Design Considerations for Nano-Electromechanical Relay Circuits

Matthew Spencer

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2015-195
http://www.eecs.berkeley.edu/Pubs/TechRpts/2015/EECS-2015-195.html

August 14, 2015

Design Considerations for Nano-Electromechanical Relay Circuits

by

Matthew Edmund Spencer

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Engineering - Electrical Engineering and Computer Sciences

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Elad Alon, Chair
Professor Tsu-Jae King Liu
Professor Liwei Lin

Summer 2015

Design Considerations for Nano-Electromechanical Relay Circuits

Abstract

Design Considerations for Nano-Electromechanical Relay Circuits

by

Matthew Edmund Spencer

Doctor of Philosophy in Engineering - Electrical Engineering and Computer Sciences

University of California, Berkeley

Professor Elad Alon, Chair

Complementary metal oxide semiconductor (CMOS) technology has a minimum energy per operation, and that limitation is one of the myriad hurdles CMOS faces as it reaches small scales. This minimum energy is set by the balance between leakage energy and dynamic energy in subthreshold CMOS circuits, and sets floors on the achievable energy of digital units. A new, post-CMOS device with a sharper subthreshold slope than CMOS would be able to sidestep this minimum energy constraint.

A candidate device called a nano-electromechancial (NEM) relay has recently emerged. NEM relays are small, integrated, capacitively-actuated, mechanical switches. The devices have demonstrated extremely high subthreshold slopes: ten orders of magnitude over a millivolt of swing. However, in the same lithographic process they are twenty times larger than a minimum sized CMOS device, their gate capacitance is ten times that of a minimum sized CMOS device, and their mechanical motion is an order of magnitude slower than a CMOS inverter. Can NEM relays improve digital systems even with these drawbacks?

With proper circuit design, simulations say "yes". This dissertation examines three of the critical components of digital systems – logic, timing, and memory – and proposes NEM circuits which mitigate the weaknesses of the technology while achieving design goals. Simulations show that optimized relay logic, which arranges for all of the slow movement of relays to happen at the same time, can achieve an improvement of 10x in energy-per-operation below the CMOS minimum energy point at a penalty of 10x in delay and 3x in area. This logic style is experimentally demonstrated. In addition, relay latch based timing with staticization in the feedback path is simulated, which results in a working relay pipeline with zero mechanical delays of timing overhead. Finally, a new device called NEMory is proposed to build dense, non-volatile, mechanical memory. A hybrid NEMory/CMOS array is simulated, and its performance is compared to other memory solutions. The NEMory density is higher than any non-volatile memory except for multi-level cell, off-chip Flash, and its read and write energy are lower than any other non-volatile technology. Finally, the scaling and process limits of realizing mechanical devices are discussed in the context of future work.

To my parents, Selden and Jean Spencer.

Contents

List of Figures

List of Tables

Acknowledgments

Many hands other than mine have touched this document directly or indirectly, and I owe a huge debt of gratitude which I hope to tabulate here. That gratitude has to start with my advisor, Elad Alon, who has been everything that a person could ask for in a mentor. Elad is brilliant, insightful, a clear communicator, a genial negotiator and a hawk for his students interests, all of which made my path through graduate school easier. Even better, I get to take a little of his perspective into my future career: when I am stumped I am haunted by the question "What would Elad ask me about this?" In those moments my internal Elad simulator helps me through.

Other professors have been terrific mentors as well. Dejan Markovic and Vladimir Stojanovic have been very close collaborators on the circuit side of the NEM relay project, and they have provided many astute insights throughout our years of collaboration. Tsu-Jae King-Liu has led the device development efforts that made the project possible, and her meticulous insight on device design, failure analysis, and communication have made it even more successful. Liwei Lin has been a valuable member of my committee. Jaijeet Roychowdhury helped me lay the foundations of the simulator models that undergird this work, and he has been very supportive of my efforts to publicize the models. I enjoyed my brief time working with Vivek Subramanian, Michel Maharbiz, and Kris Pister on the unrelated, awesome, VAPR project.

Many students have been in the trenches with me building and testing relay circuits. Hossein Fariborzi and Chengcheng Wang have been my other two musketeers, and we have fought it out with relay chips year in and year out. I'm glad to have had them at my side throughout the process: I still vividly remember our music selections from the CLICKR1 testing process. I learned most of what I know about CAD tools and circuit debugging from Fred Chen, who was a wonderful trailblazer for us. Sumit Dutta, Kevin Dwan, Abhinav Gupta, Patrick Kwong, and Pierce Oeflein have provided tremendous help along the way.

However, the circuit musketeers would have nothing to work on were it not for the valiant, backbreaking efforts of students on the device side of the NEM project. I am in awe of their collective device processing ability, and I am tremendously grateful to all of them; Anderson Kam, Rhesa Nathanael, Jaeseok Jeon, Vincent Pott, Filip Chen, Tim Chen, Jack Yaung, and Louis Hutin made this work possible.

This project has brought me into fruitful contact with excellent industrial partners and I've invariably enjoyed my time working with them. Ian Young and Uygar Avci were wonderful mentors at Intel, and every meeting of the minds I had while working with them in the Components Research Group was fascinating and inspiring. Lance Barron was an excellent collaborator, guide to TI and friend: I enjoyed working with him immensely. Mike Mignardi, Rick Oden, and all of the other DLP suspects also made a very positive impact. John Crossley, Matt Weiner, Chintan Thakkar, Alberto Puggelli, Douglas Adams and Jonathan Spaulding helped bridge my transition from Berkeley to these industrial sites, provided hours of insightful discussion, and have remained steadfast friends.

Many EECS and BWRC staff members have enabled me to stay focused on this work. Their number includes the inestimable Ruth Gjerde, Dana Jantz, and Shirley Salanio, who have shepherded many other students and me through all of the hurdles in the EECS program. I enjoyed working with the instructional lab staff, particularly Winthrop Wilson and Ferenc Kovacs, during my stints teaching labs. The BWRC staff: Tom Boot, Olivia Nolan, Leslie Nishiyama, Sarah Jordan , Grace Lovio, Bira Coelho, Fred Burghardt, Brian Richards, James Dunn, Gary Kelson, Dave Allstot, and Deirdre McAuliffe Bauer kept the center running like a well oiled machine and provided a wonderful space in which to work. I reserve special thanks to the IT, lab and CAD staff for keeping tools running: I couldn't do my work if they didn't keep the lights on. I've also had wonderful, if limited, interactions with the Microlab staff, and they all deserve kudos.

I've been supported by a variety of sources including the E3S, C2S2, BWRC, an Intel Fellowship and NSF Infrastructure Grant 040237.

I pursued other projects at the same time as my graduate work. Of those projects, my work with the EE Outreach @ Berkeley organization is my proudest extracurricular accomplishment. I'm grateful to have worked with tons of gifted volunteers and good friends through the group. A brief, and doubtless woefully incomplete, list of my collaborators is Reinaldo Vega, Justin Valley, Zach Jacobsen, Sam Burden, Rachel Nancollas, Adair Gerke, Dan Calderone, Sam Coogan, Ben Keller, Achintya Madduri, Aaron Bestick, Kelvin So, Amy Whitcombe, and Dorsa Sadigh. Thanks to them and everyone else who helped along the way.

I'm also proud of my work with the EEGSA, and I have many people to thank for setting me up, taking the baton from me, or helping along the way. I can't name everyone who helped with the organization, which I regret, but I can summarize some people with whom I worked especially closely: Gireeja Ranade, Zach Jacobsen, Rehan Kapadia, Adair Gerke, Ben Keller, Bobby Schneider, Kevin Weekly, Clair Lochner, Samarth Bhargava, Ben Keller, Stephen Twigg, Stephan Adams, Vasuki Narasimhaswamy.

I've been accompanied on the trip through graduate school by a wealth of wonderful friends who have helped make the journey to this dissertation joyful. They're named in big groups here for fear of eating many pages and still forgetting crucial names. I'm thankful to the all generations of Keith House residents, my many friends from Cal Wushu (especially Darren and Tami as my first, encouraging coaches), Team Moustache and other curlers and curling groupies, salseros and salseras, EECS board game groups, Fort Awesome, all of my friends from Elad's research group, the Lion Semiconductor team, the control theory happy hour , my integrated circuits first year cohort, South Bay and Stanford suspects, MIT ex-pats, and many others.

Maren has been a cornerstone in late chapters of my thesis, and my family has been the soil in which I took root. I love them tremendously.

Chapter 1

Introduction

A dissertation on mechanical computing may seem behind the times: mechanical computers went out of style in the 1950's – ENIAC was one of the last great examples of the technology – because the introduction of transistors shifted the underlying physics of computation. Information could be stored as clumps of electrons controlled with electric fields rather than stress in a physical spring controlled with power-hungry magnetic fields. The rewards of this shift in the underlying physics of switching devices were immense: the dawn of the transistor and Moore's law have resulted in enormous societal good and social change. However, the physics of transistors have shifted again over the long life of Moore's law, and the looming issues that face today's tiny transistors could allow mechanical computing devices to have another day in the sun.

There are many challenges facing transistors, and some quick examples include random dopant fluctuation and shrinking gate dimensions, which have lead to increasing variability in planar bulk devices and various types of tunnelling leakage current respectively. One issue of particular import is the increasing relevance of the complementary metal oxide semiconductor (CMOS) minimum energy point: each operation con be optimized to consume a minimum amount of energy by balancing leakage and dynamic energy components. If devices operate at the minimum energy point, then clearly energy per operation can't be reduced even by sacrificing throughput, which means that architectural techniques like parallelism fail to provide any energy consumption benefits.

Many techniques exist to reduce the power consumption of systems even in the face of this issue: power gating blocks when they're not in use, using specialized blocks that can perform some operations at lower energy, and using heterogeneous cores for different work loads. Even so, fundamentally shifting the minimum energy point would result in energy-per-operation gains. Unfortunately, the energy per operation is set by physics that are fundamental to the operation of a transistor: dynamic energy is a byproduct of putting charge on a capacitance and drain to source leakage occurs in any system that modulates the height of an energy barrier to gate electrons. Naturally, there is wide investigation of different ways to control the flow of electrons which will in turn reduce leakage. For example, tunnel field effect transistors (FETs) attempt to modulate the alignment of bands in order

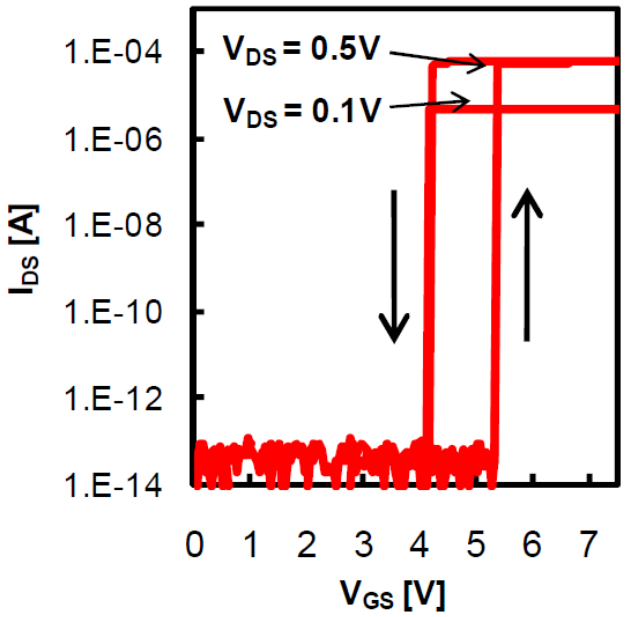


Figure 1.1: Current vs. voltage in a NEM relay, exhibiting very sharp slope during on and off transitions. Replicated from [1].

to determine whether tunnelling is possible.

A more primal way to control tunnelling is to modulate the width of a tunnelling barrier by physically making it larger or smaller. Moving the electrodes on the sides of the barrier material achieves this. Further, making contact between those electrodes provides an even greater boost to the on-off ratio of the current, so mechanical switches built at the micro and nano scale show promise in creating a very low leakage switch. This promise has been confirmed by a range of prototypes; switches with less than femtoamperes of current have been demonstrated [1]. These devices have dramatic subthreshold slopes: their current-voltage (I-V) characteristics show changes of ten decades of current in millivolts of swing. An example of this transition is shown in Figure 1.1. With such sharp *on-off* characteristics these seem like natural candidates to replace CMOS switches and reduce energy-per-operation.

Of course, the I-V curve is only part of the story. Mechanical devices, even at the micro scale, switch very slowly compared to electrical devices. This mechanical delay is approximately one-thousand times longer than the electrical delay in an equivalent technology node. Further, each device is significantly larger than a minimum sized CMOS device, consuming about twenty times the area, and each device has larger gate capacitance than a minimum sized device, about ten times the capacitance. Clearly, CMOS circuits with the devices replaced by mechanical devices would be very large, very slow and very hungry for dynamic power unless the supply voltage was aggressively scaled.

However, these doom and gloom predictions assume naive circuit designs. Co-optimizing the circuits and the mechanical devices that comprise them can result in significantly im-

proved performance, reduced device count and accordingly lower area and energy consumption. This dissertation examines the circuit/device co-optimization process for several canonical classes of digital circuits: logic, timing circuits and memory. The remainder of this chapter takes a closer look at the cause of the CMOS minimum energy point, the implications that has for designing digital systems, and the device technologies which attempt to alleviate the minimum energy problem. Chapter 2 introduces typical relay devices, a model appropriate for simulating them, and circuit design techniques suitable for building relay logic. These techniques result in logic blocks which operate in a single mechanical delay. The chapter includes experimental demonstrations of the circuits built in the resulting logic style. Chapter 3 examines timing circuits suitable for relay logic, observing that standard flip-flops would triple the delay of a system. The chpater proposes a latch based relay timing circuit which has zero mechanical delays of timing overhead and which is verified by simulations. Chapter 4 addresses the poor density of mechanical memory by proposing a new, non-volatile, high-density memory device, analyzing its performance and verifying the analysis with simulations. It includes a comparison of many modern non-volatile memories. Chapter 5 concludes the dissertation with ruminations on the fundamental limits of NEMory and the relation of those limits to future work.

1.1 CMOS and the Minimum Energy per Operation

CMOS circuits have a well-defined minimum energy per operation [2] which is defined by optimally balancing the leakage and dynamic energies consumed by a digital block. Specifically, [2] shows the energy consumed during each transition of a digital circuit can be represented as:

$$\begin{aligned} E_{op} &= E_{dynamic} + E_{leak} && (1.1)\\ &= C_{block}V_{dd}^2 + V_{dd}I_{leak}t_{op} && (1.2)\\ &= C_{block}V_{dd}^2 + V_{dd}(W_{block}I_0e^{(-V_T)/n\phi_{th}})\left(\frac{L_DC_{inv}V_{dd}}{I_0e^{(V_{dd}-V_T)/n\phi_{th}}}\right) && (1.3)\\ &= V_{dd}^2(C_{block} + W_{block}L_DC_{inv}e^{-V_{dd}/n\phi_{th}}) && (1.4) \end{aligned}$$

where $E_{dynamic}$ is the component of energy lost to charging capacitors, E_{leak} is the component of energy lost to leakage current, C_{block} is a generalized switching capacitance for the block each cycle which accounts for glitching and activity, V_{dd} is the supply voltage, I_{leak} is the leakage current through the block, t_{op} is the amount of time required to perform an an operation, I_0 is the dark current of the leaking CMOS diode, V_T is the device's threshold voltage, n is the device non-ideality factor, $\phi_{th} = kT/q$ is the thermal voltage, W_{block} is an averaged "leakage width" which represents the total transistor width in the circuit weighted for the states which reduces leakage, C_{inv} is the output capacitance of an inverter, and L_D is the delay of the block measured in inverter delays: the logic depth. This model assumes that subthreshold current is dominated by gate-modulated drain-to-source leakage.

This energy is mostly controlled by the supply voltage. It is independent of the threshold voltage because any change in V_T will decrease the leakage current by the same amount that it increases the delay, though a performance constraint specifies a value for V_T. The supply voltage is a particularly interesting design knob because there is clearly an optimum value for it. The V_{dd}^2 term increases with V_{dd} and it is multiplied by the $e^{-V_{dd}/nV_{th}}$ term which decreases with V_{dd}, implying that some V_{dd} will result in the smallest possible value. Differentiating E_{op} with respect to V_{dd} and equating to zero gives that value:

$$\frac{\partial E_{op}}{\partial V_{dd}} = 2V_{dd}C_{block} + W_{block}L_D C_{inv}\left(2V_{dd}e^{-V_{dd}/n\phi_{th}} - \frac{V_{dd}^2}{n\phi_{th}}e^{-V_{dd}/n\phi_{th}}\right) = 0 \tag{1.5}$$

$$2 + \frac{W_{block}L_D C_{inv}}{C_{block}}\left(2 - \frac{V_{dd}}{n\phi_{th}}\right)e^{-V_{dd}/n\phi_{th}} = 0 \tag{1.6}$$

$$\left(2 - \frac{V_{dd}}{n\phi_{th}}\right)e^{2-V_{dd}/n\phi_{th}} = \frac{-2C_{block}e^2}{W_{block}L_D C_{inv}} \tag{1.7}$$

$$V_{dd,opt} = n\phi_{th}\left(2 - \text{lambertW}\left(\frac{-2C_{block}e^2}{W_{block}L_D C_{inv}}\right)\right). \tag{1.8}$$

If V_{dd} is set to the optimum value, then the only factors which affect the total energy are C_{block}, W_{block}, L_D, C_{inv} and n. Designers have little control over these factors: n and C_{inv} are set by the technology, and C_{block}, W_{block} and L_D are mostly dependent on the function being implemented. Certainly, care should be taken to minimize C_{block}, W_{block} and L_D, but beyond ensuring that the digital block uses and the proper circuit architecture and layout techniques that care has rapidly diminishing returns.

Which is a frightening prospect: this means that at a fixed technology node there is a minimum energy required for a given function and no architectural tricks can improve that value. Scaling to a smaller technology node doesn't affect L_D for an for a given architecture, but it can reduce C_{block} and W_{block} by the scaling factor S. Unfortunately, the density of transistors on the chip will increase at a rate of S^2, so the power per unit area will increase. Obviously, power dissipated per unit area has a limit before damaging the chip, so does this imply the end of Moore's law?

The remaining variables that control energy, n and C_{inv} provide little evidence otherwise. C_{inv} is tightly liked to the drive current of the transistor and increases with S, and n has a physical limit of one: the transistor can't do better than perfect electrostatic control of the channel. Emphasizing that last point, the 60mV per decade limit on subthreshold slope set by the Boltzman constant is a fundamental limit on the energy per operation achievable by subthreshold CMOS transistors.

1.2 Beyond Boltzman

That statement leaves some wiggle room: CMOS transistors are constrained by the Boltzman limit on subthreshold slope, but could other devices turn on or off more quickly? An investigation of transistor physics can shed some light on the question.

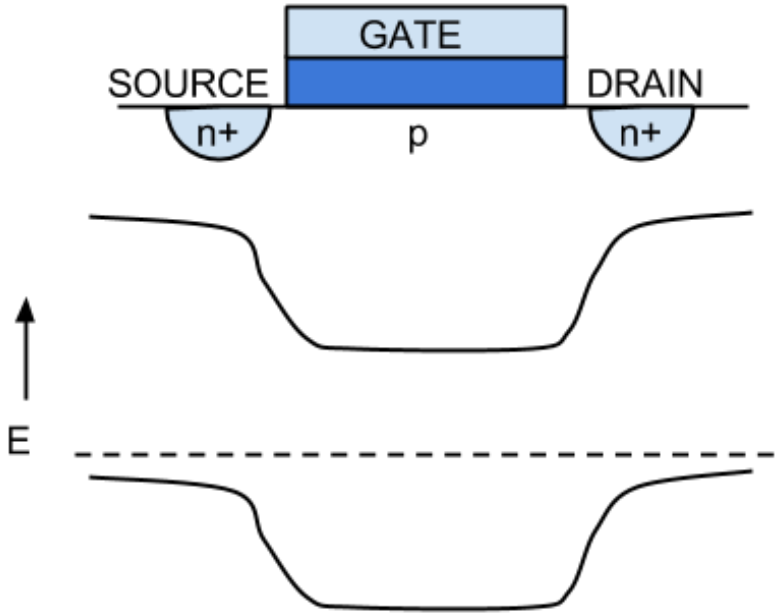


Figure 1.2: Drawing and band diagram of the parasitic, leakage BJT inside of a NMOS device.

Subthreshold current is passed through a parasitic bipolar junction transistor (BJT) passing between the drain and source of a MOSFET, so the band diagram for subthreshold leakage can be described using the same model as a BJT as pictured in Figure 1.2. This band diagram is symmetric because the source and drain are doped symmetrically, which is unlike standard BJTs. Current has an exponential relationship to the voltage difference between the gate and source because the height of the band gap is modulated directly by the voltages applied to those terminals and thermionic current emission is exponentially sensitive to the height of an energy barrier.

The non-ideality factor describes the MOS gate's electrostatic control of the voltage at the non-inverted MOS channel which serves as the base of the parasitic BJT. In a planar bulk MOSFET the gate voltage influences the channel through a planar capacitive divider, so the non-ideality factor is described as $n = 1 + C_{bulk}/C_{ox}$, where C_{bulk} is the channel to bulk capacitance per unit width and C_{ox} is the gate to channel capacitance per unit width. One technological thrust to reduce the energy per operation of CMOS systems has been to target the non-ideality factor and exert greater control over the channel. Both fully-depleted-semiconductor-on-insulator (FDSOI) and finFET (either bulk or SOI) technologies

result in the gate capacitance exerting more control over the channel, either by increasing C_{ox} (finFETs) or decreasing C_{bulk} (FDSOI). These have had impressive results, with subthreshold slopes approaching 60mV per decade [3, 4].

Adjusting V_T doesn't change the exponential relationship between the gate-to-source voltage, V_{gs}, and current, it just changes the barrier height between the drain and source. This can reduce leakage at the cost of subthreshold "on"-current, but can't affect the total energy consumption as discussed above. The threshold voltage is an ineffective knob because subthreshold conduction relies on thermionic emission as its switching mechanism. A different switching mechanism could result in a steeper subthreshold slope and thus reduce the minimum energy per operation of a technology. This philosophy has led to investigations of a variety of novel devices, and tunnel FETs [5] are among the most mature of them.

A tunnel FET relies on a complex band structure, a cartoon example of which is shown in Figure 1.3, to achieve current switching. A large intrinsic region between the drain and source suppresses leakage because injected carriers tend to recombine, so there is little conduction across the structure without an external voltage applied. When one is applied the valence band of the drain aligns with the conduction band of the channel, which enables tunneling between them. The width of the tunneling barrier is just set by the width of the band gap in the channel, and it can be narrow enough to promise significant current density [6].

Even so, demonstrated tunnel FETs struggle to demonstrate both significant on-current and high subthreshold slopes at the same time. In particular, defects in the channel tend to create mid band traps that greatly enhance off-state tunnelling [7]. The highest demonstrated subthreshold slope is 20mV, but that was at a current density of only $0.1\mu A/\mu m$ [8]. More aggressive tunnel FETs have demonstrated $1mA/\mu m$ at a subthreshold swing of 60mV/decade [9], but this doesn't represent a significant improvement over state of the art finFETs. In short, the subthreshold slope of tunnel FETs isn't improved relative to CMOS while the on-current is degraded. Possibly as a consequence, tunnel FETs have not seen large scale circuit demonstrations thus far.

Turning to other physical domains holds some promise for switching technologies. For instance, magnetic logic gates have been demonstrated to have very low energy per switching operation [10]. These devices store state in the orientation of magnetic domains rather than in the presence or absence of charge, and this magnetic spin by modulates the resistance of the devices with the giant magnetoresistive effect (GMR). Interlocking domain and spin injection sites can result in assemblies which propagate logic through magnetism with relatively little electrical work. Ultimately, however, the devices switch because current is applied to them. Relatively low (μA) currents are required to switch their state, but the devices are resistive at both their inputs and their outputs which necessitates aggressive architectural tricks to minimize power consumption [11]. Comparisons to CMOS reveal that the devices are not energy or delay competitive with CMOS gates, consuming pJ of energy for ns delays [12] at the gate level. Both of those metrics are an order of magnitude higher than their CMOS counterparts.

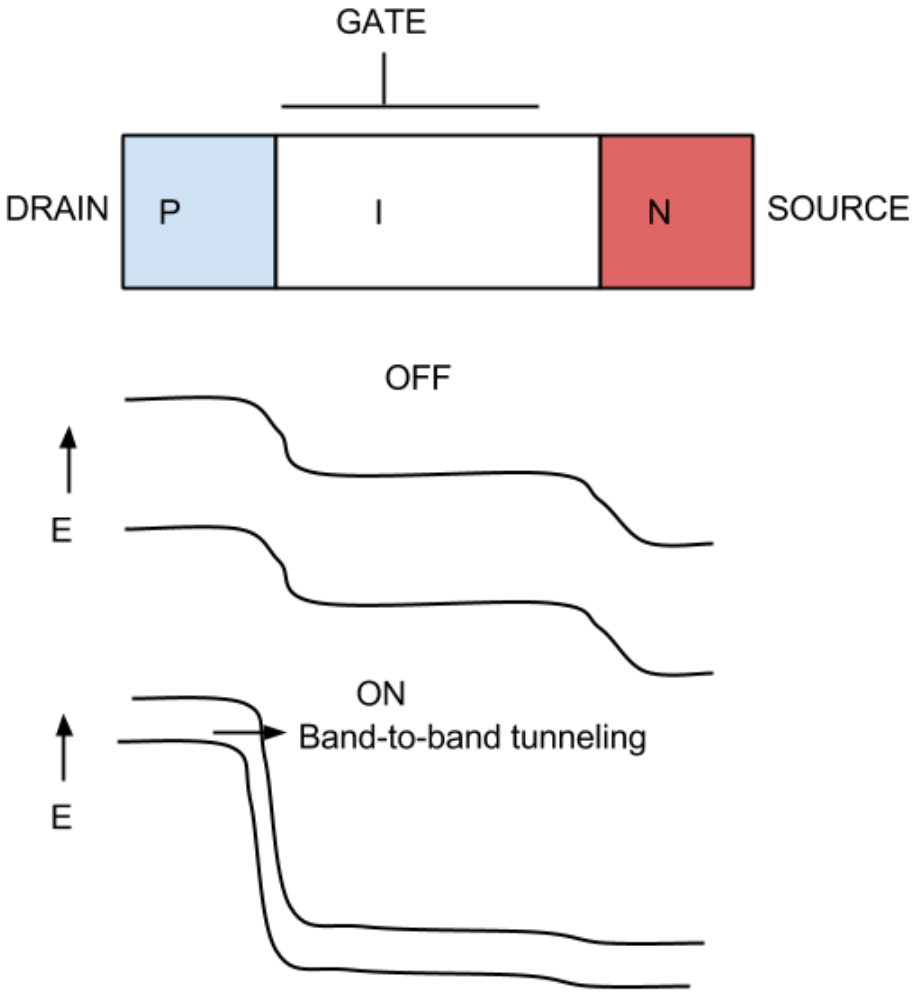


Figure 1.3: Drawing and band diagram of a TFET.

1.3 Electromechanical Devices to the Rescue?

Mechanical motion of an electrical conductor is another way to cause switching without resorting to a thermionic energy barrier. Two pieces of metal that are not in contact are separated by an energy barrier with the relatively tall height of the metal's work function and the relatively large spatial dimension of the gap size. This results in very little tunnelling current. In contact, the pieces of metal have an ohmic connection and can conduct very high levels of current.

Micro-electromechanical systems (MEMS) have been extensively studied, which has resulted in ample proof both that moving, air-separated structures can be integrated on-chip and actuated electrostatically[13]. But traditional MEMS devices are hundreds of microns on a side and require tens of volts to actuate. Both of those features – the large area and the high operating voltages – make MEMS devices unsuitable as digital very large system integration (VLSI) switches.

However, tantalizing demonstrations of logic suitable MEMS have changed those characteristics in recent years [1]. These devices demonstrate low switching voltages as a result of their small gap sizes, and high reliability because of their hard contacts. These virtues come at the expense of isolation and contact resistance. An example of the I-V characteristics of these mechanical switches, referred to as micro-electromechanical (MEM) relays, is pictured in Figure 1.1.

The I-V characteristic is hysteretic because of surface forces and the non-linear nature of the electrostatic force. This hysteresis sets the minimum swing that is required to actuate the MEM relay. Even in these early prototypes, the hysteresis window is one volt, and the change in current is ten orders of magnitude. That corresponds to a subthreshold swing of 100mV per decade, which is tantalizingly close to planar bulk CMOS even before process optimization and scaling. However, these switches are large and slow; the switches consume 450 μm^2 of die area and 300ns of mechanical actuation time.

Scaling can help to mitigate these problems. The virtue of scaling is that MEMS switches are field effect devices: electrostatic actuation depends on the density of electric field between moving and fixed electrodes. Thus, traditional Dennard scaling can reduce their operating voltage, area, delay and power consumption of MEM relays in much the same way as it has for CMOS transistors [14]. Adders composed of these scaled MEMS devices were examined in [15] and found to be competitive with CMOS in terms of their energy and delay performance: hypothetical cantilever style relay devices could achieve a 10x improvement over the CMOS minimum energy point for 32 bit Sklansky adders at a 10x delay penalty in equal die area.

This is an impressive result for relay based logic, and it is largely due to the circuit design used to build the adders. Instead of making a series of small gates, each of which incurs the long delay of a device moving, the circuits are composed of large gates which incur only a single mechanical delay. However, those circuits are not based on a real devices or experiments. Chapter 2 of this work examines the basic principles of relay-based logic and design for single-mechanical delay circuits, which are the same as the aforementioned paper, and puts those principles and designs to experimental test in a pair of demonstration

chips which show functionality of common logic circuits. Projections of the behavior of future devices are made based on the demonstrated relays, and an energy delay analysis is performed on adders made of the projected devices.

Relay logic alone isn't enough to build really large digital systems. Logic needs to be partitioned into feasibly sized chunks, which requires synchronization between different logic blocks. Traditional flip-flops can be implemented using relays, but flip-flops contain several back-to-back inverters internally, so a realy flip-flop would incur two additional mechanical delays on top of the delay of the logic block the flip-flop serves. Further, the master-slave arrangement of traditional flip-flops guarantees that one buffer drives another in order to operate the system. Timing circuits that are suitable for use in relay-based systems need to have no mechanical delays on the forward path, and thus the staticization buffer needs to be moved into their feedback paths. Chapter 3 proposes a relay-based latch with a staticization buffer in the feedback path and a timing scheme such that the latch incurs zero mechanical delays of overhead in a relay VLSI system. Of course, the system incurs an electrical delay because nothing is free, but the overhead is shown to be negligible in simulations of a relay based pipelined accumulator.

VLSI systems need memory in addition to logic and timing circuits. Fitting sufficient quantities of memory on to chips requires high memory density, and the large size of relays prevents the construction of high density memory. A standard static random access memory (SRAM) cell composed of relays would consume as much area as twenty CMOS SRAM cells. Chapter 4 asks how to improve the density of relay memory, and after touring through an experimental demonstration of three relay dynamic random access memory (DRAM), it proposes a new device referred to as NEMory. The NEMory device achieves high density and non-volatility because it is a clamp-clamp beam with actuation electrodes both above and below the moving flexure. The clamp-clamp beam structure can be designed for an easy array layout, and the two electrodes allow the beam to be held in place by Van der Waals forces to achieve non-volatility. Immunity to sneak paths is achieved by careful materials selection of the beam such that a Schottky diode contact forms at the contact point between electrodes. Models of this device, analytical calculation, and simulations of hybrid NEMory/CMOS arrays are used to benchmark the device's performance and compare it to state of the art memory technologies.

These chapters suggest that large VLSI systems can be built from electromechanical devices, and chapter 5 explores the future prospects of building those systems and extending Moore's law with mechanical devices.

Chapter 2

Design with Relays

This chapter will discuss a logic style for mechanical relays that reduces their delay and power. An electromechanical model suitable for design work will be described and then used to inform the development of the relay logic style. Then measurements of a test chip will illustrate the salient points of the model. Finally, a scaled model will be used to simulate a relay adder in this design style. The scaled relay adder will be compared to a CMOS adder.

2.1 Physical Structure of MEM Relays

Figure 2.1 shows a diagram and SEM image of a four terminal MEM relay device. The device consists of a movable Poly-SiGe gate structure suspended by folded flexures which act like springs. The bottom of the gate is covered with a layer of insulating Al_2O_3, and a strip of metal called the channel is attached to the bottom of the Al_2O_3 layer. The gate and channel have vertical deformations called dimples. The gate, gate oxide and channel are suspended above several metal electrodes, which are referred to as body, drain and source. The dimples align with the drain and source.

