DESIGN AND EVALUATION OF CIRCUITS AND ARCHITECTURES
BASED ON BEYOND-CMOS DEVICE TECHNOLOGIES

A Dissertation

Submitted to the Graduate School
of the University of Notre Dame
in Partial Fulfillment of the Requirements
for the Degree of

Doctor of Philosophy

by

Indranil Palit

________________________________________
X. Sharon Hu, Co-Director

________________________________________
Michael T. Niemier, Co-Director

Graduate Program in Computer Science and Engineering
Notre Dame, Indiana

February 2017
DESIGN AND EVALUATION OF CIRCUITS AND ARCHITECTURES
BASED ON BEYOND-CMOS DEVICE TECHNOLOGIES

Abstract
by
Indranil Palit

It is well known that CMOS scaling trends are now accompanied by less desirable byproducts such as increased energy dissipation. To combat the aforementioned challenges, solutions are sought at both the device and architectural levels. With this context, this work focuses on developing a systematic circuit design methodology for Nanomagnet Logic (NML) devices, and exploring the benefits of utilizing beyond-CMOS emerging transistors to process information with the non-binary/non-von Neumann computer architecture of Cellular Neural Networks (CNNs).

Nanomagnet Logic (NML) is a device architecture that performs logic operations through fringing field interactions between nano-scale magnets. The design space for NML circuits is large and so far there exists no systematic approach for determining the parameter values (e.g., device-to-device spacings, clocking field strength etc.) to generate a predictable design solution. This work presents a formal methodology for designing NML circuits that marshals the design parameters to generate a layout that is guaranteed to evolve correctly in time at 0K. The approach is further augmented to identify functional design targets when considering thermal noise associated with higher temperatures.

A Cellular Neural Network (CNN) is a highly-parallel, analog processor that can significantly outperform von Neumann architectures for certain classes of problems.
This work shows that emerging technologies like tunneling field effect transistors (TFETs) can be successfully employed in CNNs to solve binary classification problems. Such systems provide significant power savings when compared to the conventional resistor-based CNN. Moreover, TFET-based CNN reduces implementation footprints by eliminating the hardware required to realize output transfer functions.

It is also shown how emerging, beyond-CMOS devices could help to further enhance the capabilities of CNNs, particularly for solving problems with non-binary outputs. CNNs based on devices such as graphene transistors – with multiple steep current growth regions separated by negative differential resistance (NDR) in their I-V characteristics – could be used to recognize multiple patterns simultaneously, (This would require multiple steps given a conventional, binary CNN.) Also, a circuit built from TFETs is used in CNN to perform similar tasks. With this approach, more “exotic” device I-V characteristics are not required – which should be an asset when considering issues such as cell-to-cell mismatch, etc. As a case study, a CNN-cell design is presented that employs TFET-based circuitry to realize ternary outputs, and employed to efficiently solve a tactile sensing problem. The total number of computation steps as well as the required hardware could be reduced significantly when compared to an approach based on a conventional CNN.

Much existing work reports energy dissipation for CNNs at the chip level, which includes dissipation of sensors, actuators, and other components. As such, the impacts of various system variables, e.g., application templates, characteristics of the resistive element, etc., on the energy profile of a CNN cannot be easily determined. This work also proposes analytical models to estimate CNN power and performance. Power dissipations, and settling times obtained via the models for different linear, and non-linear characteristics are verified through circuit simulation. Simulation results show that the proposed models predict power dissipation and settling time with less than 1% and 3% errors, respectively. Case studies are performed by using these
models, for a tactile sensing problem, and a pattern recognition problem to compare power and performance between tunneling field effect transistor (TFET) based non-linear CNN and conventional linear resistor based CNN.

Traditional, CMOS based, von Neumann architectures face daunting challenges in performing complex computational tasks at high speed and with low power on spatio-temporal data, e.g., image processing, pattern recognition, etc. This work studies the potential of analog/mixed-signal information processing using TFETs in the context of cellular neural networks (CNNs). A TFET-based, mixed signal CNN architectures for spatio-temporal information processing is presented. Assuming a 14 nm InAs Homojunction TFET for the proposed architecture, power efficiency of more than 2,000 GOPS/W is projected for a number of different CNN templates. By comparison, state-of-the-art hardware assuming CMOS technology promises a power efficiency only close to 1,000 GOPS/W.
3.3.2 Conditions for Hard Axis Stability ........................................ 25
3.3.3 Conditions for Correct Switching ........................................ 27
3.3.4 Overall Approach .......................................................... 29
3.4 Detailed Design Approach .................................................... 30
3.4.1 Making Interior Magnets Non-critical .................................... 30
3.4.2 Switchability Test ........................................................... 33
3.4.3 Determination of Clock Field ............................................. 34
3.5 Case Studies ................................................................. 37
3.6 Thermal Noise Consideration ............................................... 39
3.7 Conclusion ................................................................. 42

CHAPTER 4: TFET-BASED CNN .................................................. 43
4.1 Overview .............................................................................. 43
4.2 The Cell Circuit ............................................................... 44
4.3 Operation Principles ........................................................ 46
4.4 Simulation Results ........................................................... 47
4.5 CNN Power Dissipation ....................................................... 49
4.5.1 Spice Model for TFET ...................................................... 51
4.5.2 TFET-based CNN vs. Conventional CNN ......................... 51
4.5.3 TFET-based Cells with Different $V_{gs}$ ................................ 54
4.5.4 TFET-based Cells with Different Operating Ranges ................ 54
4.6 Conclusion ................................................................. 55

CHAPTER 5: MULTI-VALUED CNN ............................................. 58
5.1 Overview .............................................................................. 58
5.2 Desired Characteristics ....................................................... 59
5.3 Example Multi-valued CNN ................................................ 60
5.4 Circuits for Multi-valued Problems ....................................... 63
5.5 Case Study ................................................................. 64
5.6 Conclusion ................................................................. 68

CHAPTER 6: MODELING CNN ENERGY DISSIPATION ..................... 70
6.1 Overview .............................................................................. 70
6.2 Model Derivation ............................................................... 71
6.2.1 Model for Power Dissipation ............................................ 71
6.2.2 Model for Settling Time .................................................... 75
6.3 Model Validation ............................................................. 78
6.3.1 Model Validation for Power ............................................. 78
6.3.2 Model Validation for Settling Time .................................... 81
6.4 Case Studies ................................................................. 82
6.5 Conclusions ................................................................. 87
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>MIXED SIGNAL ARCHITECTURE FOR CNN</td>
<td>90</td>
</tr>
<tr>
<td>7.1</td>
<td>Overview</td>
<td>90</td>
</tr>
<tr>
<td>7.2</td>
<td>Architecture for Zero-feedback CNN Applications</td>
<td>92</td>
</tr>
<tr>
<td>7.3</td>
<td>MS Architecture for Uncoupled CNN Applications</td>
<td>94</td>
</tr>
<tr>
<td>7.4</td>
<td>Evaluation</td>
<td>99</td>
</tr>
<tr>
<td>7.4.1</td>
<td>Benefits of TFETs in MS Architecture</td>
<td>101</td>
</tr>
<tr>
<td>7.4.2</td>
<td>Evaluating MS Architecture</td>
<td>102</td>
</tr>
<tr>
<td>7.5</td>
<td>Delay-based Operation via MS Architecture</td>
<td>105</td>
</tr>
<tr>
<td>7.6</td>
<td>Conclusions</td>
<td>107</td>
</tr>
<tr>
<td>8</td>
<td>CONCLUSIONS</td>
<td>109</td>
</tr>
<tr>
<td></td>
<td>BIBLIOGRAPHY</td>
<td>110</td>
</tr>
</tbody>
</table>
FIGURES

2.1 (a) AF-line moves information; (b) an AF-line has a new input, and an external clocking field is used to facilitate re-evaluation of the line; (c) as the field is removed, devices relax along their easy axes; (d) Energy landscape of a representative magnet when subjected to different field combinations. ............................ 13

2.2 Example schematic of anti-ferromagnetically ordered lines on clad copper line [4]. (This figure appears in [54].) ............................ 14

2.3 (a) CNN topology; (b) Circuit layout for a conventional CNN cell. ............................ 17

2.4 The output transfer function for saturating the output at $\pm 1 \text{V}$. ............................ 19

3.1 Phase diagram of a $39 \times 63 \times 5 \text{nm}^3$ Co magnet with biaxial anisotropy constant $K_1 = 40 \text{kJ/m}^3$; generated via (a) micro-magnetic simulation, (b) analytical approach. Each point in the diagrams depicts the angle of magnetization when subjected to different field combinations. 24

3.2 Design methodology flowchart. ............................ 31

3.3 (a) Initial layout for an AND gate. (b)-(j) the desired switching order of the magnets during logic evaluation for input $ab=\downarrow\downarrow$. ............................ 32

3.4 The final AND layout with $d_x = 9 \text{nm}$ and $d_y = 21 \text{nm}$. Both the helpers are $87 \times 39 \times 5 \text{nm}^3$ in dimension. Helper1 is $18 \text{nm}$ left from $C$ and Helper2 is $6 \text{nm}$ right from $o_3$. ............................ 36

3.5 Final layout for (a) a corner turn, (b) a majority gate, and (c) a cascaded AND-OR logic. ............................ 38

3.6 Final layout (at 300K) for (a) a majority gate, and (c) a cascaded AND-OR logic. ............................ 40

4.1 TFET I-V characteristics [59] used as the basis for this work. ............................ 45

4.2 TFET-based CNN cell circuit. ............................ 45

4.3 Horizontal Line Detection with TFET-CNN: (a) input image; (b) output image. ............................ 48

4.4 DP plots for HLD. Here $k = 40 \times 10^{-6}$ such that the state equilibrium points (marked with open circles) are generated approximately at $-0.8 \text{V}$ (for a white output), and $0.1 \text{V}$ (for a black output). ............................ 48
4.5 Simulation results of TFET-based CNN array for different image processing templates. .................................................. 50
4.6 (a) SPICE circuit model for the TFET; (b) TFET I-V characteristics for $V_{gs} = 0.3 \, V$ [59] and the fitted SPICE model. ............ 52
4.7 Dynamic and static power estimation assuming different device operating characteristics for transitions to both white and black pixel values. ................................................................. 56
4.8 Charge required for TFET-based cell, $Q_T$ and conventional cell, $Q_R$. For $Q_R$ to be equal to $Q_T$, $|I_{eff}^R|$ needs to be higher than $|I_{eff}^T|$. ... 57
4.9 Assumed models for low voltage TFETs. ............................ 57

