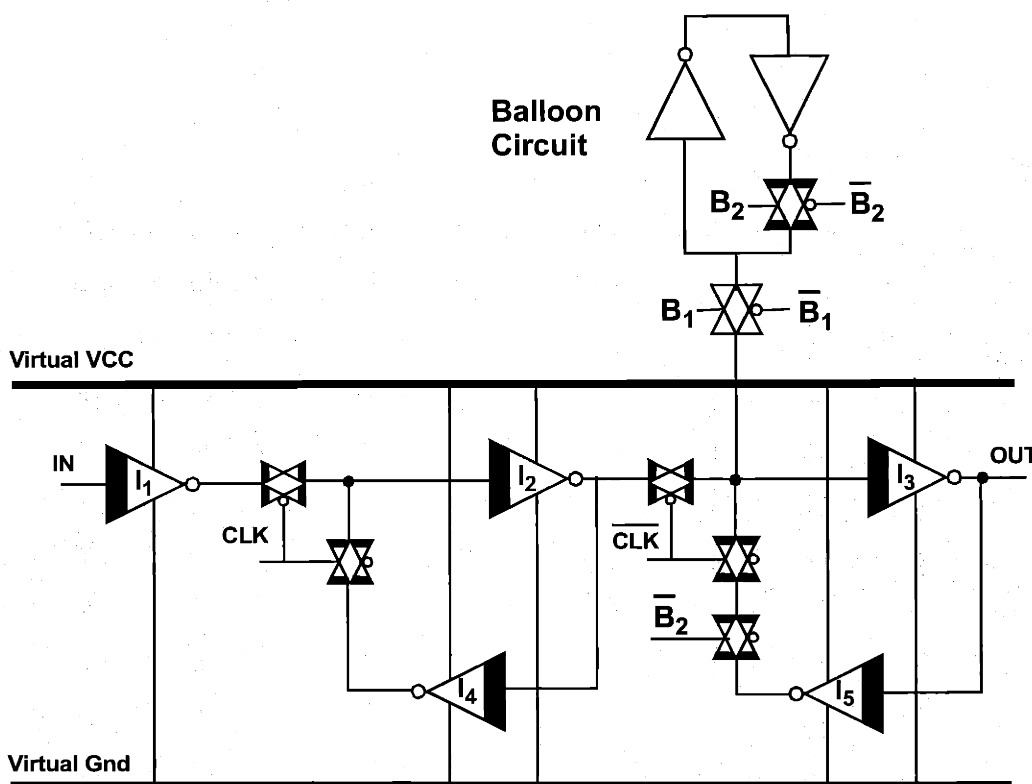


FIGURE 4-3. MTCMOS balloon circuit schematic applied to master slave D flip flop.

Although the basic operation is theoretically simple, the balloon circuit approach still requires a complex timing methodology and redundant circuitry that must be used for each flip flop. This extra circuitry and complex control is a necessary trade-off that one must accept in order to provide a clean abstraction between high V_t balloon circuits and

MTCMOS logic blocks, and to ensure that any interactions between the high V_t and low V_t circuits are decoupled. Routing the extra control signals throughout the chip to each flip flop can also be costly in a large design. As a result, balloon circuits, while theoretically attractive, can be difficult to implement in a practical circuit. Other MTCMOS sequential circuit schemes that are more area efficient and are simpler to control are explored in this chapter.

Because of the control and area costs of previous sequential circuits, much research has been done to seek alternative solutions. Work in [34] used an interesting periodic refresh mechanism to save state, but suffers from extra overhead and potential noise issues. Two other methods of state retention in MTCMOS blocks have been previously proposed in the literature in [35] and [36]. In these approaches, the MTCMOS structure is modified with diode devices such that during the sleep mode, internal nodes do not float. As a result, logic gates continue to operate on a reduced voltage swing, and leakage currents are still reduced. However, the leakage reduction amounts will not be as great as in a conventional MTCMOS circuit, and there may be robustness issues to insure functionality in a complicated circuit. Although clever circuit techniques have been explored to maintain state during the standby mode, there is still a need for robust, fast, and efficient solutions.

4.2 Improved MTCMOS Latch

In this section, an improved version of the conventional MTCMOS latch (using both high and low V_t transmission gates) is explored more thoroughly. This analysis illustrates some of the circuit issues involved with combining high V_t and low V_t devices, and also shows how sneak leakage paths can effectively be eliminated during the standby state. The latch is basically the same as the previous circuits described in [9] and [33] except

that passgate P_1 is made low V_t while passgate P_2 is made high V_t. As before, local sleep high V_t sleep transistors of both polarity types are necessary to eliminate leakage currents.

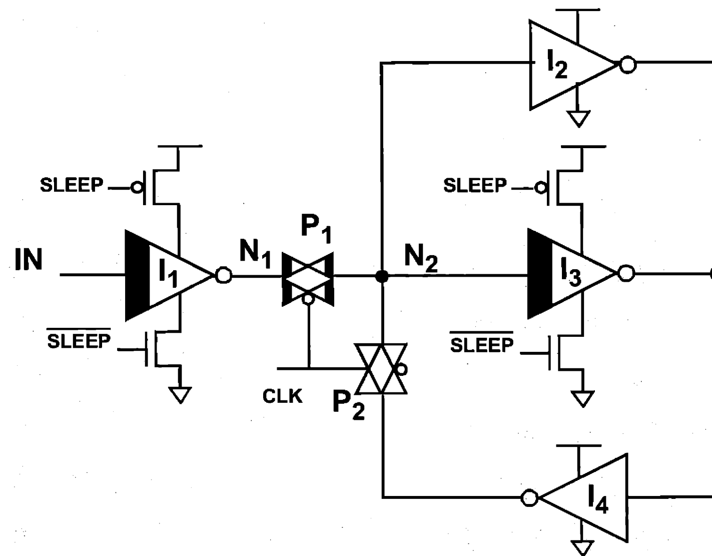


FIGURE 4-4. MTCMOS latch that holds state during sleep.

Inverters I_2 and I_4 are used to hold the state of the latch during the standby mode even though I_1 and I_3 are disconnected from the power supply. The input to the latch is also buffered through an MTCMOS inverter I_1 to provide a robust interface to outside circuitry. This is important because a noisy signal directly driving through a low V_t passgate could cause feedthrough errors.

4.2.1 Active state operation for the MTCMOS latch

During the active period, the MTCMOS latch functions like an ordinary latch. However, because subthreshold currents can be very high, one must be careful to properly size devices to ensure proper functionality. This is especially important when using low V_t passgate devices in combination with high V_t gates because large leakage currents can arise, which can cause strong leakage currents to fight with weak active high threshold devices. Because different components of the latch are implemented with devices of varying strength and leakage characteristics, careful simulations are imperative to ensure circuit robustness.

During the transparent latch operation period, the high V_t power switches are turned on and the clock is low. The critical transitions occur through low V_t gates I_1, P_1, and I_3 to maintain high performance. Furthermore, passgate P_2 is implemented with high V_t devices so that during the transparent mode, I_4 is strongly decoupled from I_1, which actually tends to improve the latch performance. If P_2 were implemented with low V_t devices, then the feedthrough delay would be slowed down because I_1 would have to fight the leakage of P_2 during the transient switching until I_1 and I_4 both flip.

For the latch hold condition during the active state, the low V_t passgate P_1 is turned off while P_2 is kept on to hold the latch data. Unfortunately, this configuration may result in large leakage currents if nodes N_1 and N_2 are driven to opposite polarities because P_1 cannot be turned off strongly. However, these leakage currents are no larger than the typical leakage for MTCMOS inverters during the active mode, so the energy overhead is small compared to the dynamic power dissipation. On the otherhand, the fact that nodes N_1 and N_2 can be driven to opposite rails and separated through a low V_t passgate P_1 is potentially dangerous because devices must be sized properly to ensure that noise margins are maintained. This arises because the logic level at N_2 is being driven through a relatively weak high V_t path (through P_2 and I_4 which traditionally can be made minimum sized), yet it needs to fight against a relatively large leakage current. If on the otherhand only low V_t devices were sued, this would not be a problem because the increased off currents would be countered by increased on currents as well. For the dual V_t latch however, it is important to correctly size the high V_t passgate P_2 and high V_t inverter I_4 large enough such that it can maintain a large enough drive to keep node N_2 at the proper logic level. At a minimum, it must be sized large enough such that the DC output voltage falls within the noise margin tolerance of low V_t device I_3. Although it can be further upsized to provide more noise margin, I_4 and P_2 need not have fast transient behavior because the recirculation path merely holds rather than switches the logic state.

4.2.2 Standby state

During the standby mode, the clock should be high so that the latch is in the holding configuration, and high V_t sleep transistors should be turned off, thereby disconnecting the low V_t devices from the power supplies. The recirculation path between I_2, I_4 and P_2

remain active, holding the state of the latch during the idle period. Again, since the recirculation path is not responsible for signal switching, it can be slow and implemented with high V_t devices.

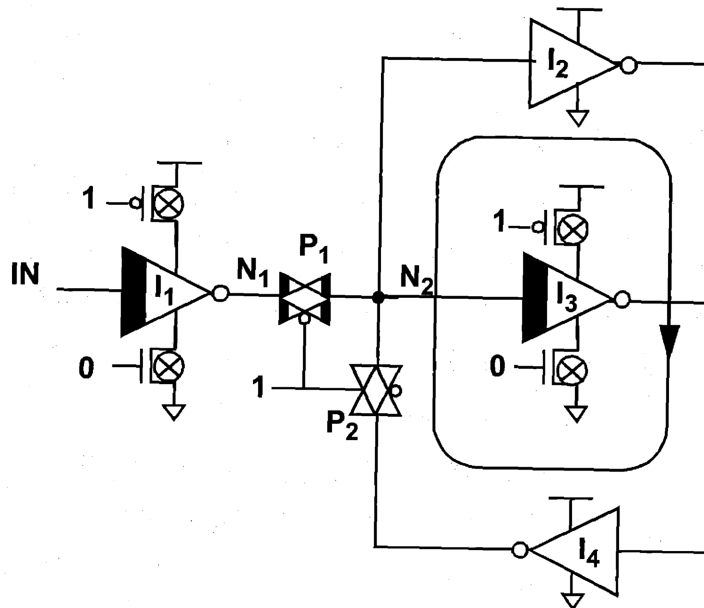


FIGURE 4-5. Latch during standby mode with data retention activated.

In the standby state, all paths from V_{CC} to ground encounter a high V_t off device. The problem with excessive leakage in the holding state during the active mode is eliminated in the sleep state because inverter I_1 is disconnected from both power supplies through high V_t off devices. By using local sleep devices of both polarities, all sneak leakage paths, which are described in more detail below, can be eliminated. However the cost of eliminating these sneak leakage paths is that the area overhead in this latch circuit are large. The MTCMOS gates in this circuit cannot share high Vt sleep transistors (like in combinational MTCMOS blocks) and both polarity devices are required.

4.3 Sneak Paths in MTCMOS Sequential Circuits

One of the problems with sequential circuits that utilize feedback and parallel devices is that sneak leakage paths may exist during the standby state if one is not careful. The previous latch is a good example of how local high V_t sleep transistors of both polarities were

needed for each gate in order to eliminate sneak leakage currents during the standby condition. Sneak leakage paths can typically arise whenever the output of an MTCMOS gate is electrically connected to the output of a CMOS gate because these configurations can provide sneak paths from V_{CC} to ground that may bypass the off high V_t devices that are supposed to disconnect the MTCMOS circuit block from the power supplies.

For combinational logic blocks implemented with only MTCMOS gates, a single high V_t switch (either PMOS or NMOS) is sufficient to eliminate subthreshold leakage currents during the standby state. No sneak leakage paths exist in these configurations because every path from V_{CC} to ground must pass through an off high V_t device. However, when CMOS gates and MTCMOS gates are combined together in sequential circuits, sneak leakage paths can arise that bypass the off high V_t devices. By carefully understanding the different types of sneak leakage path mechanisms that can arise in MTCMOS circuits, one can develop new circuit architectures that eliminate these sneak paths and yet can be implemented using shared sleep transistors to save area.

4.3.1 Sneak leakage paths due to CMOS-MTCMOS parallel combinations

Figure 4-6 shows a simple configuration where sneak leakage paths may arise when a single polarity device is used.

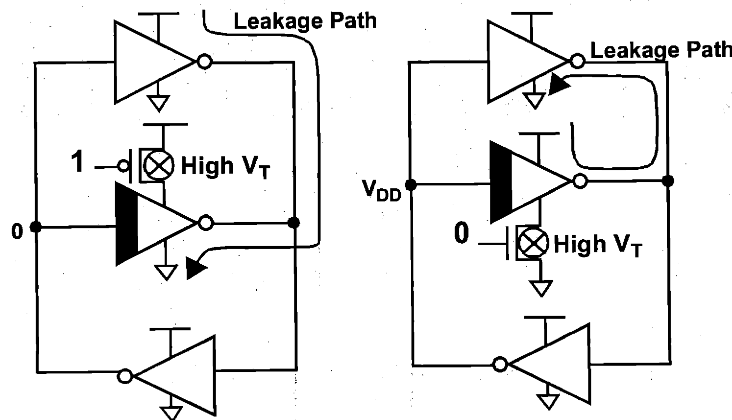


FIGURE 4-6. Leakage paths when only one polarity sleep transistor is used.

In this configuration, a CMOS gate is placed in parallel with an MTCMOS gate and a sneak leakage path can arise which is restricted only by an off low V_t device. An easy way to control this type of sneak path is to simply ensure that both polarity high V_t sleep devices are used for any MTCMOS gates that share an output with a parallel CMOS gate.

4.3.2 Sneak leakage paths due to CMOS-MTCMOS connection through low V_t passgates

Another mechanism for sneak leakage paths during the standby mode is when the output of a CMOS gate can be electrically connected to the output of a MTCMOS gate through low V_t passgate devices. For example, Figure 4-7 shows a sneak leakage path that again may arise if only one polarity sleep device is used.

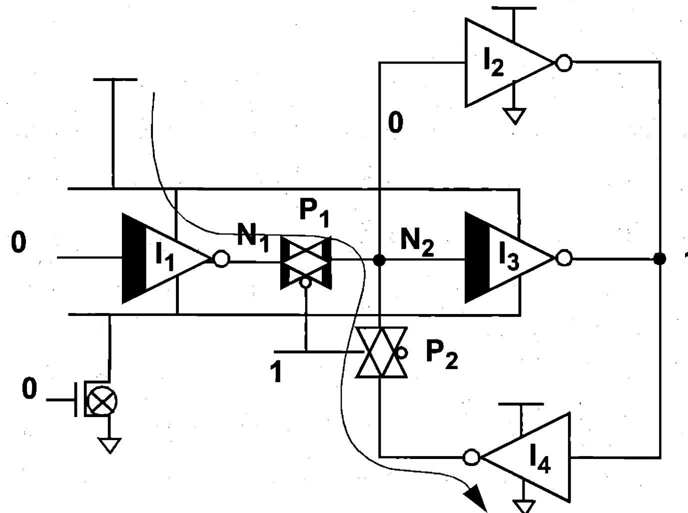


FIGURE 4-7. Leakage paths when only one polarity sleep transistor is used.

When the latch input is driven low, and the latch output is holding a "1" in the standby state, nodes N_1 and N_2 will be strongly driven to opposite polarities, and because passgate P_1 cannot turn off strongly, it will result in large leakage currents. Even if the input to the latch were to go high, there would still be a reasonably large leakage path that exists from the turned off low V_t PMOS of I_1 in series with the low V_t passgate P_1. Because the leakage path will be through two series off devices, there will be a reduction

in leakage currents, but it still can be an order of magnitude larger than necessary. Adding both polarity sleep devices on inverter I_1 will eliminate this path completely.

In the dual case where the latch input is driven high, the output is holding a "0," and a PMOS sleep device is used, then again N_1 and N_2 will be driven to opposite polarities, resulting in large leakage currents during sleep as shown in Figure 4-8. It is interesting to note that for those node data configurations that cause leakage paths through the low V_t passgate, the sneak leakage path associated with the parallel combination of CMOS I_2 and MTCMOS gate I_3 is eliminated. This illustrates how sneak leakage currents are highly dependent on the state of MTCMOS sequential circuits, so circuits must be carefully analyzed for all cases to ensure no leakage paths exist.

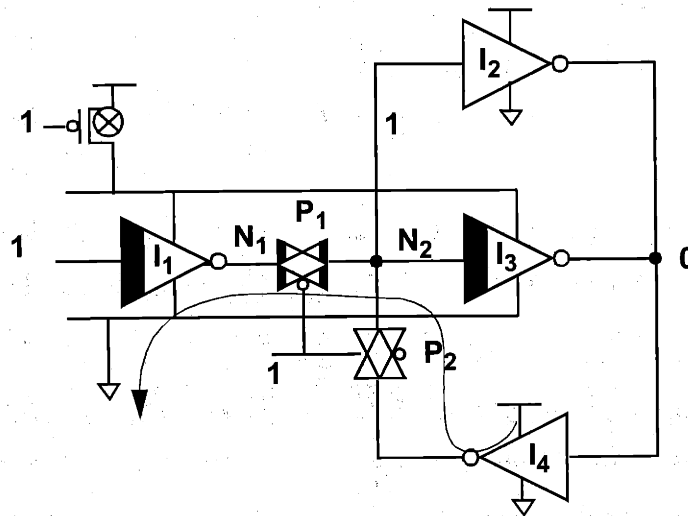


FIGURE 4-8. Leakage paths during standby state when only one polarity sleep transistor is used.

4.3.3 Sneak leakage paths due to CMOS-MTCMOS reverse conduction paths

Two straightforward sources of sneak leakage current paths were illustrated above and can be eliminated by using both polarity sleep devices. However, in some cases, simply using both polarity sleep devices is not enough. If high V_t sleep devices are shared among multiple MTCMOS gates, then there exists the possibility of sneak leakage paths that arise due to reverse conduction paths. These scenarios can arise if there are several MTCMOS-CMOS gate pairs that have a common output node, or are connected with low V_t pass-

gates, and share common virtual V_{CC} or virtual Ground lines. An example of such a sneak path is shown below in Figure 4-9.

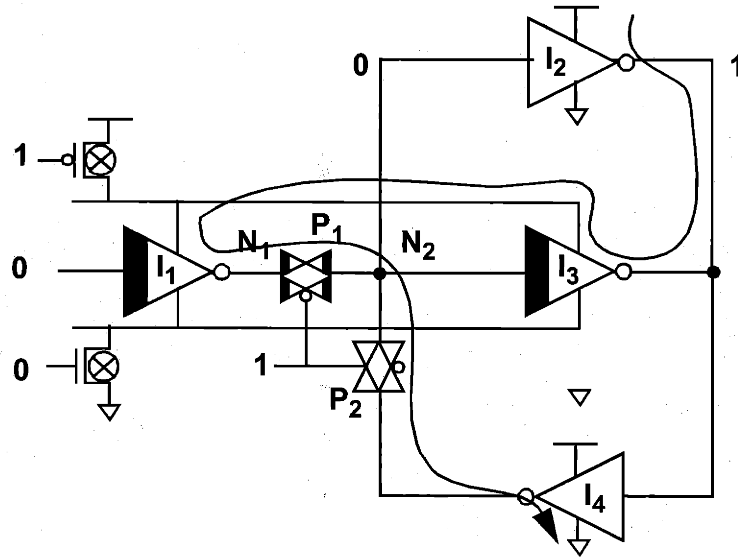


FIGURE 4-9. Leakage paths during standby state due to reverse conduction sneak paths.

The reverse conduction sneak path originates from V_{CC} of one of the CMOS gates, travels through the virtual power lines and exits to GND through another CMOS gate. The flow of current thus travels through an MTCMOS gate in a reverse conduction path, i.e. a PMOS current that flows from the device output up towards virtual V_{CC}, or a NMOS current that flows from virtual GND up to the device output. For the sneak path shown in Figure 4-9, the only impediment to leakage current flow from V_{CC} to ground is through the turned-off low V_t passgate P_1. All other devices in the leakage path are strongly turned on: the PMOS of inverter I_2 is on, the PMOS of I_3 is on, the PMOS of I_1 is on, passgate P_2 is transparent, and the NMOS of I_4 is on. As a result, the leakage from this reverse conduction sneak path can be quite large. Even if the input to I_1 were driven with a logic high instead, the leakage path would be limited by the low V_t series combination of I_1 and P_1 which is still much higher than the leakage through an off high V_t device.

In general, one can eliminate this type of sneak leakage path for a block with common virtual power or virtual ground lines by ensuring that no more than one CMOS-MTCMOS gate pair has a common output node. This is because the reverse conduction

sneak leakage path must travel from V_{CC} of one CMOS gate to the ground terminal of another CMOS gate, as illustrated in Figure 4-9. Another way to eliminate this leakage path is to use separate local high V_t sleep devices for those MTCMOS gate with outputs that are electrically connected to CMOS outputs. By doing this, one can break the reverse conduction paths because virtual V_{CC} and virtual ground lines are no longer shared among these MTCMOS gates.

4.3.4 Techniques to eliminate sneak leakage paths

The root cause for the sneak leakage paths described in the previous sections were due to the fact that MTCMOS gate outputs were somehow connected to CMOS outputs either directly or indirectly through low V_t passgates. However, by using both polarity sleep devices and local power switches, these leakage paths can be completely eliminated during the standby states. The drawback however is that local power switches of both polarities require more area because sleep devices are not shared among multiple blocks like in MTCMOS combinational logic. However, this penalty is not too severe because having local control of sleep devices makes it easier to size the sleep transistors themselves, and also decouples noise from different switching blocks from sensitive storage nodes.

Another way to tackle these sneak leakage paths is to explore different circuit architectures that minimize connections where CMOS outputs can leak into MTCMOS outputs. If an MTCMOS block can be designed such that no output is connected through a leakage path to the output of a CMOS gate, then it is sufficient to connect all these MTCMOS gates to a common virtual ground or virtual V_{CC} line in order to eliminate subthreshold leakage currents. In fact, only one polarity sleep device is necessary to lower subthreshold leakage currents by several orders of magnitude (though incrementally more leakage reduction could be achieved if both polarity sleep devices are used to turn off both supplies).

The "balloon" flip flop described earlier and first presented in [33] are good examples of how sneak leakage paths can be avoided by completely disconnecting the high V_t CMOS gates used for state retention from the MTCMOS blocks with a high V_t passgate. When the high V_t passgate is turned off, the CMOS supplies cannot leak into the MTC-

MOS blocks. As described earlier, the drawback of using this clean and robust interface is the need for more circuitry and more complex control signals. Another circuit architecture that can lead to a reduction in the number of potential sneak leakage paths is shown below in Figure 4-10. This circuit is a simple modification to the MTCMOS latch of Figure 4-4, which attempts to decouple the MTCMOS gates from CMOS high V_t inverters. This enables the modified MTCMOS latch to partially utilize virtual V_{CC} and virtual ground lines by sharing high V_t sleep devices with other MTCMOS blocks as shown below

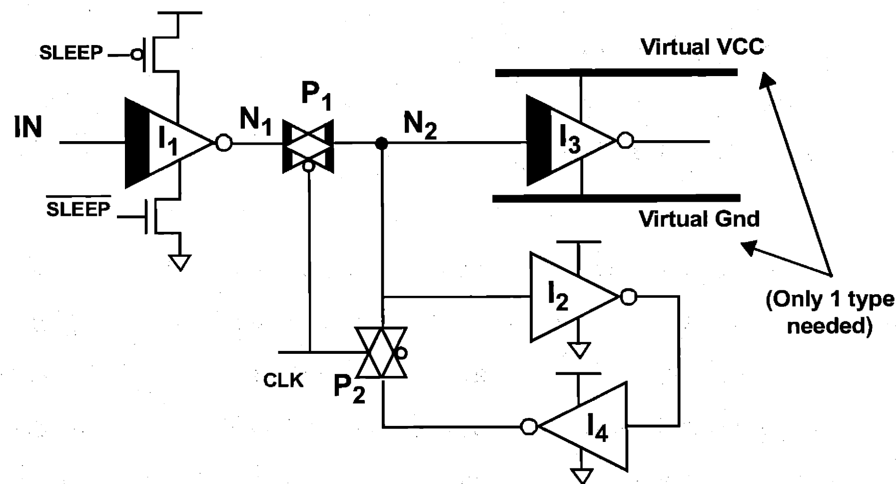


FIGURE 4-10. MTCMOS latch with reduced sneak leakage paths.

For the modified latch of above, the parallel high V_t CMOS inverter is simply disconnected from the latch output, and the high V_t recirculation path operates independently from the MTCMOS critical path. During the active period when data is written into the latch, I3 propagates signals to the output rapidly. Similarly, when CLK goes high and the latch enters the opaque state, high V_t inverters I_2 and I_4 serve to just hold the data and thus do not need to transition quickly. During the standby state, the data of the latch is recirculated in the I_2 and I_4 feedback path, which is always connected to power. In a sense, this type of circuitry is similar to the balloon storage elements of [33] in that the high V_t storage elements are completely separated from the core MTCMOS circuitry. However, in the case of the MTCMOS latch of Figure 4-10, the high V_t feedback path is utilized during the active period as well to recirculate data when CLK goes high.

Since the MTCMOS inverter I_3 no longer drives a node that is simultaneously driven by a CMOS gate, potential sneak leakage paths through I_3 are eliminated. Thus, the real advantage of this implementation is that I_3 can be connected to virtual V_{CC} and virtual Gnd, which can be shared among other MTCMOS circuits (combinational logic or other sequential circuits). In fact, only one polarity high V_t sleep device is necessary in this scheme as well. With this modified latch, inverter I_1 still needs to have local high V_t sleep transistors of both polarities to prevent sneak leakage currents from arising due to interactions between I_1 and I_4, who's outputs are both connected through a low V_t pass-gate during the standby period. Nonetheless, in a large register or more complicated circuit block, the area savings achievable by sharing high V_t power switches for a portion of the MTCMOS gates can be substantial. Of course this area savings is attainable at the expense of more complicated transistor sizing requirements that must ensure performance is maintained for all operating conditions.

4.3.5 Short circuit currents due to MTCMOS-CMOS interfaces

A final source of unexpected power dissipation that may occur during the standby mode is from short circuit currents due to MTCMOS - CMOS gate interfaces. During the standby condition, MTCMOS gates can float, so they cannot directly drive CMOS gates. If a CMOS gate has an intermediate floating input during the idle mode, then very large short circuit currents can develop. As a result, one cannot directly combine these two logic families directly. If an MTCMOS gate output needs to interface to CMOS gate, one must be sure that the input signal is actively driven during the standby state. This can be accomplished by using helper high V_t gates, or an intermediate latch circuit that holds the previous data during the entire standby period. Later in this chapter a leakage feedback gate is introduced that serves as a very efficient interface between MTCMOS and CMOS logic families.

4.4 Conventional MTCMOS Static Flip Flop

Figure 4-11 below shows an MTCMOS flip flop that can effectively cutoff all subthreshold leakage paths in the standby mode. The flip flop master stage is the same as that of the

MTCMOS latch presented earlier, but the slave stage uses a modified feedback structure and does not require a feedforward high V_t inverter.

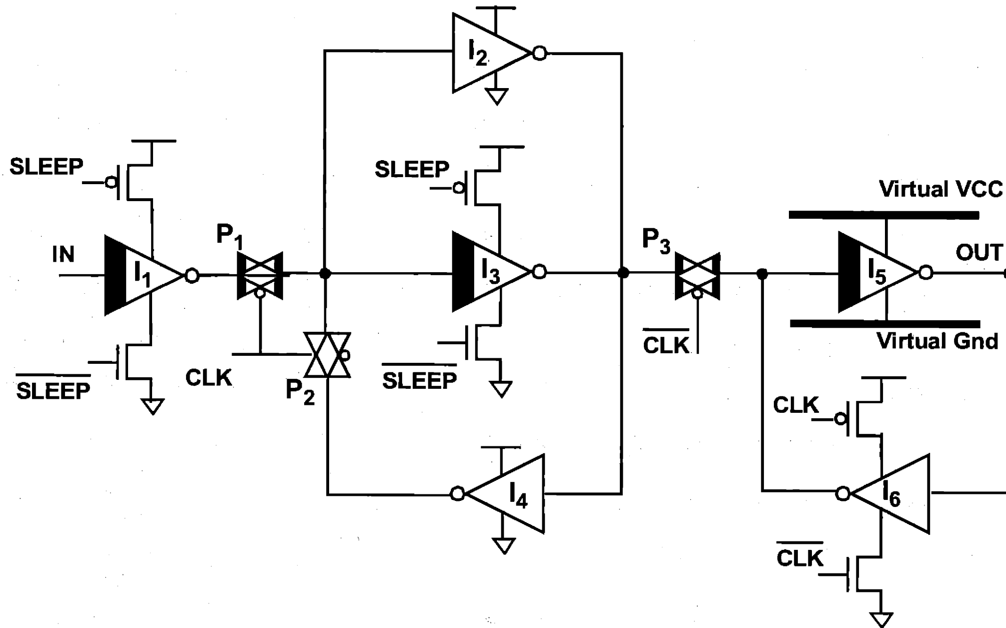


FIGURE 4-11. MTCMOS Flip Flop with no sneak paths.

During the active mode, the MTCMOS flip flop shown above operates just like a conventional master slave flip flop. The setup, hold, and propagation delays are all very fast because the critical transitions occur through the fast low V_t gates I_1, P_1, I_3, P_3, and I_5. As in the case for the stand alone latch, passgate P_2 was made high V_t so that there is less contention through the low V_t passgate during the transparent stage for the master. Similarly, for the transparent stage of the slave, I_6 is clocked "off" with high V_t devices as well. Since I_4 and I_6 are used simply to hold the state of the master and slave latches, they can be implemented with high V_t as well.

During the sleep condition, data is stored in the master latch, where CLK is high and I_2, I_4 and P_2 recirculate the stored data. After the clock goes high and the master stage enters the holding stage, the high V_t sleep transistors can be turned off thereby disconnecting gates I_1, I_3, I_5, and I_6 from the power supplies. As described earlier, localized dual polarity sleep transistors were required in order to ensure that there are no sneak leakage paths arising from MTCMOS-CMOS gate interactions. During this sleep condition, state

is recirculated in the master stage, and the slave stage is disconnected from the power supplies. Since the output is floating, the slave feedback inverter I_1 can not be implemented as a stand alone CMOS inverter followed by a transmission gate as found in the standard latch structure. Instead, the inverter followed by a passgate structure was replaced by a clocked inverter structure, where I_6 is completely disconnected from the power supplies through high V_t devices. Since the clock must be high during the standby state in order to store data in the master stage, it suffices to simply disconnect inverter I_6 from the supplies whenever clock is high to eliminate short circuit currents due to a floating output node during the standby state.

4.5 MTCMOS Flip Flop With CMOS Compatible Outputs

Because the slave stage is floating during the standby state, the MTCMOS flip flop of Figure 4-11 cannot interface directly to a CMOS gate. It can only interface to another MTCMOS logic block, which must be disconnected from the power supplies during the sleep state in order to prevent short circuit currents from forming during the idle mode. One way to solve this problem so that the MTCMOS flip flop can drive standard CMOS blocks is to modify the slave stage to be driven during the standby mode as shown in Figure 4-12, where both the master and slave stages utilize MTCMOS latch structures. During the standby mode, CLK is still "high" and data is recirculated in the master stage. However, because I_7 is always powered, it can still drive the output to a solid voltage level, thus enabling the flip flop to drive a CMOS gate even during the sleep mode. Furthermore, since node OUT no longer floats during the standby mode, inverter I_6 can be implemented as a standard CMOS gate as well. Although this flip flop prevents internal nodes from floating, the drawbacks are the extra area and capacitive loading due to I_7.

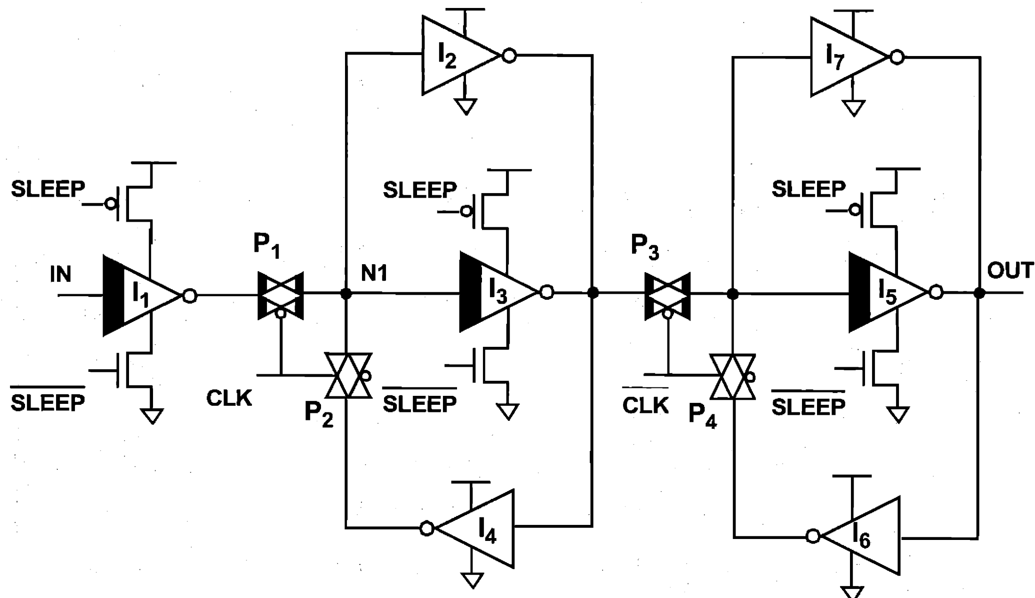


FIGURE 4-12. MTCMOS Flip Flop that has non floating output during standby state and sleep when clk high.

Although the MTCMOS flip flop above shows symmetrical master and slave stages, it still must be stalled in the clk set "high" state with the data stored in the master stage. If the flip flop enters the standby mode while clk is "low", then the data will circulate in the slave stage. As a result, node N_1 will float and inverters I_2 and I_4 can both have large short circuit currents.

4.5.1 MTCMOS flip flop that stalls when clock is low

Figure 4-13 shows an alternative permutation of a positive edge triggered MTCMOS flip flop that can be placed in the sleep mode when clock is low. This structure maintains the data in the slave stage during the sleep condition, and also provides an actively driven output stage at all times. As a result, this type of flip flop can also be used to interface to CMOS gates. However, the master stage must be modified from the previous implementations because nodes N_1 and N_2 can float during standby, so the master stage latch must not

have a feedforward path, and the recirculating inverter must be implemented as a clocked inverter.

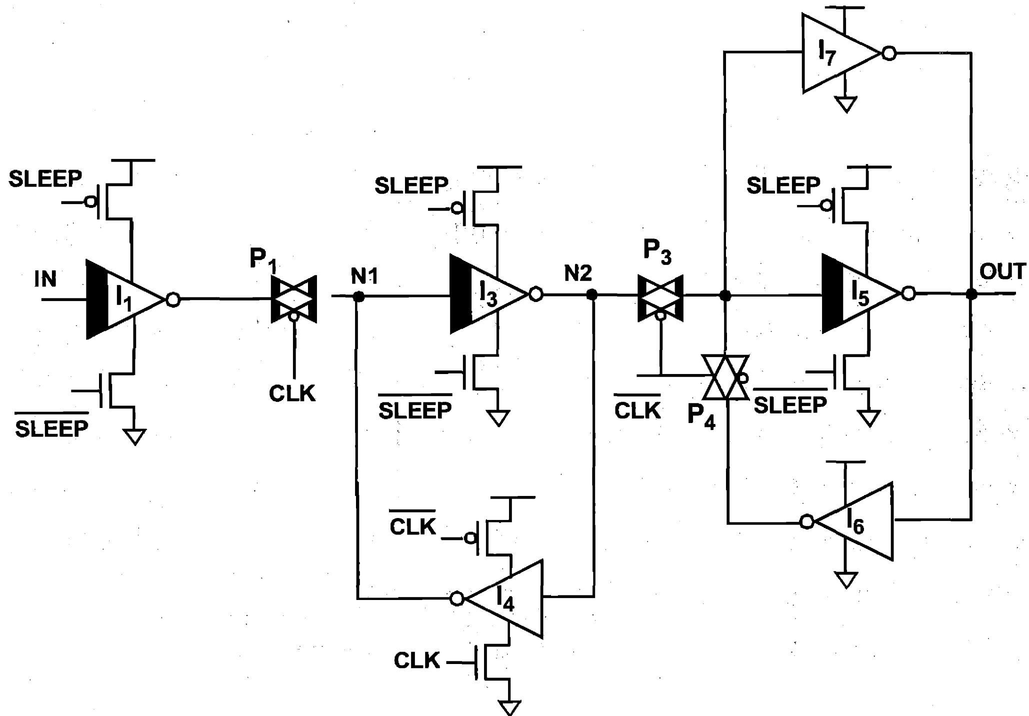


FIGURE 4-13. MTCMOS Flip Flop that can drive an output during sleep state, and sleeps when clk is low.

4.6 Impact of Parallel High V_t Circuits

The flip flops described above utilize parallel high V_t inverters to maintain state during the standby mode. However, introduction of parallel inverters to a critical path can degrade overall performance because the increase in capacitive loading can outweigh any benefit in improved current drive.

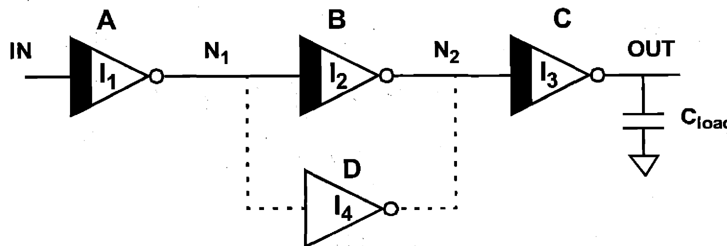


FIGURE 4-14. Inverter chain showing insertion of possible high V_t parallel device.

For example, adding a high V_t inverter I_4 to the inverter chain of Figure 4-14 will contribute to added gate capacitance on node N_1 and added diffusion capacitance on node N_2, but the amount of current gain per unit increase in capacitance will be much less than if a low V_t inverter were added in parallel instead (or equivalently if "B" was upsized). In the case where the high threshold voltage is much larger than the low threshold voltage, and the circuits are sized appropriately such that input rise/fall times are comparable to output fall/rise times, then the high V_t parallel device can turn on too late to be of much use during the critical output transition. In such cases, the added current drive due to the parallel device will be negligible compared to the degradation due to increased capacitance. Even if the addition of high V_t inverter I_4 did provide an incremental improvement in performance, more gain could be achieved by simply upsizing the existing low V_t inverter I_2 instead.

4.7 MTCMOS Flip Flop Without Parallel High V_t Devices

Since fundamentally parallel high V_t devices can only hurt performance, it makes sense to explore alternative MTCMOS sequential circuits that do not employ parallel high V_t devices, yet still can retain state during the standby mode. In some technologies, the high V_t drive can be so much worse than the low V_t drive, that it might be beneficial to implement an MTCMOS flip flop as shown in Figure 4-15 below.

FIGURE 4-15. MTCMOS flip flop without parallel high V_t paths.

This flip flop implementation avoids using parallel high V_t devices in the critical path, and instead attempts to reduce loading capacitance as much as possible. For example, rather than connecting the outputs of I_4 and I_2 to improve drive, it is better to modify the flip flop structure to minimize loading on node N_3, and to natively size the low V_t devices in the critical path to maximize performance. Although the high V_t inverter I_4 still increases the load seen on node N_2, this implementation can eliminate the extra loading due to the diffusion capacitances associated with I_4 and gate capacitance of I_5 on node N_3. As a result, the overall capacitance in the critical path is reduced significantly, which can compensate for the incremental increase in drive capability associated with the more heavily loaded case utilizing the high V_t parallel inverter. This flip flop design is also slightly more noise tolerant than a conventional MTCMOS flip flop because the recirculation path is decoupled from node N_3. This benefit is even more important if the slave stage is implemented with the regeneration path (for example if the slave stage holds the state during the standby state) because the output stage would then drive an outside block, which is completely decoupled from the recirculation path.

An added benefit of this flip flop architecture is that some sneak leakage paths that occur when MTCMOS outputs are electrically connected to CMOS outputs can be eliminated. Similar to the latch of Figure 4-10, by decoupling the high V_t feedback inverters from the MTCMOS critical path, one can utilize common virtual V_{CC} or virtual ground lines that are shared among other MTCMOS flip flops or logic circuits. Again, only one polarity sleep device is necessary for the shared power lines, although local sleep devices of both polarities are still required for MTCMOS gate I_1. As a result, using a decoupled feedback path not only eliminates some sneak leakage paths, but as described before can actually have better performance than a conventional MTCMOS flip flop that uses parallel high V_t inverters.

4.8 Leakage Feedback Gates

Another way to retain state during the standby mode without using parallel high V_t devices is to utilize a leakage feedback gate. This type of gate is a very novel derivative of an MTCMOS gate structure, which has the beneficial property of being able to actively

drive it's output to either V_{CC} or ground and still be in a low leakage state during the standby mode. Furthermore, by properly switching on or off helper high V_t devices, the leakage feedback gate can be configured such that the output of the gate during standby can be held regardless of the input signal.

When the output of an MTCMOS logic gate drives a high impedance node (i.e. the gate of a transistor), then the MTCMOS logic gate needs only one polarity sleep transistor to completely eliminate leakage currents. This is the typical configuration for standard combinational logic blocks. By utilizing only PMOS sleep transistors, it is possible for such an MTCMOS gate to hold a logic "0" during the standby state. Similarly, an MTCMOS gate that utilizes only NMOS sleep transistors can hold a logic "1" during the standby state.

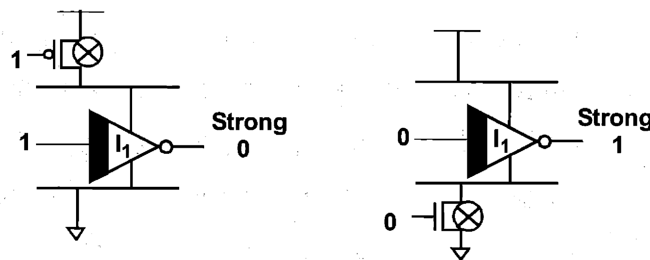


FIGURE 4-16. Low leakage but with driven outputs.

The MTCMOS leakage feedback gate is a novel type of gate that can hold either a logic "1" or a logic "0" during the standby state by selectively choosing either PMOS or NMOS high V_t sleep devices to cutoff leakage currents. Figure 4-17 below shows an MTCMOS leakage feedback gate, where minimum sized high V_t leaker devices P_2, N_2, and inverter I_3 are added. The important characteristic of this type of gate is that depend-

ing on the state of the latest output, one but not both, helper sleep devices (P_2 or N_2) is turned on.

FIGURE 4-17. Leakage feedback gate principle.

During the active mode, the behavior of the leakage feedback MTCMOS gate is not very interesting. Devices P_1 and N_1 are turned on when active, thereby connecting the gate to the power supplies. Depending on the state of the output, either P_2 or N_2 will also be "on", but the effect of these helper sleep devices is negligible on circuit performance. For example, if the output is low, then N_2 is "on" and P_2 is "off". The "on" N_2 tends to help lower the resistance of the NMOS sleep transistor, while the off P_2 has no impact on the "on" P_1. However, a gate transition will cause the output to go from low to high, so the reduced effective NMOS sleep transistor resistance will not be of any use since the desired transition is in the opposite direction. Similarly for an output high condition, the added helper sleep devices will not be beneficial either. In fact, the addition of the extra inverter I_3 will cause more loading and cause the MTCMOS gate to be slower than a conventional gate. As a result, I_3, P_2, and N_2 should all be minimum sized in order to minimize the area overhead and loading effects. To further save power, an alternative implementation could be to tri-state inverter I_3 so that it does not constantly switch during the active computation.

During the sleep state however, the leakage feedback MTCMOS gate has the beneficial property of being able to hold its data. When the sleep signal is asserted, high V_t sleep devices P_1 and N_1 strongly turn "off", but based on the existing output signal, either P_2 or N_2, but not both, will remain "on" as before. As a result, the output voltage of the gate will continue to be strongly driven to the original logic level, but at the same time the

leakage currents will be reduced several orders of magnitude due to the "off" high V_t devices. The leaker devices P_2 and N_2 can be minimum sized because they are simply responsible for holding the current state of the output during the idle state. The actual signal switching will occur during the active mode through the large P_1 and N_1 devices. With a leakage feedback gate, an MTCMOS gate can be designed such that it will no longer float during the standby mode because the output will always be driven to one rail or another. In order to provide this functionality, the high V_t sleep transistors P_1 and N_1 need to be locally inserted into the MTCMOS gate. In other words, the virtual V_{CC} and virtual ground lines can not be shared among other MTCMOS blocks without causing sneak leakage paths.

4.8.1 Leakage feedback gate with floating inputs

Another benefit of the MTCMOS leakage feedback structure is that the output during the standby mode will be held regardless of any change in the input. In theory, the leakage feedback gate can operate in two different modes during the standby condition. In the nominal case, the input signal is unchanged after entering the standby mode and the correct pullup or pulldown path will be continuously activated to actively drive the output to the proper rail. In the other operating mode, the input signal can switch to the complementary state or even float. In this case neither an active pullup or pulldown path will exist on the output of the gate. However, due to the imbalance of several orders of magnitude between subthreshold leakage currents in high V_t versus low V_t devices, the output voltage can still be held to the previous state. In effect, the difference in leakage currents will pin the output voltage of the gate to the previous value. The output voltage is fedback to the gate to maintain the proper structure to set the proper leakage relationship between the pullup and pulldown paths, thus giving rise to the notion of "leakage feedback." However, since the "leakage feedback" mechanism is used to only maintain a voltage output during the standby state (the currents used for output switching would be actively driven), operating in the subthreshold regime is still acceptable.

For example, Figure 4-18 below shows two such scenarios where the input to a leakage feedback gate switches after entering the standby mode. As a result, in the first case the input is driven "low" after entering the standby mode, but the output remains

"low", while in the second case the input is driven "high" after the standby mode but the output remains "high". In the first case, even though a low V_t PMOS is turned "on", the path to V_{CC} must go through strongly turned off high V_t devices. On the otherhand, the path from the output to ground is through a strongly turned "on" high V_t device, and a turned "off" low V_t device. Thus the effective resistance to ground is much lower than the effective resistance to V_{CC}, so the output remains low. In the second case, the opposite scenario occurs, where the low V_t PMOS is turned off, but the high V_t PMOS is turned on. Similarly, the resistance to V_{CC} is much lower than the resistance to ground, and the output is held high. One caveat though is that if the input signal does change, there is a possibility that charge sharing can cause the output of this gate to glitch. However, there are several ways to minimize this problem. First of all, if the input to the leakage feedback gate stays at the same voltage signal for any period of time, the charge sharing amount will be significantly reduced because the leakage currents will tend to equalize the output voltage with the voltage on the virtual ground or virtual power nodes. Thus if the input signal does switch, the amount of charge sharing would be reduced. Another way to reduce this problem is simply to ensure that the output load capacitance is large compared to the local virtual power and ground line capacitances.

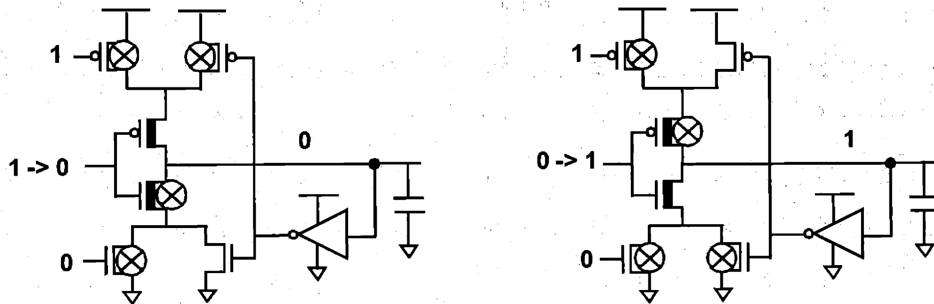


FIGURE 4-18. Leakage Feedback output retains state regardless of changes to input

Because of the leakage discrepancy between high V_t and low V_t devices, the DC operating point of the leakage feedback gate will be close to V_{CC} or ground regardless of

the input value. Figure 4-19 shows equivalent circuits for the scenarios where an off high V_t device is placed in series with an off low V_t device for both holding conditions.

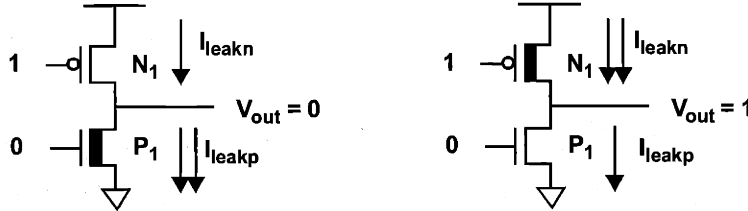


FIGURE 4-19. Output state held by leakage currents.

The DC operating point for the configuration of Figure 4-19 can be calculated by equating the leakage currents of P1 and N1 shown in Eq 4-1 and Eq 4-2 (neglecting DIBL) and solving numerically.

$$I_{leakP} = I_0 e^{\frac{-V_{tp}}{nV_{th}}}\left(1 - e^{\frac{V_{out}-V_{CC}}{nV_{th}}}\right) \qquad \text{(EQ 4-1)}$$

$$I_{leakN} = I_0 e^{\frac{-V_{tn}}{nV_{th}}}\left(1 - e^{\frac{-V_{out}}{nV_{th}}}\right) \qquad \text{(EQ 4-2)}$$

For a more intuitive derivation of the DC operating point, a graphical solution to the circuits in Figure 4-19 is more appropriate. For example, Figure 4-20 shows I-V curves (plotted on a semilog scale) relating leakage current versus V_{out} for a turned off high V_t PMOS and a turned off low V_t NMOS. Since the devices are placed in series, the currents must be equal, which corresponds a DC operating point at the intersection of the two curves. The curves below assume a technology $V_{thigh} = 0.4$, $V_{tlow}=0.2$, $V_{CC}=1.2$, sub-threshold slope = 100mV/Decade.

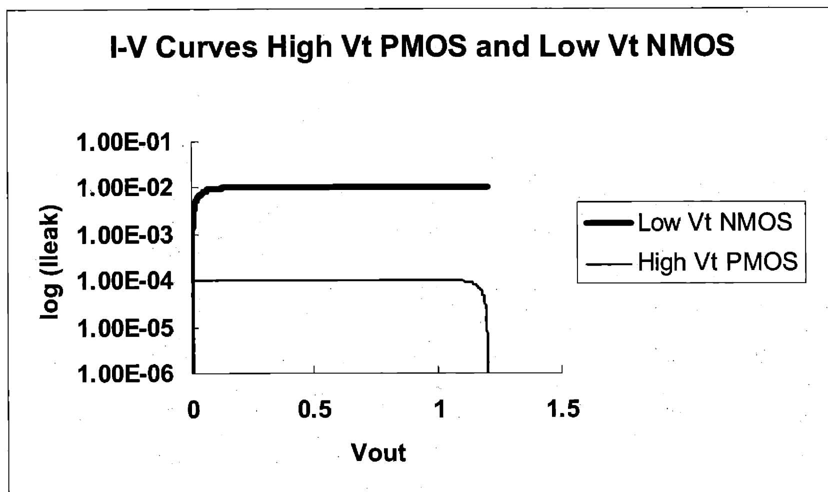


FIGURE 4-20. Intersection showing DC operating point at logic "0"

Because of the exponential dependency of leakage current on V_{out}, we can see that the cross over point occurs where the high V_t PMOS curve is relatively flat and the low V_t curve is sharply transitioning. As a result, the DC operating point for node V_{out} is very close to ground. Even for small differences between the high V_t and low V_t values, the steep drop off in the low V_t I-V curve ensures that the crossover point will be at a logic "0" value. In the example above, the high and low V_t threshold voltages are separated by 200mV, which corresponds to leakage currents which differ by 2 orders of magnitude (assuming a subthreshold slope of 100mV/Decade), and the equilibrium point is virtually pinned to ground. Even for separation of less than 100mV, the intersection point would still correspond to a logic "0."