Different metals have been used to make the channel, drain, source and body electrodes in different iterations of the relay design. The electrodes were made of tungsten in early designs and ruthenium is later ones. Both Tungsten and Ruthenium were selected because of their high hardness: measurements and analysis suggest that 90nm relays with contacts made from these two metals can withstand 10^{15} on-off cycles [18]. However, ruthenium was used in later relay designs to improve the conductivity of the device over time when exposed to atmosphere. Both metals form an oxide on their surfaces when they are exposed to air, but tungsten forms an insulating oxide while Ruthenium forms a conductive oxide [19, 20]. This conductive oxide allows the device to operate stably in atmosphere over many cycles.

Figure 2.1 also illustrates the basic operation of the relay. Applying a voltage between the gate and the body creates an electrostatic force on the gate, causing it to move and to deform the folded flexures. When the voltage between the gate and the body is increased above a critical value called the pull-in voltage, V_{pi}, the gate moves as close as it can to the

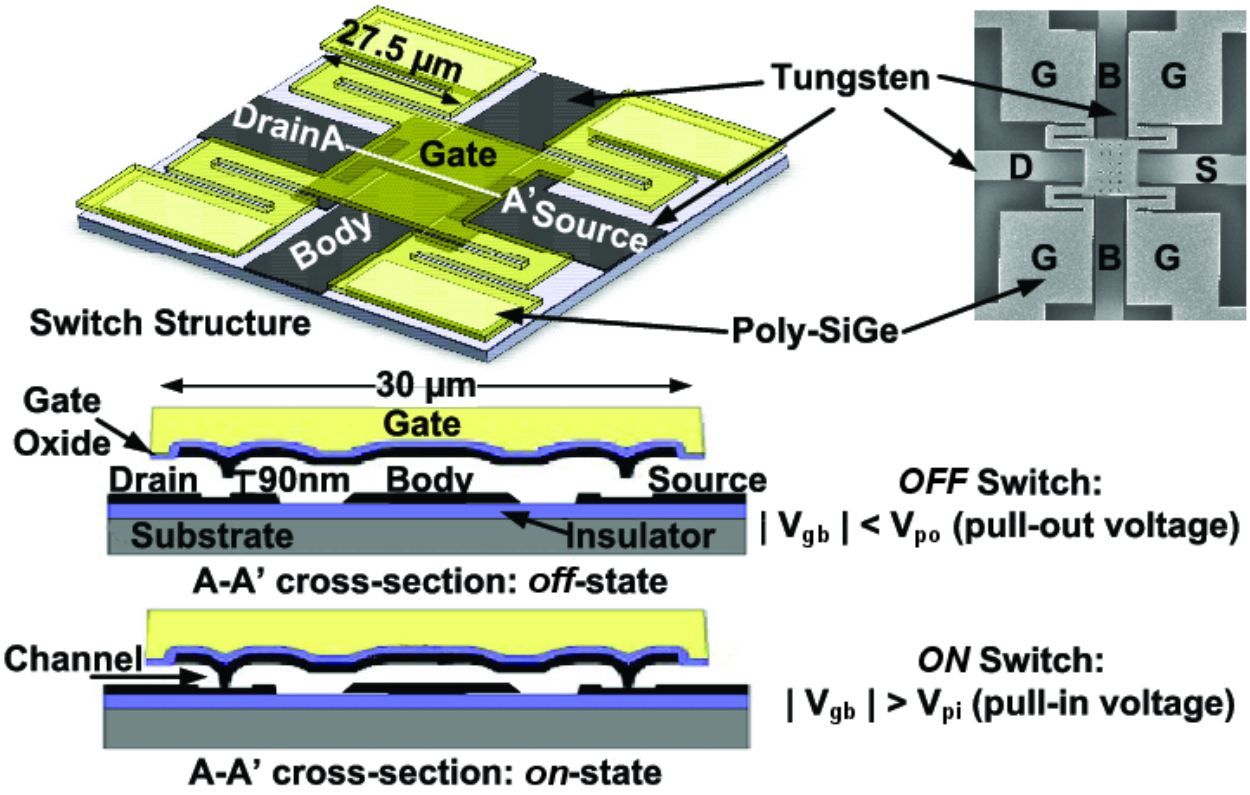


Figure 2.1: Diagram of a four terminal MEM relay [16, 17].

body. This motion is stopped when the dimples on the channel are brought into contact with the drain and source. This forms a conductive path from drain to source, and the relay is said to be in the *on* state when such a connection is made. When the gate-body voltage is decreased below a different value called the release voltage, V_{rl}, the gate is pulled back to its original position by the spring forces exerted by the deformed folded flexures. This breaks the contact between the channel and the drain and source so that there is no conduction between them. When the drain-source connection is broken the device is then in the *off* state.

The relay pictured in Figure 2.1 shows the channel drawn from the left side of the relay to the right. However, there's no reason not to have the relay create a shorter loop from the right side to the right side. Two such channels can be included on a single device, which allows for improved functionality: a single gate can control two separate switching paths like a DPST switch. Such an arrangement is pictured in Figure 2.2 and is referred to as six-terminal, or 6T, relay. The terminals are the *gate* and *body* which work the same as the relays discussed above, and two drain/source pairs referred to as *drain right / source right*, and *drain left / source left*. The standard relays discussed up to this point are referred to as 4T relays, but when this text refers to a relay it should be assumed to be 4T unless

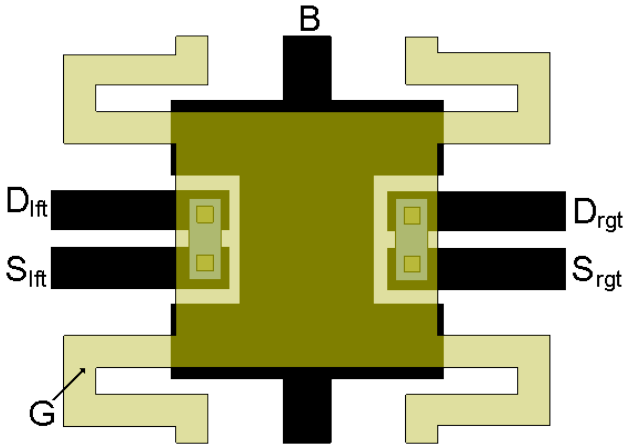


Figure 2.2: Diagram of a six terminal (6T) relay.

specifically denoted otherwise.

2.2 Mechanical Model of the Relay

Designing a relay based system requires that sufficient voltage and time are budgeted for the relays to move from state to state. Modeling the underlying physics of the relay provides insight into the their operating voltages and dynamics.

The dynamics of the motion of a relay can be described as a second order spring-mass-damper system being driven by a non-linear, electrostatic force [15, 1]:

$$m\ddot{x} = F_{elec}(x, V_{gb}) - b\dot{x} - kx \tag{2.1}$$

where x is the displacement of the gate, b is the damping coefficient of the gate's motion, k is the effective spring constant of the folded flexures, V_{gb} is the voltage between the gate and the body, and F_{elec} is the electrostatic force between the gate and the body.

Equation 2.1 glosses over modeling the forces which arise when the relay is in contact with the substrate. The force is effectively infinite since the gate can't pass through the drain and source electrodes, but modeling it as such is unsuitable for simulators. Instead, it is modeled as an exponential force which turns on sharply for large values of x. This choice of model clashes somewhat with classical atomic interaction theory since the Lennard-Jones potential integrated in three dimensions results in a ninth order polynomial for the repulsive surface, but using an exponential in its place has precedent in scientific literature referred to as the Buckingham approximation[21]. Exponentials are often handled with special delicacy by circuit simulators, which makes models converge more easily, so the Buckingham approximation is used here[22].

The folded flexures confine the motion of the relay so that it only moves up and down. This means that the relay's electrostatic behavior can be modeled as a pair of moving parallel

plates. Assuming a parallel plate model for the relay results in a well known expression for F_{elec}:

$$F_{elec} = \frac{\epsilon_0 A_{ov} V_{gb}^2}{2(g_0 - x)^2} \tag{2.2}$$

where ϵ_0 is the permittivity of free space, A_{ov} is the area of overlap between the gate and body, and g_0 is the distance between the gate and body when V_{gb} is zero – i.e. the native gap spacing.

If a voltage is applied to the relay the gate will be displaced. This results in the spring force, kx, increasing linearly and the electrical force F_{elec} increasing in an inverse quadratic fashion. If the displacement is large enough, the electrical force will be greater than the spring restoring force for all larger values of displacement. This force imbalance will cause the relay to rapidly displace downwards until it is stopped by contact between the drain, source and channel. This phenomenon is called pull-in and it happens at a well-defined voltage V_{pi} [13]:

$$V_{pi} = \sqrt{\frac{8}{27} \cdot \frac{k g_0^3}{\epsilon_0 A_{ov}}} \tag{2.3}$$

The relay will be held in the *on* state as long as the electrical force applied to it is greater than the spring force pulling it upward. Both the spring and electrical force increase during pull-in since the displacement of the gate increases, but the electrical force increases much more than the spring force because the inverse quadratic is a larger function when x is near g_0. This means that V_{gb} must be reduced to a value lower than V_{pi} in order for the device to release. That value is referred to as V_{rl}.

There is a delay while the relay moves to switch between the *off* and *on* states. The delay can be found by simulating Equation 2.1 or through various fits to the solution to that equation [14]. Applying a larger V_{gb} to the relay results in a shorter *off-on* delay because more electrical force is applied to the structure. Similarly, a structure with less mass or a weaker spring constant is easier to displace, so m and k affect how quickly the device switches. These effects can be summarized as

$$t_{pi} \propto \sqrt{\frac{m}{k}} \cdot \left(\frac{V_{pi}}{V_{gb}}\right) \tag{2.4}$$

where t_{pi} is the pull-in time of the relay. This represents a fit to a common solution range of the NEM relay [14].

Note that the *on-off* dynamics of the relay are usually much simpler than the *off-on* dynamics because there is usually no applied electrical force during the *on-off* transition. Accordingly, the gate displacement can be represented by a second order spring mass damper system with initial conditions. This means the time for the relay to return from a conducting state to its initial position, t_{rl} is longer than the time to move into contact initially. However, the conducting path between the drain and source electrodes of the relay is broken very soon after the relay begins moving (after traveling about 1 nm), so the relay exhibits a very short

time to break the connection, t_{off}, and "break-before-make" behavior even though t_{rl} is greater than t_{pi}.

A relay is a spring-mass-damper system, so after an *on-off* transition it will oscillate about its resting position if the quality factor of the device is high. If a relay is being actuated from *off-on* after an on-off transition then this ringing can affect the effective actuation gap. In the worst case, the relay would have zero displacement but the maximum possible upward velocity when the actuation signal arrived, which results in the mechanical delay being approximately doubled. Unexpectedly, this suggests that low Q relays are often significantly faster to operate than their high Q counterparts, though significant timing precautions are taken during the operation of high Q relays can result in reduced mechanical delays[23].

This physical model has interesting implications for the response of the relay to environmental disturbances and thermal noise. It is very difficult to get the relay to actuate by accident using only external acceleration because the inertial force of the gate, mg, is tiny compared to other forces in the system. Tor a device which is tens on the tens of microns scale, The small mass of the gate means that an acceleration of 20,000 g is required to balance out the spring force and actuate the relay. Devices which are smaller than tens of microns will require even greater inertia to actuate since their mass shrinks cubically with scale while the spring constant shrinks only linearly.

A device which is already in contact with the surface can only be shaken off of it if the sum of the spring force and the inertia is greater than the electrical force holding it in place. Again, examining a tens of microns device reveals that it is tremendously resistant to disturbance when closed: it would take nearly 600,000g to remove it from the surface.

Similarly, thermal energy (Brownian motion) doesn't cause significant displacements in the relay. Since the relay is confined to one degree of freedom, it has $k_BT/2$ Joules of thermal energy. This energy is stored as spring energy, and the relation between spring energy and displacement, $kx^2/2$, can be compared to the available thermal energy to find the RMS displacement of the structure: $\sqrt{k_BT/k}$. At room temperature, the σ of this displacement is only ≈ 8 pm for relays at the tens of microns scale, which is 0.005% of the actuation gap. At smaller scales thermal noise becomes more significant, but even at the 90 nm node it will only cause RMS displacement of 0.5% of the actuation gap.

2.3 Electrical Models of the Relay

A model of the resistances and capacitances in a relay is needed to discuss the power consumed by actuating a relay and to gain a complete picture of the delay involved in passing a signal through a relay. This subsection develops an electrical model suitable for those considerations.

Three physical phenomena contribute to the *on*-state resistance between the drain and the source: the resistance of the channel (R_{ch}), the resistance of the contact (R_{con}), and the resistance of any chemicals or oxides in the contacting surface (R_{surf}). There is also some resistance in the wires leading to and from the device (R_{trace}). R_{con} and R_{surf} are the largest

of these resistances. As seen in [24] the resistance of a metal-metal contact is a function of the material properties of the contacting metals and the applied pressure:

$$R_{con} = \frac{4\rho\lambda}{3A_r} \tag{2.5}$$

where ρ is the resistivity of the contacting material, λ is the mean free path of electrons in the contact material, and A_r is the effective contact area given by

$$A_r \approx \frac{F_{elec}(g_d)}{\xi H} \tag{2.6}$$

where H is the hardness of the material and ξ is the deformation coefficient. The deformation coefficient is 0.3 for all of the materials discussed in this document because all of the contacting materials are hard metals that make elastic contact.

R_{surf} lumps together the effects of any chemicals or oxides which have formed on the surface of the channel, drain, and source [19]. This can include deliberate surface coatings as in [17], friction polymers [20], parasitic oxides [19], and other substances. These chemical components contribute most of the resistance of the relay contact in large scale relays. Notably, parasitic oxides can easily result in additional resistances that range from hundreds of kΩ to hundreds of MΩ.

The electrical delay, the time it takes a signal to propagate across a relay after it is closed, and dynamic power consumption of the relay are dependent on its load capacitance. The relay has several intrinsic capacitances that can contribute to the capacitive loading, and wires contribute extrinsic capacitance Computing the capacitive contribution of wires has been thoroughly discussed elsewhere [25], so this discussion will focus on the device capacitances. Since the gate overlaps the drain, source, channel and body, there are capacitances between the gate and each of those terminals. Because a properly designed drain and source are small, the dominant contributions to the gate capacitance are the air capacitor formed between the gate and the body (C_{gb}) and the oxide capacitor formed between the gate and the channel (C_{gc}). Both of these capacitors are modeled as parallel plates:

$$C_{gb} = \frac{\epsilon_0 A_{ov}}{g_0 - x} \tag{2.7}$$

$$C_{gc} = \frac{\kappa_{gox}\epsilon_0 A_{ch}}{t_{gox}} \tag{2.8}$$

where κ_{gox} is the relative permittivity of the gate oxide, t_{gox} is the thickness of the gate oxide, and A_{ch} is the area of the overlap between the gate and the channel.

Other minor capacitances exist such as the gate-to-drain and gate-to-source capacitors (C_{gd} and C_{gs}). These capacitors are also modeled as parallel plates with the same separation and relative permittivity as C_{gb}, so the ratio of the gate-to-drain/source cap to the gate-to-channel capacitor is set by the ratio of the gate-to-drain/source overlap area and gate-to-body

overlap area. By design, this ratio is kept small so that voltages on the drain or source don't apply forces to the gate [26].

There are also capacitances between the drain/source and the channel (C_{cd} and C_{cs}), and these capacitances present a modeling challenge since the separation of the channel and the drain/souce goes to zero as the relay is actuated. Using a standard parallel plate model would result in infinite capacitance as the relay turns *on*. There are various ways to model this which trade off accuracy and stability in simulation, but the simplest and most expedient method is to add a small offset term to the separation so that the final capacitor model is:

$$C_{cd} = \frac{\epsilon_0 A_{con}}{g_d - x + \delta} \tag{2.9}$$

where A_{con} is the area of the channel contacts in which the dimples are formed and δ is the new separation offset term.

The electrical and mechanical models of the relay have been combined into a relay device model which has been verified by various experiments [1, 16, 17]. The model was implemented in Verilog-A to enable circuit design with relays. The performance of the computer model and the relays is well correlated [1]. Notably, this model captures the switching delay, pull-in and release voltages, and electrical delay of the devices. Figure 2.3 summarizes the model graphically.

2.4 Relay Circuits

Using relays as logic switches is different from using CMOS as logic switches because relays turn *on* and *off* based only on V_{gb} and because relays' delay is dominated by their mechanical motion. These two assertions will be explored below.

Static Relay Switching Characteristics

The electrical force on a relay is controlled by V_{gb}^2 according to Equation 2.2. This is significant, the body of each relay can be individually set to a different potential even when the relays are close together. Also, unlike CMOS transistors, the voltage between the gate and the source of a relay has no effect on its ability to drive current. Consequently, any relay can be designed to serve as an active low device (a "PMOS" that also has a strong pull-down) or as an active high device (a "NMOS" that also has a strong pull-up) by setting one of its gate/body terminals to the supply voltage or ground respectively. This is pictured in Figure 2.4. When a body terminal is attached to ground, the relay will turn *on* when the gate voltage is raised to a high value (greater than V_{pi}). Conversely, when a body terminal is connected to the supply voltage then the relay will turn on when the gate voltage is lowered below $V_{supply} - V_{pi}$. Note that if the gate voltage is allowed to travel outside of the supply voltage, then the relays can be actuated by sufficiently high or low values as seen in figure 2.5.

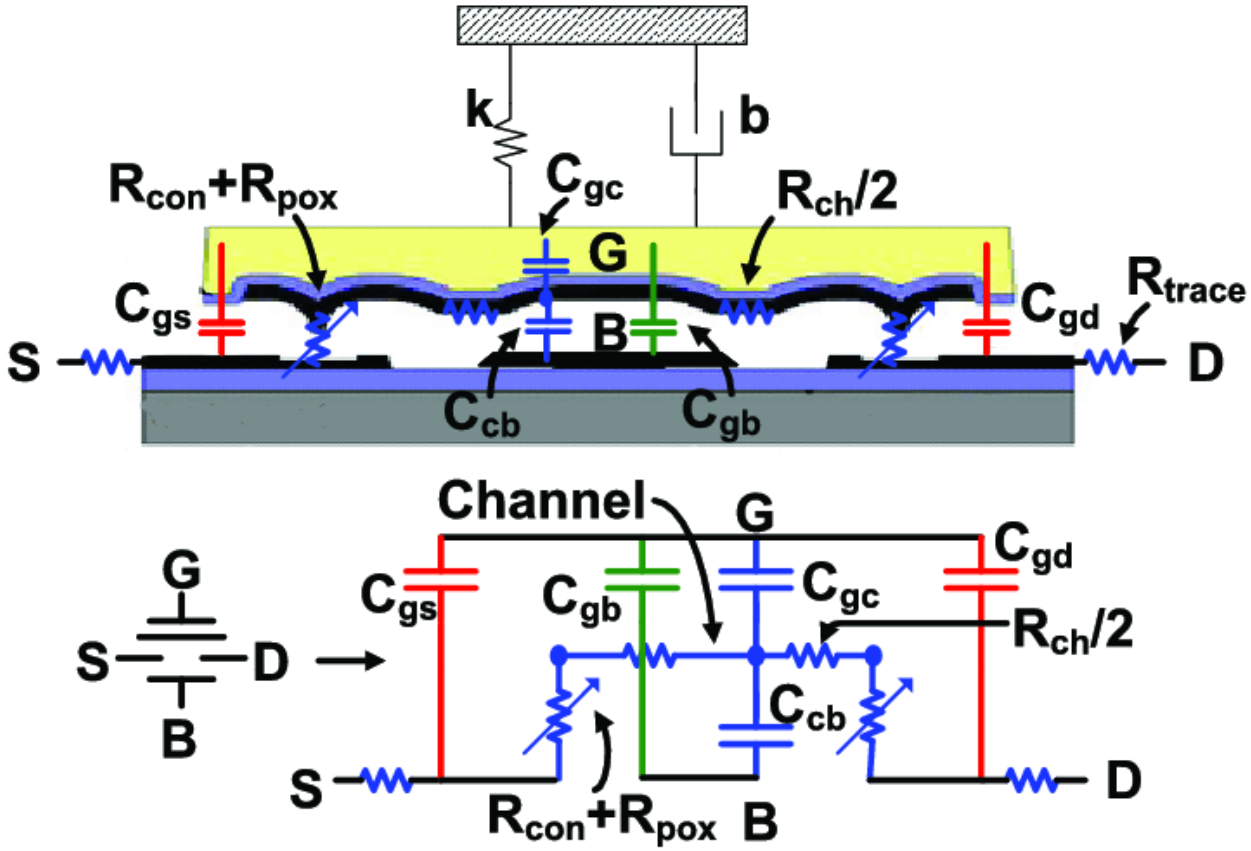


Figure 2.3: Schematic indicating the structure of the relay Verilog-A modlel [16, 17].

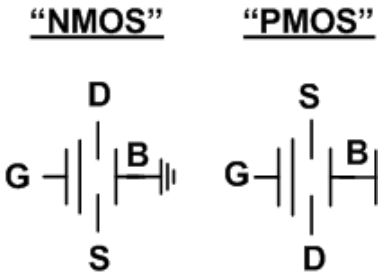


Figure 2.4: A relay can be configured to behave in the same way as a NMOS or PMOS transistor by attaching its body to the supply voltage or ground (assuming the supply voltage is larger than V_{pi}).

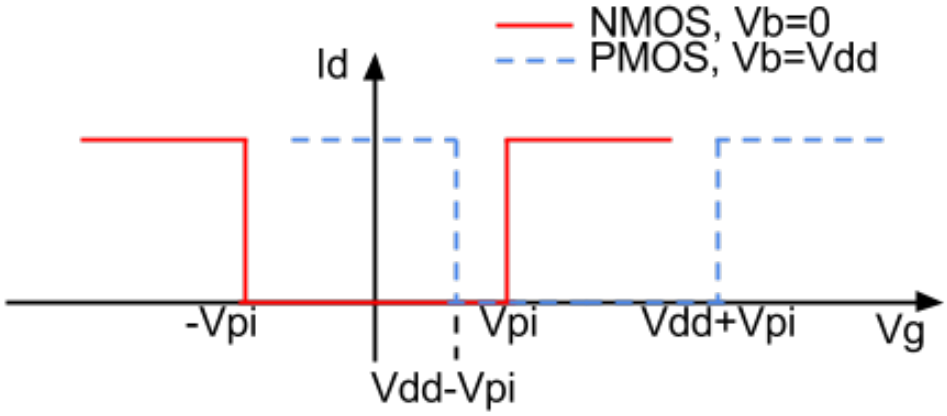
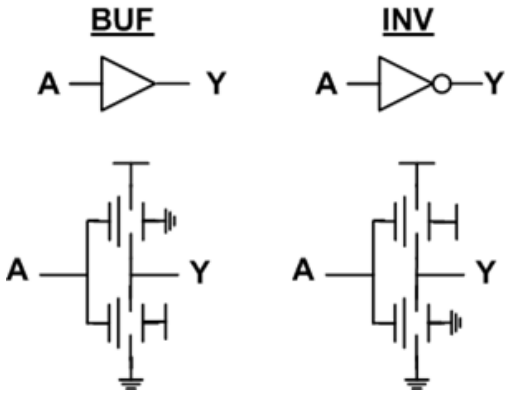
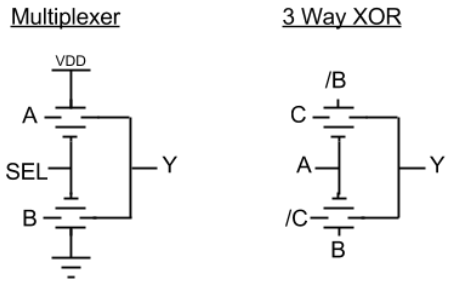


Figure 2.5: Relay $I_d - V_g$ curves are ambipolar; the state of a relay is determined by V_{gb}^2 and thus a relay can be shut with either sufficiently high or low voltage.



(a) Schematics of a relay buffer and a relay inverter.



(b) Schematics of a relay multiplexer and a relay XOR.

Figure 2.6: Schematics of relay logic gates which leverage the ambipolarity and V_{gs} insensitivity of relays to make more compact logic gates. Because relays can pull up or down and be active either high or low, it is possible to build non-inverting logic and native, highly integrated XORs.

Using this configurability, relays can be used to build non-inverting logic in single gates. For instance, it is possible to make both buffers and inverters out of relays as pictured in Figure 2.6a. Even better, it is easy to make a relay into a 3-way XOR because the relay can be actuated by a difference between the gate and body. Such an XOR is pictured in Figure 2.6b.

This suggests that relays can be used to build compact adders and multipliers because

those mathematical operations require many XORs. The advantages of using relays to construct XORs will be explored below. However, the dynamics of the relays will have a tremendous impact on the performance of any relay-based logic gates, so relay logic-dynamics are explored first.

Dynamic Relay Switching Characteristics

The calculation of delay in relay circuits is significantly different from CMOS circuits because of the mechanical delay that occurs while relays are moving from the *off* to *on* states. Though there is an electrical delay in relay circuits which is calculated the same way as the delay through CMOS gates, the physical motion of the relay across the actuation gap takes significantly longer than the electrical delay of charging up the relay's load capacitances in most cases, and the resistance and fanout of gates can be much larger in order to offset the penalty of incurring a mechanical delay.

This can be shown by examining an example. The RC delay of a relay is most significant compared to its mechanical delay at small scales where the electrical force is low (leading to high contact resistance) and the mechanical delay is small (because of high natural frequencies). A relay model for a 90 nm node, which is later used for energy delay comparisons, has been prepared to examine the relation between resistance and mechanical delay at scaled nodes. The parameters of the model are found in Table 2.1. The expected RC delay of a relay gate which has 100 series devices in a gate driving a fanout of 50 is

$$100R_{on} \cdot 50(C_{gc} + C_{gb}) + 100^2 R_{on} C_{gd/s} = 5000t_{inv} + 10000t_{int} = 12.5 - 17.5\text{ns}, \tag{2.10}$$

where $t_{inv} = R_{on}(C_{gc} + C_{gb} + 2C_{gd/s})$ is the characteristic RC delay of a relay inverter with a fanout of one and $t_{int} = R_{on}C_{gd/s}$ is the delay of a relay driving an internal node of a gate. In this extreme example, the RC delay is a factor of two smaller than the mechanical delay of an 90nm relay when the gate is driven near V_{pi} about equal to the delay at higher levels of gate overdrive.

This example can be generalized by rewriting the electrical delay as

$$t_{elec} = DFt_{inv} + D^2 t_{int} \tag{2.11}$$

where t_{elec} is the electrical delay of the gate, D is the number of relays in series between the source and the load (the depth of the gate), and F is the load capacitance measured in multiples of a unit relay inverter's capacitance (the fanout). t_{int} is very small in relays that are electrostatically sound: most of the actuation area should be devoted to C_{gb} with $C_{gd/s}$ determined by the minimum sized contacts. In the 90nm example relay t_{int} is an order of magnitude smaller than t_{inv}. That means its contribution remains below 10% of the total delay for values of D less than ten and below 50% for D less than 30.

D and F are the design parameters which specify the maximum size of gates. They are constrained by the requirement that

$$t_{elec} < t_{mech} \tag{2.12}$$

where t_{mech} is the amount of time it take the relay to move from the *off* state to the *on* state. This inequality is true because it maximizes the amount of electrical delay absorbed per mechanical delay [17]. Substituting in for t_{elec} shows

$$D^2 t_{int} + DFt_{inv} < t_{mech} \tag{2.13}$$

$$D^2 + D\frac{Ft_{inv}}{t_{int}} - \frac{t_{mech}}{t_{int}} < 0 \tag{2.14}$$

which has the solution

$$D = \frac{t_{inv}}{t_{int}}\left(\sqrt{\frac{t_{mech}}{t_{inv}} \cdot \frac{t_{int}}{t_{inv}} + \frac{F^2}{4}} - \frac{F}{2}\right). \tag{2.15}$$

The precise relationship between F and D is complicated, but insights can be extracted from it with simplifying approximations. t_{mech}/t_{inv} is approximately 1000, and t_{int}/t_{inv} is approximately 1/1000 so the first term under the radical is near to one for most technologies. As a result, the quantity inside the parentheses, which decreases as F increases, is going to be greater than 0.1 for $F < 5$ and greater than 0.01 for $F < 50$. This quantity is multiplied by t_{inv}/t_{int}, so an allowable depth of nearly 100 devices is common for moderate fanouts and 10 for very large fanouts. Broadly, this implies that very deep, complex gates are desirable to amortize the cost of expensive mechanical motion unless the electrical delay is going to be exceptionally expensive.

An oscillator fabricated in a 1μm technology was used to verify that the mechanical delay of a device is much longer than the electrical delay. A die shot of CLICKR1, the test chip containing the oscillator, appears in Figure 2.7. A schematic of the experiment and the results are included in Figure 2.8. The schematic shows a single relay attached to a pull-up resistor in order to form a pseudo-inverter. Unlike CMOS, a single relay inverter in feedback will oscillate because of the hysteresis introduced by the difference in the pull-in and release voltages. The pull-in voltage of the relay used in the oscillator was measured before the oscillator was started, then the oscillation period, rise time and fall time were measured. The load capacitance of the test setup could be estimated from the RC delay of the rise time, the *on* resistance of the relay could be measured from the RC delay of the fall time, and the mechanical delay could be measured by comparing the time at which the output crossed the pull-in voltage of the relay to the time at which the output started to fall. Notably, the mechanical delay of the relay was 34 μs, which is much longer than the falling time created by the relay driving the test setup's load capacitance. That falling time isn't indicative of the electrical delay of an individual relay because the oscillator was driving the load capacitance of the test setup. If the falling time were recalculated using the measured *on*-resistance of the relay and the expected capacitance of the relay, 15fF, then the electrical time constant would be 300ps.

The mechanical delay measured in the oscillator is a clear worst case: the relay is made in a large technology, and the overdrive is as low as possible during pull-in. The pull-in delay is very sensitive to changes in overdrive when the overdrive is low [14]. Later measurements

Parameter	CLICKR1	CLICKR2	CLICKR3	CLICKR6	90 nm model
Sources	[16, 17]	[17, 27]	[28, 29]	[20]	[17]
Relay Type	4T	4T	6T	6T	4T or 6T
Contact Material	Tungsten	Tungsten, ALD TiO_2	Tungsten	Ruthenium	Tungsten
		Layout Parameters			
Actuation Area Length [μm]	30	30	7.5	25	7.7
Actuation Area Width [μm]	14	30	7.5	20	0.6
Flexure Length [μm]	25	27.5	5.5	15	3.5
Flexure Width [μm]	5	5	1	2.25	0.1
Channel Length [μm]	25	29	1	3	0.7
Channel Width [μm]	2	2	0.5	1	0.15
		Ideal Technology Parameters			
Structural Thickness [μm]	1	1	0.25	1	0.05
Young's Modulus [GPa]	130	130	14	145	130
Channel Thickness [nm]	50	50	10[?]	15Ru/70W	10
Channel Resistivity [nΩm]	55	55	55	55	55
Density [kg/m^3]	3826	3826	3826	3826	3826
Gate Oxide Thickness [nm]	80	50	10[?]	50	10
		Model Parameters			
A_{ov} [μm]^2	384	731	51.25	372	0.77
g_0 [nm]	200	180	100	150	10
g_d [nm]	100	90	30	75	5
R_{ch} [Ω]	13.8	16.0	11.0	2.36	25.6
$R_{con}(V_{pi})$ [Ω]	0.07	0.05	2.2	0.5	3880
R_{surf} [Ω]	500	100	100	500	500
C_{gc} [fF]	1185	128.3	4.42	66	0.9
$C_{gb}(x=0)$ [fF]	16.9	31.5	4.5	16.5	1.46
$C_{gd/s}(x=0)$ [aF]	13000	5.0	5.0	88	0.6
k [N/m]	83.2	62.5	2.62	193	0.07
m [fkg]	2961	3443	52	1626	0.86
		Measurements			
t_{mech} [μs]	0.1				0.02-0.08
V_{pi} [V]	8-10	6-8	4-6	3-5	0.04

$C_{gd/s}$ represents the capacitances of C_{gd} or C_{gs}

A_{ov} is not equal to actuation width multiplied by actuation length because of etch holes and channel cutouts. Fringing is not accounted for.

The 90nm model is always used with high overdrive to make it operate quickly and to reduce R_{con} to the 40-400 range.

[?] Estimates based on other parameters. Records are lost.