5.1 (a) Example characteristics of a CNN device for multi-valued problems; (b) I-V characteristics of the double-layer graphene device at room temperature [13], and its Piece-wise linear (PWL) approximation. ... 61
5.2 DP plots for the representative cases of the four classes with $k = 7 \times 10^{-9}$ and $\varepsilon = 3 \times 10^{-9}$. The state equilibrium points are marked with open circles. ................................................................. 63
5.3 Simulation results for quaternary classification. .................... 65
5.4 (a) Circuit for a ternary cell. (b) Its transfer characteristic with $V_{B1} = V_{B4} = 0.1 \, V$, $V_{B2} = V_{B3} = -0.1 \, V$, $V_{DD} = -V_{SS} = 0.25 \, V$. The width of all TFETs is 1 $\mu m$. (This figure appears in [75].) ............... 65
5.5 (a) Sweeping $V_{B1}$ and $-V_{B2}$ from 50 mV to 150 mV, (b) Sweeping $V_{B4}$ from 50 mV to 150 mV. (Other parameters are similar to those of Fig. 5.4b. This figure appears in [75].) ............... 66
5.6 (a) Profile of the vertical component of sheer stress, $T_y$ from [50]. In region ‘a’, nothing is grasped. $T_y$ increases proportionally with the weight, in region ‘b’. During ‘c’ the object starts to slip out of the robot’s grasp. In ‘d’, $T_y$ becomes stable as it equal with the kinetic friction force; (b) The gradient profile of $T_y$. ....................... 67
5.7 The Universal Machines on Flows (UMF) [82] description of the TFET circuit-based CNN for detecting slippage. .................. 69
5.8 DP components for (a) TFET circuit-based CNN with the center matrix element of the feedback template, $a_{00} = 7 \times 10^{-6}$; (b) Conventional CNN with 1 $\Omega$ resistance. ................................. 69

6.1 Assumed CMOS-based OTA driven CNN cell circuit. ............ 72
6.2 PWL approximation of DP plot for a state transition from $x_0$ to $x_n$. .......................... 76
6.3 Simulation outputs of a $32 \times 30$ CNN for (a) Optimal edge detection, and (b) Convex corner detection tasks. Correct operations are observed. 79
6.4 OTA circuits from [61] built of (a) CMOS devices, (b) TFET devices. 80
6.5 Histograms of ten thousands simulated power dissipations for (a) Optimal edge detection, and (b) Convex corner detection problems.

6.6 Different non-linear characteristics considered in Table 6.1.

6.7 The Universal Machines on Flows (UMF) description for detecting slippage via: (a) Conventional linear CNN; and (b) TFET circuit based ternary CNN. Each horizontal bar represent individual step of a given CNN work flow.

6.8 Computational steps for the pattern recognition problem.

7.1 (a) CNN architecture for zero-feedback applications; (b) a cell in $i^{th}$ row and $j^{th}$ column. (This figure appears in [90].)

7.2 Voltage to pulse-width conversion and measurement. (This figure appears in [90].)

7.3 Uncoupled CNN cell’s Driving point plot for different values of $\alpha$ (R = 1 $\Omega$). $(-x_{max}, x_{max})$ defines the range of input voltages.

7.4 Direct Digital Synthesizer (DDS). (This figure appears in [90].)

7.5 Augmented MS cell circuit for uncoupled CNN templates.

7.6 TFET-based (a) Comparator circuit, and (b) Ramp generator from [90].

7.7 Simulated ramp voltage and its error for TFET-based and CMOS-based ramp generators.

7.8 Simulation results for power dissipation of 8 bit counter and DDS block. All supply voltage is 0.4V.

7.9 Verilog simulation outputs of a $32 \times 30$ CNN for (a) Optimal edge detection, and (b) Convex corner detection tasks.

7.10 Comparisons between the TFET-based architectures to different CMOS digital and analog processors (GOPS/W vs. MOPS). $\nabla$ and $\circ$ indicate existing digital and analog implementations, respectively.

7.11 Operation of MS architecture for a template with two functions $f1$ and $f2$. (a) For regular CNN template (no delay) computation, (b) for delay-based template computation.

7.12 Simulation outputs of the delay-based motion detection task for an object moving (a) at a given threshold speed, (b) slower than the threshold, and (c) faster than the threshold. Object is identified by the presence of ‘black’ pixel.
ACKNOWLEDGMENTS

First, I would like to thank my advisor, Dr. X. Sharon Hu for her guidance throughout the journey. She motivated me to push myself, constantly challenged me to search for appropriate insights, and ensured that I developed as a researcher. I also thank my co-advisor, Dr. Michael T. Niemier for his time, energy, and counsel.

I sincerely thank Dr. Joseph Nahas who have always found time for my questions. Similarly, I am thankful to fellow graduate students, Dr. Steve Kurtz, Dr. Shawn Liu, Qiuwen Lou, Xunzhao Yin, and former post-doc Dr. Behnam Sedighi, all of whom have worked closely with me over the years.

I am also grateful to the Semiconductor Research Corporation (SRC), Nanoelectronics Research Initiative (NRI) for their financial support of my research.

Finally, I would like to thank my father Bimal Kanti Palit, my mother Debi Palit, and my brother Soumitra Palit who have been continuously supporting me throughout my life, and encouraging me in all my ventures.
CHAPTER 1

INTRODUCTION

1.1 Overview

For the last several decades, transistors have continued to enjoy exponential growth dictated by the Moore’s Law. As a direct consequence of this growth, Complimentary Metal-Oxide-Semiconductor (CMOS) technology experienced remarkable improvements in chip-level performance [9]. Each technology generation has approximately reduced the device size by 50%, and device delay by 70% when compared to the previous one. Following such trends, performance improvements over generations could be sustained for the same power budget provided that supply voltage had also been scaled by 30% per generation. Apart from improving device-level performance, these scaling trends have also improved application-level performance by enabling advanced architectural innovations, e.g., pipelining, on-chip memory (caches), out-of-order execution, etc.

However, as feature sizes approach physical limits (i.e, atomic dimensions) – lithographic challenges, and device-to-device variation associated with such miniaturization make it very difficult to scale the supply voltage\(^1\) [12]. As such, the exacerbated off-state leakage currents of devices lead to unmanageable chip-level power densities (more than 100 W/cm\(^2\), a practical limit for air cooling [97]). Hence, in 2003 and 2004, the scaling trend for the devices became unsustainable [64].

\(^1\)The ITRS [45] predicts that supply voltage would be reduced by approximately 0.3V over the next 13 years, which falls very short from the desired 30% reduction per generation.
Advanced techniques such as high-κ gate dielectrics [11], multi-gate transistors [10], etc., have still allowed to continue scaling by helping to keep power budgets in-check. Additionally, multi-core computer architecture [53] can help to increase throughput by leveraging application-level parallelism when compared to single-core alternative with same device-level performance. However, sustaining the performance trend via core scaling does not look promising [33] as (i) the overall improvement in performance is contingent on the amount of inherent parallelism available in the program, and (ii) each core is essentially still limited by the aforementioned physical constraints. A recent study [12] demonstrates that performance gains (over 11 technology generations) accounting for both the technology scaling, and the architectural advancement, have begun to “flat line.” If core scaling continues as is, power densities will approach 500 W/cm² [12].

To continue the performance scaling trends associated with Moore’s Law, solutions are sought at both the device and architectural levels. To this end, the Semiconductor Research Corporation (SRC) launched the Nanoelectronics Research Initiative (NRI) to identify and demonstrate new devices capable of replacing CMOS Field-Effect Transistors (FET). Also, the STARnet initiative of SRC/DARPA is looking for new energy efficient architectures to replace the boolean logic based, traditional von Neumann (i.e., stored program) computing architecture. As part of these initiatives, my research has found two ramifications: (i) Developing a systematic circuit design methodology – for the emerging Nanomagnet Logic (NML) devices [43, 24] – that is capable of identifying functional circuit layouts deterministically. In NML devices, state is represented without the use of electric charge. Hence, it has a great potential to alleviate undesirable side effects of CMOS scaling; and (ii) Exploring the benefits of utilizing low voltage, steep sub-threshold, emerging transistors (e.g., Tunneling Field-Effect Transistors (TFETs) [88], Graphene transistors [13], etc.) to process information with the non-binary/non-von Neumann Cellular Neural Networks.
(CNNs) \cite{21, 22}. CNN is a powerful analog array processor architecture that can significantly improve both the power and performance of various computation-intensive information processing applications, e.g., image processing, pattern recognition, motion detection, real-time DNA analysis, etc. \cite{20}.

1.2 My Contributions

In this section I briefly outline my scholarly contributions in the context of NML circuit design, and emerging device based CNN architectures.

1.2.1 NML Circuit Design

In NML lithographically-defined magnetic islands represent states by their direction of magnetization. Boolean logic operations and signal propagation are accomplished via fringing field interactions between the magnets \cite{43, 24}. NML demonstrates a number of potential benefits over CMOS: (i) devices can be made to function above superparamagnetic limit, and thus can maintain states without active power supply (i.e., non-volatile). This can potentially enable new application spaces; (ii) devices are intrinsically radiation hard – making them suitable for systems operating in harsh environment, e.g., in outer space, and (iii) switching device state is a low power operation, etc.

However, despite its promising features NML lacks research efforts towards developing a systematic design methodology for a target logic. The existing approach for designing NML circuits is more like a ‘Trial and Error’ type \cite{24, 17}. A fair number of design variables like spacings between the magnets, clocking field strength, helper cell dimensions and positions etc. are involved in the design process. The designer tries to make good educated guess for these parameter values based on previous experience. Often the predicted behavior contradict with the actual behavior of the magnets – prompting some changes in the design and repeated trials.
The magnetization state of a magnet is determined by the external magnetic fields the given magnet is subjected to. In an ensemble any magnet could influence any other magnet through unwanted fringing field interactions. Furthermore, stochastic thermal noise associated with ambient temperature can also potentially impact the desired switching order of the magnets, and thus lead to logically incorrect magnetization states. In the followings, I discuss the specific contributions of my research to address the aforementioned challenges.

1.2.1.1 Phase Diagram Based Analysis

To systematically design a functional NML circuit, it is very important for the designer to have: (i) a “knowledge” about the effects of external magnetic fields over a given magnet, and (ii) “control” over the external magnetic field such that a given magnet can be subjected to proper excitations. To capture the “knowledge”, I propose the concept of “Phase Diagram” that records a device’s response when subjected to different external fields (see a representative example in Fig. 3.1a). Each point in a phase diagram represents the angle of magnetization at which a given magnet settles when subjected to a pair of x-directed ($H_x$) and y-directed ($H_y$) fields. There exists a unique phase diagram for a magnet with given material properties and dimensions. Two approaches for generating the phase diagram are proposed: (i) via micro-magnetic simulation, and (ii) via analytical approach. Once generated, the phase diagram can be continuously re-used when designing any circuit.

To manifest the “control”, I have also proposed an analysis technique – based on the phase diagram – to determine the necessary conditions for a given magnet to remain in or transition to a desired magnetization state. The proposed conditions are formulated around some quantifiable metrics that can be derived from the phase diagrams.
1.2.1.2 Systematic Design Methodology

Using the aforementioned analysis, I have developed a methodology for generating NML layouts that are guaranteed to function correctly. Once the necessary conditions for maintaining/transitioning the magnetization states are discovered, the next logical research problem is how those constraints can be satisfied to ensure that a given magnet (i) switches to a logically correct magnetization state at an appropriate time, and (ii) otherwise remains in a neutral/metastable state. To this end, a series of well defined algorithmic steps, and rules are outlined. The cumulative objective of all these steps/rules is to address the vulnerability (if there any) of each magnet in the ensemble. By supplying a target structure and desired ground state, the proposed methodology can be employed to determine device sizes, spacings, clock field strength etc. The methodology can also be implemented as a CAD tool to facilitate faster layout identification.