For the opposite configuration, where the off PMOS is implemented as a low V_t device and the off NMOS is implemented as a high V_t device, a graphical solution can show that the equilibrium voltage is very close to V_{CC} for the series connected circuit.

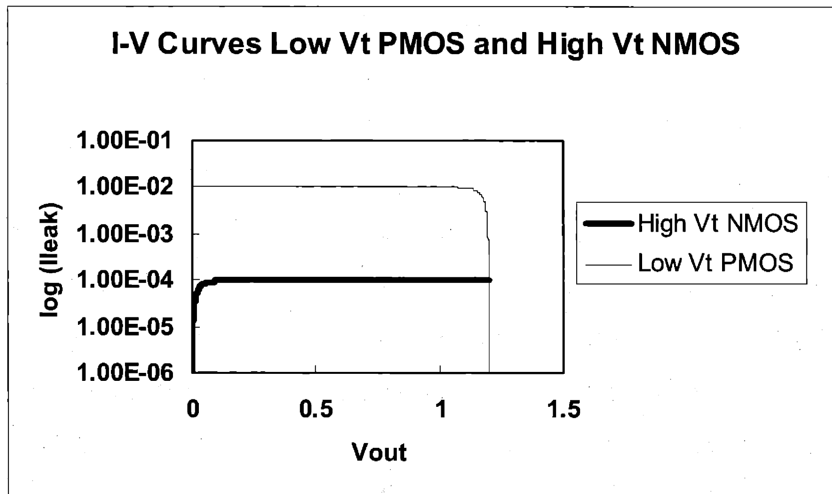


FIGURE 4-21. Intersection showing DC operating point at logic "1"

Here the cross over point is again where the high V_t curve is flat and the low V_t curve has a high slope. As a result the DC operating point is virtually pinned at V_{CC} because of the extreme slope.

As illustrated above, the leakage offsets between high and low V_t devices can be used to sustain a logic "0" or a logic "1" when both series connected devices are turned off. As long as there is a modest offset between high and low V_t devices, then an arbitrary logic output can be maintained. Although leakage currents can satisfactorily hold a DC output voltage, the response to transient noise will be slower than in the case using actively driven gates. However, as threshold voltages continue to scale, leakage currents will become increasingly large, so that the output voltage of a leakage held gate has a low enough impedance such that noise coupling is small. Furthermore, during the standby condition, the leakage feedback gate is only supposed to hold an output signal stable rather than perform any switching. Thus, utilizing leakage currents to hold the output data value (when the input signal floats or switches to the opposite rail for example), can still provide consistent results.

Simulations were performed to confirm the data holding capability of leakage feedback gates. The simulations were performed in a 0.14µ technology with high V_t approximately 0.15V and low V_t approximately .05V (defined at the 10nA @ 10um point), and show how the output of a leakage feedback gate will retain the output voltage during the sleep regardless of the input signal. Figure 4-22 shows waveforms for a clock input being fed into a simple leakage feedback inverter that is enters the sleep state when the input is high. As a result, the output is in the low state prior to when the sleep signal is asserted, and continues to remain low from then on. When the input signal is driven high, the output voltage is held to ground through an active pulldown path. However, when the input signal is driven low, the output voltage is held to ground through a leakage path. In either case, the output remains a constant DC value.

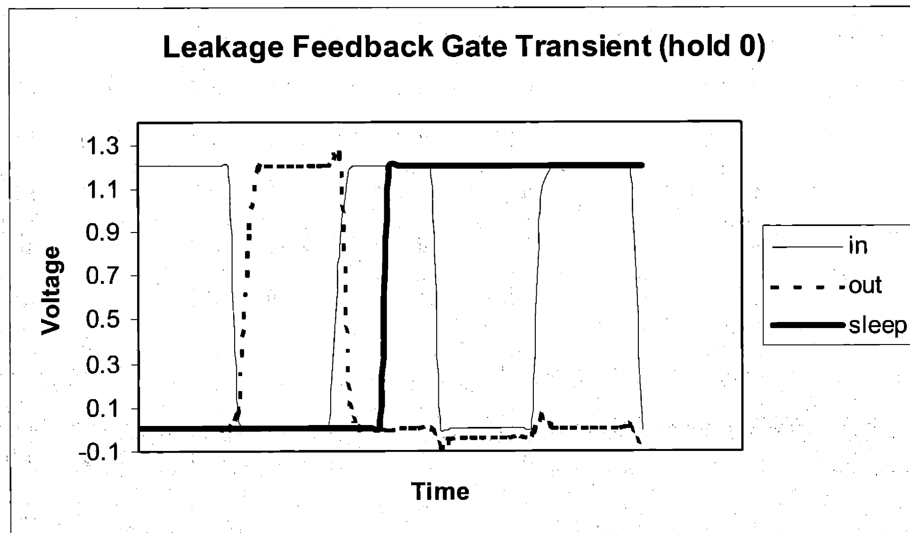


FIGURE 4-22. Leakage feedback transient holding 0.

Figure 4-23 shows the opposite case where the sleep mode is asserted when the input clock period is low. In this scenario the output is high before entering the sleep state, and remains high throughout the standby period. Similarly, when the input signal is low, the output is driven high through an active pullup path, but when the input signal is driven high, the output voltage is held to V_{CC} through a leakage path.

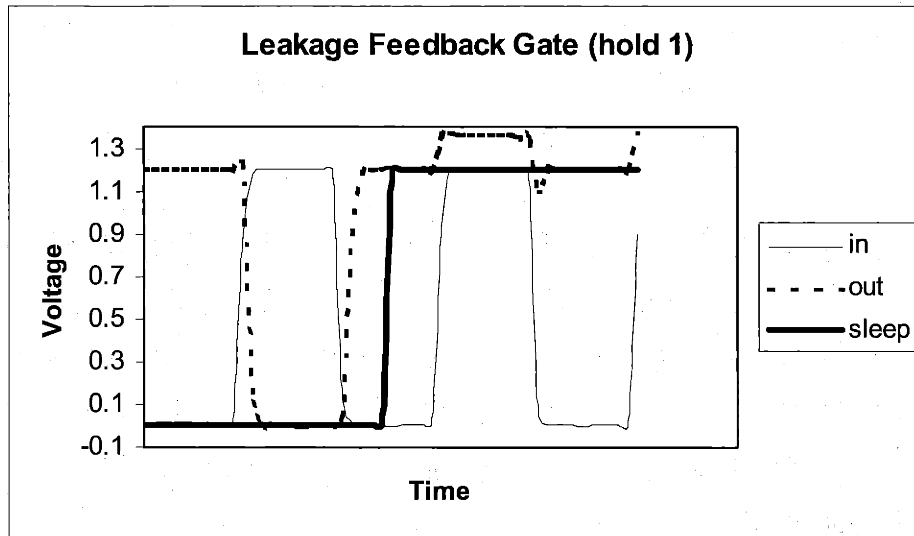


FIGURE 4-23. Leakage feedback transient holding 1.

4.8.2 Complex leakage feedback gates

The leakage feedback gate shown in Figure 4-17 is implemented with an ordinary inverter. In actuality, the leakage feedback gate can be implemented as a complex gate consisting of all low V_t devices. The behavior of a complex leakage feedback gate is similar to that of a standard inverter implementation. The helper high V_t devices are switched accordingly to maintain the output voltage of the complex gate during the standby state. If the inputs to the leakage feedback gate are held, then the gate will actively drive the output voltage. On the otherhand, if the leakage feedback inputs float, then the gate could be held through a leakage feedback mechanism. For the leakage mismatch to function properly, it is still important for the DC operating point to be close to V_{CC} or ground depending on the leakage relationship between the path through an off high V_t device and an off low V_t network. In otherwords, it's important for the leakage of the off low V_t circuit to dominant over the leakage through the helper high V_t device. In a complex gate, the circuits can have stacked devices, and a series of off low V_t devices will have lower leakage currents due to source biasing effects. As a result, the V_t spread must be large enough, and the helper high V_t devices sized small enough such that any worst case low V_t series chain will still have a lower effective resistance than the off high V_t helper device.

4.8.3 Leakage feedback interface circuitry

An immediate use of the leakage feedback gate is that it can serve as an interface circuit between MTCMOS and CMOS logic blocks. A leakage feedback gate will hold it's previous state during the standby mode, and as a result can be used to directly drive a CMOS stage during the standby state. If a standard MTCMOS block drives a CMOS stage, then the CMOS inputs can float when the MTCMOS block is placed in a standby mode, and can cause severe short circuit currents. As a result, interfacing between MTCMOS and CMOS stages generally would require the use of extra latches or flip flops to maintain signals during the sleep state in order to actively drive a CMOS block. Unfortunately, this would result in significant overhead in extra circuitry and area that provides no other useful function except to ensure that nodes do not float during the standby state. Furthermore, the extra circuitry can cause extra loading that degrades performance during the active period as well.

Fortunately, leakage feedback gates can be used as a very simple means to provide interface capabilities between MTCMOS blocks and CMOS blocks. By modifying the last stage of a MTCMOS block to be a leakage feedback gate, the output signal will never float and can drive a CMOS block even during the standby mode. To further streamline the leakage feedback gate, the feedback inverter can be collapsed into the next stage logic rather than utilizing a separate feedback inverter. For example, if an interface signal connects to only 1 CMOS inverting gate, then it is safe to simply use the output of this gate as the feedback signal to enable or disable the helper high V_t sleep devices as shown below.

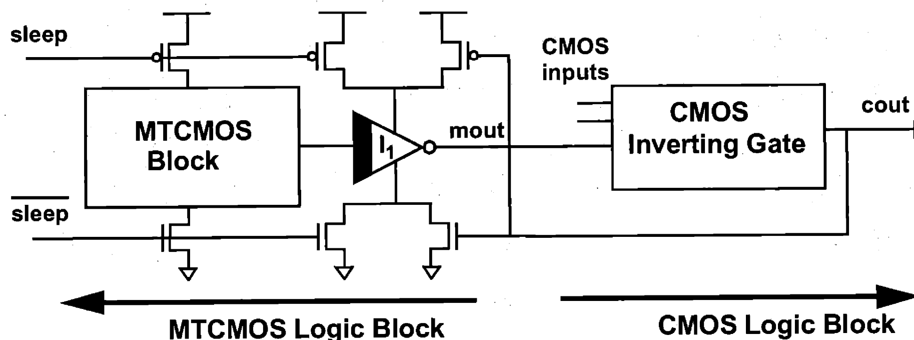


FIGURE 4-24. Leakage Feedback gate as interface between MTCMOS and CMOS blocks (sharing CMOS output signal)

The next stage CMOS gate output can be used as a feedback signal because this will ensure that during the sleep stage, the output of the leakage feedback gate, "out" will correspond to a voltage level that ensures the CMOS logic block output stays the same. For example, if "mout" and the CMOS gate output "cout" are opposite in polarity, then the feedback circuitry will reinforce the leakage feedback gate to maintain that output level. However, if the "mout" and "cout" are the same polarity, then the feedback signal would cause the leakage feedback gate to flip the state of "mout." It can be shown though that in this scenario, regardless of the state of "mout," the CMOS inverting gate would be driven to the proper output voltage simply due to the CMOS inputs to the gate For example, if "mout" is high for example and "cout" is already high, then if "mout" went low, "cout" would have to remain high. This is because a high to low signal to a CMOS inverting gate can result in either no transition or a low to high transition, and vice versa for the opposite case. Thus, if it happens that "mout" and "cout" are the same polarity, then even if "mout" floats the CMOS gate will properly drive it's output voltage. As a result, even if the leakage feedback gate needs to transition through a leakage mismatch mechanism (to reach the DC operating point of the leakage feedback gate), there will still not be any short circuit currents.

Figure 4-24 shows a special case where the MTCMOS output signal drives a single CMOS gate. If on the otherhand this signal is routed to several different gates, then one would have to use a standard leakage feedback gate like that of Figure 4-17, which uses a stand-alone feedback inverter to ensure that the CMOS block is driven with the same voltage levels during the sleep condition as before. Also, Figure 4-24 shows the leakage feedback gate implemented as an ordinary inverter. A more efficient implementation could be to implement the leakage feedback as a complex gate that performs part of the logic computation of the previous MTCMOS block.

By using leakage feedback gates as interface circuits, the CMOS input voltages are held to their previous values during the standby state. In effect, the leakage feedback gates serve as latches which are transparent during the active state, and are opaque during the standby state, thereby holding valid CMOS compatible outputs. For some circuit designs, it may be beneficial to have this added functionality where CMOS blocks are driven to

appropriate logic levels even during standby. In such cases, one must be careful on how the circuit behaves when returning from the standby mode back to the active mode. When the sleep transistors turn back on, it will take some time for the MTCMOS blocks to be driven back to their active levels (since these blocks floated were floating). If the leakage feedback gate is turned on immediately, then the following CMOS blocks may glitch before returning to the proper voltage level. Two approaches can be taken to solve this problem. First, one can delay turning on the leakage feedback gate until after the MTC-MOS block has settled. Second, if state is important to keep, then a flip flop can be used to retain state.

Since during the standby mode no computation takes place, it may be acceptable for the CMOS block to switch to some intermediate value during the sleep condition, and just return transition back to the proper value after the circuit enters the active mode. This is a simple control sequence- one simply waits a short period time after switching into the active mode for node voltages to resettle to their appropriate values. However, if this methodology is employed then it is even easier to provide interface circuitry between MTCMOS and CMOS blocks. For example, the leakage feedback gate can simply be modified by removing the feedback path, and instead, simply tie the helper high Vt devices to an arbitrary node. For example, if the high Vt devices are driven to be a logic high, then the leakage feedback gate will always drive a logic 0 during the standby state. However, with this implementation, the output of the leakage feedback gate may transition through leakage currents (rather than just being held by leakage currents), so one must be sure it is fast enough such that any crowbar currents induced in the attached CMOS blocks are acceptable.

4.9 Leakage Feedback Static Flip Flop

The leakage feedback gate can also be integrated into MTCMOS flip flops to take advantage of it's data holding ability during the sleep stage. The leakage feedback approach is advantageous because it provides a nonvolatile flip flop solution that avoids utilizing parallel high V_t devices (which can limit performance as described earlier), and thus avoids

extra loading on critical nodes. A leakage feedback MTCMOS flip flop that stores state in the master stage and has CMOS compatible outputs is shown below.

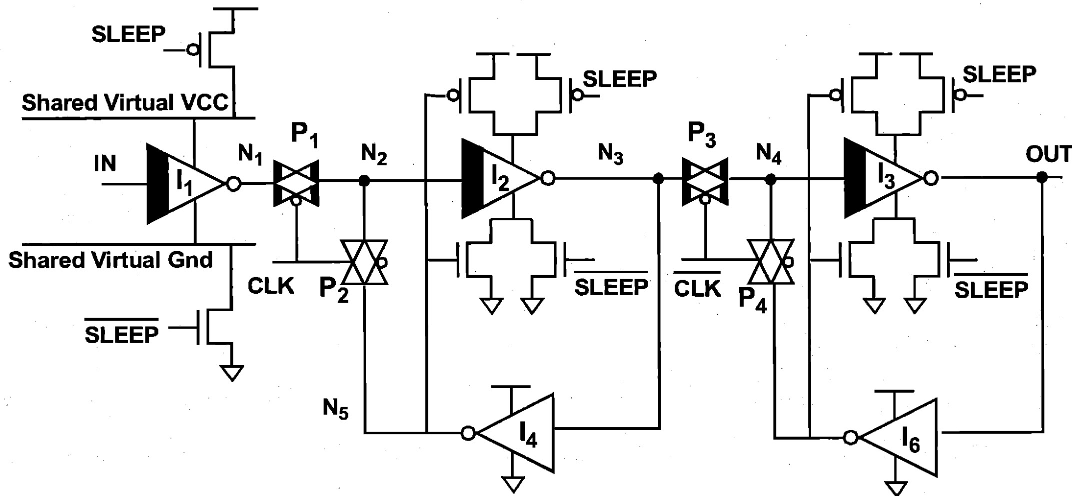


FIGURE 4-25. MTCMOS leakage feedback flip flop that has non floating output during standby state and sleeps when CLK is high.

4.9.1 Active operation

The flip flop of Figure 4-25 is closely related to the MTCMOS flip flop of Figure 4-12, which utilizes parallel high V_t devices in both the master and slave stages. Both flip flops recirculate data in the master stage and provide CMOS compatible outputs, but the leakage feedback flip flop has better performance characteristics. During the active mode, the leakage feedback flip flop behaves exactly like a standard master slave flip flop, as the helper high V_t devices are only used to retain state during the standby state. When either the master or slave latch is in the holding state, the feedback path is turned on, and the stored value is recycled through cross coupled inverters.

As described earlier, the addition of high V_t parallel devices (to provide state retention) in the conventional MTCMOS flip flop tends to degrade performance because it adds more load to the critical path, yet may not provide enough current drive to offset the increase in capacitance. The leakage feedback flip flop on the otherhand avoids the use of parallel devices, and thus does not introduce any extra loading to the critical path. In fact, the leakage feedback flip flop has even better performance than the MTCMOS flip flop of

Figure 4-11, which utilizes a parallel high V_t inverter only in the master stage and does not have CMOS compatible outputs.

Compared to a basic master slave flip flop, the leakage feedback flip flop merely adds helper high V_t PMOS and NMOS devices to I_2 and I_4, which are driven by the outputs of high V_t inverters I_4 and I_6. Fortunately, these extra loads are not part of the critical path, and take up very little extra area. As a result, the critical path loading of a leakage feedback gate is comparable to that of a basic MTCMOS master slave flip flop that does not even retain state during the standby condition. In otherwords, the leakage feedback master slave flip flop has a critical path with loading corresponding to the minimum amount of circuitry needed to perform the master-slave functionality during the active mode. The state retention and CMOS compatible outputs during the standby mode come almost for free from a performance trade-off point of view. However, there is a small area overhead, corresponding to the helper high V_t devices, that is necessary to provide this extra functionality. The helper devices add to the total area for the power and ground high V_t devices, but do not actually increase the effective width of the high V_t sleep transistor widths because the helper device associated with the next transition direction would be off. However, the incremental capacitance on the virtual power and ground lines of I_2 and I_3 can still be beneficial to switching because it helps prevent the virtual power lines from bouncing.

4.9.2 Standby condition

During the standby condition, data is stored in the master latch, with the clock held high. The leakage feedback gate switches the appropriate helper high V_t device on such that the previously held data in the master stage is actively maintained.

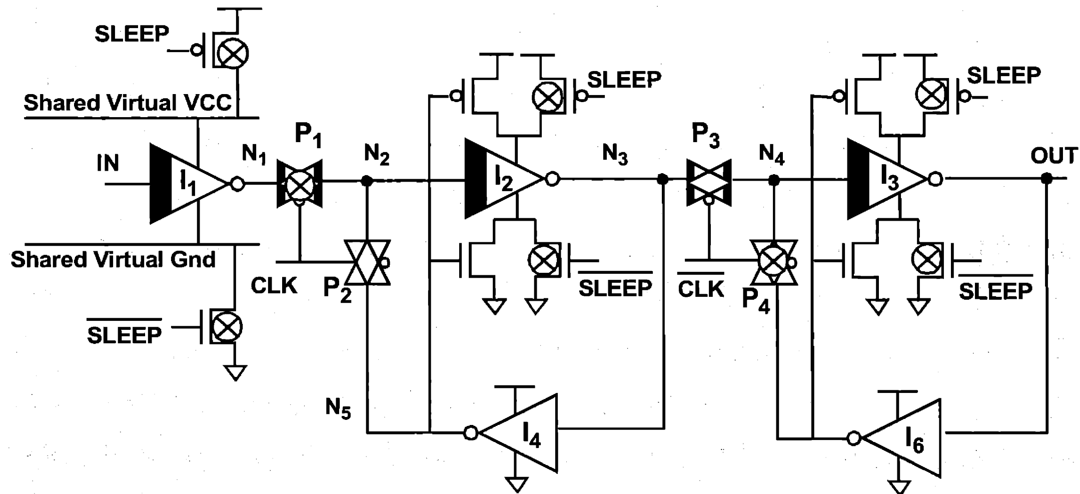


FIGURE 4-26. MTCMOS Leakage Feedback flip flop in the standby mode

Due to the structure of the leakage feedback flip flop, the feedback path through passgate P_2 is driven so the input to the leakage feedback gate, node N_2 in Figure 4-26 is correctly held to the target value. As a result, the output of the master stage leakage feedback gate is held by an active pullup or pulldown path rather than through a leakage path. .

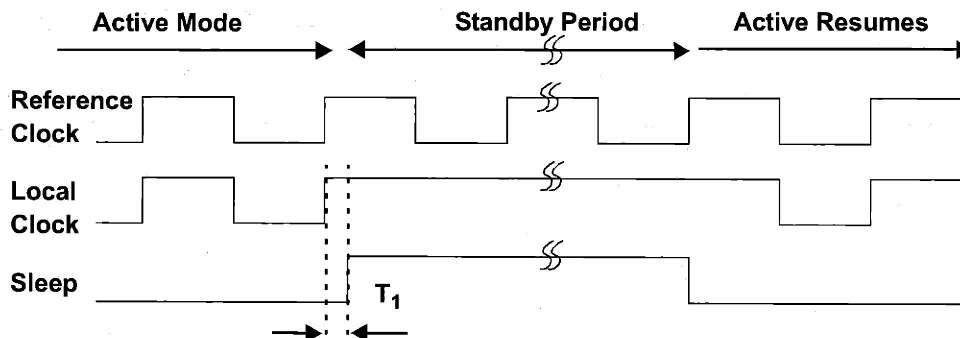


FIGURE 4-27. Sleep Mode timing diagram MTCMOS leakage feedback flip flop.

Figure 4-27 shows a timing diagram for how the standby mode is entered and how the circuit returns to the active mode. The signal is asserted after the local clock goes from low to high and remains high for the remainder of the sleep period. However, the sleep signal is asserted a time T_1 after the clock edge. Depending on the length of T_1, the slave stage, which is in the transparent state when clock is high, will store either the previous or current value of data during the standby state.

The minimum delay past the rising clock edge for the sleep signal is 0. Assuming that the setup time is maintained for the flip flop and that node N_5 is properly switched before the clock signal goes low to high, then the sleep signal can be activated immediately. However, by entering the sleep mode at the same time the clock goes low to high, the slave stage will enter the standby mode before the new data can propagate through the slave logic. As a result, the slave stage will hold it's previous data and not reflect the updated data stored in the master stage. Furthermore if node N_4 does transition after the leakage feedback gate is placed in the sleep mode, then it is possible for the output of the slave leakage feedback gate to no longer be actively driven. Instead, the output can be held through leakage currents since the output of the slave stage will no longer correspond to the inversion of the input.

On the otherhand, if T_1 is large enough, then the new data will have time to propagate through the slave stage before entering the sleep state. In this case, the proper helper high V_t device will be turned on to match the polarity of the signal on node N_4, which is driven by the master stage. As a result, during the standby state, the output of the flip flop will reflect the data stored in the master stage and will be driven with an active path to power or ground. Thus for either large or small values of T_1, the output of the leakage feedback flip flop will continue to hold a valid output during the standby state, and thus can be used to drive a CMOS logic block. Either the data will be actively driven and reflect the state of the stored data in the master stage, or it will be held through leakage feedback and reflect the previous stage of the flip flop. For practical purposes, it is probably better to implement the sleep condition with enough lag between the sleep and rising clock edge such that the slave stage was able to fully transition before entering the standby

mode. There is less timing uncertainty with this approach, and the output of the flip flop is actively driven, which makes it less susceptible to noise interference.

4.9.3 Comparison of MTCMOS leakage feedback flip flop to conventional one

The leakage feedback MTCMOS static flip flop has better performance and virtually the same functionality of an equivalently sized MTCMOS flip using parallel high V_t inverters in both the master and slave stages. The active operation of both flip flops are identical, yet the leakage feedback implementation has inherently better performance because of reduced loading to the critical paths. During the standby mode, both implementations significantly reduce leakage currents, hold their data indefinitely, and also provide CMOS compatible outputs. The conventional design utilizes parallel high V_t devices to continue driving the internal nodes of the flip flop, while the leakage feedback flip flop cuts off leakage currents by strongly turning off a path to ground or V_{CC} but simultaneously continuing to drive the output of the gate to the proper signal level. The leakage feedback gate thus provides an active pullup or pulldown path to power or ground during the standby state. Thus the only difference in standby operation is that the output of the leakage feedback slave stage (which is in the transparent state) can be locked into the previous output state if the sleep signal is asserted too close the rising clock edge. By waiting until the slave stage transitions before asserting the sleep signal on the leakage feedback flip flop, one can ensure that the internal node voltages are held through active paths to power or ground rather than through a leakage path.

Even compared to the flip flop of Figure 4-11, where a high V_t parallel inverter is only used in the master stage, the leakage feedback flip flop still has better performance. As described before, this is because the leakage feedback gates provide data holding functions without providing any extra load to a basic master slave flip flop architecture. However, the trade-off of using a leakage feedback gate is that extra area is required for the helper high V_t devices (which can improved by using minimum sized devices) and each leakage feedback gate must have localized PMOS and NMOS sleep transistors. Localized sleep devices are required because each leakage feedback gate can selectively connect to either power or ground depending on the previous output value. As described earlier, the penalty of using localized sleep devices is not too severe because sleep transistor sizing

requirements are simplified, and sensitive memory elements are decoupled from external circuitry.

However, if area is a primary concern, then being able to share high V_t power switches among several sequential circuit gates may be useful. For the MTCMOS flip flops of Figure 4-11, localized sleep transistors were used as well, so theses flip flop implementations still have comparable area requirements as the leakage feedback static flip flop approach. However, the conventional flip flops can be further optimized to eliminate many sneak leakage paths, which will allow sharing of virtual ground and virtual power lines, as illustrated in Figure 4-15 (although this flip cannot drive CMOS outputs). Other techniques such as "balloon" circuits can also be used to maintain state yet still allow sharing of high Vt sleep devices. On the otherhand, leakage feedback gates fundamentally cannot be shared, so the use of localized sleep devices is unavoidable. Nonetheless the simplistic control, ability to actively drive CMOS downstream gates, and superior performance of leakage feedback static flip flops makes them very good alternatives to conventional MTCMOS sequential circuits.

4.10 MTCMOS Dynamic Flip Flop with Standby Data Retention

One especially compelling application of leakage feedback is to implement dynamic flip flops that retain state during the standby mode yet still have very fast circuit performance during the active state. Dynamic flip flops store state dynamically on intrinsic gate capacitances, thus eliminating the need for any feedback paths, and exhibit extremely high performance during switching. However, the noise margins in dynamic flip flops are smaller than that of a robust static implementation, so the flip flop must also be clocked higher than a minimum frequency to insure that the dynamic nodes can hold charges during the clock cycle. A simple MTCMOS dynamic flip flop that does not retain state during idle modes is shown below:

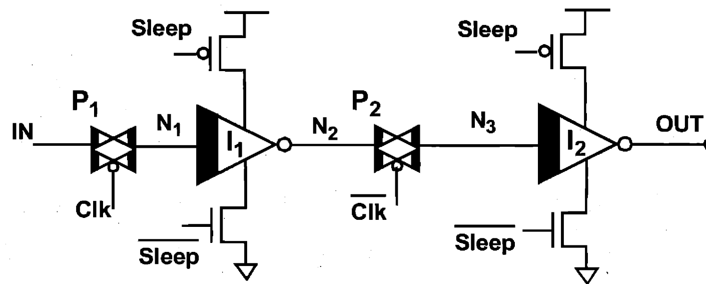


FIGURE 4-28. Basic MTCMOS dynamic flip flop.

The MTCMOS dynamic flip flop is the same as a basic CMOS implementation except that the inverters utilize high V_t power switches to cutoff leakage currents during the standby state. As technology scales and leakage currents continue to increase, the dynamic nodes are even less stable and storage capacitances shrink as well, thus requiring higher minimum operating frequencies to ensure functionality. However, with aggressive techniques and careful noise analysis, dynamic flip flops can provide very high performance in future technologies.

A leakage feedback dynamic flip flop implementation is shown below in Figure 4-29. This leakage feedback dynamic flip flop has the same structure as the standard MTCMOS dynamic flip flop except that the master stage is replaced with a leakage feedback gate, which serves to hold state during the standby mode. The leakage feedback gate adds very little extra load to the critical path compared to the basic dynamic flip flop structure. Only an extra minimum sized feedback device I_3 is introduced, but this enables the flip flop to retain state during standby mode through a leakage feedback mechanism, yet still exhibit very high performance because of the minimal loading on critical nodes

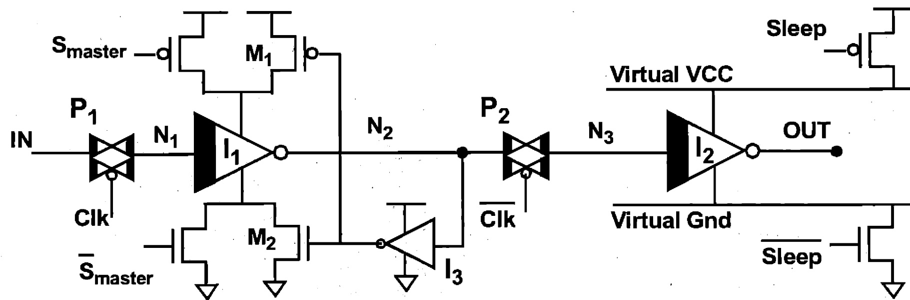


FIGURE 4-29. MTCMOS leakage feedback dynamic flip flop.

4.10.1 Active operation

During the active mode, the flip flop acts like an ordinary dynamic latch, where data is stored dynamically on the input capacitances of I_1 and I_2. When CLK is low, passgate P_1 is turned on and P_2 is turned off so that master stage begins to process the incoming data. Meanwhile, the slave stage holds the previous data on the dynamic node N_3. When CLK goes high, the slave stage becomes active, and the new state of the master latch is stored on the dynamic node N_1. Since P_1 and P_2 are low V_t transmission gates, leakage currents can degrade the data storage node, so care must be taken to restrict the circuit operation to those with a high enough clock speeds to retain functionality.

The leakage feedback devices M_1, M_2, and I_3 do not serve any purpose during the active stage, but play an important role during the standby condition since it can continue to hold the state of the flip flop indefinitely. The extra loading due to M_1 and M_2 are not in the critical path of the inverter, but the minimum sized inverter I_3 slightly slows down the leakage feedback dynamic flip flop compared to a conventional one.

During the active mode, the state of the flip flop is held on dynamic nodes, and thus must continually be clocked to refresh the state and cannot be clock gated to save power. On the otherhand, during the standby state, the dynamic flip flop will act like a static flip flop and continually hold it's state even when the power supplies are cutoff and the CLK signal is gated.

4.10.2 Standby operation

To enter the standby state, the leakage feedback dynamic flip flop must first be clocked high so that the flip flop state is stored on node N_1. Next, the sleep signals (sleep, smaster, and their complements) are activated to turn off the high Vt power line switches a short time after the rising clock edge. This is important because the new data must propagate through I_1 and I_3 before entering the sleep mode so that the leakage feedback gate can turn on one of the minimum sized high V_t helper transistors (M_1 or M_2). Unlike the static version of the leakage feedback flip flop, the minimum setup time of the dynamic flip flop may not be long enough to ensure that the leakage feedback gate will capture the new data if the sleep signal is immediately activated. As a result, a small delay is required to ensure operation. During the standby state, the proper helper high V_t device will be turned on to maintain the output voltage. However, because the input node N_1 is free to float during the standby state, the leakage feedback gate could be held through a leakage path. Again, because of the nature of the leakage feedback gate, the dynamic leakage flip flop needs local high V_t sleep transistors, which cannot be shared among multiple MTCMOS blocks. Fortunately, this is not a major concern because local sleep transistors are important to decouple the sensitive dynamic flip flop from outside circuitry.

With leakage feedback dynamic flip flops, one can enable a standby gating methodology where unused blocks using dynamic flip flops can be stalled in time when not in use, and yet be woken up later to finish a computation. Clock gating alone is not sufficient because during the active state the dynamic flip flop's dynamic nodes would simply leak away. However, by combining clock gating during the sleep mode, the dynamic flip flops can be made to retain their state through leakage feedback. Traditional dynamic flip flops are incapable of maintaining data during the idle state, and as a result can not directly be stalled and restarted. Instead, with traditional dynamic flip flop based circuits, non volatile memory circuits must be used to record the state of the block before stalling the clock. For example, if a datapath pipeline is implemented with traditional dynamic flip flops, either the state at each pipeline stage must be saved externally, or the pipeline must be flushed out so that computation is completed before being stalled. However, with a design methodology utilizing leakage feedback dynamic flip flops, a pipeline stage can be arbi-

trarily stalled and placed in a low leakage standby mode, and then restarted without risk of losing data and without using peripheral circuitry or complex control sequences to retain state. By stalling the clock as well as asserting the standby mode, one can reduce both dynamic power dissipation as well as leakage power. The leakage feedback dynamic flip flop thus provides functionality to architectures using dynamic flip flops that was not available before, and is a novel way to effectively utilize leakage feedback gates.

4.10.3 Exiting standby mode

A more serious problem with the leakage feedback dynamic gate is that the control sequences when exiting from the standby state is slightly more complicated than previous clocking requirements. For the conventional MTCMOS flip flop or leakage feedback static flip flop described earlier, the timing sequence was straightforward. To enter the standby state, the sleep transistors need to turn off after the clock goes high. Similarly, when returning to the active mode, the sleep transistors simply turn back on, and the clock resumes from the "high" phase. For the leakage feedback dynamic flip flop, the timing requirement is more involved. The timing sequence for entering the sleep state is the same as before- the high V_t devices simply turn off after a time T_{min} after the clock goes from low to high, and the master stage stores the proper data. (assuming the helper high V_t devices have transitioned to the proper value before entering sleep mode to avoid race conditions.) However, when exiting from the sleep state, the master latch must enter the active mode half a cycle after the main sleep devices are turned on, as illustrated in the timing diagram below

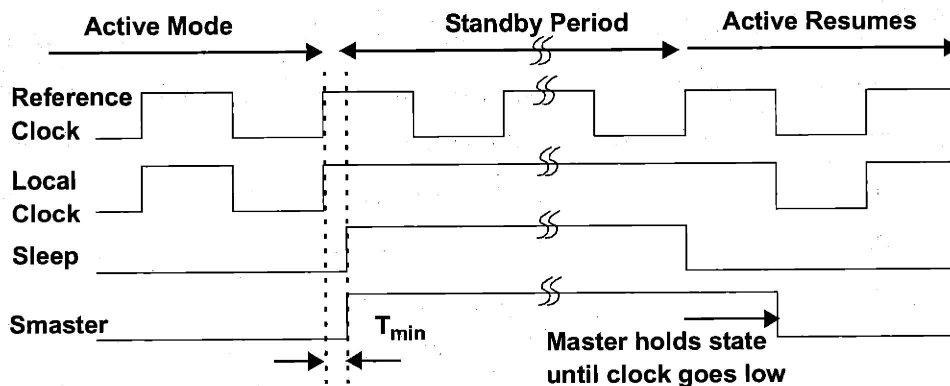


FIGURE 4-30. Sleep/ active mode timing diagram for dynamic leakage feedback FF.

This timing constraint arises because during the standby mode, the memory is stored in the leakage feedback gate and the input node N_1 is free to float. As a result, when the circuit block transitions from the sleep mode to the active mode, one cannot immediately turn on the master latch high V_t sleep transistors because that might accidentally cause the data to flip state since the input could have an incorrect value, which would overwrite the contents of the flip flop. Instead, after the sleep signal is deasserted for the MTCMOS block and slave stage, the master stage should continue to hold the stored values at nodes N_2 and N_3 through the leakage feedback mechanism. Since nodes N_2 and N_3 do not need to transition, it is acceptable to leave the master stage in the sleep state, and simply utilize leakage mismatch characteristics to hold the output data to the proper power supply. Once the clock transitions from high to low, passgate P_2 turns off, data is stored in the slave stage, and the leakage feedback master stage can then be reactivated to perform active switching. Thus, the master stage of the dynamic leakage feedback flip flop utilizes out of phase sleep signals S_{master} and $\overline{S}_{master}$, while the slave and block circuitry uses the standard sleep signals Sleep and $\overline{\text{Sleep}}$.

4.11 MTCMOS Flip Flop Simulation Comparison

Several of the MTCMOS flip flop architectures described in the earlier section were simulated in a 0.14μ technology with high V_t approximately 0.15V and low V_t approximately 0.05V (defined at the 1nA @ 1um point) to verify the performance gains by using leakage feedback gates. As expected, leakage feedback flip flops, with their reduced loading effects, have better performance than using parallel high V_t devices to recirculate data during the standby modes. Below are simulations of flip flop delays ($T_{setup} + T_{CQ}$) for the conventional MTCMOS static flip flop (utilizing a parallel high V_t device in the master stage) of Figure 4-11, for the static leakage feedback flip flop of Figure 4-25, and the leakage feedback dynamic flip flop of Figure 4-29 as functions of the total sleep transistor W/L ratio as a percentage of the total flip flop width.

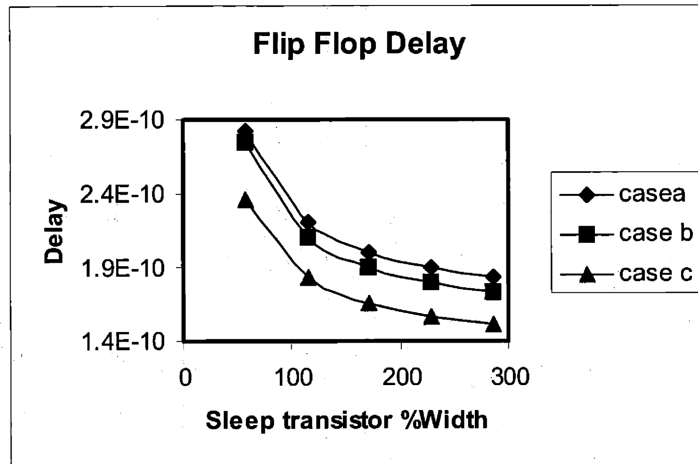


FIGURE 4-31. Flip flop delays (Tsetup + Tcq) for a)MTCMOS static FF, b)leakage feedback static FF, and c) leakage feedback dynamic FF.

As can be seen above, the performance of the leakage feedback static flip flop exceeds that of the conventional MTCMOS flip flop that utilizes a parallel high Vt device to recirculate data during the standby state. Furthermore, the conventional MTCMOS flip flop of Figure 4-11 does not even have CMOS compatible outputs, and providing another parallel high Vt device in the slave stage as shown in Figure 4-12 will result in even further delays. On the otherhand, the leakage feedback static feedback can retain state and utilize CMOS compatible outputs almost for free since no extra loading is introduced in the critical path.

The flip flop with the best performance can be seen to be the leakage feedback dynamic flip flop, as expected. This circuit minimizes all critical path loading by employing dynamic techniques, and thus is well suited for very high performance applications. Even though a standard MTCMOS dynamic flip flop would be slightly faster, the leakage feedback dynamic flip flop can still retain state during the standby state, which enables one to implement an aggressive, yet simple, clock gating/ idling strategy to reduce power dissipation.

Finally, Figure 4-32 shows how leakage currents are successfully reduced in the sleep condition. All three flip flop implementations are shown to significantly reduce leakage currents during standby operation, although the exact amount of leakage reduction is dependent on choice of technology and the selection high V_t and low V_t levels. To the first order, the sleep condition leakage currents correspond to the leakage of an all high V_t implementation.

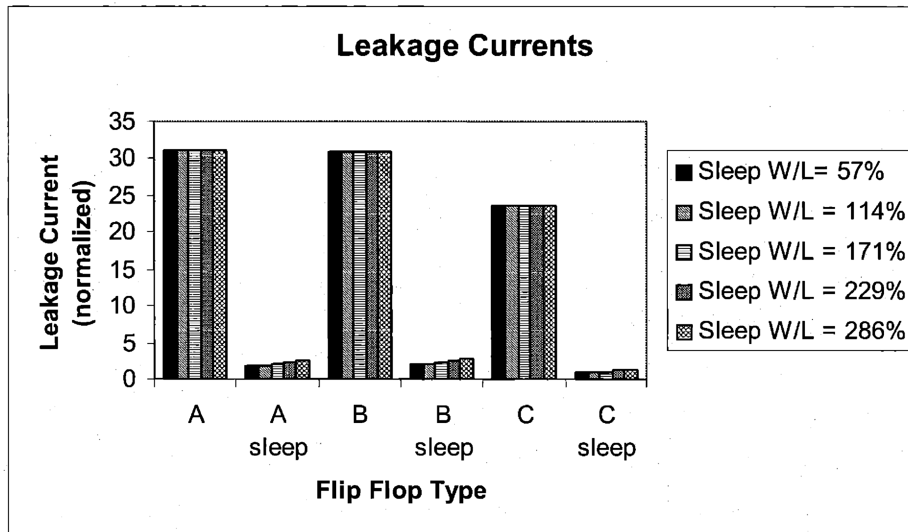


FIGURE 4-32. Leakage current reduction during sleep modes for a)MTCMOS static FF, b)leakage feedback static FF, and c) leakage feedback dynamic FF.

Chapter 5

Variable V_t Techniques - Body Biasing

Dual threshold voltage circuit techniques have been shown to be effective at controlling subthreshold leakage currents especially during standby modes. High V_t devices can be used to reduce leakage currents, while low V_t devices can be used to provide high performance operation. However, as described earlier, new circuit techniques and tools must be developed to effectively utilize these dual V_t techniques. For example, CAD tools need to be developed that optimally parse circuit blocks into high V_t and low V_t gates and still ensure that critical path timing constraints are met. Other tools need to be developed to provide sleep transistor sizing methodologies for MTCMOS blocks, and new libraries and standard cells need to be created to help facilitate MTCMOS synthesis and design methodologies. As a result, a whole of host of design, verification, synthesis, and routing tools will be required in the future to provide the necessary design infrastructure to effectively utilize dual V_t devices. Although it is possible to implement many of these dual V_t techniques in a full custom design environment, new tools will be necessary in order to apply these techniques to larger digital systems.

An alternative approach for subthreshold leakage reduction is to utilize the body effect to directly change the threshold voltage of individual devices through body biasing. This approach requires the use of a triple well technology, or similar advanced technology,

where PMOS and NMOS bodies can be independently biased so that device threshold voltages can be increased during the standby mode to reduce subthreshold leakage currents. Fortunately, this technique is fully compatible with existing CMOS design and tools, and can be applied in a very straightforward manner. Furthermore, being able to dynamically adjust body biases values will enable designers to explicitly tune threshold voltages to help compensate for parameter variations.

5.1 Body Biasing Theory

A typical triple well technology is shown below in Figure 5-1. The main feature, compared to a standard CMOS technology, being that an extra well is used for NMOS and PMOS devices, which allows one to independently control the body contact of individual devices.

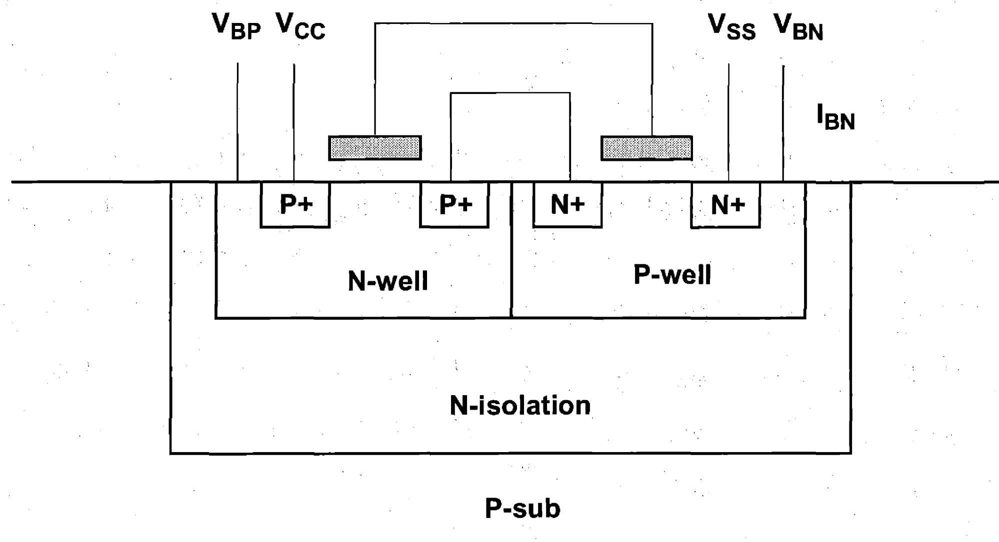


FIGURE 5-1. Triple well technology

By applying reverse bias to the body of the devices, the threshold voltages can be adjusted because of the body effect. For example, biasing an NMOS device body with a voltage

lower than Ground, or biasing a PMOS device body with a voltage higher than V_{CC} will increase the threshold voltage, as illustrated in Eq 5-1.

$$\Delta V_t = \gamma(\sqrt{2|\phi_p| + |V_{sb}|} - \sqrt{2|\phi_p|}) \qquad \text{(EQ 5-1)}$$

where γ is the body-effect parameter given by

$$\gamma = \frac{\sqrt{2\varepsilon_s q N_a}}{C_{ox}} \qquad \text{(EQ 5-2)}$$

For typical body factor values in modern technologies, a 100mV change in the body bias will result in approximately 20mV change in V_t. For example, if 500 mV reverse body biasing is applied to a circuit during the standby mode, then the V_t will change by approximately 100mV, which results in approximately a 10x reduction in leakage currents. One issue however is that the body factor tends to degrade with technology scaling, so body biasing may be less effective in future technologies.

With a body biasing methodology, and a suitable triple well technology, it is very straightfoward to place a circuit block into a low leakage standby mode. One simply needs to apply maximum reverse body bias to all the PMOS and NMOS devices, which will raise the device threshold voltages and lower the subthreshold leakage current exponentially. This approach to subthreshold leakage reduction does not use any multiple threshold devices or extra series power switches, but instead directly manipulates the intrinsic threshold voltages of the existing devices in the circuit. As a result, there are no sleep transistor sizing issues or gate partitioning tools that need to ensure correct operation during the active mode. Instead, circuits have the exact same structure as in a conventional CMOS implementation, and thus can be designed, verified, and tested using existing CAD tools and techniques. The only modification to an existing CMOS design is that standby circuitry must be provided to supply appropriate bias voltages (greater than V_{CC} and less than ground) to be the device body terminals.

During the standby state, all circuits are still connected to power and ground and are actively driven, so memory circuits will continue to hold their values, and there will be

no risk of floating nodes. In fact, during the standby mode, circuits behave the same as before, except that the devices will have higher threshold voltages, and as a result have slower performance but lower subthreshold leakage currents as well. As a result, switching a block into a standby mode is straightforward, and simply changes the transistor characteristics rather than altering any circuit functionality. Furthermore, if a clock gating strategy is already used to lower dynamic power during idle modes, it is simple to extend this to a deeper sleep condition by applying reverse body bias and reducing subthreshold leakage currents as well.

Body biasing approaches can also be used to help control subthreshold leakage currents in the active mode as well, just like with the dual V_t partitioning scheme described in the background chapter. The basic idea is to use slower device for non critical paths because they do not limit the chip performance. The only difference is that instead of using explicit high V_t devices, the non critical gates can be directly reverse biased to shift threshold voltages higher. One drawback of this approach is that the reverse bias signals must be independently routed to distinct body wells for these non critical gates, whereas a dual V_t approach would simply involve a different implant processing step. As described earlier though, the number of noncritical gates in a well balanced circuit may be limited, which could limit the effectiveness of this leakage reduction technique during the active modes.

5.2 Control of parameter variations

Although body biasing can be an effective and simple way to reduce subthreshold leakage currents during the standby state, and to some extent in the active state, the true benefit of body biasing is that the threshold voltages can be dynamically controlled during runtime. Instead of just switching between no body bias or maximum reverse body bias, it is more useful to be able to selectively choose a body bias value that can be used to actually fine tune threshold voltages to meet performance and leakage specifications[37][38][39].

As technology scales, variations in threshold voltage will increase due to worsening short channel effects, and can pose a serious limit to V_{CC}/V_t scaling [40][41][42]. The effect of V_t variation, (which in modern technologies can range +/- 50mV for 1 sigma) on

circuit performance is amplified at low V_{CC}/V_t levels. Eq 5-3 below shows how propagation delay is impacted by supply voltage and threshold voltage values.

$$T_{pd} = K\frac{CV_{CC}}{(V_{CC}-V_t)^{\alpha}} \tag{EQ 5-3}$$

Differentiating with respect to V_t yields

$$\frac{\partial T_{pd}}{\partial V_t} = K\frac{\alpha CV_{CC}}{(V_{CC}-V_t)^{\alpha+1}} \tag{EQ 5-4}$$

dividing both sides by T_{pd} yields

$$\frac{\partial T_{pd}}{T_{pd}} = \frac{K\frac{\alpha CV_{CC}V_t}{(V_{CC}-V_t)^{\alpha+1}}(\partial V_t/V_t)}{K\frac{CV_{CC}}{(V_{CC}-V_t)^{\alpha}}} \tag{EQ 5-5}$$

which simplifies to

$$\frac{\partial T_{pd}}{T_{pd}} = \frac{\alpha(\partial V_t/V_t)}{V_{CC}/V_t-1} \tag{EQ 5-6}$$

This equation shows how percent variation in propagation delay depends on the variation in threshold voltage. As can be seen in Eq 5-6, the percent variation in propagation delay is proportional to the percent variation in threshold voltage, but is also amplified when V_{CC} is close to V_t. In earlier designs with supply voltages much larger than threshold voltages, the variation in V_t had a smaller impact on overall performance. However, with more aggressive V_{CC}/V_t scaling, the effects of V_t variation on circuit performance cannot be ignored. As technology scaling continues, the percent variation in V_t, or dV_t/V_t, will itself become worse as short channel effects worsen, and percent variations in critical dimensions and oxide thickness increase. As a result, unless technology processing is improved, delay spreads will become increasingly large as one continues to scale supply

and threshold voltages. V_t variation can thus be a limiting factor to future technology scaling.

However, with the ability to tune threshold voltages through body biasing, it is possible to tighten the distribution of threshold voltages and delay variations. Figure 5-2 below shows a possible distribution of critical path delays for a sample of dies both before and after reverse body biasing adaptively is applied to the samples. With application of reverse body biasing, fast samples are slowed down to the target frequency, which saves active leakage power. As described earlier, standby leakage power can also be reduced by merely applying maximum reverse body bias during the idle modes.

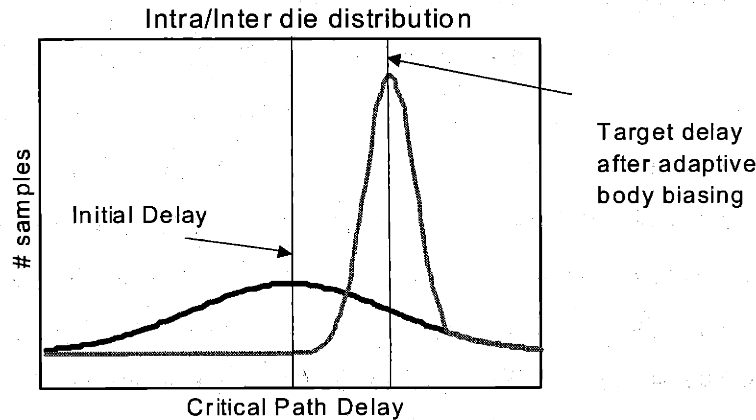


FIGURE 5-2. Body biasing to tighten distributions.