Table 2.1: Table of relay dimensions and relay model parameters

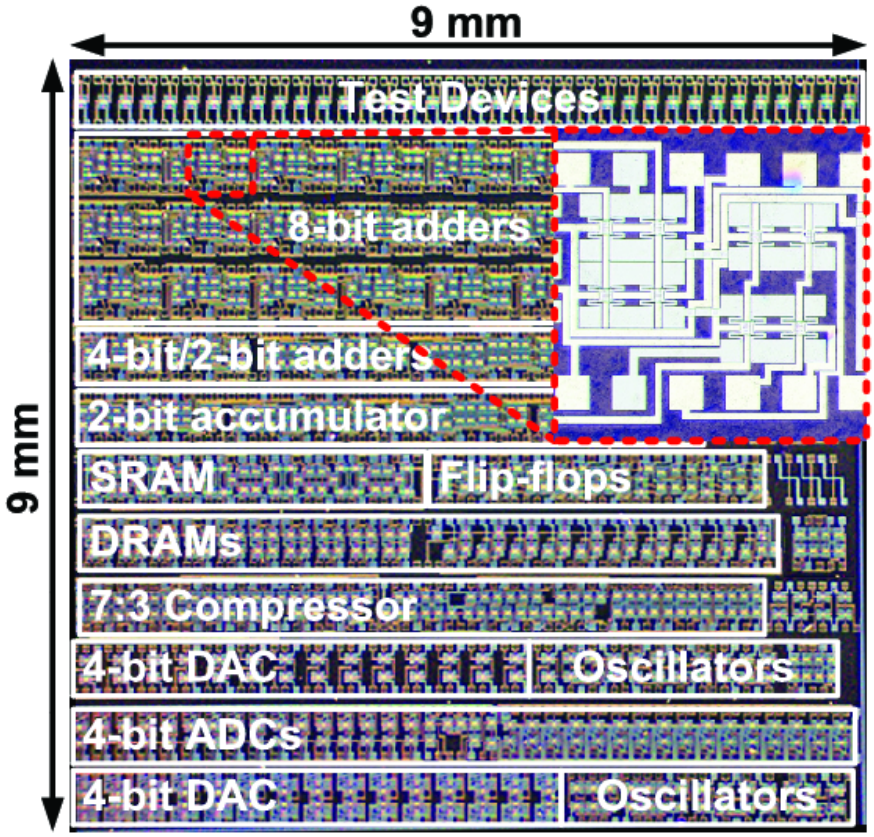


Figure 2.7: Die shot of CLICKR1 test chip. This test chip contained the oscillator experiment featured here. [16, 17].

using more scaled devices suggested a mechanical delay of 100ns[30, 1]. This is an order of magnitude higher than the expected electrical delay for a single relay driving a single relay.

This disparity between mechanical and electrical delays and the analysis above suggest that it is advantageous to include many relays in a single, functionally-complex gate that stacks multiple relays in series between the supply and the output. Gates built this way look similar to the logic gates used for pass-transistor logic, and an example comparing relay and CMOS implementations of the AOI function appears in Figure 2.9. This design style allows for all of the relays to move at the same time because the input signals directly drive the gates of every mechanical device. To reiterate, a tree structure where all of the stages of the tree are driven by an input signal at the same time guarantees the gate will only require a single mechanical delay to achieve complex functionality. Per the above discussion, it will incur a relatively small penalty in electrical delay because of the much faster electrical time constants in a relay based system. This "pass gate" design style uses a smaller number of devices than an equivalent CMOS implementation.

These tree-like, pass-gate structures can be synthesized from binary-decision-diagram representations of logic functions. Preliminary work on that synthesis has been performed

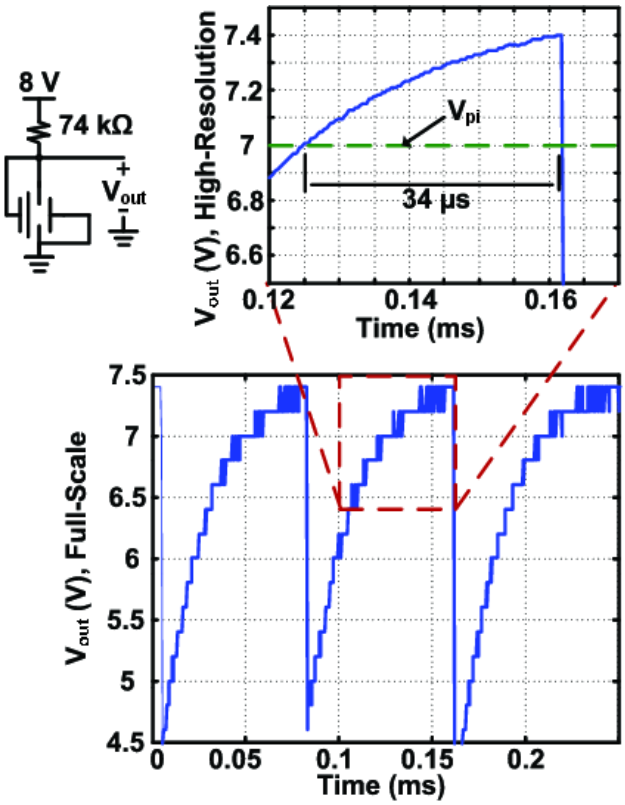


Figure 2.8: Waveforms captured from a relay oscillator which demonstrate the difference between electrical and mechanical time constants in relay systems [16, 17].

[31, 32], but there are still many topics to be addressed in the field of relay synthesis. The circuits discussed below were custom designed.

A relay-based full adder was developed on the 1µm test chip which illustrates this design style. The circuit and measured input and output waveforms are pictured in Figure 2.10. This circuit is based on the Manchester Carry Chain, an adder that was designed before the advent of transistors which considers many of the issues that nanomechanical relays face. In this adder the input signals are used to generate intermediate logic signals – propagate, generate, and kill – for each bit. If the generate or kill signals are created, the carry out is set to one or zero respectively, otherwise a connection is made between the previous bit and the next. This can result in long stacks of devices between the input and output, but all of the propagate, generate and carry signals can be evaluated at the same time. This fits the criteria for our relay based design style. An illustration of how a full relay-based Manchester Carry Chain could be assembled is in Figure 2.11.

The performance of the relay-based Manchester Carry Chain was simulated using the 90 nm Verilog-A model specified in Table 2.1 and illustrated in Figure 2.12. The performance of the adder in this simulation was impacted by the size of the load. In particular, 100fF

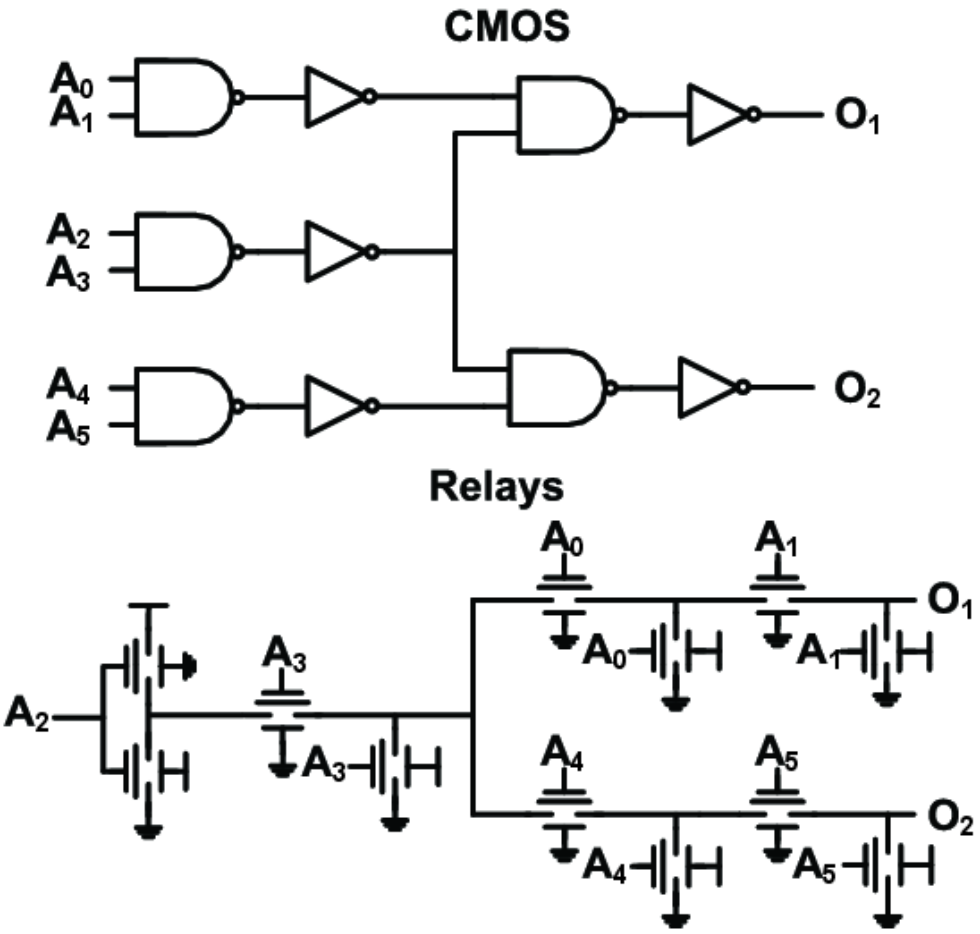


Figure 2.9: Schematics of an AND-OR-INVERT function implemented in CMOS and in relays [16, 17]. The CMOS version consists of many small gates, while the relay version is a single large gate that has only one mechanical delay. The relay version also uses half the number of devices as the CMOS version.

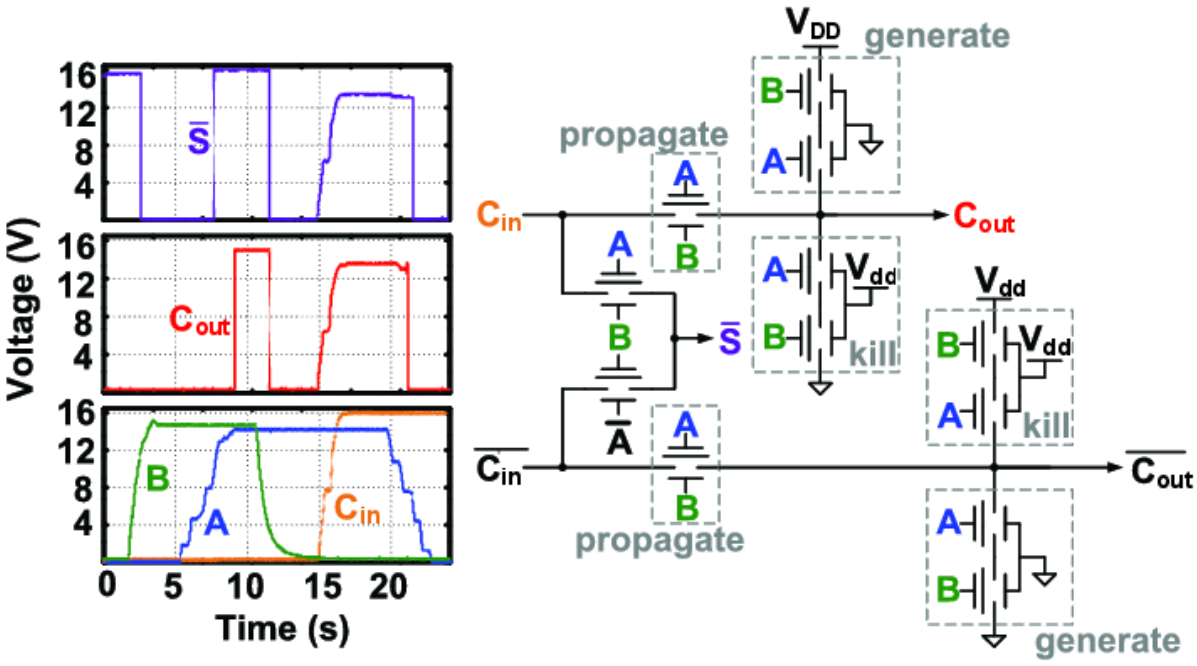


Figure 2.10: Waveforms from and schematics of a relay based adder implemented on a 1µm test chip.

loads on the SUM outputs added enough electrical delay to the adder it started to impact overall performance. The delay was reduced by adding an additional buffer stage at the output of the relay and isolating the loads from the conducting path of the chain. This cost an additional mechanical delay and some area, but improved the performance of the design.

This circuit design was compared against 90nm CMOS adders. The relay adders, the same as those pictured in Figure 2.11 were composed of 12 devices per cell without the buffers (14 with buffers). It would require 24 transistors for a CMOS implementation, which reduces the area penalty of the individually larger relays. The single, compound-gate, 32-bit add thus requires 384 relays (448 with buffers). Each relay is slightly less that 12µm,^{2} so assuming a wiring overhead of 30% the relay based adder would occupy $\approx$ 6000 µm^2 ($\approx$7000µm^2 with buffers). The CMOS adders were Sklansky adders [33], which have been shown to be the minimum energy adder topology over a wide range of delays [34]. The most salient point to compare against relays is the minimum-energy, maximum delay point. When designed at that point – i.e.: synthesized from standard cells with no delay constraint – the adder uses 836 gates and occupies an area of about 2000 µm^2. The energy delay characteristics of the adders were pulled from [34] and compared against the relay adder simulations.

The CMOS adder reaches its minimum energy point [2, 34] for delays above 1ns. Thus, at delays of 10-50 ns, a single MEM-relay adder offers an improvement of $\tilde{1}$0x in energy at an area overhead of 3.5x compared to the CMOS adder. There is a clear advantage to

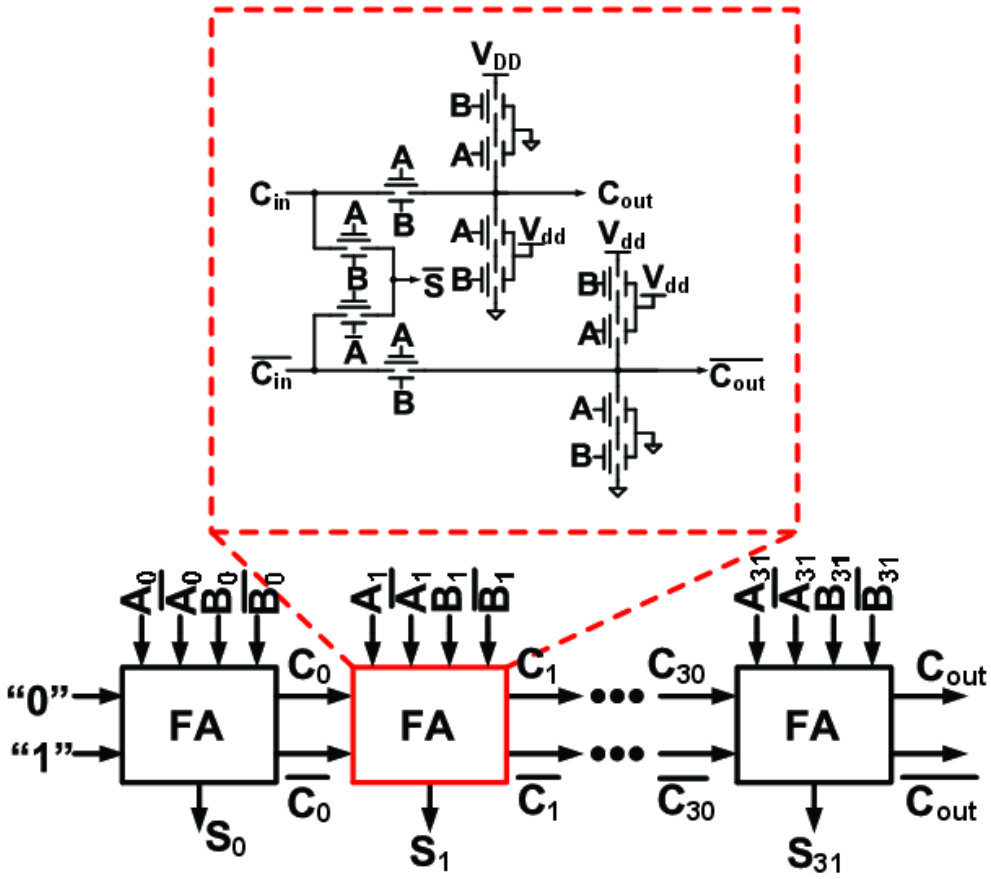


Figure 2.11: Schematic of a Manchester Carry Chain adder.

this technology for applications requiring 20 MOPS/s or less. Relays adders can be put in parallel to achieve higher throughputs. This trades off area overhead with performance. For instance, the parallelized curves in Figure 2.13 would require 100x the area of a CMOS adder. This area penalty can be improved with optimized relay layouts.

The area penalty can be further mitigated by using 6T relays. The unit cells of the Manchester carry chain from Figure 2.11 are composed of both a true and a complement path to avoid using additional inverters to generate the sum. This design saves a very costly mechanical delay, but almost doubles the area and device count of the circuit. This seems especially wasteful since the generate, kill and propagate blocks in both the true and complement paths are controlled by the same signals: A on gate and B on both for propagate, a series combination of two devices with A and B on gates and ground on the bodies for generate, and a series combination of two devices with A and B on gates and V_{dd} on the bodies for kill. In each case, there are separate devices controlled by the same gate/body signals which carry different drain/source signals.

These devices can be merged into a single 6T device where the gate and body signals

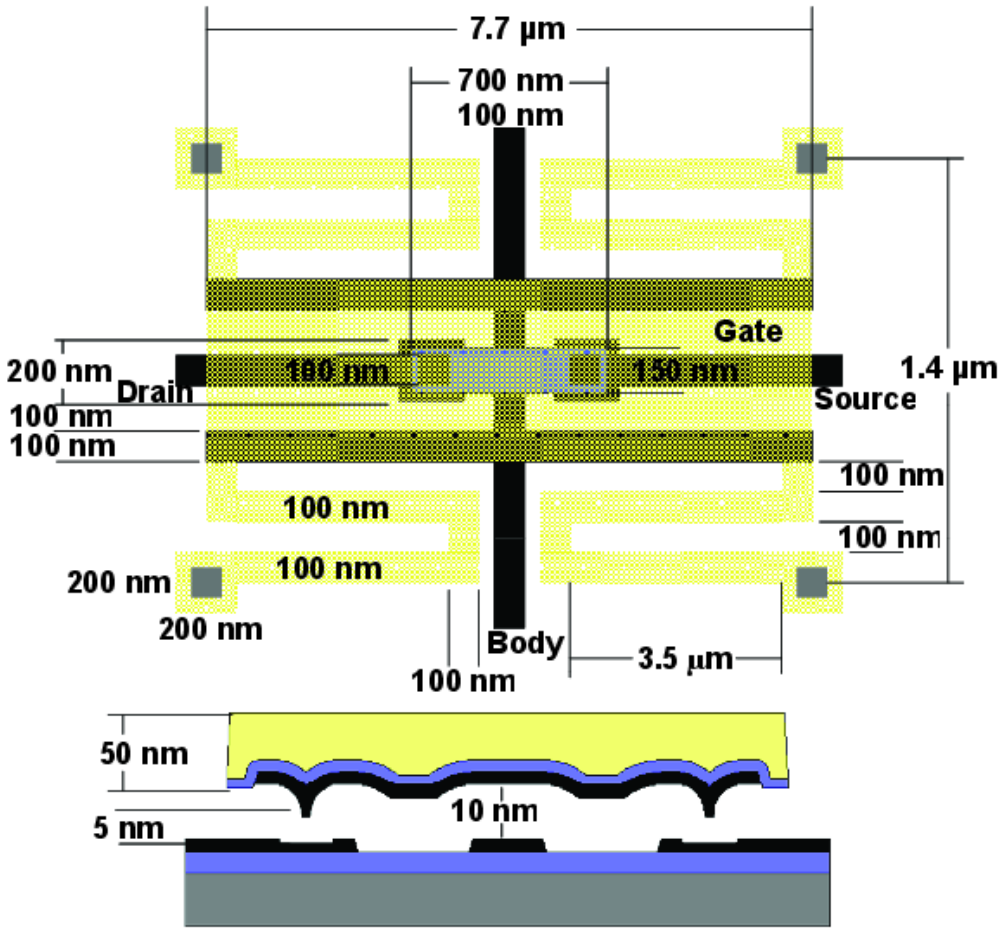
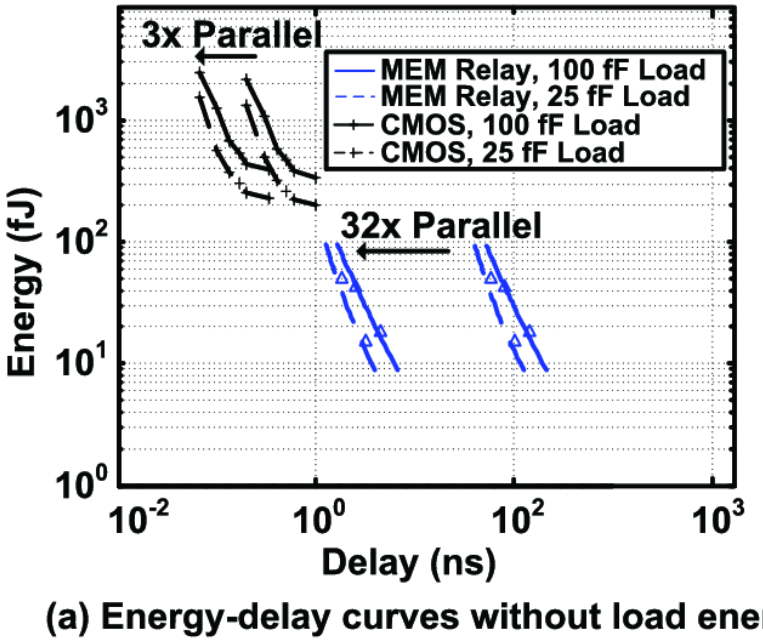


Figure 2.12: Possible layout of a 90nm 4T relay.

control two separate drain/source pairs. A modified Manchester Carry full adder appears in Figure 2.14. Merging devices in this way cuts the total number of devices in the adder by 41.5% (each unit cell uses seven devices instead of 12), and accordingly cuts the total energy per operation by the same amount. This costs some area on each device, which theoretically reduces the area available for actuation and the speed of the device, however that penalty is negligible and this reduction is power and device area comes essentially for free.

6T relays have been evaluated against CMOS multipliers [29] and the relay multiplier circuits compare to the CMOS multipliers even more favorably in terms of energy and delay than the relay adder circuits above. The 6T relay multipliers show a larger energy benefit (10x) and a smaller delay penalty (4x) and area penalty (1.5x). This confirms that the circuit techniques described here scale to larger logic blocks. However, large logic blocks aren't enough to build a computing system, and these results point to future work demonstrating the other necessary components which are needed to assemble large VLSI systems: timing and memory.



(a) Energy-delay curves without load energy.

Figure 2.13: Energy delay comparison of CMOS Sklansky adders against relay Manchester Carry Chain adders [16, 17].

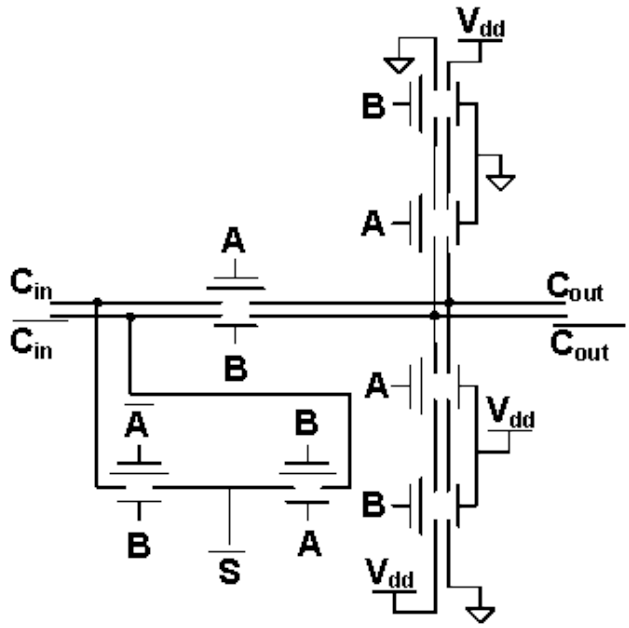


Figure 2.14: Manchester Carry full adder implemented with 6T relays. Using 6T relays reduces the number of devices needed from twelve to seven at very minimal delay cost.

Chapter 3

Sequential Relay Circuits

Sequential logic is obviously an important component of any VLSI design, including one made from relays. Though the previous Chapter 2 suggests that large, single-mechanical delay gates are the best logic style for relays, any sufficiently complex logic function will result in an explosion of area if it isn't broken down into smaller subunits. In addition, the limitations on D for relay gates requires that logic be broken up into multiple stages. Sequential logic is needed to order the operations of those smaller chunks of logic.

Consequently, the CLICKR1 test chip was used to demonstrate a latch made from relays. The schematic and experimental results for the relay latch appear in Figure 3.1. The latch is shown to successfully transition between opaque and transparent state and to pass information from D to Q while transparent.

This latch is implemented in the same was as a CMOS latch would be. Two pseudo-inverters made of "N"-biased relays and pull-up resistors are placed back to back to make a buffer, and relay pass gates are used as a multiplexer to select whether the buffer's input is driven by the D input to the circuit or feedback from Q.

The discussion in Chapter 2 suggests that circuits which are designed by mapping relays to CMOS implementations are often suboptimal because they incur more than one mechanical delay. That is obviously true of this latch, which would incur two mechanical delays for each transition of the input: the edge would cause the first relay to change state which in turn would cause the second relay to change state. It is possible for the latch to incur even more mechanical delays if the pass gates which switch the input of the buffer are set in motion after Q reaches a final value. However, the CLK signal can be adjusted to arrive early enough that the mechanical delay of the multiplexer occurs at the same time as the second pseudo-inverter.

Using "P"-biased relays would reduce the delay of each latch to a single mechanical delay, but a flip-flop composed of two back-to-back latches would incur two mechanical delays in its operation: one for each buffer. The CLK signal can be assumed to arrive early so that the pass gate delay happens at the same time as the buffer delays. A relay flip-flop that incurs this penalty is pictured in Figure 3.2 with a two mechanical delay path sketched through it.

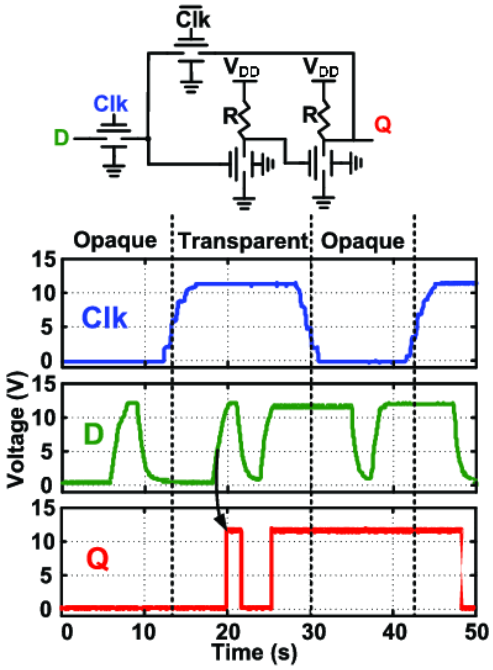


Figure 3.1: Relay latch

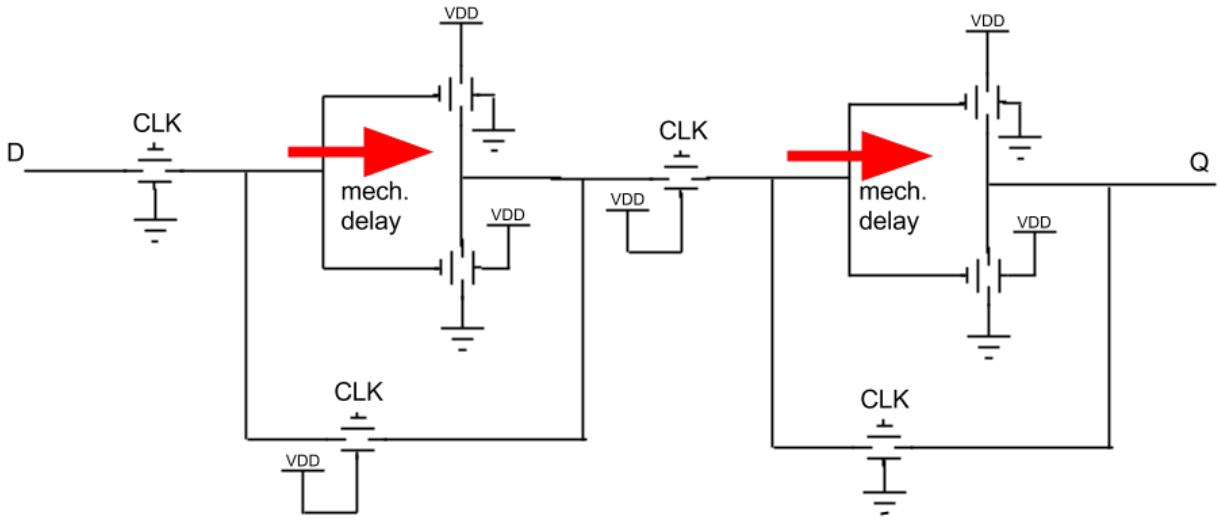


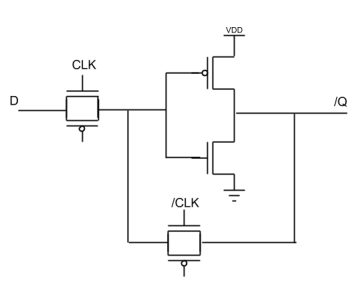
Figure 3.2: Relay flop

This relay flip-flop adds a significant amount of delay to a VLSI system. Combining one mechanical delay from logic with two from a flip-flop would triple the overall delay of a logic stage relative to non-sequential logic, which is obviously detrimental to the overall system performance. The remainder of this chapter will explore different ways to build sequential logic out of relays in order to reduce this delay penalty. A new latch will be introduced, and its timing constraints will be elaborated. Latch based timing will be suggested to share the staticization delay of the latch with the mechanical delay of turning on a relay, and the multiplexing element of the latch will be examined to ensure that the instant, coordinated feedback transitions can happen even when individual devices switch slowly. Simulations will illustrate these new sequential elements working in pipelined systems.

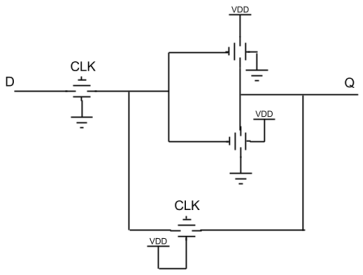
3.1 A Comparison of Relay State Elements

Even the most basic CMOS state element, a pass-gate latch, would map poorly to relays. A direct mapping is pictured in Figures 3.3a and 3.3b. The only accommodation to relay based design in the figure is the substitution of the CMOS inverter with a relay buffer to take advantage of the ambipolar switching characteristics of relays. A natural way to remove the staticization penalty is to move the relay buffer into the feedback path of the latch as in Figure 3.3c. This adds capacitance to the output of the logic stage driving node D, but that contributes an insignificant amount to the overall delay per the discussion in the previous chapter. In exchange for this capacitance penalty, the staticization buffer of the latch can be driven at the same time as the logic of the following stage. If the transitions of the pass gate latches are perfectly timed then this arrangement hides the mechanical delay of staticization. An example of a latch circuit appears in Figure 3.4 and a timing diagram showing the simultaneous staticization of data and drive of the next stage is pictured in Figure 3.5.

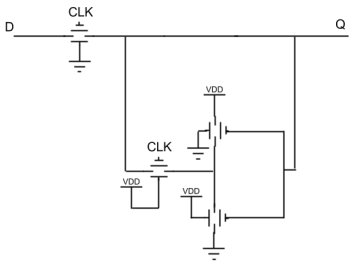
The operation of this latch is important for understanding the pipelines to follow, so a discussion the relationship between the mechanical delays and the electrical edges shown Figure 3.5 follows. The figure shows that the devices in BUF1 experience a mechanical delay of transition while $D1$ is unknown. $D1$ is unknown during that period because a data transition has just happened on $Q0$, and it takes a mechanical delay to conclusively propagate a data transition through a gate (CLB0 in this case). $D1$ is marked as unknown rather than simply transitioning a mechanical delay later because $Q0$ could be attached as an input to the source of a relay (like the carry input of the Manchester Carry Chain in the previous chapter), and a data input on a relay source could race through a combinational logic block quickly: on the time scale of an electrical delay. The $Q0$ transition that prompted the uncertainty in $D1$ is caused by the forward pass gate closing and connecting the $Q0$ node to the $D0$ node. $Q0$ becomes staticized when the reverse pass relay closes, which happens simultaneously (at the scale of mechanical delays) with the configuration of the feedback buffer. $Q0$ becomes unknown at the end of the diagram because it is not statically driven by either the forward or reverse pass relays.



(a) A latch made of CMOS transistors.



(b) A latch made of relays using the same design as the CMOS latch.



(c) A latch made of relays with the staticization buffer in feedback.

Figure 3.3: Schematics of several different latches. A relay latch could be implemented in the same style as a CMOS latch, which is pictured in 3.3b. There are a few optimizations to the relay latch: it uses a buffer rather than an inverter and its pass gates are controlled by a single clock phase. However, it would incur a mechanical delay from D to Q. A latch with the buffer in the feedback path avoids that penalty.