The methodology is applied to identify layouts for 2-input AND gates, 3-input majority gates, “corner turns” (comprised of anti-ferromagnetic lines that transition to ferromagnetic lines), and cascaded AND-OR logic structures which eventually could be used to construct any boolean circuit. The functionality of these layouts are verified via micro-magnetic simulations.

1.2.1.3 Addressing Thermal Noise

I have also enhanced the proposed design methodology to augment initial layouts (targeted for 0K simulations) such that logically correct ground states could be achieved when considering stochastic effects associated with ambient temperatures. Thermal energy essentially introduces an additional random magnetic field for all the magnets in the ensemble. This noise can have unwanted impact on the behavior of NML circuits and could lead to logically incorrect magnetization states.

To systematically address the thermal issues, I have identified three distinct types
of unwanted effects that thermal excitation can have on individual magnets: (i) Premature switching, (ii) Over-stability, and (iii) Race condition. The general strategies for overcoming each of these types of problem are discussed. Following these strategies all the initial layouts – designed for operating at 0K – are augmented for room temperature (300K), and verified via micro-magnetic simulations for 100 different random seeds.

1.2.2 CNN Based on Emerging Devices

Historically, most information processing hardware essentially implements a von Neumann (i.e., stored program) architecture which is based on Boolean logic, and operates at discrete time on discrete, binary coded data. This paradigm has obviously and deservedly continued to enjoy exponential growth due to Moore’s Law scaling. Given that scaling trends are at horizon, investigations on power-efficient architectures are of high interest. A CNN is a highly parallel, analog processor [21, 22] that can significantly outperform von Neumann architectures for certain classes of problems. CNN can be particularly efficient when compared to functional equivalents that are executed on more traditional programmable hardware (e.g., for image processing applications). Quantitatively, for complex, 2D image processing functions, a commodity, CNN-based processor with an area of 1.4 cm² and a power budget of just 4.5 W could match the performance of the IBM cellular supercomputer (i.e., traditional programmable hardware) with an area of 7 m² and a power budget of 491 kW [83].

Emerging devices with characteristics like low-voltage operability, steep subthreshold swing, have great potential for ultra-low power application spaces. For example, TFETs provide higher drive current than the MOSFET at lower voltages (e.g., 0.1 V to 0.3 V) [88]. TFETs have shown to be extremely power efficient for logic circuits [18]. Furthermore, devices like Graphene transistors [13], and Symmetric tunneling
FETs (SymFETs) [99], etc. demonstrate useful properties in their I-V characteristics that can lead to non-binary outputs, and thus can help to reduce the number of computational steps to solve a particular problem. Given that, in this work, I consider how the emerging transistor technologies could impact the performance, efficiency, and application spaces of CNN. In the followings, I discuss the specific contributions made by this work.

1.2.2.1 TFET-based CNN

In conventional CNN, a linear resistor is used that performs the computation by settling the state (i.e., stored charge in a capacitor connected in parallel to the resistor) to an equilibrium voltage. As the state is linearly related with the programmed input currents, conventional CNN employs a non-linear output transfer function that saturates the output to either a ‘high’/‘low’ signal if the state overshoots a predefined threshold. It is also possible to introduce non-linearity within the state voltage itself by using a non-linear device/circuit instead of the linear resistor. Utilizing non-linear devices like Resonant Tunneling Diodes (RTD) in CNN architectures was first investigated in [39], and further extended by [44] and [65]. However, fabrication of RTD devices have been challenging in practice [15].

I have demonstrated that the non-linear properties of TFETs can also be successfully employed in a CNN system to solve various image processing tasks. In general, such introduction of non-linearity at state-level obviates the necessity of the output transfer function which in turn reduces the hardware footprint for each cell in a CNN array. I have also calculated the power dissipated at the cell-level, and illustrate that power dissipated by a TFET-based cell would always outperform the power dissipated by a conventional cell. My studies suggests that further power savings are possible by carefully engineering lower voltage, lower current TFET devices without sacrificing performance.
1.2.2.2 Desired Device Characteristics

Although, TFET-based non-linearity can successfully solve tasks, with positive impacts on power, performance, and area; still the capabilities of such CNN system is limited to binary classifications only. To further enhance the capabilities of CNNs for solving problems with non-binary outputs (e.g., ternary/quaternary classifications), one first need to know the expected non-linear characteristics that can satisfy the classification requirements. To this end, I have outlined the desired characteristics that a particular non-linearity should demonstrate to solve multi-valued classification problems. The characteristics are outlined by considering properties including output stability and flexibility as well as sensitivity to noise.

1.2.2.3 Multi-valued CNN

I also demonstrate how emerging, beyond-CMOS devices – that conform with the outlined desired non-linear characteristics – could help to further enhance the capabilities of CNNs for solving problems with non-binary outputs. Recent work on double-layer graphene transistors [13] suggests that a single device (Fig. 5.1b) might be able to deliver the desired I-V characteristics. In this work, it is shown that CNNs based on devices such as graphene transistors – with multiple steep current growth regions separated by negative differential resistance (NDR) in their I-V characteristics – could be used to recognize multiple patterns simultaneously. This would require multiple steps given a conventional, binary CNN – costing longer time and higher energy. Also, it is demonstrated how TFETs can be used to form circuits capable of performing similar tasks. With this approach, more “exotic” device I-V characteristics are not required – which should be an asset when considering issues such as cell-to-cell mismatch, etc. As a case study, a CNN-cell is designed that employs TFET-based circuitry to realize ternary outputs. This hardware is employed to efficiently solve a tactile sensing problem from [50] where the goal is to detect and
identify slippage of an object from a robotic arm. TFET-based cells with ternary output characteristics could reduce hardware requirements by 2X and the number of required template operations by 60% when compared to an approach based on a conventional CNN. This is irrespective of any performance/energy benefits from the technology itself.

1.2.2.4 Modeling CNN Energy Dissipation

To estimate the energy dissipation of a CNN system, I have proposed analytical models for estimating power, and performance (measured as settling time) for a given CNN application. Similar to designing von Neumann architectures, being able to quickly estimate power and performance of CNN is critical for exploring different design options. Existing model for calculating a CNN’s settling time [38] is applicable only on linear resistor based CNNs. The proposed models can be used for both linear and non-linear CNNs. Aside from making very good estimations, the proposed models also explain the impact of various system parameters (e.g., the synaptic weights, resistive elements’ characteristics, etc.) on energy dissipation. Applying the performance model, it is shown that more convex non-linear characteristics can enable faster signal processing. This might be a critical feature when considering potential emerging devices in the context of non-linear CNN architectures.

Electrical circuit simulation results suggest that the models can predict power and performance very reliably with errors less than 1% and 3%, respectively. These models are also applied to compare power and performance between a TFET-based non-linear CNN, and a conventional linear CNN for a tactile sensing and a pattern recognition problem.
1.2.2.5 Mixed Signal Architecture for CNN

I also propose a TFET-based non-von Neumann, mixed signal (MS) architecture for energy efficient processing of spatio-temporal applications. The traditional, CMOS-based von Neumann architecture performs poorly when processing spatio-temporal information, where a high volume of signals from different space and time coordinates needs to be processed simultaneously to make some inference. Examples of such applications include image processing, video analytics, pattern recognition/formation, etc. Furthermore, CMOS scaling trends are accompanying unwanted side effects like increased device mismatch, power densities, etc. That said, exploring non-von Neumann computing architectures in the context of new device technologies are of heightened interest.

The proposed MS architecture is augmented from an existing TFET-based architecture in [90]. The architecture in [90] can only process zero-feedback CNN operations, and it cannot limit its output within a preset threshold. The MS architecture is not restricted by these limitations, and can perform a much wider range of complex applications. In contrast to existing works, the proposed MS architecture adopts a predefined algorithmic steps to infer the state equilibrium. Simulation results suggest more than 2X improvement in power efficiencies for this architecture over state-of-the-art architectures that assume CMOS technology and that performs similar tasks.
CHAPTER 2

BACKGROUND

In this chapter I briefly review the technological concepts to help the readers comprehend the context of my research contributions.

2.1 Nanomagnet Logic

NML devices are single domain, nano-scale magnets that can represent, move, and process binary information. In the absence of any externally applied magnetic fields, a magnet’s shape, crystal structure dictate a preferred direction of magnetization which is called the “easy” axis. Magnet with an aspect ratio greater than 1 (i.e., taller than it is wide) tends to be magnetized in the y-direction (longer axis) due to shape anisotropy. The two possible stable states in the y-direction (i.e., “up” and “down”) can be used to represent a logic ‘0’, and logic ‘1.’

2.1.1 Circuit Constructs

A wire (Fig. 2.1a) can be formed from a horizontal line of magnets that are anti-ferromagnetically (AF) coupled with each other. Similarly through ferromagnetic coupling it is also possible to build vertical interconnect. This coupling can be utilized to move data and process logic functions. Both horizontal and vertical interconnects have been demonstrated experimentally [43, 77]. A functionally complete logic set can be realized with combinations of majority voting gates where output is determined by the majority of three inputs. By setting one input to a logic 0 or
1, the gate can execute an AND/OR (or NAND/NOR) function [43]. This “parts library” has been expanded to include programmable majority gates, fanout, and non-majority AND/OR logic [71]. All have been experimentally demonstrated at room temperature.

2.1.2 NML Clocking

Ideally, NML circuits remain in a stable ground state determined by the previous input combination (Fig. 2.1a). However, the energy difference between the two stable magnetic states can be large. Therefore, an external stimulus is needed to re-evaluate a magnet ensemble (e.g., Fig. 2.1a) with new inputs (Fig. 2.1b, c) to overcome (i) the intrinsic demagnetizing energy barrier (EB) and (ii) any zeeman energies contributed by fringing fields from neighboring magnets that reinforce the original state. The external stimulus can be provided in the form of a clocking magnetic field applied along the hard axes of the magnets.

Ensembles of magnets could be grouped in clocking zones. Fig. 2.1b illustrates the effects of the clocking field: magnets of clocking zone N and N+1 are put into a metastable state (i.e., along the hard/short/x-axis) against the preferred shape anisotropy. As magnets always tend to be magnetized along a direction such that the total energy is minimized, if a driving neighbor provides a y-directed field, the driven magnet’s magnetization rotates toward a preferred easy (long/y) axis (Fig. 2.1b-i). When the clock is removed, magnet \( m_1 \) will relax to a new, low energy ground state (Fig. 2.1b-ii, iii). Succeeding magnets \( m_2, m_3, \) and \( m_4 \) should then relax in order.

The external clocking field can be generated in a number of ways. In this work, we mainly consider the approach proposed in [70], where to modulate the EBs of magnet ensembles “on-chip,” a current-carrying copper wire is wrapped in a ferromagnetic material such as permalloy (see Fig. 2.2). When the wire is conducting current, a magnetic field is generated above the surface of the wire where the magnets are
Figure 2.1. (a) AF-line moves information; (b) an AF-line has a new input, and an external clocking field is used to facilitate re-evaluation of the line; (c) as the field is removed, devices relax along their easy axes; (d) Energy landscape of a representative magnet when subjected to different field combinations.