The distributions above are shown for a hypothetical sample of dies that takes into account both wafer to wafer and chip to chip variation components. Because of potentially large spreads in chip performance, the target operating speed would have to be much slower than the average chip delay in order to achieve a reasonable yield figure. Thus, an adaptive body biasing approach to slow down fast chips so they operate only as fast as necessary can help reduce active leakage power. Another strategy that can be employed is to simply bin the samples into fast and slow chips, which are then separated and used for fast or slow parts as is commonly done with microprocessors. However, in some applica-

tions binning might not be an option and an adaptive body biasing approach would be more appropriate.

5.3 Adaptive Body Biasing

One way to use adaptive body biasing compensate for parameter variations is to utilize a delay feedback mechanism that automatically develops the appropriate reverse body bias needed to maintain a fixed delay criteria. A replica of the critical path for the chip is used as a matched delay line, and the body bias is adjusted in a feedback loop until the clock period matches that of the critical path. The body bias generators can be configured to provide different ranges of reverse body bias, which will subsequently set the control range achievable with an adaptive body biasing scheme[43].

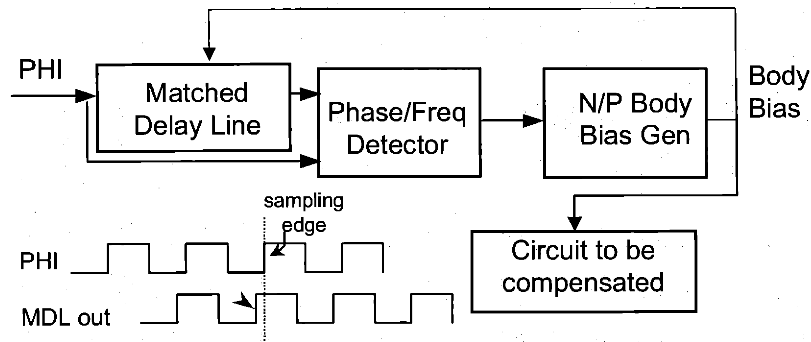


FIGURE 5-3. ABB block diagram.

The matched delay line uses the exact same transistor sizes, device orientation, and gate structures as those found in the actual critical path of the circuit to be compensated, so it will closely track performance for different operating conditions including supply voltage, body bias value, and temperature. Furthermore, if the matched delay line is physically close to the critical path circuitry, and the dominant component of parameter variations are from wafer to wafer or die to die variations, then the critical path replica and actual critical path operation will track very closely. As a result, by adjusting the body biases until the matched delay line has the same latency as the clock period, the chip will be compensated to operate only as fast as necessary, which helps reduce active subthresh-

old leakage currents. The ABB approach thus can help reduce active leakage currents in the sense that parameter variations that cause circuits to be faster than necessary can be slowed down.

As described earlier, further reductions in active leakage currents can also be achieved by slowing down non critical gates also. These non critical path devices can be explicitly set to higher threshold voltages, and do not need to be tuned in conjunction with the adaptive body biasing compensation values. This is because the delay feedback mechanism only needs to tune the critical path threshold voltages in order to match the target operating frequency, and those gates that are not in the critical path can have higher static threshold voltage since they do not limit circuit performance. However, one must be sure that these high V_t non critical paths are not slowed down so much that they start to become critical, especially in the presence of parameter variations.

5.3.1 Implicit parameter variation tuning

As described earlier, one of the problems with aggressively scaled technologies is that threshold voltages can vary as a result of process variations and worsening short channel effects, resulting in large subthreshold leakage currents during the active period. However, rather than adjusting body biases to tune threshold voltages directly, the adaptive body biasing framework shown in Figure 5-3 uses a feedback strategy that closes the loop around the performance of a matched delay line. This is in contrast to other research work such as [38], which attempt to directly compensate for V_t variations by measuring leakage current through a representative device on the chip. However, work by [44] on the otherhand uses a very similar approach to the matched delay line technique presented in this research. Our research first explored the use of tuning threshold voltages based on a matched delay line in [43], which was applied to dual gated SOI (SOIAS)[45].

By compensating for performance, rather than for a measurement of the threshold voltage directly, adaptive body biasing will implicitly, rather than explicitly, tune out threshold voltage variations. However, compensating for performance is beneficial because the final criteria in a functioning chip is performance rather than threshold voltage. As a result, applying the appropriate reverse bias to match the delay line with the tar-

get clock frequency will ensure that the chip functions properly at the target speed, yet will also skew the threshold voltages as high as possible to reduce subthreshold leakage currents.

By closing the feedback loop around the performance of the critical path, the adaptive body biasing strategy also compensates for other components that cause delay variations. For example, temperature fluctuations, mobility degradation, and even power supply variations (if slow enough) can all be compensated by adjusting body biasing to ensure that the chip operates exactly at the target frequency. Although the distribution of critical path delays in a sample of dies would be tight, it is possible for the distribution of threshold voltages to be somewhat larger due to compensation of other variation components. Still, the chips will be compensated such that the threshold voltages are chosen to give the lowest possible leakage currents for the given target frequency and process and environmental conditions.

5.3.2 2 dimensional matching considerations

The adaptive body biasing scheme relies on controlling both PMOS and NMOS body terminals. Because this is essentially a two dimensional control problem, there exists an infinite number of V_{BP} and V_{BN} values that will still satisfy the performance criteria of the matched delay line. For example, a fast PMOS and slow NMOS implementation, or a slow PMOS and fast NMOS combination, or anything in between can still satisfy the feedback loop. One simple solution is to simply bias the body terminals by the same amount for both the PMOS or NMOS terminals, and to effectively convert the biasing function into a one dimensional control problem. A more advanced strategy could shift the V_{BP} and V_{BN} bias values at different rates to account for different body factors between the PMOS and NMOS devices. However, in general the PMOS and NMOS devices will track reasonably well in a good process, so this is not necessary. Figure 5-4 below shows how I_{ds} varies with applied body bias for both PMOS and NMOS devices in an aggressive

0.14μm triple well technology with $V_{tn} = |V_{tp}| = .05$ (defined as V_{gs} = 1nA for a 1μm device).

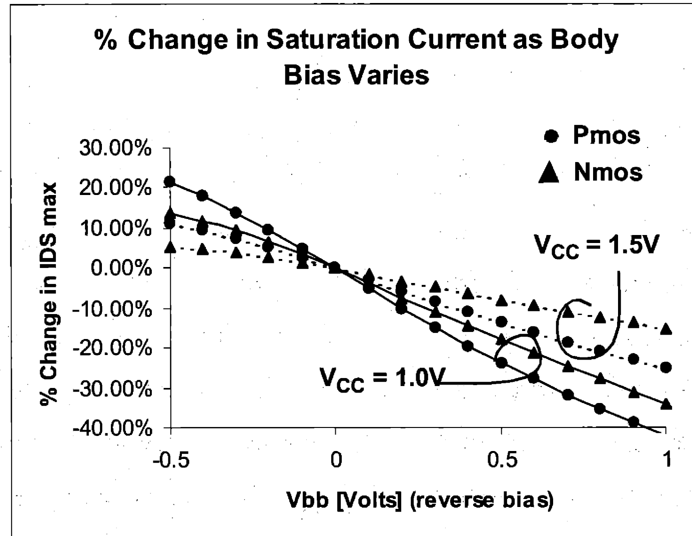


FIGURE 5-4. Percent deviation in saturation currents as function of PMOS and NMOS body bias.

Curves are shown for both V_{CC}=1.0V and at 1.5V, and illustrate how body biasing will effect the percent change in maximum saturation currents. Ideally, the effect of body biasing should have the same impact on both PMOS and NMOS devices, but deviations arise because of slight differences in body factor and device parameters. The divergence between the PMOS and NMOS curves can be attributed to differences in body factor, which cause more deviations as body bias is applied. Nonetheless, one can see that mismatches are within 10% for as much as 1V of reverse body bias. The curves also show that at higher supply voltages, the effect of reverse body biasing on maximum device currents is reduced. This is simply due to the fact that at larger supply voltages, the effects of modulating threshold voltages is less. In the context of an adaptive body biasing controller, it is thus reasonable to simply adjust both PMOS and NMOS body biases by the same amounts when trying to speed up or slow down a circuit. Since both currents tend to change by similar amounts, the rise and fall time behavior of the compensated circuitry will track reasonably well, and it is unlikely that one type of device will develop drastically different switching characteristics than the other. As a result, tuning PMOS and

NMOS body bias voltages in unison reduces a 2 dimensional control problem into a 1 dimensional problem.

After device fabrication, often times there are systematic offsets between PMOS and NMOS devices because of variations in processing or mismatches within the circuits themselves. By modifying the adaptive body biasing scheme of Figure 5-3, it is possible to compensate for this fixed difference between PMOS and NMOS threshold voltages by maintaining a fixed offset between the V_{BP} and V_{BN} bias values. For example, if the body bias generators are implemented as simple D/A converters, then one can easily maintain a fixed digital offset between the two digital words, which are subsequently converted to analog V_{BP} and V_{BN} voltages. One way to select the proper offset between the PMOS and NMOS body terminals is to provide extra circuitry that can measure the saturation currents between two reference devices. The adaptive body biasing circuitry can then be broken down into a two step process, where the first step is to perform a calibration where the faster device (PMOS or NMOS) is slowed down through body biasing until the saturation currents match. Next, this built in body biasing offset is simply stored and continuously applied after the matched delay feedback loop is activated and incremental changes to body bias values are applied equally to both types of devices.

5.3.3 Adaptive body biasing vs. DVS for compensating parameter variations

Instead of adaptively adjusting body bias values to compensate for parameter and process variations, an alternative approach is to tune supply voltages (also using a matched delay line approach) to speed up or slow down circuits as necessary so they operate at the target frequency. In this regard, if threshold voltage variations cause devices to be too fast and leaky, one can lower the supply voltage to compensate for these variations[46][47][48]. At high V_{CC} and high nominal V_t circuit operation, it actually turns out to be more energy efficient to drop supply voltages wherever possible rather than using reverse body biasing to increase threshold voltages back to their nominal values. This is because the optimum energy operating point lies at nominally lower supply voltages and lower threshold voltages, so the effect of threshold lowering variations is actually good for reducing power. As a result, any parameter variations that tends to speed up devices by lowering threshold voltages would actually be beneficial because supply voltages could be dropped. How-

ever, if parameter variations tend to slow down devices by increasing threshold voltages, then increasing supply voltages to maintain performance would tend to worsen the power efficiency. In this case, using a body biasing approach with forward biasing (described in later sections) to speed up devices would be more energy efficient.

As supply and threshold voltages continue to scale though, using a V_{CC} modulation approach to tune out parameter variations becomes less useful. This is because at the optimal V_{CC}/V_t operating point, if one were to modulate V_{CC} to compensate for V_t variations, the energy efficiency would decrease. Instead, it makes more sense to directly tune out the effects of any threshold voltage variations by adjusting the device body bias values directly. As technology scales, threshold voltage variations become increasingly dominant, so schemes that directly compensate for V_t variations are more effective. Even though other types of parameter variations might be more efficiently compensated by modulating V_{CC}, threshold voltage variations are likely to be more dominant so that adaptive body biasing approaches that directly tune V_t's are better.

From a practical point of view, adaptive body biasing can also be better than using a dynamic voltage scaling scheme to tune out the effects of parameter variations. First of all, a dynamic voltage scaling approach requires a more complicated variable power supply scheme, while an adaptive body biasing approach can more easily bias device body values because of limited current requirements. Another area were ABB approaches are more flexible are when multiple bias regions are utilized on a single chip. As described in the next section, using multiple ABB generators within a single chip can be much more effective at controlling overall variations than tuning the chip as a whole. However, a variable supply scheme can be costly to implement for multiple regions within a chip, and furthermore, interface issues would arise when a logic block operating with one supply voltage domain communicates with a different domain. On the otherhand, an ABB approach can easily be applied to multiple regions within a chip, and the bias voltages can be generated more easily.

5.4 Within die adaptive body biasing

Adaptive body biasing has been described in the previous section for die level control. In otherwords, the whole chip was adaptively biased higher or lower to equate the matched delay line with the target frequency. This is a useful technique if wafer to wafer and die to die variations are dominant, and intra die variations are negligible. However, as technologies continue to scale, chip sizes increase and intra die variation components can become large as well. From a fabrication point of view, inter-die (lot-to-lot, wafer-to-wafer, die-to-die) variations have been more important to control in the past because this is a direct consequence of manufacturing accuracy. For example, the resolution of lithography equipment, accuracy of aligners, or quality of a CMP process can all result in imperfections that manifest themselves as inter-die variations. However, from a circuit point of view, intra-die variations are becoming important as well because these directly effect the local matching between parameters in a single chip and can ultimately limit yield and performance results. As described earlier, one way to address inter-die variations is to use a binning strategy where individual chip samples can be placed into nominally fast or nominally slow parts and used for different applications. However, in a large chip where each internal island can have significant die level variations, it is not possible to subdivide the chip into fast or slow local regions. Instead, the chip can only function as fast as the slowest part of the chip, and thus intra-die level variations can put a fundamental limit on yield and active leakage currents for a target operating frequency.

Intra-die variations can become larger as technologies continue to scale and short channel effects become larger. The act of reverse body biasing itself can actually worsen short channel effects as well and contribute even more to intra die variations. For example, the die level adaptive body biasing approach described earlier could tighten the distribution of average chip delays by applying the appropriate bias values to a die in order to shift the critical path to the target frequency. However, the biased region within the chip itself suffers from worsened short channel effects, which amplifies threshold voltage variations. Thus, the act of reverse biasing itself, which is meant to tighten variation distributions from chip to chip, tend to worsen the variations within the chip itself[49]. As a

result, for larger dies and aggressively scaled technologies, intra die variations cannot be ignored.

5.4.1 ABB for compensating intra die variations

In order to address intra-die variation issues, it is useful to extend the adaptive body biasing technique described earlier so that a chip can be subdivided into multiple regions that are individually biased. Assuming that local devices are spatially correlated, then applying multiple adaptive body biasing generators on a smaller intra-die scale can reduce overall chip delay variations. Although intra-die variations do exist, devices that are close to one another are usually correlated, while devices that are far apart are less correlated and may exhibit systematic variation components. This fact is exploited quite heavily in analog circuit design where local matching and symmetric layout techniques are used extensively to reduce mismatches between devices. Likewise in a digital system, if a matched delay line for a particular subblock of a chip is placed closely and oriented similarly to the internal critical path, then it will closely match parameter variations as well. Thus, if a large system such as a system-on-a-chip can be broken down into multiple "islands" that are each characterized by a matched delay line that closely tracks the local critical path, then it can be beneficial to separately tune each "island" with a local body biasing generator to tighten delay distributions throughout the chip. Individually tuning these separate blocks to run exactly at the target frequency will significantly reduce the active leakage current components. For example, if a chip is composed of several complex functional blocks, like in a system-on-a-chip, then it might be possible to simply break down the chip into "islands" according to the functional block, each characterized by an appropriate local critical path. Figure 5-5 shows a hypothetical example of a die that can be broken into N blocks, each of which exhibit a smaller distribution (for a sample of

dies) for the local block critical path delay. In this case, tuning each region independently to the target frequency is more efficient than tuning the die as a whole.

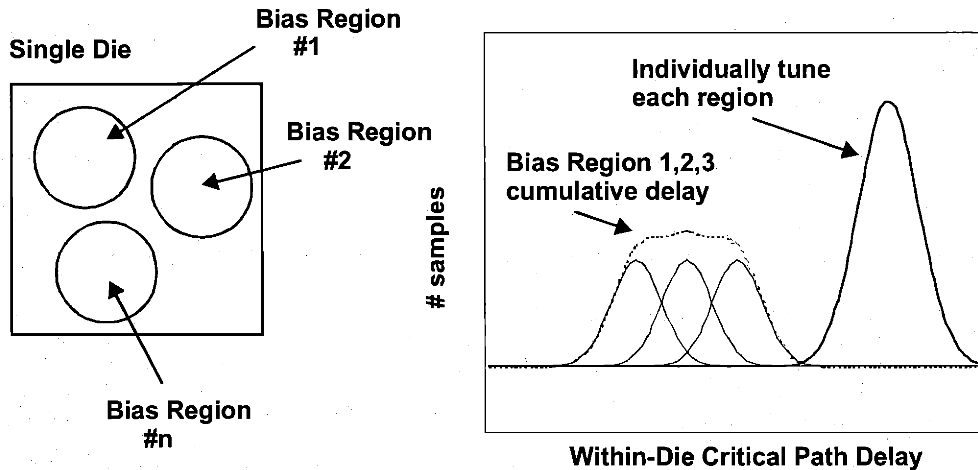


FIGURE 5-5. Within die body biasing scheme tightens overall distribution

5.4.2 Intra die variations and local matching criteria

Adaptive body biasing is useful only if local parameter variations can be tracked with a matched delay line (so that the frequency of the matched delay lines closely track the frequency of the local block). However, if the intra-die variations in a chip are dominated by a purely random component that overshadows any spatial correlations, then adaptive body biasing may not be applicable because the variation between local devices would be uncorrelated and could not be tracked with a matched delay line. For example, as devices become smaller and smaller, random dopant placement can become an increasingly large factor in threshold voltage variations because the device volumes are so small that device characteristics can be significantly effected by a small number of dopant atoms[50][51]. This random process for different devices is thus uncorrelated and will not have any spatial correlation component, so even local device threshold voltages might vary significantly.

However, for modern circuits, purely random dopant variations are not the dominant source of intra-die threshold voltage variations. Instead, other sources such as oxide

thickness, device orientation, channel length, and doping variations can all contribute more to threshold voltage variations, and these parameters can exhibit local correlations or systematic variations across the chip[52][53][54][55]. Parameter variations other than device threshold voltage will also contribute to frequency variations between local islands. Mobility, device geometries, interconnect geometries, and ILD thickness are a few chip parameters that are likely to match between a block critical path and the associated matched delay line, yet can exhibit large fluctuations across a chip. For example, local pattern dependent layout and density variations can easily effect ILD thickness during CMP (chemical mechanical planarization) and result in a systematic intra-die variation component that differs across the chip. Wafer level variations can also create systematic offsets in chip parameters that have a gradual gradient across an entire wafer that changes across the die as well. Finally, variations in chip temperature can also play a very large factor in threshold voltage variations and local critical path performance variations across a large die.

These variation mechanisms can create large differences between local islands, but still provide local matching between the block's critical path and associated matched delay line. Even if there is some local mismatch between the matched delay line and the local critical path, this would likely still be small compared to the mismatch between separate local islands. Another component of systematic offsets between local islands within a system on a chip can be attributed to different types of critical paths in different local blocks as well.

The combination of different parameter variations throughout the chip will manifest themselves as fluctuations in local critical path frequencies. The adaptive body biasing scheme, if applied to individual local islands, will thus be able to compensate for these delay variations by tuning body biases until a target frequency is achieved. Although not directly reducing parameter variations themselves, the body biasing scheme instead will tune threshold voltages to compensate for the impact of different parameter variations on overall critical path delay. As described earlier, this is useful because threshold voltages will naturally be shifted as high as possible (slow devices as much as possible) to exactly meet the target frequency, thus reducing active leakage currents.

5.5 Qualitative Benefit of Within Die Adaptive Body Biasing

Intra die parameter variations can typically arise from several different mechanisms. In one case, the variations are purely random and independent, in which case adaptive body biasing is not useful. Fortunately, as described earlier this is not the dominant variation component in modern technologies. If on the otherhand the parameter variations are correlated on a local scale, but are uncorrelated between distant regions on a chip, then local adaptive body biasing can be beneficial. Matched delay lines will be able to track local critical paths, and separate regions can be independently tuned to optimize the threshold voltages in individual blocks to ensure that circuits operate only as fast as necessary. A final type of parameter variation can arise from systematic mechanisms that are deterministic, rather than random, in nature. Nonetheless, these systematic variations often times are functions of local positioning and orientation, so typically the matched delay line will closely track the behavior of the local critical path. In many cases, these systematic components are a dominant form of intra-die variations, and thus can again benefit from an adaptive body biasing approach.

Although systematic variation components theoretically can be derived deterministically, often times there is not enough information available to characterize these variations a priori. For example, systematic ILD variations due to layout dependencies on CMP processing may not be modeled accurately, machine misalignment variations that give rise to wafer level variations may not be fully understood, or temperature gradients across a chip may not be fully understood. Thus, although the variation characteristics for a sample of dies might result in a systematic variation component, one might not be able to predict this variation beforehand. In order to characterize the benefits of adaptive body biasing on a representative die, it is useful to simply abstract these multiple variations components into a single random variable that captures the uncertainty associated with local critical path delays within different parts of a large chip.

Figure 5-6 below shows a hypothetical chip that is broken into N "islands" that can be separately biased to tighten delay distributions. Each region is characterized by a local matched delay line that represents the critical timing path of the local block.

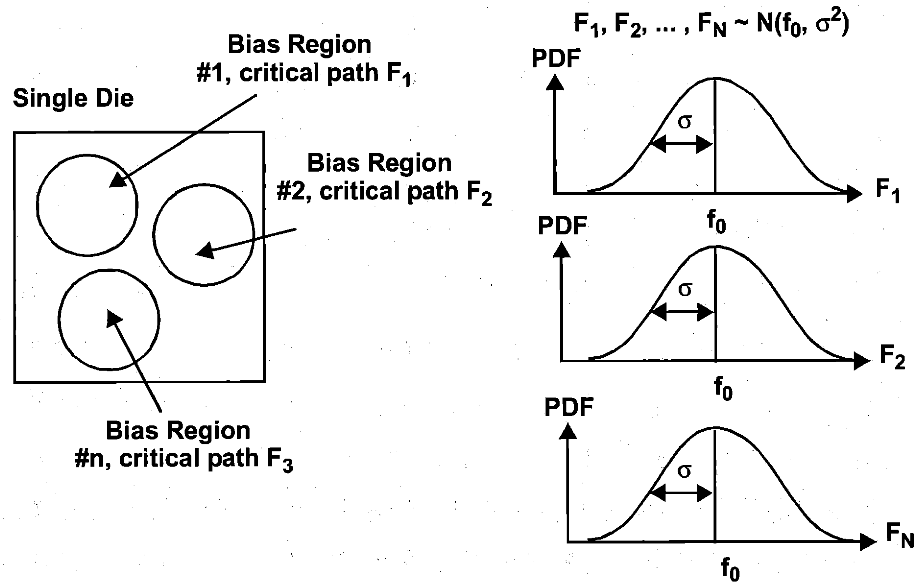


FIGURE 5-6. Statistics of each local critical path of each island modeled with normal distribution.

For each of the N regions, the operating frequency of the local critical path (before body biasing) can be represented by random variables F_1, F_2, ... F_N, which to the first order can be modeled as independent, identically distributed gaussian random variables characterized by mean f_0 and standard deviation σ. By assuming no knowledge of the types of circuits used, or any systematic variation components, it is useful to simply lump all possible variations into a normal distribution $\sim N(f_0, \sigma^2)$ and to assume that the frequency of each matched delay line is close to the local block critical path speed.

These frequency random variables implicitly model the impact of random variations in threshold voltages and other delay parameters for each local island. Furthermore, these random variables can be interpreted as modeling only intra-die variation components rather than wafer-to-wafer or chip-to-chip variations. This subset of variations is important to look at because intra die variations cannot be compensated for by using a binning strategy, whereas chip and wafer level variations can utilize a binning strategy to

classify individual chips. Instead, a chip can only function as fast as the slowest part of the chip, and thus die level variations can put a fundamental limit on yield and active leakage currents for a target operating frequency.

5.5.1 Impact on yield and performance

Given that a complicated system on a chip can be abstracted into N different bias regions, each characterized by a random variable F_i, then it is useful to estimate the impact of variations on yield and on active leakage currents. As described earlier, the random variables for the local "island" frequencies are modeled as identically distributed, independent random variables, which simplifies the model for how variations can impact chip behavior.

If a chip has N local regions each with frequency F_i, then for the chip yield to be greater than 90%, the target operating frequency would have to be chosen such that there is a 90% chance that every local island will be functional. This simply corresponds to choosing a target operating frequency F_t such that

$$Prob((F_1 > F_t) \cap (F_2 > F_t) \cap \dots \cap (F_n > F_t)) > 90\% \qquad \text{(EQ 5-7)}$$

Because random variables F_1, F_2,... F_n are assumed independent, and identically distributed, the probability of having a yield greater than 90% can be simplified to

$$(Prob(F > F_t))^n > 90\% \qquad \text{(EQ 5-8)}$$

which implies that the target operating frequency must be chosen such that

$$Prob(F > F_t) > (90\%)^{1/n} \qquad \text{(EQ 5-9)}$$

Considering only intra die variations, and assuming uniformly distributed frequencies for each local island, we can see that there is a significant speed penalty to ensure that the chip meets a reasonable yield figure. This can be intuitively understood because if chips are increased in size with more "islands," the risk of the chip having a slower block

will continue to worsen, and the chip can only operate at this worst case operating frequency. Because the target frequency for a complex system-on-a-chip has to be significantly lower than the average speed for each local region, a typical chip consisting of several local "islands" would very likely have several regions operating much faster than necessary. With low V_{CC}/V_t technologies, these unnecessarily fast regions will have smaller threshold voltages, and can result in large excessive leakage currents. Theoretically, the expected leakage power for the chip can be calculated as well, which will show that large amounts of leakage currents are wasted because there is a finite probability of chip regions operating much faster than necessary, since the chip operating speed is set by the slowest block.

However, by using an adaptive body biasing scheme with reverse biasing control like the one described earlier for die-level compensation, it will be possible to significantly reduce active leakage currents by explicitly slowing down all local regions so they operate only as fast as necessary. The frequency variable distributions after adaptive body biasing for example will show a much tighter distribution at the operating point. Although application of reverse body bias will not change the yield of this distribution of chips, the expected active leakage power can be significantly reduced with this technique.

5.5.2 Critical path systematic variations

For illustrative purposes, the frequencies of each local island for the previous example were considered to be independent and identically distributed random variables. In actuality, the behavior of different local islands within a chip might be very different. The random variations in frequencies in separate regions might be correlated, the probability distributions might differ, or there might be a systematic frequency offset between different regions. Systematic variations can be a large part of intra die variations due to wafer level and pattern dependent variations. As a result, even if random variations between local islands are small, systematic variations between blocks can result in large frequency discrepancies across a chip. In the earlier analysis, variations in local critical path frequencies were assumed to be caused by systematic and random variations in circuit parameters such as threshold voltages, mobility, oxide thickness, interconnect geometries,

etc. As described before, these parameter variations can result in large deviations in performance as one continues to scale supply and threshold voltages in future technologies.

However, a particularly important source of systematic variations between individual blocks in a large system (like a system-on-a-chip) are the local critical paths themselves. In the previous section it was assumed that local "islands" were characterized by nominally equivalent critical paths that all need to satisfy the global chip timing requirements. In an actual circuit implementation though, the critical path for each local "island" is not necessarily a critical path for the global chip. Instead, some functional blocks might be faster than others, and only some of them will have global critical paths that limit the operating frequency of the whole chip. For example, a large chip might be comprised of several different modules where some of the modules might have long critical paths and operate slowly, while other modules might have very short critical paths and complete computations at a much faster rate than the chip core frequency. As a result those regions that are significantly faster than necessary are ripe candidates for adaptive body biasing, where device threshold voltages are automatically tuned higher to slow down the block and minimize active leakage currents. In many systems, large systematic offsets due to differences in local critical path structures themselves can be the dominant source of delay variations between different parts of a chip and can possibly drown out the effects of intra die parameter variations.

Even though there are many underlining sources for how critical path delays vary between regions of a large chip, it is still useful to model the nominal frequency of each local critical path as a random variable like before. However, the random variables no longer are simplified as being independent and identically distributed. Instead a more accurate model for frequency variations should take into account random and systematic parameter variations, correlations between local regions, and systematic offsets attributed to local circuit critical path implementations. As a result, a chip that is divided into N local islands will be characterized by frequency random variables F_1, F_2...F_N that could be correlated and could have means whose offsets can be large compared to the standard

deviations. The frequency random variables thus take into account how different regions within a large chip have different fundamental critical delays.

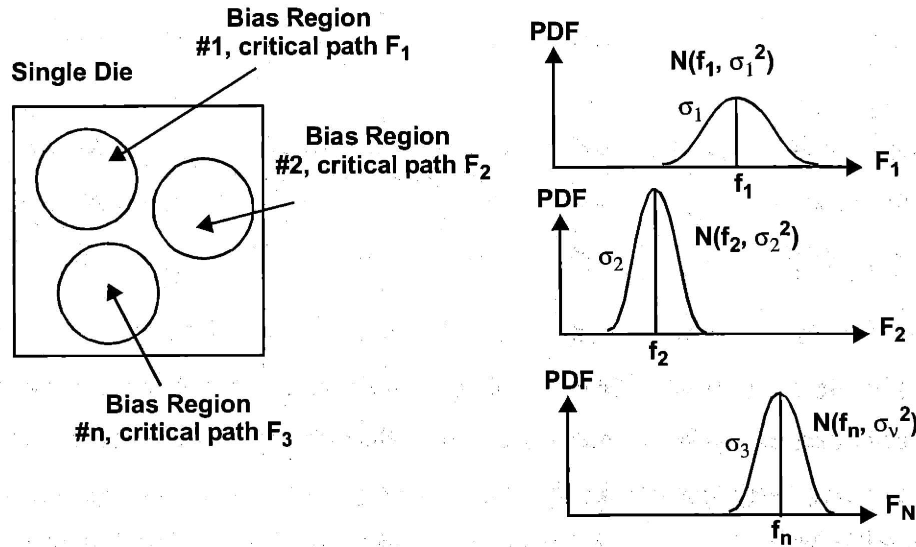


FIGURE 5-7. Statistics for local critical path frequencies with different distributions and systematic offsets.

Similar to the case for IID gaussian frequency distributions described earlier, the chip target operating point must be chosen to satisfy a certain yield requirement. For example, to ensure a yield of 90%, a target frequency must be chosen such that the probability that all blocks within the system will operate faster than the target frequency is also greater than 90%.

$$Prob((F_1 > F_t) \cap (F_2 > F_t) \cap \dots \cap (F_n > F_t)) > 90\% \qquad \text{(EQ 5-10)}$$

However, because the random variables F_1, F_2,...F_N can be dependent, have different distributions, and can have systematic offsets, it is not possible to provide a generic solution like before. (For example, joint probability distributions would be necessary.) Intuitively though, with this framework the chip yield is limited by the slowest parts of the chip. On the otherhand, those frequency variables corresponding to very short critical paths, the expected target frequency is so high that the probability that these regions functions properly is virtually guaranteed. However, like before, the target frequency will

probably be significantly lower than the expected frequencies of those critical regions to take into account parameter variation impact.

For a fixed yield requirement, a large portion of the chip will be operating at speeds much faster than necessary. This can be a result of simple variation differences between different blocks, or can be due to the simple fact that some blocks are systematically more complex than others in a large system. However, having large offsets in operating frequencies between different chip regions is precisely the scenario where a multiple adaptive body biasing strategy can be most beneficial. Blocks which are operating faster than necessary can be adaptively biased so that they are slowed down to the target frequency and active leakage currents are reduced. Again, at low V_{CC}/V_t operating points, these savings in leakage currents can be quite significant. Even if the random variations between threshold voltages of local islands is reduced with improved processing and technologies, the frequencies can still exhibit large systematic offsets because of local critical path differences. Thus in large system-on-a-chip designs, there is much potential to apply adaptive body biasing to slow down unnecessarily fast regions. In effect, non critical blocks can be reverse biased so that their performance more closely matches the target operating frequency. This is yet another reason why it is better to close the adaptive body biasing loop around frequency rather than threshold voltage directly, because in certain regions of a chip having significantly slower devices is desirable. Even if the parameter variations between two different blocks is small, there might be an opportunity to apply reverse body biasing if one block had a much shorter critical path than the other.

5.5.3 Compensating for systematic variations

As described earlier, one of the benefits of the adaptive body biasing strategy is that the body bias values are automatically set through the feedback control. Therefore, parameter variations, unknown delay variations, and time varying temperature changes can all be automatically compensated during the chip runtime even though these deviations may be difficult, or impossible to predict beforehand. Systematic variations in frequency due to critical path circuit variations theoretically can be modeled beforehand and corrected for in an open loop fashion, but can still be difficult to predict accurately during the design phase. On the otherhand, a closed loop adaptive body biasing scheme will automatically

tune the threshold voltage to optimize the local block variables for the target frequency and also dynamically adjust for time varying or random parameter variations. With a closed loop approach, there is no need to model or predict parameter variations for a random process, because for each sample of the random experiment, the adaptive body biasing technique will provide custom compensation necessary to meet target frequencies. As long as the matched delay line matches reasonably well with the local critical path, the adaptive body biasing methodology applied at the local block level could dramatically improve delay distributions in a chip in a very straightforward manner.

While the previous analysis was geared towards adaptively biasing local block regions so that the local critical paths meet the specified target delays, it is also possible to systematically slow down non critical gates within each block to further reduce active leakage currents. This is basically an extension of the gate partitioning technique described for die level ABB control, that is simply applied to the local block scale. Even though the blocks threshold voltages are tuned to meet a fixed target speed, not all gates in the block need adaptively back biased because the non critical paths do not limit performance.

5.6 Forward and Reverse Body Biasing

Reverse body biasing will allow one to tune threshold voltages to slow down devices such that a chip's critical path operates only as fast as necessary. Although leakage currents can be reduced with this technique, it is not possible to improve yields because nothing can be done with those circuits that are too slow. A simple modification to the adaptive body biasing methodology described earlier is to enable some degree of forward body biasing as well, which can actually speed up circuits and improve yields. This can be as simple as adjusting the N/P body bias generators to range from a slightly positive bias voltage to a negative voltage for NMOS bodies and from bias voltages slightly below V_{CC} to higher voltages for PMOS bodies

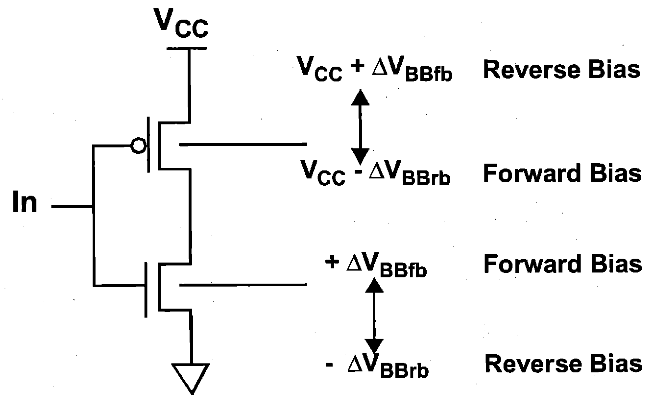


FIGURE 5-8. Forward and reverse body bias ranges.

The limitation however is that the forward bias amounts must be small enough so that junction diodes do not turn on. This limits the forward bias range to be below the PN diode built in potential, giving an acceptable forward bias range of about 500mV. Forward bias has been shown not only to improve performance, but to also reduce short channel effects[56]. The adaptive body approach using a matched delay line thus can easily provide reverse or forward body biasing. In essence, it is just an arbitrary distinction between a continuum of body bias settings applied to devices in order to match a target performance.

5.7 Adaptive Body Biasing Test Chip

A test chip to explore adaptive body biasing methodologies was designed in a 0.18μ technology. The goal of this chip was to explore the effectiveness of adaptive body biasing on a modern technology, and to characterize intradie variations before and after body biasing is applied. Ultimately, the test chip was meant to demonstrate the reduction of active leakage currents that can be achieved in a large design by independently biasing each "local" island to tighten delay distributions throughout a chip. Unfortunately, the test chip was implemented in a single well technology, so only the PMOS bodies could be independently biased. To quantify the effects of body biasing on NMOS bodies, the entire sub-

strate would have to reverse biased to emulate the effect of a shift in the NMOS threshold voltages.

Figure 5-9 shows a diagram illustrating the test chip setup. To quantify the intra-die variations throughout a large multi project test chip, several different adaptive body biasing generator clusters were replicated throughout the die. Each cluster was then comprised of 12 individual body bias generators. With this setup, one can explore how intra die variations will give rise to large variations between clusters, but that local ABB generators within each cluster would track more closely.

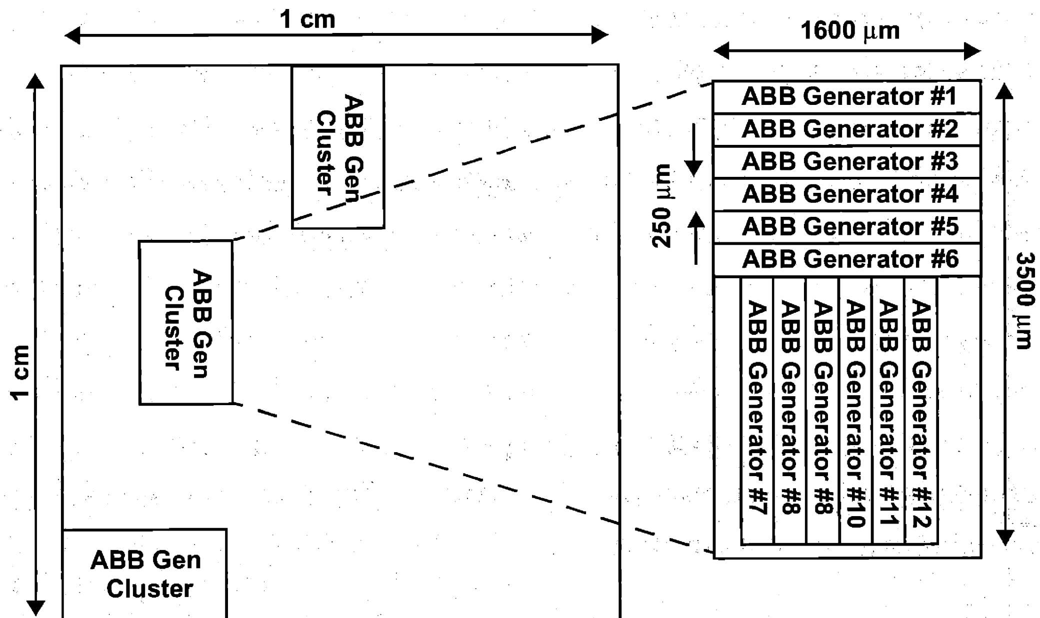


FIGURE 5-9. Adaptive body biasing test chip (multi-project chip) consisting of distinct ABB clusters comprised of multiple ABB generators.

5.7.1 Adaptive body biasing generator implementation

Figure 5-10 below shows a block diagram for the adaptive body bias generator, which is used to comprise the cluster regions for the test chip. The ABB generators are based on

the matched delay line approach illustrated in Figure 5-3, but the implementation is done with a digital control loop.

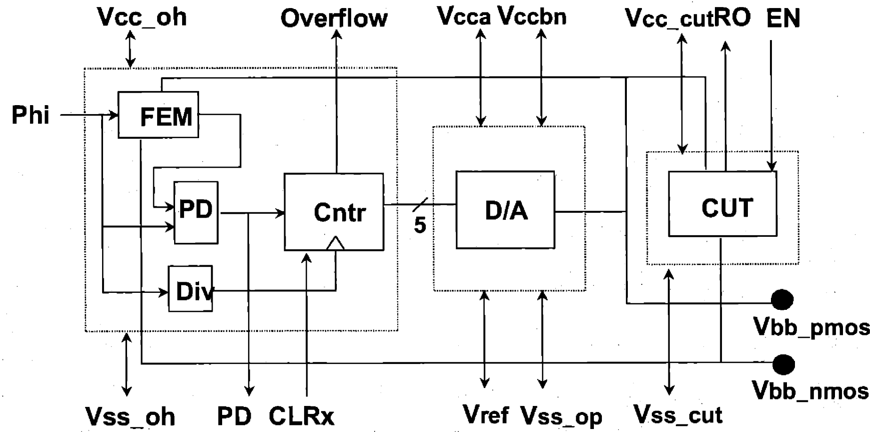


FIGURE 5-10. Test chip architecture for adaptive body biasing generator

The circuit under test (CUT) for this test chip consists of several parallel chains of complex inverters that model the critical path of a microprocessor, known as a FEM chain. One of these chains is used as the extracted critical path that serves as a matched delay line in the feedback loop.

The adaptive body biasing generator should ideally operate as follows. A target clock frequency is input to the system and fed to the matched delay line. The output of the matched delay line is then compared with the input clock period using a phase detector to determine whether the matched delay line is faster, slower, or equal to the target clock period. The counter is then incremented or decremented (clocked with a divided version of the input clock), which steps a simple D/A that biases the PMOS bodies and the NMOS bodies respectively. These body bias settings then complete the feedback path by tuning the threshold voltages of the devices in the circuit under test and matched delay lines, which causes the matched delay line to become faster or slower as necessary.

This adaptive body biasing mechanism is a digital control loop that operates at a speed significantly slower than the incoming target frequency (The counter and D/A

update period corresponds to a divided version of the input clock). The matched delay line deviation from the target frequency is a result of either fixed parameter mismatches that do not vary with time or very slowly changing parameters such as temperature, hot carrier degradation, or electromigration effects. As a result, the bandwidth requirement for the adaptive body biasing feedback loop can be very low, and simple circuit techniques can be used to implement the feedback loop.

Unlike traditional PLLs and DLLs where tracking dynamics and acquisition time are important, the adaptive body biasing controller does not need to be very fast. Instead, the controller merely needs to sense the appropriate delay mismatch that arises from static and pseudo-static variations, and compensate the device threshold voltages until the critical path delay is nominally the same as the target frequency. The adaptive body biasing controller is not designed to compensate for high frequency noise or jitter between the output of the matched delay line and the input clock. In fact, such an accurate tolerance would serve no real purpose because the matched delay line is only meant to approximate the critical path of the circuit block, and as such, cannot be used to compensate for high frequency delay variations that actually effect the critical path. Subsequently, the matched delay line must be padded with extra margins to ensure that the circuit under test functions properly and accounts for mismatches between the critical path modeling, clock skew, jitter, and worst case power supply fluctuations. The adaptive body biasing controller merely compensates the circuit under test so that the nominal critical path delay is closer to the desired target.

Stability in the digital control loop is straightfoward as well. As long as the update period is slow enough so that circuit transients will completely settle before the next counter transition, the digital control loop should function properly. Furthermore, the digital controller can easily be designed to ensure that the digital feedback loop yields a damped response. Once the appropriate bias values for the PMOS and NMOS in the local island are determined, it is not necessary to continue activating the adaptive body biasing loop. In fact, to save power one can simply save the appropriate bias "words" in memory, turn off the feedback circuitry and matched delay line, and then directly drive the D/A converters in an open loop fashion. To account for slowly varying parameter variations

such as temperature and mobility degradation, the adaptive body biasing loop can be periodically refreshed. One of the advantages of using a digital approach to implementing the adaptive body biasing controller is that the it is easy to turn off the feedback loop and drive the D/A converters from a stored bias value, which can significantly reduce power overhead.

5.7.2 Test chip simplifications

For the specific test chip that was designed to evaluate the adaptive body biasing mechanism, some simplifications were made to facilitate testing and design. First, the technology available for the test chip was a standard single well CMOS process, so only PMOS bodies could be independently biased. As a result, only an onchip PMOS body bias generator was implemented, and to model the effects of NMOS threshold voltage tuning, the entire substrate would need to be biased from off chip. Because the testchip was intended to merely test the adaptive body biasing principle, and not to represent a functioning adaptive body biasing system, directly biasing both body terminals manually off chip could still provide useful data. Furthermore, to simplify the ABB circuitry, a straightforward, but non optimal, implementation was chosen since the primary goal of the test chip was to test variation issues and body biasing effectiveness rather than to provide an efficient or low power circuit solution for adaptive body biasing.

5.7.3 Phase detector implementation

A simple binary phase detector consisting of two series connected static flip flops was used in the adaptive body biasing test chip. This phase detector was chosen because functionally it is very simple to design, yet gives a reasonably accurate measure of local matching between the delay line and target frequency. The flip flop phase detector uses

the input clock signal to sample the output of the matched delay line. If it samples a 1 then the matched delay line is too fast, and if it samples a 0 the matched delay line is too slow.

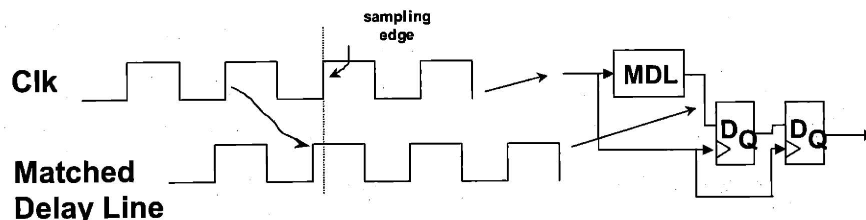


FIGURE 5-11. Flip flop phase detector operation

The phase detector output thus is a basic binary choice (fast or slow), whereas more complex phase detectors can output a voltage signal that is proportional to the phase difference. However, in the digital control scheme shown in Figure 5-10, the control loop does not adjust the body bias value based on the magnitude of the phase difference. Instead, it incrementally increases or decreases the device body biases in a binary fashion, so a complicated phase detector is not necessary. This is allowable because the control loop bandwidth can be very slow. However, because of the simplicity of this phase detector, the input frequency must be close enough to the matched delay line frequency to ensure that the phase detector does not accidentally lock at a multiple of the target frequency.

By using a phase detector like that shown in Figure 5-11, there are 2 basic mechanisms that can be used in the adaptive body biasing feedback generator. First, it is possible to let the feedback loop transition in only one direction so that as long as the matched delay line is too fast, the bias generator will increment the D/A to increase the amount of reverse body biasing. As soon as the phase detector crosses over from a 0 to a 1, then the matched delay line will nominally match the input target frequency and the feedback loop can be stalled. By periodically resetting the controller (so that zero body bias, or maximum forward body bias is applied) and reactivating the feedback loop, this controller can take into account slowly varying parameter variations. Because of the simplicity of this

flip flop phase detector and operation, this approach was used in the test chip implementation.

Another possible control mechanism is to allow the counter to both increment or decrement the D/A converter. Due to the limitations of the simple phase detector, there is no way to signal a perfect match. Instead, when the target frequency is close to the matched delay line frequency, the phase detector will oscillate between 1 and 0. This oscillation occurs because the feedback loop increases reverse body biasing until the matched delay line is too slow, in which case it decreases the reverse body biasing until the matched delay line is too fast, and back and forth. Thus, simply allowing the feedback loop to continue oscillating will provide the appropriate body biases necessary to nominally equate the target frequency with the matched delay line frequency.

5.7.4 Flip flop based phase detector performance

The flip flop phase detector's natural operation is to drive the circuit into a metastable state since the input edge is shifted closer and closer to the clock edge until the setup time of the flip flop no longer holds and the flip flop samples a 0 instead of a 1. By cascading the flip flop with another edge triggered flip flop however, the probability of entering a metastable state that cannot be resolved within the sample period becomes exceedingly low. While in a strict sense it is a poor circuit implementation that results in metastable outputs, the probability of metastability is sufficiently low for a cascaded flip flop implementation that the phase detector is quite acceptable for a test chip.

The resolution of the phase detector, or the minimum phase difference discernible between the phase of input clock and the output of the matched delay line is limited by the $T_{setup} + T_{hold}$ of the flip flop. This is necessary because in order for the flip flop to correctly sample a value at the input, it must be valid a time T_{setup} before the clock edge and continue to be held a time T_{hold} after the clock edge. Thus, if the input clock transition occurs during the uncertainty period of the flip flop, the output can theoretically go to any value. The difference between sensing a valid 1 or a valid 0 in the phase detector is on a resolution scale comparable to $T_{setup} + T_{hold}$. However, the critical path used in the test chip contains several cascaded complex gates, so a mismatch in resolution of less than 1

flip flop delay is small and still yields a very accurate measure of the critical path delay. Thus in this implementation, the binary phase detectors give 1-shot outputs based on the phase relationship between the clock and matched delay line output at fixed sample periods. For more reliable operation, the phase detector can be modified to average the phase detector readings over many cycles before indicating an up or down bias adjustment at the feedback loop sample frequency. A simple way to accomplish this is to digitally filter the output of the phase detector and compute a moving average.

5.7.5 Alternative phase detector implementations

Alternative phase detector architectures can provide improved performance over the simple flip flop based approach described earlier. For example, a phase detector that can output up, down, and wait signals to the counter is shown below.

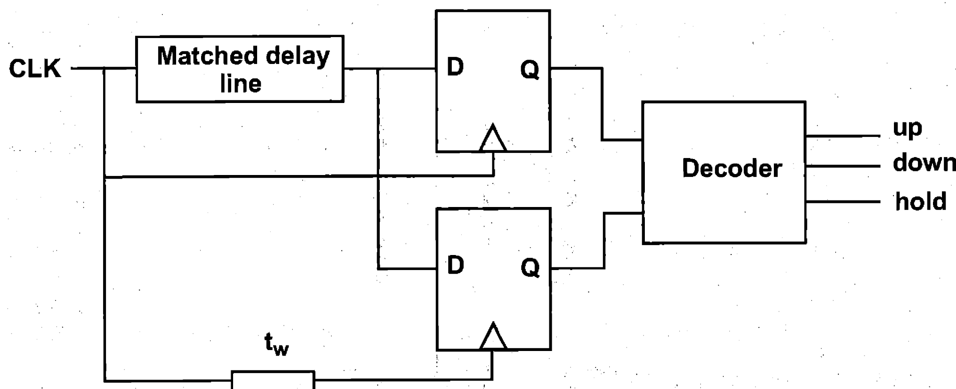


FIGURE 5-12. Alternative phase detector with up, down, and hold outputs

This phase detector can sense a dead zone period, set by the t_W delay tap, which defines a short window about the clock sampling edge where the matched delay line can be assumed to be nominally matching the input target frequency. As a result, it can be useful in adaptive body biasing control loops that can shift bias voltages in both directions, and will also prevent the feedback loop from oscillating about a target delay. Although in general having a deadzone period is not beneficial for precision analog DLL and PLL applications, in a digital approach this can simplify implementation and controller operation. The reduced accuracy of the phase detector due to the deadzone is acceptable in a digital

approach because the matching between the critical path replica and the target operating frequency is less stringent than in traditional analog applications such as frequency synthesizers or PLLs for clocking. With this phase detector capability, the adaptive body biasing loop can be left on (rather than completing one lock cycle and stopping the controller), which allows the circuit body bias settings to be continuously updated in both directions. However, to ensure proper functionality, it is important to have a high enough step resolution such that the incremental change in body bias voltage will result in a small enough change in the critical path delay such that the output transition will be able to fall into the "dead zone" of the phase detector. For example, if the "dead zone" is too small, then the control loop will constantly hunt about the optimum point if the sample edge fails to align within the dead zone period. However, since the gain from a change in body bias voltage to a change in critical path is small, this should be an easy problem to avoid. Increasing the D/A step resolution can also improve the accuracy of the adaptive body biasing loop if necessary. Unfortunately, this phase detector implementation still suffers from potential metastable behavior as signal edges become aligned with clock edges. One way to ameliorate this problem is for the finite state machine to sample the results from the flip flop only after a short period after the clock edge (for example inserting another register), which exponentially reduces the probability of propagating a metastable state.

For more accurate phase detector implementations that do not use flip flop sampling structures, one can utilize traditional phase detectors borrowed from PLL and DLL circuits designs as well. A well known phase frequency detector that generates UP and DOWN pulses whose difference in average values are proportional to the phase difference is shown below[57].

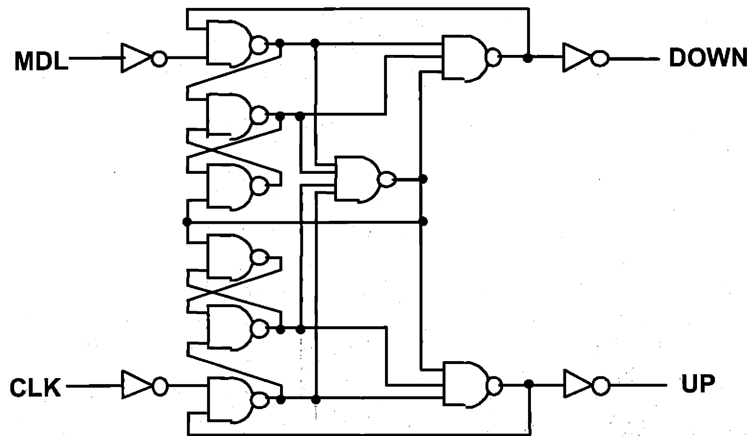


FIGURE 5-13. Conventional phase detector.