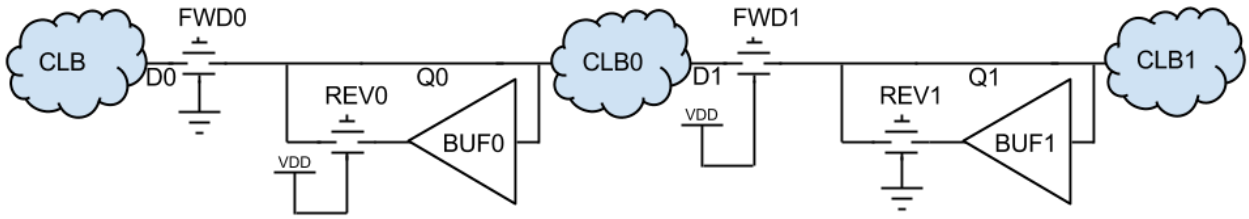


Figure 3.4: A schematic which shows, in general, how sequential logic could be made from relay latches and combinational logic blocks.

It is tempting to remove $BUF0$ and $BUF1$ in order to reduce the number of relays needed for timing elements. If the buffers are removed, the state is stored as charge on the $Q0$ and $Q1$ nodes rather than in the position of the staticization relays. Since relays have no leakage, this dynamic storage is nominally safe until the storage node is no longer high impedance. However, this optimization is risky because of coupling and charge sharing. Automatically placed and routed VLSI systems won't take care to keep $Q0$ and $Q1$ isolated from aggressor wires which could transiently change their states. Further, the fanout of the latch could reduce the voltage stored on $Q0$ by distributing it to many other gates. Though this dynamic storage could be made to work with careful design and layout, and though it might yield performance benefits in highly optimized systems, this discussion focuses on the much more generally useful staticized latch.

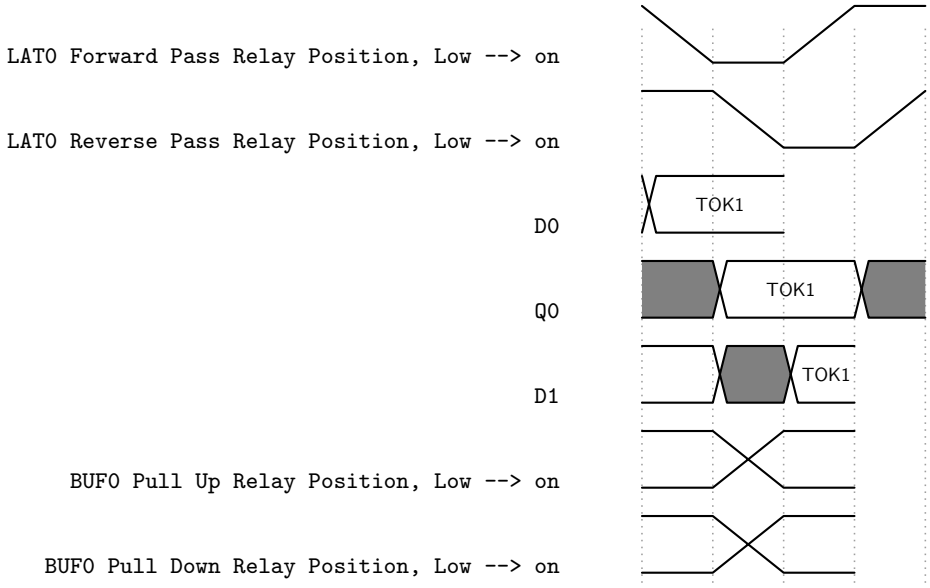


Figure 3.5: Timing diagram illustrating how staticization and driving the next stage happen at the same time in the pipeline in Figure 3.6. Each time increment is one mechanical delay.

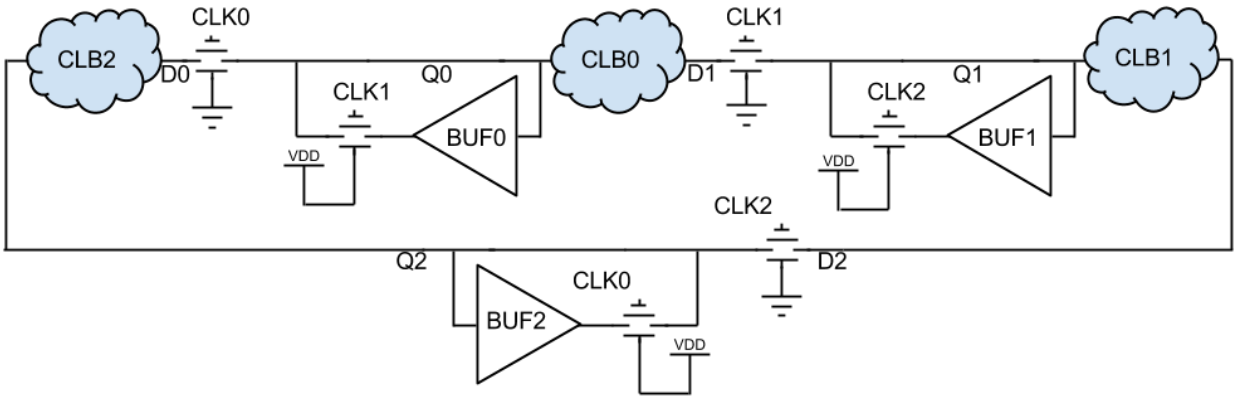


Figure 3.6: A pipeline made of relay latches and relay combinational logic blocks.

3.2 Three Phase Clocked Relay Pipelines

To show that this latching scheme is suitable for general use in VLSI systems, it is necessary to show that it can be made into an infinite length pipeline. This is usually shown by demonstrating that a pipeline can drive itself. Figure 3.6 shows relay latches assembled into an infinite pipeline and Figure 3.7 shows a timing diagram that indicates the operation of this pipeline.

The pipeline is clocked by a three phase clock with a 66% duty cycle in order to ac-

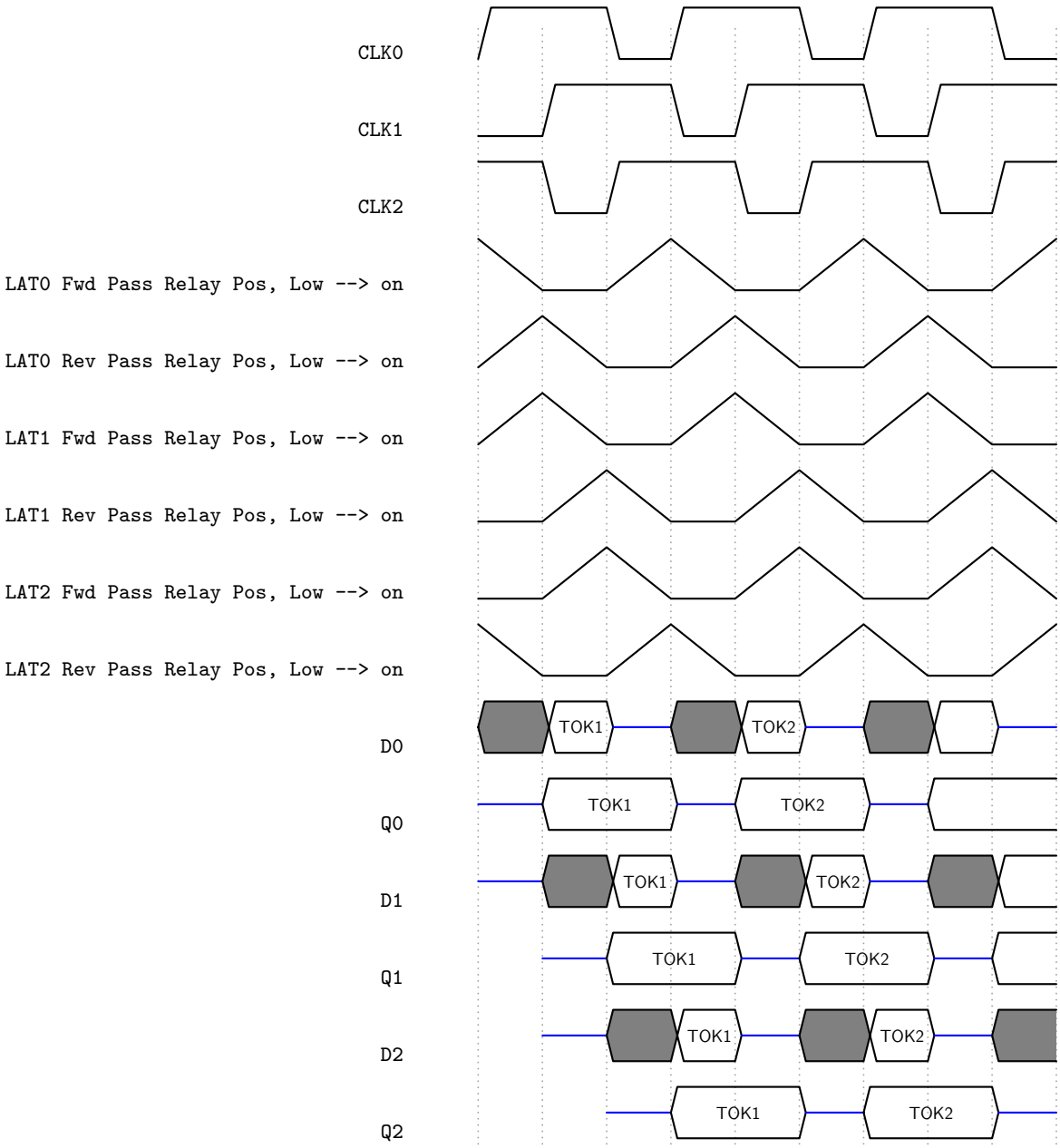


Figure 3.7: Timing diagram illustrating the progression of two tokens, `TOK1` and `TOK2`, through the relay pipeline from Figure 3.6. Gray cells are unknown data and blue lines are high impedance states where the node is not driven by any relay.

commodate the motion of the pass gates. When a phase of the clock first goes high the associated pass gates begin moving from *off* to *on*. One mechanical delay later the pass gates have made contact. At that point the clock remains high for an additional mechanical delay to allow the pass gates to configure the logic in the following stage. For instance, in the third time interval the $LAT0$ reverse relay and $LAT1$ forward relay are *on* and $D2$ is being configured, which is reflected by its unknown state. Finally, the clock goes low for a mechanical delay to allow the pass relays to return to their original position. Thus the three phases of each clock could be labeled with their purpose: *turnon, configure, turnoff*.

Driving the pass gates in this way allows the token to move through the pipeline. First the D input of a latch is configured, which results in it being unknown for a mechanical delay. This can be seen in $D0$ in the first time interval. While the D value is being configured, the forward pass gate of the latch is simultaneously closing. The $LAT0$ fwd. pass relay position in the first time interval shows this. The unknown state is prevented from reaching the Q node of the latch until the relay closes, but the clock of the relay arrives early to ensure that the state on D is immediately passed to Q when it is resolved. That D to Q transition occurs on $Q0$ in the second time interval. The change in Q results in a potential logic transition on the D of the next stage (for instance, $D1$, second interval), so the forward latch connecting D to Q and the feedback latch of the stage driving D are held in place for one mechanical delay to guarantee stable inputs to that logic. The feedback relay for this stage is being closed at the same time ($LAT0$ rev, second interval), which staticizes the state to drive the next stage. After that, the feedback relay is released ($LAT1$ rev., 4th interval), which results in Q being undriven and entering a high impedance state ($Q0$, fourth interval). Though Q is undriven in this phase, the state is likely to be preserved because there is no leakage off of the Q node. However, in this state Q is susceptible to charge coupling from nearby wires.

The token passes through three combinational blocks during one three phase cycle of the clocks. This can be seen on the timing diagram: $TOK1$ passes through $D0$, $D1$ and $D2$ in intervals two, three and four. Thus the pipeline has a throughput of three operations every three mechanical delays, or one mechanical delay per operation. This recovers the performance lost when using CMOS style flip-flops.

This theoretical analysis suggests that there is no sequencing overhead for introducing this sequential element, which reflects the coarse time scale being used to analyze the system. Naturally, non-idealities like electrical delay, degrade this performance. These nonidealities will be discussed later in the chapter.

Though this pipeline improves the throughput of sequential relay VLSI systems, the arrangement isn't without its wrinkles. Data needs to be fed into this pipeline in phase with the clock. This can be clearly visualized if $CLB0$ is assumed to have many inputs. A transition on any $CLB0$ input would cause $D1$ to become unknown for one mechanical delay as the $CLB0$ relays moved to their final states. This would have no effect on the pipeline's overall behavior if such a transition occurred during the first time interval because the state of $D1$ would resolve before it being taken by $Q1$. The same can't be said for a $CLB0$ input transition in the second interval because a change at the input would result in a full mechanical delay of uncertainty, which could allow an uncertain state to reach $Q1$. This

constraint has some implications for synchronizing inputs, but the most dramatic effect is that feedback in the pipeline is severely constrained: a stage can only feed back to a stage which is on the next clock phase. Said another way, feedback in this pipeline always needs to happen across a number of stages which is a multiple of three.

3.3 Two Phase Relay Pipelines

Reducing the number of clock phases would make this feedback requirement less onerous. One tempting, but ultimately wrong, way to do this would be to eliminate the *turnoff* phase of clocking. This approach initially looks promising because the relay will stop conducting after moving only a small distance away from the surface, which suggests the timing scheme doesn't need to allocate a full mechanical delay for the relay to return all the way to its resting position. However, this clocking scheme steals time from the previous stage: if a clocking scheme allowed for a "pulsed turnoff" of a relay where it didn't return all the way to its starting position, then the previous stage would have to configure the logic in the pulsed window. Further, trying to actuate a pulsed relay a second time after the first pulsed turnoff would result in the relay turning *on* faster because the *turnon* time is a function of the separation of the gate and contact. Each subsequent application of a normal length *turnon* and *configure* pulse with a short *turnoff* pulse would result on the relay remaining *on* for longer. Eventually this causes the relay being permanently stuck shut.

A better option can be found by examining the timing constraints carefully. Specifically, there seems to be some slack in the system because the staticization buffer and combinational logic are capable of operating faster than the pass gates. These logic blocks are only "operational" during two stages: the *configure* phase when relays are moving to a stable state and the *drive* phase where they drive the next stage. The logic blocks are idle during the *turnoff* phase, neither driving nor being driven.

As established above, the pass gates need to go through three phases: *turnon*, *drive*, and *turnoff*, which is different from the two "operational" logic phases. The difference between the number of clock phases required by the logic and the pass gates is resolved in the three phase pipeline by slowing the logic down with a third *tristate* phase where the logic is undriven. During this phase the previous pass gates turn *off* and the next pass gates turn *on*. However, the latches could operate faster if the pass gate structure was replaced with something that required only two clock phases to operate.

A latch which uses such a pass gate appears in Figure 3.8, a pipeline made of those latches appears in Figure 3.9. The forward and reverse pass relays of the previous example have been replaced by somewhat unusual pass structures. The pass structures consist of one relay with a body attached to the supply in parallel with a relay with its body connected to ground. The two parallel relays share gates, drains and sources. At first glance, the structure would appear to serve no purpose, since any logical value of CLK would result in one relay being *on*, so that the structure is always conducting. This structure is used instead of a wire because any transition on the CLK will result in both relays moving for one

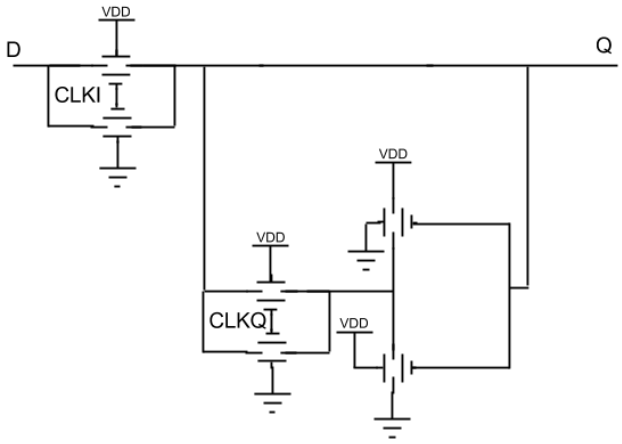


Figure 3.8: A relay based latch capable of adjusting its forward and revers pass gates on a two mechanical delay cycle.

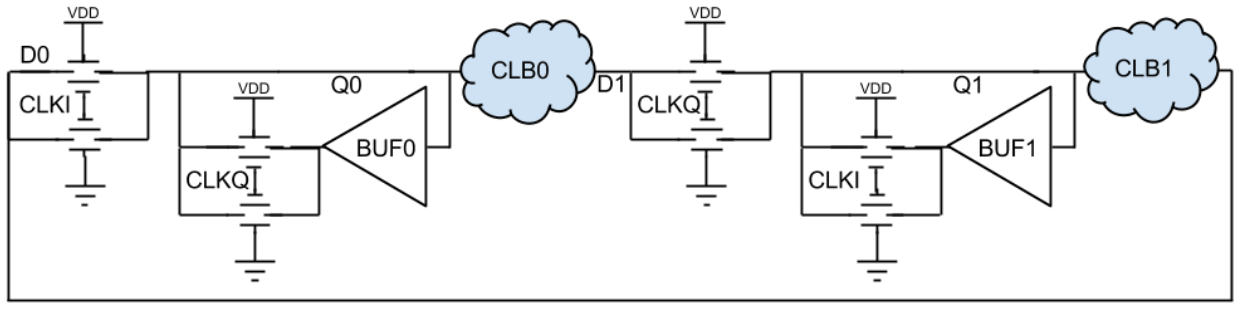


Figure 3.9: A pipeline made of relay latches with two mechanical delay cycles on the forward and reverse latches.

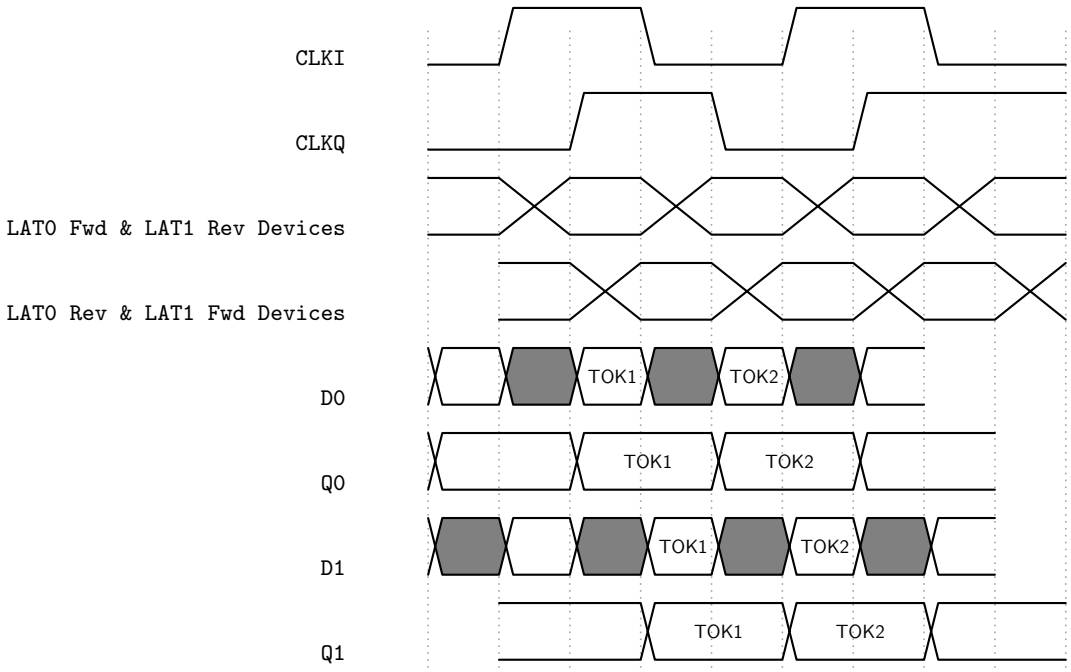


Figure 3.10: Timing diagram illustrating the progression of two tokens, `TOK1` and `TOK2`, through the relay pipeline from Figure 3.9. Gray cells are unknown data. The timing diagram indicates the positions of the pairs of relays which comprise forward and reverse pass structures in some strips. Because the relay bodies are biased to opposite voltages, they are always in opposite states and can be depicted on the same strip. A clock transition causes one relay in the pass structure to transition *off-on* while the other transitions *on-off*. The net effect is that the pass structure is transparent while one relay is *on* and the other is *off*, and it is opaque for a mechanical delay while both devices are transitioning.

mechanical delay: one moves from *off-on* and the other from *on-off*. During this time, the pass structure is opaque since neither relay is in contact. The structure is always transparent at all other times because one relay is *on* and the other is *off*. In summary, opacity is only caused by a recent transition on CLK.

This pipeline is clocked with a two phase, quadrature clock with a 50% duty cycle and a period of four mechanical delays as seen in Figure 3.10. The figure also illustrates the process of a token through the pipeline. `TOK1` is introduced to $D0$ and $Q0$, which causes $D1$ to become unknown as $CLB0$ configures. A transition on `CLKQ` arrives at the same time as the introduction of `TOK1` to isolate $Q1$ from the unknown $D1$ for a mechanical delay while $CLB0$ configures. The isolation is achieved by causing a transition in the forward pass structure of $LAT1$. One mechanical delay later, the forward pass structure of $LAT1$ becomes transparent, the reverse pass structure of $LAT0$ staticizes $D1$ as $D1$ resolves, and $CLKI$ transitions. The forward structure becoming transparent passes $TOK1$ to $Q1$ at the same time it resolves on $D1$, and the transition on $CLKI$ results in the reverse pass structure of

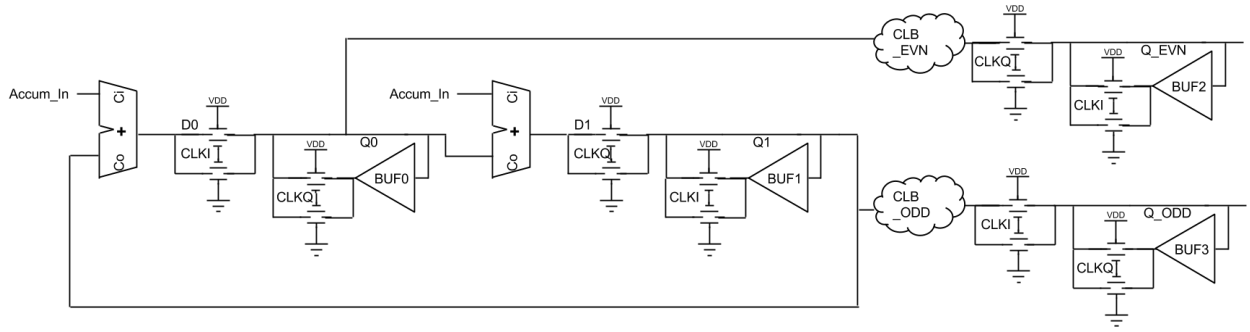


Figure 3.11: An accumulator for a relay based system showing separate logic for odd and even samples and an output serializer.

$LAT1$ becoming opaque so there is no contention on $Q1$. The transition on $Q1$ makes $D0$ uncertain, but the $CLKI$ also isolates $D0$ from $Q0$ to prevent contamination. The cycle repeats after that.

This clocking scheme still achieves one mechanical delay per stage even though the clock period is longer than the previous pipeline example. Because each clock edge cause a one mechanical delay period of opacity, each full clock cycle actually creates the same set of states in the pass structure twice: *opaque-transparent-opaque-transparent*. The pipeline operates on these repetitive half-clock cycles, and relies on the opacity to prevent uncertain data from leaking into the next stage. For instance, `TOK1` passes through two combination stages reaching nodes $D1$ and $D2$ during the half clock cycle in the third and fourth time intervals.

Feedback into this pipeline still needs to happen in phase with the clock, but there are fewer clock phases to keep track of. This pipeline could feed back into itself after two cycles. However there are some structures which demand feedback in one clock cycle. Accumulators are a classic example, and they are a useful example to demonstrate general methods of dealing with single cycle feedback. An example accumulator is pictured in Figure 3.11. Two stages of pipeline allow for two additions over two cycles, such that the accumulator's current value appears on nodes $Q0$ and $Q1$ alternately.

This creates a separate stream of "even" and "odd" results from the accumulator which appear on different nodes and on quadrature clocks. This data can be handled in an assortment of ways. The simplest option is processing them separately separately in even and odd logic which is clocked by odd and even latches. That option is shown in Figure 3.11. However, some logical operations will depend on both even and odd values. Those operations can be created by aligning the even and odd values through back to back latches. The early data passes through three latches and the late data passes through only two latches. This guarantees that both pieces of data are available on the same clock phase, though at the cost of some latency.

Odd and even samples can be serialized onto a wire by using a pair of relays driven by CLKI and CLKQ. An example serializer appears in Figure 3.12 and a timing diagram for

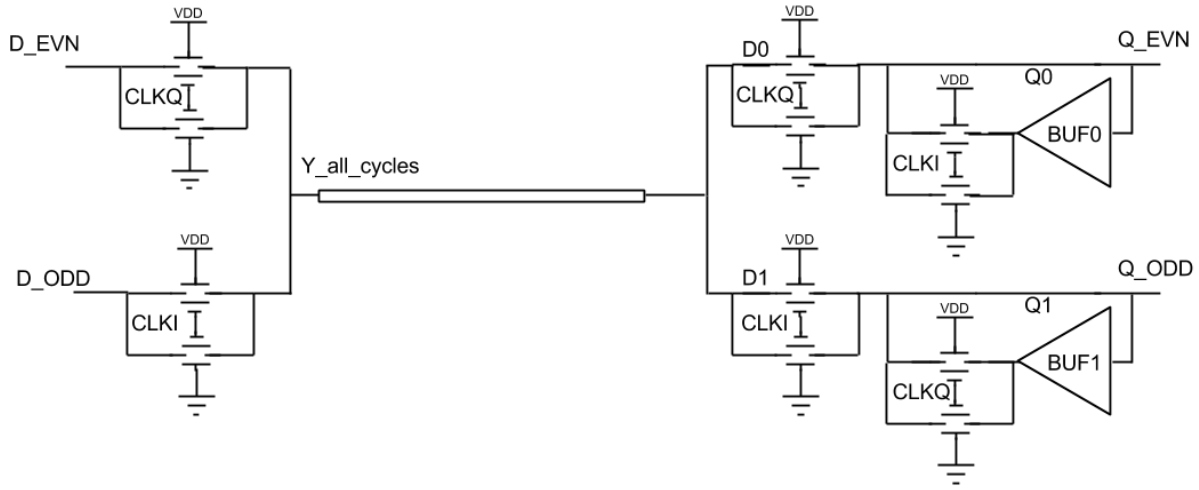


Figure 3.12: Serialization and deserialization of odd and even samples onto a single wire.

the serialization process is in Figure 3.13. The wire will receive a new value each mechanical delay. These serialized wires are useful for transmitting data to external systems. However, the wire needs to terminate on a latch, and in a relay VLSI system each latch is either odd or even. As a result, only the odd or the even samples will be pulled off of the wire. A pair of receivers, one on the odd phase and one on the even phase, could pull all of the data off of a serialized wire and resume separate computation of the odd and even samples.

3.4 Skew Tolerance

Up until now the analysis of the pipelines has assumed that each transition of a device takes exactly one mechanical delay, that the mechanical delay is the same for every device, and that the electrical delay is negligible. These assumptions are safe to first order, but the clocking scheme should have some way to account for timing elements which violate these assumptions. That flexibility can be found by carefully analyzing the exact requirements of each of the clock edges in a relay clocking scheme.

A closer look at the timing of a two phase latch from Figure 3.8, shown in Figure 3.14, suggests that the timing constraints can be relaxed from "quadrature clocks with a period of four mechanical delays." The clock attached to the pass structure connected to D and Q, referred to as the forward clock, doesn't need to be perfectly in quadrature with the clock of the pass structure connected to the buffer and Q, the reverse clock. Instead the forward clock only needs to be high for two mechanical delays around the edge of the reverse clock. An edge on the reverse clock makes the reverse pass structure opaque, and these relaxed timing requirements ensure that the Q node is driven by D during that opaque period. The first mechanical delay of the forward clock pulse lets the forward pass structure configure so that

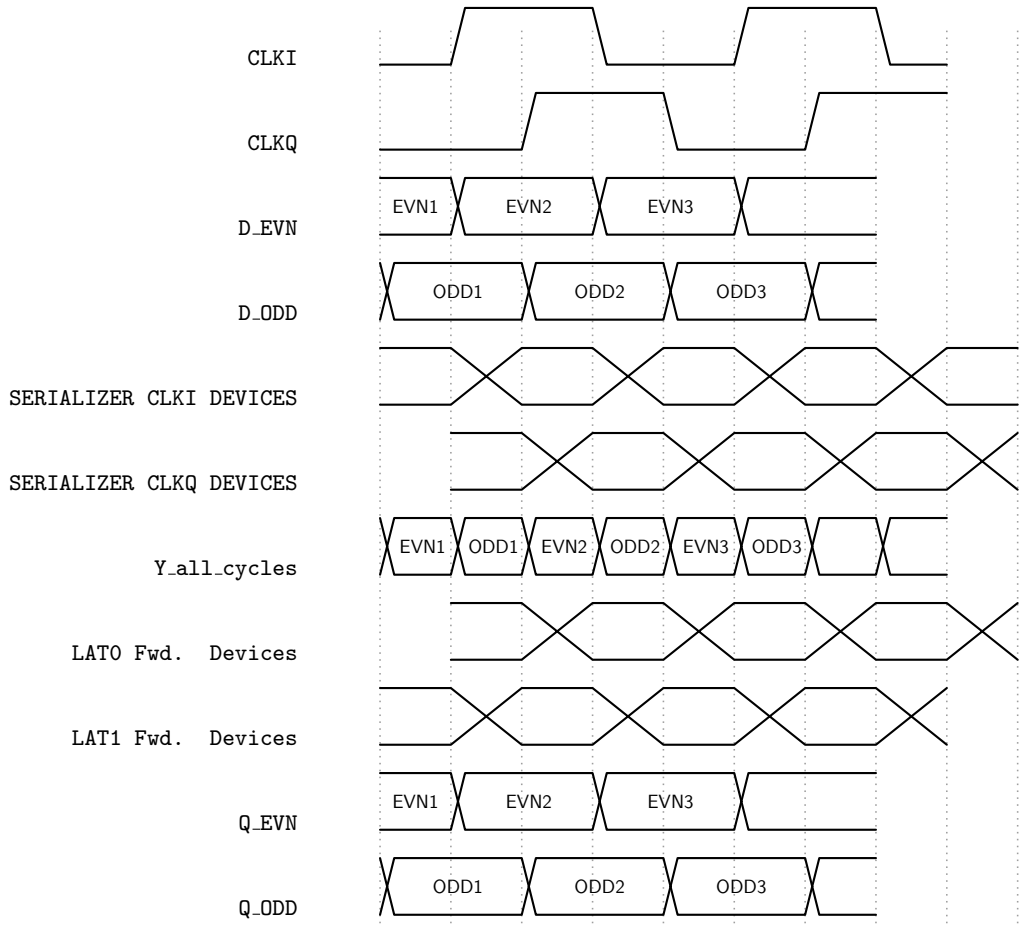


Figure 3.13: Timing diagram illustrating the serialization of odd and even values onto a wire.

it can pass data and the second holds the value on the storage node while the staticization buffer and feedback pass structure configure. The reverse clock does need to arrive with a tight time relationship to the forward clock – the reverse clock edge must be one mechanical delay after the forward clock edge – to ensure there's no contention when D drives the node. The period of the pipeline is set by the timing of the reverse staticization clocks, the forward clocks just provide transparency in order to grab each incoming data transition.

These modified requirements suggest modifications to the latch structure. By driving the two devices in the clock differently, the opaque and transparent periods of the clock can be extended to accomodate non-idealities like electrical delays. A modified version of the clocking structure which can accommodate this clock stretching appears in Figure 3.15. By extending the overlap between $CLKI1$ and $CLKI2$ or $CLKQ1$ and $CLKQ2$ it is possible to keep the forward or reverse pass structures transparent or opaque for more time.