The ferromagnetic cladding helps to direct increased magnetic field over the magnets. As suggested by [30], the critical path for NML circuits could be in the order of tens of magnets. As such, multiple clock wires are placed adjacent to each other (as in Fig. 2.2).
Figure 2.2. Example schematic of anti-ferromagnetically ordered lines on clad copper line [4]. (This figure appears in [54].)

Clock energy could be amortized over 100,000s of devices as a single clock line could control many parallel ensembles. Clock lines can be placed in series and in multiple planes to minimize driver overhead [71]. Regarding experiments, CMOS compatible clocks [70] have been (i) fabricated [8], (ii) used to switch the state of individual magnetic islands [3], and (iii) used to re-evaluate line and gate structures with new inputs [4].

Apart from the aforementioned “group clocking” approach, other approaches for clocking have also been proposed in the literature. [5, 6, 8] have considered using on-chip solenoids for clocking individual magnets. Such approach of controlling individual magnets facilitates excellent control over information propagation, and also renders a simplified model of switching order for the ensemble. However, the required width of the solenoid coil is 1 nm, and so far there is no feasible way of fabricating these very small features. On the other hand, [17, 93] proposed global clocking by simultaneously nulling and then relaxing all the magnets in an ensemble. This approach assumes the magnets to remain in a metastable state until the appropriate
preceding magnet switch – which is very unlikely in the presence of thermal noise.

Another approach for clocking NML circuits involves introduction of multiferroic materials that provide a mean to control magnetism via electric fields. In multiferroic materials ferroelectricity (FE) coupled with magnetism leads to magnetoelectric susceptibility \[18, 36, 78, 86, 92\]. \[34\] suggests that multiferroic NML devices can be individually controlled by low voltage pulses (10s of milli-volts) – allowing for energy-efficient operation. Finally, clocking based on domain walls \[37, 26\] might represent another potential approach. While current driven clocking is the context for this work, these alternative forms of clocking are also potentially applicable.

2.2 Cellular Neural Networks

CNN is a powerful analog array processor architecture \[21, 22\] that can significantly improve both the power and performance of various computation-intensive information processing applications, e.g., image processing, pattern recognition, motion detection, real-time DNA analysis, etc. \[20\].

2.2.1 Historical Perspective

The human brain is an extremely complex non-linear system consisting of approximately 100 billions of identical computational units called neurons. Existing von Neumann digital computing architectures faces daunting challenges while solving many practical problems e.g., image processing, pattern recognition, etc., whereas human brain performs those task on the fly. Inspired by human brain’s capabilities, early research initiatives introduced a simplified artificial model of the brain – called Artificial Neural Networks (ANNs) \[41\]. In ANNs the identical computational units – also called neurons – perform simple non-linear functions. The Hopfield network \[42\] represents one important class of fully connected ANNs where the output values of all neurons are fed back to the inputs of all neurons. This high degree of connec-
tivity of the Hopefield network renders its VLSI realization infeasible. In 1988, Chua and Yang proposed the cellular neural networks (CNNs) which is a special case of continuous-time Hopfield network. The key features of CNN that makes it attractive are: (i) the interconnections between the processing units – called cells – are local, which means a cell is connected only to its neighbors; (ii) the cells and their synaptic interconnections are usually space-invariant, which makes CNN very suitable for VLSI implementation; (iii) continuous-time processing of analog signals in a highly concurrent manner by the cells allows for massive real-time, fast signal processing with low energy dissipation. CNN can be particularly efficient for “brain-like” computations (e.g., image processing) when compared to functional equivalents that are executed on more traditional programmable hardware. Significant research efforts have been devoted to realize numerous CNN systems using CMOS technology (e.g., \[84, 47, 52, 49\]) for a wide range of applications. Extensive sets of image processing templates for CNN also exists \[25\].

2.2.2 CNN Basics

The conventional single-layer, spatially invariant CNN architecture proposed by Chua and Yang \[21\] is an \(M \times N\) array (see Fig. 2.3a) of identical cells, where each cell, \(C_{ij}, (i, j) \in \{1, \ldots, M\} \times \{1, \ldots, N\}\), has identical synaptic connections with all the adjacent cells in a predefined neighborhood, \(N_r(i, j)\) of radius \(r\). The size of the neighborhood depends on how far the central cell, \(C_{ij}\), connects to its adjacent cells, which is \(m = (2r + 1)^2\), where \(r\) is a positive integer. For example, if \(r\) is equal to 1, the neighborhood of a given cell would include all the 8 nearest neighbors and the cell itself.

A conventional CNN cell (see Fig. 2.3b) consists of one resistor, one capacitor, \(2m\) linear voltage controlled current sources, one fixed current source, and one non-linear voltage controlled voltage source. The node voltages \(u_{ij}, x_{ij}, \text{ and } y_{ij}\) correspond
to the input, state, and output of a given cell $C_{ij}$, respectively. The input and output voltages of the $m$ neighbors contribute $m$ feedback and $m$ control currents, respectively, to a given cell through the linear voltage controlled current sources. The dynamics of the CNN can be expressed by a system of $M \times N$ ordinary differential equations (ODEs), each of which is simply the Kirchhoff’s Current Law (KCL) at the state nodes of the corresponding cells:

$$C_i \frac{dx_{ij}(t)}{dt} = -\frac{x_{ij}(t)}{R} + \sum_{C_{kl} \in N_r(i,j)} a_{ij,kl} y_{kl}(t) + \sum_{C_{kl} \in N_r(i,j)} b_{ij,kl} u_{kl} + Z \quad (2.1)$$
where $a_{ij,kl}$ and $b_{ij,kl}$ act as weighting parameters for the feedback and control currents from cell $C_{kl}$ to cell $C_{ij}$. Due to the space invariant nature of the synaptic connections between cells, these parameters are denoted by two $3 \times 3$ matrices (for $r = 1$), namely the feedback template $A$ and the control template $B$. The fixed bias current, $Z$, provides a mechanism to adjust the total current flowing into the cell, and thus enhances design flexibility.

At any given time, each cell in the network would have a net current, $I_{eff}$, flowing into its state node $x$, from the voltage controlled current sources, and the fixed bias. This current would depend on (i) the input signals (at node $u$), (ii) the output voltages (at node $y$) of all connected neighbors, and (iii) the magnitude of the bias current. $I_{eff}$, in conjunction with any pre-existing initial voltage at the state node (due to the charge stored in the capacitor), results in a new stable voltage eventually such that all the current components at node $x$ come to an equilibrium (according to Eq. 2.1). To be specific, at equilibrium the current through the capacitor would be zero, and $I_{eff}$ would be equal to the current conducting through the functioning device (which is resistor in this case). By carefully selecting the values of the $A$ and $B$ templates as well as $Z$, it is possible to solve a wide range of image processing problems or more general ‘binary classification’ problems.

To ensure fixed binary voltage levels at the output, a conventional CNN cell typically employs a non-linear sigmoid-like output transfer function $f(x_{ij}(t))$, as depicted in Fig. 2.4. The transfer function:

$$y_{ij}(t) = \frac{1}{2} (|x_{ij}(t) + 1| - |x_{ij}(t) - 1|)$$

(2.2)

cause the output state to saturate at either a high or low voltage (e.g., $+1\,V$ or $-1\,V$).
Figure 2.4. The output transfer function for saturating the output at $\pm 1 \, V$. 
CHAPTER 3

NML DESIGN METHODOLOGY

3.1 Overview

It is well known that CMOS scaling trends are now accompanied by less desirable byproducts such as increased energy dissipation. Devices where state is represented without the use of electric charge could alleviate undesirable side effects of CMOS scaling – especially when considering low power application spaces. NML (Nanomagnet Logic) is one such technology. [1] suggests that NML could best CMOS hardware equivalents at iso-performance even after accounting for clock overhead.

Still, a systematic method for designing NML circuits – for experimental and simulation targets – does not exist, and nearly all design efforts have been performed in a “trial and error” fashion [24], [17]. While the structure of a given NML ensemble is often obvious – e.g., for fanout or for a majority gate – the design parameter space is large. Device-to-device spacing, magnet size, aspect ratio, material, crystal structure, thickness, and shape can all impact whether or not a magnet ensemble evolves to a logically correct ground state. Finding a functionally correct layout is non-trivial. Any magnet in an ensemble could influence any other magnet through unwanted fringing field interactions. Thermal noise can also impact switching order and produce undesirable states.

Here I present a systematic design method for NML circuits that marshals all of the aforementioned design parameters to generate a layout that will function correctly during initial physical-level simulations. The approach identifies design parameters
that lead to safe operating windows. The end result is a layout that can be generated via an automated process where devices evolve in a predictable, and logically correct manner.

The main contributions of this chapter are as follows:

1. An analysis technique is introduced to determine the necessary conditions for a given magnet to remain in or transition to a desired magnetization state.

2. Using the aforementioned analysis, I have developed a methodology for generating NML layouts that are guaranteed to function correctly. By supplying a target structure and desired ground state, the methodology can be employed to determine device sizes, spacings, etc. in an automated fashion.

3. The design methodology is enhanced to augment initial layouts (targeted for 0K simulations) such that logically correct ground states could be achieved when considering stochastic effects associated with higher temperatures.

The proposed methodology has been used to identify layouts for 2-input AND gates, 3-input majority gates, “corner turns” (comprised of anti-ferromagnetic lines that transition to ferromagnetic lines), and cascaded AND-OR logic structures. The rest of the chapter is organized as follows: First in Section 3.2, I describe the simulation setup that is used to verify the proposed layouts’ functionality. In Section 3.3, the core concepts associated with the design methodology are discussed. In Section 3.4, I illustrate the use of the method for designing a 2-input AND gate. Section 3.5 reports the layout results for different logic structures. Section 3.6 describes how the methodology can be used to modify 0K design targets such that they switch reliably at room temperature. Finally, I conclude in Section 3.7.

3.2 Simulation Setup

Micro-magnetic simulation is a valuable tool for predicting the behavior of an NML ensemble. In this work, I use the Objected Oriented Micro-Magnetic Frame-

1 The identified layout is not claimed to be an optimal one; rather a working target is identified, without trial and error simulations, that can be optimized further.
work (OOMMF) \cite{31} developed by NIST as the simulation tool. All simulations assume magnets made from Cobalt (Co) with a magnetocrystalline biaxial anisotropy constant $K_1 = 40 \, KJ/m^3$, which further promotes hard axis stability during switching \cite{17}. Per \cite{17}, a saturation magnetization of $10^6 \, A/m$ and an exchange stiffness constant of $1.3 \times 10^{-11} \, J/m$ are used. A damping coefficient of 0.05 is assumed. A rounded rectangular shape with a $39 \times 63 \times 5 \, nm^3$ footprint (a multiple of the $3 \times 3 \times 5 \, nm^3$ simulation mesh) is used for magnets in horizontal and vertical wires. Helper magnets with easy axes parallel to the direction of the applied field can provide static, hard-axis biasing fields over a target magnet when needed. The design methodology is also employed to size and place these helper structures.