There are several ways that this phase detector can be integrated into a digital control loop to control the body biasing value. For example, one simple method could simply be to convert the UP and DOWN pulses into a binary signal by filtering and then sampling with a comparator as shown below.

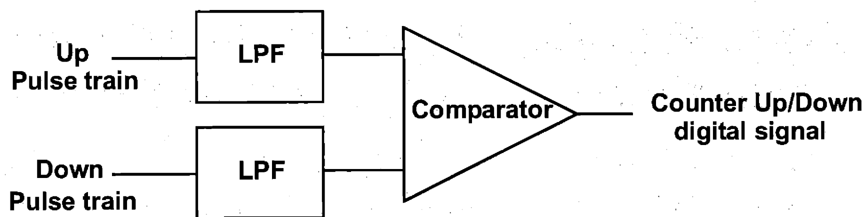


FIGURE 5-14. Converting a conventional phase detector to digital outputs levels.

The resultant digital signal can then easily be used in a digital control loop like in the test chip to drive an up/down counter that steps a D/A converter. One of the benefits of this phase detector implementation is that it avoids the metastability problems of the earlier implementation, and it also relies on an average phase detector reading over many clock cycles. However, the drawback is that this type of phase detector is better suited for

an analog feedback loop, while the previous flip flop phase detectors fit very nicely with the digital control loop scheme.

5.7.6 D/A Converter for PMOS Body Bias

To generate the proper bias values for the PMOS bodies, a simple D/A converter was used. The D/A input is driven by a standard counter that is enabled when the output of the phase detector is high (matched delay line is too slow), and holds when the phase detector goes low (nominally matched condition). The D/A converter does not need to have very high resolution, linearity, or accuracy because the output voltage is controlled by a feedback loop so any nonlinearities will be compensated out. In fact, as long as the D/A converter exhibits monotonic behavior, the adaptive body biasing generator should be able to lock on to the appropriate bias value.

The resolution of the D/A does not need to be very fine because the transfer function between body bias to critical path delay has relatively low gain. For example, a simple equation for the threshold voltage equation can be written as

$$V_t = V_{t0} + \gamma\left(\sqrt{2|\phi_p| + |V_{sb}|} - \sqrt{2|\phi_p|}\right) \qquad \text{(EQ 5-11)}$$

where differentiating with respect to body bias voltage gives

$$\frac{dV_t}{dV_{sb}} = \frac{\gamma}{2\sqrt{2|\phi_p| + |V_{sb}|}} \qquad \text{(EQ 5-12)}$$

which can be on the order of 0.2 for reasonable device parameters. As before, the delay of a critical path can be modeled as

$$T_{cp} = nK\frac{CV_{CC}}{(V_{CC} - V_t)^{\alpha}} \qquad \text{(EQ 5-13)}$$

and differentiating with respect to V_t yields

$$\frac{dT_{cp}}{dV_t} = nK\frac{\alpha C V_{CC}}{(V_{CC} - V_t)^{\alpha + 1}} \tag{EQ 5-14}$$

The gain factor from a change in the substrate bias value to a change in total critical path delay is then simply

$$\frac{dT_{cp}}{dV_{sb}} = \frac{dV_t}{dV_{sb}}\frac{dT_{cp}}{dV_t} \tag{EQ 5-15}$$

The D/A converter should have a resolution where the body bias step is small enough so that the change in critical path delay is on the order of the phase detector resolution. This ensures that the controller D/A tuning resolution is matched to that of the sensing ability of the phase detector. If the tuning resolution is much coarser than the phase detector resolution, then there would be sub-optimal matching between the critical path delay and the target operating frequency. On the other hand, if the tuning resolution is much finer than the phase detector resolution, the controller would be more complicated than necessary since the extra tuning resolution would not be resolved by the phase detector.

For the flip flop based phase detector shown in Figure 5-11, the resolution was on the order of the setup plus hold told time of a flip flop. In terms of the total critical path delay, this would correspond to a fraction of a single gate delay. Thus, the D/A resolution should provide a step size on the order of

$$\Delta T_{cp} < fK\frac{C V_{CC}}{(V_{CC} - V_t)^{\alpha}} \tag{EQ 5-16}$$

where f indicates some fraction (less than unity) of a gate delay. Substituting from Eq 5-15 yields

$$\Delta V_{sb} < \frac{fK\dfrac{C V_{CC}}{(V_{CC} - V_t)^{\alpha}}}{\dfrac{\gamma}{2\sqrt{2|\phi_p| + |V_{sb}|}}nK\dfrac{\alpha C V_{CC}}{(V_{CC} - V_t)^{\alpha + 1}}} \tag{EQ 5-17}$$

which can be simplified to

$$\Delta V_{sb} < f\frac{V_{CC} - V_t}{\frac{\gamma}{2\sqrt{2|\phi_p| + |V_{sb}|}} n\alpha} \qquad \text{(EQ 5-18)}$$

For V_{CC} of 1.0V, V_t of 0.35V, n=20, α=1.5, and dV_t/dV_{sb} of approximately 0.2, this yields a body bias step resolution that should be on the order of a fraction of 100mV. Thus, assuming a phase detector resolution corresponding to one quarter of a standard gate delay, the necessary D/A converter need only provide steps on the order of 25mV to align the matched delay line with the target clock frequency using a crude phase detector. With these operating conditions, the matched delay line can be tuned to within +1.25% of the target operating period, which is acceptable for test purposes.

Because only a very low performance D/A converter is required, a simple 5 bit R-2R implementation like the one shown below was implemented.

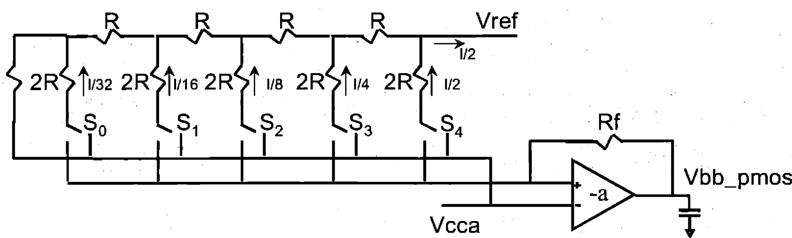


FIGURE 5-15. Simple R-2R D/A converter.

This D/A converter uses an R-2R ladder to generate the appropriate body bias values. An R-2R ladder structure was chosen because the D/A output will depend only on the relative ratio between resistors rather than on the absolute value of the resistors themselves. This was important because to get relatively high resistance values, N-well resistors were used in the test chip, but their absolute values can have very large fluctuations. However, the R-2R structure uses only one type of N-well resistor (giving the 2R), and a parallel combination (giving the R). As a result, the resistor ratios are closely matched

even though the actual resistance values might be difficult to control, and the D/A will still have relatively good performance. The R2R D/A provides an output voltage that can be selected based on the binary switch settings according to the equation below

$$V_{bbpmos} = \frac{R_f}{R}\frac{(0, 1, 2, \ldots 31)}{32}(V_{CCA} - V_{ref}) + V_{CCA} \qquad \text{(EQ 5-19)}$$

In effect, the D/A converter provides a reverse biasing range (provided V_{CCA} is set to the supply voltage V_{CC}) with a maximum range of V_{CCA}-V_{CC}, which is broken down into 32 equal steps depending on the binary switch weights. For example with a 500mV reverse biasing range, the D/A converter can provide 15.6mV steps, which is small enough to accurately tune the matched delay line to the resolution of the phase detector. This D/A converter architecture is also flexible because it is easy to provide forward biasing capabilities as well. For example, if V_{CCA} is biased to be less than V_{CC}, then the D/A converter range will straddle both the forward and reverse body bias regimes for the PMOS devices. However, the operation of the D/A and adaptive body biasing feedback loop will still operate the same as before.

The R-2R D/A utilizes switches S_0-S_4 to steer current either from the positive or negative amplifier terminals. In effect, the appropriate currents are then summed by the amplifier and converted to one of 32 different voltage levels. The current steering switches can be implemented as shown below in Figure 5-16.

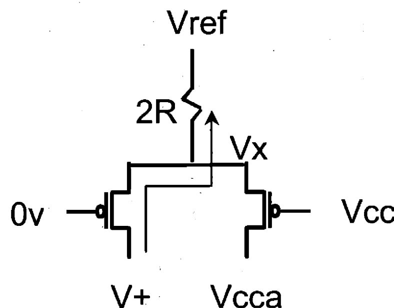


FIGURE 5-16. Steering switches used for R-2R D/A converter.

Since V_{CCA} is greater than V_{ref}, the steering switches can be constructed with PMOS devices. A logic "0" will turn on one of the switches strongly, while a logic "1" will turn off the other device strongly. This works because both V_{CCA} and V+ are smaller or equal to the supply voltage V_{CC} (even when forward biasing is employed), yet they are large enough to ensure that the on PMOS will have enough current drive to drive node V_x to V_{CCA}. This is because the PMOS "on" resistance is much smaller than the 2R resistor, and so the current steering operation functions properly.

5.7.7 NMOS Body Bias generator

Again because the testchip was implemented in a single well process, it was not possible to implement a complementary body bias generator for the NMOS body terminals. However, in a triple well process it would be possible to build a similar substrate bias generator that automatically adjusts NMOS body biases as well. To truly gauge the effectiveness of adaptive body biasing, both PMOS and NMOS body terminals must be biased accordingly. If only PMOS devices are reverse body biased for example, roughly only one half of the off devices in the local bias regions will have reduced leakage currents. Even if an extremely large amount of reverse body bias were applied to every PMOS device, the leakage power could at most be reduced by a factor of 1/2. This is because half the gates in a CMOS implementation would have an "on" PMOS path to V_{CC} with a very leaky path to ground through unbiased NMOS devices.

Thus even though the test chip can model an automatic adaptive body biasing mechanism using the integrated D/A for PMOS bodies, it will provide suboptimal results for leakage reduction. In order to fully evaluate the benefits of adaptive body biasing for this test chip, one must rely on external bias control of both PMOS and NMOS devices simultaneously. By biasing the entire substrate to tune NMOS devices, it will then be possible to emulate the effects of body biasing in a triple well process. The test chip was thus designed so that the phase detector output could also drive an offchip signal, which can be used in an external software feedback loop which sets the appropriate body bias settings externally through the testing equipment. In this way, both PMOS and NMOS device bodies could be appropriately biased from off chip sources.

5.7.8 Analog adaptive body biasing control alternatives

The adaptive body biasing feedback function is very similar to the operation of a delay lock loop. As a result, it is instructive to explore the feasibility of leveraging conventional analog DLL architectures to implement an analog adaptive body biasing feedback loop instead of the digital approach described in the previous sections. With an analog approach, the adaptive body biasing controller could be implemented in a smaller area, and possibly with lower active power compared to a digital control technique. For example, a digital approach requires the use of dividers, counters, and flip flops for peripheral logic, but an analog approach could be implemented in a much more straightforward fashion with fewer elements. One simple way to implement a bodybiasing feedback loop is to utilize a standard charge pump DLL approach as shown below

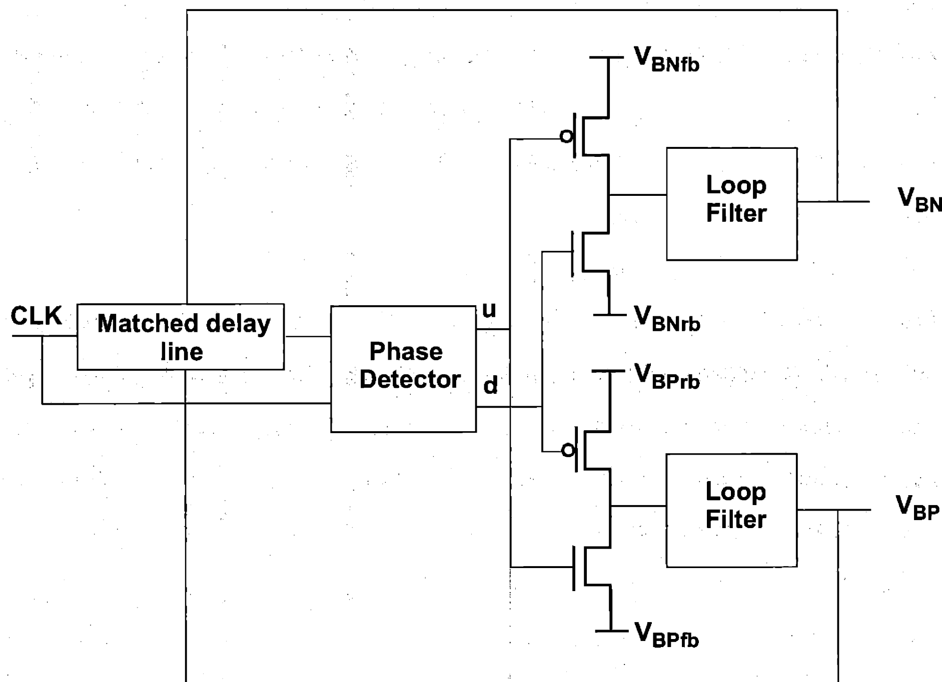


FIGURE 5-17. Analog DLL approach using charge pump to implement ABB controller.

This architecture uses a straightforward phase-frequency detector that outputs pulses whose average value are proportional to the phase differences. These pulses directly drive a standard charge pump (asymmetrical though) which automatically adjusts

the body bias until the matched delay line is set properly. A control loop can be made separately for the NMOS and also the PMOS devices. Compared to a digital approach, the analog adaptive body biasing controller is much more compact.

Other analog techniques can also be used to generate onchip voltages higher than V_{CC} (for PMOS reverse body biasing) and lower than GND (for NMOS reverse body biasing). Figure 5-18 shows a Dickson charge pump architecture that can be used to provide onchip reverse body bias voltages for PMOS and NMOS devices[58].

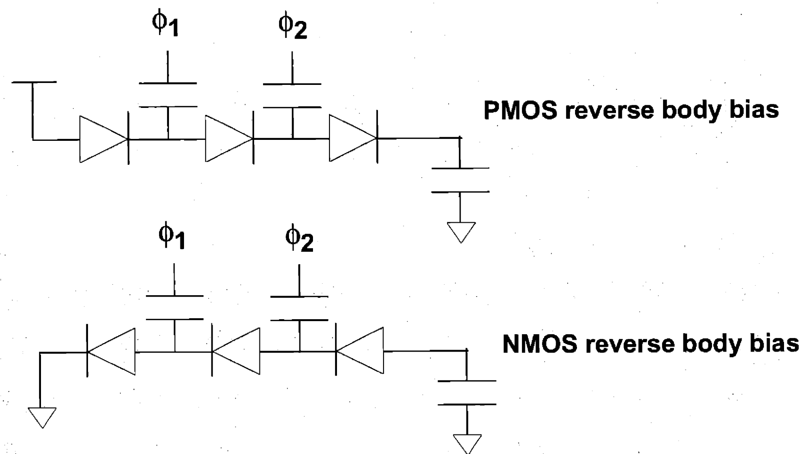


FIGURE 5-18. Dickson charge pump illustrating on chip generation of voltages higher than V_{CC} and lower than ground.

With a Dickson charge pump, the adaptive body bias generators do not need to be supplied with external bias voltages. Instead, the reverse body biasing voltages can be generated internally using a capacitive boosting mechanism. The Dickson charge pump operates by creating a series of diodes whose internal nodes are capacitively coupled to a clock signal. By using two out-of-phase pulse sequences, charge can be pumped up the diode chain to build up large voltages at the output node for the PMOS body case, and similarly can pump out charge to create negative voltages in the NMOS body case. The output range is limited by the number of diode stages chosen. By gating the pulse sequences depending on the state of a binary phase detector, one can integrate a Dickson charge pump into an adaptive body biasing controller scheme.

Although on the surface analog adaptive body biasing approaches seems elegant and efficient, there are some serious limitations that need to be overcome before they can be used to successfully implement adaptive body bias controllers. One major problem is that the PMOS and NMOS analog biasing schemes can drift apart in time, which will make it difficult to maintain a constant offset relationship between N-well and P-well reverse bias settings. For example, if independent charge pump are used to bias PMOS and NMOS body terminals, it may be possible to always pump charge in the same direction for both wells (both wells becoming more reverse biased or more forward biased), but it is not possible to ensure that the relationship between the two charge pump voltages is maintained. As described earlier, there are an infinite combination of PMOS and NMOS body bias values that can satisfy a delay criteria, since a fast PMOS can compensate for a slow NMOS, or vice versa. If the two loops operate in an independent fashion (no cross coupling effect), then the 2 dimensional control mechanism is underconstrained, and the amount of reverse bias applied to the PMOS and NMOS bodies will drift apart in time. For example, if either the PMOS or NMOS well bias leaks current more rapidly than the other, or if the charge pump in one loop is stronger than the other, then the skew between PMOS and NMOS bodies will continue to spread apart until one of the wells voltages is pinned at either the maximum or minimum bias setting. On the otherhand, in the digital control approach, the amount of reverse body bias applied to the PMOS and NMOS devices can be accurately controlled. Furthermore, a constant offset between the bias settings for the PMOS and NMOS devices can easily be maintained during the control period in order to compensate for any systematic mismatch between the two types of devices. Thus, in order for an analog scheme to function properly, a complex two dimensional control scheme must be use to not only maintain the proper speed of the matched delay line, but also to maintain the proper relationship between PMOS and NMOS delay characteristics.

Another problem with using an analog approach with imbedded charge pumps is that the output terminals may not be driven with a sufficiently low output impedance. In the digital scheme, stand-alone buffers were used to drive the PMOS and NMOS body terminals, and these could be designed to have very low output resistance with large driving capability. However, charge pumps that directly drive PMOS and NMOS body terminals

may not be able to hold these terminals strongly enough to filter out noise or to supply enough current to meet charge injection requirements. In the case for the standard charge pump DLL approach, it would be possible to use buffers to help drive the well bias voltages. On the otherhand, for the Dickson charge pump approach to generate on chip reverse bias voltages, buffers could not be used, and thus excessively large bypass capacitors might be required to hold the bias voltages strongly.

A final drawback to an analog approach is that overall power dissipation might actually turn out to be higher than for a digital control approach. With a digital approach, the feedback loop can be stalled in time, with the body bias values stored in memory and used to directly drive the D/A converter in an open loop fashion. As described before, the dynamic variations are so slow (temperature, hot carrier degradation, static parameter variations etc.) that the control loop simply needs to be refreshed only periodically in order to maintain the proper N-well and P-well bias values. However, with an analog charge pump approach, it is not possible to stall the controller loop because N-well and P-well bias voltages must constantly be refreshed because output charge will leak away and voltages will drift in time. Because the matched delay line, phase detector, and charge pump are constantly active in the analog case, the total power dissipation might turn out to be greater than the power dissipation of the digital controller that operates less frequently.

In summary, analog adaptive body bias control techniques, while initially attractive, have many limitations that need to be solved. As a result, one cannot direct apply analog DLL circuit techniques to an adaptive body biasing controller without making significant modifications that take into account the 2 dimensional nature of compensating both PMOS and NMOS body voltages. However, digital control techniques have been shown to be very efficient at providing adaptive body biasing capabilities. Although the area requirements are larger than the analog case, the ease of operation and implementation, the low output resistance stand-alone buffers, and the ability to turn off the control loop and drive the body biases open loop make digital control techniques very attractive.

5.8 Test Chip Simulations

Because of proprietary reasons, the adaptive body biasing testchip results could not be included at the time of this thesis (although as of this writing, the chip has been shown to be fully functional and data is currently being collected). Instead, simulations were performed to estimate the benefits of adaptive body biasing on a hypothetical sample of dies. To model the effects of parameter variations, it was assumed that a representative "local island" could be characterized with a frequency target having a gaussian probability distribution ~ $N(f_0, \sigma)$. A hypothetical die is then constructed out of "N" of these characteristic regions so that dies with more of these subregions can be thought of as simply being bigger. The impact of having these "N" regions varying independently and the benefits of utilizing adaptive body bias to tighten distributions and reduce active leakage currents is explored.

To estimate the random variations that a hypothetical "local island" critical path will exhibit, 16 different process skews that correspond to variations in the SPICE device models were simulated. These skews include models representing typical, fast, and slow corners, and a variety of parameter mismatches. As an estimate, these process corners were then simulated for the circuit under test, ordered from lowest frequency to highest frequency, and then mapped to a normal frequency distribution where the mean corresponds to the frequency of the TTTT corner (typical parameters), and the standard deviation corresponds to some fraction of the frequency offset of the LMAX corner (maximum device length variation). The normal distribution (a continuous random variable) is then modeled with a discrete probability mass distribution consisting of 448 discrete samples, each corresponding to the probability that a frequency lies within an appropriate frequency band. The 16 different process skews were thus simulated to derive exact frequency quantities and serve as anchor points from which the rest of the frequency samples are linearly extrapolated. In effect, the underlining assumption is that the actual mechanism for random variations of the circuit delay can be modeled by the random variables derived from these 16 process skews. Clearly this is gross simplification and as such cannot be relied on to give accurate results. However, it is useful as a first order approxima-

tion to explore the benefits of adaptive body biasing on active subthreshold leakage reduction.

The sample space for this random experiment (i.e. the performance of a typical local island critical path) corresponds to these 448 discrete process skews, some of which correspond to simulated process corners while others are derived from interpolation. The process skew interpolations and probability distributions are chosen such that the frequency random variable, which can be thought of as simply a function of the different process skews, will have an appropriate gaussian distribution. In effect, process skew random distributions are estimated going backwards by trying to match the frequency mapping from these random skews to have a desired "normal" distribution.

However, by focusing on the process skews themselves as the underlining random process, then it becomes easier to model distributions for other related variables that are functions of these random outcomes. For example, for each random outcome (TTTT, SSSS, FFFF etc.) one can map a corresponding leakage quantity that can be derived from simulations of the circuit under test. Likewise it is possible to perform an adaptive body biasing procedure on the circuit for each process skew and to estimate the new frequency distribution or compensated leakage distributions. Because the probability mass functions of each skew is known, the simulation results for these new random variables (which are simply functions of the random process skew) can also be calculated.

Figure 5-19 below illustrates the basic procedure to derive probability distributions for different random variables such as frequency, leakage currents, adaptive body biasing frequency, etc.

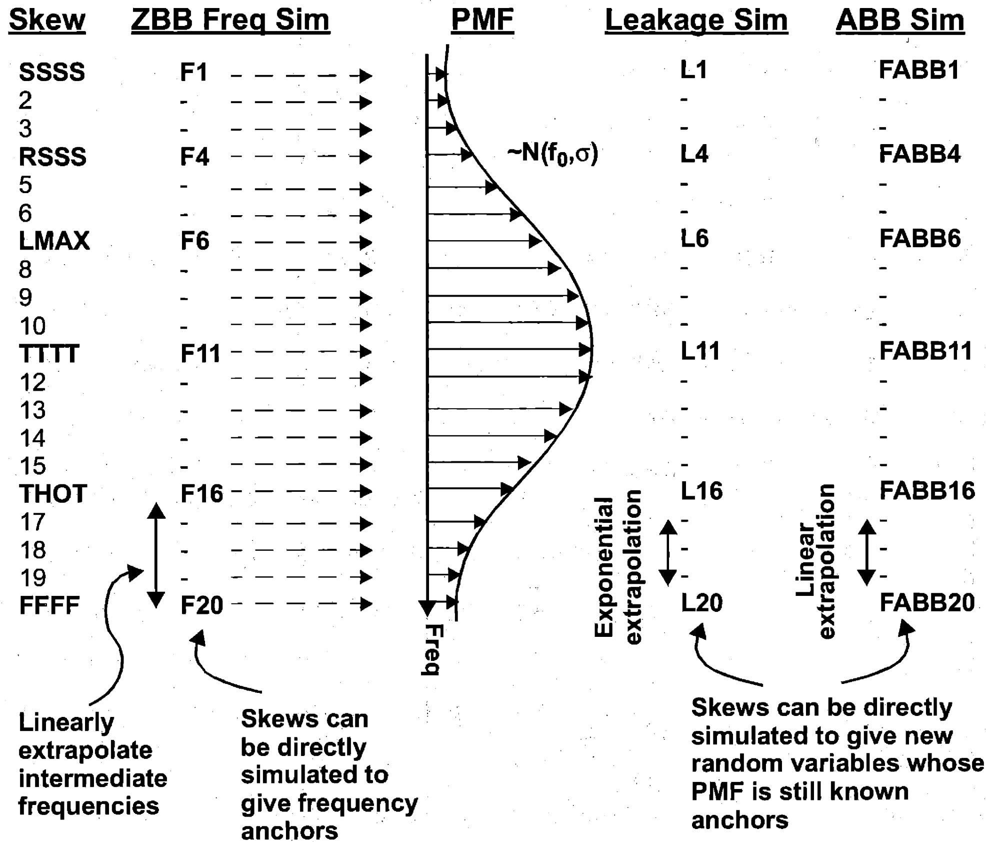


FIGURE 5-19. Example of how process skews can serve as anchor points to estimate probability distributions.

The first column shows the set of process skews that define the simulation "anchor" points. The second column shows the zero body bias frequencies that can be simulated from each process corner, with extra samples inserted that are linearly interpolated between the anchor points. These samples are chosen so that the ZBB frequency random variables have a complete sample space that spans the skew frequencies. For the purposes of the testchip simulations, 494 total sample points were used to provide the desired frequency resolution. The next column then shows a graphical representation of the probability mass functions, which have the distribution of a sampled gaussian distribution. This

distribution is based on a normal probability distribution, except that it shows discrete probability mass values that correspond to the probability of falling within a small frequency band centered about each sample point. By varying the standard deviation of this gaussian, one can explore how the statistics of the process skews can effect overall performance and leakage currents. The final two columns show two examples of possible random variables that can be derived from the process skews. Leakage currents can be computed, and the effects of using adaptive body biasing on frequency distributions can also be studied. Again, the samples are derived from direct simulations, and the interpolated skews are derived by interpolation between the sample simulation results. For the ABB frequency column, a linear interpolation scheme was used, but for the leakage column, an exponential interpolation scheme was used. This better models the exponential characteristics of the subthreshold leakage currents.

This process of simulating statistical samples to derive the output statistics of the system is related to the Monte Carlo simulation technique. However, the difference is that in this case, the assumption (though inaccurate) is that statistics of the random experiment are completely characterized by the 448 skew samples, which can then be exhaustively simulated to explore adaptive body bias impact on active leakage currents. Although inaccurate, this assumption can still provide first order models for how adaptive body biasing can significantly reduce active leakage currents and tighten distributions.

5.8.1 Limitations of the Modified Monte Carlo Simulations

This modified Monte Carlo framework for modeling the intra die variations with process corner simulations allows one to begin exploring how adaptive body biasing can affect frequencies and impact leakage currents. However, the assumption that the simulation process corners accurately reflect real random mechanisms for intra die variations is not very accurate. First of all, the process corner models are only rough estimates of process variations and most likely reflect inter-die variation mechanisms. Yet in the previous analysis, the process skews are used to estimate the intra-die random variations to help quantify the benefits of adaptive body biasing on a large die divided into "N" local islands. Second, the number of process corner models is very limited, and thus represent a very small sampling of possible random variations outcomes for the circuit under test. As a

result, the process skews cannot accurately span the truly complex scope of random variations that a chip can exhibit. Finally, the process skews in the above model are assumed to reflect actual samples of an experimental outcome of a circuit implementation. As such, the process skews should ideally represent the cumulative effect of all types of parameter variations on the overall sample. However, the process skews typically characterize the variation of one type of parameter (threshold voltage, gate length, mobility deviations) while holding others constant, and thus does not accurately model the cumulative variation a typical sample might exhibit.

Despite the obvious limitations of modeling intra-die variations using process corner simulations, this technique can still provide valuable first order estimates, especially in characterizing potential benefits of adaptive body biasing. The set of simulation skews can serve as a starting point for characterizing some potential variations that a chip might experience. One can also map these skews onto a variety of different distributions (for example, try different normal distributions by varying the standard deviation) to determine trends and impact directions of adaptive body biasing. The benefit to this approach however is that it can use actual circuit simulations to characterize the impact of adaptive body biasing on active leakage currents. By simulating body biasing impact on a skew by skew basis, the effectiveness of tuning threshold voltages with body bias for a variety of circuit conditions can be explored.

Ideally, the best way to characterize the impact of adaptive body biasing on a large system is to experimentally collect data from a large number of physical samples. The different variation mechanisms in an actual circuit are not fully understood and the benefit of adaptive body biasing on active leakage currents can be directly quantified through measurements. However, simulation approaches like the modified Monte Carlo methodology described earlier can at least give first order understanding on the impact of adaptive body biasing on a large system. Furthermore, it is easier to extend these simulations to explore how the impact of ABB varies with different chip designs, sizes or variation distributions. Although outside the scope of this research, more advanced techniques can be further explored to use flexible simulation techniques to better quantify the effects of parameter

variations on circuit performance and the improvements achievable through adaptive body biasing.

For example, one simple approach is to develop improved SPICE sample skews that will reflect realistic circuit variations that might occur in an actual circuit implementation. By simulating the circuit under test using a large number of these variation models (which model cumulative effects), it is possible to use a brute force technique to characterize the impact of parameter variations on a large circuit. However, more efficient techniques borrowed from statistical metrology research can also be explored. In this case, fundamental parameter variations can be individually modeled using systematic or random techniques, and overall circuit performance statistics can be using through Monte Carlo simulations or other analytical techniques[52][59]. Statistical metrology techniques can thus be very useful for quantifying more efficiently and accurately the benefits of adaptive body biasing for a variety of circuits and systems.

Even though simulations on the impact of parameter variations and adaptive body biasing on circuit performance can be significantly improved, the previous simulations still show that adaptive body biasing is generally beneficial at reducing active leakage currents. The purpose of this research was not to develop sophisticated simulation techniques to evaluate the impact of parameter variations on performance, but instead to propose an adaptive body biasing to help reduce active leakage currents. From theory and from the above analysis, it is clear that if intra die variations are significant, then application of adaptive body biasing can help tighten critical path distributions, which will always be beneficial. Reverse body biasing will definitely help reduce active leakage currents, and if used, forward body biasing, can improve yields. Technology scaling trends also show that intra-die variations will continue to increase, necessitating the need for a circuit based approach for compensating parameter variations. Furthermore, as described earlier systematic variations, including different types of critical paths in a system-on-a-chip, will also yield circuits that have different operating limits in different parts of the a chip. For all these reasons, adaptive body biasing appears to be a promising technique that could become more important with future technologies.

5.8.2 Simulation results

In this section, simulation results are shown that illustrate how frequency distributions can be tightened using adaptive body biasing with a maximum range 500mV in both the forward and reverse bias directions. The reverse bias range was limited simply because of device reliability concerns, so theoretically more range is possible if the circuits are designed carefully. The simulation was performed assuming that the frequency variations of a "local block" could be characterized by a gaussian distribution, with mean normalized to 1 (corresponding to the frequency at the TTTT corner), and with two different standard deviations normalized to 0.121 and 0.0907. These correspond to sigmas of 12.1% and 9.07% variations in frequency, which are reasonable amounts of variation that one would expect in future technologies. Simulations were also performed assuming N= 1, 10 and 100 blocks in the system. This corresponds to progressively larger systems where the intra die variations of the local blocks increasingly effect the total chip yields. For example, with increasing size, the target frequency has to be lower to ensure that all blocks within the chip operate correctly. A target yield of 90% was assumed, which means that the target operating point of the system must be chosen such that there is a 90% chance that frequency of all the local block exceeds this target. This 90% target frequency actually was chosen assuming that forward biasing could be utilized. As a result, slow nominal devices could actually be speeded up, so the 90% target frequency can be pushed higher than the case for the nominal zero body bias.

N=1 Case

In the first example, the system is modeled as being comprised of a single block. The frequency probability distributions before and after adaptive body biasing is applied are shown below for the case of sigma=0.121 [normalized Hz]. With this frequency distribution, the target frequency for the chip to achieve a 90% yield corresponds to a normalized target frequency of 0.967 [normalized Hz].

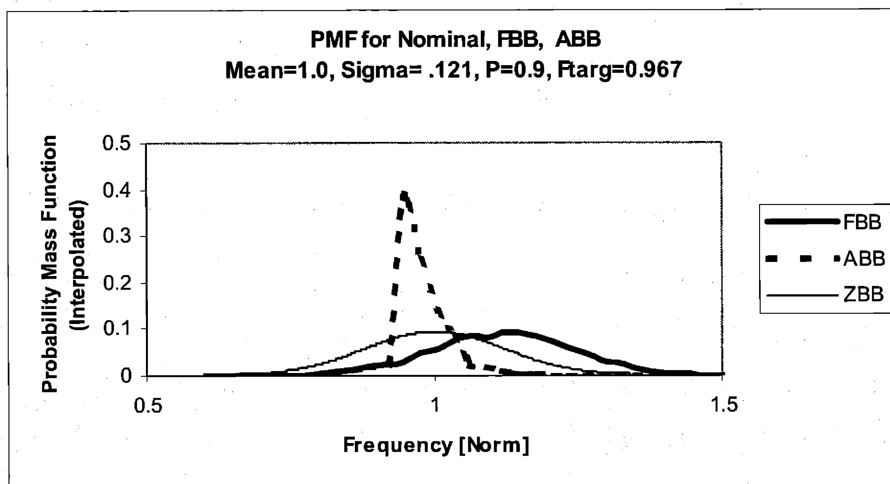


FIGURE 5-20. PMF for system consisting of N=1 blocks, standard deviation .121, targ frequency 0.967

The curves in Figure 5-20 correspond to probability mass functions for the 448 different random skews, but they are drawn as a continuous curve for clarity. The middle curve illustrates the nominal distribution for the process with zero body bias (ZBB). Also shown is the probability distribution of the critical path delay variation if maximum forward body bias (FBB) is applied to all circuits. As one can see, the effect of applying forward body bias to the chip is to shift the frequencies higher. In effect, the ability to utilize forward body biasing increases the actual yield of the chip because chips that were nominally too slow could actually be sped up. Most interesting however is the curve showing the probability mass function of the chip frequencies after adaptive body biasing (ABB) is applied. As can be seen, sample points that are slower than the target frequency are sped up with forward body bias, while the sample points that are faster than the target frequency are slowed down with reverse body bias. The ABB distribution gradually tapers off on both ends as well because this reflects maximum reverse and forward bias settings for those samples which are simply too far away from the mean to be fully compensated.

One of the difficulties with the above probability mass functions is the fact that the original random skew samples are processed to give rise to the arbitrary ABB frequency

samples. Because of this, individual sample points are mapped onto a new frequency random variable, whose discrete sample space corresponds to the 448 different frequencies values, some of which can be very close together. However, the PMF for this type of random variable is not intuitive because the sample space is staggered (with tighter spacing close to the target frequency) and not uniform. On the otherhand, the PMFs in the above figure actually show a new type of random variable that is derived by taking the ABB frequency or FBB frequency results, and "re-bining" them into equally spaced frequency ranges. In effect, probability mass values of different samples, which correspond to closely spaced frequencies (like when ABB is applied), are lumped into a single frequency sample whose probability corresponds to the sum of all the previous samples. This procedure is necessary so that the ZBB, FBB, and ABB frequency probability distributions can all be plotted on the same axis, and the scales can be compared in a meaningful way. Unfortunately, depending on the size of the "bin" used to collect the different frequency samples, the appearance of the PDF can vary. Also as described earlier, the curves are drawn as a continuous waveform for ease of viewing, when in fact the underlining mechanism is a discrete probability mass distribution. The rebinning and interpolating procedure are thus only performed to qualitatively illustrate the effects of adaptive body biasing on circuit critical path statistics.

One way to avoid the difficulties associated with the PMF rebinning and interpolating schemes is to simply use the cumulative distribution function instead. The CDF for a random variable F at value f_0 shows the cumulative probability that the random variable will evaluate to less than f_0. As a result, this distribution is more accurate at illustrating the shifts in probability distributions after ABB is applied. Even though multiple samples can be shifted towards the target frequency when adaptive body biasing is applied, the cumulative distribution functions for ABB, ZBB, and FBB random frequencies can easily be

compared directly without the need for a rebinning equalization step. Figure 5-21 below shows the CDF for the previous system.

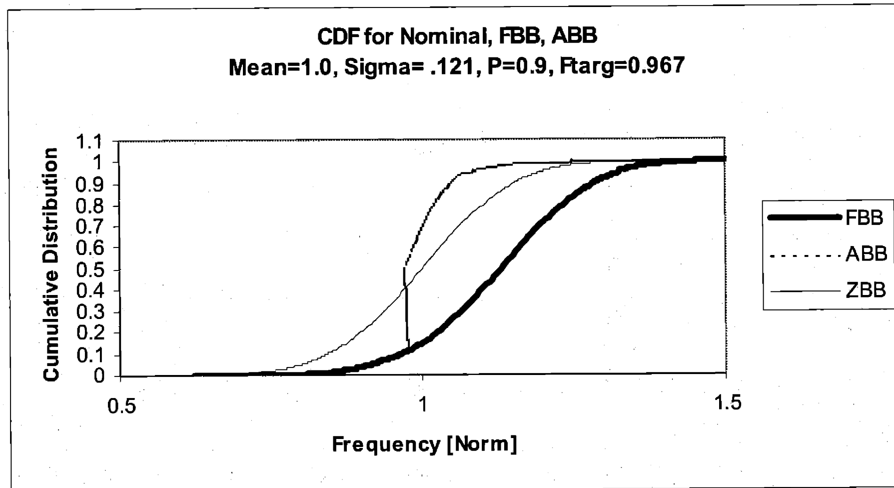


FIGURE 5-21. CDF for system consisting of N=1 blocks, standard deviation .121, targ frequency 0.967

As can be seen above, the CDF for the ABB frequency random variable is much sharper than the corresponding curves for the ZBB or FBB case. This is beneficial because it means that the distribution is much tighter so that the probability that the critical path frequency lies within a tight band is very high.

The tail of the ABB frequency distribution in both the PMF and CDF curves can be seen to coincide with that of the FBB frequency distribution at low frequency values. This is true because the ABB scheme used for this simulation would attempt to provide maximum forward body bias to those samples which are too slow. The point at which the ABB and FBB curves begin to deviate corresponds to the frequency point were maximum forward bias is on the verge of meeting the target frequency. This point also corresponds to the target frequency for the ABB generator. This is because for frequencies below this value, maximum forward bias is not large enough to speed up slow circuits to meet the target specification. At frequencies higher than this value, less than maximum forward bias (possibly reverse bias even) is sufficient to meet the target specification.

Another way to understand the operation of the dual direction adaptive body biasing mechanism is to imagine starting from a circuit condition where all devices are driven with maximum forward body bias to maximize performance and yield statistics. Next, the adaptive body biasing controller is activated, which then selectively adjusts the body bias voltages in the reverse bias direction to slow down any fast samples. With this interpretation, it is straightforward to see how the adaptive body biasing frequency distribution should match the FBB distribution at low frequencies, but that faster samples are pushed down to operate closer to the target frequency.

Figure 5-22 and Figure 5-23 illustrate the critical path statistics when the local island is characterized by a smaller standard deviation of 0.0907 [normalized frequency] instead of the larger standard deviation of 0.121[normalized frequency] that was assumed in the previous simulations.

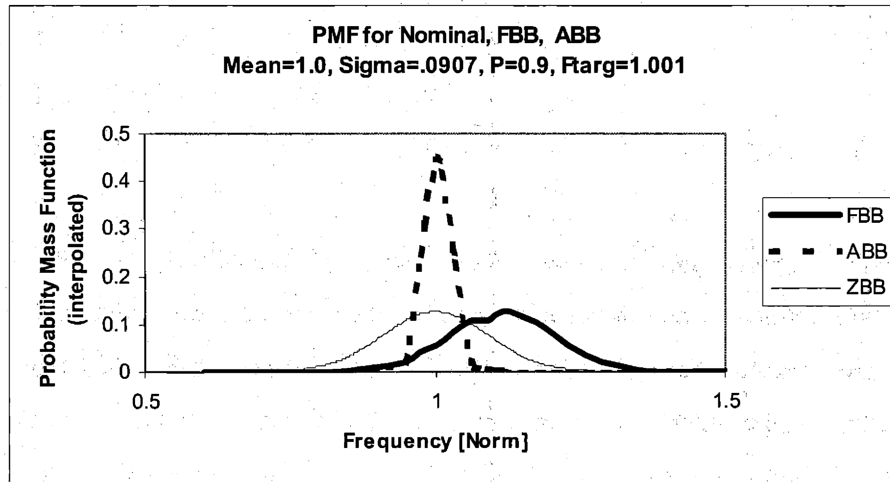


FIGURE 5-22. PMF for system consisting of N=1 blocks, standard deviation .0907, target frequency 1.0

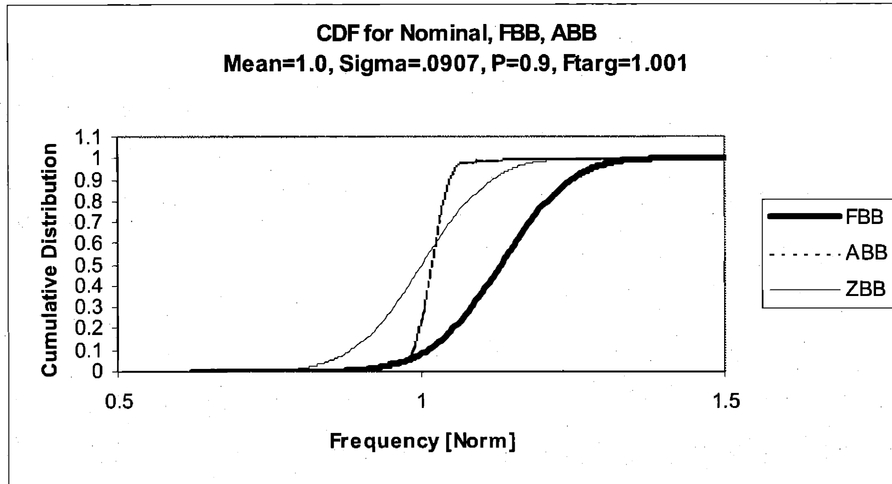


FIGURE 5-23. CDF for system consisting of N=1 blocks, standard deviation .0907, target frequency 1.0

With a smaller standard deviation, the target operating frequency is higher at 1.0 [normalized Hz]. Furthermore, the distribution is tighter because more of the samples are likely to fall within the controllable bias range.

By using simulations to determine the probability distributions for various random variables, it becomes possible to quantitatively characterize the benefits of adaptive body biasing. Although not shown in the above figures, probability distributions were made for the leakage currents of the test circuit as well. These were derived by simulating and interpolating between the "anchor" process skews described earlier. Finally the expected value of the leakage currents for the functioning chips were then calculated. Table 5-1

below summarizes the results for the simulations applied to the test chip modeled as having 1 local block.

TABLE 5-1. Simulation results for system with N=1 blocks, and different sigmas.

	P=0.9 Target Frequency (normalized)	% Change in target freq w/ forward bias	Expected Leakage Currents (normalized)	% Reduction in E(leakage) from FBB condition (for same sigma)
ZBB (σ=.121)	0.844	0%	0.4282	-57.2%
FBB (σ=.121)	0.967	14.6%	1	0%
ABB (σ=.121)	0.967	14.6%	0.4057	-59.4%
ZBB (σ=.091)	0.882	0%	0.40623	-60.9%
FBB (σ=.091)	1.001	13.5%	1.0393	0%
ABB (σ=.091)	1.001	13.5%	0.5105	-50.9%

The second column show normalized target clock frequencies which correspond to a 90% probability that the critical path speed will exceed this frequency. The frequency values are normalized to the nominal critical path frequency (corresponding to the typical skew). As can be seen, the frequency target for the zero body bias case is lower than that for the forward and adaptive body bias cases. This is because FBB and ABB provide a mechanism for speeding up slower devices with forward body biasing, which in turn shifts the frequency distribution to a higher value. The fourth column shows the expected leakage currents associated with each distribution. However, they are computed only over those samples that satisfy the yield requirement, so they truly reflect the average leakage currents one would expect in a functional batch of chips. These leakage values are normalized to the forward body bias expected leakage currents for the sigma=0.121 case as a reference. The final column shows the percent reduction in expected leakage currents for the ZBB and ABB cases compared to the FBB case. The FBB case is used as the nominal comparison scenario because this distribution corresponds to a higher target frequency than the ZBB case. This higher target frequency was the speed chosen for the adaptive body biasing controller as well, so it is more meaningful to compare ABB leakage currents with the FBB leakage currents in order to characterize the benefits of adaptive body biasing. The zero body bias can achieve a 90% yield only at a lower target frequency, so the leakage currents of the ZBB is unfairly biased to a lower value compared to the leakage currents associated with the FBB and ABB cases.

The benefits of adaptive body biasing are clearly illustrated in the above table. ABB allows one to increase the target operating clock frequency compared to the ZBB case, and also shows a significant reduction in active leakage currents compared to the FBB case. In the first scenario where the standard deviation is larger at 12.1% (vs. 9.1%), it can be seen that the expected leakage currents of the ABB case is actually smaller than the expected leakage of the ZBB case, even though it is operating at a higher target frequency as well.

The table also illustrates how the critical path variation statistics impact the benefits of adaptive body biasing control. When critical path frequency standard deviations are smaller, the spread of frequency samples are more tightly coupled. As a result, the target frequency is slightly higher than the more dispersive case with the larger standard deviation. This has two basic effects: in one sense there is a trend for leakage currents to increase because the expected operating frequency is higher. However, an opposite trend tends to lower expected leakage currents because it is far less likely to have very fast, leaky devices since the distribution is tighter. Depending on the ratio between these two effects, the relationship between the magnitude of leakage currents between the large standard deviation and smaller standard deviation samples will vary. For example in the zero body bias case, the leakage current of the large standard deviation case is larger than the leakage current of the small standard deviation case because there is a much greater chance of having fast leaky devices. In the FBB case, both distributions are shifted higher with forward body bias. However, in this case, the expected leakage currents of the slightly smaller standard deviation sample is actually a bit higher. This can be attributed to the fact that with forward body bias, more samples that were on the slow side could be shifted up to a higher frequency with FBB. On the otherhand, for the large standard deviation case, there are many samples which were simply too slow to be shifted, and furthermore the target operating frequency was lower in this case. Finally, in the ABB case, the leakage current of the smaller standard deviation is now much higher than the case for the larger standard deviation. This is because with ABB, the sample with the large standard deviation can be significantly tightened by slowing down the many fast samples. As a result, the active leakage currents can be reduced significantly.

The relationships between active leakage currents between large and small standard deviations for the ZBB, FBB, and ABB cases described above is not fixed. Fundamentally, one can exhibit higher or lower leakage currents depending on the complex interactions between different mechanisms that effect leakage currents. For example, having a larger target frequency, having a larger probability of more fast samples, and varying effectiveness of reverse and forward body biasing on very fast samples versus mildly fast samples, etc., can all effect the total expected active leakage currents. This difficulty is compounded by the fact that two scenarios with different standard deviations will have different target operating clock frequencies so the leakage currents are not easily comparable. Indeed as seen in simulation examples later in this chapter, for some target frequency settings, the FBB and ABB expected leakage current of the larger standard deviation could turn out to be higher than the scenario with the smaller standard deviation.

However, one trend that is fundamental (and can be seen in all examples in this section) is that the percent savings in active leakage current reduction by using an adaptive body biasing approach over a forward body biasing methodology is much greater in the case of a larger standard deviation than a smaller standard deviation. This basically is true because distributions with a larger standard deviation have a larger spread of fast devices, and thus adaptively slowing down these devices gives large savings in leakage currents. Furthermore, the larger standard deviation will typically have a lower operating target clock frequency as well, which permits the ABB controller to lower the leakage currents even more.

N=10 Case

Simulations were next performed assuming a hypothetical system consisting of 10 different local blocks. In effect, this can be thought of as a chip that is constructed using 10 of the previous local islands each characterized by the normal distribution N(TTTT freq, sigma). Again, a yield target of 90% was assumed, which means that the target frequency must be chosen so that there is a 90% chance that all system blocks will function properly. This in turn translates into choosing a target frequency such that the probability of a single block satisfying this frequency target is greater than 99% (corresponding to the tenth root of 0.9).

Because the test chip is now considered to be significantly larger than the previous example, the target frequency will be much lower than before, since every subblock in the system must be functional in order to ensure that the chip operates properly at the target frequency. For a standard deviation of 12.1%, the target normalized frequency must be set to 0.8378, while for the case with a standard deviation of 9% the target normalized frequency would be set to 0.9024, where both scenarios assume that forward body biasing is available to maximize performance for a given yield.

The figures below show the simulation results showing the ZBB, FBB, and ABB frequency distributions (the PMF and CDF) for the 12.1% standard deviation.

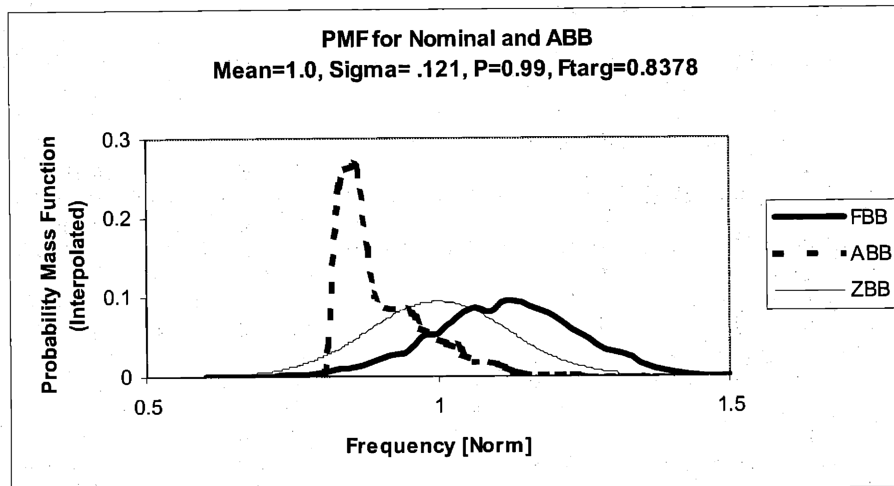


FIGURE 5-24. PMF for system consisting of N=10 blocks, standard deviation .121, target frequency 0.8378

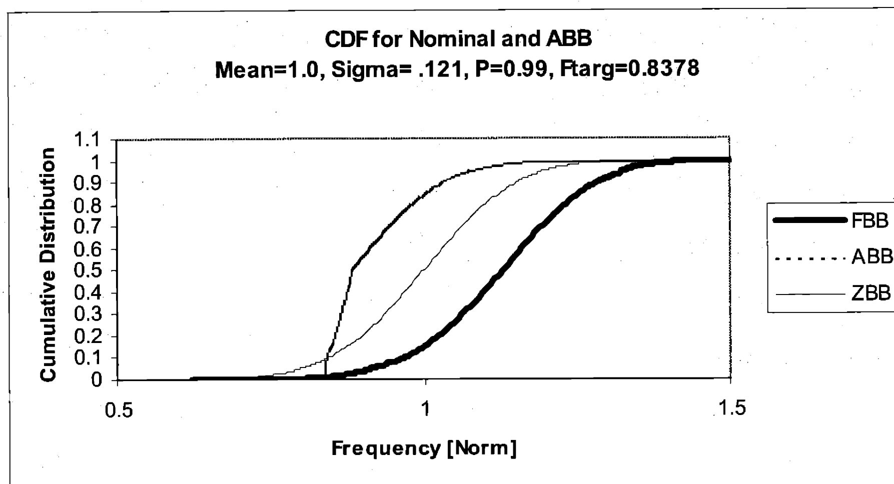


FIGURE 5-25. CDF for system consisting of N=10 blocks, standard deviation .121, targ frequency 0.8378

An interesting feature of the probability distributions is that the there is a sharp peak in PMF at the target frequency, (and a high slope in the CDF at the target frequency), but there is a more gradual roll off and wider distribution at higher speed samples. Because the target frequency for this chip needs to be substantially lower frequency than the nominal TTTT frequency, there are a large number tuning ranges that need to be compensated by the adaptive body biasing scheme. However, for extremely fast devices the maximum reverse body bias range saturates, and the operating frequency cannot be slowed down enough to reach the target frequency. As a result, if target frequencies are too low, then the ABB distribution curves will not exhibit a sharp peak, but instead slower roll-off.