Controlling this two phase latch requires many closely timed clocks, and distributing those

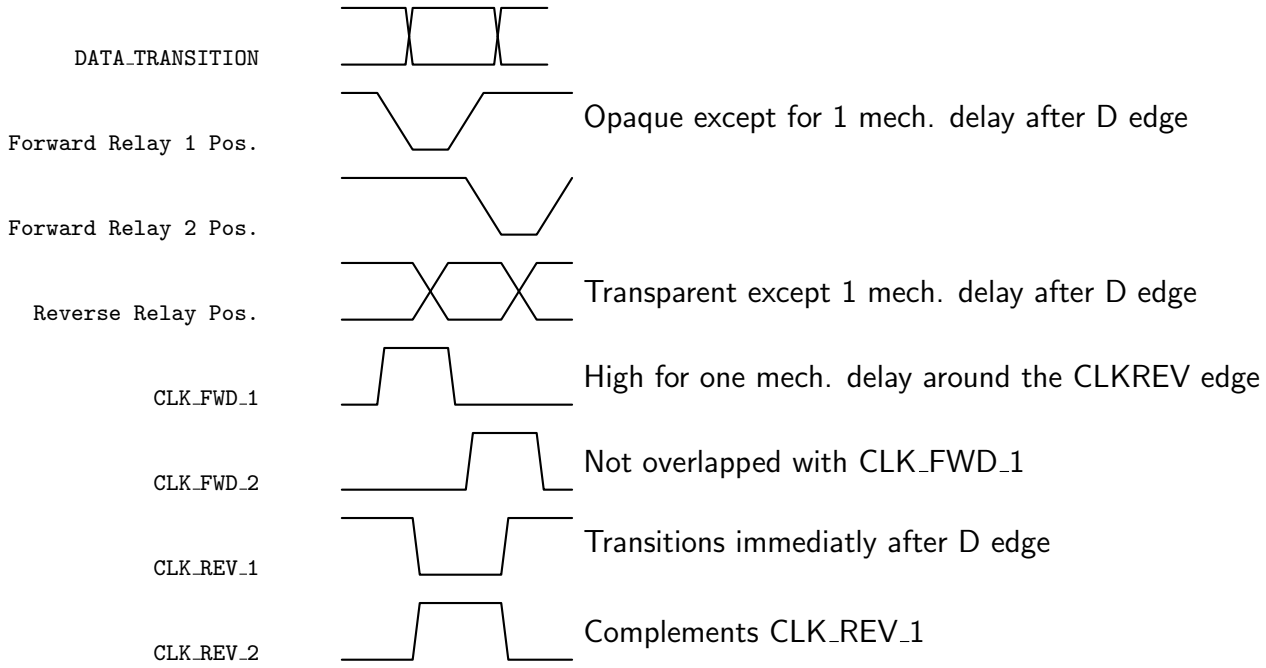


Figure 3.14: Timing diagram illustrating minimum requirements for clocking a two-phase relay latch and the relationship between a global clock and local latch clocks.

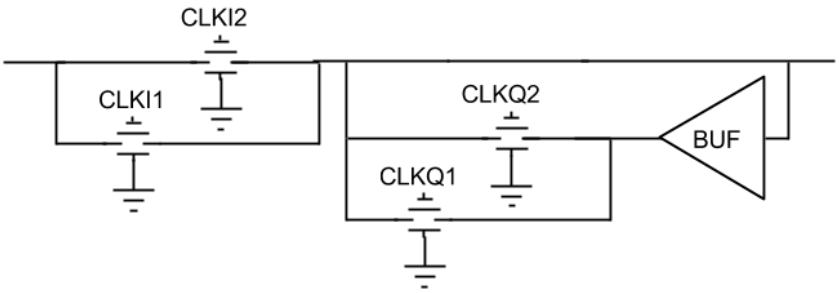


Figure 3.15: A latch with separate clocks on each device in the forward and reverse pass structures. The independent clocks allow the latch to remain opaque for longer than a single mechanical delay in order to tolerate sources of skew in the circuit.

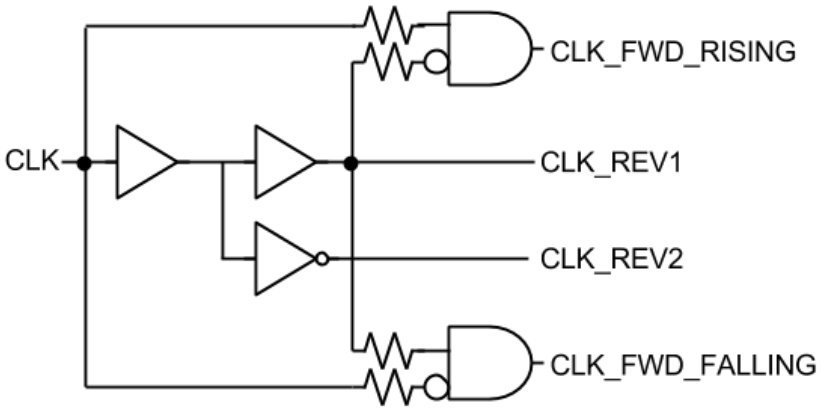


Figure 3.16: A circuit which can generate forward and reverse clocks for a relay latch based on a global clock.

clocks in phase poses a tremendous challenge. This challenge can be neatly sidestepped by using a local clock generation circuit to derive the four latch clocks from quadrature $CLKI$ and $CLKQ$. A local clock generation circuit is pictured in Figure 3.16. The global clock is copied through the chain of buffers and inverters to appear two mechanical delays later in true form on CLK_{REV1} and in complement form on CLK_{REV2}. CLK_{FWD_RISING} and $CLK_{FWD\_FALLING}$ are created by the compound NAND-NOT gates such that they appear a mechanical delay before and after the transition on CLK_{REV1} and CLK_{REV2}. These gates are implemented using relays arranged in a standard CMOS logic style. As usual, the relay implementations of standard CMOS logic style create multiple mechanical delays, but that is desirable in this case.

This circuit also supports varying the amount of non-overalap time allocated to the stage it is driving by varying the resistors embedded in it. These resistor values would be adjusted at design time to replicate the worst case path through the clocked circuitry. By building the resistors out of relays with bodies at ground and gates at the supply voltage the resistance will track variations in V_{dd}. The local clock generation circuit will also track local process variations because it is located physically close to the logic it serves.

3.5 Simulated Results

These techniques have been simulated using a model of a seesaw relay built in a 200nm node to verify their operation. A seesaw relay is an electrostatically actuated, torsional switch, and it differs from the 4T and 6T relays examined previously because those devices are vertical switches. An example seesaw device is pictured in figure 3.17. The device's electrical schematic representation also appears in that figure. The seesaw relay has a gate electrode like 4T and 6T relays, but instead of being supported by linear springs which exert force in the direction normal to the substrate, it is supported by torsional springs which apply clockwise

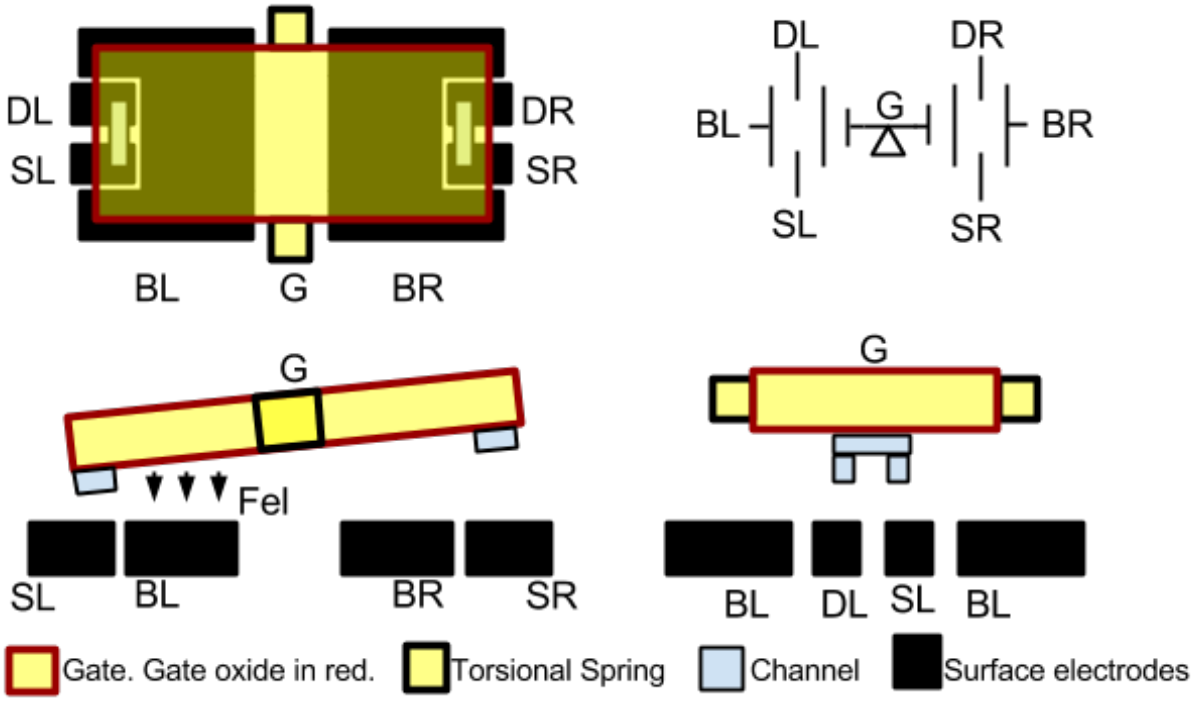


Figure 3.17: A seesaw relay and it's electrical schematic representation.

or counterclockwise torques to keep the gate flat. The gate is actuated by electrostatic forces which appear between the two body terminals and the gate. One body terminal, *body left*, overlaps the left side of the gate, and the other, *body right* overlaps the right side of the gate. Any infinitesimal slice of either terminal applies a vertical electrostatic force to the gate slice above it, but these forces are applied far away from the axis of rotation of the seesaw. Thus, the forces translate to clockwise or counterclockwise electrostatic torques. These torques fight against the torsional spring, resulting in a torque balance equation that is similar to the force balance of linear springs vs. vertical electrostatic forces in a 4T or 6T relay. The details of torsional force calculations for certain MEMS structures have been well summarized in [35], and specific experiments with seesaws are discussed further in [36, 37].

The left and right ends of the seesaw have channels mounted on them which can contact a drain source pair. When the torsional electrostatic force overcomes the torsional spring the MEMS structure experiences a pull-in effect and the channel is brought into contact with the drain/source pair on the side of the seesaw where the force is applied. This causes the other channel to be moved further away from its drain/source pair and its body. Thus, a seesaw functions like a pair of relays where only one is allowed to be in the *on*-state at any given time. Accordingly, a seesaw is represented in schematics as a pair of relays which share a gate contact. A small triangle is added to the diagram to indicate that the device is a seesaw and not just a pair of 4T relays.

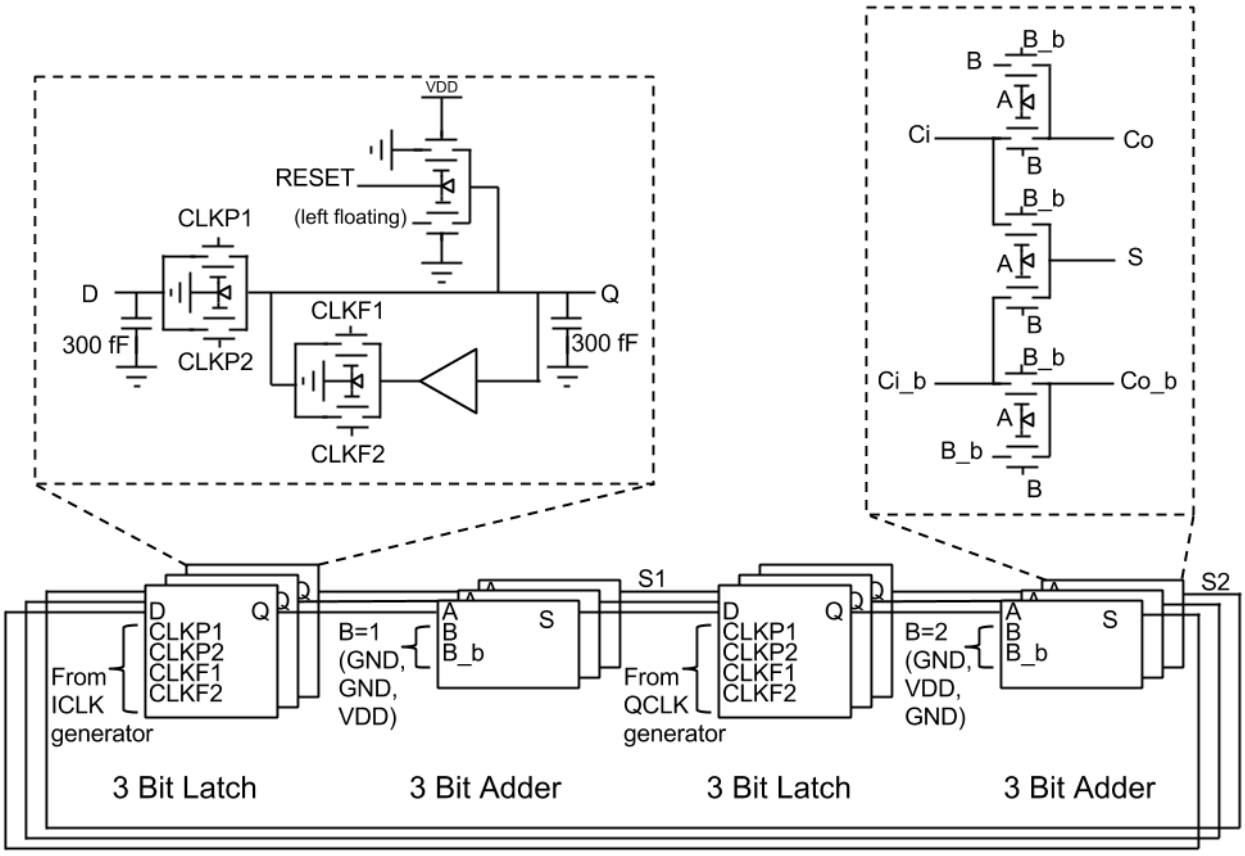
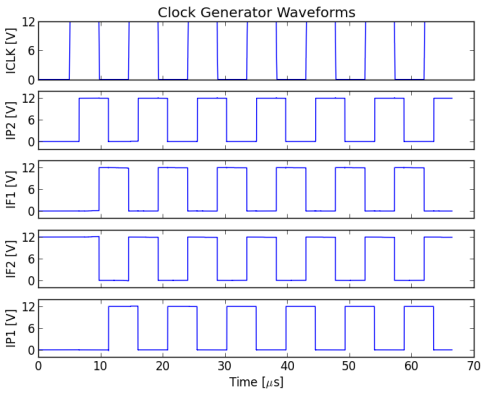


Figure 3.18: Schematic of simulated system. ICLK and QCLK generators are instances of the clock generator in Figure 3.16 driven by quadrature clocks as shown in Figure 3.19.

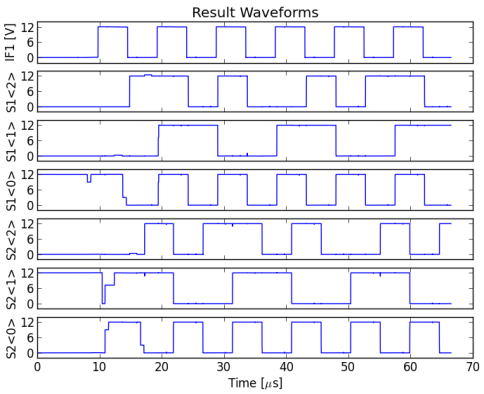
The schematics of seesaw-based elements for a demonstration pipeline appear in Figure 3.18. Specifically, the figure shows a two phase relay pipeline latch, an adder, a clock generator, and their arrangement into an accumulator. The circuit featured in that schematic was simulated and the results appear in Figure 3.19.

These schematics feature minor modifications from their earlier appearance in the chapter. The input and output of the seesaw based latch have had fixed capacitors added to them in order to smooth out spurious voltage transients on the state nodes of the device. In addition, two seesaws have been added to initialize the pipeline to a known state before running the simulation.

These results show the non-overlapping clocks generated by local clock sources and the successful accumulation of values on the sum nodes of the output circuit. The circuit is configured to add one and two on alternate clock phases to the running sum in the accumulator. Thus, the sum on any particular node of the circuit should increase by 3 each time



(a) Clock generator input and output.



(b) SUM results (S1 and S2) appearing on the output nodes.

Figure 3.19: Simulated results of a two phase pipeline with local clock generation.

it transistions. This can be seen in both the $S1$ and $S2$ outputs. The output logic levels feature some glitches at the start of the simulation, which represent drive fights between D and the stored data as the motion of the seesaws first begins. These glitches can't cause data errors because the feedback devices that store the old, incorrect state all transition to the *off*-state before the forward device does.

These simulations confirm that it is possible to build pipelines of relay based logic with minimal timing and area overhead. This is an important step towards building VLSI systems out of mechanical logic, but the question of how to build memory still remains.

Chapter 4

NEMory

Building memory in a mechanical VLSI system poses challenges because the density of memory is crucial and each individual relay in a mechanical system is many times larger than its CMOS counterpart. However, careful device and circuit design can lead to mechanical memories which preserve density even in highly scaled processes. This chapter discusses the challenges of mechanical memory, the current state of the art, and presents a co-optimized device and circuit which can improve the density of mechanical systems.

4.1 Challenges for Mechanical Memory

Implementing CMOS-style 6T SRAM in a relay technology would result in poor performance. A standard CMOS SRAM cell contains six devices: two access devices and a self-staticizing loop of inverters. This is pictured in Figure 4.1. If this were ported to relays there would be a stiff delay penalty, and the area would be large: writing the cross-coupled inverters would require two mechanical delays, and each relay is about the size of twenty CMOS devices in an equivalent technology node. Even worse, the sharp, hysteretic non-linearity of relays cause the cell to hold its state in the case of a drive fight between the wordline and an inverter: the voltage would fall near half the supply voltage which is in the middle of the relay's hysteretic region.

A SRAM cell similar to the CMOS 6T cell, but optimized for relay operation appears in Figure 4.2. Staticization is achieved in this circuit using a buffer feeding back on itself rather than two inverters. An additional write-assist device is included to break the feedback loop and allow data onto the buffer. This structure only needs to be accessed on the input side of the buffer in order to write it – the input side can't cause a drive fight when the feedback is broken and the input side obviously controls the buffer state. As a result, the total number of devices and metal lines in this cell is smaller than a CMOS SRAM, but the area penalty of relays still makes this cell much larger than a CMOS cell.

The density of memory can be improved further by leveraging the extremely low leakage characteristics of relays to build long-term DRAM storage. DRAM typically requires refresh

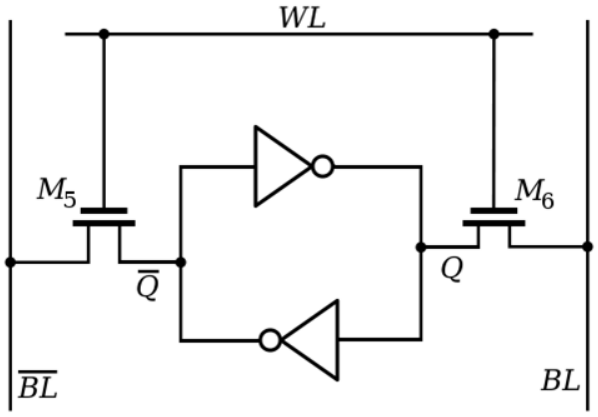


Figure 4.1: CMOS SRAM Cell. BL is the bit line and $\overline{BL}$ is its complement, WL is the word line, $M5$ and $M6$ are names for the pass transistors, Q is the stored bit and $\overline{Q}$ is its complement.

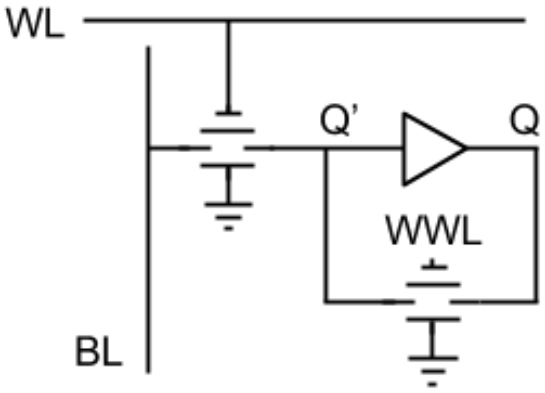
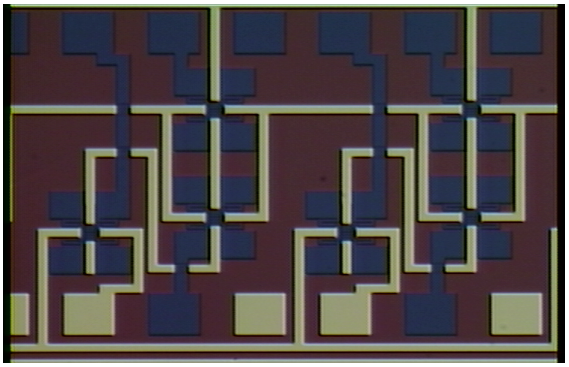
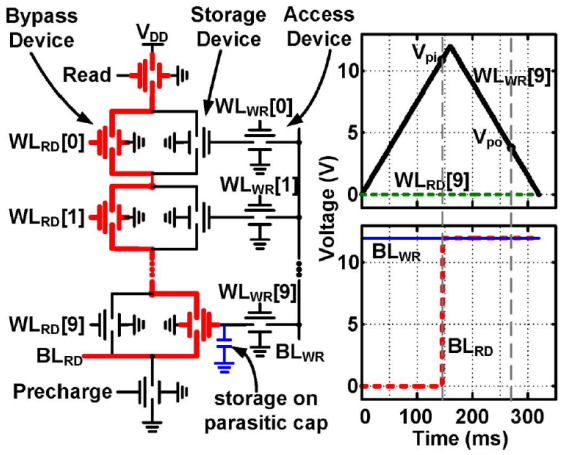


Figure 4.2: Relay SRAM. BL is the bit line, WL is the word line, WWL is an additional write word line which is asserted when the word is being written, Q is the stored bit and Q' is the bit being driven from or to the word line.



(a) Micrograph.



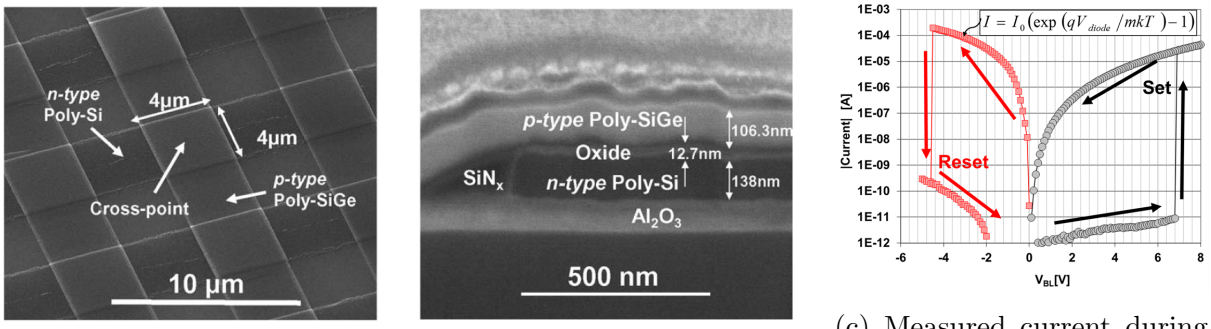
(b) Schematic and measured waveforms.

Figure 4.3: Relay DRAM implmented on CLICRK1.

because the nodes of the circuit lose charge to the junctions in the CMOS access device and the deep junction capacitor. However, relay based technologies have extremely low leakage in both the devices and capacitors, and this makes DRAM especially appealing for longer term storage in relay systems. Further, it has been shown that hypothetical, scaled cantilever DRAM can achieve high memory density and low energy consumption [38]. A proof of concept Relay DRAM was built on the CLICKR1 test chip. A schematic, micrograph, and measured traces from that circuit appear in Figure 4.3.

The CLICKR1 DRAM circuit works like NAND flash. Each cell is comprised of a storage device, an access device and a bypass device. The state of the cell is indicated by charge stored on the body of the storage device. The body of that device is connected to or disconnected from the write bit line by the access device, which is controlled by the write word line. During read, the read word line closes every bypass device except the device for the word of interest. By having the bypass devices default to an on state, this read delay can be made quite quick, since it is a *turn-off* relay delay rather than a *turn-on* relay delay. The read bit line is composed of relays connected drain to source: the conducting bypass devices in every word except the word of interest, and the storage device in the word of interest. If charge is stored on the body of the storage device, then the device will be shut and the read bit line will be conductive. This conductivity is tested by trying to charge a pre-discharged output node through the read bit line.

Though the CLICKR1 test circuit demonstrated the functionality of the relay DRAM, it was unable to explicitly show the retention characteristics of the device. The storage node of the device needed to be externally wires, which meant the leakage was set by the surface leakage of the test PCB – significantly higher than leakage through a relay – and the parasitic capacitance was set by the test probe. The leakage and the capacitance were both artificially enhanced, which made it impossible to measure the characteristics of the cell itself.



(a) SEM top view.

(b) SEM side view.

(c) Measured current during Set and Reset cycle.

Figure 4.4: Micrograph and waveform from the prototypical NEMory in [39]. Images reproduced from [39].

Even though the relay DRAM reduces the device count per cell, its area is still at best comparable to CMOS. The optimistic device layout in [38] was used to propose memory area that had area parity with CMOS arrays, but the area is significantly worse than CMOS if you analyze the footprint of a memory cell using the scaled devices from [17]. The large size of a single device makes it difficult to meet CMOS in density without significant modifications to the design of the mechanical switching element.

4.2 Introduction To NEMORY

Other mechanical memories have been demonstrated which show the promise of scaling more aggressively to small scales. Notably, the nano-electro-mechanical memory (NEMory) demonstrated in [39] provides an interesting launching off point for a deeply scaled mechanical memory. This device relies on electrostatic force to pull-in: a sufficiently high voltage between the BL and WL will deflect the WL into contact with the BL. The deflected WL experiences a spring force pulling it back upward toward its original position. It also experiences Van Der Waals forces pulling it down towards the BL surface and a small "built-in" electrostatic force based on the accumulation of space-charge in the metal-semiconductor $BL-WL$ junction. The spring force must be engineered to be greater than the Van Der Waals force in order to make the device both readable and writable: if the spring force is smaller than the Van Der Waals force then the device will be permanently stuck shut after it closes once. Another control know is the small electrostatic force which can be modulated by applying a reverse bias to the $BL-WL$ junction, so if the spring force is greater than the Van Der Waals force but less than the sum of Van Der Waals and space-charge electrostatics then the memory is non-volatile. This effect is demonstrated in [39], data from this early NEMory device is reproduced in Figure 4.4.

The excellent idea in [39] is using a simplified mechanical structure – a clamp-clamp beam

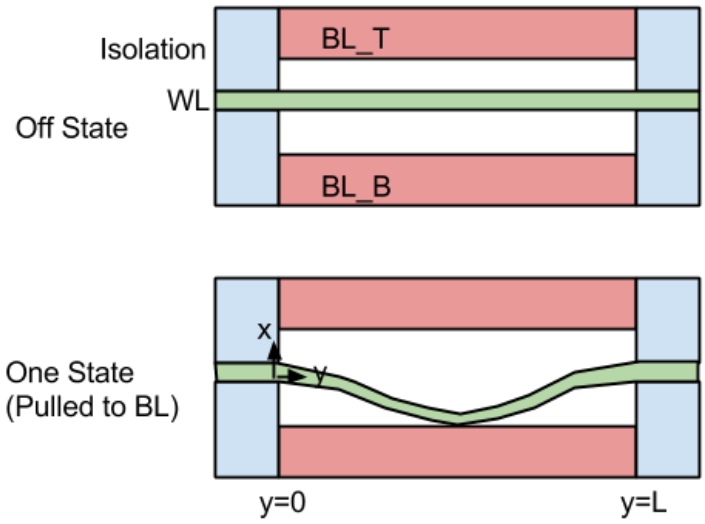


Figure 4.5: Modified NEMory structure with active pull-off. BL_T is the top bit line, BL_B is bottom bit line, which stores the complement of the bit line, WL is the word line. Isolation is an insulating material that is mechanically anchored to the substrate. The device has three states: *zero*, *one* and *off*. In the *one* state WL is in contact with BL_T, in the *zero* state WL is in contact with BL_B, and in the *off* state the WL is not in contact with either BL.

in this case – to reduce the number of features required to draw a single mechanical element. As Figure 4.4 shows, the device can be made of a single 1Fx1F square of metal contact and an additional 1F given over to separation from the next device, making for a 2Fx2F cell. However, the forces acting on this version of NEMory are of vastly different scales, and the electrostatic force required to hold the device closed is very small compared to the spring force and electrostatic force. In order for the device to operate, it must be fabricated so that the electrostatic and spring force are almost exactly the same, to within the tiny margin of the built-in electrostatic force. This causes low device yield for this design.

The mechanical memory cell pictured in Figure 4.5 avoids this problem. The device is very similar to the prototype device, but it relies on having two bit line electrodes to deactuate a word line that has deflected into the surface. Like the prototype device, this new device is pulled shut by electrostatic forces between a bit line electrode and the word line electrode. The electrostatic force the word line electrode to deflect into contact with the bit line electrode where it is held in place by Van Der Waals forces and forms a Schottky diode. This contact is non-volatile because the device is engineered such that Van Der Waals forces are bigger than the spring forces restoring it to its original position. However, the device differs from the earlier prototype in how it is removed from the non-volatile state. A voltage is applied between the opposite bit line electrode and the word line to deactuate a device stuck to one bit line. This bias applies an electrostatic force that overcomes the Van Der Waals force and pulls the word line into contact with the opposite bit line electrode.

When the WL is in contact with BL_B the device is said to be in the *zero* state, and when it is in contact with BL_T the device is said to be in the *one* state. If the WL is not contacting any BL then it is said to be in the *flat* or *off* state.

It's natural to compare this device to the seesaw relay from Chapter 3. Both devices have two fixed electrodes which deform a moving electrode to two positions, and the influence of the fixed electrode that is not currently in contact with the moving electrode is diminished by an an increased gap size. There are, of course, differences in how the electrical force evolves as the device moves because of their different geometries. However, there is one more fundamental difference between the two: the NEMory is designed for non-volatile operation while the seesaw is always actively driven. Removing all gate-to-body voltages from a seesaw will cause the device to settle to a flat state. Not so for a NEMory, which will remain stuck in its most recent state by Van der Waals forces.

4.3 Analyzing NEMory

Mechanical Modeling

The voltage required to create the "pull-off" electrostatic force determines if this element can be included in VLSI systems. If this device were to be integrated in a modern CMOS process, then the breakdown voltage of the CMOS devices would limit the total voltage that could be applied to the device. In a relay system, the applied voltages could be more flexible because relay drain/source voltages are independent of gate/body voltages and relays generally have a high voltage tolerance. These two properties make it relatively cheap to build relay level shifters that could drive a high voltage NEMory array. The CMOS case is more restrictive, so this design will attempt to meet the constraints posed by a 14nm process. Typical processes have drain-to-source breakdown voltages of 0.8V on logic devices and 1.8V on thick-oxide, high-threshold voltage devices. Using logic devices is more desirable for density reasons.

Predicting the switching voltages of NEMory requires a mechanical model of the word line element. The force balance equation for the word line is given by:

$$F_{WL}(x, V_{up}, V_{dn}) = F_{el,up}(x, V_{up}) - F_{el,dn}(x, V_{dn}) + F_{vdw,up}(x) \\ - F_{vdw,dn}(x) - F_{surf,up}(x) + F_{surf,dn}(x) + F_k(x) \quad (4.1)$$

where F_{WL} is the net force on the word line, $F_{el,up}$ is the electrical force exerted between the word line and the uppper bit line (BL_T in Figure 4.5), $F_{el,dn}$ is the force exerted between the word line and the lower bit line (BL_B in Figure 4.5), $F_{vdw,up}$ is the Van Der Waals force between the word line and the upper bit line, $F_{vdw,dn}$ is the Van Der Waals force between the word line and the lower bit line, $F_{surf,up}$ is the contact repulsive force between the word line and the upper bit line, $F_{surf,dn}$ is the contact reupulsive force between the word line and the lower bit line, and F_k is the spring restoring force of the deformed word line electrode. The F_{vdw} and F_{surf} terms are considered in this equation when they were neglected in Chapter 2 because these devices are designed to be much smaller to meet the density and voltage

requirements of memory. Further, the F_{vdw} terms are critical in determining the volatility of the device.

The small scale of these devices requires that the model for computing the electrostatic force be changed relative to Chapter 2. In the earlier chapter, the relay could safely be modeled as a single, flat, plate because its motion was confined to one dimension. Not so with the proposed NEMory, where different components of the beam will deflect different amounts. However, any differential element of the beam can be modeled as a parallel plate, so an integral across the length of the beam can determine the total downward force applied to it. This technique requires knowledge of the mode shape. In [40], the mode shape is approximated as a cosine curve with little error, so this is chosen as the mode shape for the purpose of this model. That is to say, for the NEMory in Figure 4.5, the mode shape can be expressed as:

$$mode(x, y) = x \cdot \frac{1 - \cos(2\pi y/L)}{2} \tag{4.2}$$

where L is the length of the beam and *mode* is the deflection of a point y on the beam when the center has deflected by x.