3.3 Overview of Phase Diagram Based Design

In this section a fundamental concept of the proposed design methodology is introduced – the phase diagram. I explain how a phase diagram captures a device’s response when subjected to different field combinations, as well as how it can be used to determine the required external fields over a given magnet to ensure that it (i) switches to a logically correct, easy axis magnetization state at an appropriate time, and (ii) otherwise remains in a metastable state. I end the section by presenting the overall approach.

3.3.1 Phase Diagrams

In an ensemble, the external magnetic field that a magnet experiences varies over the course of a computation as neighboring magnets switch and/or the magnitude of the clock field changes. Thus, the energy profile of a magnet also evolves over time, and an energy minimum can shift to different positions (Fig. 2.1d) causing the magnetization vector to settle at different angles. To correctly predict the magnetization state of a magnet over time, it is important to capture how it reacts under
the influence of any external field. I introduce phase diagrams (see a representative example in Fig. 3.1a) to achieve this goal. Each point in a phase diagram represents the angle of magnetization at which a given magnet settles when subjected to a pair of x-directed \( H_x \) and y-directed \( H_y \) fields. In Fig. 3.1a, one can clearly distinguish three mutually exclusive regions (defined by overlaid dashed lines) on the phase diagram corresponding to three magnetic states: ‘Up’ (\( \uparrow \)), ‘Down’ (\( \downarrow \)) and ‘Metastable’ (\( \rightarrow \)). A magnet is assumed to be in a \( \uparrow \) or \( \downarrow \) state if its angle of magnetization is greater than 50° or less than −50°, respectively. If the angle of magnetization is between −10° and 10°, the magnet is considered to be in a metastable state. (The two remaining areas do not precisely represent any stable magnetization state, and hence those regions are avoided.)

For a magnet with given material properties and dimensions, a single corresponding phase diagram can be (i) generated via either micro-magnetic simulation or analytical approach, and (ii) continuously re-used when designing a circuit (even if device-to-device spacing changes).

3.3.1.1 Phase Diagrams via Simulation

Phase diagrams can be generated via micro-magnetic simulators (e.g., OOMMF) by subjecting a single magnet to varying \( H_x \) and \( H_y \) fields in the range that a device may experience from its neighbor’s fringing fields and/or the clock. When generating a phase diagram, the device is initially hard axis biased – an intermediate state associated with clocking (see Fig. 2.1b). It is then subjected to different field contributions and allowed to relax for a sufficient period of time such that the magnetization vector reaches a steady state. (Here, 1.5 ns was sufficient.) Final angles of magnetization are then plotted for each \( H_x \) and \( H_y \) pair.
Figure 3.1. Phase diagram of a $39 \times 63 \times 5 \text{ nm}^3$ Co magnet with biaxial anisotropy constant $K_1 = 40 \text{ kJ/m}^3$; generated via (a) micro-magnetic simulation, (b) analytical approach. Each point in the diagrams depicts the angle of magnetization when subjected to different field combinations.
3.3.1.2 Phase Diagrams via Analytical Approach

Phase diagrams can also be generated by analytically identifying the energy minimum for a given magnet. For a given angle of magnetization, a magnet’s energy is the sum of the following energy terms: (i) exchange energy, (ii) magnetostatic energy, (iii) magnetoelastic anisotropy energy, (iv) magnetocrystalline anisotropy energy, and (v) zeeman energy. The first four energy terms are not dependent on the external applied fields, and can be calculated easily through simulation. The zeeman energy for a given external field, can also be calculated by using the following expression:

\[ E_{\text{zeeman}} = -\mu_0 VM_S (H_x \cos \theta + H_y \sin \theta) \]  

(3.1)

where, \( \theta \) is the angle of magnetization. Thus, by adding the energy terms, one can generate an energy profile for a magnet as a function of the angle of magnetization (as in Fig. 2.1d). The local energy minimum closest to the initial angle of magnetization (i.e., 0°) corresponds to the angle at which the magnet would finally settle. An example phase diagram generated through this approach is shown in Fig. 3.1b. Diagrams generated via this approach do not capture the simulation behavior as accurately as the diagrams generated via the previous approach (Fig. 3.1a). However, as the three regions of interests (i.e., ↑, ↓ and →) are in good agreement between the two versions of the phase diagrams, one can use either of them to design a functional layout. Moreover, the analytical approach generates phase diagrams significantly faster than the simulation approach.

3.3.2 Conditions for Hard Axis Stability

For a magnet ensemble to evolve to a logically correct ground state, a given device must be able to: (i) remain in a metastable state such that it does not switch prematurely, and (ii) deterministically switch to a logically correct, easy axis magnetization
state based on fringing fields from an appropriate neighbor. I first define the criteria for (i), and discuss those for (ii) in the next subsection.

Let the input space \( I = \{I_0, I_1, \ldots, I_{2^n-1}\} \) represent all \( 2^n \) possible input combinations for an \( n \)-input gate or circuit structure. For a given magnet \( M \) and given input \( I_j \), let \( t_{sw}^M \) be the time when \( M \) should begin switching to a logically correct state such that correct ensemble switching order is preserved. Also, let \( H_x(I_j, t) \) and \( H_y(I_j, t) \) be the x- and y-components of the total magnetic field experienced by \( M \) for input \( I_j \) at an arbitrary time \( t < t_{sw}^M \).

Given the values of \( H_x(I_j, t) \) and \( H_y(I_j, t) \), one must determine whether or not a magnet will remain metastable. The phase diagram is employed to find the minimum value of \( H_x \) that is required to keep the magnet metastable. For example, per point Q in Fig. 3.1a, if \( H_x(I_j, t) = 15 \) mT and \( H_y(I_j, t) = 2.5 \) mT then the x-directed field should be at least 45 mT (per point P) to ensure that the magnet remains in the metastable region of the phase diagram. This minimum required x-directed field is referred to as \( H_{min}(I_j, t) \). If the existing field \( H_x(I_j, t) \) does not put the magnet in a metastable state (point Q), some additional x-directed field (from the clock and/or a helper cell) must be provided. This minimum required additional x-directed field can be expressed as:

\[
H_{minExtra}(I_j, t) = \begin{cases} 
0 & \text{if } H_x(I_j, t) \geq H_{min}(I_j, t) \\
H_{min}(I_j, t) - H_x(I_j, t) & \text{otherwise}
\end{cases}
\] (3.2)

**Definition 1.** \( H_{minExtra} \): The minimum required additional x-directed field to keep a given magnet metastable for all input combinations and for any \( t < t_{sw}^M \). It can be expressed as:

\[
H_{minExtra} = \max(\{H_{minExtra}(I_j, t)|I_j \in I, t < t_{sw}^M\})
\] (3.3)
Thus, the necessary and sufficient condition to keep a magnet in a metastable state such that it does not switch prematurely (implying that a local bias $H_x$ is sufficient) is:

$$H_{\text{minExtra}} = 0 \quad (3.4)$$

### 3.3.3 Conditions for Correct Switching

Here, I introduce criteria for a magnet to successfully switch to a logically correct, easy axis magnetization state when set by an appropriate neighbor (i.e., at time $t_{sw}^M$). Once a magnet has switched in a properly designed NML circuit, it should remain in a logic 1/0 state until re-evaluated with another input.

From the phase diagram one can determine the maximum tolerable value of $H_x$ such that a magnet remains in a region corresponding to a $\uparrow$ or $\downarrow$ state. For example, per point G in Fig. 3.1a, if $H_x(I_j, t_{sw}^M) = 20 \text{ mT}$ and $H_y(I_j, t_{sw}^M) = -10 \text{ mT}$, $H_x$ must not exceed 50 mT (indicated by point F). This upper limit of $H_x$ is denoted as $H_{\text{max}}(I_j, t_{sw}^M)$. For successful switching, the maximum tolerable additional x-directed field for input $I_j$ can be expressed as:

$$H_{\text{maxExtra}}(I_j) = \begin{cases} 
0 & \text{if } H_x(I_j, t_{sw}^M) \geq H_{\text{max}}(I_j, t_{sw}^M) \\
H_{\text{max}}(I_j, t_{sw}^M) - H_x(I_j, t_{sw}^M) & \text{otherwise}
\end{cases}$$

**Definition 2.** $H_{\text{maxExtra}}$: The maximum tolerable additional x-directed field so that a magnet can successfully switch for all input combinations. It can be expressed as:

$$H_{\text{maxExtra}} = \min(\{H_{\text{maxExtra}}(I_j) | I_j \in I\})$$

Thus, the necessary condition for a magnet to switch to a logically correct easy axis
state is:

\[ H_{\text{maxExtra}} > 0 \]  \hspace{1cm} (3.5)

This implies that for all input combinations, \( H_x < H_{\text{max}} \). If for any input, \( H_x \geq H_{\text{max}} \) – i.e., \( H_{\text{maxExtra}} = 0 \), a magnet will not be able to transition to the corresponding easy axis. The above analysis results in the following design rule:\footnote{The term “design rules” is used to signify conditions that a design must not violate.}

**Rule 1.** If for any magnet \( H_{\text{maxExtra}} = 0 \), a design is not feasible.

The conditions for hard axis stability (Eqn. 3.4), and for correct switching (Eqn. 3.5) can be used to categorize the magnets in an ensemble into two groups per the following definitions:

**Definition 3. Non-critical magnet:** A magnet is ‘non-critical’ if, given the existing \( H_x \) (from the ensemble and/or clock field), (i) it is capable of remaining in a metastable state such that it does not switch prematurely (i.e., \( H_{\text{minExtra}} = 0 \)), and (ii) it can switch to a logically correct easy axis when needed (i.e., \( H_{\text{maxExtra}} > 0 \)). That is, a magnet is ‘non-critical’ if:

\[ (H_{\text{minExtra}} = 0) \land (H_{\text{maxExtra}} > 0) \]  \hspace{1cm} (3.6)

**Definition 4. Critical magnet:** A magnet is ‘critical’ if, given the existing \( H_x \) (from the ensemble and/or clock field), (i) it requires some additional \( H_x \) (i.e., from the clock and/or a static helper) to remain in a metastable state (i.e., \( H_{\text{minExtra}} > 0 \)), and (ii) it can switch to a logically correct state (i.e., \( H_{\text{maxExtra}} > 0 \)). That is, a magnet is ‘critical’ if:

\[ (H_{\text{minExtra}} > 0) \land (H_{\text{maxExtra}} > 0) \]  \hspace{1cm} (3.7)

The term “design rules” is used to signify conditions that a design must not violate.
Each magnet in an ensemble should remain metastable until set by an appropriate neighbor. Thus, all magnets must satisfy the condition for non-criticality (Eqn. 3.6).

3.3.4 Overall Approach

The central idea of the design methodology is to make each critical magnet non-critical by setting appropriate values for device-to-device spacing, and by providing x-directed fields from helper cells and/or the clock.