Simulations were also performed showing the frequency distributions for the case where the standard deviation is 9.07%. As predicted, the ZBB and FBB distributions are tighter than before, and the ABB distributions are also more accurately tuned to the threshold voltage. This is because the frequency range of the samples that need to be reverse biased is reduced

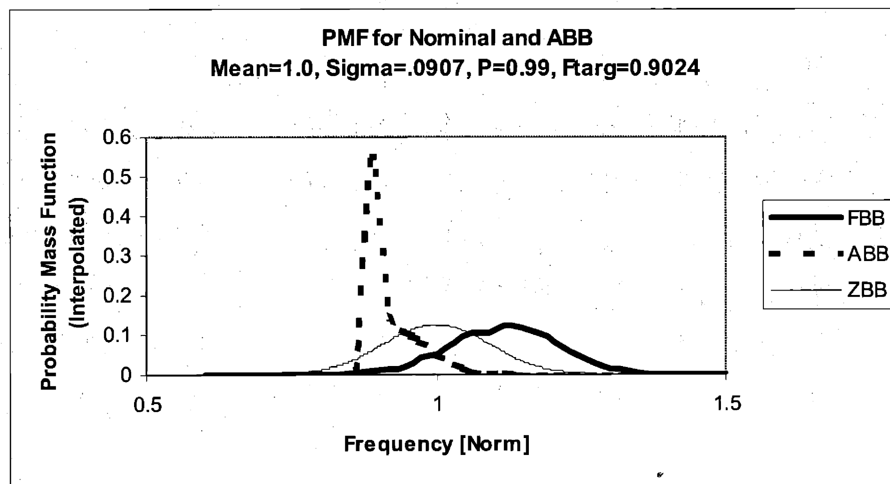


FIGURE 5-26. PMF for system consisting of N=10 blocks, with standard deviation .0907, targ frequency 0.9024

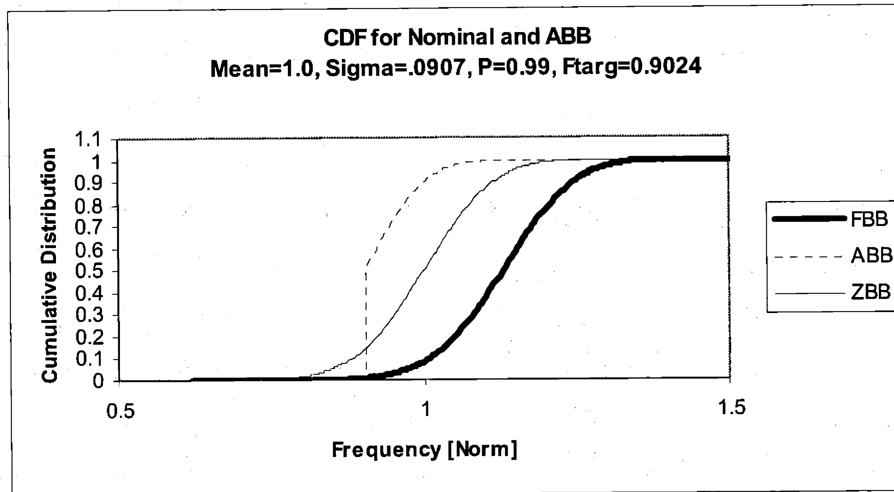


FIGURE 5-27. CDF for system consisting of N=10 blocks, with standard deviation .0907, targ frequency 0.9024

The simulation results for the N=10 block system is shown in the table below. Because the target frequency is lower, the adaptive body biasing control mechanism is more effective than in the previous case for the smaller chip consisting of only N=1 blocks.

TABLE 5-2. Simulation results for system with N=10 blocks, and sigma 1 and sigma2

	P=0.99 Target Frequency (normalized)	**% Change in target freq w/ forward bias**	**Expected Leakage Currents (normalized)**	**% Reduction in E(leakage) from FBB condition (for same sigma)**
ZBB (s=.121)	0.7200	0%	0.4042	-58%
FBB (s=.121)	0.8378	16.4%	0.9630	0%
ABB (s=.121)	0.8378	16.4%	0.2437	-74.7%
ZBB (s=.09)	0.7869	0%	0.3698	-60.9%
FBB (s=.09)	0.9024	14.7%	0.9455	0%
ABB (s=.09)	0.9024	14.7%	0.2541	-73.1%

Interestingly, for this test case where the target operating frequency is reduced because of the large chip size (10 blocks versus 1 block) the adaptive body bias version turns out to have both superior performance and lower leakage power compared to the

zero body bias case. As shown in the above case for the 12.1% standard deviation model, the ABB (forward bias enabled) target operating frequency is 0.8378 compared to a ZBB target operating frequency of only .7200, thus representing a 16.4% improvement in performance. However, the leakage current of the zero body bias case is 0.4042, yet the leakage current of the ABB case is only 0.2437, which corresponds to a 40% reduction in leakage currents as well. Clearly, this simulation illustrates how adaptive body biasing can be very useful in tightening distributions and lowering leakage currents when critical variations are significant and distributed throughout a large chip.

N=100 Case

As a final example illustrating the benefits of adaptive body biasing, simulations were explored for a test chip scenario consisting of N=100 separate local islands. For this extreme case, the target frequency must be chosen to be even lower than the previous example. For a yield of 90%, all 100 local islands must be greater than the target frequency, which translates that each block must have a probably of greater than 99.9%. In this case, the target operating frequency for the 12.1% standard deviation case corresponds to 0.7338, while the 9.1% standard deviation case yields a 0.8378 normalized operating frequency. Compared to the N=10 scenario, the N=100 chip operates at reduction in target operating frequency by another 12.4% and 7.2% respectively for the larger and smaller standard deviation models.

Probability distributions (probability mass function and the cumulative distribution function) for the 12.1% standard deviation are shown below.

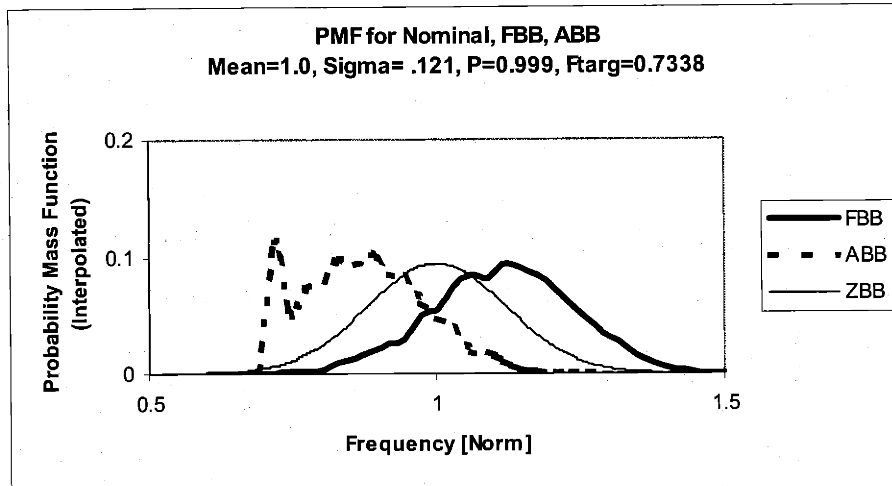


FIGURE 5-28. PMF for system consisting of N=100 blocks, standard deviation .121, targ frequency 0.7338

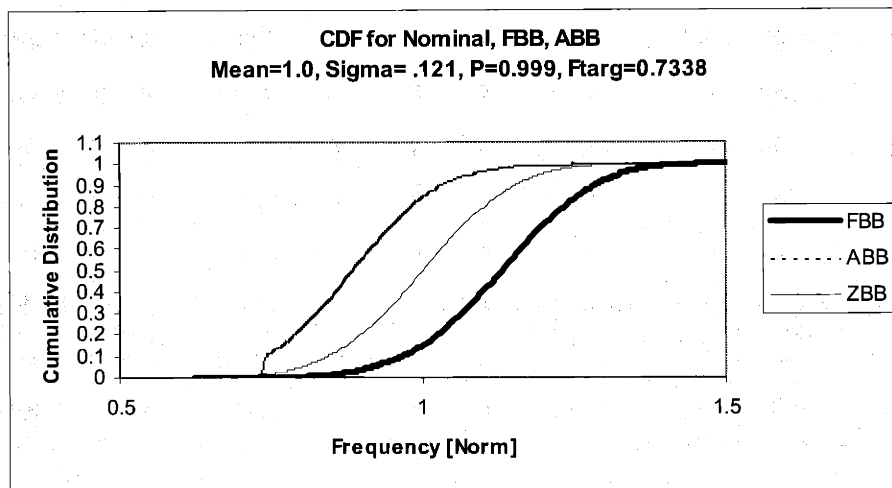


FIGURE 5-29. CDF for system consisting of N=100 blocks, standard deviation .121, targ frequency 0.7338

For this scenario, the PMF shows an interesting peaking structure at the target operating frequency of 0.7338 [normalized Hz]. However, because the target frequency is so low, more than half of the frequency samples were limited by the maximum reverse bias range of 1.5 Volts. As a result, only a small number of samples could be biased to the appropriate target frequency. The ABB probability distribution at higher frequencies thus retain the same basic distribution form as that of the nominal distribution, except that most

of these samples are all shifted by the same maximum reverse bias amount. However, the PMF of Figure 5-28, as described earlier is actually a rebinned interpretation of the simulation sample points. As a result, the binning resolution has a large impact on the actual appearance of the PMF, and thus the details of the ABB distribution waveform can vary for different binning scenarios. The general characteristics of having a small peak centered about the target operating frequency, followed by a "shifted distribution" would still be apparent though. This example though illustrates how the CDF distribution can actually be better at demonstrating the statistics for the frequency random variables. As described earlier the CDF does not suffer from any binning issues because the function is a continuous function that simply integrates the probabilities below any particular value. The CDF of Figure 5-29 illustrates how there is only a small region where the slope is higher, which corresponds to a small peak in the PMF curves. The rest of the CDF curve tends to track the shape of the ZBB cumulative distribution function, which indicates that there isn't much squeezing of frequencies because the reverse body bias amounts have saturated to the maximum applied body bias set by the biasing circuitry.

Simulation were also performed for the N=100 case where the standard deviation was smaller at 9%. As can be seen below, the probability distributions are slightly tighter than the previous case and the target frequency is slightly higher. As a result, after adaptive body biasing, the frequency squeezing range shows a tighter peak as well.

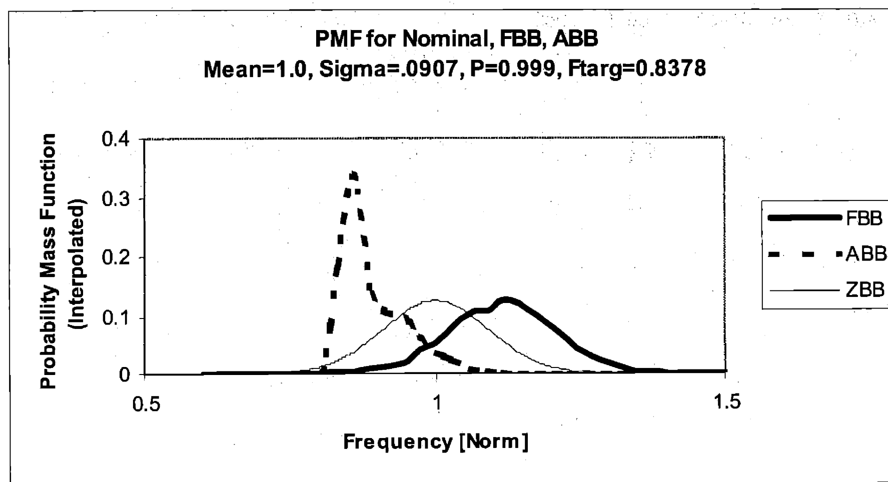


FIGURE 5-30. PMF for system consisting of N=100 blocks, standard deviation .0907, targ frequency 0.8378

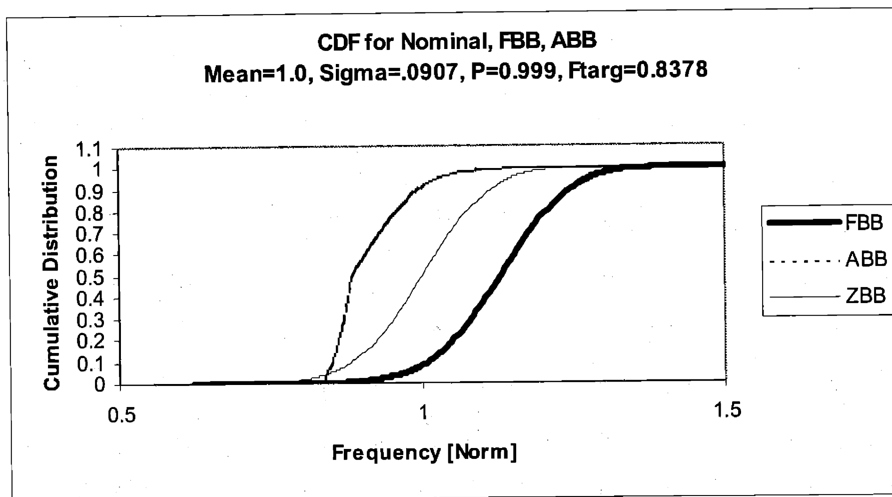


FIGURE 5-31. CDF for system consisting of N=100 blocks, standard deviation .0907, targ frequency 0.8378

The simulation results for the N=100 case are summarized in the table below. Unlike before for the N=1 and N=10 cases, this example shows that the ABB expected leakage current of larger standard deviation case is slightly larger than for the smaller standard deviation case. This can be attributed to the fact the larger standard deviation case had too many high frequency samples that could not be slowed down enough. As theorized earlier however, as variations become larger, the effectiveness of ABB compared to a FBB scenario still improves because there are larger opportunities to slow down fast devices. For example, for the larger standard deviation yield a savings of 78.4% in leakage currents when ABB is utilized, while the smaller standard deviation case yields a savings of 78%. Furthermore, these savings are significantly larger than the case for N=1 and N=10 systems. This illustrates how adaptive body biasing becomes more and more promising as parameter variations worsen with future technologies and systems like a system-

on-a-chip become larger and more complicated, giving rise to more local blocks that limit the critical operation of a chip.

TABLE 5-3. Simulation results for system with N=100 blocks, and sigma 1 and sigma2

	P=0.9 Target Frequency (normalized)	% Change in target freq w/ forward bias	Expected Leakage Currents (normalized)	% Reduction in E(leakage) from FBB condition (for same sigma)
ZBB (s=.121)	0.6246	0%	0.40167	-58.1%
FBB (s=.121)	0.7338	17.48%	0.95753	0%
ABB (s=.121)	0.7338	17.48%	0.20726	-78.4%
ZBB (s=.09)	0.7200	0%	0.36736	-60.9%
FBB (s=.09)	0.8378	16.37%	0.94013	0%
ABB (s=.09)	0.8378	16.37%	0.20639	-78%

5.9 Adaptive Body Biasing Effectiveness

Body biasing techniques have been shown to be effective at reducing subthreshold leakage currents during standby and also active modes by using reverse body bias to increase device threshold voltages. Of particular value were techniques to provide adaptive body biasing approaches to tune threshold voltages both at the die level and the intradie level. The effectiveness of these adaptive body techniques largely depends on the variation amounts that can be compensated out, although as illustrated in the previous example, first order simulations show that for reasonable variations, greater than 50% savings in leakage currents can be achieved with even more savings available for larger dies that use multiple bias generators.

One constraint for effective use of ABB approaches is that the overhead in the bias generators must be less than the savings in energy achievable. Fortunately, with a digital implementation approach, the feedback loop can be turned off after lock, so the overhead in maintaining the body bias value is very low. The loop can be refreshed periodically (on the order of microseconds to seconds) to compensate for slowly varying parameters such as temperature or hot carrier effects. Thus the energy cost of locking the adaptive body bias generator is amortized over extremely long periods. Since leakage currents can easily contribute upwards of 30% of total active power dissipation in future technologies the

reduction in leakage power through adaptive body biasing will easily be greater than the overhead costs. As a generic example, consider a chip with normalized power dissipation of 1W, with active leakage power contributing to 30% of the total. With an adaptive body bias approach that reduces active leakage currents by 50%, the savings in power would be 150mW. Assuming that the ABB generator is 5% of the chip area, the ABB power consumption might be on the order of 50mW of power. Assuming that the ABB generator takes a long time of 1μs to lock (because of slow dynamics for the digital loop), then the energy consumed by the ABB generator would be 50nJ, but this could easily be balanced out by the savings in reduced leakage currents if the chip operates larger than 333nS. Since the refresh time for the ABB generator is on the order of milliseconds or seconds, clearly the savings in energy is several orders of magnitude greater than the overhead energy of the ABB generator.

Another constraint that limits the effectiveness of the ABB generator is the area overhead costs. This limits the granularity at which one can break a larger chip into regions that are independently tuned. For example if the regions are too small, then the ABB power overhead might be too large, and the reduced die space available for logic functions may become unacceptable. However, it only makes sense to subdivide a chip into independent regions if the sub-blocks can be decoupled into regions characterized by there own local critical paths, and only if the intra-region variations are significant between these blocks. This tends to make the multiple biasing strategy only applicable to very large chips such as a system-on-a-chip, where there are many large sub-blocks that could be independently tuned. For these scenarios, the overhead in ABB area is limited because the generators can be made with only a few blocks.

Chapter 6

Optimal V_{CC}-V_t Circuit Operation

In the previous chapter, an adaptive body biasing control methodology was described that dynamically adjusts threshold voltages in a triple well process to compensate for parameter variations. With body biasing, (forward and reverse), device threshold voltages could be raised or lowered depending on speed requirements for a device. Maximum reverse body bias could be used to place a circuit in an idle low leakage state during the standby state, while an adaptively selected body bias could be adjusted during active runtime to tune the circuit block so that it only operates as fast as necessary and thus reduces active leakage currents and improves process yields. For these adaptive body biasing methodologies a constant supply voltage scheme was assumed so that only threshold voltages were tuned.

With the ability to tune both threshold voltages and supply voltages though, digital circuits can actually be configured to operate at a theoretical minimum power level during the active mode. This minimum power operating point balances subthreshold leakage power with dynamic switching power so that overall power dissipation is minimized. In modern integrated circuits, typical operating voltages and threshold voltages are much larger than optimal. From a power perspective there is much opportunity to save power with proper choice of V_{CC} and V_t for a given circuit. This chapter explores techniques to

provide automatic V_{CC} and V_t tuning methodologies for a circuit, and also explores how the optimal operating point varies with different operating conditions, which makes dynamic V_{CC} and V_t tuning important for aggressive low power control techniques.

6.1 V_{CC}- V_t optimization overview

For a given chip or circuit, it is possible to choose an appropriate V_{CC} and V_t combination that optimizes the combination of dynamic power and leakage power for a given performance. Typically, for a fixed performance requirement, there is a locus of V_{CC} and V_t combinations that could be used. For example, a reduction in supply voltage can be compensated by a reduction in V_t, thereby maintaining high performance operation. As the supply voltage drops, the dynamic power is reduced quadratically, but as the threshold voltage drops the leakage power is increased exponentially. The optimum V_{CC} and V_t combination corresponds to the point where the incremental decrease in dynamic power with a change in V_{CC} is offset by the incremental increase in leakage power due to a change in V_t.

For a single chip, there is an optimal supply voltage and threshold voltage operating point that minimizes the overall energy for a given circuit operating condition. For the active leakage reduction technique presented in the previous chapter, fine grain adaptive body biasing was described where a system could be divided into individual local regions that are locally compensated. Unfortunately, this fine grain control can not be directly applied to supply voltage scaling because of difficulties that arise from interfacing between and generating different voltage ranges for each local logic block. As a result, V_{CC} / V_t optimization is geared towards chip level control where the die as a whole is tuned to have a global optimum supply and global optimum threshold voltage. Although theoretically different sub-blocks in a system could operate with their own fine grain V_{CC} / V_t operating point and result in lower power levels (than by using die level control), it would be difficult to implement in an actual circuit. A simple alternative to provide some fine grain control over local block variations is to combine the adaptive body biasing scheme described in the previous section for local block control with a global V_{CC} / V_t optimization scheme for the whole chip. This might be accomplished by first finding an

optimal V_{CC} / V_t operating point for the entire chip, and then to selectively apply reverse body biasing as necessary to ensure no block operates faster than necessary. This can help compensate for parameter variations or systematic offsets between block operating speeds within a chip.

Another way to further reduce subthreshold leakage currents during the active mode after global V_{CC} / V_t scaling is to selectively slow down gates (within a block, or within the whole chip) that are not in the critical paths. One assumption in the earlier analysis is that in order for the chip (or a local block region) to stay on a constant performance locus as one scales the supply voltage, all the device threshold voltages must scale as well. Technically, this is not true because only the critical paths need to scale this aggressively because non critical gates within these circuit blocks can still be operated with higher threshold voltages. As a result, after a minimum V_{CC} / V_t operating point is reached for a circuit, it may be possible to further reduce active leakage currents by selectively slowing down gates that are not in the critical paths. This is similar to the standard dual V_t partitioning procedure described in the background chapter for reducing subthreshold currents during the active modes. However, in the context of the rest of this chapter, it is useful to simply consider all devices in the chip in the same fashion. This is reasonable because many aggressive datapath circuits are designed to be well balanced where a large fraction of the gates are in the critical path. The rest of this chapter simply explores die level V_{CC} / V_t optimization, where all devices are tuned in the same fashion, and local block variations (as described in the previous chapter) can be dealt with afterwards.

6.2 Theoretical V_{CC}-V_t Optimum

As described earlier, the optimal bias setting corresponds to the point on the V_{CC} / V_t locus (giving a fixed performance) where the incremental change in dynamic power due to V_{CC} scaling is offset by the incremental change in leakage power due to V_t scaling. Figure 6-1 below shows theoretical iso-performance curves for power versus supply voltage, where the threshold voltage is implicitly a one-to-one function of V_{CC}. The power curves are divided into two parts. The right curve $P_{dynamic}$ shows dynamic power versus V_{CC}

while the left curve $P_{leakage}$ shows leakage power versus V_{CC}. The sum of these two curves thus provides the total power dissipation for the circuit.

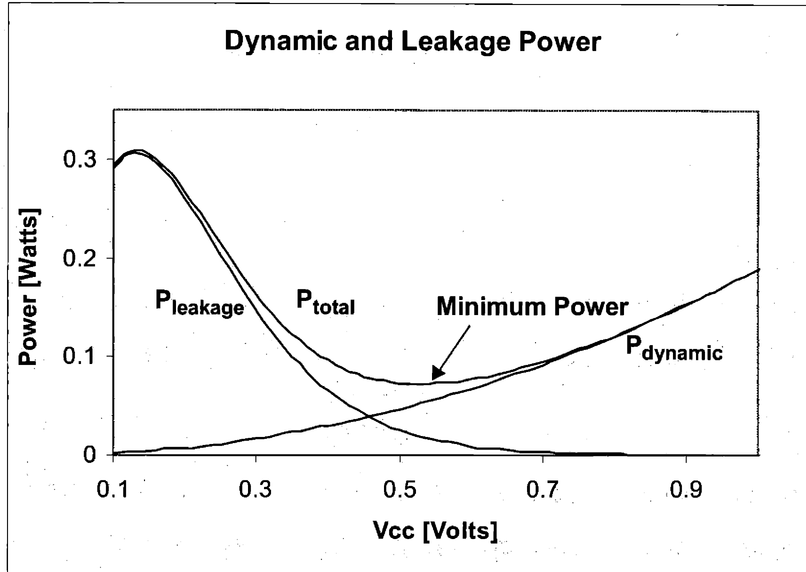


FIGURE 6-1. Optimal Vcc/Vt biasing point trading off dynamic power with leakage power for constant performance.

The minimum of the total power curve corresponds to the point where the slope of the two curves are opposite in sign but equal in magnitude. It is important to note that the minimum operating point isn't necessarily located where the two curves intersect, but rather where the two curves have equal and opposite slopes. This is true because the minimum power point occurs in a region where one curve is monotonically increasing while the other curve is monotonically decreasing, and the slopes change monotonically as well.

Several different metrics have been used in the literature to characterize low power circuit performance. Power delay products, energy delay products, and average power consumption have all been useful measures for comparing different circuits. Power delay and energy delay curves have been especially useful for comparing different circuits because both energy consumption as well as circuit speed are both taken into account in the metric. Some theoretical work was also done in [60] to mathematically compute minimum EDP values taking into account leakage and dynamic power. In our context of V_{CC} / V_t scaling though, the circuit of interest is fixed and the performance is also fixed for a

given application, yet the lowest power solution is still desired. As a result, it is useful to simply consider the average power as the metric of interest. For example, a DSP or a microprocessor might be required to operate at a specified clock frequency, so the optimal V_{CC} / V_t combination is simply the one that minimizes the total power consumption. In these scenarios, the chip frequency is fixed, so it does not make sense to minimize the power delay product or energy delay product, but rather to simply minimize overall power dissipation.

It is useful to construct a model to characterize active power dissipation in order to better understand the interaction between dynamic power and leakage power, and how the optimum V_{CC} / V_t operating point varies as performance constraints change. The propagation delay for a CMOS gate written again for reference is

$$T_{pd} \propto \frac{KV_{CC}}{(V_{CC} - V_t)^{\alpha}} \qquad \text{(EQ 6-1)}$$

where K is a constant, and α is the velocity saturation term that models short channel effects. This equation, for a given performance requirement thus defines a locus of V_{CC} and V_t combinations that will satisfy the overall performance. Assuming that the critical path delay (between registers for example) has a logic depth of n gates, then the operating frequency can simply be represented as

$$f \propto \frac{(V_{CC} - V_t)^{\alpha}}{nKV_{CC}} \qquad \text{(EQ 6-2)}$$

The power dissipation can be written as the sum of two different components that are dominant in current and future technologies. The dynamic power dissipation due to charging and discharging of capacitances during active computation can be represented as

$$P_{dynamic} = C_{eff}V_{CC}^2 f \qquad \text{(EQ 6-3)}$$

whereas the subthreshold leakage power component (ignoring DIBL) can be written as

$$P_{leakage} = VccI_0 10^{\frac{-V_t}{S}} \quad \text{(EQ 6-4)}$$

The total power consumption for a digital circuit is dominated by these two components giving

$$P_{total} = C_{eff}V_{CC}^2 f + VccI_0 e^{\frac{-V_t}{S}\ln(10)} \quad \text{(EQ 6-5)}$$

In the above equations, C_{eff} corresponds to the effective switched capacitance, S the subthreshold slope, and I_0 the nominal subthreshold leakage current constant (i.e. Ids when Vgs= Vt).

Eq 6-5 models dynamic power consumption and subthreshold leakage currents. As described earlier, other sources of power dissipation exist, but are small in comparison. For example junction currents are negligible, and for well designed circuits with comparable rise and fall times, short circuit currents are negligible as well (or can be lumped into C_{eff}). However, research has shown that future technologies may suffer from large amounts of gate leakage, which also must be considered in future low power circuit techniques. This is beyond the scope of this research through, but may become an additional leakage mechanism that must be addressed in future designs.

The minimum V_{CC} / V_t operating point can be mathematically derived from the above equations. One straightforward way to do this is with the Lagrange multiplier technique where the power in Eq 6-5 is minimized subject to the constraint from Eq 6-2. A more direct approach could also be to simply use direct substitution of Eq 6-2 into Eq 6-5 to give

$$P_{total} = C_{eff}V_{CC}^2 f + V_{CC}I_0 e^{\frac{-(V_{CC} - (nfKV_{CC})^{1/\alpha})}{S}\ln(10)} \quad \text{(EQ 6-6)}$$

which can be differentiated with respect to V_{CC} and set to 0 to compute the extrema. The resultant expression can then be solved numerically for the optimal V_{CC} and V_t combination that minimizes overall power.

Differentiating Eq 6-6 and setting to zero is mathematically equivalent to equating the slope of the dynamic power curve with the negative of the slope of the leakage power curve (both as functions of V_{CC}). Because the dynamic power curve is monotonically increasing over all V_{CC} values, the leakage power is monotonically decreasing for V_{CC} values of interest, and the slopes are monotonically increasing as well for these regions of interest, there is a single optimum point where the slope of both curves are equal and opposite.

Intuitively, this point can be understood to be an optimum operating point. With the constraint that the slopes of the power curves change monotonically, it is easy to see why the slopes of the two curves must be equal and opposite at the optimum point. Assume an intermediate abscissa value, V_0, is chosen for both curves so that the leakage power is P_{leak0} and the dynamic power is P_{dyn0}. If the voltage variable changes by ΔV, then the dynamic power changes to

$$P_{dyn}(V_0 + \Delta V) = P_{dyn0} + \Delta V \frac{d}{dV}(P_{dyn}(V))\Big|_{V = V_0} \qquad \text{(EQ 6-7)}$$

where the derivative is positive.

Similarly the leakage power will change to Eq 6-8 below where the derivative yields a negative value.

$$P_{leak}(V_0 + \Delta V) = P_{leak0} + \Delta V \frac{d}{dV}(P_{leak}(V))\Big|_{V = V_0} \qquad \text{(EQ 6-8)}$$

If the magnitude in slope of P_{leak} and P_{dyn} differ, then a change ΔV in one direction will cause the total power (P_{dyn} + P_{leak}) to increase from the original value (P_{dyn0} + P_{leak0}), and ΔV in the opposite direction will cause total power to decrease from the original value. As a result it is incrementally more energy efficient to operate at a voltage shifted by ΔV to reduce overall power dissipation. However, with this appropriate shift in V, the slope with the higher magnitude will decrease in magnitude, while the slope with lower magnitude will increase in magnitude. This comes from the fact that the slopes of the curves are monotonic in nature. Clearly, there must be a crossover point where the slopes

are equal and opposite, where the change in power dissipation in the dynamic and leakage curves are balanced. In this case, shifting V by a small amount ΔV will cause one curve to have a higher slope than the other in magnitude, which is a non optimal condition. Therefore, this point also corresponds to the optimal point that minimizes the overall power.

6.3 Optimal V_{CC}/V_t Scaling Trends Based on Theoretical Models

The basic theoretical models obtained by substituting Eq 6-2 into Eq 6-5 that characterizes total power consumption can be useful for understanding how circuit parameters will effect the interactions between dynamic and leakage power. The optimal point corresponds to a balance between these two power sources, which can vary depending on the circuit parameters. To better quantify the interactions between leakage and dynamic power components, it is useful to examine the models using realistic parameters for a modern technology. Table 6-1 below shows the necessary parameters that are based on technology values for a modern SSH4 microprocessor from Hitachi. The values reflect a reasonable technology used today for low power applications, and show that significant energy savings can be achieved by optimally scaling V_{CC} and V_t.

TABLE 6-1. Typical model parameters based on Hitachi SSH4 low power microprocessor.

Ceff	Effective Switched Capacitance	****
K	delay constant	****
n	logic depth	****
alpha	velocity saturation term	****
I_0	Leakage constant (Vgs=Vt)	****
S	Subthreshold slope	****

Many of the equation parameters shown above are technology dependent constants that a designer must simply adhere to. Other parameters such as the effective switched capacitance C_{eff}, the logic depth n, and target frequency f_{targ}, can be adjusted based on architectural design choice or system operating conditions. As a result, it useful to study how these parameter choices affect dynamic power and leakage power interactions, and ultimately how they impact the optimal V_{CC}-V_t point.

6.3.1 Role of switched capacitance on optimal V_{CC}-V_t scaling

The switched capacitance C_{eff} corresponds to the average capacitance that is switched per cycle. When this value increases, the dynamic power consumption increases without directly impacting leakage currents. As can be seen in the equation for total power consumption, an increase in C_{eff} will correspond to an increase in the dynamic power equation, and the equilibrium point will shift as seen in the figure below

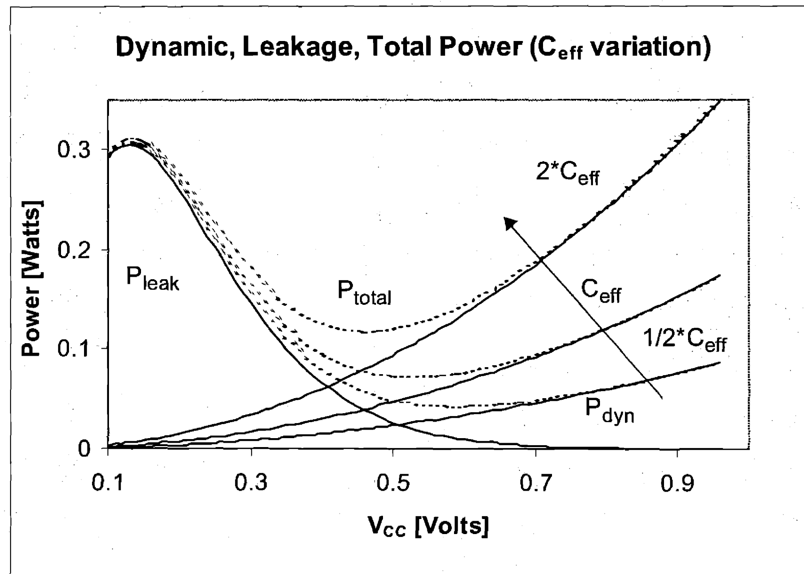


FIGURE 6-2. Effect of effective switched capacitance variation on optimal V_{CC}/V_t biasing points.

As a circuit becomes more heavily dynamic power dominated (by increasing the switching capacitance), the optimal V_{CC}-V_t point moves towards lower supply voltages and lower threshold voltages. This effectively weights the circuit more strongly with leakage currents because the larger dynamic power component can be more effectively controlled by lowering supply voltages.

6.3.2 Role of I_0 on optimal V_{CC}-V_t Scaling

Similarly, if the leakage constant I_0, increases, the reverse effect occurs. The leakage component increases without effecting the dynamic power curve, and the equilibrium point shifts to the right as illustrated in Figure 6-3.

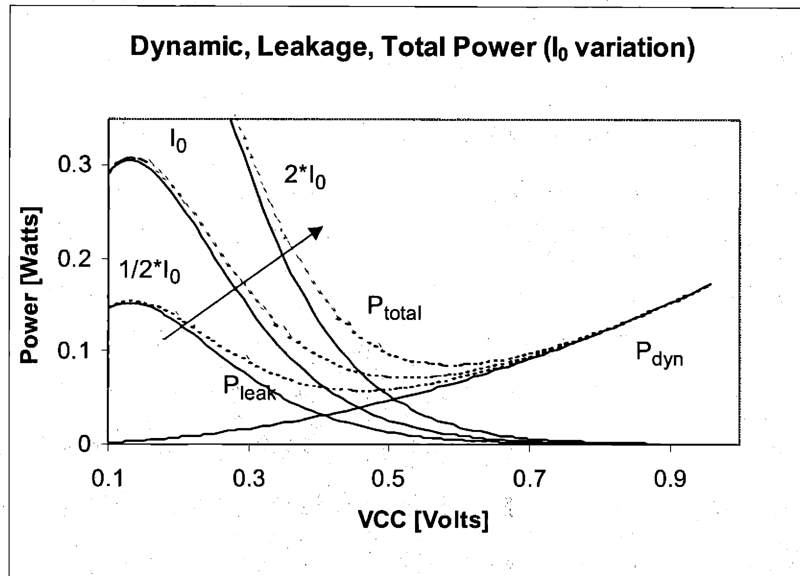


FIGURE 6-3. Effect of I_0 variations on optimal V_{CC}/V_t biasing points.

This occurs because the optimum point should more strongly reduce leakage currents by increasing the threshold voltage, and consequently increasing the supply voltage as well. In a real implementation though, it is likely that any change in I_0 will also impact C_{eff}, the effective switched capacitance. This is because from a circuit point of view, an increase in I_0 would probably arise from an increase in the circuit effective width, which would also translate into a larger switched capacitance due to increased loading. As can be seen from the figure, the difference in optimal operating point between a circuit condition where I0 varies from 50% below nominal to 100% above nominal, for example, can be quite substantial.

6.3.3 Optimal V_{CC}-V_t positioning

From the previous two sections one can see a general trend how the optimal V_{CC} / V_t point lies in relationship to the dynamic and leakage power curves. As described before, the optimal point lies where the slopes of the dynamic and leakage power curves are equal and opposite. This point can mathematically lie to the right or left of the crossover point between the leakage power curve and the dynamic power curve as seen in Figure 6-1. If the optimal V_{CC} / V_t point lies to the right of the crossover point, then the optimal operating condition will be one where the dynamic power component is larger than the leakage power component. If on the otherhand the optimal operating point occurs at the same V_{CC} value as the intersection point then the dynamic and leakage power components are equal, and finally in the case were it crosses over to the left, the leakage power will be dominant. Practically however, the optimal V_{CC} / V_t point always crosses over to the right of the intersection point in the power vs. V_{CC} curves for dynamic and leakage components. This makes intuitive sense because it would be wasteful from a thermodynamics perspective to have leakage power (which does not contribute to any computation) be greater than the dynamic power (which is directly related to computation).

Mathematically, one can calculate what conditions will ensure that dynamic power is greater than the leakage power at the optimal V_{CC} / V_t operating point. A simple way to determine this condition is to go back to the intuitive argument for deriving the optimal V_{CC} / V_t operating point in the previous section. If the dynamic power is more dominant than the leakage power at the optimal V_{CC} / V_t point, then at the mathematical intersection between the dynamic and leakage power curves, the slope of the leakage power curve in magnitude must be greater than the slope of the dynamic power curve. This means that the optimal point must lie to the right of the intersection point, where the dynamic power will be greater than the leakage power. This condition is illustrated below in Eq 6-9 and Eq 6-10 where V_0 corresponds to the voltage where the dynamic power curve and leakage power curves cross over.

$$P_{dyn}(V_0) = P_{leak}(V_0) \qquad \text{(EQ 6-9)}$$

$$\frac{d}{dV}P_{dyn}(V)\Big|_{V=V_0} < -\frac{d}{dV}(P_{leak}(V))\Big|_{V=V_0} \quad \text{(EQ 6-10)}$$

Substituting parameters into these equations yield

$$C_{eff}V_0^2f = V_0I_0e^{\frac{-V_t(V_0)}{S}\ln(10)} \quad \text{(EQ 6-11)}$$

where the threshold voltage can be defined as a function of the supply voltage

$$V_t(V_0) = (V_0 - (nfKV_0)^{1/\alpha}) \quad \text{(EQ 6-12)}$$

and

$$2C_{eff}V_0^2f < -\frac{d}{dV}\left(VI_010^{\frac{-V_t(V)}{S}}\right)\Bigg|_{V=V_0} \quad \text{(EQ 6-13)}$$

These equations can be solved numerically to calculate the parameter boundaries that will ensure that the dynamic power component will be greater than the leakage power component at the optimal V_{CC} / V_T operating point.

Analytically, Eq 6-11 - Eq 6-13 can also be combined and reduced to the constraint

$$3\frac{S}{\ln(10)} < V_0\left[1 - \frac{1}{\alpha}(nfKV_0)^{\frac{1}{\alpha}-1}nfKV_0\right] \quad \text{(EQ 6-14)}$$

For reasonable circuit parameters, the term in the bracket of Eq 6-14 is less then unity, and as a result that condition will still be met if the following more stringent criteria is true as well

$$V > 3\frac{S}{\ln(10)} \quad \text{(EQ 6-15)}$$

For a typical subthreshold slope value of 90mV/ Decade, Eq 6-15 shows that if the cross-over point V_0 occurs at a point greater than 0.1V, then the dynamic power component will dominate the leakage power component at the optimal V_{CC} / V_t operating point. This criteria is easily met in general, and thus one can see that for reasonable parameters that dynamic power will be larger than leakage power at the optimal bias point as predicted.

6.3.4 Role of frequency target on optimal V_{CC}-V_t scaling

In the previous examples, the impact of increasing C_{eff} and I_0 illustrated how the circuit optimal V_{CC} / V_t point varies. For these conditions, the chip operating frequency was held constant and therefore the optimum point stayed on the same V_{CC} / V_t locus that defined the fixed frequency. If the chip operating frequency does change, the optimal V_{CC} / V_t operating point will also shift, and will actually reside on a different V_{CC} / V_t locus curve that defines the new frequency. Figure 6-4 below shows theoretical V_{CC} / V_t locus curves that define constant frequency targets.

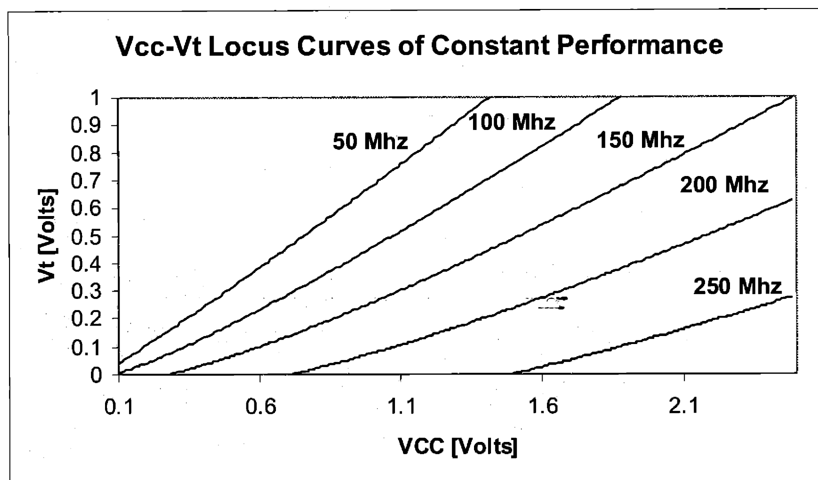


FIGURE 6-4. Constant frequency curves showing isoperformance V_{CC}-V_t locus based on theoretical models.

When frequencies increase, the optimal operating conditions usually result in a slight decrease of the threshold voltage and a slight increase of the supply voltage, both of which tend to improve performance. This makes intuitive sense because for higher frequencies, one cannot just rely on lower threshold voltages to meet targets, but also larger

supply voltages as well. Figure 6-5 below shows typical curves that illustrates total power consumption (leakage plus dynamic) versus V_{CC} for several different frequency targets. As can be seen, as frequency increases, the curves tend to move towards higher V_{CC} values, and higher power dissipation levels.

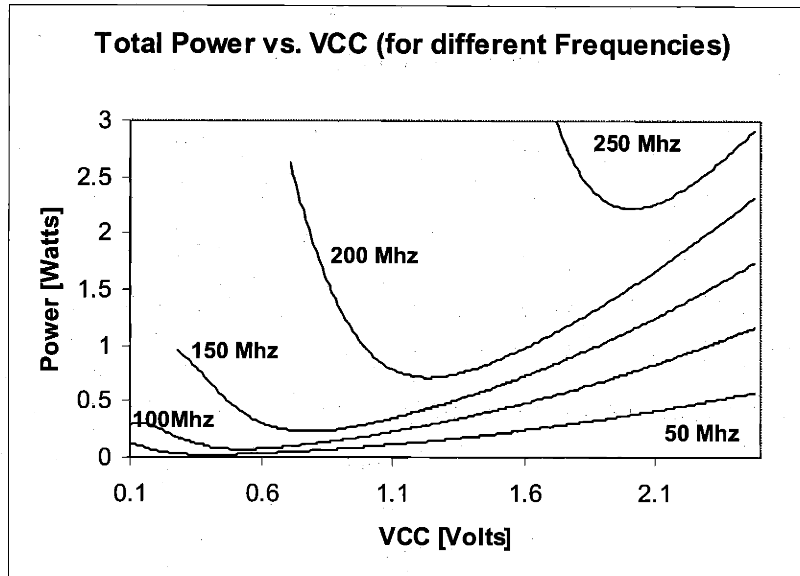


FIGURE 6-5. Power vs. V_{CC} curves for different target frequencies based on theoretical models.

Similarly, Figure 6-6 shows power versus threshold voltage for constant performance curves. The relationship between Figure 6-5 and Figure 6-6 is simply a remapping of the curves onto a different abscissa variable. For a fixed performance level, the mapping from V_{CC} to V_t can easily be derived from the appropriate locus of points as illustrated in Figure 6-4. The trend shows that as frequencies increase, the optimal V_t point shifts towards lower values, which indicates that low V_t fast devices are more power efficient at high operating frequencies.

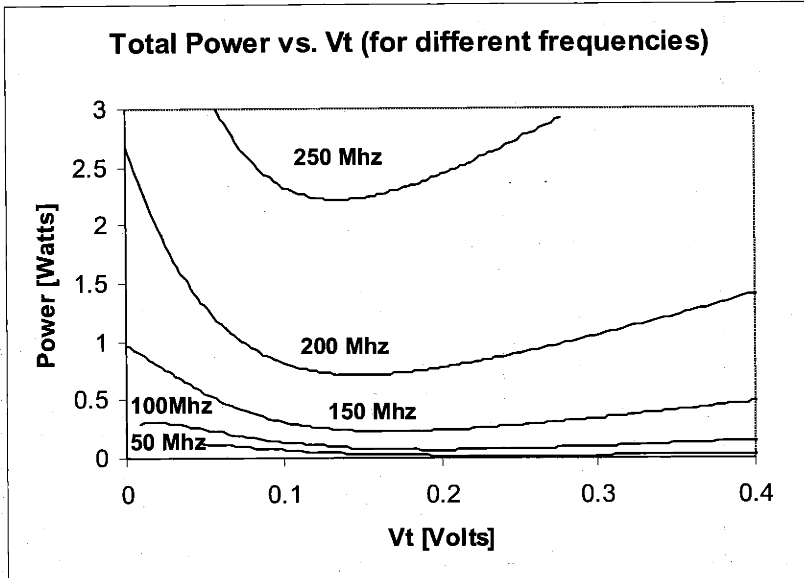


FIGURE 6-6. Power vs. V_t curves for different target frequencies based on theoretical models.

Figure 6-7 expands upon the previous two graphs, and shows in more detail how the optimum V_{CC} and V_t values will vary for a wide range of frequency targets. As described earlier, the general trend is that as frequencies increase, V_{CC} tends to increase and V_t tends to decrease.

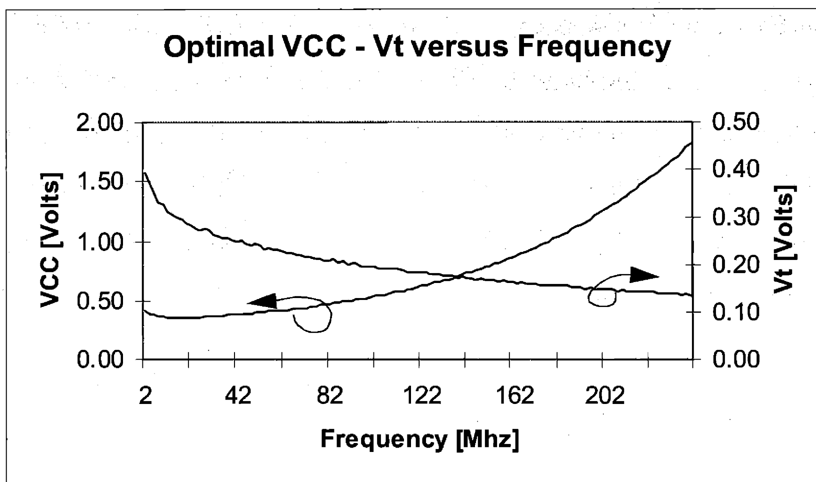


FIGURE 6-7. Optimal V_{CC} /V_t as a function of frequency for theoretical models.

One interesting feature of this curve though is that at higher operating frequencies, it is more optimal to use lower threshold devices. This is because at these operating conditions, the dynamic power becomes more dominant since this component of power increases linearly with frequency and quadratically with supply voltages. In order to minimize the dynamic power component, it is better to lower V_{CC} wherever possible, and to compensate by using faster devices (lower threshold voltages) that increases the fraction of total power dissipation due to leakage currents.

As frequencies decrease, the optimal V_{CC} / V_t operating points shift in the opposite direction so that threshold voltages increase (leakage currents reduce), while supply voltages decrease. However, at very low frequencies, leakage currents become more and more dominant because so few dynamic transitions occur. In this limit, one can see that as frequencies continue to drop, the actual optimal V_{CC} value increases. This is an unexpected result because typically one assumes that if the frequency of the chip drops, then the best way to lower power is to lower V_{CC}. However, from Figure 6-7, one can see that more power savings can be achieved by increasing the threshold voltage and increasing V_{CC}. This characteristic will become even more apparent as leakage currents become more dominant in future technologies.

6.3.5 Comparison to dynamic voltage scaling

The previous examples show how total power consumption can be minimized with proper choice of V_{CC} and V_t, which changes depending on the circuit structure, operating conditions, and operating frequencies. For a fixed technology where a single V_{CC} and V_t is used, the process can be engineered to minimize overall power consumption for a particular circuit operating at a particular frequency, but in general the V_{CC} and V_t values would not be optimal for other scenarios.

If a triple well, or similar technology, is available though, then both the threshold voltage and supply voltages can be dynamically tuned during runtime to minimize power dissipation even though workload conditions or system parameters might change. The interaction between V_{CC} and V_t becomes important as supply voltages continue to scale, and leakage power start to become a larger impact of total power dissipation. Figure 6-7

illustrated that the optimal V_{CC} / V_t operating point is a strong function of the desired operating frequency. In many applications, the operating frequencies are not fixed and can vary depending on workload. For example, a DSP that is processing image data will have different workloads depending on the dynamics of the image data, an encryption processor will require differing amounts of processing power depending on the level of security desired, or a microprocessor will operate at differing speeds depending on the application or the need to conserve battery power.

One straightforward approach that has been used in the past to address changing workload requirements is to simply lower the operating frequency as necessary. This lowers the dynamic power dissipation linearly because of it's direct dependency on frequency. This approach is only useful when the actual workload is reduced. If the amount of computation is the same but the frequency is lowered and the computation period increased, then there will be no net savings in energy. A better technique at reducing power for variable workload applications that has been proposed is to use dynamic voltage scaling (DVS)[62][63][64]. In addition to scaling the frequency, with dynamic voltage scaling, the supply voltage is adjusted so the circuit switches only as fast as necessary. This results in linear power savings from the frequency reduction as well as the quadratic dynamic power savings from voltage scaling. In effect, the dynamic voltage scaling is the dual of the adaptive body biasing methodology of the previous chapter. In the adaptive body biasing case, the threshold voltages were tuned to compensate for parameter variations so that circuits operate only as fast as necessary. The dynamic voltage scaling on the otherhand adjusts the supply voltage until circuits operate as fast as necessary depending on a variable target operating frequency.

However if threshold voltages are dynamically controllable, then both V_{CC} and V_t can be tuned more efficiently to optimize circuits for differing operating conditions. This technique is more general than dynamic voltage scaling because not only does it adjust V_{CC} to meet a target specification, but it also adjusts V_t to optimize the dynamic and leakage power components to minimize overall power dissipation. For applications where operating frequencies can vary in time, one can achieve much more power savings by optimizing both V_{CC} and V_t together than simply using a stand-alone dynamic voltage

scaling scheme. Figure 6-8 below gives a comparison of how total power varies with operating frequency for the cases where V_{CC} / V_t optimization is employed versus where stand-alone V_{CC} scaling is employed. Figure 6-8b merely shows a zoomed in portion of the graph for clarity.