This mode can be used to calculate the electrical force applied to the differential element from y to $y + \Delta y$ for a deflection x. The electrical force on the differential element is simply given by a standard parallel plate model:

$$F_{el,up}(x, y, y + \Delta y, V_{up}) = \frac{V_{up}^2 \epsilon_0 W \Delta y}{2(g_0 - mode(x, y))^2}, \tag{4.3}$$

where ϵ_0 is the permittivity of free space, W is the width of the beam, V_{up} is the voltage between the word line and the upper bit line, and g_0 is the nominal gap between the word line and the bit line at zero deflection (i.e.: the gap when x=0). W is assumed to be constant across the beam because the maximum deflection gap, g_0, is assumed to be significantly shorter than L. Integrating this expression with respect to y will reveal the total upwards electrical force on the beam: $F_{el,up}(x, V_{up})$. That integration is carried out numerically. Similarly, $F_{el,dn}(x, V_{dn})$ is found by integrating:

$$F_{el,up}(x, y, y + \Delta y, V_{dn}) = \frac{V_{dn}^2 \epsilon_0 W \Delta y}{2(g_0 - mode(-x, y))^2}, \tag{4.4}$$

where V_{dn} is the voltage between the word line and the lower bit line. This is the same expression as the upward force except for referring to a different voltage, V_{dn} instead of V_{up}, and using the opposite of the deflection, $-x$ instead of x.

The mode can be used to find $F_{vdw,up}(x)$, and $F_{vdw,dn}(x)$ in the same way. The Van Der Waals force on a differential element is given by:

$$F_{vdw,up}(x, y, y + \Delta y) = W \Delta y \frac{A_{12}}{6\pi} \frac{z_0}{d^3(d + z_0)} \quad \text{where} \quad d \equiv g_0 - mode(x, y) \tag{4.5}$$

where A_{12} is the Hamaker constant of the surfaces and z_0 is the Van Der Waals screening distance of the materials (typically a few Å), and the intermediate variable d is the separation between the surfaces. This expression was derived from material in [41], and explicitly includes the $d+z_0$ term for screening of Van der Waals forces at wide separation. The downward Van Der Waals force uses the same expression with d redefined to be $g_0 - mode(-x, y)$. Again, the total force can be found for these expressions by numerically integrating across y.

The surface repulsion forces receive the same treatment. A discussion of combining the models of Van Der Waals and surface repulsive forces appears in [42], where a derivation from the Lennard-Jones potential reveals the following expression for pressure between two surfaces:

$$P_{lj} = \frac{A_{12}}{6\pi d_0^3}\left[\left(\frac{d_0}{d}\right)^3 - \left(\frac{d_0}{d}\right)^9\right] \tag{4.6}$$

where P_{lj} is the Lennard-Jones pressure, d_0 is the initial separation between the surfaces, and d is the separation between surfaces as above. This expression includes both a Van der Waals term, which omits the screening factor introduced in [41], and a surface repulsion term. Differential expressions for $F_{surf,up}(x)$ and $F_{surf,dn}(x)$ for the NEMory structure can be found by removing the Van der Waals potential from the expression and multiplying by the area:

$$F_{surf,up}(x, y, y+\Delta y) = W\Delta y \frac{A_{12}}{6\pi d_0^3}\left(\frac{d_0}{d}\right)^9 \quad \text{where} \quad d \equiv g_0 - mode(x, y). \tag{4.7}$$

$F_{surf,dn}(x)$ can be found by reversing the d variable to be $g_0 - mode(-x, y)$ like before, and the total force can be found by numerical integration over y.

The spring force is represented using a cubic spring model derived from [13]. Senturia describes two different cubic spring models that are relevant depending on the aspect ratio of the deforming structure: a model for clamp-clamp beams and a model for plates. Each of these models consists of a linear constant, k, and a cubic constant k_3, such that the overall force expression is given by the familiar duffing spring:

$$F_k(x) = kx + k_3 x^3. \tag{4.8}$$

The values for k and k_3 were interpolated from the beam and plate constants based on the ratio of W and L. The linear plate constant is given by

$$k_p = \frac{2\pi^4}{3}\frac{E}{1-\nu^2}W\left(\frac{t}{L}\right)^3 \tag{4.9}$$

where E is the Young's modulus of the material, ν is the Poisson ratio of the material, and t is the thickness of the material. The cubic plate constant is

$$k_{3p} = \frac{\pi^4}{4}\nu_f\frac{E}{1-\nu}W\frac{t}{L^3} \quad \text{where} \quad \nu_f = \frac{(7-2\nu)(5+4\nu)}{32(1+\nu)}. \tag{4.10}$$

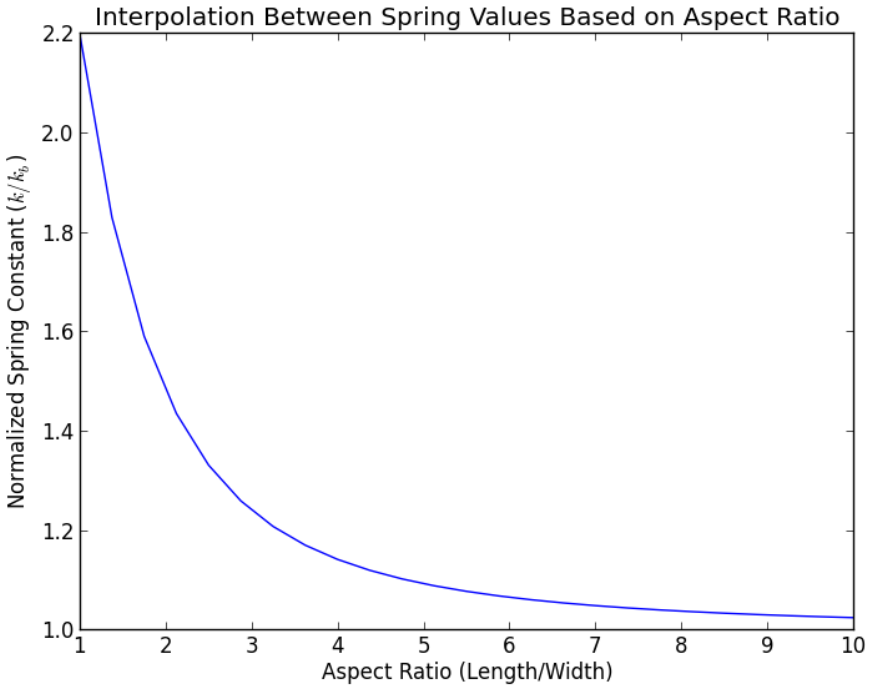


Figure 4.6: Interpolation between linear plate and beam spring constants based on aspect ratio of structure. The value at an aspect ratio of one is equal to k_p. The value at large aspect ratios is k_b.

The linear beam constant is

$$k_b = \frac{\pi^4}{3} EW \left(\frac{t}{L}\right)^3, \tag{4.11}$$

and the cubic beam constant is

$$k_{3b} = \frac{\pi^4}{3} EW \frac{t}{L^3} \tag{4.12}$$

The linear and cubic coefficients were interpolated based on the log of the aspect ratio using a hyperbolic tangent. The equation describing this interpolation is

$$k = k_p - (k_p - k_b) \tanh\left(\log\left(L/W\right)\right), \tag{4.13}$$

and Figure 4.6 shows the interpolated values for normalized spring constants. The cubic coefficients were interpolated using the same formula where k, k_p and k_b are replaced by k_3, k_{3p} and k_{3b}. When L is equal to W this interpolation returns the value k_p and when L/W is large this returns k_b.

These force expressions can be combined in the force balance equation, Equation 4.1, to make predictions about relay behavior. For example, it is possible to test for non-volatility

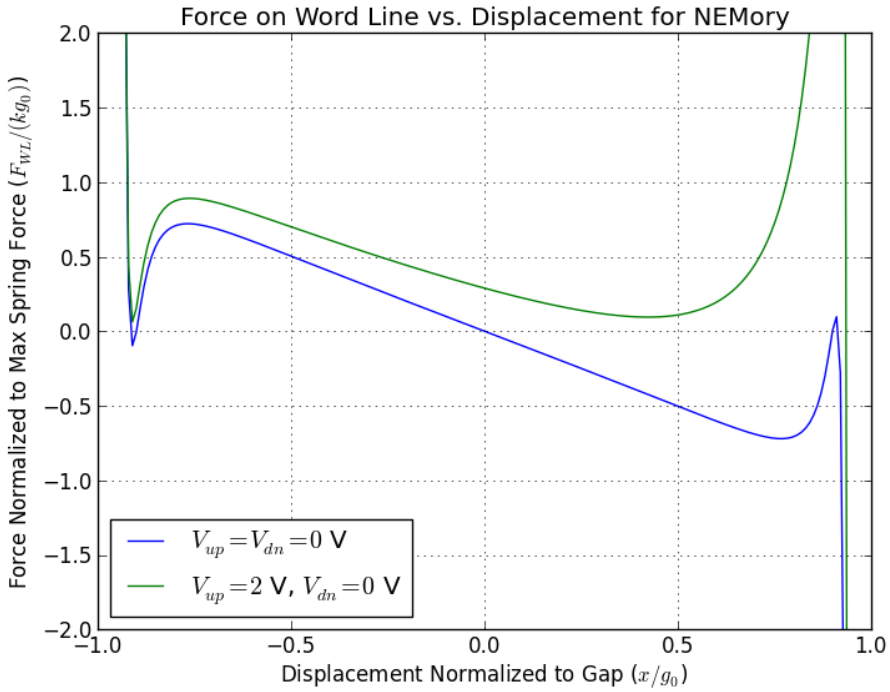


Figure 4.7: Force on the word line of an example NEMory device as a function of displacement. Positive force represents force in the upwards (towards positive x) direction.

by plotting the force experienced by the word line for every value of x with V_{up} and V_{dn} set to zero. A plot of an $F_{WL}(x, 0, 0)$ seesaw is featured in Figure 4.7, and the figure shows a non-volatile system. This can be seen by examining the roots of the force equations and looking for stable equilibria. A stable equilibrium will have a negative slope so that increases in x will result in a negative force that opposes the change in position. The equilibrium at zero is stable, it represents the resting state of the NEMory, and the most extreme left and right equilibria are stable, representing non-volatile zero and one states. The zero and one state equilibria are only present if the negative dip in force caused by the Van der Waals interaction overcomes the spring force and causes the net force to become negative. This makes intuitive sense: if the spring is too strong then it's impossible for the relatively weak Van der Waals force to hold the device shut.

Applying voltage to the NEMory structure causes the the curve to move upwards or downwards as the $F_{el,up}(x, V_{up})$ and $F_{el,dn}(x, V_{dn})$ contribute to the expression. This can eliminate the *zero* or *one* state equilibrium, which causes the structure to experience forces that pull it towards the *flat* or opposite state. An example curve with voltage applied appears in Figure 4.7, and the zero equilibrium at -0.9 has been eliminated such that the force always pulls the device towards the *one* state. The point at which an equilibrium is eliminated represents a pull-in voltage for the structure, but unlike the standard relay there

are multiple possible pull-in voltages. Depending on the relative forces, there can exist a *flat-side* pull-in voltage which represents the voltage needed to move from the *flat* state to the *zero* or *one* state, and that voltage can be different from the *side-side* pull-in voltage which moves the switch from the *zero* state to the *one* state or vice versa.

These voltages can be calculated by separating the force balance equation into displacement dependent and voltage dependent components

$$F_{WL}(x, V_{up}, V_{dn}) = f(x) + g(x)V_{up}^2 - g(-x)V_{dn}^2 \tag{4.14}$$

$$\text{where} f(x) \equiv F_k(x) + F_{vdw,up}(x) - F_{vdw,dn}(x) - F_{surf,up}(x) + F_{surf,dn}(x) \tag{4.15}$$

$$\text{and} \quad g(x) \equiv \int_0^L \frac{\epsilon_0 W dy}{2(g_0 - mode(x,y))^2}. \tag{4.16}$$

Like many of the integrals above, the $g(x)$ integral can be carried out numerically. Note that $g(x)V_{up}^2 = F_{el,up}(x, V_{up})$, i.e. $g(x)$ is only the constant and x dependent part of $F_{el,up}$.

This separation of components leads to a convenient way to find the pull-in voltages of the NEMory if certain facts about the operation are observed. First, at a pull-in point, the force applied to the word line will be zero because an equilibrium will be crossing the zero axis as the curve moves upward. Second, during operation the NEMory is only going to be pulled in one direction at a time, so assuming the driver circuits do a good job, either V_{up} or V_{dn} can be set to zero when calculating the pull-in voltages. Applying these facts the equation becomes

$$0 = f(x_{pi}) + g(x_{pi})V_{up,pi}^2 \tag{4.17}$$

$$V_{pi} = \sqrt{f(x_{pi})/g(x_{pi})} \tag{4.18}$$

where x_{pi} is the displacement at which pull-in happens and $V_{up,pi}$ is the upward pull-in voltage.

The next challenge is finding find x_{pi} and $V_{up,pi}$ based on f and g, which are known. This can be achieved by plotting

$$V_{up,eq}(x) = \sqrt{|f(x)/g(x)|} \tag{4.19}$$

where $V_{up,eq}(x)$ represents the amount of voltage that needs to be applied for the NEMory to reach equilibrium at any given x. The absolute value is included in this equation because $f(x)/g(x)$ is sometimes negative despite representing a squared value. This is because $f(x)$ is a force and contains information about the direction the device will move in its sign, which is irrelevant to determining how much voltage is needed to move the device to a given point.

Equation 4.19 is plotted in Figure 4.8, and the curve has local maxima. Raising the voltage above a local maximum means that the voltage is higher than is required to bring the system into equilibrium for points to the right of the maximum. This rightward directionality arises because this plot considers only the upward force, and the upward electrical force can only increase x since it is always positive. The condition of having more electrical force than is needed to bring the system into equilibrium is exactly what is necessary for an

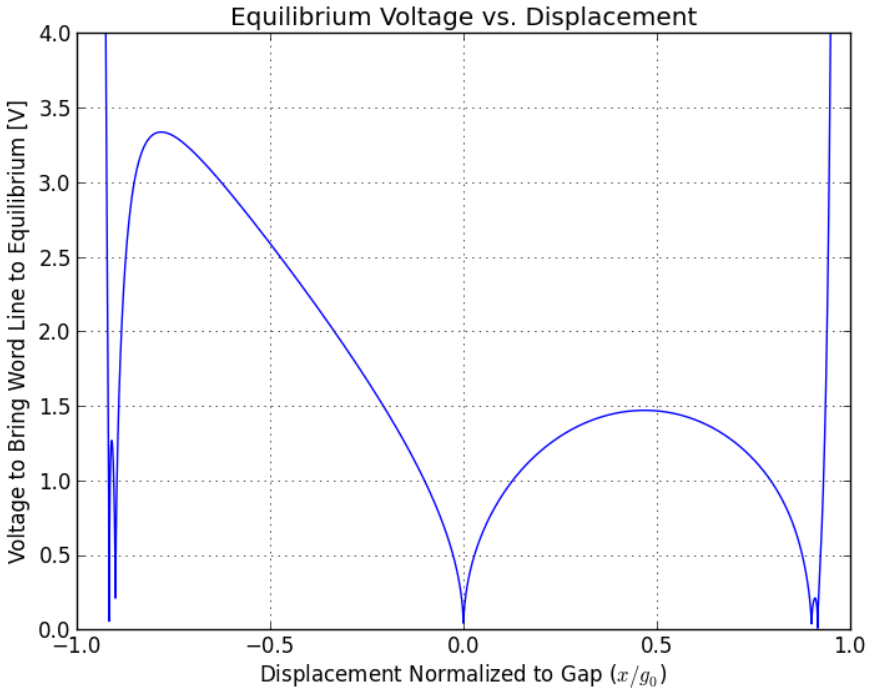


Figure 4.8: The voltage required to bring the word line of an example NEMory device into equilibrium for any value of displacement, calculated by separating the force balance equation into displacement dependent and voltage dependent components.

electrostatic pull-in non-linearity, so the NEMory's $V_{pi,up}$ are the maxima of the $f(x)/g(x)$ curve which are located to the right of stable equilibria. The x_{pi} at which the pull-in occurs are at the locations of those maxima. Four maxima are present, but the two located at normalized displacements of -0.9 and +0.4 represent the points at which the word line leaves stable equilibria and start to move right. The left maximum at approximately 1.25V is the side-side pull-in voltage and the right maximum at 1.5V is the flat-side pull-in voltage. The equivalent plot for the downward force is the same, but mirrored about the $x = 0$ line because it relies on $g(-x)$ instead of $g(x)$.

In this example device, the flat-side pull-in voltage is higher than the side-side pull in voltage because of the relatively strong spring force. As a result, the flat-side pull-in voltage sets the voltage that the driver needs to supply. In general, the driver needs to supply the higher of the flat-side and the side-side pull-in voltages, which will be referred to as the NEMory operating voltage.

One additional, non-physical force is included when simulating this model as a convergence aid. A very steep exponential force that pushes the solution back towards the *flat* state is introduced for values of x which are signicantly greater than x_{pi}.

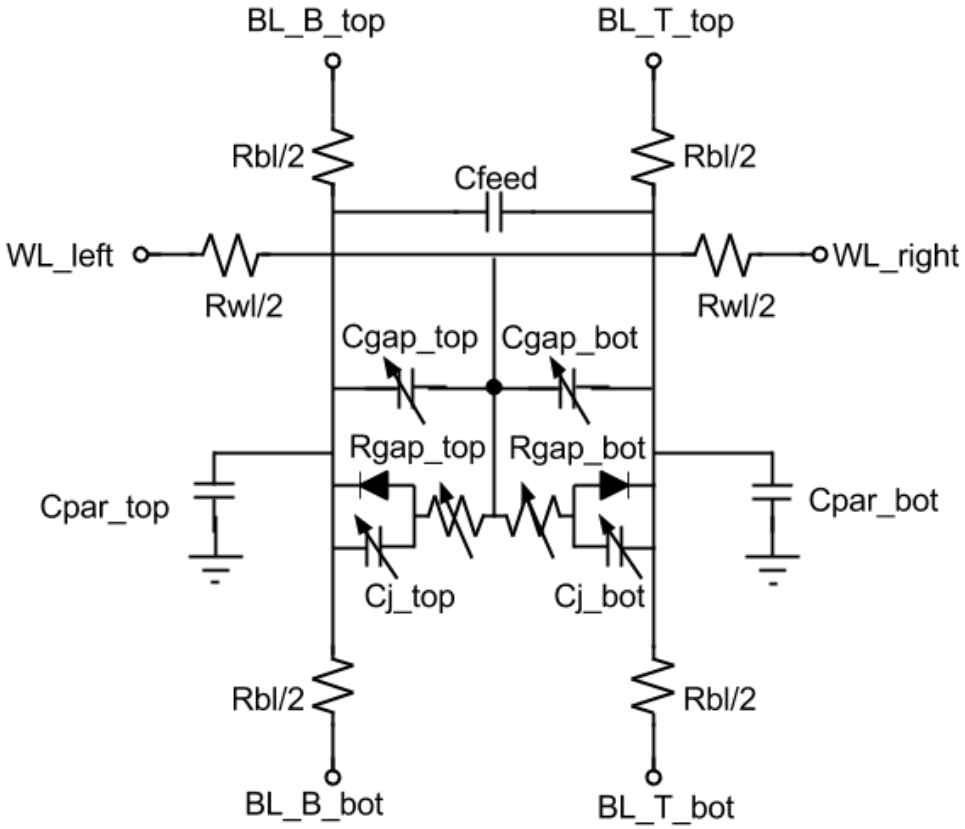


Figure 4.9: Electrical model of NEMory cell.

Electrical Modeling

An electrical model of the NEMory is important for determining the power consumption and delay of memory circuits made from NEMory. A model appears in Figure 4.9.

The resistors in the electrical model represent the resistance of the word line and the bit line. They form a T model with the electrical diode contact in the middle of the cell. The value of the resistors is calculated from the dimensions of the cell and the standard resistivity formula:

$$R_{BL} = \frac{\rho_{BL} L_{BL}}{t_{BL} W_{BL}} \text{ and } R_{WL} = \frac{\rho_{WL} L_{WL}}{tW}, \tag{4.20}$$

where R_{BL} and R_{WL} are the resistance per NEMory cell of the bit and word lines, ρ_{BL} and ρ_{WL} are the resistivity of the bit and word lines, L_{BL}, W_{BL} and t_{BL} are the length, width and thickness of the bit line, and L_{WL} is the electrical length of the word line. t and W are inherited from the mechanical model, since the word line thickness and width are used in both electrical and mechanical calculations. However, the word line length values used for electrical and mechanical calculation, L_{BL} and L, are different because the anchor in the layout prevents a portion of the word line from moving.

The air gap capacitors in the electrical model represent the capacitors which create the electrostatic force, and thus are computed using the same technique of integrating parallel plate models over the beam's mode shape. In particular, a differential element of the beam between y and $y + \Delta y$ has a capacitance $C_{air}(y, y + \Delta y)$ given by a standard parallel plate equation:

$$C_{air}(x, y, y + \Delta y) = \frac{\epsilon_0 W \Delta y}{g_0 - mode(x, y)}. \tag{4.21}$$

That expression can be numerically integrated over y to find $C_{air}(x)$, which is used for C_{gap_top}. $C_{air}(-x)$ is used for C_{gap_bot}.

The gap resistance models the contact between the word line and the bit line. Ideally, it is infinite when x indicates the word line is far from the surface, and it is small when x is near g_0. Like modeling the surface force in Chapter 2, this infinite non-linearity is not suitable for circuit simulators and is replaced by a tanh based step function. As expected, the top and bottom resistors are functions of x and $-x$ respectively.

The diode and junction capacitor models are the same for the top side and bottom side of the device. The diode model is

$$I_{diode} = I_0 \exp\left(\frac{qV_{diode}}{nkT}\right), \tag{4.22}$$

where I_{diode} is the current through the diode, I_0 is the dark current of the diode, n is the non-ideality factor of the diode, q is the charge of an electron, k is Boltzman's constant and T is temperature. I_0 was found by extracting the dark current from the measured diode in [39] (12nA) and dividing it by the ratio of the contact area in the prototype device to the contact area of this design (1300). The contact area of this design was taken as $W \cdot L/6$ as in [40]. n was assumed to be the same as [39] (5).

The junction capacitor model is a Schottky diode junction:

$$C_j(V_j) = \frac{\epsilon_{Si} W L/6}{\sqrt{2\epsilon_{Si}(SBH - V_j)/(qN_d)}}, \tag{4.23}$$

where C_j is the junction capacitnace, V_j is voltage across the junction, ϵ_{Si} is the permitivity of silicon, SBH is the Schottky barrier height (0.82V), and N_d is the number of dopants in the silicon ($10^{20}/\mathrm{m}^3$). The denominator of this expression represents the depletion width of the Schottky junction.

The remaining capacitors: C_{feed}, C_{par_top} and C_{par_bot} represent parasitic capacitance between the two bitlines and the rest of the world. They are calculated using a coarse parallel plate approximation based on the area of the bit lines and the minimum separation to nearby metal lines.

4.4 NEMory Design

The electrical and mechanical models for NEMory can be used to select critical dimensions for the device design, operating voltages for the array, and driver circuits for the array. These

design decisions are examined below, and are separated into device design decisions, circuit and array design decisions for the write operation, circuit and array decisions for the read operation, and layout decisions.

Device Design

The final design of the NEMory device consists of picking the materials for use in the device and setting the dimensions of the device. Titatium Nickel (TiNi) is selected as the word line material because of its low Young's Modulus and high strain limit [40], and heavily doped (10^{20} dopants/cm^3) Poly-Silicon is selected as the semiconducting bit line material. The air gap of the NEMory device is set to 2nm to prevent tunneling leakage across the air gap [14]. The thickness of the word line is set to the minimum possible thickness for an ALD deposited film, 5nm[43]. The thickness of the bit line is much less constrained, so it is set to a value consistent with low level metal layers in characteristic processes: 300nm.

The NEMory needs to conform to the restrictive design rules of highly scaled processes and also needs to achieve a pull-in voltage which the process can provide without breakdown. The design rules assumed for the NEMory are those of the bottom few metal layers of the 14nm target process. This allows the NEMory to be built on top of active devices. As a result, the minimum feature of the layers which comprise NEMory is 32nm, and because this feature size is small NEMory features are constrained to fall onto a 32nm grid.

The width of the word line should be set to the minimum possible value of 32nm to maximize device density. The mechanical length was picked as 64 nm in order to fit on the 32nm grid while producing a low pull-in voltage. Shorter lengths lead to higher pull-in voltages because the spring force becomes very strong, longer lengths lead to higher pull-in voltages because the spring force is too weak to assist pulling the word line out of the *zero* or *one* states where the Van der Waals force is high.

The Hamaker constant determines relative strength of the Van der Waals forces in the system and it has a very large effect on the pull-in voltage and the desired spring force. The Hamaker constant for a TiNi / micromachined poly-silicon surface was assumed to be A_{12} = 35zJ based on extrapolation from tables in [44] and [45] and the assumption that passivating elements could be introduced to the rough, micromachined surface to engineer the Hamaker constant. This, quite low, value of Hamaker constant needs to be controlled tightly: a 5% increase in the Hamaker constant would increase the side-to-side pull-in voltage above 1.6V. The flat-side pull-in voltage is not affected by the Hamaker constant because the Van Der Waals force falls off quickly as the separation between surfaces increases.

The bit line width should nominally be the same as the word line length to minimize the total footprint of the cell. However, other layout concerns which are discussed later in the chapter necessitate slightly more routing space under each cell, so the width of the bit line is set to 96nm. This is slightly longer than desired mechanical length of the word line. The extra space on the word line layer is taken up by an expanded anchor, and 96nm is used as the electrical length for the purpose of calculating parasitics.

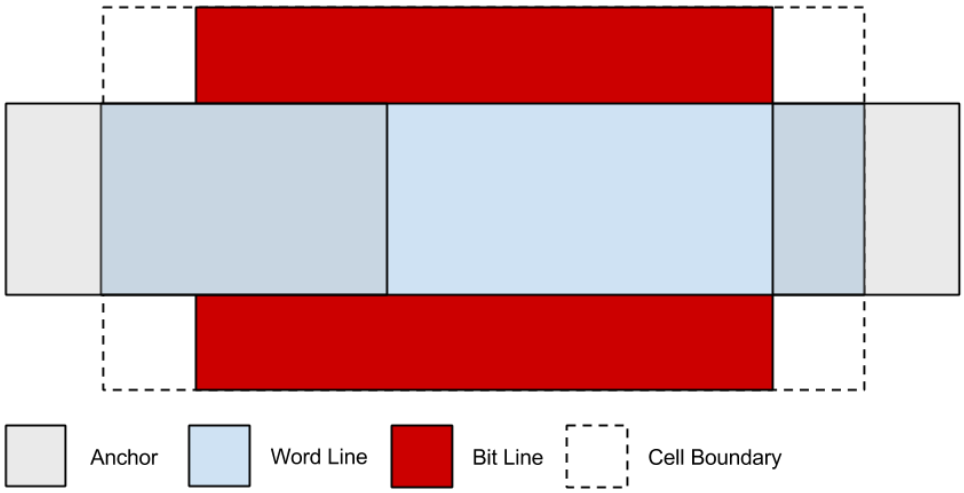


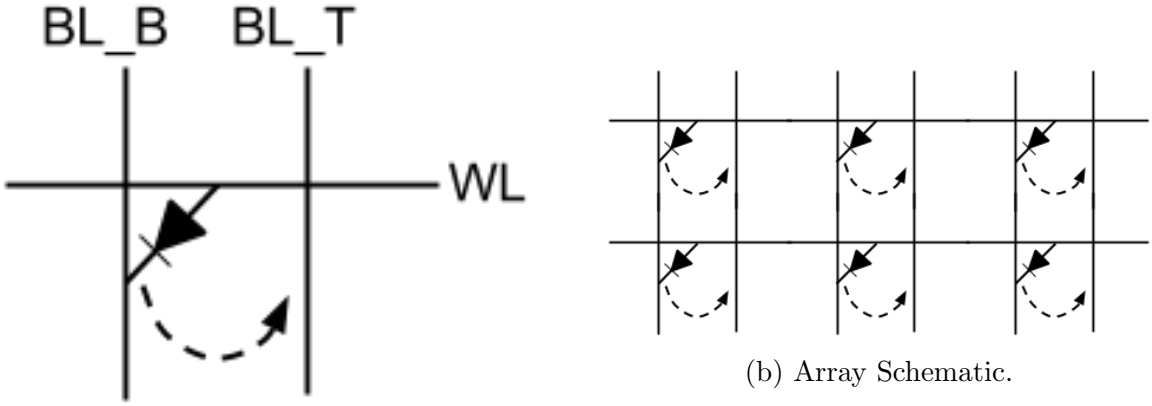
Figure 4.10: Layout for a NEMory cell. Though all layers continue past the cell boundary, the anchors have been explicitly extended since both the left and right anchor shapes are short. By contrast, the word line and bit line shapes extend the entire length of the array.

The dimensions discussed in this section were used to produce the example plots in the previous section: Figures 4.7 and 4.8. Thus the device as designed has an effective minimum operating voltage of 1.5V. This is just under twice the breakdown voltage of of logic devices in the 14nm process. A layout of the device is in Figure 4.10.

Array Design for Writing

The NEMory cell design show in Figure 4.10 is readily tiled by abutting bit line to bit line in the vertical direction and word line to word line in the horizontal direction. This allows for the construction of a memory array. A simple, lumped electrical model of the NEMory cell is useful for determining the array's performance and requirement, so a simple symbol for a NEMory cell appears in Figure4.11a. NEMory cells arranged into an array appear in Figure 4.11b. The symbol evokes a diode attached to the word line and swinging back and forth between the bit lines as a reminder that the metal-semiconductor contact of the NEMory will create a Schottky diode and that it can only be attached to one side at once.

Writing the array requires applying a voltage greater than the operating voltage between the word line and the bit line of targeted cells while preventing any non-selected cells from changing value. To write a cell, the operating voltage is applied the the cell's bit line and the cell's word line is held to zero. Non selected word lines can be preserved by applying a small positive voltage to the non-selected word lines. This makes the voltage difference between the non-selected word line and the bit slightly less than the operating voltage so that the cells are not disturbed. For this array, 200mV was selected as the non-select voltage because it was significantly below the supply voltage (0.8V) and significantly above any of



(a) Symbol.

(b) Array Schematic.

Figure 4.11: NEMory symbol and array.

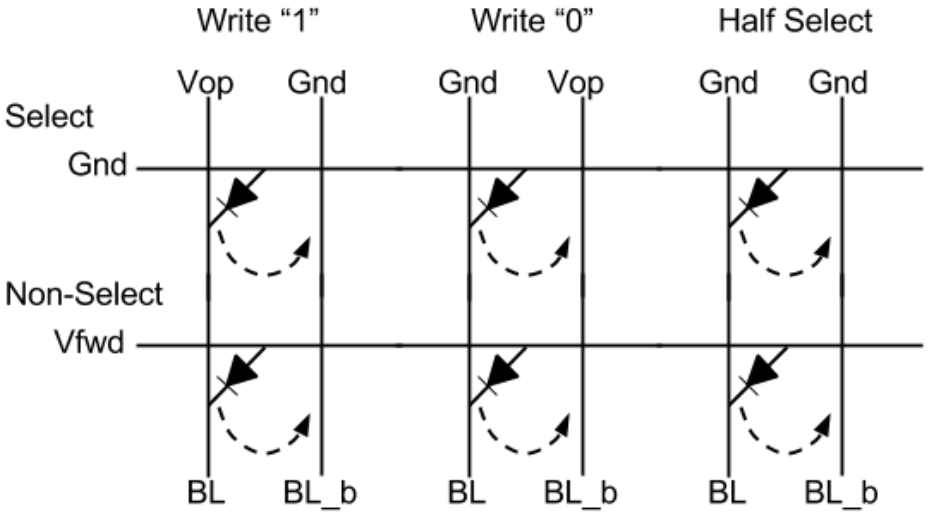


Figure 4.12: A NEMory array configured for writing.

the expected bit line swings so that coupling would not disturb it. Half selected columns can be preserved by applying zero volts to both the top and bottom bit lines. This writing scheme is pictured in Figure 4.12.

One hazard of this writing scheme is the forward bias applied to word lines results in a DC path to ground through forward biased diodes on potentially every word line. If the state of the NEMory in a cell is such that it is connected to the zero potential bit line, then the forward bias used to deselect the line will be forward biasing the cell's diode. If the diode is instead hooked to the bit line biased at the operating voltage, the word line only

sees the reverse leakage current of the diode. As a result the total power loss due to forward biased diodes will be data dependent. Keeping the forward bias as small as noise margins will allow and keeping the writing period short can work to mitigate the power loss posed by this forward bias.

During write, the bit lines need to be driven to the operating voltage, which is twice the core logic level. A level shifter circuit is required to drive this voltage, but the level shifter needs to account for a hidden stability risk when writing to the NEMory. NEM devices which are experiencing an actuation voltage will deform more than those that are not, and that additional deformation represents the storage of additional stress energy in their mode shapes. If the applied voltage is removed too quickly, that stored energy can be converted into enough momentum to break the device free of the Van Der Waals force. As a result, the edge rate seen by any NEMory cells needs to be sufficiently low to prevent loss of data during the discharge of the bit line after a write.