A complete design flow is captured by Fig. 3.2. The design process takes as input an initial, conceptual circuit layout, and the desired time evolution of the magnets. An example initial layout for an AND gate is shown in Fig. 3.3a. The slanted magnet edge induces an energy barrier shift that facilitates a 2-input AND [69]. The initial layout also includes the clock boundaries. For example, the AND layout is divided into two separate clocking zones (as shown by the overlaid line) to avoid a potential race condition at magnet C. The initial layout functionally resembles stick diagrams for a CMOS layout. The desired switching order of the magnets during logic evaluation for input ab =↓↓, is shown in Fig. 3.3b-3.3j. At t1 the inputs (magnets a1 and b1) are set. A clock field acts on zone 2 (see Section 2.1.2) until time-step t3 and is removed at the beginning of t4. The appropriate switching time, \( t_{sw}^M \) for a given magnet M is assigned to one of the time-steps \( t_0 \) to \( t_8 \) during which the magnet is expected to start relaxing. For example, \( t_{sw}^C = t_4 \) and \( t_{sw}^{a3} = t_{sw}^{b3} = t_2 \).

The design approach first finds permissible pairs of horizontal and vertical spacings (\( d_x \) and \( d_y \), respectively) between devices that allow the interior magnets (IM) in all horizontal lines to become non-critical (steps 1a, 1b in Fig. 3.2). The process then iteratively tests the permissible \( d_x, d_y \) pairs (step 2) to verify if the rest of the magnets could be made non-critical via: (i) a switchability test which checks if too much hard axis stability would inhibit switching (steps 3a, 3b), and by (ii) determining the clock field and/or helper cell design and placement to preserve metastability.
prior to switching (steps 4, 5). Once a suitable \( d_x, d_y \) pair is found, the methodology modifies the initial layout by accommodating the newly identified helper cells (step 6). The modified layout is again tested to verify that: (a) all its IMs are non-critical (steps 7a, 7b), and (b) the rest of the magnets are non-critical (step 8). The design process completes after determining device-to-device spacings, helper cells’ size and positions, and a common clock field magnitude.

3.4 Detailed Design Approach

I now describe the key steps of the design methodology. The design of an AND gate is used as a running example.

3.4.1 Making Interior Magnets Non-critical

Here, I discuss how to achieve non-criticality for the interior magnets (IM) in any horizontal, anti-ferromagnetically ordered line (steps 1a, 1b in Fig. 3.2). (Note that interior magnets in vertical lines can be made non-critical by using helpers and/or clock field which will be discussed in Section 3.4.3.) One can make a critical magnet non-critical by: (i) setting appropriate values for device-to-device spacings, and/or (ii) providing appropriate \( x \)-directed fields from helper cells and/or the clock field if necessary. However, helper cells cannot be used to preserve metastability for IMs (e.g., \( a_2, b_2, o_1, o_2 \) in Fig. 3.3a) in an anti-ferromagnetically ordered line. Additionally, any applied clocking field on the said device might be removed well before it should switch. Therefore, assuming that material and crystalline structure do not change, the only way to make IMs non-critical is to augment device-to-device spacing.

Procedure 1 illustrates how to find permissible \( d_x \) and \( d_y \) pairs such that each IM satisfies the non-criticality condition. Here, \( D_x \) and \( D_y \) are the set of values of \( d_x \) and \( d_y \) to be tested. The cartesian product \( D_x \times D_y \) contains all possible pairs of \( d_x \) and \( d_y \). For a given \((d_x, d_y)\), the phase diagram is used to calculate \( H_{minExtra} \).
for all IMs (line 2 of Procedure 1) by utilizing Eqn. 3.2 and Eqn. 3.3. For a given magnet \( M \), Eqn. 3.2 needs: (i) the minimum required x-directed field to keep \( M \) in a metastable state which is readily found from the phase diagram, and (ii) the existing x-directed field over \( M \) exerted by the rest of the magnets in the ensemble. Note that this can be obtained with a micro-magnetic simulator. With Eqn. 3.3 all the time-steps that precede \( t_{sw}^M \) are considered. Similarly, the values of \( H_{maxExtra} \) are calculated for all IMs. For the given \( (d_x, d_y) \), if all the IMs satisfy the non-criticality condition (Eqn. 3.6) the pair is included in the set of permissible pairs.

For the running example of the AND gate design in Fig. 3.3a, \( D_x \) and \( D_y \) range...
Figure 3.3. (a) Initial layout for an AND gate. (b)-(j) the desired switching order of the magnets during logic evaluation for input $ab = \downarrow\downarrow$. 
from 6 \text{nm} to 15 \text{nm} and 15 \text{nm} to 36 \text{nm} respectively. (Only intervals of 3 \text{nm} – the smallest multiple of the simulation mesh – are considered.) Thus, the set of permissible pairs are found to include all \((d_x, d_y)\) with \(d_x: 6\) to \(9 \text{nm}\) and \(d_y: 15\) to \(36 \text{nm}\). Note that, a designer can further filter the permissible set with fabrication constraints.

### 3.4.2 Switchability Test

Now, I introduce the \textit{switchability test} (Procedure 2) to determine – for a given \((d_x, d_y)\) – whether magnets that are not IM can switch to a logically correct, easy axis state (steps 3a, 3b in Fig. 3.2). Any permissible \((d_x, d_y)\) pair already guarantees the \textit{non-criticality} of IMs. The design process iteratively invokes this test to find a permissible \((d_x, d_y)\) pair that allows the non-IMs to satisfy the second requirement for \textit{non-criticality} – being able to switch to an easy axis.

For a given \((d_x, d_y)\), for each of the non-IMs, the values of \(H_{\text{minExtra}}\) and \(H_{\text{maxExtra}}\) are again calculated. If a magnet is found to be \textit{non-critical} (line 3, Proc. 2), the test simply continues to the next non-IM. However, if \(H_{\text{maxExtra}}\) is equal to zero for a magnet (line 4, Proc. 2), i.e., if it cannot switch, the design becomes infeasible according to Rule 1 (see Sec. 3.3.3). If a magnet is not \textit{non-critical}, and does not fail to switch according to Rule 1, it is \textit{critical}.

I now consider \textit{critical} magnets in more detail. A \textit{critical} magnet \(M\) is defined as \textit{TypeA}, if time-step \(t_{sw}^M\), when \(M\) begins to switch, is not the same as time-step, \(t_c\), when the clock field is removed, i.e., \(t_{sw}^M \neq t_c\). In contrast, if \(t_{sw}^M = t_c\), i.e., switching begins as a direct consequence of clock field removal, a magnet is \textit{TypeB}. In the AND gate initial layout (Fig. 3.3a), if initially \textit{critical}, magnet \(C\) would be a \textit{TypeB} magnet, and magnets \(a_3, b_3\) and \(o_3\) would be \textit{TypeA}.

For a \textit{TypeA} \textit{critical} magnet, \(H_{\text{minExtra}}\) must be less than \(H_{\text{maxExtra}}\) to enable successful switching. As an example, assume a given magnet \(M\) is \textit{TypeA critical},
Procedure 1 Find Permissible Pairs of \((d_x, d_y)\)

1: for each pair \((d_x, d_y) \in D_x \times D_y\) do
2: Calculate \(H_{\text{minExtra}}\) and \(H_{\text{maxExtra}}\) for each IM using the phase diagram.
3: if all the IMs satisfy non-criticality condition then
4: Include \((d_x, d_y)\) within the set of permissible pairs.

Procedure 2 Switchability Test for a permissible \((d_x, d_y)\)

1: for each magnet \(M\) among the magnets that are not IM do
2: Calculate \(H_{\text{minExtra}}\) and \(H_{\text{maxExtra}}\) for \(M\) using the phase diagram.
3: if \((H_{\text{minExtra}} = 0) \land (H_{\text{maxExtra}} > 0)\) then
   \(M\) is non-critical, Continue.
4: else if \(H_{\text{maxExtra}} = 0\) then return False.
5: else if \((H_{\text{minExtra}} > H_{\text{maxExtra}}) \land M \text{ is TypeA}\) then
   return False.
6: return True.

for which \(H_{\text{minExtra}}\) is 20 mT and \(H_{\text{maxExtra}}\) is 15 mT. One need to provide at least 20 mT additional x-directed field on this magnet to keep it in a metastable state before \(t_{sw}^M\). But at \(t_{sw}^M\), the upper bound for tolerable additional x-directed field is 15 mT. As such, the magnet will never be able to switch. For TypeB critical magnets, if \(H_{\text{minExtra}} > H_{\text{maxExtra}}\), the applied additional x-directed field will be reduced at \(t_{sw}^M\) as the clock field is removed, thus the same problem does not occur.

The following design rule results:

Rule 2. A design is not feasible if for any TypeA critical magnet, \(H_{\text{minExtra}} > H_{\text{maxExtra}}\).

For the running example, permissible \((d_x, d_y)\) pairs are iteratively tested for switchability and finally \((9 \text{ nm, } 21 \text{ nm})\) and \((6 \text{ nm, } 21 \text{ nm})\) are identified as acceptable pairs.

3.4.3 Determination of Clock Field

I now discuss steps 4, 5 (Fig. 3.2) of the design methodology. Since a \((d_x, d_y)\) pair that passes the switchability test results in a layout where all magnets are either
non-critical or critical, the focus now is to make all critical magnets non-critical (i.e., ensure that critical magnets remain metastable and do not switch prematurely). One can achieve this by leveraging the clock field and/or helper cells to provide additional x-directed fields on the critical magnets.

Here I focus on finding the clock field (Proc. 3). The clock field only effects magnets that reside next to the clock boundary and switch at or before \( t_c \) (e.g., \( a_3, b_3, C \)). For each such magnet, the minimum required \( (C^M_{\text{min}}) \) and the maximum tolerable \( (C^M_{\text{min}}) \) clock fields (line 2-5) are determined. For magnets that switch before \( t_c \) (e.g., \( a_3, b_3 \)), \( C^M_{\text{min}} = 0 \) (required additional field coming from helper), and \( C^M_{\text{max}} = H_{\text{maxExtra}} \) (when no helper is used). A magnet switching at \( t_c \) experiences the clock field only before switching. Thus, it can tolerate an infinite clock field (line 3). At \( t_c \), this magnet experiences field contribution only from a helper, and this field can at most be \( H_{\text{maxExtra}} \) (otherwise, the magnet will not switch). Therefore, before switching, the magnet must see a clock field of at least \( H_{\text{minExtra}} - H_{\text{maxExtra}} \) (line 4) such that the total field for metastability becomes \( H_{\text{minExtra}} \). Note that, if \( H_{\text{minExtra}} \leq H_{\text{maxExtra}} \), the helper alone (contributing \( H_{\text{maxExtra}} \)) can satisfy metastability, hence \( C^M_{\text{min}} = 0 \) (line 5). A range for the clock field is determined in line 6-7. A value within this range is chosen as a clock field (line 8). If the clock field is not enough for metastability, additional field is provided via a helper cell.