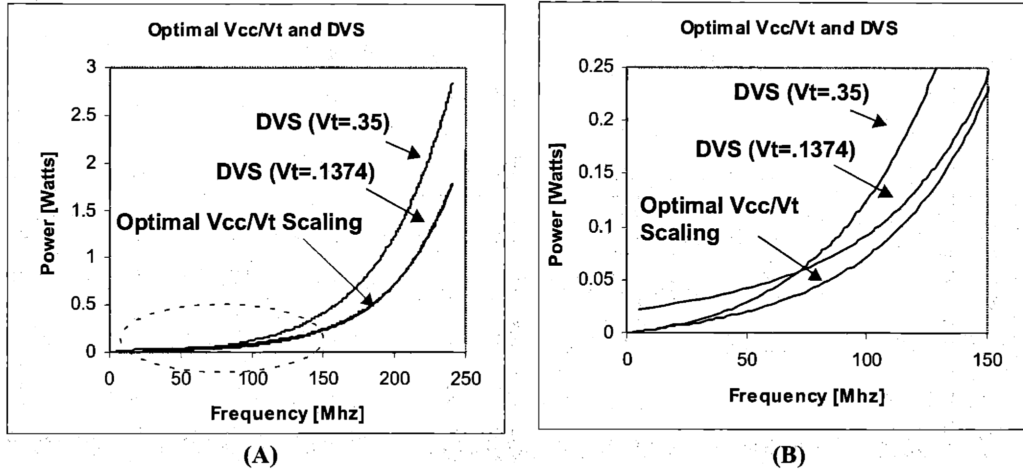


FIGURE 6-8. Optimal V_{CC} /V_t scaling power savings versus dynamic voltage scaling (V_t=.35 and Vt=.14)

Two cases for dynamic voltage scaling were explored. In the first case, the nominal process threshold voltage of 0.35V was chosen as constant. In the other case, it was assumed that the process threshold voltage was engineered to be a static 0.1374V, which corresponds to the optimal V_t value (with a corresponding V_{CC} of 1.7V) for an operating frequency of 240 Mhz. For high frequencies a threshold voltage of 0.35V is significantly larger than ideal, and as a result, the supply voltage must be larger than necessary and the total power dissipation deviates greatly from the optimum case. In the case where the target threshold voltage is fixed at 0.1374, this will give rise to optimal DVS operation at 240 Mhz, but at lower frequencies, the threshold voltage will be lower than optimum, and again there will be significant deviation from the minimum power case. In general, with a fixed threshold voltage, the minimum power point cannot be achieved because there is no way to trade-off dynamic and leakage power for a range of frequencies. It may be true that for a fixed threshold voltage, there corresponds a particular operating frequency where dynamic voltage scaling will result in minimum power consumption. But in general, if the

operating frequency changes, then the threshold voltage would be either too high or too low, and thus dynamic voltage scaling would give suboptimal results.

Figure 6-8 is derived from the models introduced in Eq 6-6 and illustrates how dynamic voltage scaling is less effective than optimal V_{CC} / V_t scaling. Depending on the fixed threshold voltage value for the DVS system and the target operating frequency, the power consumption will be different amounts over the minimum target. However, in general substantial energy savings can be achieved when V_{CC} / V_t is used instead of basic dynamic voltage scaling.

Figure 6-9 illustrates the overhead in power using DVS for the theoretical model parameters defined earlier.

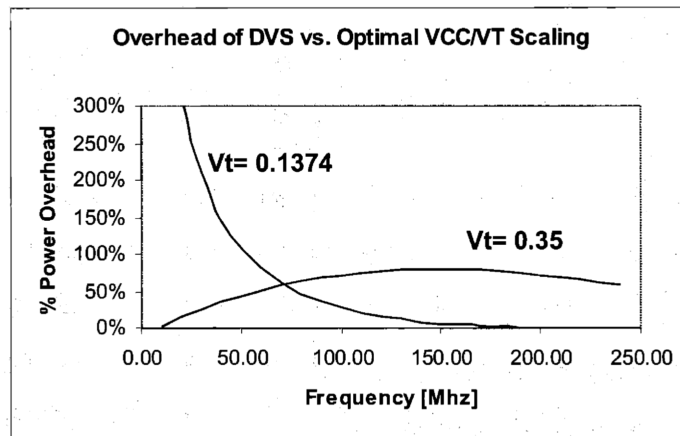


FIGURE 6-9. Percent overhead in power by using dynamic voltage scaling instead of optimal V_{CC}/V_t scaling.

The table below more clearly quantifies how much more power is consumed using a DVS approach compared to the minimum power level for a large frequency range. As can be seen, dynamic voltage scaling alone can result in significantly less energy efficiency than optimal V_{CC} / V_t scaling.

TABLE 6-2. Detailed penalty of using DVS over VCC/VT scaling

Frequency target [MHZ]	% penalty with DVS Vt=.1374	% penalty with DVS Vt=0.35	Frequency target [Mhz]	% penalty with DVS Vt=.1374	% penalty with DVS Vt=0.35
10	640.19%	3.55%	130.00	11.82%	79.33%
20.00	326.47%	15.46%	140.00	8.62%	80.06%

TABLE 6-2. Detailed penalty of using DVS over VCC/VT scaling

Frequency target [MHZ]	% penalty with DVS Vt=.1374	% penalty with DVS Vt=0.35	Frequency target [Mhz]	% penalty with DVS Vt=.1374	% penalty with DVS Vt=0.35
30.00	209.15%	26.21%	150.00	6.17%	80.16%
40.00	146.67%	35.81%	160.00	4.30%	79.56%
50.00	107.55%	44.11%	170.00	2.90%	78.34%
60.00	80.81%	51.51%	180.00	1.88%	76.59%
70.00	61.65%	58.03%	190.00	1.15%	74.38%
80.00	47.29%	63.59%	200.00	0.64%	71.80%
90.00	36.29%	68.32%	210.00	0.32%	68.94%
100.00	27.79%	72.31%	220.00	0.12%	65.86%
110.00	21.13%	75.44%	230.00	0.03%	62.65%
120.00	15.92%	77.78%	240.00	0.00%	59.38%

6.4 Triple Well Test Chip

From the mathematical models in the previous sections, it has been shown that optimal V_{CC} / V_t scaling can play a significant role in lowering overall power dissipation. Even more so, for applications where workload variations can change dynamically, significant savings in energy can be achieved with dynamic V_{CC} / V_t scaling. Modern integrated circuits do not operate close to this optimal point that is achievable through proper balance of threshold voltages and supply voltages. One of the main reasons is that threshold voltages in traditional technologies are fixed, and cannot be tuned during device processing to be optimized for a particular circuit architecture operating at a particular frequency. However, for a technology with a dynamically tunable threshold voltage, like in a triple well process, it is possible to individually adjust threshold voltages during runtime so that the optimum V_t is used depending on the local operating conditions and architecture of each circuit. Threshold voltage tuning in this regard is slightly different than the threshold voltage tuning scheme presented in the previous chapter. Earlier, the adaptive body biasing scheme was used to compensate for process variations to ensure circuits operate only as fast as necessary for a fixed supply voltage. In this chapter, the motivation for threshold voltage tuning is instead to provide the proper balance between leakage currents and dynamic power dissipation in order to minimize total energy consumption. Both techniques can be combined together though in future circuits to compensate for parameter

variations on a local scale, yet provide a global V_{CC} / V_t operating condition that minimizes the power for the chip.

A DSP test chip was fabricated in a triple well process to verify the benefits of optimal V_{CC} / V_t scaling in a modern integrated circuit. The chip consists of 16 parallel multiply-accumulate units that can be used to model the core operation of a DSP, and thus provides realistic data for how leakage power and dynamic power will interact in a real circuit. A diagram of the circuit layout is shown below. A die photo was also taken of this chip, but unfortunately the circuit was covered with layer metallization, so the relevant features are not visible.

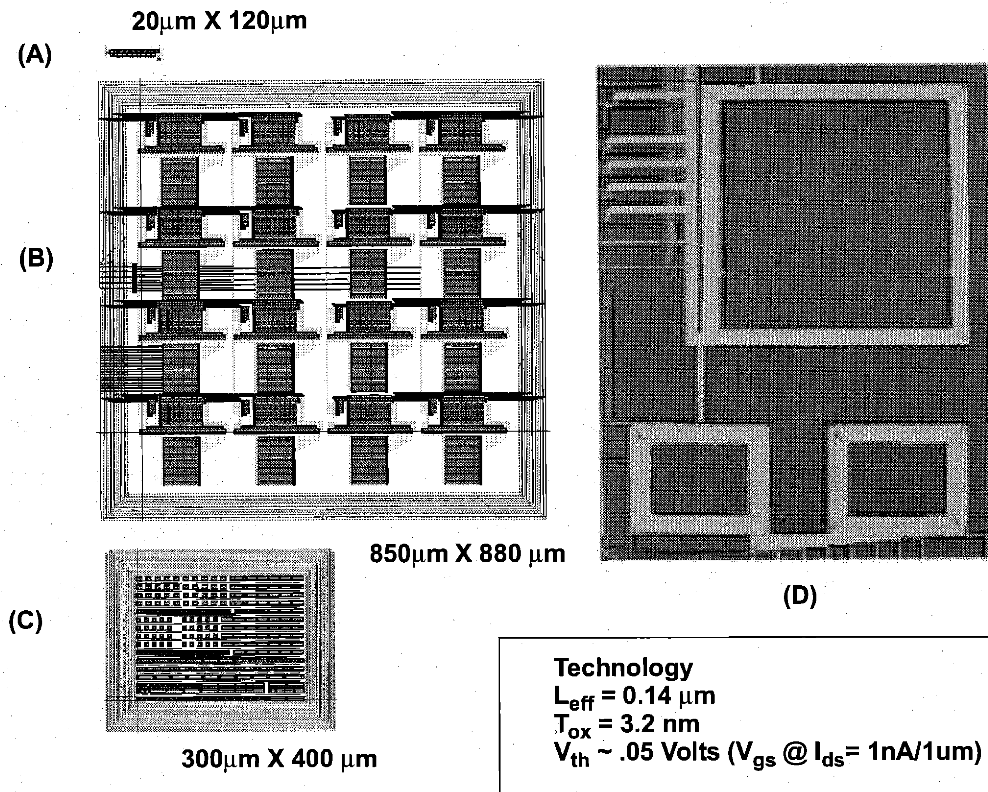


FIGURE 6-10. Layout and die photograph of the variable Vcc/Vt MAC chip. A) Ring Oscillator B) 4x4 MAC C) Adaptive body bias generator D) Die photo.

The actual packaged die was comprised of several different experiments, with the variable V_{CC} / V_t DSP test circuit being one of the circuits. Appendix A more clearly illustrates the test chip operation, schematics, pin outs, and testing setup.

6.4.1 Threshold voltage tuning limitations

The base technology used for this triple well process has a nominal V_t of 0.05V, defined as the V_{gs} corresponding to 1nA drain current for a 1um device. This roughly translates to approximately a standard linear extrapolated V_t of about 0.35V. With body biasing, this V_t can be shifted approximately down to .3V using 500mV forward bias and up to .45v with 500mV reverse bias according to spice level simulations. Unfortunately, these threshold voltages are still high compared to the optimal V_t's that are needed for a circuit operating at a reasonably fast frequency. Furthermore, from wafer level testing the actual fabricated chip had threshold voltages that were even slightly slower (55mV for the NMOS, -110mV for the PMOS @ 1nA/1um) than expected.

In order to explore circuit behavior where leakage power and dynamic power become comparable, the test chip was configured to operate at very low voltages below 1.0V and very low frequencies below 200 Mhz. Only at these very low voltages and low operating frequencies can the forward biased devices have a low enough V_t such that subthreshold leakage currents can become comparable to the dynamic power dissipation and result in a minimum power operating point.

Another limitation of this technology is that forward body bias of devices increases junction currents (junction diodes are on the verge of being turned on), which can also impact total power dissipation. As a result, the optimal V_{CC} / V_t operating point not only balances dynamic power and subthreshold leakage power components, but also junction leakage components as well. If the test chip technology had a nominally more aggressive V_t, then less forward body bias (or even reverse body bias) would be sufficient to bias the circuit into an optimal V_{CC} / V_t operating point, and forward bias currents would be less of a problem.

6.4.2 Test chip block diagram

A simplified circuit block diagram is shown in Figure 6-11 and illustrates the main components of the DSP test chip. A more accurate block diagram showing all control signals is shown in the appendix.

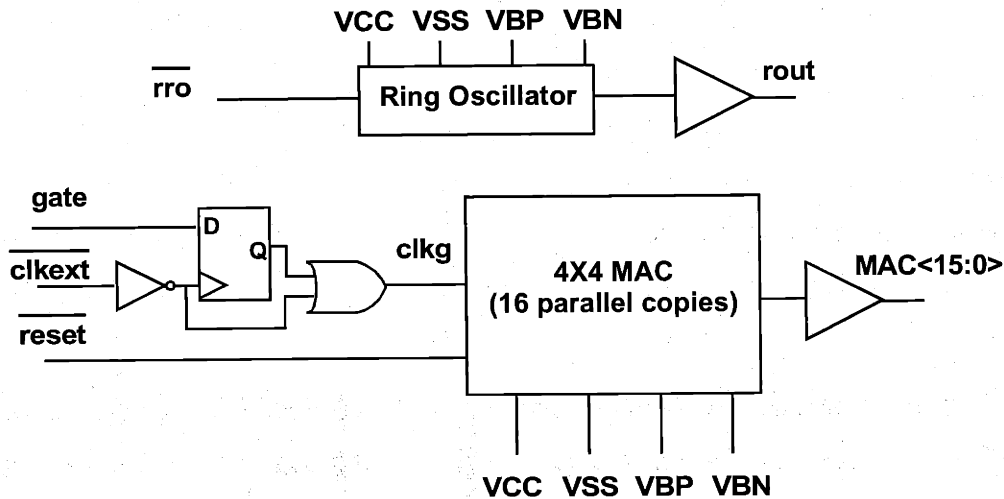


FIGURE 6-11. Test chip simplified global block diagram.

The main block of the test chip is the MAC mesh, which consists of a 4x4 array of multiply accumulate units. Within this block, the control signals are buffered before driving each internal MAC unit. By carefully equalizing buffer delays, each MAC operates in parallel. This was utilized in order to increase the power dissipation of the chip in order to simplify measurements. Also, separate power supplies are used for the MAC core circuitry and for the peripheral buffers so that the power consumed by the DSP core can be measured directly.

The individual MAC unit is shown below in Figure 6-12, and consists of an 8x8 array multiplier followed by a standard 24 bit ripple carry adder and accumulator. The data vectors are generated directly with a linear feedback shift register. The MAC circuit implementation is straightforward, but the chip is still sufficiently complex that it can be

used to model the interactions between dynamic power and leakage power as a function of operating conditions for a typical DSP.

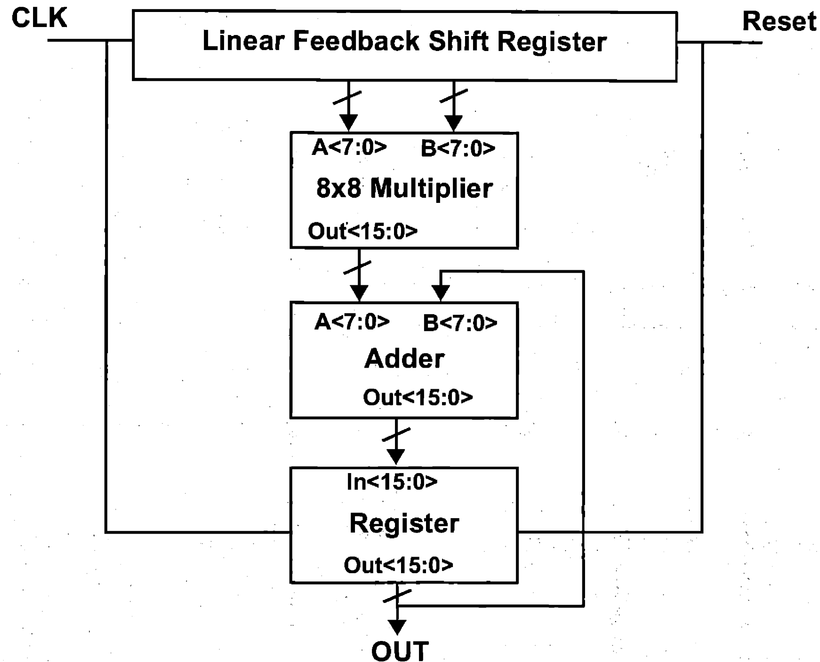


FIGURE 6-12. Block diagram of individual MAC operation.

The basic functional blocks of the MAC unit were implemented using straightforward CMOS logic gates. For example, the multiplier was implemented as a simple 8x8 array multiplier, while the adder was implemented as a simple ripple carry adder that uses standard CMOS mirror adder cells. Standard CMOS gates were used throughout the chip because the test chip was designed to operate at as low a supply voltage as possible. CMOS gates are very robust to noise and can function at extremely low supply voltages, whereas faster logic family such as dynamic or DVSL for example are much less noise tolerant and not functional at ultra low supply voltages.

The MAC inputs are driven with a linear feedback shift register that is used to automatically generate pseudorandom test vectors in the MAC. The output of the MAC are buffered through inverter chains that directly drive the output pins and the core MAC circuitry has externally biased PMOS and NMOS wells that can be separately controlled to tune threshold voltages.

Because of the relatively slow operation, and low supply voltages though, one concern for the test chip was the effectiveness of having the MAC buffers directly drive the package pin outputs. In order to circumvent this potential problem, the MAC was designed so that it could be stalled by asserting a clock gating signal after a fixed number of cycles are completed so that the MAC final output is held constant. Even if the output capacitance is very large, the output pins could be driven to V_{CC} very slowly. It turns out though that because the operating frequency was slow enough and the pin capacitances small enough that the output pin rise/fall times were relatively short compared to the clock periods so that the MAC outputs could be driven satisfactorily.

6.4.3 V_{CC}/V_t biasing procedure

The DSP testchip is a useful vehicle for quantifying the benefits of V_{CC} / V_t scaling, and illustrates how leakage currents and dynamic power can be balanced in a modern circuit. For testing purposes, the optimal V_{CC} / V_t operating points were manually generated off chip through separate power supplies that control V_{CC} as well as the NMOS and PMOS body biases (V_{BN} and V_{BP}). This manual approach was a simple way to quantify the benefits of optimal V_{CC} / V_t scaling for different operating frequencies.

Although not shown in the above block diagram, another version of the circuit (on the same test chip) was designed with an adaptive body bias generator similar to that of the previous chapter to automatically tune device well voltages to meet a fixed target fre-

quency. This adaptive body bias generator was based on work done by [44], and was supplied by Hitachi for this joint test chip.

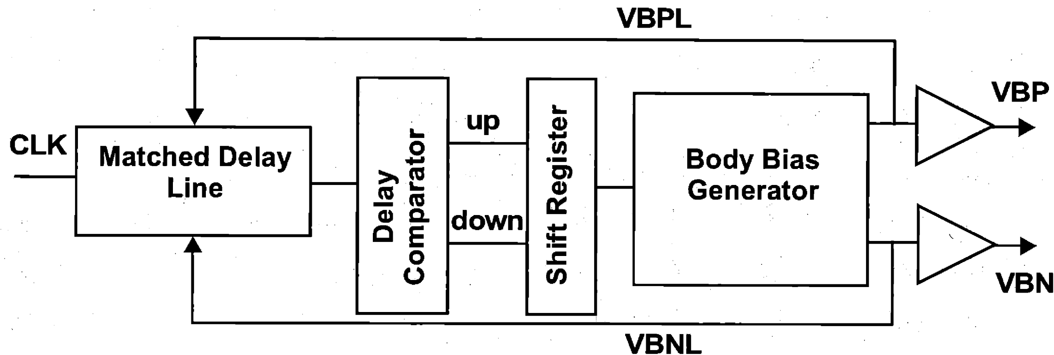


FIGURE 6-13. Adaptive body bias generator version courtesy of Hitachi [Miyazaki].

Unfortunately, time limitations prevented us from performing measurements and testing using this adaptive body generator, but instead measurements were taken using external V_{CC}, V_{BP} and V_{BN} tuning. However, future experiments can combine this adaptive body bias loop with a variable supply voltage converter to illustrate automatic V_{CC}/V_t tuning.

6.4.4 LFSR for automatic vector generation

As illustrated in Figure 6-12, a linear feedback shift register was used to generate test vectors for each MAC in the test chip. This technique is useful because it simplifies the testing procedure because test vectors are automatically generated on chip. In order to verify the operation of the chip, one simply needs to reset the LFSR, and let the MAC compute for a fixed number of cycles. The final output value of the MAC can then be compared to the predicted target to determine whether or not the MAC is fully functional. The LFSR output sequence is fixed and can easily be predicted, yet the signal has a white spectrum and can provide realistic input vectors. If a length N LFSR is implemented correctly, then it can be made to only repeat after 2^N cycles, which ensures that all possible vector sequences can be exercised. However, if the length is too long, then it may be impractical to sequence all possible combinations, although a very good random sampling could still be achieved. A great deal of research has been proposed using LFSRs in built in self test

(BIST) structures, and more advanced techniques can be used to provide better fault coverage[61].

The MAC serves to integrate previous output samples because of the accumulator function. As a result, any functionality error that occurs in earlier cycles would likely be propagated to the later accumulator output results. Thus, by merely comparing the very last sample of the MAC to the theoretical result, one can conclude whether or not the circuit was completely functional with a very high degree of probability.

6.4.5 Critical path replica

A critical path replica was used to characterize the target operating frequency of the MAC circuitry. This critical path was made into a ring oscillator that could be measured off chip and still track the MAC operating frequency (which is twice the ring oscillator frequency) for varying supply voltages and body bias settings. A separate critical path was also used to implement the matched delay line for the adaptive body biasing loop that automatically sets the N and P well biases until the target frequency is reached.

The critical path replica gives a very good representation of the worst case delay through the MAC, and thus can be used to measure the V_{CC} / V_t locus that defines each iso-performance curve. Because it is important for the critical path replica to match as closely as possible the actual core circuit delay, the critical path was implemented using the identical gates and identical loadings of the worst case critical path in the MAC. For example, the critical path replica consisted of complex gates with series connected devices and representative gate loadings found in the core circuitry. Each gate was configured to behave like an "inverter" that has the worst case switching characteristics of the gate. This was accomplished by ensuring that the switching paths occur through worst case series paths and through the minimum number of parallel chains. As a result, each gate in the critical path replica switches with worst case dynamics, and the total delay represents the worst case critical path for the MAC circuitry. By using the exact same circuit structures found in the chip core for the critical path replica, it is very straightforward to extract a worst case critical path. Because the gate sizings and loading effects are replicated, the

critical path can be made to very accurately track the performance of the whole circuit over different supply and parameter variations.

In some previous work[38], critical paths of a circuit have been modeled by a simple chain of basic inverters rather than with complex gates configured for worst case delays. Theoretically, the relative tracking of all CMOS gates in a circuit tend to track approximately the same as any other circuit. For example adder, ALU, multiplier, or even microprocessor speeds all tend to track one another in the same fashion. If the delays for these circuits are normalized to that of the delay of a basic inverter chain (with the same operating conditions), the resultant should be approximately constant. Although this is a useful abstraction for understanding the general impact of scaling on circuit behavior, it is still only an approximation. What is required for the critical path replica of the test chip though is for it to match and track the actual circuit as accurately as possible without incurring any scaling or matching errors. Figure 6-14 illustrates how a standard inverter chain can somewhat track the behavior of the critical path replica using complex gates, although it can still deviate significantly as parameters change.

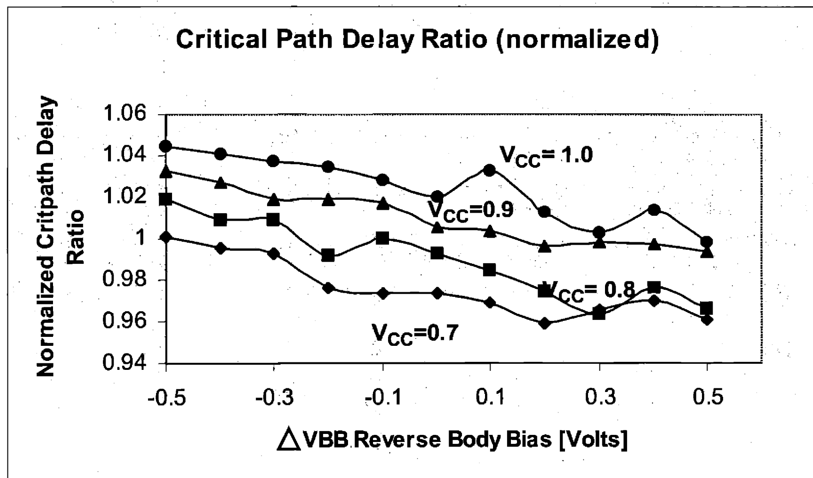


FIGURE 6-14. Normalized ratio between delay of complex critical path and delay of 10 sequential inverters.

The plot shows simulations that illustrate how the normalized ratio between the critical path replica delay (using complex gates) and an inverter chain (consisting of 10 fanout of three gates) will vary with changes in body bias and supply voltage. For differ-

ent V_{CC} and different ΔV_{BB} values (positive ΔV_{BB} values are assumed to be reverse biased, and negative ΔV_{BB} values are assumed to be forward biased for both PMOS and NMOS devices), ratios between the delays of the two different circuits were simulated and then the ratios themselves normalized. Ideally, if both CMOS circuits track the same for all bias conditions, then lines should all be constant and equal to unity. Clearly though significant deviations can arise. For even lower supply voltages (not shown above), the mismatches are even larger with deviations > 15%. This illustrates how second order effects like stacked devices, mismatches in drive, varying loading conditions can all effect how circuits scale with supply and body bias voltages. Thus, the best way to accurately model the worst case delay of a circuit is to replicate the critical path using as much of the same circuit structure as possible. By doing so, not only does the critical path replica accurately track the performance of the actual circuit, but the replica can be designed in a straightforward manner without the need for a scaling factor or equalization step to provide matching normalization as would be necessary for a standard inverter chain model.

6.5 Test Chip Simulations

Simulations were performed on the test chip to explore the benefits of V_{CC} / V_t scaling using power supply and body bias scaling. The optimal power point thus corresponds to an appropriate choice of V_{CC} and ΔV_{BB} (which implicitly sets the V_t). Because the nominal threshold voltage of this technology is relatively high, the optimal threshold voltage could only be achieved by lowering V_t's by applying forward body bias. As described before, the maximum forward bias would be on the order of 500mV so that the junction diodes are not turned on.

6.5.1 Isoperformance V_{CC}-ΔV_{BB} simulation locus

In order to reach an equilibrium point between dynamic power and leakage power, it was necessary to simulate the test chip with very low supply voltages and with very low operating frequencies. Only under these conditions is it possible to tune threshold voltages low enough so that the optimal V_{CC} / V_t bias point could be reached. Figure 6-15 shows

simulated iso-performance curves of V_{CC} versus ΔV_{BB} (similar to those of Figure 6-4 that used mathematical models) for the test chip.

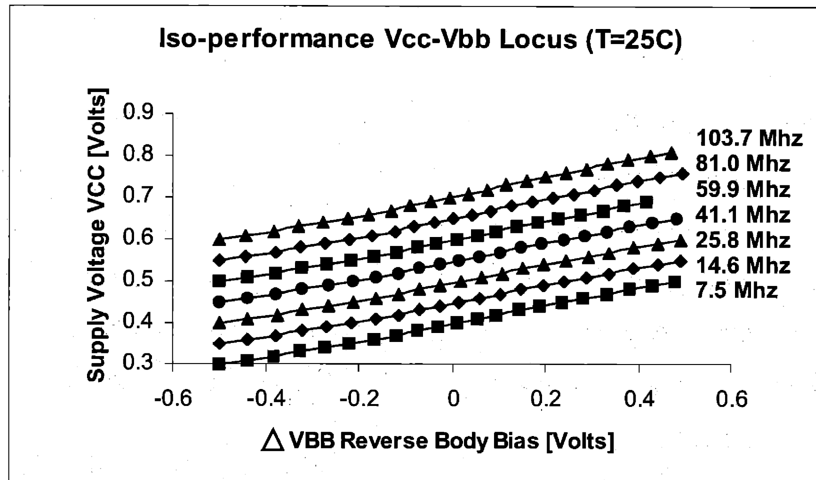


FIGURE 6-15. V_{CC}-V_{BB} **locus for fixed target frequencies.**

For each fixed operating frequency, the V_{CC}-ΔV_{BB} locus was derived from simulations performed on the critical path replica. Once these locus curves were derived, then the entire MAC could be biased for each operating frequency and simulated for total power consumption. Figure 6-16 below shows more iso-performance curves for supply and body bias locus pairs that correspond to a fixed frequency. However, this graph shows how temperature and process variations can impact these locus lines of constant performance.

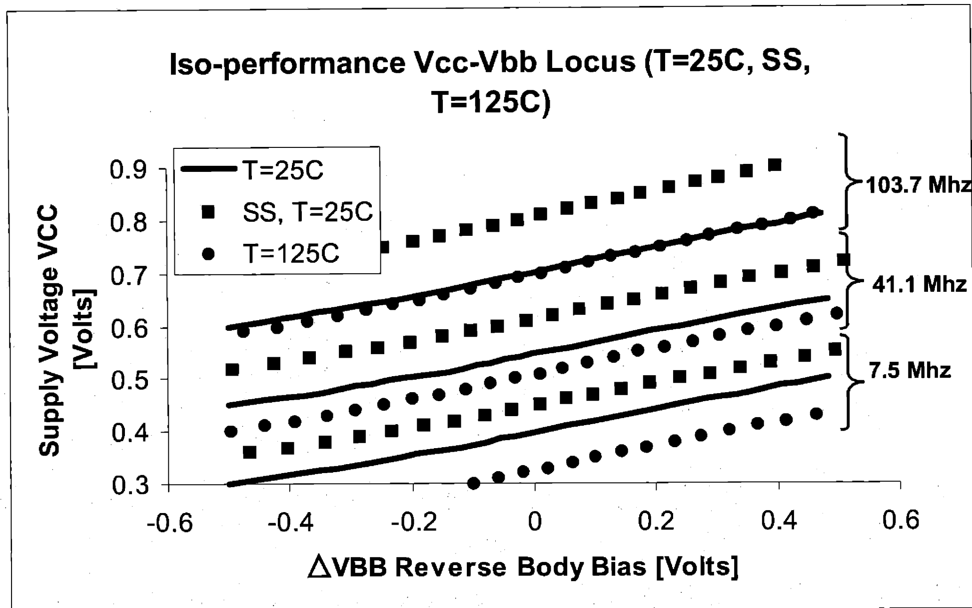


FIGURE 6-16. Temperature and process variation impact on V_{CC}-ΔV_{BB} locus for fixed target frequencies (simulations).

For example, for a circuit that corresponds to a slow process corner (SS corner shown above), for the same frequency, the V_{CC}-ΔV_{BB} curve shifts higher. This makes sense because higher supply voltages or more forward body bias is necessary to speed up slow devices to meet the target specification. However, the temperature impact on the V_{CC} / ΔV_{BB} locus is slightly different. It turns out that for very low supply voltages, higher temperatures can actually improve performance.

When temperature increases, there are two effects on circuit performance. First, threshold voltages drop, which causes devices to speed up, but at the same time mobility degrades, which causes devices to slow down. This conflicting behavior thus causes circuits to speed up with temperature at very low voltages, but slow down with temperature at higher voltages. This can be seen in the behavior of Figure 6-16, where at low V_{CC} values, the T=125 curve is actually below the T=25 curves. In otherwords, for a fixed ΔV_{BB}, performance can be maintained with a lower supply voltage at higher temperature than at lower temperature. On the otherhand for higher V_{CC} values, the T=125 curve is closer to

the T=25 curve. At even higher frequencies, the T=125 curve would actually be higher than the T=25 curve, indicating that the temperature impact becomes detrimental to circuit performance.

6.5.2 Power vs. V_{CC} simulations for different frequencies

The iso-performance curves from Figure 6-15 and Figure 6-16 were simulated using the critical path replica. Simulations were next performed on the entire MAC circuitry using the V_{CC}-ΔV_{BB} biasing locus points that define a fixed operating frequency. The optimal V_{CC}-ΔV_{BB} point then corresponds to the point on this locus that gives the lowest overall power. Results for several different frequencies are shown below in Figure 6-17. These power versus V_{CC} curves are consistent with the mathematical models described earlier where higher frequencies have operating conditions with higher supply voltages and lower threshold voltages. As operating frequency changes, the optimal V_{CC}-ΔV_{BB} operating point changes significantly as well.

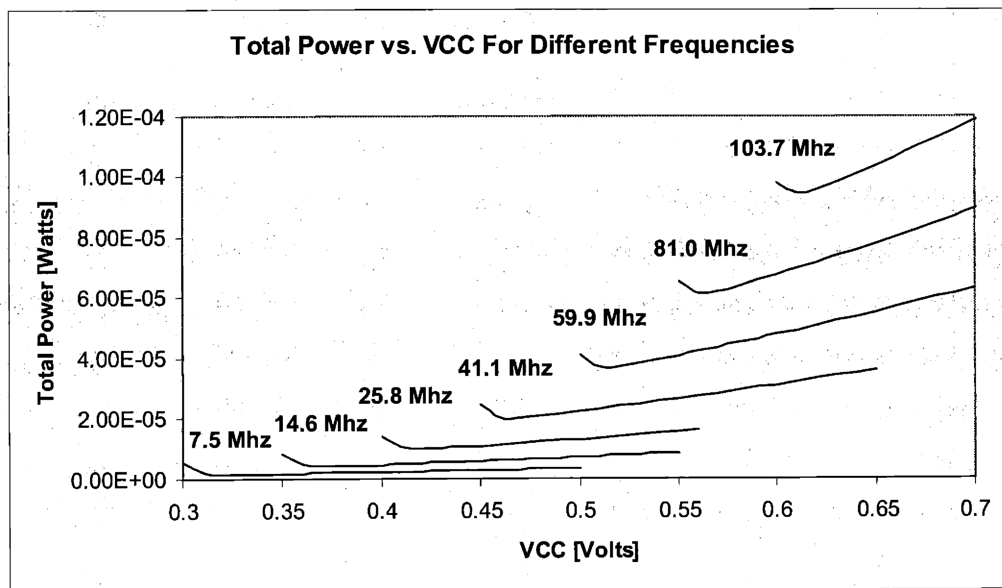


FIGURE 6-17. Power as a function of supply voltage for different operating frequencies (simulations).

Figure 6-18 shows additional simulations performed on the MAC circuit block to illustrate how temperature and process variations can greatly effect optimal V_{CC}-ΔV_{BB} operating points. In addition to the optimal power vs. V_{CC} curves for nominal conditions, curves are also shown for a high temperature condition (T=125C), and also for a very slow process corner (SS corner corresponding to 3 sigma deviation).

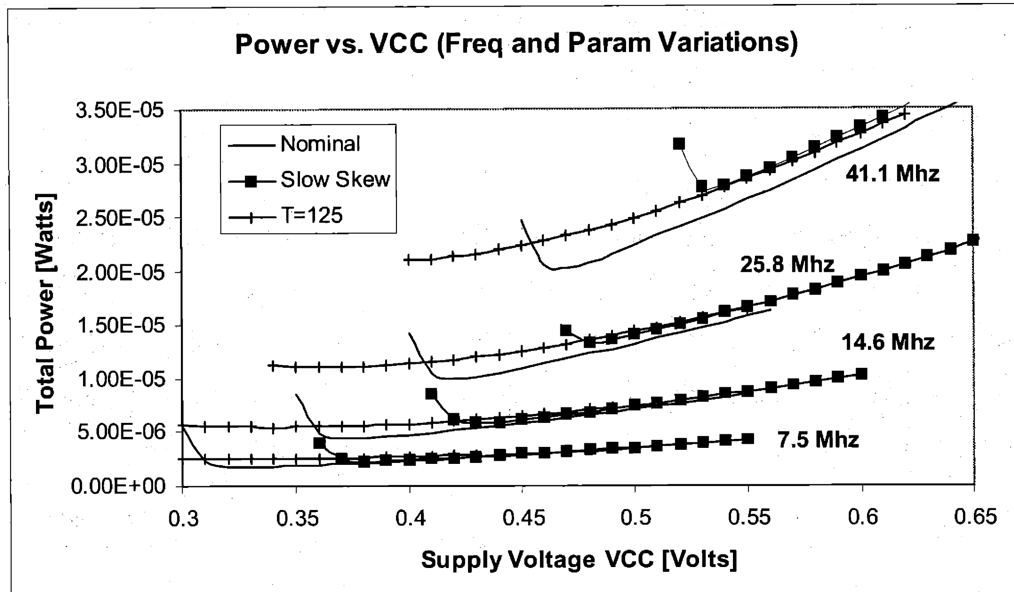


FIGURE 6-18. Power vs. V_{CC} for different operating frequencies for nominal, slow process skew, and high temperature conditions (simulations).

Theoretically, if the parameter variations that impact a circuit only effect device threshold voltages, and the effect of body biasing is to only shift device threshold voltages, then the optimal power-V_{CC} curve should remain identical in all cases regardless of the variation. This is because the applied body bias voltage (implicitly a function of V_{CC}) would exactly compensate out any V_t shift, and the optimal operating point would still correspond to the same V_{CC} / V_t ratio, with the ΔV_{BB} body bias shifted accordingly. Realistically though, parameter and circuit variations will effect more than just the threshold voltage, and device body biasing will do more than just shift threshold voltages. For example, in the forward bias state, junction leakage currents play a larger role in total power consumption, and process and temperature variations can effect other important parameters such as mobility, subthreshold slope, leakage constant I_0, DIBL etc. As a

result, the optimal V_{CC} / V_t point can shift depending on circuit conditions. With process and temperature variations, the dynamic power curve stays basically constant (because it is only dependent on V_{CC}, frequency and effective switched capacitance), but the leakage power curves can be impacted greatly and shift the optimal power point higher or lower. For high V_{CC} values, the curves for varying temperature and process parameters all tend to meet asymptotically. This reflects how the total power in these regimes are dominated by the dynamic power components, which are constant for a fixed frequency.

6.5.3 Process corner variation impact on minimum power point

For the slow process corner, one can see that the optimal operating point corresponds to a significantly different V_{CC} value. The SS corner models significant V_t slow downs, but also models effects such as worst case supply voltages, transistor critical length variations, etc. Thus compared to the typical case, for a fixed frequency and fixed V_{CC} level, the SS corner would correspond to ΔV_{BB} body bias amounts significantly greater (more forward bias direction) than required for the typical case. These large forward bias amounts are necessary to shift the threshold voltages faster not only to compensate for the slower process V_t's of the devices, but also to compensate for other parameters that tend to slow down the performance. Furthermore, these extra amounts of forward bias would result in more junction leakage currents for the locus point for a given fixed frequency. All these effects thus tend to push the leakage power to higher levels, which shifts the optimal operating point towards higher V_{CC} and higher overall power compared to the typical case. Although the SS process corner corresponds to a very extreme case of maximum variations, it still illustrates how the optimal operating point can shift. For more reasonable variations for circuits, the optimal V_{CC} operating point would likely not shift too greatly, and thus result in reasonably low power for a variety of process variations.

6.5.4 Temperature impact on minimum power point

For the high temperature case, the optimal V_{CC}-V_t operating has also been shown to vary significantly. However, one interesting effect from these simulation models is that the optimal V_{CC} operating point tends to be shifted towards lower supply voltages (but at slightly greater power) for this particular technology. This is somewhat unintuitive

because from a theoretical point of view, the effect of high temperature ought to increase subthreshold leakage power and shift optimal V_{CC} operating points to the right. One troublesome feature of the curves in Figure 6-18 is that the constant performance curve of power vs. V_{CC} for the T=25 case actually intersects the curve for the T=125 case. Theoretically, from a subthreshold leakage point of view this should not happen. For example, consider a fixed V_{CC} value where the T=25 curve power is higher than the T=125 curve. For the nominal case an appropriate forward bias amount ΔV_{BBnom} is applied to meet the desired performance level. When the temperature goes up, the threshold voltage drops, and the subthreshold slope worsens. For low supply voltages, the current drive for the T=125 case is higher than nominal, so the circuit actually can operate faster than necessary. As a result, the applied body bias ΔV_{BBnom} can be increased (towards reverse body bias direction) so that the threshold voltage is raised and the critical path speed is reduced. However, because the subthreshold slope is larger in the case for T=125, even if the threshold voltages is increased to equate current drives, the subthreshold leakage current should still be higher than for the T=25 nominal condition. As a result, even though the body bias value increases (more reverse bias) in the T=125 case to increase threshold voltages, theoretically the subthreshold leakage currents should still be larger than the case for the T=25 operating condition. Thus, the power-V_{CC} curve for the T=25 case should never exceed the curve for the T=125 case.

However, the simulation models show otherwise. The reason for this deviation in simulation result from predicted circuit behavior is that at large forward body bias levels, junction currents can start to play an increasingly large part of overall power dissipation. For example, for a fixed V_{CC} and fixed performance goal, the amount of forward body bias needed for the T=25 case is greater than the amount of forward body bias needed for the T=125 case (because the threshold voltages are decreased as temperature increases). As a result, although the subthreshold leakage current in the T=125 case might still be theoretically larger, the diode junction leakage currents for the T=25 case might become more dominant. This explains the sharp increase in total power consumption on the left hand side of the T=25 power - V_{CC} curves, and why the power for a fixed V_{CC} (but more forward bias) bias condition at T=25 can actually turn out to be greater than for the same fixed V_{CC} (but smaller amount of forward bias) bias condition at T=125.

6.5.5 Optimal V_{CC}/V_T scaling comparison with DVS

Figure 6-19 finally shows simulation comparisons between the effectiveness of dynamic voltage scaling versus optimal V_{CC}/V_t scaling through body biasing in a triple well technology. Simulation were performed for a nominal process corner at T=25.

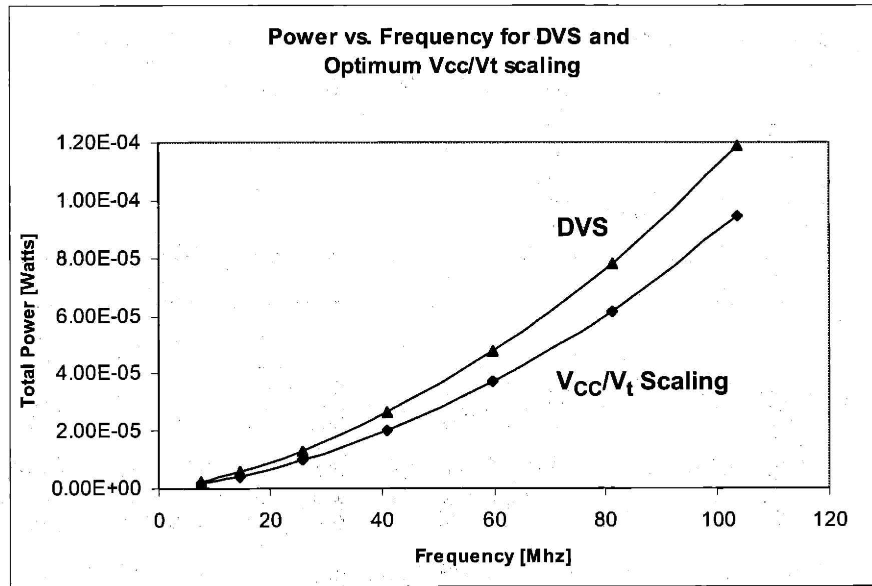


FIGURE 6-19. Simulated power vs. frequency for dynamic voltage scaling versus optimal V_{CC}/V_t scaling.

Table 6-3 below shows in more detail how DVS is less energy efficient than optimal V_{CC} / V_t scaling at several different operating frequencies. The numbers are strong functions of the technology used, target frequencies, and circuit under tests, but nonetheless illustrate that significant energy savings can be achieved by using an optimal V_{CC} / V_t scaling technique.

TABLE 6-3. Simulation results of penalty of using DVS over VCC/VT scaling

Frequency [MHZ]	Vcc/Vt Scaling Power [Watts]	DVS Scaling Power [Watts]	% overhead due to DVS Scaling
7.5	1.74e-6	2.32e06	32.9%
14.6	4.39e-6	5.85e-6	33.3%
25.8	9.93e-6	1.32e-5	33.1%
41.1	2.03e-5	2.65e-5	30.9%
59.9	3.7e-5	4.76e-5	28.5%

TABLE 6-3. Simulation results of penalty of using DVS over VCC/VT scaling

Frequency [MHZ]	Vcc/Vt Scaling Power [Watts]	DVS Scaling Power [Watts]	% overhead due to DVS Scaling
81.0	6.13e-6	7.78e-5	26.9%
103.7	9.45e-5	1.19e-4	25.6%

The dynamic voltage scaling simulations assumed a nominal V_t value corresponding to zero ΔV_{BB} case where no body biasing is applied. As can be seen, significant power savings can be achieved by appropriately tuning threshold voltages as well as supply voltages depending on the target operating frequency. For different operating conditions and process variations, the dynamic voltage scaling can exhibit more or less degradation for given target frequencies. Because the threshold voltage in a DVS scheme are not tunable, over a range of operating frequencies one would always expect an optimal V_{CC} / V_t scaling approach to be more energy efficient.

6.6 Test Chip Measurements

The MAC test chip was actually fabricated in a 0.14μm triple well Hitachi process, and actual measurements were taken to verify the principle of V_{CC} / V_t scaling for optimal power efficiency. In the lab, the chip was measured to operate as low as 0.175 volts, with the ring oscillator functioning as low as 0.1 volts with zero body bias applied. The ring oscillator was constructed using exact replicas of the MAC critical path circuitry, so theoretically both should track exactly. For voltages greater than 0.175V, the ring oscillator does accurately track the speed of the MAC (so that the MAC is fully functional at the critical path clock speed). However, the MAC could not operate quite as low a supply voltage as the ring oscillator, most likely due to the fact that the chip flip flop circuits failed at these low voltages. Nonetheless, a functional DSP operating at 0.175V was shown, which shows that very low voltage circuit operation can be achieved when static CMOS design approaches are used. At these very low voltages, the circuits are operating virtually in the subthreshold regimes since V_{CC} is comparable or even less than the device threshold voltages.

Figure 6-20 shows a scope waveform corresponding to the complex ring oscillator operating at 0.1V with the power supply shown as well.

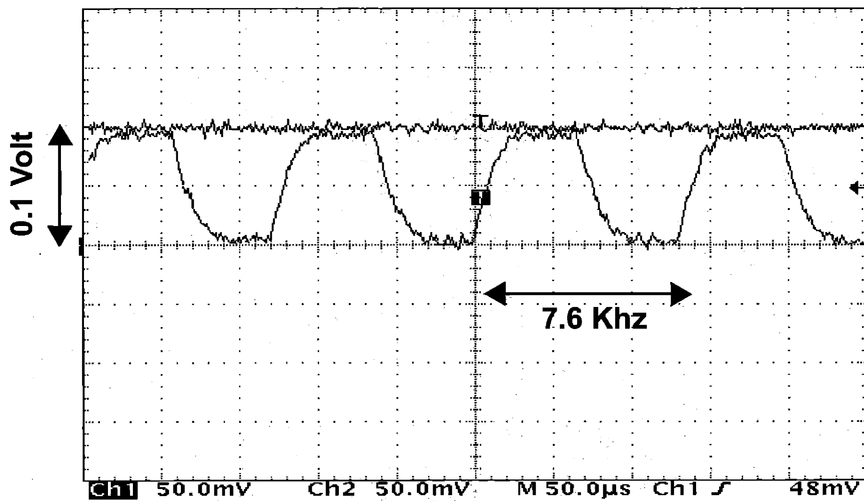


FIGURE 6-20. Scope trace of critical path ring oscillator replica showing functionality at 0.1V operation

Figure 6-21 shows the functioning DSP MAC with the input clock and a representative output pin.

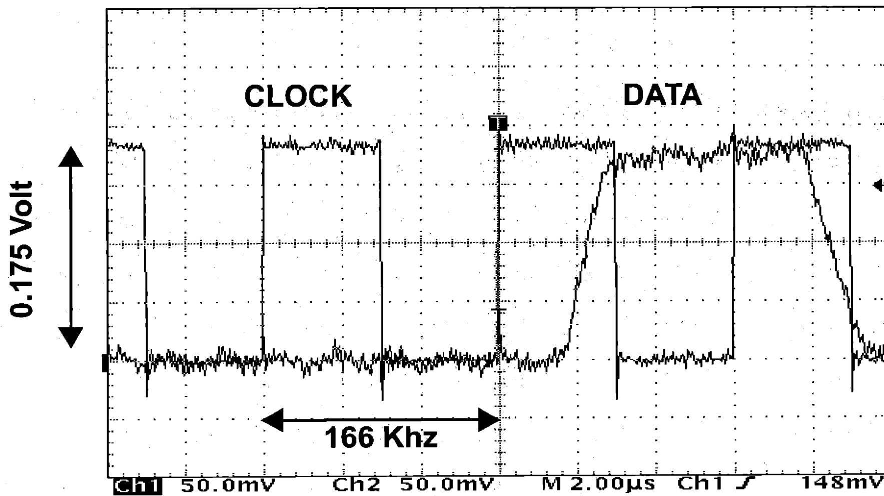


FIGURE 6-21. Scope trace showing DSP chip operation at 0.175V.

Measurements were also taken for the critical path ring oscillator frequency as a function of applied ΔV_{BB} for different V_{CC} values as illustrated below in Figure 6-22. The body bias amounts were applied equally to the PMOS and NMOS devices, where a bias of ΔV_{BB} corresponds to an NMOS well bias of 0V - ΔV_{BB}, and the PMOS well bias corresponding to $V_{CC} + \Delta V_{BB}$. As such, a positive ΔV_{BB} corresponds to reverse body biasing, while a negative ΔV_{BB} corresponds to forward body biasing.

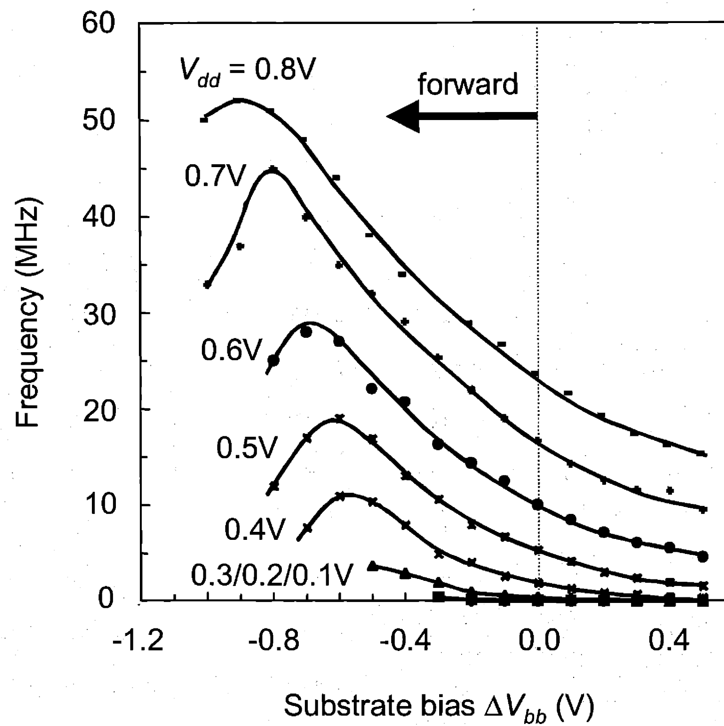


FIGURE 6-22. Critical path ring oscillator measurements of frequency versus forward and reverse body bias [Miyazaki].