Standard level shifting techniques are very capable of doubling the logic level of 0.8V, which allows them to drive the operating voltage of 1.5V. Adding an RC filter to the output of the level shifter can reduce the edge rate to preserve the array's stability. A sample bit line driver, featuring the level shift and the output filter, is pictured in Figure 4.13. All devices are minimum sized and use the highest available threshold voltage. The three PMOS devices in the pull-up path suppress leakage when the bit line driver is not in use. They are necessary to enable the array's read mode, which is discussed below, and they don't affect performance because the resistance of the array is dominated by the bit line and word line resistance. The RC filter is implemented using 187 dummy NEMory devices, which are biased with 0.8V on the word line to prevent their actuation. The bit-line resistance and the bit-line to word-line capacitances provide the R and C values. These devices fit into the NEMory array with no area overhead, which is explained when discussing layout below.

One such level shifter is needed for each column. The $BLUP\_b$ and $BLDN$ signals can be generated from the data with simple CMOS logic. Word line drivers are related to writing because they are needed to drive the forward bias onto the non-target word lines to prevent actuation. However, they also have other requirements based on reading the array, so they will be elaborated after discussing the array's read mode.

Array Design for Reading

To read the array, the bit lines can be pre-discharged and then made high impedance (by setting $BLUP\_b$ to one and $BLDN$ to zero in the driver), and a reading voltage can be applied to the word line. The reading voltage will cause current to flow through the diode to the bit line to which it is attached, and a sense amplifier can measure the difference in voltage between the bit lines to determine the state of the cell. A schematic representation of this appears in Figure 4.14.

The value of the reading voltage needs to be carefully considered because of the resistance in the NEMory array. Cells which are far from the driver will have significant series resistance along the word line between the driver and the diode, while cells near the driver will look

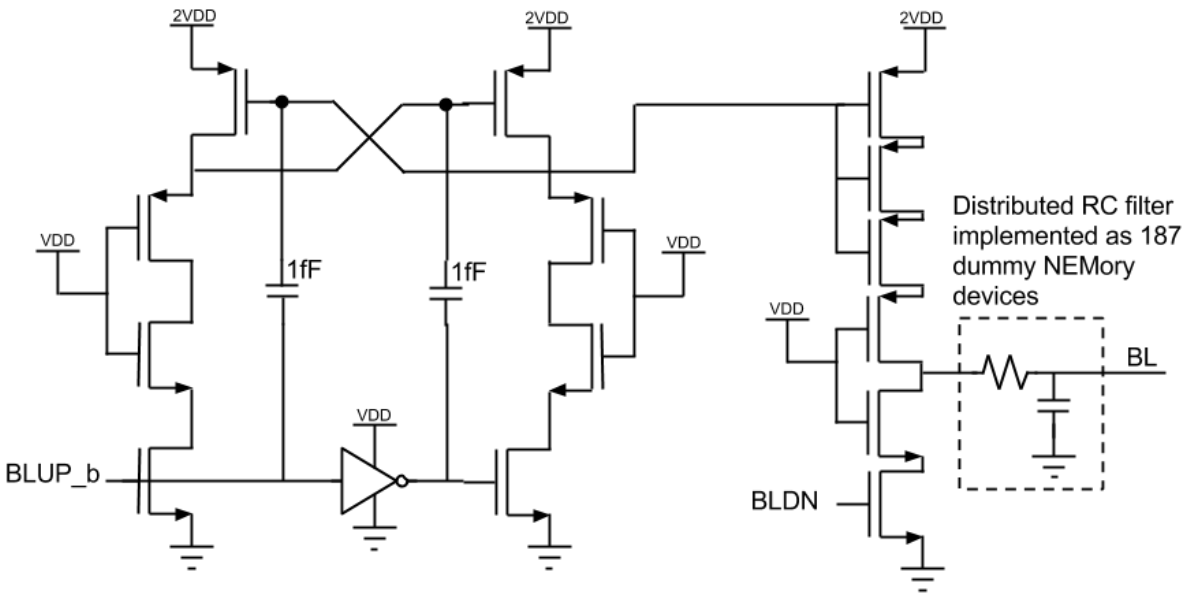


Figure 4.13: A level shifter which is capable of driving the bit line of a NEMory above V_{dd}.

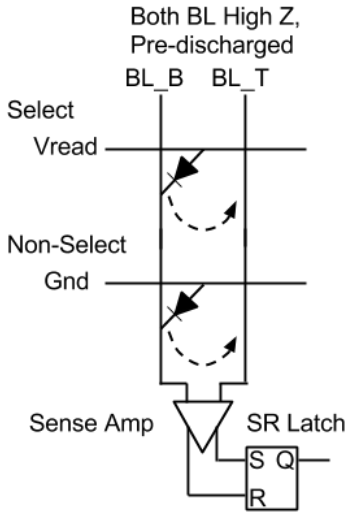


Figure 4.14: A NEMory array configured for reading.

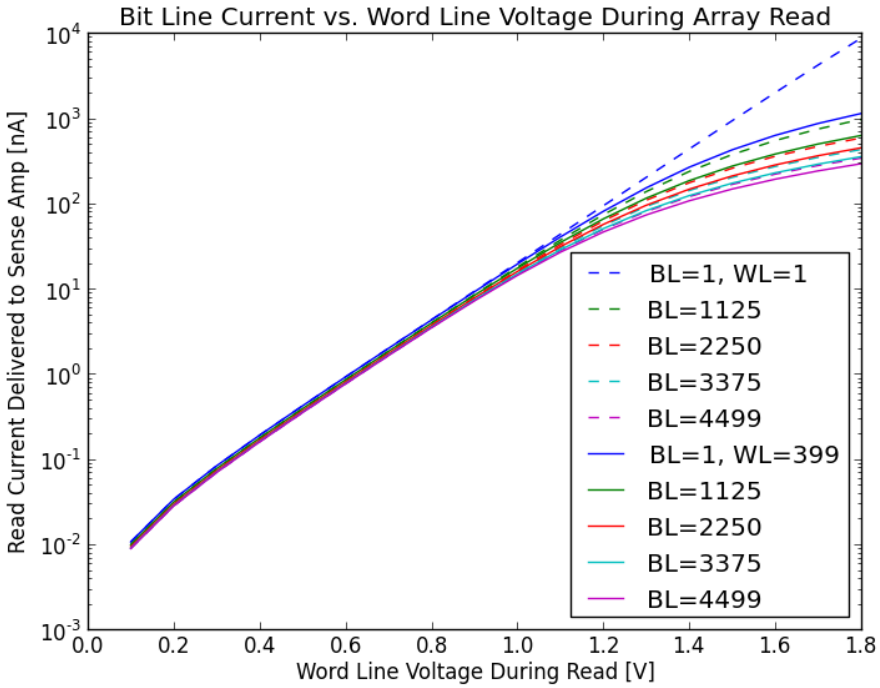


Figure 4.15: Difference in current delivered to the sense amplifier when reading different array locations. Different colored lines correspond to different amounts of resistance on the bit line. There are two lines of each color, which correspond to the sides of the word line close to and far from the word line driver. In most cases, the bit line resistance dominates so the word line location has little affect. At low voltages the diode dominates conduction and all currents fall within 50% of each other.

almost like ideal diodes. This difference could cause different columns to experience different charging rates depending on their proximity to the driver. Different currents would result in more total power being spent each read cycle as the quickest charging bit lines reach higher voltages than the slowest charging bit lines, and different currents would complicate sense amplifier timing.

The difference in cell charging currents is shown in Figure 4.15. The figure reveals that applying a low voltage keeps the charging current difference between different bit lines small. This is because low voltages don't fully turn on the diodes in the NEMory cells, so all of the cell currents are limited by the diode rather than the resistive path they have to pass through. Conveniently, the maximum logic voltage of 0.8V falls at the upper limit of the diode-limited voltage range, and can thus be used as the read voltage.

For the sense amplifier to work properly, the bit line contacted by the target cell needs to rise to a higher voltage than the non-contacted bit line. However, the worst case situation for reading the array makes this somewhat difficult. In the worst case, every cell on a bit

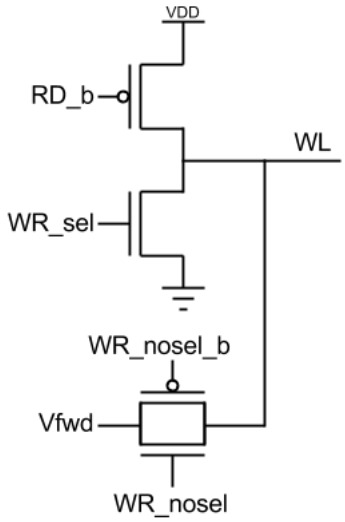


Figure 4.16: Word Line Driver in NEMory array.

line stores the same value, so all of the NEMory diodes are in contact with the same bit line. The diodes on the non-target cells will leak currents backward through their junctions, which reduces the amount of read current through the target cell that goes into charging the sense amplifier capacitors. If the array is big enough and the reverse leakage is high enough (both true of the example array in this chapter) then the reverse leakage also sets a maximum voltage to which the bit line will rise: at some voltage the forward current of the target cell is exactly cancelled by the reverse leakage. In addition, the non-target cell diodes contribute to the capacitance of the target bit line, making it much more capacitive than the non-target bit line.

These factors combine to make the read operation very sensitive to leakage currents. In order for the voltage on the target bit line to rise as quickly as the non-target bit line, the leakage-degraded forward current needs to be many times larger than the leakage through the bit line drivers. Specifically, the ratio of the forward current to the bit line driver leakage needs to be greater than the ratio of the target bit line's capacitance to the non-target bit line's capacitance. Because of this, the bit line driver has many features devoted to reducing its leakage during read operation as shown above.

The word line driver, however, is still quite simple. It needs to be able to drive three different voltages: ground, the write non-select voltage, and the reading voltage. A word line driver capable of driving these three values is pictured in Figure 4.16, and it consists of a PMOS pull up to the reading voltage, an NMOS pull down to ground, and a pass gate that connects the word line to the non-select forward bias. A pass gate is necessary for the non-select voltage because the word line needs to be driven from an unknown state, either logic V_{dd} or ground, to the mid-rail, non-select voltage, which means the driver needs to pull both up and down.

The sense amplifier is implemented as a Strongarm latch with PMOS inputs. NMOS sampling devices have thier sources connected to the Strongarm inputs, their drains connected to BL and BL_B, and their gates connected to a sampling signal which is triggered shortly before the sense amplifiers evaluation signal. The NMOS samplers isolate the bit lines from the Strongarm kickback, which would affect them differently because of the large difference in bit line cap in the worst case. The Strongarm latch is followed by an SR latch per standard practice. All devices in both latches are minimum sized for density reasons, which poses a problem from a variability standpoint: the offsets can easily swamp out the voltage difference on the bit lines.

4.5 Layout Concerns

In NEMory the memory devices are built in layers above the active layer, so it is possible to build the memory array on top of the drivers in order to maximize the density. However, each driver (as picture in Figure 4.13) is larger than an individual NEMory device. NEMory devices are approximately a single wire pitch even and the minimum sized devices used to assemble the level shifter are significantly larger. The drivers need to be carefully laid out to To mitigate this size mismatch and achieve parity between the driver area and the NEMory device area.

Arranging for parity in area between the drivers and the cell array requires exploiting the fact that the number of drivers grows as the sum of the number of rows and columns while the number of NEMory cells grows as the product. To do this, the bit lines will be oriented at ninety degrees to the longest dimension of the bit line drivers. For the sake of discussion, assume the bit lines run vertically – in the y direction – and the longest dimension of the bit line drivers runs horizontally so that it would rest on the x axis. The array has a maximum size in the x direction which is given by the sum of the long dimension of the bit line drivers, the bit line read circuitry, and the word line drivers necessary to cover all of the word lines which fit in the width (short dimension) of one bit line. The NEMory will need to have enough bit lines to cover the maximum x extent, and the number of word lines will be determined by the total number of bit lines.

A diagram of the floor plan that results from this strategy is pictured in Figure 4.17. The minimum size array that results from this floor planning strategy is $40.9\mu m \times 319.0\mu m$, which is 319 bit lines wide by 4785 word lines deep. The array's size is 1.52Mb.

Even with this floor planning effort the array winds up slightly shorter in the y dimension than the underlying drivers when the x dimensions are matches. The spare space is devoted to the dummy NEMory cells which reduce the edge rate of the bit line drivers.

Contacting the NEMory poses a challenge because the electrodes are integrated vertically. In particular, the upper bit line and the word line are both separated from the drivers and underlying metal layers by the lower bit line layer. The array is dense enough that it is impossible to get a via through it. Therefore, it is necessary to wrap wires around the end of the array to contact the bit lines and the word lines. The bit lines are deliberately

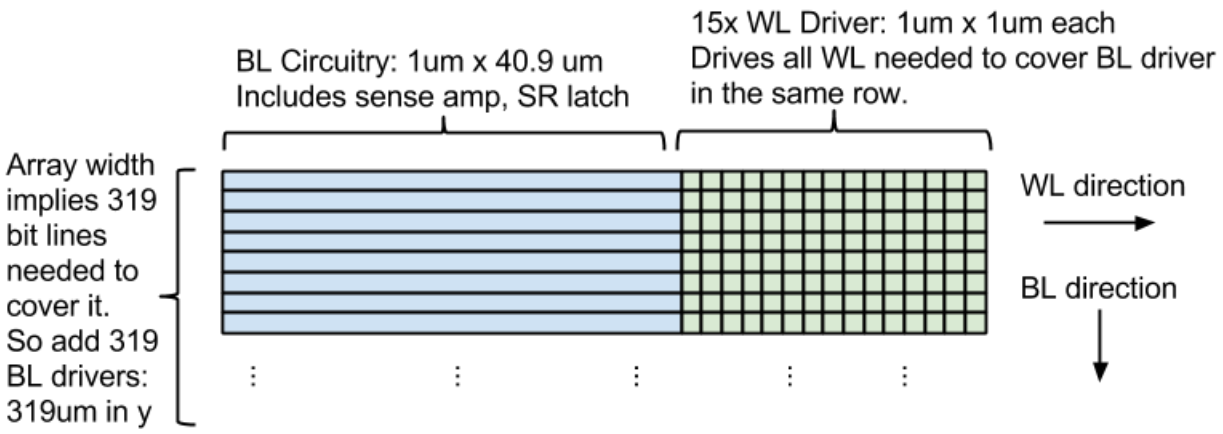


Figure 4.17: Floor plan for NEMory array.

slightly wider than the length of the cell so that two wires can fit beneath them. One wire is contacted to the lower bit line using vias, but the other wire connects to a stack of vias at the north end of the array which extends the upper bit lines. The word lines at the east end of the array can similarly be wrapped around to the word line drivers.

4.6 Performance and Comparison to Other Technologies

Having derived a sample array it is possible to benchmark the array's performance and energy consumption in order to compare it to existing technology. This can be done by using classic electrical models for delay and energy and comparing the models against simulations of the electrical and mechanical behavior of the device. The simuations of the NEMory arrays are built using the force, circuit, and device models pictured above. Like previous chapters, the device models are implemented in Verilog-A so that they can be directly co-simulated with the circuits. In order to keep the simulations tractable and save simulation time, only a single row and column of the NEMory were simulated. Though there is nominally reverse leakage into the rest of the array, the single row/column "cross model" is a good approximation for demonstrating the functionality of the overall system.

The read performance of the NEMory is a purely electrical phenomenon that can be readily calculated based on the unit capacitance and resistance of the bit and word lines and the current through the diode during read mode.

$$t_{read} = t_{WL} + t_{BL} \tag{4.24}$$

$$= \ln(2) N_{WL}(N_{WL}+1) C_{WL} R_{WL} + N_{BL} C_{BL} V_{BL} / I_{rd} \tag{4.25}$$

where t_{WL} is the distributed RC delay of charging the word line, t_{BL} is the delay of charging the bit line, N_{WL} is the number of cells on the word line, C_{WL} is the unit capacitance of a cell on the word line, R_{WL} is the unit word line resistance of a cell, N_{BL} is the number of cells on the bit line, C_{BL} is the unit bit line capacitance of a cell, V_{BL} is the voltage that cells are charged to before the sense amp is triggered, and I_{rd} is the read current through the target cell's diode.

Setting V_{BL} (and, implicitly, t_{BL}) requires careful attention to the voltages and currents during the read transient, because the reverse leakage of the other NEMory cells on the bit line means that the read waveforms won't look exactly like a current source charging a capacitor. Simulations show the bit line voltages and relevant currents during a read transient in Figure 4.18. Based on the simulation, V_{BL} was set to 7mV because that is the value at which there is the largest difference between the bitline voltages during the read. This is because the reverse leakage current through the array prevents charging the bit line any higher than 10mV, while the bit line driver leakage current will continue charging the other, less capacitive bit line well past 10mV.

C_{BL} is comprised of several capacitances: a parasitic capacitance (C_{par_top} or C_{par_bot}), possibly a junction capacitance (C_j) and an air gap capacitance (C_{gap_top} or C_{gap_bot}). Many of these parameters are variable on a number of conditions: the gap capacitance is large when the word line is in contact with the bit line and small otherwise, the junction capacitance also is only present if the word line is contacting the bit line but it is further dependent on the bit line voltage, and the parasitic capacitance is larger for the top cells because of the wraparound wire needed to contact them. As a result, there are a range of possible values for t_{BL} which depend on the stored data of the cell.

The total delay of writing a NEMory cell comes from the electrical delay of charging the bit line. The word line delay is significantly faster, and it can be "hidden" by charging the word line at the same time as the slower bit line. The same is true of the mechanical delay because the devices have high natural frequencies and the bit lines have very large RC constants. The electrical delay of the word and bit lines can be readily calculated from the same distributed RC model used for the word line read delay. Simulations of a test array, described in greater detail below, were carried out to measure the write delay, and transient waveforms showing the results are pictured in Figure 4.19. This simulation confirmed that the mechanical delay is not a significant contributor to the overall delay

The shape of the displacement waveform in Figure 4.19 is interesting because, at first glance, the wave might suggest that the mechanical delay is longer when driving a larger capacitive load. This is not a correct conclusion because the mechanical delay is measured from the time when the bit line reaches the operating voltage of the relay, which is the flat-side pull-in voltage for this device. The flat-side pull-in voltage only demarcates the final pull-in at the end of the mechanical transition, and the duration of that final transition is the same for both cases. However, the displacement varies slowly beforehand in the worst case bit line and quickly in the best case. That is because the NEMory is in equilibrium during the middle of its transition after the initial pull-off, and the slow electrical transition is responsible for the slow mechanical changes until the flat-side pull-in voltage is reached.

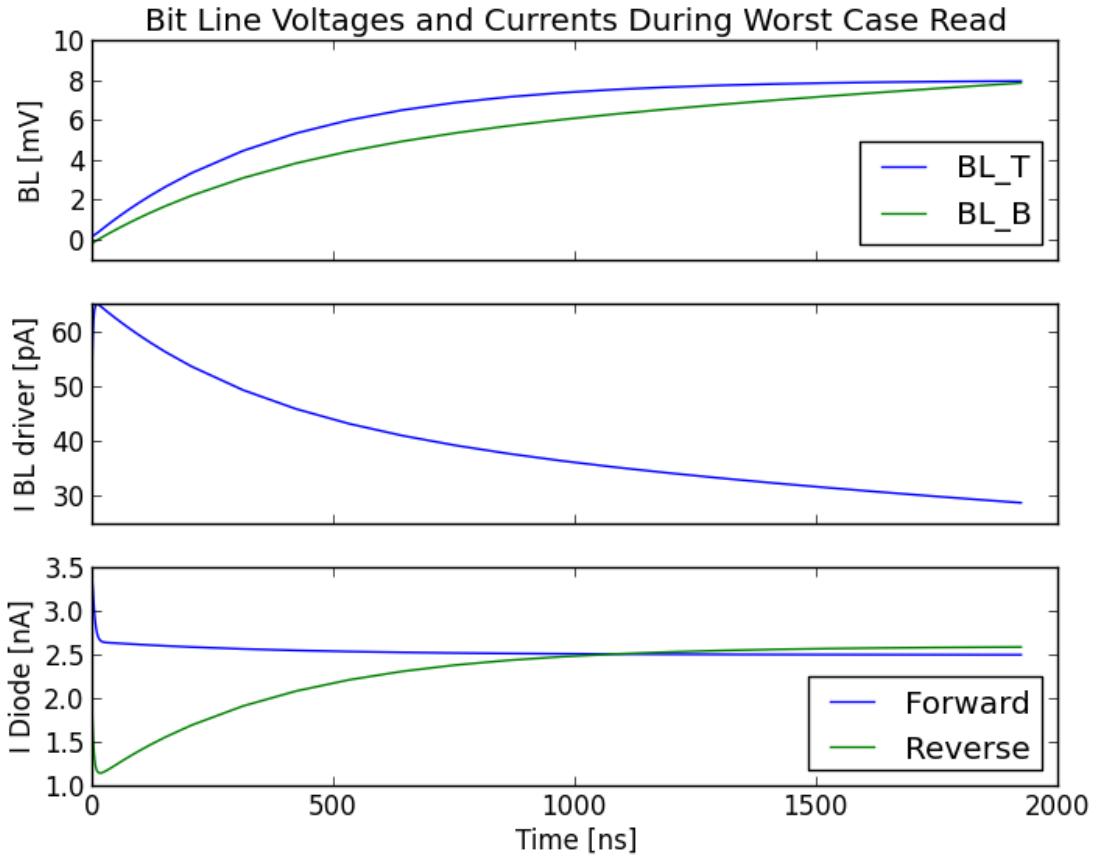


Figure 4.18: Transient waveforms showing the write process for a NEMory cell.

When the capacitive load is low the voltage changes quickly and reaches the higher flat-side pull-in voltage more quickly.

The read and write energy of the NEMory can also be predicted analytically based on the amount of capacitance charged and discharged and the DC currents in the array. During a read there are no DC currents in the array, so only the capacitive energy needs to be considered:

$$E_{read} = E_{BL} + E_{WL} \tag{4.26}$$

$$= N_{WL}(C_{gap\_top} + C_{gap\_bot} + C_j(V_{rd}))V_{rd}^2 + N_{WL}N_{BL}C_{cell,BL}V_{rd}V_{BL} \tag{4.27}$$

where V_{rd} is the word line voltage during a read operation (0.8V per earlier discussion), V_{BL} is the voltage the bit line is allowed to charge to during read operation (7mV in the worst case, per earlier discussion), and $C_{cell,BL}$ is the capacitance that each cell attaches to the bit line. As above, all of the NEMory cells are oriented the same way. They are all disconnected

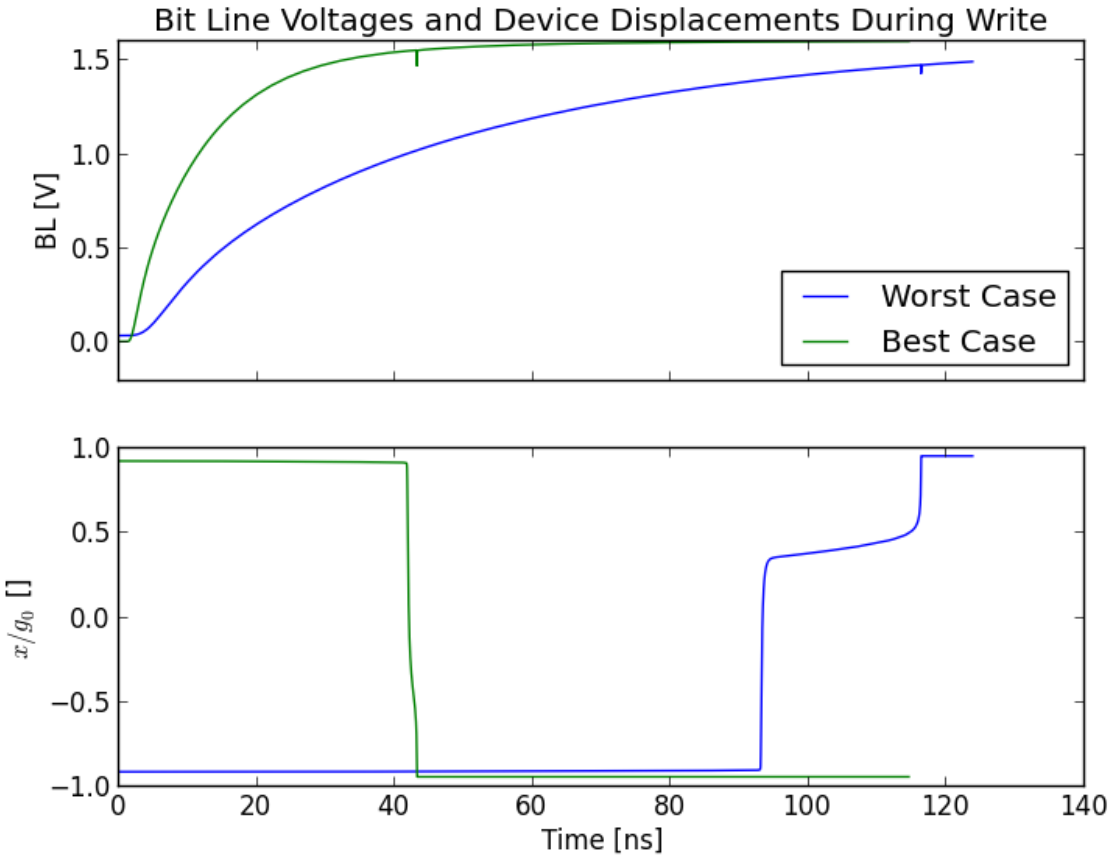


Figure 4.19: Transient waveforms showing the write process for a NEMory cell.

from the bit line in the best case so that $C_{cell,BL} = C_{air}(-g_0)$, and they are all connected to the bit line in the worst case so that $C_{cell,BL} = C_j(0.8) + C_{air}(g_0)$.

The write energy can be expressed as

$$E_{write} = E_{WL,nonselected,cap} + E_{WL,nonselected,DC} + E_{BL} \tag{4.28}$$

$$= N_{WL}(N_{BL} - 1)V_{DD}V_{nonsel} + N_{WL}(N_{BL} - 1)V_{dd}I_{nonsel} + N_{WL}N_{BL}C_{cell,BL}V_{wr}^2 \tag{4.29}$$

where $E_{WL,nonselected,cap}$ is the energy contributed to charging the capacitance of the non-selected word lines, $E_{WL,nonselected,DC}$ accounts for the current that passes through the forward biased diodes during write, E_{BL} is the energy of charging the bit line cap, V_{nonsel} is the voltage applied to the word lines of non-selected words to prevent them from being written, I_{nonsel} is the current that passes through the nonselected cells because of the deselect bias, V_{DD} is the supply voltage (0.8V) and V_{wr} is the voltage applied to write the cells (1.6V because of the voltage doubling drivers). The non-select current can be found using Equation

4.22 because the voltage applied to the array is low enough that the resistance of word and bit lines have little effect on the current passing through the cells. The DC contribution to the energy features I_{nonsel} being multiplied by V_{DD} because it is assumed that V_{nonsel} is generated in a linear way from the supply voltage.

These energy and delay predictions have been checked against a simulation of a NEMory array. The CMOS circuit schematics were implemented in an appropriate PDK and the PDK models were used to simulate them. Models of the NEMory devices were built in Verilog-A based on the analytical models discussed above. The array was sized based on the floor plan shown above, resulting in an array with 315 bit lines and 4785 word lines. It was not possible to simulate the entire array at the same time because of memory limitations, instead one row and one column of the array were simulated. The delay of reading and writing the cell at the intersection of the column were measured, and the read energy was readily determined by measuring the energy extracted from the power supply during a read operation. Extracting the write energy was somewhat more complex because this model only captures the leakage energy of a single column, so the leakage energy was measured using a separate supply and multiplied by the number of columns. Transient results of the simulation appear in Figure 4.20.

These results are summarized and compared against other memory solutions in Table 4.1. Some columns of the table deserve discussion because their entries were interpolated from multiple papers. These entries are marked. The energy consumption of the representative SRAM [46] was calculated by finding a paper describing a similar technology [3] in order to find the $C-V$ characteristics of the NEMory array. These $C-V$ characteristics were used to calculate read energy:

$$E_{rd,SRAM} = N_{WL,SRAM}(2C_{gate})V_{dd,SRAM}^2 + N_{BL,SRAM}N_{WL,SRAM}\gamma_{gd}(1+\gamma_{dm})C_{gate}V_{dd,SRAM}V_{sw,SRAM}, \quad (4.30)$$

where $N_{WL,SRAM}$ and $N_{BL,SRAM}$ are the number of SRAM cells on the word and bit lines, C_{gate} is the gate capacitance for a minimum sized SRAM device (45fF based on [3] and the size of gates extracted from the picture of the cell in [46]), $V_{dd,SRAM}$ is the supply voltage of the SRAM (0.8V), γ_{gd} is the ratio of gate to drain capacitance (assumed to be 1), γ_{dm} is the ratio of the metal-to-metal capacitance of the bit line to the total drain capacitance on the cell (assumed to be 2), and $V_{sw,SRAM}$ is the swing allowed to develop on the bit lines before the sense amplifiers were triggered (assumed 100mV).

Write energy was calculated in a similar way:

$$E_{wr,SRAM} = N_{WL,SRAM}(2C_{gate})V_{dd,SRAM}^2 + N_{WL,SRAM}N_{BL,SRAM}\gamma_{gd}(1+\gamma_{dm})C_{gate}V_{dd,SRAM}^2. \quad (4.31)$$

This equation reflects the one of every pair of bit lines must be charged to full rail, rather than only discharged through a sense amplifier swing. Finally, the write delay was actually lifted from a tentatively related SRAM exemplar [47].

Two papers describing the same piece of eFlash memory were used to compile the eFlash entry [48, 49] because each featured different, relevant waveforms.

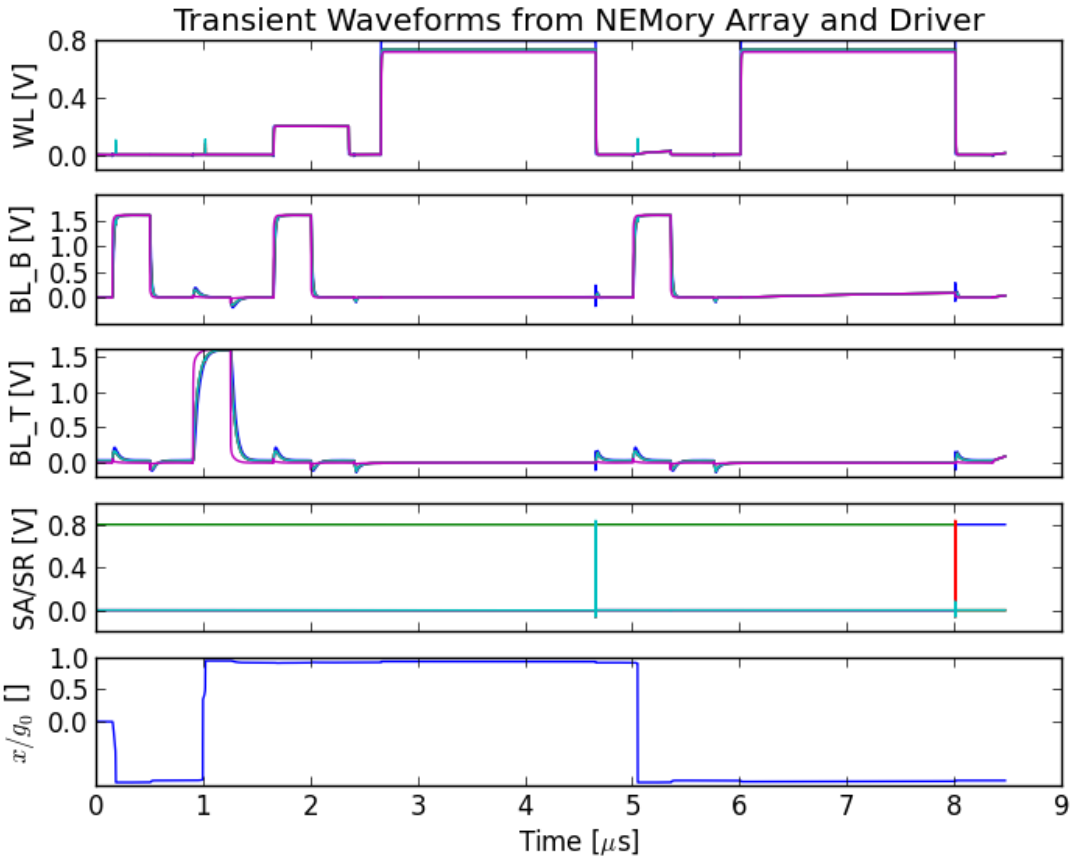


Figure 4.20: Transient waveforms depicting two read and write cycles of a NEMory array. The array is loaded with worst case data: all cells are oriented towards the top bit line, BL_T, resulting in higher capacitance and reverse leakage on that bit line.