For the AND gate with \((d_x, d_y) = (9 \text{ nm}, 21 \text{ nm})\), the corresponding values for the critical magnets \( a_3, b_3, o_3 \) and \( C \) are shown in Table 3.1. The clock range is found to be 29 to 37 mT. From this range the clock field is chosen as 35 mT. With this clock field, no helper is required for \( a_3 \) and \( b_3 \). For magnet \( C \), at least 6 mT must be provided via a helper cell, and this helper contribution cannot exceed 12 mT. Using OOMMF, a \( 87 \times 39 \times 5 \text{ nm}^3 \) helper placed 18 nm away from \( C \) is found to satisfy the above requirement. The helper cell for magnet \( o_3 \) must provide a field between
Procedure 3 Find Clock field

1: for each critical magnet $M$ next to the clock boundary with $t_{sw}^M \leq t_c$ do
2: if $t_{sw}^M < t_c$ then $C_{max}^M = H_{maxExtra}$, $C_{min}^M = 0$
3: else if $t_{sw}^M = t_c$ then $C_{max}^M = \infty$
4: if $H_{minExtra} > H_{maxExtra}$ then $C_{min}^M = H_{minExtra} - H_{maxExtra}$
5: else $C_{min}^M = 0$
6: $C_{max}^M$ = minimum of the $C_{max}^M$ values.
7: $C_{min}^M$ = maximum of the $C_{min}^M$ values.
8: if $C_{max}^M \geq C_{min}^M$ then
    choose any value between $C_{min}$ and $C_{max}$ as clock.
9: else clock not found, return False.

Figure 3.4. The final AND layout with $d_x = 9 \, nm$ and $d_y = 21 \, nm$. Both the helpers are $87 \times 39 \times 5 \, nm^3$ in dimension. Helper1 is 18 nm left from C and Helper2 is 6 nm right from $o_3$.

20 mT and 35 mT. A $87 \times 39 \times 5 \, nm^3$ helper cell placed 6 nm away from $o_3$ provides a 23 mT field. (This helper cell mimics an adjacent clock group, and would be omitted in a larger design.)

The initial layout is modified by accommodating the helper cells (step 6 in Fig. 3.2). These helper cells will exert magnetic fields on all devices in the circuit, and thus can alter their characteristics. Therefore, for the modified layout, further testing as shown in Fig. 3.2 is then conducted, and the clock field is again determined following the aforementioned procedure. With the running example, the finalized clock field is 30 mT. To verify the final layout (Fig. 3.4), it is simulated in OOMMF at 0K. The time evolution of the design was correct for all inputs.
### Table 3.1

VALUES FOR THE CRITICAL MAGNETS

<table>
<thead>
<tr>
<th>Magnets:</th>
<th>( a_3, b_3 )</th>
<th>( C )</th>
<th>( o_3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( H_{\text{minExtra}} ) (mT)</td>
<td>35</td>
<td>41</td>
<td>20</td>
</tr>
<tr>
<td>( H_{\text{maxExtra}} ) (mT)</td>
<td>37</td>
<td>12</td>
<td>35</td>
</tr>
<tr>
<td>( C_{\text{max}}^M ) (mT)</td>
<td>37</td>
<td>( \infty )</td>
<td>-</td>
</tr>
<tr>
<td>( C_{\text{min}}^M ) (mT)</td>
<td>0</td>
<td>29</td>
<td>-</td>
</tr>
</tbody>
</table>

### 3.5 Case Studies

I have also applied the methodology to design a corner turn, a 3-input majority gate, and a cascaded AND-OR logic structure. In a corner turn, a signal transitions from an anti-ferromagnetic line to a ferromagnetic line, and thus it functions as interconnect. The final layout for a corner turn is shown in Fig. 3.5a where, \( d_x = 9 \text{ nm} \) and \( d_y = 24 \text{ nm} \). All the helper cells are \( 9 \text{ nm} \) away from the magnets. The helpers of magnet \( a_5 \) are \( 39 \times 39 \times 5 \text{ nm}^3 \), and those for magnet \( a_6 \) and \( a_7 \) are \( 87 \times 45 \times 5 \text{ nm}^3 \) in dimension. While zone 2 is clocked by a \( 23 \text{ mT} \) field, input propagates correctly up to magnet \( a_4 \). After the clock is removed, magnet \( a_5, a_6 \) and \( a_7 \) switch in order.

The final layout for a majority gate (Fig. 3.5b) requires \( d_x = 9 \text{ nm}, d_y = 15 \text{ nm} \) and a clock field of \( 50 \text{ mT} \). Three \( 87 \times 39 \times 5 \text{ nm}^3 \) helper cells are placed \( 9 \text{ nm} \) right from magnet \( a_3, o_3, c_3 \) and two \( 39 \times 39 \times 5 \text{ nm}^3 \) helper cells are placed \( 9 \text{ nm} \) away from magnet \( a_4 \) and \( c_4 \) on both sides. While zone 2 is clocked, inputs propagate correctly up to magnet \( a_3, b_2, \) and \( c_3 \). After the clock is removed: i) magnet \( a_4 \) and \( c_4 \) switch first, ii) then the compute magnet \( C \) switches, and iii) lastly magnet \( o_1, o_2, \) \( o_3 \) switch in order.

The final layout for a cascaded AND-OR logic (an AND gate followed by an OR
Figure 3.5. Final layout for (a) a corner turn, (b) a majority gate, and (c) a cascaded AND-OR logic.

gate) is shown in Fig. 3.5c. To achieve the OR functionality, a slanted edge at the top left corner of the compute magnet [69] is used. Here, $d_x = 9\text{ nm}$ and $d_y = 21\text{ nm}$. All the helper cells are $87 \times 39 \times 5\text{ nm}^3$ in dimension. Helper1 and Helper2 are $18\text{ nm}$ left from magnet $c_1$ and $c_2$ respectively, and Helper3 is $6\text{ nm}$ right from magnet $o_8$. While zone 2 and 3 are clocked by a $32\text{ mT}$ field, input ‘a’ and ‘b’ propagates correctly up to magnet $a_3$ and $b_3$. After the clock is removed from zone 2: (i) the AND compute magnet (c1) evaluates correctly and the signal propagates up to magnet $o_5$, (ii) input ‘d’ propagates up to magnet $d_3$. When the clock is removed from zone 3, the OR
compute magnet \( c_2 \) evaluates and magnet \( o_6, o_7 \) and \( o_8 \) switch in order. All these three layouts evolve correctly in time for all input combinations when simulated at 0K.

3.6 Thermal Noise Consideration

Thermal noise can have a significant impact on the behavior of NML circuits and could lead to unwanted magnetization states. This is due to the fact that thermal excitation essentially introduces a random magnetic field. Specifically, thermal noise in a micro-magnetic model can be expressed as an equivalent stochastic magnetic field \([14, 55]\) where \( g(t) \) is a Gaussian random distribution with unit variance and zero mean, \( V \) is the volume of the nanomagnet, and \( \Delta t \) is the discretization in time:

\[
H_{\text{thermal}} = \frac{1}{\sqrt{V \Delta t}} \sqrt{\frac{2kT \alpha}{\mu_0 \gamma M_s}} g(t) \tag{3.8}
\]

By setting \( \Delta t \) to the relaxation period of the clocking zone (here, it is 1 ns), one can get a representative value for thermal noise. At 300K, for 1 million Gaussian-distributed random numbers, \( H_{\text{thermal}} \) varies from \(-2.4 \text{ mT}\) to \(+2.4 \text{ mT}\). In the presence of \( H_{\text{thermal}} \), a layout designed for 0K might not follow the desired switching order (e.g., Fig. 3.3b-3.3j), and thus could fail to evolve to a logically correct state.

Thermal simulations of the 0K design layouts demonstrates three types of problem that an individual magnet in the layout can suffer from:

1. **Premature Switching**: A given magnet switches earlier than its preceding neighbor. The magnet is expected to operate in the ‘Metastable’ (→) region of the phase diagram as long as the preceding magnet does not switch. However, if the operating point resides very close to the border (e.g., point R in Fig. 3.1a) thermal noise may easily shift the operating point out of the metastable region, and thus trigger premature switching. For example, with a y-directed \( H_{\text{thermal}} \) of \(-2.4 \text{ mT}\), R would shift to the bottom end of the overlaid vertical line shown in Fig. 3.1a. To combat this problem, point R is moved further to the right into the metastable region such that the point can have more margin for thermal shift and...
reside within the metastable region. This can be achieved by providing additional x-directed field on the vulnerable magnet (from helper magnets, clock fields, or by reducing inter-magnet spacings, etc.). Premature switching can also be addressed by eliminating any unwanted y-directed magnetic field.

2. Over-stability: This is the opposite case of the aforementioned problem where a given magnet does not switch even after its preceding neighbor has completed switching. The magnet is expected to operate either in the ‘Up’ (↑) or ‘Down’ (↓) region of the phase diagram. If the operating point resides very close to the border, thermal noise can shift the operating point in to the metastable region. Thus the magnet becomes over-stable and cannot switch. To combat this problem, the operating point is moved further to the left into the corresponding easy-axis region – allowing more margin for thermal shift – by reducing the x-directed field on the given magnet (via smaller helper magnets, reduced clock fields, or wider
3. **Race Condition**: A given magnet experiences a race condition when magnets that provide its inputs switch at different speeds (due to disparate thermal noises acting on them), and thus potentially switches to a logically incorrect state. This problem can be solved by separating the given magnet from its input magnets into a different clocking zone.

During simulation of the initial corner turn design (Fig. 3.5a) at 300K, magnet \(a_3\) switches prematurely to a wrong easy axis state for input \(a = \uparrow\). This is solved by reducing the distances between magnet \(a_3\) and \(a_2\), \(a_3\) and \(a_4\) to 6 nm (from 9 nm in the 0K design).

For AND gate (Fig. 3.4) simulation at 300K, for input \(ab = \uparrow\downarrow\), magnet \(C\) does not switch to a \(\downarrow\) state within 1 ns, and remain metastable. When a longer relaxation period was allowed, magnet \(C\) sometimes required 4 ns to switch. This would obviously have a negative impact on clock period. To make the switching faster, the spacing between magnet \(C\) and \(Helper_1\) is increased to 30 nm and clock field is increased to 34 mT. 100 simulations at 300K evolved correctly for all input combinations where magnet \(C\) switched within 1 ns.

Thermal simulations of the initial majority gate design (Fig. 3.5b) at 300K illustrated three problems: (i) magnet \(a_2\) and \(c_2\) switch prematurely, (ii) the state of magnet \(C\) is dominated by \(b_2\) if \(a_4\) and \(c_4\) do not switch to their corresponding easy axis states sufficiently fast, and thus creates a race condition, and (iii) magnet \(o_1\) switches prematurely. The layout for 300K (Fig. 3.6a) solves case (i) by bringing the neighbors of \(a_2\) and \(c_2\) closer. The distances between \(a_1\) and \(a_2\), \(c_1\) and \(c_2\) in the modified layout are 6 nm. For case (ii) the race condition on \(C\) is eliminated by separating it into a different clocking zone such that it can respond only after all of its adjacent previous neighbors are set. The more complex clocking zones can be supported by multiferroic voltage controlled clocking \[34\]. Clocking fields for zone 2, 3, and 4 are 40, 50, and 30 mT, respectively. Clock field is first removed from zone
2, then from zone 3 and lastly from zone 4. To solve case (iii) of magnet \( o_1 \), both clocking zone separation and x-directed field increment (by reducing inter-magnet spacing) were required. The modified distances between magnet \( C \) and \( a_4 \), \( C \) and \( c_4 \) are 21 \( \text{nm} \) and distance between magnet \( C \) and \( o_1 \) is 6 \( \text{nm} \).