The above measurements from the ring oscillator can easily be transformed into V_{CC} / ΔV_{BB} locus curves that define fixed target frequencies by plotting V_{CC} versus ΔV_{BB} for specified frequency targets. The critical path frequencies would correspond to twice the measured ring oscillator frequencies because the ring oscillator period consists of two circulations through the critical path. The curves from Figure 6-22 are themselves useful though, and show intuitively the impact of body biasing on circuit performance. As more forward body bias is applied, devices are sped up and frequency improves. How-

ever, for large amounts of forward body bias, the frequency actually peaks and then begins to roll off. This strange behavior is actually due to the limitations of using forward body bias for this technology at low supply voltages. The roll off feature occurs because of a parasitic PN diode between the PMOS and NMOS wells themselves. This diode can turn on if enough forward bias is applied, and worsens when the operating voltage drops. Another impact of forward body biasing is that junction currents can increase when junction-well diodes become forward biased as well. These forward bias current components are illustrated in Figure 6-23.

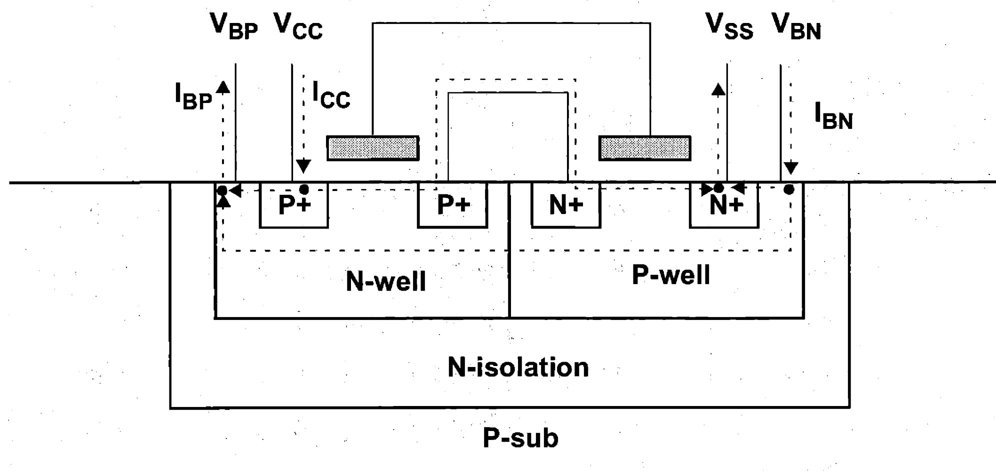


FIGURE 6-23. Forward bias currents that can arise in a Hitachi triple well process (shown for an inverter).

With a forward body bias amount of $|\Delta V_{BB}|$, the V_{BN} bias goes to a voltage of $|\Delta V_{BB}|$ while the V_{BP} bias goes to a voltage of $V_{CC} - |\Delta V_{BB}|$. As a result, the Pwell-to-N+ junctions and the Nwell-to-P+ junctions become forward biased and will start to conduct large amounts of current if this $|\Delta V_{BB}|$ voltage exceeds the PN junction built in potential. As long as the forward bias voltage is less than this critical amount (500mV for example) these junction currents should be relatively small. On the otherhand, the voltage across the P-well and N-well diode is a strong function of V_{CC} and can easily be forward biased if the supply voltage is low enough. For example, for a forward bias amount of $|\Delta V_{BB}|$, the voltage across the Pwell-Nwell junction will be $2|\Delta V_{BB}| - V_{CC}$. For example, if V_{CC} is 0.3V, and $|\Delta V_{BB}|$ is 0.5V, then the Pwell-Nwell junction bias will be 0.7V forward

biased, which shows that at low V_{CC} values, the well junctions can easily be forward biased. When the Pwell and Nwell junctions are forward biased, the circuit behavior is degraded because currents are actually flowing from the PMOS and NMOS body wells, so the devices do not operate correctly.

To the first order, if the forward bias source and drain junction currents for the PMOS and NMOS are comparable, then the currents of I_{BN} and I_{BP} of Figure 6-23 are approximately the same. The figure below shows measurements of I_{DD} and I_{BB} currents as a function of the applied body bias. The graph shows how the Pwell-Nwell body currents are not only a strong function of the applied forward body bias amounts but also on the supply voltage as well.

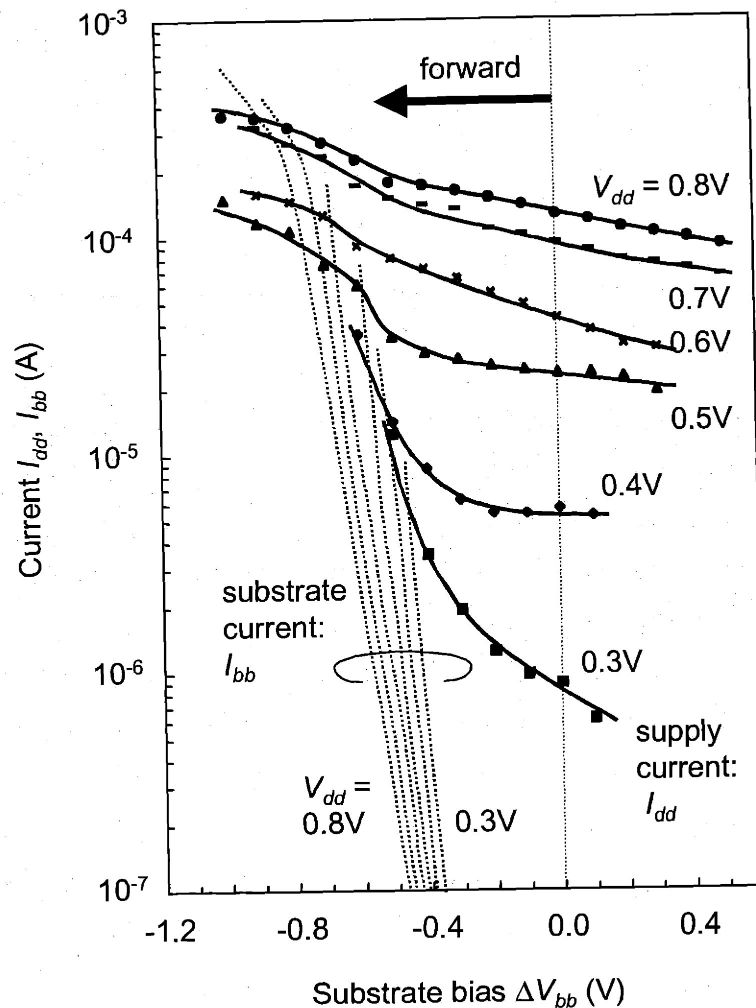


FIGURE 6-24. Supply and body current measurements for the test chip [Miyazaki]

Figure 6-24 supports the assumption that at low voltages, forward body bias causes large diode leakage currents to develop between the actual PMOS and NMOS wells. For example, for a fixed amount of applied body bias, ΔV_{BB}, the amount of substrate current associated with a low V_{CC} value is much higher than that associated with a high V_{CC} value. If it were true that the substrate currents are dominated by the highly doped source junction currents for forward bias conditions, then there would be a very small dependency on supply voltage. This is because the voltage across the junction PN diodes are fixed by the applied ΔV_{BB} regardless of the supply voltage. On the otherhand, the voltage across the Pwell and Nwell diode, is a strong function of the supply voltage.

As the body currents increase exponentially, they can become comparable to the supply currents. At these limits, the performance of the circuit are degraded as illustrated in Figure 6-22, and further application of forward body bias will only worsen circuit performance. The problem of forward biased Pwell and Nwell junctions arise only at low supply voltages. At higher supplies, V_{CC} is large enough that the well voltage differential (given by $2|\Delta V_{BB}| - V_{CC}$) does not become excessively forward biased. However, at low supply voltages only small amounts of forward bias are tolerable, and the amount of maximum forward bias varies with the actual V_{CC}. There are several ways to avoid this problem in the future though. A modified technology can be used that isolates the P and N wells, or the intrinsic device threshold voltages can be processed to be low enough that large forward bias tuning ranges are not required.

6.6.1 Measured optimal V_{CC} / ΔV_{BB} operating points

Measurements were next taken for total power consumption of the DSP as a function of different operating frequencies and different V_{CC} / ΔV_{BB} combinations. Theoretically we were expecting to find that the optimal power point corresponded to a trade-off between dynamic switching power and subthreshold leakage currents. What we found though was that when forward bias was used to lower threshold voltages, junction diode currents and Pwell - Nwell diode currents became dominant. This exponential increase in currents due to forward biased diode connections resulted in large current spikes that caused the power curve to increase dramatically after a certain level of forward biasing was applied.

Although subthreshold leakage currents also increased as threshold voltages were reduced, forward bias currents increased even more rapidly, and cause the power versus V_{CC} curve to suddenly shoot up as V_{CC} is lowered and more forward bias is applied. As a result, there is still an optimal V_{CC}-ΔV_{BB} operating point for this DSP that minimizes total power consumption, but the mechanism is slightly different than theoretically predicted.

Figure 6-25 shows the power versus V_{CC} curves for several different target frequencies, where the ΔV_{BB} amount is implicitly chosen to satisfy the target frequency. As can be seen, there is a definite operating point that minimizes the total power dissipation, but at lower V_{CC} values, the power shoots up because of the forward bias currents. In fact, there is a hard limit where the circuit will not function below a minimum V_{CC}.

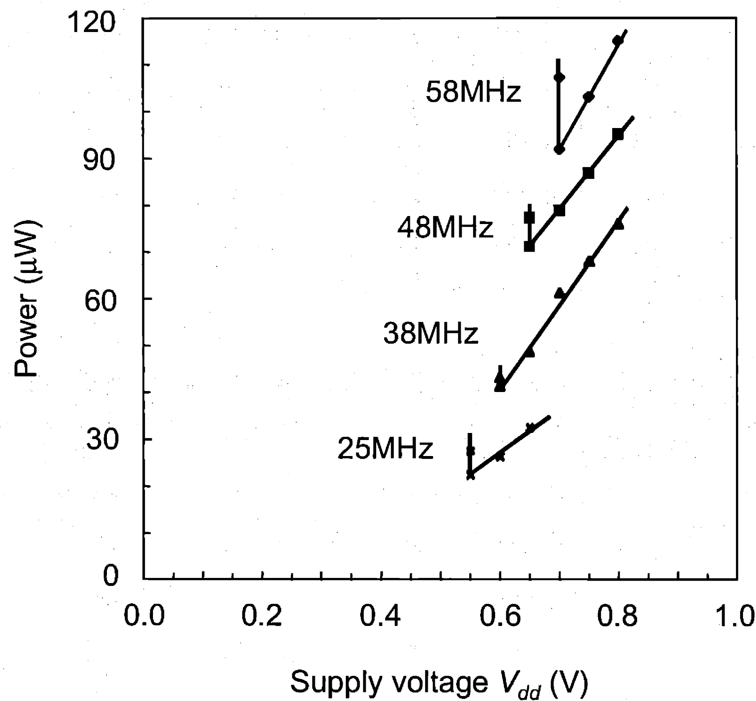


FIGURE 6-25. Power versus supply voltage measurements for varying frequencies (forward bias current limited) [Miyazaki].

Conversely, Figure 6-26 shows the power curves plotted against the applied body bias ΔV_{BB} instead of the supply voltage. In this case, the supply voltage is implicitly set by the choice of ΔV_{BB} needed to satisfy the target frequency. As can be seen, the local

minimum in power for lower operating frequencies shifts to the right in the power vs. ΔV_{BB} curve. This is because the Pwell and Nwell can become forward biased more easily at lower V_{CC} operating levels, and also because at low frequencies, the dynamic power components are reduced and the optimal point is shifted towards reducing leakage currents.

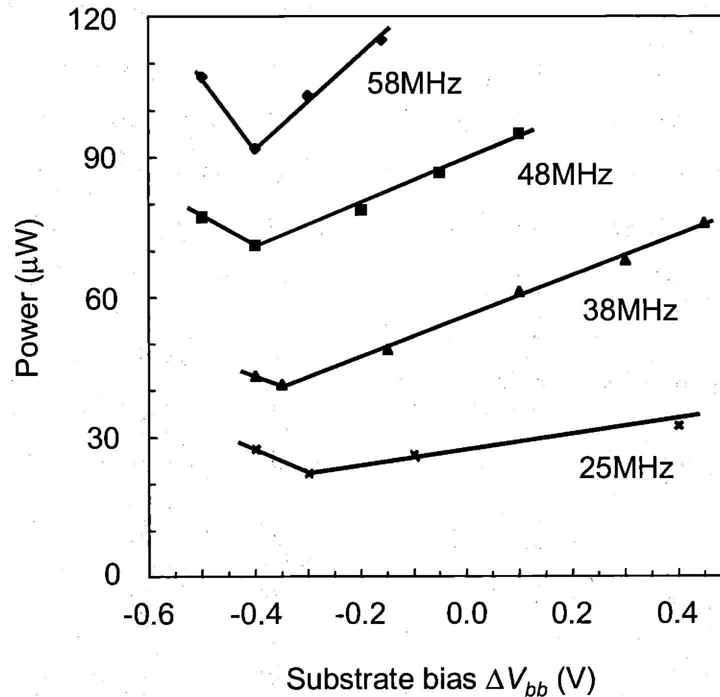


FIGURE 6-26. Power versus threshold voltage measurements for varying frequencies (forward bias current limited) [Miyazaki].

The power measurements for the DSP test chip showed that the optimal V_{CC}-V_t operating was limited by forward bias currents (especially the parasitic Pwell and Nwell diode that can easily be forward biased at low supply voltages). This result, though interesting, is not really a fundamental limit for optimal V_{CC} / V_t scaling. Rather, it is a consequence of the fact that the nominal threshold voltage of the process was actually too high to be shifted adequately with forward body bias tuning. In effect, limitations in forward body bias effectiveness (especially at low V_{CC} values) prevented us from lowering the threshold voltages enough to improve device speeds and to trade-off subthreshold leakage power for dynamic switching power. Initially, the goal of this test chip was to operate at

very low operating voltages and very low frequencies in order to bias the power consumption more towards subthreshold leakage currents versus dynamic power, but this turned out not to be sufficient. The low voltage operation turned out to worsen the forward bias currents due to the Pwell and Nwell becoming more easily forward biased at low V_{CC} values.

The test chip thus showed that the technology used still has a nominal threshold voltage that is much higher than optimum. As technologies continue to scale though, threshold voltages will also scale to the point where V_t's will become low enough that body biasing techniques can be used to effectively balance dynamic power dissipation with subthreshold leakage currents as theorized earlier. For current technologies though, power savings can still be achieved by lowering threshold voltages as much as possible with forward body biasing. Figure 6-27 below shows power versus frequency curves for the case where both V_{CC} and ΔV_{BB} are optimized and for the case of standard voltage scaling where ΔV_{BB} is kept at 0. From these sample points, one can see that significant energy savings can still be achieved by tuning both V_{CC} and V_t rather than employing simple dynamic voltage scaling.

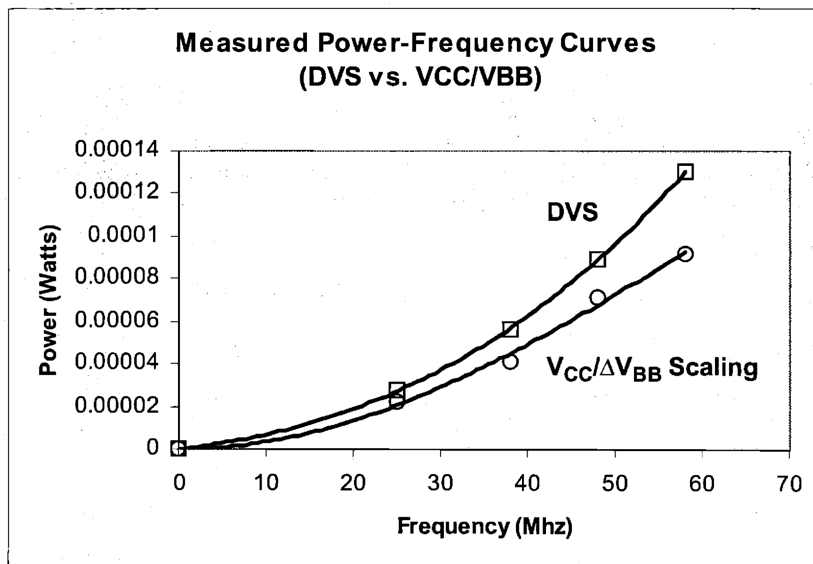


FIGURE 6-27. Measured power vs. frequency curves for dynamic voltage scaling versus optimal $V_{CC}/\Delta V_{BB}$ scaling.

6.7 Alternative Threshold Tuning Techniques

As shown in the test chip, a triple well technology can be used to effectively tune threshold voltages in order to provide V_{CC} / V_t scaling to minimize overall power dissipation for a VLSI system. For a circuit that can operate at multiple operating conditions and frequencies, it is useful to be able to dynamically tune both the threshold voltage and supply voltage so that the circuit is always ensured to be operating at a minimum power condition. However, as described earlier, a standard triple well process like that of Figure 6-23 will have limitations on the effectiveness of V_t tuning, especially with forward bias conditions that are meant to speed up devices and lower threshold voltages. Another problem with triple well technologies is that as technology scales, the body factors tend to degrade as well. Smaller body factors are beneficial from a circuit point of view because it tends to improve performance of stacked devices. However, from a V_t control point of view having larger body factors is better because it provides greater tunability of device threshold voltages. Interestingly, work has been done by [65], which suggests that having large body factors might turn out to still be acceptable. If two technologies are designed for a fixed I_{off} amount in the standby state, it might be more effective to have a larger body factor because the on current can be made higher than for a comparable technology with a lower body factor.

Another approach to dynamically tuning devices threshold voltages is to rely on novel technologies. One such promising technology is fully depleted dual gated SOI with active substrate (SOIAS)[45]. SOIAS devices are well suited for variable V_t applications because the threshold voltage varies linearly with back gate applied bias.

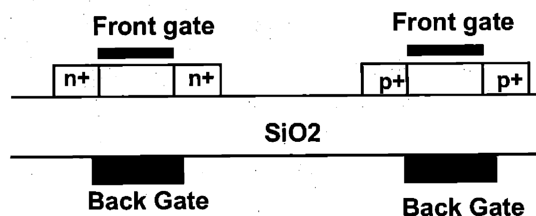


FIGURE 6-28. Dual gated SOI (SOIAS) with active substrate.

Traditionally, dual gated devices like SOIAS were primarily thought to be beneficial because of improved current drive and it's idealized subthreshold slope characteristics. However, a novel way to use these SOIAS devices is to individually tune threshold voltages by sweeping the back gate. For the N channel device, increasing the back gate voltage will increase the surface potential linearly, which decreases the V_t. Conversely, for the P channel device, decreasing the back gate voltage will decrease $|V_t|$. Because the threshold voltages are tuned through electrostatic effects, the V_t shift in both positive and negative directions can easily be controlled. Furthermore, there are no forward bias junctions difficulties like in the triple well technology that arise when V_t's are reduced. SOIAS devices are thus well suited for applications where supply voltages and threshold voltage can be dynamically scaled to minimize power dissipation. Because of the tunability of threshold voltages, the adaptive body biasing techniques described in the previous chapter can also easily be applied to SOIAS circuits in order to compensate for parameter variations. Threshold voltages can be tuned by sweeping the back gate voltages until the circuit critical path meets a target operating frequency. In fact, compared to bulk devices, fully depleted SOI devices are especially sensitive to V_t variations because of film thickness variations[66]. As a result, an adaptive threshold voltage compensation technique can be especially useful for tightening delay distributions. By employing multiple adaptive biasing controllers throughout a chip like described before, it would be possible to help tune out systematic variation components across a single die due to film thickness variation.

6.8 Automatic V_{CC}-V_t biasing

In the previous section, optimal V_{CC} / V_t operating points were explored theoretically and also through simulations and measurements of a DSP implemented in a triple well technology. In order to find the optimal V_{CC} / V_t points, the supply voltage V_{CC} and body bias value ΔV_{BB} were manually adjusted until the minimum power point was achieved. Because the main goal of this research was to simply characterize this optimal point and to show how frequency and operating conditions effect this operating point, a manual testing approach was sufficient. In this section, several possible schemes are introduced to automatically adjust supply voltages and threshold voltages to minimize overall power dissi-

pation. This is necessary so that a circuit can independently adjust the supply and threshold voltages when operating conditions change.

6.8.1 Open loop approach

The simplest way to implement a variable V_{CC} / V_t controller is to use an open loop approach where a lookup table simply sets the V_{CC} and ΔV_{BB} bias settings based on the circuit operating conditions. For example, the lookup table can be indexed by frequency and temperature, and for each entry an appropriate V_{CC} and ΔV_{BB} is selected. This read value can then be used to drive a D/A converter that drives the well body biases to set the appropriate ΔV_{BB} bias. Similarly the lookup table entry can also be used to modulate the power converter (PWM for a buck converter for example) to set the appropriate V_{CC} value. Considerable work in [67][68] has shown very effective power converter techniques for variable supply voltages that can switch rapidly with high power efficiency. Figure 6-29 below shows a typical implementation of an lookup table based controller implementation.

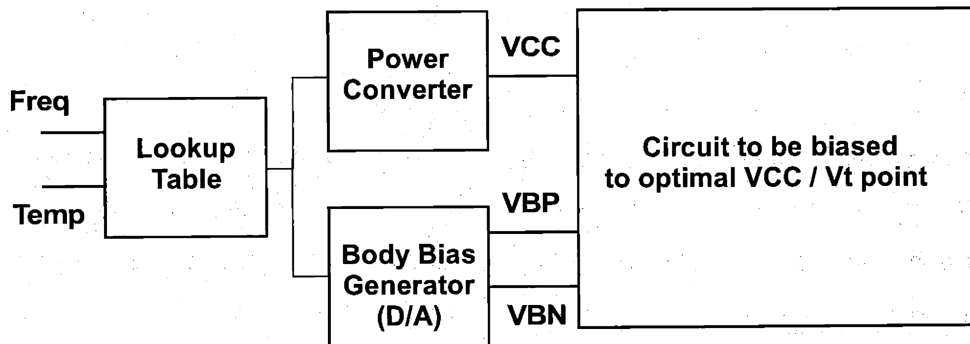


FIGURE 6-29. Open loop V_{CC} /V_{BB} controller using lookup table.

The lookup table entries can be determined through simulations or direct measurements. With simulations, the circuits should be completely characterized to determine optimal V_{CC} / V_{BB} settings that are programmed into the lookup table entries. One caveat though is that the simulations must be accurate enough to properly characterize the impact

of both V_{CC} and V_{BB} scaling, and also properly model forward body bias effects. For conventional device models, forward body bias regimes models have not been very accurate because these operating conditions are not used in traditional CMOS circuits. As a result, if forward and reverse body biasing approaches are used for V_t tuning, then these operating regimes need to be modeled more carefully in the future. Another way to determine the lookup table entries for the V_{CC} / V_{BB} controller is to perform direct measurements on the fabricated chips themselves. This is a much more accurate, though more complicated, way of determining optimal V_{CC} / V_{BB} operating conditions as compared to the previous simulation approach. The lookup table must be programmable so that the table entries can be updated after the circuits have been fabricated. Theoretically, if each individual chip is independently characterized, then V_{CC} / V_{BB} operating points can be custom optimized for each chip. This takes into account different parameter variations for each chip, but would be an impractical solution. Instead table entries would probably correspond to nominal operating conditions that are used to determine the optimal operating conditions of all chip samples. The drawback though is that the open loop controller would not be able to compensate for chip specific parameter variations (such as Vt variation effects), and the resulting optimal V_{CC} / V_{BB} biasing point could be suboptimal.

If the lookup table is calibrated for temperature variations as well, the control loop can be periodically refreshed to compensate for slowly drifting temperature fluctuations. As described earlier, temperature can have a large effect on optimal V_{CC} / V_t operating points, and can yield significant energy savings in a very aggressive low power system. However, time varying parameter variations such as hot carrier degradation, or electromigration effects will not be compensated because the table entries are fixed. Of course the chip can be periodically recalibrated and the lookup tables updated, but this again is not a practical solution.

6.8.2 Closed loop approach

A better approach to automatic V_{CC} / V_t scaling is to use a closed loop approach where one or both of the V_{CC} and V_{BB} bias mechanisms operates in a closed loop feedback fashion, where the circuit automatically sets the proper bias conditions. For example, a hybrid closed loop approach can be used to combine a V_{CC} lookup table with the adaptive body

bias generator described in the previous chapter. In a sense, the V_{CC} value is set in an open loop fashion through a lookup table tabulated as a function of varying workload and temperature conditions, while the V_{BB} value is automatically adjusted through the adaptive body bias controller. The adaptive body bias controller simply adjusts the PMOS and NMOS well body bias voltages (in both the forward and reverse bias directions) automatically in order to match the delay of critical path replica to that of the target frequency. This approach, for a given supply voltage, will automatically tune the threshold voltage so that the circuit operates exactly as fast as necessary. The automatic body bias controller acts to automatically adjust the threshold voltage so that the circuit V_{CC} / V_t operating point lies on the locus curve that defines a fixed frequency. This is a significant improvement over the previous open loop case where the table entries are set in an open loop fashion. With a closed loop approach, the threshold voltage will be adaptively adjusted by the chip itself to set the proper V_{BB} bias voltage for the given V_{CC} and frequency target.

In this hybrid approach, a lookup table is still used to characterize the V_{CC} optimal operating point for different frequency and temperature conditions. Again these V_{CC} targets can be derived from simulations or direct measurements of the actual die of interest, but regardless, the V_t values are set automatically to ensure that the chip operates only as fast as necessary. For the most accurate tabulation of optimal V_{CC} choices for a given frequency and temperature, measurements would still have to be performed on each chip like in the open loop case, although the testing would be easier because only one variable (V_{CC}) would need to be swept. By doing this, the lookup table entries would be optimized for the particular chip with particular random variations.

However, even if the V_{CC} values are not chosen optimally (by completely characterizing the chip for all possible operating conditions for each individual die), the V_{CC} / V_t scaling results can still be acceptable. This is because the body bias controller can attempt to compensate for any parameter variations by adjusting the V_{BB} bias voltage directly. Although theoretically when the device parameters change, both the V_{CC} and V_{BB} optimal values shift, in reality the V_{CC} fluctuations could be small depending on the nature of the variation source. As a result, even if V_{CC} is characterized by nominal optimal target values (for a given frequency and given temperature), the resultant controller could still

yield close to minimum power dissipation. A hybrid controller framework using an adaptive body biasing generator is shown below.

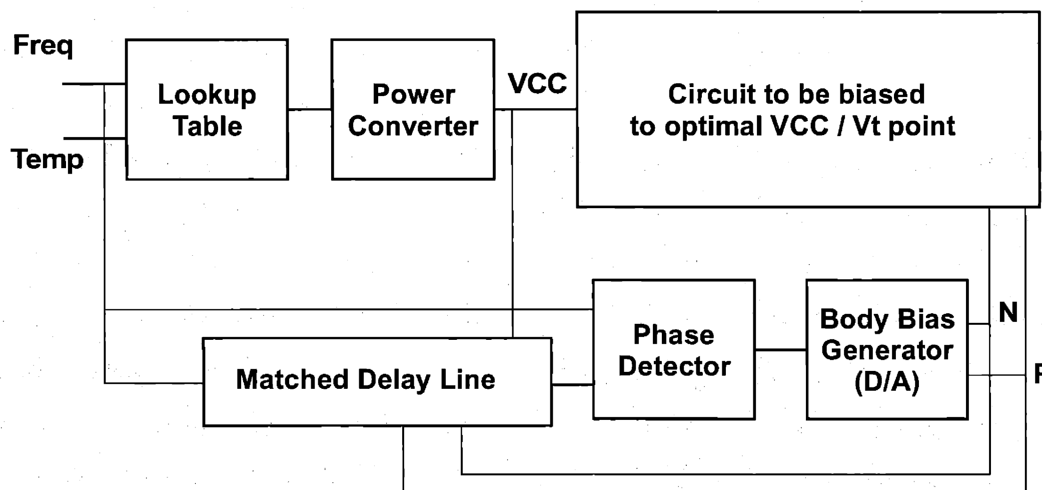


FIGURE 6-30. Hybrid loop V_{CC}/V_{BB} controller using lookup table and adaptive body biasing circuit.

The key fact is that the supply voltage is set open loop but the body bias value is set in a closed loop fashion using an adaptive body bias controller. When the loop is refreshed, or the freq or temperature parameters change, the controller must reset the body bias controller to maximum forward bias (fastest speed), so that the adaptive body bias generator selectively slows down the V_t's. This ensures that when a new lookup table entry is activated, the body bias feedback loop starts out with the fastest N, P bias values so that the circuit computation cycle time is always satisfied, and the circuit will not have to be stalled. The maximum forward bias of course must be chosen such that performance is maximized yet PN junctions are not forward biased excessively.

Like in the open loop case, this controller uses a lookup table that is also tabulated by temperature, and thus the circuit optimal operating point can be recalibrated to take into account time varying temperature changes. For example, the loop can be periodically refreshed or a control signal can reset the loop whenever the temperature changes by a certain amount. Temperature effects have a large impact on the optimal V_{CC} / V_t operating point since device parameters such as subthreshold slope, threshold voltage, and mobility

are all effected. Because the hybrid approach uses a closed loop control system to set the threshold voltage, it can compensate for other slowly changing parameter variations as well. For example, hot carrier or electromigration effects on circuit performance can be compensated by adjusting the body bias settings to ensure circuits operate at the target frequency. However, since only the body bias setting is controlled in a feedback loop, the operating speed can be maintained, but unfortunately the V_{CC} / V_t bias point may no longer coincide with the optimal operating point.

6.8.3 Dual loop control

A final alternative to optimizing V_{CC} and V_{BB} bias values for minimum power consumption is to employ a true dual loop control mechanism. With a genuine dual loop controller, both V_{CC} and V_t can be dynamically adjusted by the circuit itself to minimize the total power dissipation. Parameter variations will automatically be taken into account whenever the controller is refreshed, and thus temperature variations, parameter variations, and dynamically changing device parameters will all be taken into account when finding the optimal V_{CC}/V_{BB} operating point. Furthermore because no lookup table approach is used, this technique is cost effective because no measurements or simulations are needed to characterize the optimal operating points.

The drawbacks though are that two dimensional control loops in general are difficult to design, to stabilize and to ensure that the target circuit is functional at all times. As a result, an alternative scheme that effectively decouples the V_{CC} control loop with the V_t loop is described in the flow chart below.

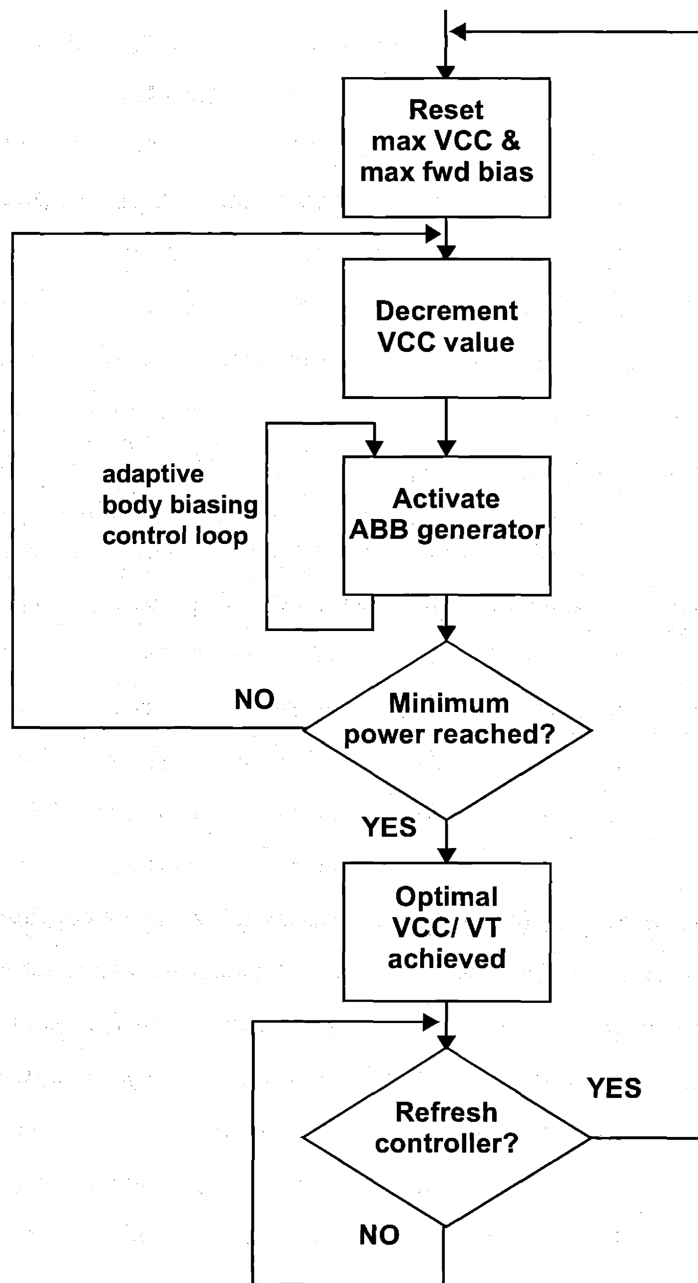


FIGURE 6-31. Dual loop V_{CC} /V_{BB} controller operation.

The basic principle is rather simple although the controller operation is slightly more difficult to implement than in the previous case. The dual loop controller uses a brute force approach that steps the supply voltage from the maximum range to the minimum range, at each sample allowing an adaptive body bias generator to automatically seek out the appropriate body bias voltage. At each step the power of a test circuit is compared to the previous value, and whenever the incremental change in power increases, the previous setting would correspond to the optimal power point. This brute force method in effect samples points on the V_{CC}-V_t locus curves of constant frequency for a series of different V_{CC} values, and directly finds the optimal operating condition. The appropriate body bias amount delta V_{BB} is automatically chosen by the adaptive body biasing loop in order to meet the target frequency at the supplied V_{CC}. This straightforward approach is very effective because the actual minimum operating power can be achieved since this point is automatically selected by the chip itself, and the V_{CC} control loop is effectively decoupled from the V_t control loop.

The control mechanism is more complicated than the open loop and hybrid approach because the controller steps down the supply voltage and compares power levels directly to minimize overall power. In order to ensure that the core circuitry remains fully functional during the V_{CC} / V_t optimization procedure, it is useful to initially fix the operating point to maximum V_{CC} and minimum V_t biasing conditions. The controller can then independently sweep V_{CC} and V_{BB} for the test circuit (which consists of the matched delay line and power estimation circuitry) independent of the core circuitry bias values. With this approach the adaptive body biasing loop can also be free to lock to the target operating frequency in either direction (speeding up or slowing down) because these bias voltages are not directly applied to the core circuitry. After the appropriate V_{CC}/V_{BB} values are determined, the core circuitry can then be appropriately biased for minimum power operation. This roundabout scheme is important to ensure the chip is completely functional at all times. This is because as the V_{CC}/V_{BB} values are varied by the controller, the critical path delay may occasionally result in performance that is faster or slower than the operating frequency, which would cause failures in the core circuitry if biased directly. If the chip can be stalled during the V_{CC}/V_{BB} optimization loop, then this constraint could be relaxed and the controller implementation simplified.

During the course of the optimization routine, the supply voltage V_{CC} is systematically sampled, and the adaptive body bias generator automatically seeks out the appropriate V_{BB} value. This can take long periods of time since for each V_{CC} sample, several iterations within the body biasing loop will be required. As a result, the V_{CC}/V_{BB}controller operates very slowly, and can take on the order of microseconds to lock onto a new operating point. However, calibrating the circuit for optimal V_{CC}/V_{BB}settings occurs very infrequently (only when workloads change, or periodic updates to compensate temperature and parameter variations) so speed is not important.

The optimal V_{CC}/V_{BB} point can be recalculated as workload requirements change and the target operating frequency varies. Likewise, by periodically refreshing the optimal V_{CC}/V_{BB} controller, the optimum operating point can be tweaked to compensate for time varying fluctuations in temperature or other parameter changes due to hot carrier effects or electromigration for example. This is in contrast to the previous lookup table approaches (both open loop and hybrid loop), where it was not possible to optimally track dynamic parameter variations unless the lookup tables were completely recalibrated during the lifetime of the chip. For example, with the lookup table based controller, periodically refreshing the control loop could help compensate for slowly varying temperature fluctuations (provided the lookup table was characterized by temperature and frequency), but the optimal V_{CC} operating point could not be dynamically adjusted due to other variation effects. Furthermore the lookup table approach provided only an approximate target value for the V_{CC}, (and V_{BB} for the open loop case), that was derived from simulations or generalized measurements. With a true dual loop control mechanism though, the ideal operating point can be automatically selected even when operating parameters for the chip change dynamically. Each time the controller is refreshed, a new optimal V_{CC}/V_{BB} biasing point is recalculated completely, which ensures that the chip operates at the lowest power condition.

6.8.4 Degenerate optimal operating points

In some cases, the true optimum cannot be reached because the bias range is too limited. For example, if the nominal threshold voltage is too high to effectively lower Vt through forward biasing, then the controller could continue dropping V_{CC} to lower power con-

sumption until the limit is reached where the target operating frequency cannot be satisfied any more. In this limit, the minimum energy point would simply correspond to the previous V_{CC} / Vbb value, and the optimal V_{CC}/Vt point would correspond to the one with maximum allowed forward bias. For circuits that are constrained by this mechanism for all target operating frequencies, one could simplify dual loop controller described earlier by simply utilizing maximum forward biasing at all times (provided this can be fixed), and to simply adjust V_{CC} as low as possible, i.e. use a dynamic voltage scaling approach. This is true because in effect, the optimal threshold voltages for these circuits can never be reached, so the best thing to do is to lower them as much as possible.

For the triple well DSP testchip described earlier in this chapter, the limitations in biasing range are further constrained by the fact that the maximum forward body bias amounts vary with the supply voltage because Pwell and Nwell junction can be forward biased at low V_{CC} values. In this case, the acceptable bias ranges actually change depending on the supply voltage, where at low V_{CC} values the forward bias range can only tolerate very small amounts to ensure that forward body bias currents are controlled. As V_{CC} is dropped, and the V_{BB} swept, there will actually be a point were the power suddenly increases dramatically because the forward bias currents are simply too large. However, rather than search for this minimum power point, it makes better sense to monitor the body currents and to limit the forward bias range to ensure that the forward bias currents do not exceed a fixed value. This ensures that only a reasonable amount of forward bias is applied. In this case the dual loop controller can also be simplified by monitoring the body current rather than total power consumption of the test chip. Since the amount of tolerable forward body biasing will vary with V_{CC} though, a dual loop control scheme is still required. As a result, for different V_{CC} values, the adaptive body bias generator should still be activated, but the cutoff point would correspond to the point where the forward body bias currents exceed a fixed amount.

6.8.5 Test circuit for power estimation

In the most general case for V_{CC}/V_{BB} tuning to minimize total power consumption, the nominal threshold voltages are low enough such that an actual minimum operating point can be reached. For these cases, circuits are not limited by extreme forward bias currents,

but instead exhibit a true balance between dynamic power and subthreshold leakage currents. This is in contrast to the examples of the previous section where forward bias limitations set how low V_{CC} could be scaled, which allowed simplified constraints such as body current limits or basic dynamic voltage scaling to set the minimum achievable power dissipation point. On the otherhand, with a general V_{CC}/V_{BB} controller for a chip that can be biased into a true minimum energy point, the actual power of the chip must be directly computed.

Thus, one of the critical elements of the dual loop controller for V_{CC}/V_{BB} optimization is to accurately measure the power dissipation of the chip itself. While a direct measurement of the chip would be ideal, it may not be practical without detrimentally affecting the impedance or the efficiency of the power converter for the chip. Instead, it is useful to measure the power delivered to a small test circuit that represents a scaled version of the total chip power dissipation. Figure 6-32 below shows a typical scenario where the currents of a test chip are measured and the I*V product computed to estimate the total power of the core chip as supply voltages, body bias, and operating frequencies change. In this example, the body currents are neglected, although the power dissipation associated with these currents can easily be taken into account as well.

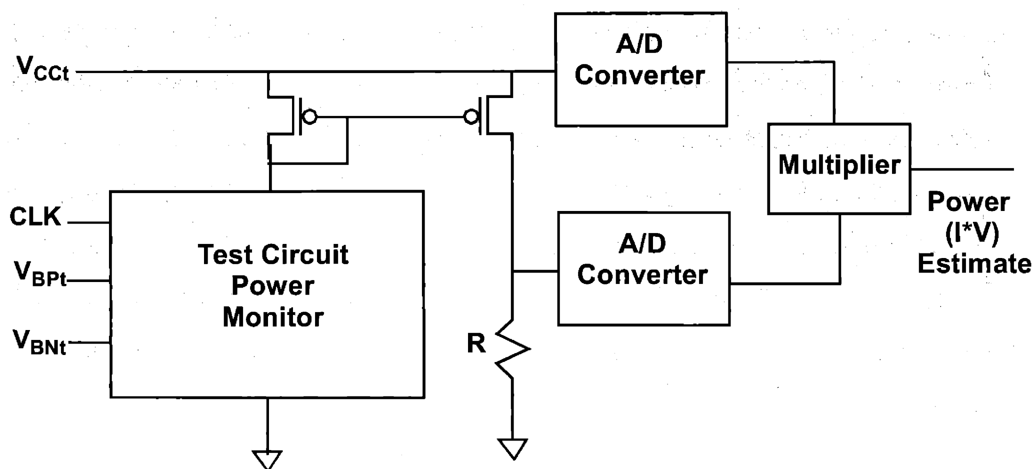


FIGURE 6-32. Possible use of test circuit to estimate total power consumption (dynamic + leakage) assuming body currents are negligible.

Because of the strong interaction between subthreshold leakage currents and dynamic switching currents on the optimal operating point, it is important to try to match the power measurement of the test circuit with that of the total chip. To do this effectively, the test circuit needs to exhibit the same ratio of subthreshold leakage currents to dynamic switching currents that would exist for the full chip operation. Provided this is true, then biasing the test circuit for minimum power dissipation would also minimize the power dissipation of the entire chip as well. Accurately designing a test circuit that will properly track the power dependency on V_{CC} and V_{BB} for the full chip is nontrivial. A first order approach to provide an effective test circuit is to use a simple inverter chain in parallel with a set of nonswitching gates, as shown in Figure 6-33. By varying the number of switching inverters versus the number of nonswitching inverters, different ratios in dynamic currents versus leakage currents can be achieved.

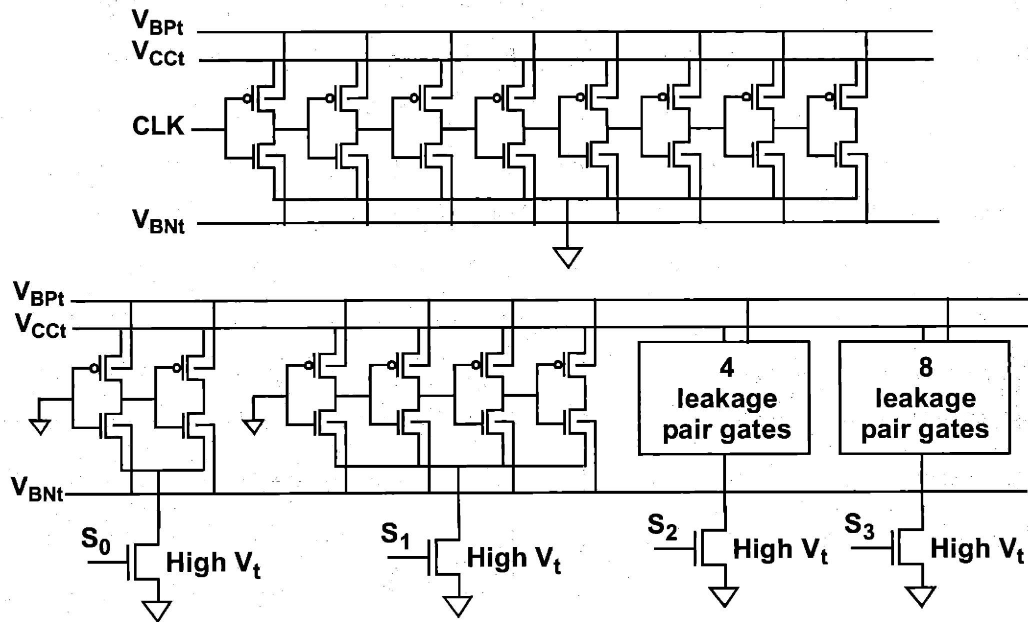


FIGURE 6-33. Test circuit with programmable taps to set ratio between dynamic and leakage power.

Theoretically from Figure 6-5, a test chip can be made to have the same power profile (to within a constant factor) as that of the whole chip as long as the ratio between the switched capacitance C_{eff} and leakage constant I_0 is maintained. The other parameters in the equations such as supply voltage, frequency, threshold voltage, and subthreshold slope

will all track between the test circuit block and the full chip. Thus, by increasing the number of switching inverters, one can adjust the nominal C_{eff} value, and by increasing the number of non switching inverters, the total leakage width can be increased, thereby adjusting the leakage constant I_0. Even though the testchip does not exhibit the same circuit architecture (for example inverters are used rather than complex gates) as the actual core circuitry, the impact of V_{CC} and V_t scaling on power tends to track approximately well. The chip can be simulated, or measured directly to determine the appropriate test circuit configuration to properly balance C_{eff} and I_0. By performing extensive simulations on the full chip over all biasing conditions, one can tweak the test chip C_{eff} and I_0 ratios, or even tweak the circuits structures themselves, to try to make a best fit approximation to the power dissipation characteristics of the total chip. The test chip configuration can also be adjusted after direct measurements are made to provide more accurate matching. For example, the programmable high V_t switches can be used to selectively turn on sets of nonswitching or switching inverters to adjust the C_{eff} and I_0 ratios based on direct measurements.

Chapter 7

Conclusions

As technology continues to scale, subthreshold leakage currents will increase exponentially. Estimates on microprocessors show that subthreshold leakage currents can easily consume upwards of 30% of the total power budget in an aggressive technology. In the past, circuit techniques and architectures ignored the effects of these currents because they were insignificant compared to dynamic currents since threshold voltages were so high. However, in modern technologies the role of subthreshold leakage currents cannot be ignored and becomes increasingly dominant with future scaling. New circuit techniques and design considerations, like the ones described in this thesis, must be developed to control leakage currents in both the active and standby modes in order to provide low power solutions. Fortunately, industry has begun to respond to this problem and subthreshold leakage has become an active research topic.

Standby leakage reduction during the sleep mode has been recognized as a key methodology to improve system lifetimes. Burst mode systems that spend majority of their time in idle modes are especially susceptible to large standby currents. Dual V_t partitioning and stack effect provide some energy savings, but using MTCMOS or body biasing techniques is much more effective at reducing leakage currents during the standby state. Since body biasing techniques are only applicable in triple well (or similar)

advanced technologies, MTCMOS techniques are becoming more mainstream technique for modern technologies. A large contribution of this thesis work was to add to the state of the art of MTCMOS technology. In particular, high V_t sleep transistor sizing issues were modeled better, and a hierarchical sleep transistor sizing methodology was developed, which for the first time presents a systematic procedure for sizing high V_t power switches to ensure that performance is always maintained. Future work integrating this transistor sizing methodology into CAD tools for MTCMOS circuit design would be very useful. This thesis further contributed to the art by providing new architectures for MTCMOS sequential circuits that retain state during the standby modes. By more fully understanding sneak leakage paths and pitfalls associated with dual V_t sequential structures, this work developed novel and improved sequential circuits that improve upon existing art. Furthermore, completely novel sequential circuits that utilize a unique way to retain state using leakage feedback gates was invented and potentially can give rise to new circuit architectures and functions (for example a dynamic flip flop that holds state during the standby mode). These techniques to retain state and eliminate leakage currents can also be applied in the future to other types of sequential circuits. For example, pulse latches and TSPC flip flop schemes can also be converted into MTCMOS sequential circuits using some of the techniques and methodologies introduced in this work. Other dual V_t techniques that provide standby current reduction without the performance degradation and sizing issues associated with MTCMOS were explored. In particular, an imbedded dual V_t technique was developed that directly used high V_t and low V_t devices within existing gates. A special case of this principle, dual V_t domino, was developed as well and shows how this technique can be effectively applied to provide low leakage during the standby state with virtually no degradation of performance. This works shows that for certain logic styles, dual V_t techniques other than MTCMOS may be more effective at reducing standby leakage.

Projections also show that active mode subthreshold leakage currents can become increasingly large as well. Unfortunately, this is a more difficult problem to control because fast active device performance inherently requires lower threshold voltages and thus higher leakage currents. Improvements in device technology in reducing subthreshold slope or leakage current constants can be helpful, but circuit techniques to enable

higher performance operation simultaneously with low leakage currents are not possible. Instead, the best one can do is to employ circuit techniques that can be used to slow down devices that are faster than necessary. Previous research shows that dual V_t gate partitioning, (and equivalently selective body biasing in a triple well process) can be useful for slowing down non critical paths in order to reduce active subthreshold leakage currents. This thesis explored another way to reduce active leakage currents by actively compensating for parameter variations in a triple well (or similar) technology. Since percent variations in threshold voltages will increase in future technologies, parameter variations can contribute to large amounts of unwanted leakage currents. The adaptive body biasing controller introduced in this thesis is a novel way to automatically tune device threshold voltages so that circuit blocks operate only as fast as necessary. By allowing forward body biasing as well, yield can also be improved. With future technology scaling and increased system integration, chip sizes will continue to increase and parameter variations will become increasingly large. Having a circuit mechanism that automatically tunes out parameter variations can be extremely valuable, thereby using circuit techniques to solve problems that might limit future fabrication tolerances.

A final contribution of this research was to provide a framework for exploring optimal V_{CC}/V_t scaling during active mode operation. By tuning both supply voltages and threshold voltages, circuits can be biased to the optimal point where subthreshold leakage currents and dynamic currents are balanced. In a sense, rather than try to limit subthreshold leakage currents during the active mode, this work showed that it is more energy efficient to tradeoff increased subthreshold leakage currents with reduced dynamic currents. A theoretical framework was developed to explore how optimal V_{CC}/V_t operating points vary with workload, temperature, and circuit parameters, and a circuit technique to automatically find the optimal V_{CC}/V_t operating point was developed. This technique has many applications and is fully compatible with existing low techniques (parallelization, or reducing switching capacitance for example), and together can lead to extremely low power yet high performance circuit solutions.

Because of the rapid progress in semiconductor technology and scaling, circuit behaviors are changing dramatically, requiring entirely new circuit techniques and design

methodologies to address these new problems. Subthreshold leakage currents, previously negligible, have become an increasingly large part of total power consumption, and in the future other sources such as gate leakage currents must be addressed as well. This thesis contributed several new techniques to help reduce subthreshold leakage currents in both the active and sleep modes. One underlining theme though is that rather than trying to solve these problems by improving process technology, this research has concentrated on using novel circuit techniques to control subthreshold leakage currents. This proactive approach requires the flexibility to design new types of circuits that exploit the advantages but compensate for the disadvantages of aggressive new technologies. With continued research into advanced and original circuit techniques, many of the stumbling blocks that seem to limit the pace at which technology can continue to scale can soon be overcome.

References

[1] A. Chatterjee, "An Investigation of the Impact of Technology Scaling on Power Wasted as Short-circuit Current in Low Voltage Static CMOS Circuits," ISLPED, pp. 145-150, 1996.

[2] A. Keshavarzi, K. Roy, C. Hawkins, "Instrinsic Leakage in Low Power Deep Submicron ICs," International Test Conference, p. 146-155, Nov 1997.

[3] V. De, S. Borkar, "Technology and Design Challenges for Low Power and High Performance," ISLPED, pp. 163-168, 1999.

[4] D. Liu, C. Svensson, "Trading Speed for Low Power by Choice of Supply and Threshold Voltages," IEEE JSSC, vol. 28, no.1, pp.10-17, January 1993.

[5] A. Chandrakasan, S. Sheng, R. Brodersen, "Low-power CMOS digital design," IEEE JSSC, Vol. 27, No 4, pp. 473-484, April 1992.