Quantity	Units	This Work	[51]	[52]	[53]	[54]	[55]	[56]	[57][3][47]	[58]	[48][49]
Read Delay	ns	750	3.3	9.1	1	6.8	1µs	12	8ps	45µs	11
Write Delay	ns	110	3		5	500	3.5µs	100	0.22†	1.2ms	10µs
Read Energy	pJ/b	2.12fJ	71°	0.12		0.43°	0.05	0.69°	0.23fJ†		0.38°†
Write Energy	pJ/b	0.35	166			0.39	1	144	0.55fJ†		140°
Read Voltage	V	0.8	1.2	1.5	1.2	0.85	0.6	1.2	0.8	3.5	3.5
Write Voltage	V	1.6	0.9	1.5	6.5	3	0.6	2.8	0.8		14
Density	Mb/mm^2	117.2	1.22	1.27	95.2	1.79	0.24	1.34	1.69	739	2†
Process	nm	32*	65		27	28	130	90	14	16	65
Technology		NEMory	MTJ	MTJ	ReRAM	ReRAM	CBRAM	PCM	SRAM	Flash	eFlash
Word Lines		4785	256	512	2048	512			128		
Bit Lines		319	4	1024	8192	512			256		

* The NEMory is made in the 32nm low level metal of a 14nm process

† Interpolated from multiple papers as discussed in the text.

° Interpolated by multiplying provided time, current and voltage.

Table 4.1: Comparison of memory technologies against the simulated and analytical results of the NEMory. Entries are measured in the units given by the units column unless other units are specifically noted in the cell.

The NEMory compares favorably to many other non-volatile memory solutions. In particular, the NEMory is a very high density memory, rivaled only by aggressive 1T1R ReRAM cells and traditional flash memory. The NEMory density is about eight times lower than flash, which is because the flash is 2Fx2F rather than the NEMory's 2Fx4F, the flash stores 2 bits per cell, and the flash is using a feature that is effectively half the size of the NEMory. The ReRAM with similar density to NEMory also has a 2Fx4F cell in a similar process, but overhead of the read and write circuitry degrades its density while NEMory can mitigate that cost by putting the cells above the drivers. The NEMory's read and write energy requirements also best all other contenders except for traditional SRAM, which is a volatile memory solution with a very different purpose.

Even so, the NEMory faces significant challenges to becoming a commonly used tool in CMOS or relay processes. The device as designed relies on releasing extremely small gaps, which is technically unfeasible right now, and is also dependent on very tight process control to ensure an unusually low Hamaker Constant. However, NEMory demonstrates some promise as a high density and low voltage non-volatile memory solutions, and other recent work examines similar, stiction-based, designs for relay and CMOS BEOL applications [50].

Chapter 5

Conclusion

This work has examined the implementation of three broad components of digital systems using nano-electromechanical devices and shown that careful circuit design can improve the performance NEM circuits relative to naive, CMOS-like implementations. The use of NEM circuits is motivated by the immeasurably low leakage exhibited by NEM devices, which promises very low energy-per-operation digital blocks. However, CMOS-like circuits implemented with relays were shown to perform poorly because NEM devices operate significantly more slowly than CMOS devices and are larger. As discussed in Chapter 2, logic circuits benefit from a tree-like logic style which ensures that all of the physical motion of the NEM devices happens at the same time. This ensures that each logical operation requires only a single mechanical delay; the additional electrical delay required to charge the load capacitance is negligible compared to the time required for mechanical motion. This logic style was demonstrated on a pair of test chips [17]. The same problem was shown to haunt timing circuits in Chapter 3, but again a clever circuit design could hide the mechanical delays of each stage of the timing circuit. In this case, the mechanical delay of the timing circuit was designed to occur at the same time as the next stage's logic, which resulted in a pipelined performance of one delay per operation. Unlike the previous two chapters, Chapter 4 observed that memory was limited by the area consumption of NEM devices rather than the mechanical delay: density is a critical parameter of memory circuits. An alternative device was proposed and analyzed to demonstrate the limits of scaling a CMOS memory. A model of the device was created and a memory array made from them was simulated to demonstrate functionality and verify the analysis of the device's delay and power. This performance was compared against a wide variety of technologies and shown to have a very high density and palatable delay and energy performance.

These circuit level optimizations have demonstrated huge improvements over naive NEM circuit designs. Relay logic shows a performance increase of 32x over a similarly constructed 32-bit adder, latch based relay timing improves performance 3x over flip-flop based systems, and NEMory is about 170x denser than CMOS-like memory cells. Thus, circuit level optimization of MEMS devices is crucial to achieving the best energy and delay performance of NEM systems, and is crucial to making fair comparisons between NEMs and its competitors.

5.1 Hurdles for the Relay Process Engineer

Though these circuit optimizations clearly improve the performance of NEM devices they don't answer the underlying question of whether the devices will be successful CMOS replacements. There are sizable process barriers to making these devices at a scale where their energy benefits could be realized. The device need to be scaled to a much smaller scale to reap energy benefits, and their reliability needs to be carefully examined.

One of these barriers is scaling the smallest devices fabricated for this work were in a 250nmnode, but the NEMory devices discussed in Chapter 4 were designed in the low-level metal of a 14nm process. Further, though depositing films for these devices and patterning them are fairly well understood, there's been very little work on releasing MEMS-like structures in extremely fine-line processes. This kind of release could require very delicate processing. For instance, if NEMory were released using HF vapor, great care would have to be taken to remove the ILD without damaging the gate dielectric.

This processing is further complicated by the fact that it is crucial to control the Hamaker constant of contact between the device and its electrodes. The Hamaker constant impacts the required spring constants of devices and, consequently, their minimum operating voltages. Variations in surface stiction across the chip could cause device failures, and providing margin against these variations results in painful overdesign that directly attacks the energy benefits of the device: operating at a higher voltage provides a quadratic energy penalty. This suggests that developing a process for NEM devices will require work on the surface science and packaging techniques. Packaging is of especial importance because the ambient environment affects the contact resistance stability of devices in addition to the chemistry of surfaces [19].

This focus on the Hamaker constant is reflective of the importance of surface forces to the achievable performance of of NEM devices. The fundamental physical limits of the devices are given by the relationship of the three governing forces controlling them: the electrical force applied by the electrodes, the spring force of the deformed structure and the attractive force of the contacting surfaces. This is discussed in [44, 50, 59], and another simple formulation will be presented here for the purpose of illuminating these tradeoffs.

What is required for a NEM device to operate properly? Broadly, multiple force regimes which will cause the device to switch appropriately. Defining those regimes depends on how the device is intended to be operated. There are two categories of operation for NEM devices: active pull-off and non-active pull-off. NEMory devices are an example of active pull-off devices since they have multiple stable contacting states and can only be pulled from one to the other by the application of an external force. The 4T and 6T relay devices of chapters 2 and 3 are non-active pull-off devices since the devices return to a non-contacting state when voltage is removed without any external forces applied to them. Achieving either pull-off behavior requires that the forces on the device obey certain inequalities:

$$F_{elec}(x_{pi}, V_{pi}) > F_k(x_{pi}), \text{ and } F_k(g_d) > F_{surf} \text{ for non-active pull-off,} \tag{5.1}$$

$$F_{elec}(V_{pi}, 2g) + F_k(g) > F_{surf} > F_k(g) \text{ for active pull-off.} \tag{5.2}$$

These inequalities can be expressed in greater detail, which will reveal that the functionality of relays depends on the areal density of relevant forces. Starting with the non-active case, we see that the first inequality gives us the classic expression for the pull-in voltage, while the second gives us the minimum spring constant required for device operation. Substituting the second into the first gives us the minimum operating voltage.

$$kg_d > F_{surf} \tag{5.3}$$

$$k > F_{surf}/g_d \tag{5.4}$$

$$V_{pi} = \sqrt{\frac{8F_{surf}g^3}{27g_d\epsilon_0 A_{elec}}} \tag{5.5}$$

where A_{elec} is the area of the electrode used for actuation. Surface forces tend to be related to the contact area, so assuming a surface force density of $\tilde{F}_{surf}$ and a contact area of A_{con}

$$V_{pi} = \sqrt{\frac{4}{27} \cdot \frac{\tilde{F}_{surf}}{\epsilon_0/2g^2} \cdot \frac{g}{g_d} \cdot \frac{A_{con}}{A_{elec}}} \tag{5.6}$$

This number of design variables in Equation 5.6 is surprisingly small. There is an optimal g/g_d [14] for minimizing the switching energy of the devices, and $\tilde{F}_{surf}$, g and A_{con} are functions of the limitations of processing since they are all set as small as possible for low voltage operation. As a result, the pull-in voltage just varies as the square inverse of the electrical area: lower voltages and energies are paid for in device area. The exchange rate is set by the ratio of the surface force and the electrical force. This is clearer if the equation is inverted for a fixed V_{pi} and F_{elec} is expressed as a force density, $\tilde{F}_{elec}$:

$$\tilde{F}_{elec}(0, V_{pi}) = \frac{\epsilon_0 V_{pi}^2}{2g^2} \tag{5.7}$$

$$\frac{A_{elec}}{A_{con}} = \frac{4}{27} \cdot \frac{g}{g_d} \cdot \frac{\tilde{F}_{surf}}{\tilde{F}_{elec}(0, V_{pi})} \tag{5.8}$$

Broadly, this suggests that the sizing of a gap is crucial for achieving device density. There are many more specific statements that could be made by assigning values to these dimensions – picking a Hamaker constant, a minimum gap size and a contact area – and this quick derivation doesn't address the impact of device sizing on energy. However, the derivation does illuminate the fundamental tradeoffs in a MEMS device between force, area and operating voltage. That fundamental tradeoff is mediated by the limits of process in providing a low surface force, a small gap and a small contact area. These process goals depend on research into releasing small gaps, understanding MEMS surface chemistry, and appropriate packaging to stabilize the surface chemistry and ambient environment. A more in depth analysis of this variety is crucial to evaluating future NEM devices.

Active pull-off devices can be treated to a similar analysis, which will reveal that the surface force is even more critical to their operation as seen in Chapter 4.

5.2 Future Work

Though realizing a NEM relay technology will require a great deal of work, there are many opportunities to improve the circuit and device design as well. Of immediate interest are more and larger scale experimental demonstrations. Many of the circuits in this work have been shown to operate in simulation, and experimental verification would be valuable. In particular, a demonstration of a microprocessor would be a great leap forward in integration and functionality of these devices. All the necessary building blocks have been discussed, so assembling them into a demonstration vehicle is an excellent and expedient proof of the value of the technology. Other demonstrations would be valuable as well: ADCs, DACs and clocking circuits would be beneficial additions to the body of relay experiments.

There are many analytical loose ends that would improve the state of knowledge of relay technolgy. Extending the tradeoff analysis in the previous section would help to ascertain the ultimate scaling limits of various relay devices. That scaling analysis could be mapped up to the circuit level to do comparisons of NEM devices against other circuits in a way that can account for developments in processing technology. Finally, none of the analysis in this work has considered variability. Device equations in terms of the randomly varying quantities have been introduced throughout the work, so mapping from those to a distribution of circuit level parameters – V_{pi}, t_{mech} – is readily pursuable and will provide a more accurate estimate of the performance of the devices and the process parameters that need to be tightly controlled.

The CAD tools introduced in this thesis are insufficient for design of really large digital systems, so a formalized synthesis technique needs to be introduced. Basic synthesis flows have been demonstrated [31], but the flow lacked timing closure, retiming, and many of the other features critical to synthesis of large systems. Because the delay characteristics of relays are different than CMOS, these features can't be acquired by modifying standard cells for a synthesis tool. Instead, custom timing calculators need to be developed. Further, there are other algorithms that could produce valid mechanical logic structures with a single mechanical delay, such as SOP. Investigating these algorithms would be beneficial to overall synthesis efficiency.

Finally, there are more exotic variants of devices that could be considered, including devices with more input and output electrodes than those considered here. For instance, a seesaw style device with three output electrodes can implement a full-adder cell in a single device footprint. A more rigorous study of the logic of multi-electrode devices and a set of rules for how to partition logical functions between additional electrodes, additional devices, additional gates and additional pipeline stages could result in a more globally optimum, denser NEM system.

5.3 Final Thoughts

Though history is littered with predictions of the end of Moore's law, the end seems to be looming larger than ever for both economic and physical reasons. That would be a

tragedy: Moore's law has driven tremendous economic growth, invaluably aided the progress of science, and enabled sweeping social changes in a tiny amount of time. However, as transistor scaling has slowed, transistor heterogeneity has increased: the presence of high-k metal-gate devices, SOI devices and fin devices in the same market attests to increasing diversification. Device heterogeneity and the use of post-Moore devices could extend that trend.

Relays are interesting from a post-Moore standpoint because scaled devices promise lower leakage than any other technology on the market. Massively parallel, high latency operations, which constitute much of scientific computing and web hosting, could benefit tremendously from the reduction in power promised by lower leakage. So with further work, NEM relays and their relative could be a powerful arrow in the post-Moore quiver. Showing the benefits of the technology will continue to require careful circuit and device co-design to maximize the strengths of the technology while minimizing its weaknesses.

Bibliography

[1] H. Kam, V. Pott, R. Nathanael, J. Jeon, E. Alon, and T.-J. K. Liu, "Design and reliability of a micro-relay technology for zero-standby-power digital logic applications," in *Electron Devices Meeting (IEDM), 2009 IEEE International*, Dec. 2009, pp. 1–4.

[2] B. Calhoun, A. Wang, and A. Chandrakasan, "Modeling and sizing for minimum energy operation in subthreshold circuits," *IEEE Journal of Solid-State Circuits*, vol. 40, no. 9, pp. 1778–1786, Sep. 2005.

[3] S.-Y. Wu, C. Lin, M. Chiang, J. Liaw, J. Cheng, S. Yang, M. Liang, T. Miyashita, C. Tsai, B. Hsu, H. Chen, T. Yamamoto, S. Chang, V. Chang, C. Chang, J. Chen, H. Chen, K. Ting, Y. Wu, K. Pan, R. Tsui, C. Yao, P. Chang, H. Lien, T. Lee, H. Lee, W. Chang, T. Chang, R. Chen, M. Yeh, C. Chen, Y. Chiu, Y. Chen, H. Huang, Y. Lu, C. Chang, M. Tsai, C. Liu, K. Chen, C. Kuo, H. Lin, S. Jang, and Y. Ku, "A 16nm FinFET CMOS technology for mobile SoC and computing applications," in *Electron Devices Meeting (IEDM), 2013 IEEE International*, Dec. 2013, pp. 9.1.1–9.1.4.

[4] Q. Liu, B. DeSalvo, P. Morin, N. Loubet, S. Pilorget, F. Chafik, S. Maitrejean, E. Augendre, D. Chanemougame, S. Guillaumet, H. Kothari, F. Allibert, B. Lherron, B. Liu, Y. Escarabajal, K. Cheng, J. Kuss, M. Wang, R. Jung, S. Teehan, T. Levin, M. Sankarapandian, R. Johnson, J. Kanyandekwe, H. He, R. Venigalla, T. Yamashita, B. Haran, L. Grenouillet, M. Vinet, O. Weber, E. Josse, F. Boeuf, M. Haond, J.-L. Bataillon, W. Kleemeier, T. Skotnicki, M. Khare, O. Faynot, B. Doris, M. Celik, and R. Sampson, "FDSOI CMOS devices featuring dual strained channel and thin BOX extendable to the 10nm node," in *Electron Devices Meeting (IEDM), 2014 IEEE International*, Dec. 2014, pp. 9.1.1–9.1.4.

[5] T. Nirschl, P. Wang, C. Weber, J. Sedlmeir, R. Heinrich, R. Kakoschke, K. Schrufer, J. Holz, C. Pacha, T. Schulz, M. Ostermayr, A. Olbrich, G. Georgakos, E. Ruderer, W. Hansch, and D. Schmitt-Landsiedel, "The tunneling field effect transistor (TFET) as an add-on for ultra-low-voltage analog and digital processes," in *Electron Devices Meeting, 2004. IEDM Technical Digest. IEEE International*, Dec. 2004, pp. 195–198.

[6] H. Lu and A. Seabaugh, "Tunnel Field-Effect Transistors: State-of-the-Art," *Electron Devices Society, IEEE Journal of the*, vol. 2, no. 4, pp. 44–49, Jul. 2014.

[7] U. Avci, D. Morris, and I. Young, "Tunnel Field-Effect Transistors: Prospects and Challenges," *Electron Devices Society, IEEE Journal of the*, vol. 3, no. 3, pp. 88–95, May 2015.

[8] L. Knoll, Q.-T. Zhao, A. Nichau, S. Trellenkamp, S. Richter, A. Schafer, D. Esseni, L. Selmi, K. Bourdelle, and S. Mantl, "Inverters With Strained Si Nanowire Complementary Tunnel Field-Effect Transistors," *IEEE Electron Device Letters*, vol. 34, no. 6, pp. 813–815, Jun. 2013.

[9] G. Dewey, B. Chu-Kung, J. Boardman, J. Fastenau, J. Kavalieros, R. Kotlyar, W. Liu, D. Lubyshev, M. Metz, N. Mukherjee, P. Oakey, R. Pillarisetty, M. Radosavljevic, H. Then, and R. Chau, "Fabrication, characterization, and physics of III #x2013;V heterojunction tunneling Field Effect Transistors (H-TFET) for steep sub-threshold swing," in *Electron Devices Meeting (IEDM), 2011 IEEE International*, Dec. 2011, pp. 33.6.1–33.6.4.

[10] M. Alam, S. Kurtz, M. Siddiq, M. Niemier, G. Bernstein, X. Hu, and W. Porod, "On-Chip Clocking of Nanomagnet Logic Lines and Gates," *IEEE Transactions on Nanotechnology*, vol. 11, no. 2, pp. 273–286, Mar. 2012.

[11] D. Bromberg, D. Morris, L. Pileggi, and J.-G. Zhu, "Novel STT-MTJ Device Enabling All-Metallic Logic Circuits," *IEEE Transactions on Magnetics*, vol. 48, no. 11, pp. 3215–3218, 2012.

[12] V. Calayir, D. Nikonov, S. Manipatruni, and I. Young, "Static and Clocked Spintronic Circuit Design and Simulation With Performance Analysis Relative to CMOS," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 61, no. 2, pp. 393–406, Feb. 2014.

[13] S. D. Senturia, *Microsystem Design.* Norwell, MA, USA: Kluwer Academic Publishers, 2001.

[14] H. Kam, T.-J. K. Liu, V. Stojanovi, D. Markovic, and E. Alon, "Design, optimization, and scaling of MEM relays for ultra-low-power digital logic," *IEEE Transactions on Electron Devices*, vol. 58, no. 1, pp. 236–250, 2011.

[15] F. Chen, H. Kam, D. Markovic, T.-J. K. Liu, V. Stojanovic, and E. Alon, "Integrated circuit design with NEM relays," in *IEEE/ACM International Conference on Computer-Aided Design, 2008. ICCAD 2008*, Nov. 2008, pp. 750–757.

[16] F. Chen, M. Spencer, R. Nathanael, C. Wang, H. Fariborzi, A. Gupta, H. Kam, V. Pott, J. Jeon, T.-J. K. Liu, D. Markovic, V. Stojanovic, and E. Alon, "Demonstration of integrated micro-electro-mechanical switch circuits for VLSI applications," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2010 IEEE International*, 2010, pp. 150–151.

[17] M. Spencer, F. Chen, C. Wang, R. Nathanael, H. Fariborzi, A. Gupta, H. Kam, V. Pott, J. Jeon, T.-J. K. Liu, D. Markovic, E. Alon, and V. Stojanovic, "Demonstration of integrated micro-electro-mechanical relay circuits for VLSI applications," *IEEE Journal of Solid-State Circuits*, vol. 46, no. 1, pp. 308–320, 2011.

[18] H. Kam, E. Alon, and T.-J. K. Liu, "A predictive contact reliability model for MEM logic switches," in *Electron Devices Meeting (IEDM), 2010 IEEE International*, Dec. 2010, pp. 16.4.1–16.4.4.

[19] Y. Chen, R. Nathanael, J. Jeon, J. Yaung, L. Hutin, and T.-J. K. Liu, "Characterization of contact resistance stability in MEM relays with tungsten electrodes," *Journal of Microelectromechanical Systems*, vol. 21, no. 3, pp. 511–513, Jun. 2012.

[20] I.-R. Chen, Y. Chen, L. Hutin, V. Pott, R. Nathanael, and T.-J. Liu, "Stable ruthenium-contact relay technology for low-power logic," in *2013 Transducers Eurosensors XXVII: The 17th International Conference on Solid-State Sensors, Actuators and Microsystems (TRANSDUCERS EUROSENSORS XXVII)*, Jun. 2013, pp. 896–899.

[21] F. Jensen, *Introduction to Computational Chemistry*. John Wiley & Sons, 2006.

[22] *Verilog-A Language Reference Manual*. Acellara Organization, Inc., 2008.

[23] S. Rana, Q. Tian, A. Bazigos, D. Grogg, M. Despont, C. Ayala, C. Hagleitner, A. Ionescu, R. Canegallo, and D. Pamunuwa, "Energy and Latency Optimization in NEM Relay-Based Digital Circuits," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 61, no. 8, pp. 2348–2359, Aug. 2014.

[24] R. Holm, *Electric contacts: theory and applications*. Springer, Aug. 1999.

[25] J. M. Rabaey, A. P. Chandrakasan, and B. Nikolic, *Digital integrated circuits: a design perspective*. Pearson Education, 2003.

[26] J. Jeon, R. Nathanael, V. Pott, and T.-J. K. Liu, "Four-Terminal Relay Design for Improved Body Effect," *IEEE Electron Device Letters*, vol. 31, no. 5, pp. 515–517, May 2010.

[27] H. Fariborzi, M. Spencer, V. Karkare, J. Jeon, R. Nathanael, C. Wang, F. Chen, H. Kam, V. Pott, T.-J. K. Liu, E. Alon, V. Stojanovic, and D. Markovic, "Analysis and demonstration of MEM-relay power gating," in *2010 IEEE Custom Integrated Circuits Conference (CICC)*, 2010, pp. 1–4.

[28] H. Fariborzi, F. Chen, V. Stojanovic, R. Nathanael, J. Jeon, and T.-J. K. Liu, "Design and demonstration of micro-electro-mechanical relay multipliers," in *Solid State Circuits Conference (A-SSCC), 2011 IEEE Asian*, 2011, pp. 117–120.

[29] H. Fariborzi, F. Chen, R. Nathanael, I.-R. Chen, L. Hutin, R. Lee, T.-J. K. Liu, and V. Stojanovic, "Relays do not leak: CMOS does," in *Proceedings of the 50th Annual Design Automation Conference*, ser. DAC '13. New York, NY, USA: ACM, 2013, pp. 127:1–127:4. [Online]. Available: http://doi.acm.org/10.1145/2463209.2488890

[30] J. Jeon, L. Hutin, R. Jevtic, N. Liu, Y. Chen, R. Nathanael, W. Kwon, M. Spencer, E. Alon, B. Nikolic, and T.-J. K. Liu, "Multiple-input relay design for more compact implementation of digital logic circuits," *IEEE Electron Device Letters*, vol. 33, no. 2, pp. 281–283, 2012.

[31] K. Dwan and D. Markovic, "Logic synthesis of mem relay circuits," Master's thesis, ECE Department, University of California, Los Angeles, May 2011. [Online]. Available: http://icslwebs.ee.ucla.edu/dejan/researchwiki/images/b/bf/Kevin_Dwan_MS_Thesis.pdf

[32] M. Jiang, "Bdd based logic synthesis of mem relay circuits," Master's thesis, May 2013. [Online]. Available: http://escholarship.org/uc/item/2mp5d00h

[33] J. Sklansky, "Conditional-sum addition logic," *IRE Transactions on Electronic Computers*, vol. EC-9, no. 2, pp. 226–231, Jun. 1960.

[34] D. Patil, O. Azizi, M. Horowitz, R. Ho, and R. Ananthraman, "Robust energy-efficient adder topologies," in *18th IEEE Symposium on Computer Arithmetic, 2007. ARITH '07*, Jun. 2007, pp. 16–28.

[35] O. Degani, E. Socher, A. Lipson, T. Lejtner, D. Setter, S. Kaldor, and Y. Nemirovsky, "Pull-in study of an electrostatic torsion microactuator," *Journal of Microelectromechanical Systems*, vol. 7, no. 4, pp. 373–379, 1998.

[36] J. Jeon, V. Pott, H. Kam, R. Nathanael, E. Alon, and T.-J. K. Liu, "Seesaw Relay Logic and Memory Circuits," *Journal of Microelectromechanical Systems*, vol. 19, no. 4, pp. 1012–1014, Aug. 2010.

[37] ——, "Perfectly Complementary Relay Design for Digital Logic Applications," *IEEE Electron Device Letters*, vol. 31, no. 4, pp. 371–373, Apr. 2010.

[38] A. Gupta and E. Alon, "Nem relay memory design," Master's thesis, EECS Department, University of California, Berkeley, May 2009. [Online]. Available: http://www.eecs.berkeley.edu/Pubs/TechRpts/2009/EECS-2009-83.html

[39] W. Kwon, J. Jeon, L. Hutin, and T.-J. K. Liu, "Electromechanical Diode Cell for Cross-Point Nonvolatile Memory Arrays," *IEEE Electron Device Letters*, vol. 33, no. 2, pp. 131–133, 2012.

[40] L. Hutin and T.-J. K. Liu, "Non-volatile Electro-Mechanical Memory (NEMory) Cell Scaling for Energy-efficient and High-density Crosspoint Arrays," in *Proceedings of the 3rd Berkeley Symposium on Energy Efficient Electronic Systems*, Oct. 2013.

[41] R. Maboudian and R. T. Howe, "Critical Review: Adhesion in surface micromechanical structures," *Journal of Vacuum Science & Technology B*, vol. 15, no. 1, pp. 1–20, Jan. 1997. [Online]. Available: http://scitation.aip.org/content/avs/journal/jvstb/15/1/10.1116/1.589247

[42] N. Yu and A. A. Polycarpou, "Adhesive contact based on the LennardJones potential: a correction to the value of the equilibrium distance as used in the potential," *Journal of Colloid and Interface Science*, vol. 278, no. 2, pp. 428–435, Oct. 2004. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0021979704005454

[43] L. Hutin, personal communication, 01 2014.

[44] C. Pawashe, K. Lin, and K. Kuhn, "Scaling Limits of Electrostatic Nanorelays," *IEEE Transactions on Electron Devices*, vol. 60, no. 9, pp. 2936–2942, 2013.

[45] J. N. Israelachvili, *Intermolecular and Surface Forces: Revised Third Edition.* Academic Press, Jul. 2011.

[46] M.-C. Chen, C.-H. Lin, Y.-F. Hou, Y.-J. Chen, C.-Y. Lin, F.-K. Hsueh, H.-L. Liu, C.-T. Liu, B.-W. Wang, H.-C. Chen, C.-C. Chen, S.-H. Chen, C.-T. Wu, T.-Y. Lai, M.-Y. Lee, B.-W. Wu, C.-S. Wu, I. Yang, Y.-P. Hsieh, C. Ho, T. Wang, A. Sachid, C. Hu, and F.-L. Yang, "A 10 nm Si-based bulk FinFETs 6t SRAM with multiple fin heights technology for 25% better static noise margin," in *2013 Symposium on VLSI Circuits (VLSIC)*, Jun. 2013, pp. T218–T219.

[47] E. Karl, Y. Wang, Y.-G. Ng, Z. Guo, F. Hamzaoglu, U. Bhattacharya, K. Zhang, K. Mistry, and M. Bohr, "A 4.6ghz 162mb SRAM design in 22nm tri-gate CMOS technology with integrated active VMIN-enhancing assist circuitry," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2012 IEEE International*, Feb. 2012, pp. 230–232.

[48] D. Shum, J. Power, R. Ullmann, E. Suryaputra, K. Ho, J. Hsiao, C. Tan, W. Langheinrich, C. Bukethal, V. Pissors, G. Tempel, M. Rohrich, A. Gratz, A. Iserhagen, E. Andersen, S. Paprotta, W. Dickenscheid, R. Strenz, R. Duschl, T. Kern, C. Hsieh, C. Huang, C. Ho, H. Kuo, C. Hung, Y. Lin, and L. Tran, "Highly Reliable Flash Memory with Self-Aligned Split-Gate Cell Embedded into High Performance 65nm CMOS for Automotive amp; Smartcard Applications," in *Memory Workshop (IMW), 2012 4th IEEE International*, May 2012, pp. 1–4.

[49] M. Jefremow, T. Kern, U. Backhausen, J. Elbs, B. Rousseau, C. Roll, L. Castro, T. Roehr, E. Paparisto, K. Herfurth, R. Bartenschlager, S. Thierold, R. Renardy, S. Kassenetter, N. Lawal, M. Strasser, W. Trottmann, and D. Schmitt-Landsiedel, "A 65nm 4mb embedded flash macro for automotive achieving a read throughput of 5.7gb/s and a write throughput of 1.4mb/s," in *ESSCIRC (ESSCIRC), 2013 Proceedings of the*, Sep. 2013, pp. 193–196.

[50] N. Xu, J. Sun, I.-R. Chen, L. Hutin, Y. Chen, J. Fujiki, C. Qian, and T.-J. K. Liu, "Hybrid CMOS/BEOL-NEMS technology for ultra-low-power IC applications," in *Electron Devices Meeting (IEDM), 2014 IEEE International*, Dec. 2014, pp. 28.8.1–28.8.4.

[51] H. Noguchi, K. Ikegami, K. Kushida, K. Abe, S. Itai, S. Takaya, N. Shimomura, J. Ito, A. Kawasumi, H. Hara, and S. Fujita, "7.5 A 3.3ns-access-time 71.2 #x03bc;W/MHz 1mb embedded STT-MRAM using physically eliminated read-disturb scheme and normally-off memory architecture," in *Solid- State Circuits Conference - (ISSCC), 2015 IEEE International*, Feb. 2015, pp. 1–3.

[52] C. Kim, K. Kwon, C. Park, S. Jang, and J. Choi, "7.4 A covalent-bonded cross-coupled current-mode sense amplifier for STT-MRAM with 1t1mtj common source-line structure array," in *Solid- State Circuits Conference - (ISSCC), 2015 IEEE International*, Feb. 2015, pp. 1–3.

[53] R. Fackenthal, M. Kitagawa, W. Otsuka, K. Prall, D. Mills, K. Tsutsui, J. Javanifard, K. Tedrow, T. Tsushima, Y. Shibahara, and G. Hush, "19.7 A 16gb ReRAM with 200mb/s write and 1gb/s read in 27nm technology," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2014 IEEE International*, Feb. 2014, pp. 338–339.

[54] M.-F. Chang, J.-J. Wu, T.-F. Chien, Y.-C. Liu, T.-C. Yang, W.-C. Shen, Y.-C. King, C.-J. Lin, K.-F. Lin, Y.-D. Chih, S. Natarajan, and J. Chang, "19.4 embedded 1mb ReRAM in 28nm CMOS with 0.27-to-1v read using swing-sample-and-couple sense amplifier and self-boost-write-termination scheme," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2014 IEEE International*, Feb. 2014, pp. 332–333.

[55] N. Gilbert, Y. Zhang, J. Dinh, B. Calhoun, and S. Hollmer, "A 0.6v 8 pJ/write non-volatile CBRAM macro embedded in a body sensor node for ultra low energy applications," in *2013 Symposium on VLSI Circuits (VLSIC)*, Jun. 2013, pp. C204–C205.

[56] G. De Sandre, L. Bettini, A. Pirola, L. Marmonier, M. Pasotti, M. Borghi, P. Mattavelli, P. Zuliani, L. Scotti, G. Mastracchio, F. Bedeschi, R. Gastaldi, and R. Bez, "A 4 Mb LV MOS-Selected Embedded Phase Change Memory in 90 nm Standard CMOS Technology," *IEEE Journal of Solid-State Circuits*, vol. 46, no. 1, pp. 52–63, Jan. 2011.

[57] T. Song, W. Rim, J. Jung, G. Yang, J. Park, S. Park, K.-H. Baek, S. Baek, S.-K. Oh, J. Jung, S. Kim, G. Kim, J. Kim, Y. Lee, K. S. Kim, S.-P. Sim, J. S. Yoon, and K.-M. Choi, "13.2 A 14nm FinFET 128mb 6t SRAM with VMIN-enhancement techniques for low-power applications," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2014 IEEE International*, Feb. 2014, pp. 232–233.

[58] M. Helm, J.-K. Park, A. Ghalam, J. Guo, C. w. Ha, C. Hu, H. Kim, K. Kavalipurapu, E. Lee, A. Mohammadzadeh, D. Nguyen, V. Patel, T. Pekny, B. Saiki, D. Song, J. Tsai, V. Viajedor, L. Vu, T. Wong, J. H. Yun, R. Ghodsi, A. d'Alessandro, D. Di Cicco,

and V. Moschiano, "19.1 A 128gb MLC NAND-Flash device using 16nm planar cell," in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2014 IEEE International*, Feb. 2014, pp. 326–327.

[59] A. Jain and M. Alam, "Prospects of Hysteresis-Free Abrupt Switching (0 mV/decade) in Landau Switches," *IEEE Transactions on Electron Devices*, vol. 60, no. 12, pp. 4269–4276, 2013.