[6] T. Sakurai, R. Newton, "Alpha-Power Law MOSFET Model and it's Applications to CMOS Inverter Delay and Other Formulas," IEEE JSSC, vol. 25, no. 2, pp. 584-594, April 1990.

[7] A. Chandrakasan, I. Yang, C. Vieri, D. Antoniadis, "Design Considerations and Tools for Low-voltage Digital System Design," 33rd Design Automation Conference, pp. 113-118, June 1996.

[8] R. Krishnamurthy, L. Carley, "Exploring the design space of mixed swing quadrail for low-power digital circuits,", IEEE Transactions on VLSI Systems, vol. 5, issue 4, pp.388-400, December 1997.

[9] S. Mutoh, S. Shigematsu, Y. Matsuya, H. Fukada, T. Kaneko, J. Yamada, "1V Multithreshold- Voltage CMOS Digital Signal Processor for Mobile Phone Application", IEEE JSSC, vol. 31, no. 11, pp. 1795-1802, November 1996.

[10] M. Horiguchi, T. Sakata, K. Itoh, "Switched-Source-Impedance CMOS Circuit For Low Standby Subthreshold Current Giga-Scale LSI's," IEEE JSSC, vol. 28, no. 11, pp. 1131-1135, Nov. 1993.

[11] T. Kawahara, M. Horiguchi, Y. Kawajiri, G. Kitsukawa, T. Kure, "Subthreshold Current Reduction for Decoded-Driver by Self-Reverse Biasing," IEEE JSSC, vol. 28, no. 11, pp. 1136-1144, Nov. 1993.

[12] Y. Ye, S. Borkar, V. De, "A New Technique for Standby Leakage Reduction in High-Performance Circuits," 1998 Symposium on VLSI Circuits, June 1998, pp. 40-41.

[13] T. Sakata, K. Itoh, H. Horiguchi, M. Aoki, "Subthreshold-Current Reduction Circuits for Multi-Gigabit DRAM's," IEEE JSSC, vol. 29, no.7, pp.761-769, July 1994.

[14] T. Sakata, K. Itoh, M. Horiguchi, M. Aoki, "Two Dimensional Power-Line Selection Scheme for Low Subthreshold-Current Multi-Gigabit DRAM's," IEEE JSSC, vol. 29, no.8 pp.887-895, August 1994.

[15] K. Itoh, "Limitation and Challenges of Multigigabit DRAM Chip Design," IEEE JSSC, vol.32, no.5, pp. 624-634, May 1997.

[16] W. Lee, et al., "A 1V DSP for Wireless Communications," ISSCC, pp. 92-93, Feb., 1997.

[17] L. Wei, Z. Chen, K. Roy, "Design and Optimization of Low Voltage High Performance Dual Threshold CMOS Circuits," IEEE Design Automation conference, 1998.

[18] S. Mutoh, T. Douseki, Y. Matsuya, T. Aoki, S. Shigematsu, J. Yamada, "1-V Power Supply High-Speed Digital Circuit Technology with Multithreshold-Voltage CMOS," IEEE JSSC, vol. 30, no. 8, pp. 847-854, August 1995.

[19] K. Seta, H. Hara, T. Kuroda, M. Kakumu, T. Sakurai, "50% active-power saving without speed degradation using standby power reduction (SPR) circuit," ISSCC Digest of Technical Papers, pp.318-319, February 1995.

[20] T. Kuroda, et. al, "A high-speed low-power 0.3um CMOS gate array with variable threshold voltage (VT) scheme," in Proceedings of CICC, pp.53-56, May 1996.

[21] T. Kuroda, T. Sakurai, "Threshold-Voltage Control Schemes through Substrate-Bias for Low-Power High-Speed CMOS LSI Design," Journal of VLSI Signal Processing Systems, vol. 13, pp.107-117, 1996.

[22] H. Mizuno, K. Ishibashi, T. Shimura, T. Hattori, S. Narita, K. Shiozawa, S. Ikeda, K. Uchiyama, "An 18uA Standby Current 1.8V 200Mhz Microprocessor with Self Substrate-Biased Data-Retention Mode," IEEE JSSC, vol. 34, no. 11, November 1999.

[23] M. Mizuno, K. Furuta, S. Narita, H. Abiko, I. Sasaki, M. Yamashina, "Elastic-Vt CMOS circuits for multiple on-chip power control," ISSCC Digest of Technical Papers, pp. 300-301, February 1996.

[24] M. Stan, "Low Threshold CMOS circuits with low standby current," ISLPED, pp. 97-99, 1998.

[25] H. Kawaguchi, K. Nose, T. Sakurai, "A Super Cut-off CMOS (SCCMOS) Scheme for 0.5-V Supply Voltage with Picoampere Stand-by Current," JSSC, vol. 35, no. 10, pp.1498-1501, October 2000.

[26] J. Kao, A. Chandrakasan, "Dual-Threshold Voltage Techniques for Low Power Digital Circuits," IEEE JSSC, vol.35, no.7, pp. 1009-1018, July 2000.

[27] J. Kao, A. Chandrakasan, D. Antoniadis, "Transistor Sizing Issues and Tool For Multi-Threshold CMOS Technology," 34th Design Automation Conference, pp. 409-414, June 1997.

[28] J. Kao, S. Narendra, A. Chandrakasan, "MTCMOS Hierarchical Sizing Based on Mutual Exclusive Discharge Patterns," 35th Design Automation Conference, pp. 495-500, June 1998.

[29] T. Sakuta, W. Lee, P. Balsara, "Delay Balanced Multipliers for Low Power/ Low Voltage DSP Core," ISLPED, pp.36-37, 1995.

[30] R. Brent, H. Kung, "A regular Layout for Parallel Adders," IEEE Transactions on Computers," vol. c-31, no.3, March 1982.

[31] J. Kao, "Dual Threshold Voltage Domino Logic," 25th European Solid State Circuits Conference, pp. 118-121, September 1999.

[32] G. Yee, C. Sechen, "Clock-delayed domino for dynamic circuit design," IEEE Transactions on VLSI Systems, vol. 8, issue 4, pp. 425-430, August 2000.

[33] S. Shigematsu, S. Mutoh, Y. Matsuya, Y. Tanabe, J. Yamada, "A 1-V High-Speed MTCMOS Circuit Scheme for Power-Down Application Circuits," IEEE JSSC, vol. 32, no. 6, pp.861-869, June 1997.

[34] H. Akamatsu, T. Iwata, H. Yamamoto, T. Hirata, H. Yamauchi, H. Kotani, A. Matsuzawa, "A Low Power Data Holding Circuit with an Intermittent Power Supply scheme for sub-1V MT-CMOS LSIs," 1996 Symposium on VLSI Circuits Digest of Technical Papers, pp. 14-15, 1996.

[35] K. Kumagai, H. Iwaki, H. Yoshida, H. Suzuki, T. Yamada, S. Kurosawa, "A Novel Powering-down Scheme for Low Vt CMOS Circuits," 1998 Symposium on VLSI Circuits Digest of Technical Papers, pp. 44-45, 1998.

[36] H. Makino, Y. Tsujihashi, K. Nii, C. Morishima, Y. Hayakawa, T. Shimizu, A Arakwa, "An Auto-Backgate-Controlled MT-CMOS Circuit," 1998 Symposium on VLSI Circuits Digest of Technical Papers, pp. 42-43, 1998.

[37] T. Kobayashi, T. Sakurai, "Self-adjusting threshold-voltage scheme (SATS) for low-voltage high-speed operation," Proceedings of IEEE CICC, pp 271-274, May 1994.

[38] T. Kuroda, T. Fujita, et al, "A 0.9V, 150MHz, 10mW, 4mm2, 2-DCT Core Processor with Variable VT Scheme," IEEE JSSC, vol. 31, no. 11, pp. 1770-1778, Nov 1996.

[39] T. Kuroda, T. Fujita, S. Mita, T. Mori, K. Matsuo, M. Kakumu, T. Sakurai, "Substrate noise influence on circuit performance in variable threshold-voltage scheme," ISLPED, pp.309-312, August 1996.

[40] S. Sun, P. Tsui, "Limitation of CMOS Supply-Voltage Scaling by MOSFET Threshold-Voltage Variation," IEEE Journal of Solid-State Circuits, vol. 30, no.8, pp. 947-949, August 1995.

[41] D. Frank, P. Solomon, S. Reynolds, J. Shin, "Supply and Threshold Voltage Optimization for Low Power Design," ISLPED, pp. 317-322, 1997.

[42] D. Burnett, K. Erington, C. Subramanian, K. Baker, "Implications of Fundamental Threshold Voltage Variations for High-Density SRAM and Logic Functions," Symposium on VLSI Technology Digest of Technical Papers, pp. 15-16, 1994.

[43] J. Kao, "SOIAS For Temperature, and Process Control," 6.374 Project, Massachusetts Institute of Technology, 1995.

[44] M. Miyazaki, et al., "A Delay Distribution Squeezing Scheme with Speed-Adaptive Threshold-Voltage CMOS for Low Voltage LSIs," ISLPED, pp. 49-53, 1998.

[45] I. Yang, C. Vieri, A. Chandraksan, D. Antoniadis, "Back gated CMOS on SOIAS for dynamic threshold control, "Proceedings IEDM, pp. 877-880, December 1995.

[46] V. Kaenel, P. Macken, M. Degrauwe, "A Voltage Reduction Technique for Battery-Operated Systems," IEEE JSSC, vol.25n no.5, pp1136-1140, October 1990.

[47] T. Kuroda, et al., "Variable Supply-Voltage Scheme for Low-Power High-Speed CMOS Digital Design," IEEE JSSC vol. 33, no.3, pp. 454-462, March 1998.

[48] V.Kaenel, M. Pardoen, E. Dijkstra, E. Vittoz, "Automatic Adjustment of Threshold and Supply Voltages for Minimum Power Consumption in CMOS Digital Circuits," ISLPED, pp.78-79, 1994.

[49] S. Narendra, D. Antoniadis, V. De, "Impact of Using Adaptive Body Bias to Compensate Die-to-die Vt Variation on Within-die Vt Variation," ISLPED, pp. 229-232, August 1999.

[50] De, V.K.; Xinghai Tang; Meindl, J.D. "Random MOSFET parameter fluctuation limits to gigascale integration (GSI)," Symposium on VLSI Technology Digest of Technical Papers, pp. 198-199, 1996.

[51] F. Frank, Y.Taur, M. Ieong, H. Wong, "Monte Carlo Modeling of Threshold Variation due to Dopant Fluctuations," Symposium on VLSI Circuits Digest of Technical Papers, pp. 171-172, 1999.

[52] D. Boning, J. Chung, "Statistical metrology: understanding spatial variation in semiconductor manufacturing," SPIE Symposium on Microelectronic Manufacturing, 1996.

[53] M. Nekili, Y. Savaria, G. Bois, "Spatial Characterization of Process Variations via MOS Transistor Time Constant in VLSI and WSI," JSSCC vol.34, no.1, pp.80-84, January 1999.

[54] M. Niewczas, "Characterization of the Threshold Voltage Variation: a Test Chip and the Results," IEEE Int. Conference on Microelectronic test Structures, vol. 10, March 1997.

[55] B. Stine, D. Boning, J. Chung, D. Ciplickas, J. Kibarian, "Simulating the Impact of Pattern-Dependent Poly-CD Variation on Circuit Performance," IEEE Transactions on Semiconductor Manufacturing, Vol. 11, No 4, November 1998.

[56] M. Miyazaki, G. Ono, T. Hattori, K. Shiozawa, K. Uchiyama, K. Ishibashi, "A 1000-MIPS/W microprocessor using speed adaptive threshold-voltage CMOS with

forward bias," ISSCC, Digest of Technical Papers, pp. 420-475, February 2000.

[57] D. Jeong, G. Borriello, D. Hodges, R. Katz, "Design of PLL-Based Clock Generation Circuits," JSSCC vol. sc-22, no.2, pp. 255-261, April 1987.

[58] J. Dickson, "On-Chip High-Voltage Generation in MNOS Integrated Circuits Using an Improved Voltage Multiplier Technique," JSSC, vol. sc-11, no. 3, pp.374-378, June 1976.

[59] V. Mehrotra, S. Sam, D. Boning, A. Chandrakasan, R. Vallishayee, S. Nassif, "A Methodology for Modeling the Effects of Systematic Within-Die Interconnect and Device Variation on Circuit Performance," Design Automation Conference Proceedings, June 2000.

[60] R. Gonzalez, B. Gordon, M. Horowitz, "Supply and threshold voltage scaling for low power CMOS," IEEE JSSC, vol. 32, no. 8, pp. 1210- 1216, Aug 1997.

[61] D. Pradhan, M. Chatterjee, "GLFSR-a new test pattern generator for built-in-self-test:, IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, vol. 18, issue 2, pp. 238-24, February 1999.

[62] V. Gutnik, A. Chandrakasan, "An Embedded Power Supply for Low-Power DSP," IEEE Transactions on VLSI Systems, vol. 5, no. 4, pp. 425-435, December 1997.

[63] J. Goodman, A. Dancy, A. Chandrakasan, "An Energy/Security Scalable Encryption Processor using an Embedded Variable Voltage DC/DC Converter," IEEE JSSCC, vol. 33, no.11, pp. 1799-1809, November 1998.

[64] T. Burd, T. Pering, A. Stratakos, R. Brodersen, "A Dynamic Voltage Scaled Microprocessor System," Proceedings ISSCC, pp. 294-295, February 2000.

[65] H. Koura, M. Takamiya, T. Hiramoto: "Optimum Conditions of Body Effect Factor and Substrate Bias in Variable Threshold Voltage MOSFETs", Jpn. J. Applied Physics., vol. 39, pp 2312-2317, 2000.

[66] D. Antoniadis, "SOI CMOS as a mainstream low-power technology: a critical assessment," ISLPED, pp. 295-300, 1997.

[67] A. Dancy, A. Chandrakasan, "Ultra low power control circuits for PWM converters," Proceedings of the IEEE Power Electronics Specialists Conference, 1997.

[68] V. Gutnik, A. Chandraksan, "An efficient controller for variable supply-voltage low power processing," Proceedings of Symposium on VLSI Circuits, pp.158-159, June 1996.

Appendix A

Optimal V_{CC}/V_t Test Chip Operation

This appendix will show in more detail the block diagram, board layout, and schematics of the DSP test chip for optimal V_{CC}/V_t tuning.

A.1 High Level Block Diagram

A global block diagram of the DSP test chip global block diagram is shown below in Figure A-1. The chip actually consists of 3 replica copies (plus 1 dummy) of a 4x4 MAC block. MAC_C is implemented with all low V_t devices, MAC_B is implemented with all high V_t devices, and MAC_A is implemented with all low V_t devices but with the N and P well body voltages automatically set by the adaptive body biasing generator. MAC_C and MAC_B utilize external supply and body bias voltages that are set manually with the test setup or from an external chip. Each DSP core has their own local power supplies in order to provide independent power measurements.

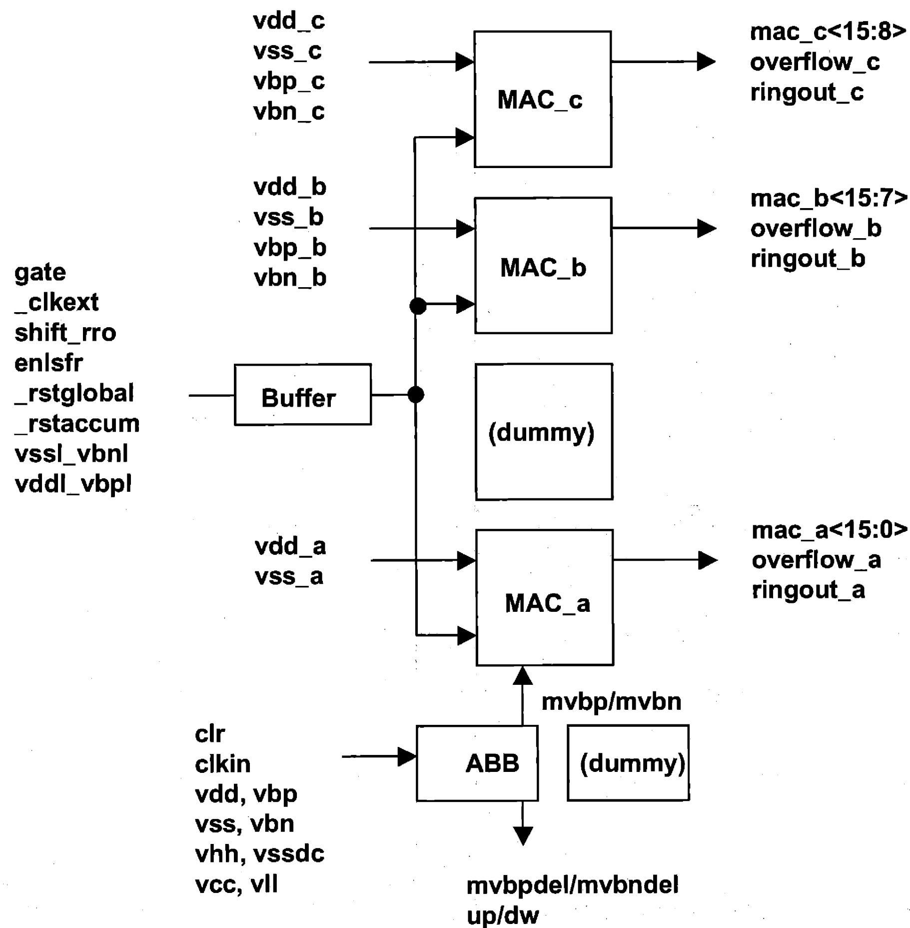


FIGURE A-1. High level block diagram of DSP test chip.

Each MAC block is driven with the same global control signals, but they each output their own accumulator result, overflow signal and ring oscillator signal. The control, I/O, and supply pins are described in the table below:

TABLE A-1. Test chip pin descriptions.

Signal	Type	Description
Vdd_abc	power	Local power line
Vss_abc	power	Local ground line
Vbp_ab	power	Local PMOS body bias (Nwell)
Vbn_ab	power	Local NMOS body bias (Pwell)

TABLE A-1. Test chip pin descriptions.

mac_abc<15:0>	output	MAC accumulator output voltage
overflow_abc	output	MAC accumulator overflow pin
ringout_abc	output	Ring oscillator output (1/2 frequency of critical path)
gate	input	Clock gating signal (active hi)
shift_rro	input	LFSR shift value and ring oscillator reset (active low) signal
enlfsr	input	Enable LFSR when hi, externally shift in data when low
_resetglobal	input	Reset entire chip (active low)
_resetaccum	input	Reset accumulator only (active low)
vssl_vbnl	power	Shared Vss, Vbn for peripheral circuitry
vddl_vbpl	power	Shared Vdd, Vbp for peripheral circuitry
_clkext	input	External clock signal (inverted)
clr	input	ABB clear
clkin	input	ABB clock signal
vdd, vbp, vss, vbn	power	ABB local power signals
vhh,vssdc,vcc,vll	power	ABB D/A bias voltages
mvpb, mvbn	power	ABB body bias buffered signals
mvpbdel, mvnbdel	power	ABB body bias signals
up, dw	output	ABB phase detector outputs

The pin signals used in the test chip are straightforward. The only complications arise from the control signals shift_rro and enlfsr, which can be used to shift external values into the linear feedback shift register for testing purposes. For standard chip operation, the LFSR has an explicit starting seed value, and the chip will automatically run a specified program. By asserting the gate signal after a fixed number of cycles, the output value can be compared to the predicted output sequence to determine the chip functionality.

The output signals are all buffered in order to directly drive the chip output pins. The input signals are also buffered, and distributed in a tree to each 4x4 MAC block. Within each 4x4 MAC block the control signals are again buffered in a tree to each local multiply accumulate unit. Detailed schematics later in this section show the different hierarchical levels in the chip. The input clock signal is also inverted before being distributed to the internal clock network. If the chip control signals are slaved to the external clock edge, then the internal registers will be configured such that the control signals will always be stable on the next internal clock edge.

The external clock signal is actually gated with the "gate" signal to stall the DSP clock as shown below

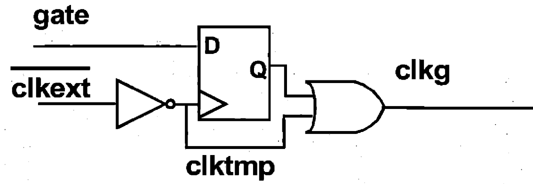


FIGURE A-2. Clock gating block.

The internal gated clock, "clkg", is either the inversion of the external clock or gated to be a constant high if the "gate" signal is asserted. This structure however only allows the gating signal to become asserted after a rising edge on "clktmp." Thus, when the gate signal is sampled by the flip flop, "clkg" will already be high, and will continue to be high. When the gate signal is deasserted, the flip flop output will change only after clktmp goes high again. As a result, when the "gate" signal goes low, the "clkg" signal will resume transitioning from the high state. This nice behavior ensures that no matter when the gate signal is applied, the clkg signal will never exhibit glitchy behavior.

A block diagram of one of the multiply accumulate cells (the MAC_abc blocks from Figure A-1 each consist of a 4x4 array of these blocks) is shown below

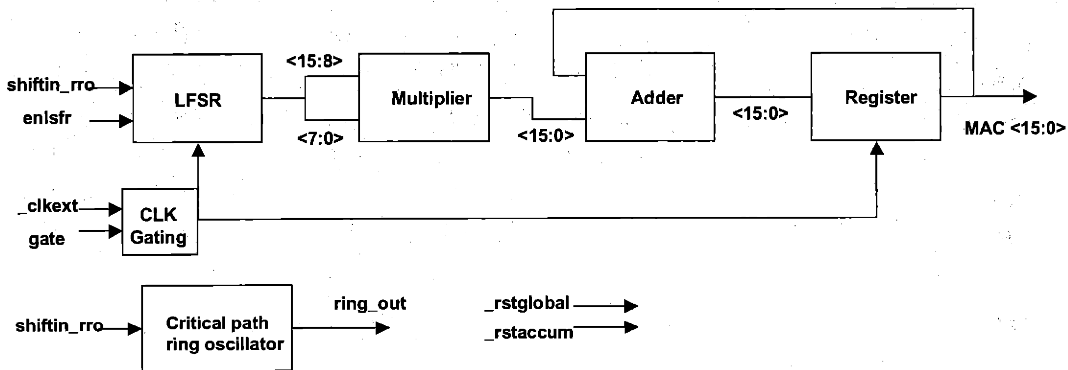


FIGURE A-3. MAC block diagram.

Similarly, the adaptive body bias function block diagram is given as

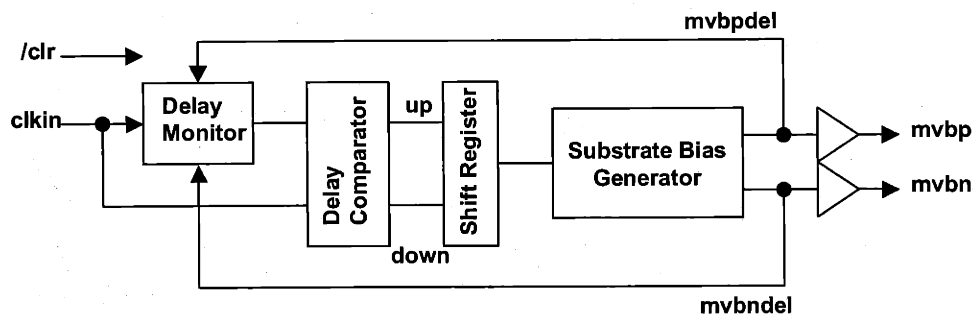


FIGURE A-4. ABB generator block diagram[Miyazaki].

where the substrate bias generator voltage is derived from a simple resistive chain

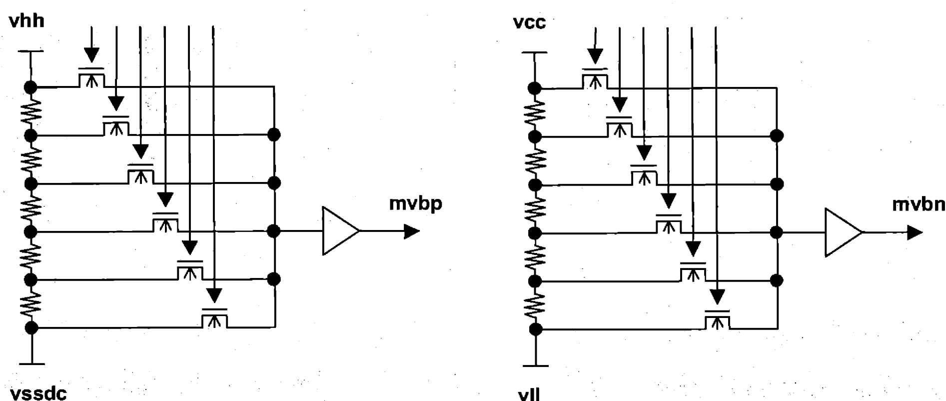
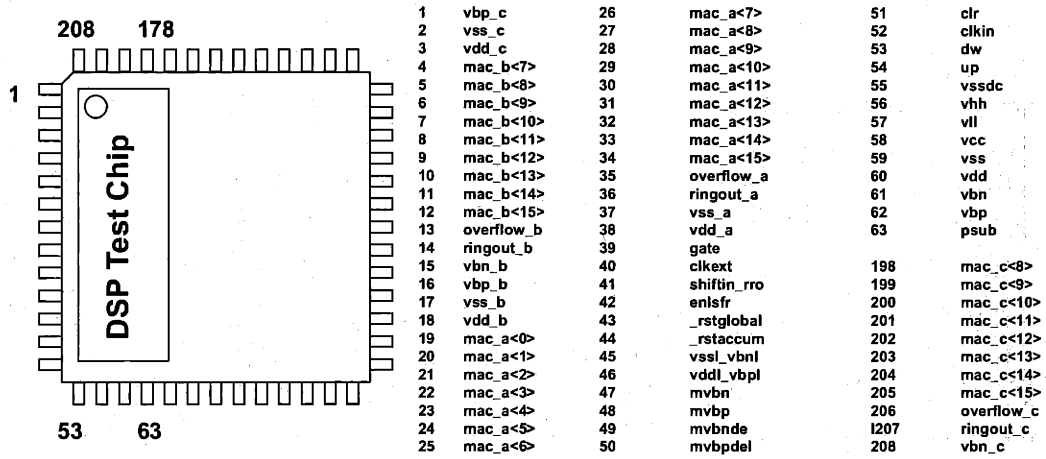


FIGURE A-5. Substrate bias generator[Miyazaki].

A.2 Chip Pinout

The chip pinouts are shown in the figure below. The actual DSP test chip corresponds to the only those pins on the left side of the die.



Pin	Name	Pin	Name	Pin	Name
1	vbp_c	26	mac_a<7>	51	clr
2	vss_c	27	mac_a<8>	52	clkin
3	vdd_c	28	mac_a<9>	53	dw
4	mac_b<7>	29	mac_a<10>	54	up
5	mac_b<8>	30	mac_a<11>	55	vssdc
6	mac_b<9>	31	mac_a<12>	56	vhh
7	mac_b<10>	32	mac_a<13>	57	vll
8	mac_b<11>	33	mac_a<14>	58	vcc
9	mac_b<12>	34	mac_a<15>	59	vss
10	mac_b<13>	35	overflow_a	60	vdd
11	mac_b<14>	36	ringout_a	61	vbn
12	mac_b<15>	37	vss_a	62	vbp
13	overflow_b	38	vdd_a	63	psub
14	ringout_b	39	gate		
15	vbn_b	40	clkext	198	mac_c<8>
16	vbp_b	41	shiftin_rro	199	mac_c<9>
17	vss_b	42	enlsfr	200	mac_c<10>
18	vdd_b	43	_rstglobal	201	mac_c<11>
19	mac_a<0>	44	_rstaccum	202	mac_c<12>
20	mac_a<1>	45	vssl_vbnl	203	mac_c<13>
21	mac_a<2>	46	vddl_vbpl	204	mac_c<14>
22	mac_a<3>	47	mvbn	205	mac_c<15>
23	mac_a<4>	48	mvbp	206	overflow_c
24	mac_a<5>	49	mvbnde	I207	ringout_c
25	mac_a<6>	50	mvbpdel	208	vbn_c

FIGURE A-6. Test chip orientation and pin mapping.

1.3 Board layout [Miyazaki]

The figure below shows the board layout (designed and built by Masa Miyazaki) to test the variable V_{CC}/V_t DSP triple well chip.

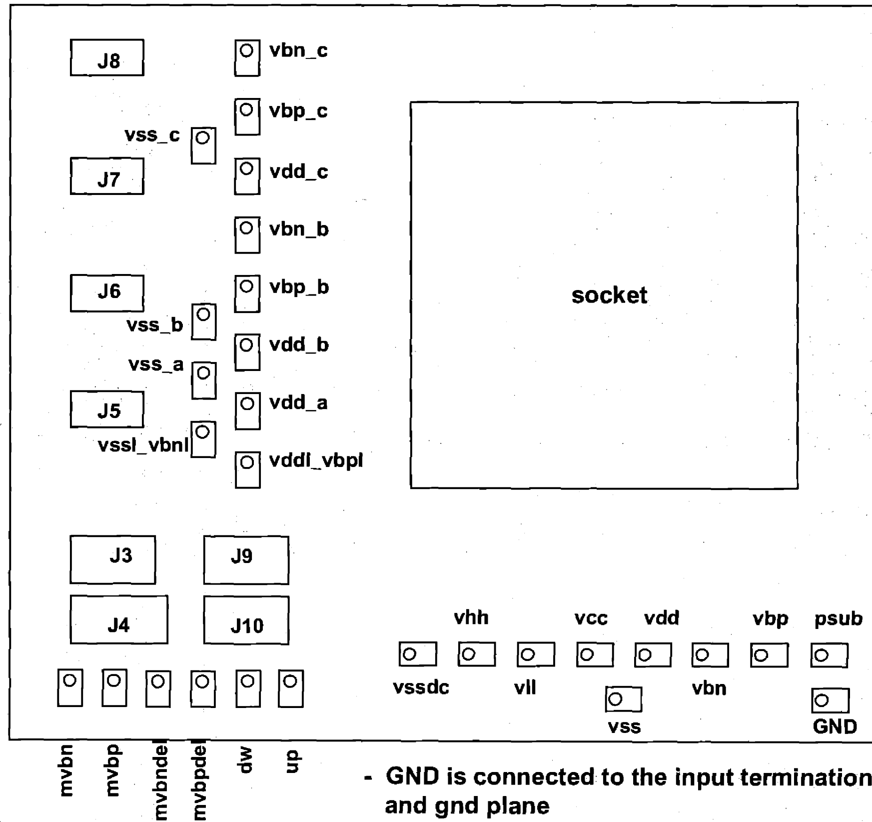


FIGURE A-7. Testing board configuration [Miyazaki].

The jumper signals are shown below. Depending on whether control signals are input to J5, J6, J7, or J8, different resistor dividers will be chosen.

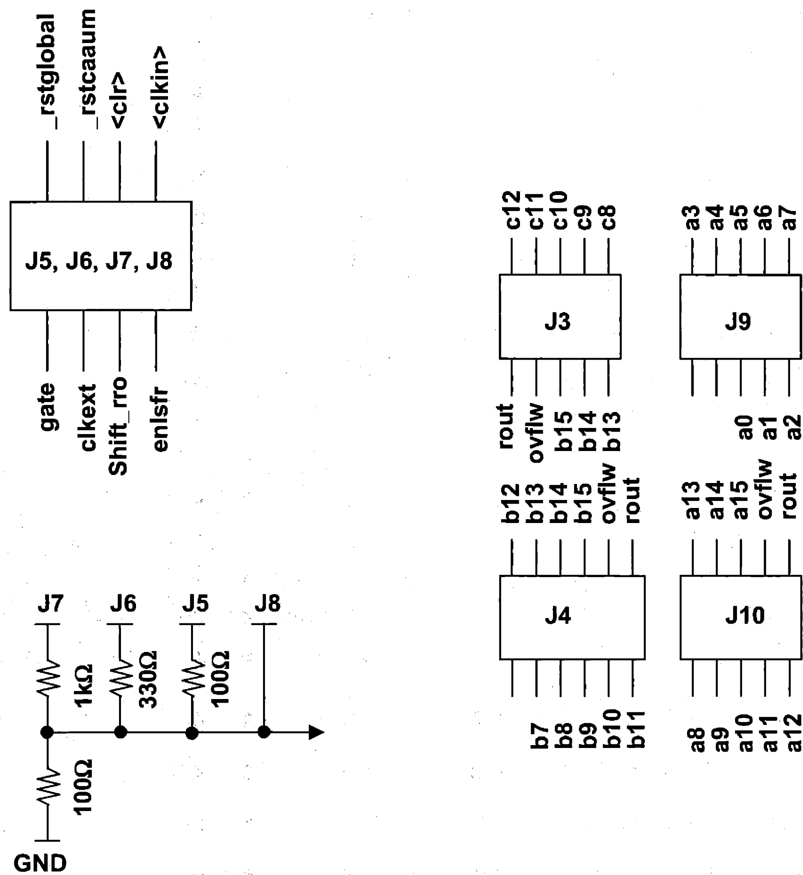


FIGURE A-8. Board I/O signals [Miyazaki].

A.4 Circuit Schematics

The following diagrams show the basic schematics for the DSP testchip, which illustrates the hierarchical nature of the design. Leaf cells are standard library elements, so are not shown.

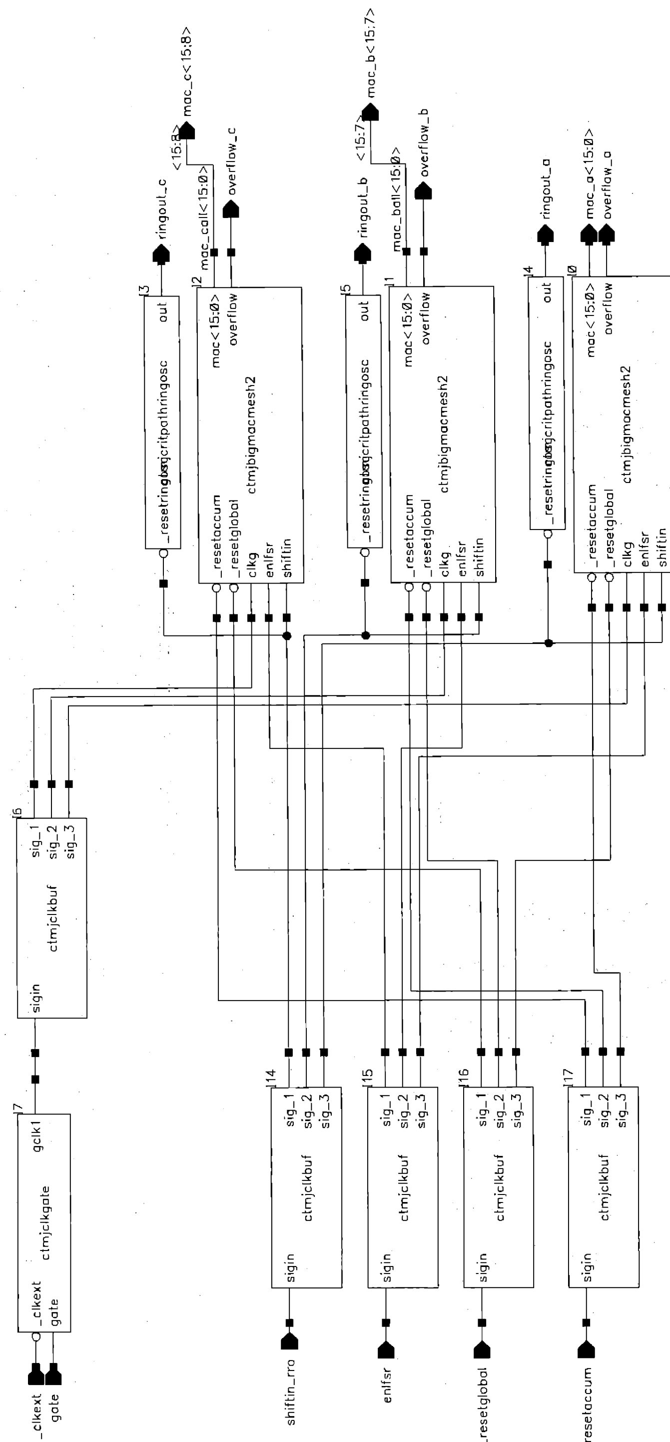


FIGURE A-9. Testchip global schematic.

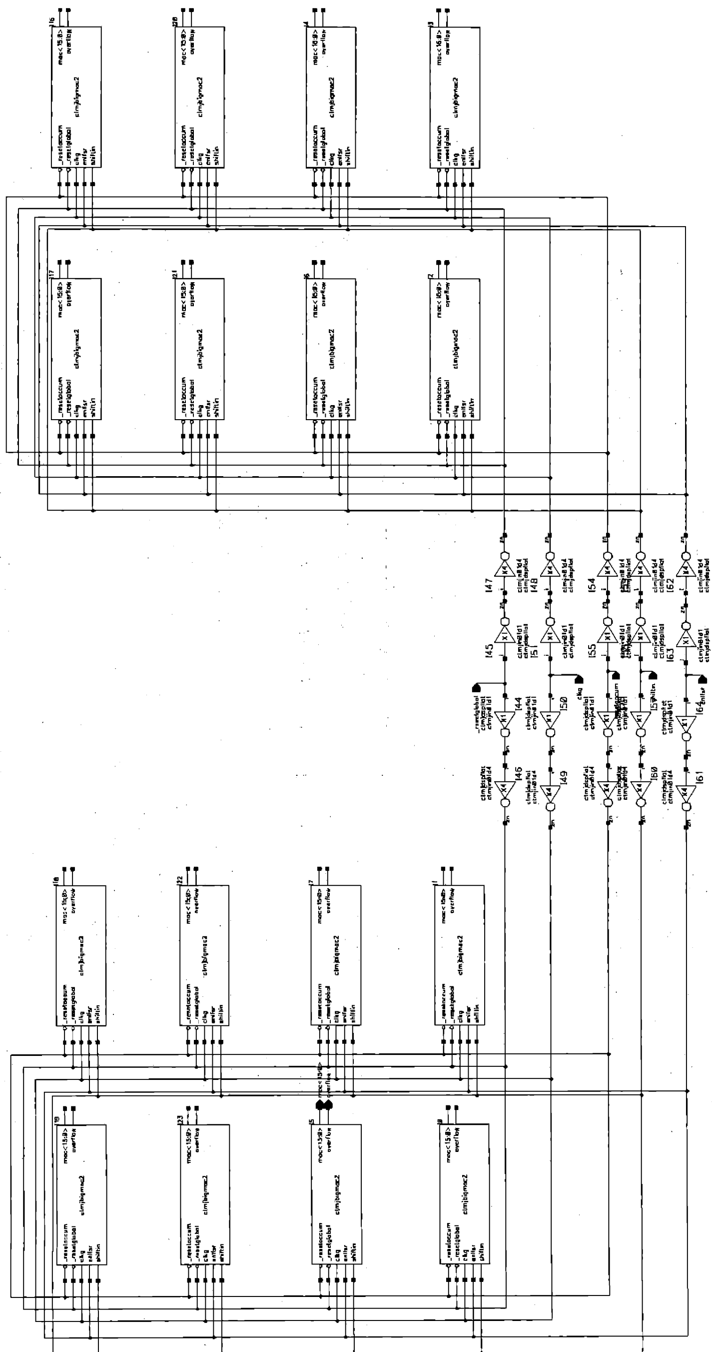


FIGURE A-10. 4x4 MAC mesh network.

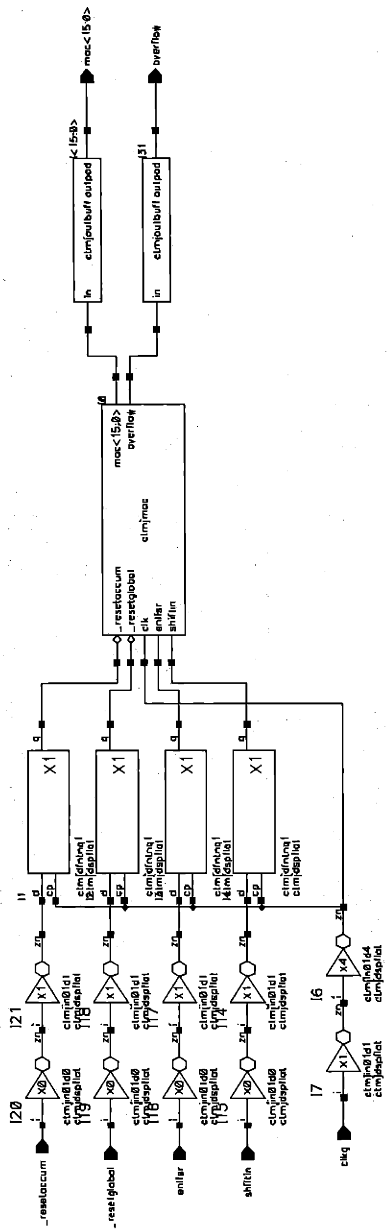


FIGURE A-11. MAC block with buffer stages shown.

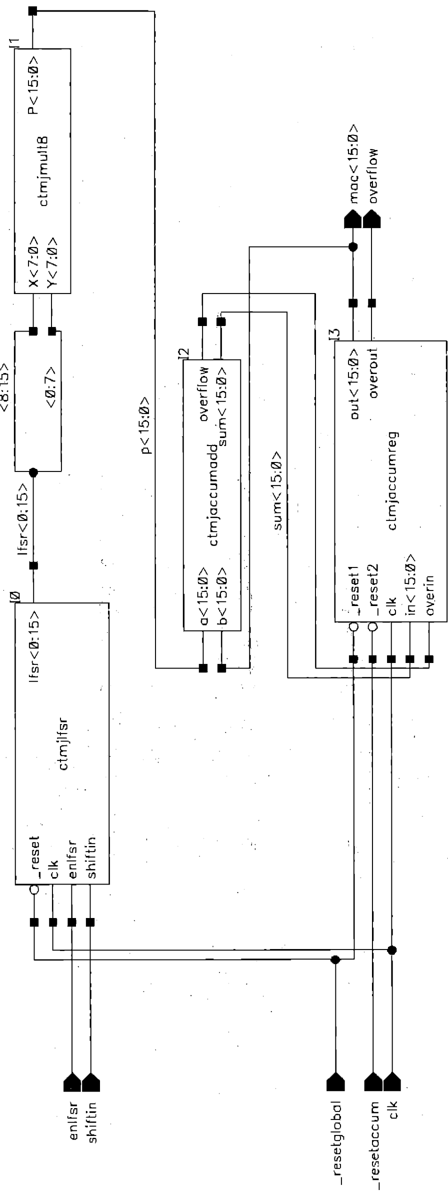


FIGURE A-12. Multiply accumulate core.

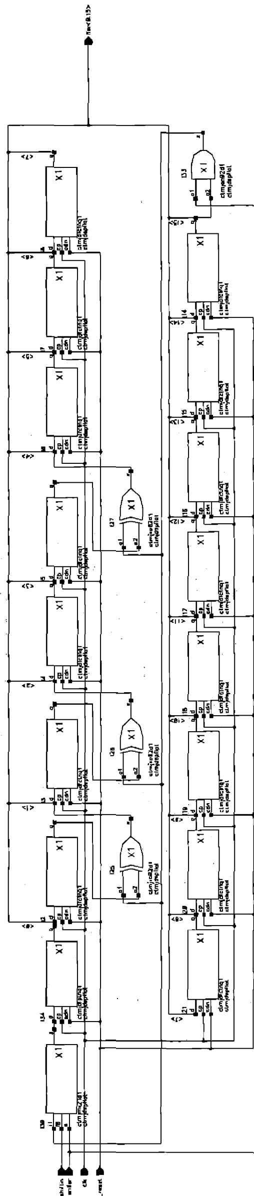


FIGURE A-13. Linear feedback shift register implementation.

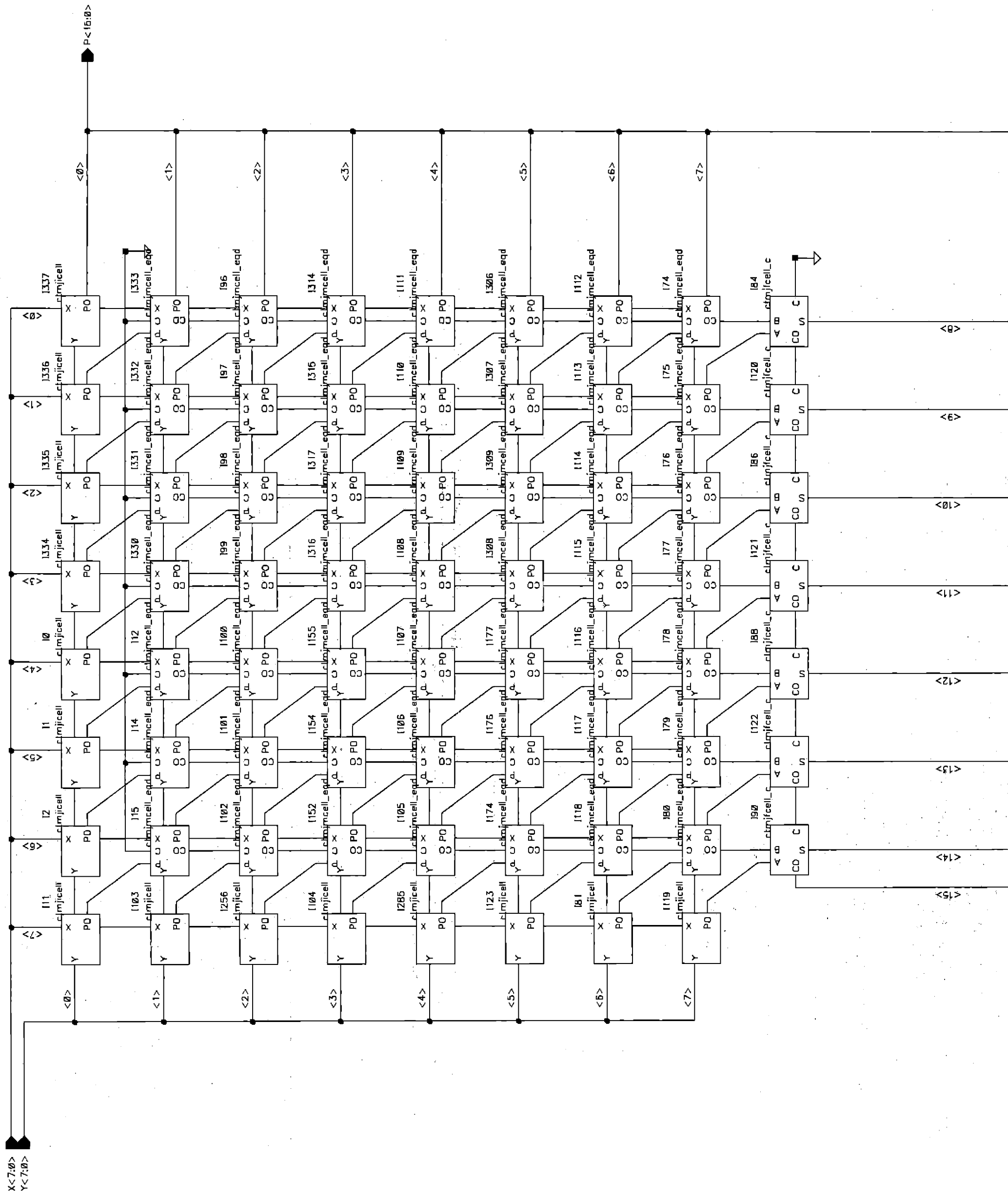


FIGURE A-14. Standard 8x8 multiplier architecture.

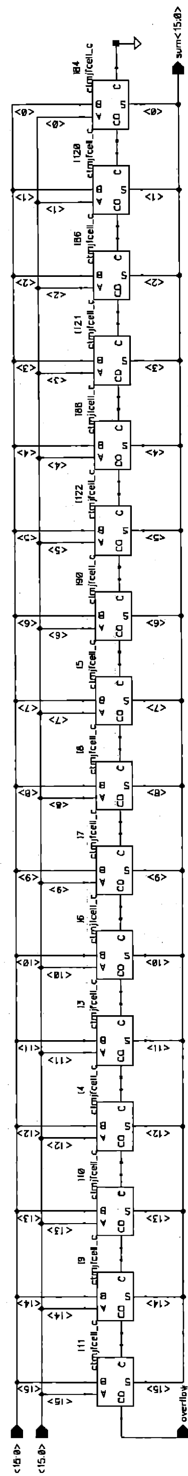


FIGURE A-15. Standard 24 bit adder.

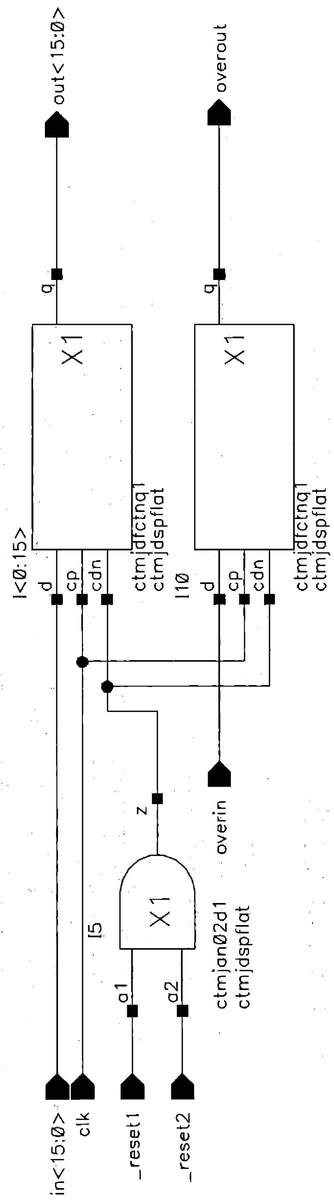


FIGURE A-16. Standard register for accumulator.

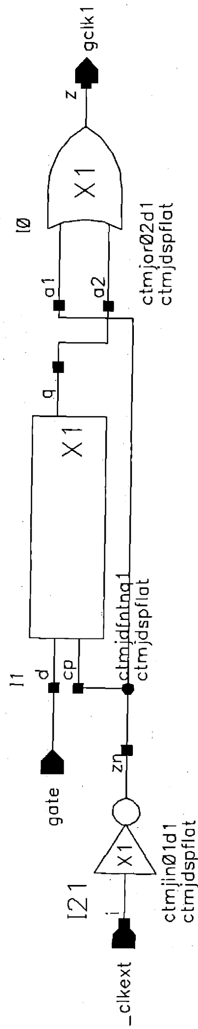


FIGURE A-17. Clock gating block.

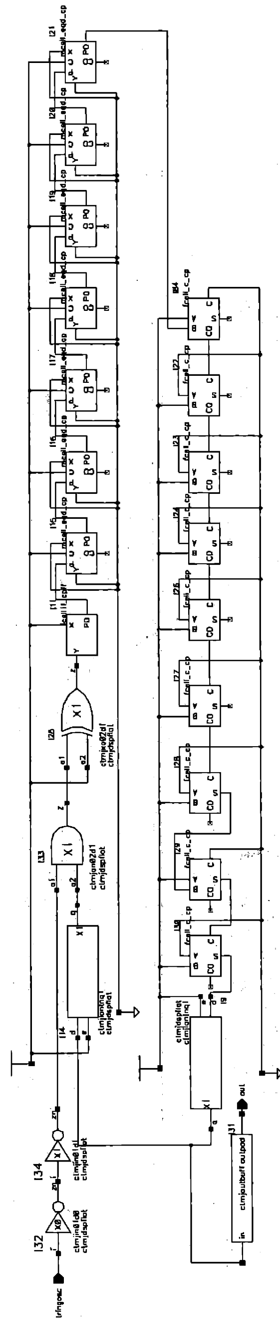


FIGURE A-18. Ring oscillator based on MAC critical path using worst case vector transitions.