The initial cascaded AND-OR logic layout (Fig. 3.5c) suffers from two problems when simulated at 300\( K \): (i) magnet \( c_1 \) and \( c_2 \) are over-stable, and (ii) magnet \( o_2 \) switches prematurely. For the first case, the distance between magnet \( c_1 \) and \( Helper 1 \), \( c_2 \) and \( Helper 2 \) are changed to 24 \( \text{nm} \) and 30 \( \text{nm} \) respectively, and the clock field to 34 \( \text{mT} \) (see Fig. 3.6b). The second problem arises from the fact that \( Helper 2 \) provides a constant negative \( H_y \) field on magnet \( o_2 \), which combined with the thermal noise might trigger premature switching. To combat this problem, a dummy helper cell \( (Helper 4) \) is placed such that field contributions from \( Helper 2 \) and \( Helper 4 \) cancel each other. The updated layout is shown in Fig. 3.6b. All these design solutions evolved correctly when simulated (with 100 different random seeds) at 300\( K \) for all input combinations.

3.7 Conclusion

In this chapter, I show that it is possible to generate predictable design solutions for NML circuits in a systematic manner. The proposed methodology can be implemented as a CAD tool to facilitate faster layout identification. The approach is applied to find functional layouts for a 2-input AND gate, corner turn, and majority gate – which eventually could be used to construct any boolean circuit.
4.1 Overview

In this chapter, I explore the possibilities of utilizing the Tunneling Field Effect Transistor (TFET) – a low voltage, steep subthreshold device – in the analog computing paradigm of Cellular Nonlinear/Neural Network (CNN). TFETs are three terminal devices capable of exhibiting steep subthreshold swing and can operate at very low voltage. Examples of experimentally demonstrated TFETs exist (see [88] for a more detailed description). The low voltage operability of TFET shows strong promises for ultra-low power application spaces. For example, in [48] TFETs have shown to be extremely power efficient for logic circuits. Furthermore, the mechanism for conducting current in TFET is band-to-band tunneling which has very weak sensitivity to temperature variation [79]. As a result, TFET operations are robust in the presence of thermal noise – an important design consideration for analog circuits.

My study reveals that TFETs can be successfully employed in a CNN system to solve various image processing problems. Moreover, due to its non-linear characteristics, TFET obviates the necessity of the output transfer function – common hardware in each CNN cell for saturating the output voltage. This will reduce the hardware footprint for each cell in a CNN array. I have also augmented a MATLAB based simulator [27] to (i) capture the dynamics of the TFET-based CNN, and (ii) verify template functionality for a variety of applications.

Resonant Tunneling Diodes (RTD) is another device that has been utilized in CNN [39, 44, 65], although device realizations are challenging in practice [15]. However,
none of these works has studied the power dissipated by the cells. In this work, I calculate the power dissipation of TFET-based cell and illustrate that it would always outperform the power dissipated by a conventional cell. Results also suggest by carefully engineering TFET devices further power savings are possible without sacrificing performance.

4.2 The Cell Circuit

The TFET-based CNN cell proposed in this chapter is illustrated in Fig. 4.2. In contrast to a conventional CNN cell (Fig. 2.3b) this circuit replaces the linear resistor with a non-linear TFET device. The drain and source of the TFET connect to the state node $x$ and ground, respectively, and the gate is tied to a fixed voltage (e.g., 0.3 V). The TFET introduces non-linearity to the state node’s transient behavior. As such, the non-linear output transfer function hardware is no longer required. The non-linearity of the TFET itself would ensure saturating state voltages. To compensate for device mismatch issues associated with current CMOS scaling, typical implementation of the output transfer function would require a large amount of area. As such, by employing TFETs in the CNN cell design, it is possible to reduce the area of each cell. In the TFET-based cell, the state voltage and output becomes equivalent of each other, hence the state equilibrium of the TFET-based CNN cell becomes:

$$C \frac{dx_{ij}(t)}{dt} = -h(x_{ij}(t)) + \sum_{C_{kl} \in N_r(i,j)} a_{ij,kl}x_{kl}(t) + \sum_{C_{kl} \in N_r(i,j)} b_{ij,kl}u_{kl} + Z \quad (4.1)$$

where $h(x_{ij})$ represents the current through the TFET as a function of the state voltage. Since the state node is connected to the drain, $h(x_{ij})$ directly corresponds to TFET’s I-V characteristics.

Note that, for a conventional CNN, a number of different values of $I_{eff}$ correspond-
Figure 4.1. TFET I-V characteristics used as the basis for this work.

Figure 4.2. TFET-based CNN cell circuit.

ing to a specific output (e.g., $-1\, V$ or $1\, V$) might result in a number of different sparse stable states. As long as the absolute values of these stable states are greater than 1, the sigmoid function would result in the output voltage saturating at either $-1\, V$ or $1\, V$. However, in the case of a TFET-based CNN cell, the saturating effect of the output transfer function has to be compensated for by ensuring that different values of $I_{eff}$ corresponding to a specific output would result in stable voltages within
a close range. For example, in Fig. 4.1 the diode-like exponential current growth of TFET near \(-0.8\) V suggests that the device is capable of conducting a wide range of different negative currents while operating at a voltage close to \(-0.8\) V. Therefore, for the TFET-based cell, \(-0.8\) V is chosen to represent a low signal. In the absence of any exponential current growth in the positive voltage side of the asymmetric TFET (assuming the device characteristics as illustrated in Fig. 4.1) I pick a voltage level for a high signal from the subthreshold growth region, e.g., \(0.1\) V.

### 4.3 Operation Principles

In this section I explain the operation principle of a TFET-based CNN cell through the concept of driving point (DP) plots [19]. To this end, the TFET cell equation is rewritten as:

\[
C \frac{dx_{ij}(t)}{dt} = -h(x_{ij}(t)) + a_{ij,ij}x_{ij}(t) + \sum_{c_{kl} \in N_r(i,j) \land kl \neq ij} a_{ij,kl}x_{kl}(t) + \sum_{c_{kl} \in N_r(i,j)} b_{ij,kl}u_{kl} + Z
\]  (4.2)

The summation of the first two right hand side terms of Eq. 4.2 is called the DP component, and denoted by \(g(x_{ij})\). The last three terms’ summation is called the offset, which is denoted as \(w\). In a DP plot, Eq. 4.2 is plotted as a function of the state voltage, \(x_{ij}\). The shape of the plot can be modulated by \(g(x_{ij})\), whereas \(w\) provides an offset to shift the plot vertically. As such, the equilibrium states of a cell can be identified at points where the plot intersects with the x-axis. Physically it means that at equilibrium, the current through the capacitor, \(C \frac{dx}{dt}\) is zero – i.e., the capacitor has completely charged or discharged to the equilibrium voltage.

As noted earlier, it is well established that CNN architectures are suited for image processing. Below, I use the Horizontal Line Detection (HLD) problem as an example to further illustrate TFET-based CNN functionality. This problem detects lines
consisting of at least two consecutive horizontal black pixels (see Fig. 4.3). Each cell of the network processes a single pixel of the input image. The input nodes of the cells are provided with the corresponding input image pixel (−0.8 V or 0.1 V for a white or black pixel, respectively). The cell states can be initialized with any random values. For the TFET-based HLD, the template values are:

\[
A = \begin{bmatrix}
0 & 0 & 0 \\
0 & 0 & 0 \\
0 & 0 & 0
\end{bmatrix},
B = k \begin{bmatrix}
0 & 0 & 0 \\
1 & 7 & 1 \\
0 & 0 & 0
\end{bmatrix}, Z = 0
\]

As the feedback template \(A = 0\), the DP component, \(g(x)\) becomes the negated version of the TFET I-V. Table 4.1 reports the offset, \(w\) values for different pixel configurations for left, central, and right cells, respectively (‘b’ denotes a black pixel, ‘w’ denotes a white pixel). From Fig. 4.4 one can see that a given cell’s equilibrium state is approximately −0.8 V if output state should be white, and 0.1 V if output state should be black. Thus, with proper template design TFET-based cells can reach desired output states without any non-linear transfer function hardware.

4.4 Simulation Results

I have developed\(^1\) a MATLAB tool to simulate the TFET-based CNN array dynamics for image processing tasks. The tool solves a system of ODEs numerically by using the Runge-Kutta method. The system of ODEs consists of \(MN\) equations each defined by Eq. 4.1 for each cell of an \(M \times N\) TFET-based CNN array. When considering cell values for \(h(x_{ij})\), I employ a TFET I-V lookup table (the device characteristics illustrated in Fig. 4.1 for \(V_{gs} = 0.3\) V \(^{52}\)) with interpolation capability.

\(^1\)The author acknowledges András Horváth from Pázmány Péter Catholic University, Hungary for helping develop the MATLAB simulator for TFET-based CNN.
Figure 4.3. Horizontal Line Detection with TFET-CNN: (a) input image; (b) output image.

Figure 4.4. DP plots for HLD. Here $k = 40 \times 10^{-6}$ such that the state equilibrium points (marked with open circles) are generated approximately at $-0.8 \, V$ (for a white output), and $0.1 \, V$ (for a black output).
TABLE 4.1
OFFSET VALUES FOR HLD

<table>
<thead>
<tr>
<th>Pixels</th>
<th>Target State</th>
<th>(w)</th>
</tr>
</thead>
<tbody>
<tr>
<td>(bbb)</td>
<td>(b)</td>
<td>0.9k</td>
</tr>
<tr>
<td>(bbw, wbb)</td>
<td>(b)</td>
<td>0</td>
</tr>
<tr>
<td>(wbw)</td>
<td>(w)</td>
<td>−0.9k</td>
</tr>
<tr>
<td>(bwb)</td>
<td>(w)</td>
<td>−5.4k</td>
</tr>
<tr>
<td>(bww, wwb)</td>
<td>(w)</td>
<td>−6.3k</td>
</tr>
<tr>
<td>(www)</td>
<td>(w)</td>
<td>−7.2k</td>
</tr>
</tbody>
</table>

Each cell in the array requires an initial value for the state voltage which defines the initial condition for the cell’s equation. The specific image processing problem to be solved is defined by the \(A, B\) templates and by the fixed bias \(Z\). Fig. 4.5 illustrates the simulation results for four image processing tasks: Horizontal Line Detection, Logical AND, Left Shadowing, and the Labyrinth problem.

4.5 CNN Power Dissipation

In this section I calculate the power dissipated by a CNN cell, and perform a qualitative analysis to illustrate the superiority of a TFET-based cell over a conventional cell in terms of dissipated power. I also discuss the effects of varying gate voltages and operation ranges on a TFET-based cell’s power and performance. However, due to the absence of a comparable RTD model (the RTD model in [39, 65] operates form −1.8 V to 1.8 V), I do not compare power to RTD-based cells.
Figure 4.5. Simulation results of TFET-based CNN array for different image processing templates.
4.5.1 Spice Model for TFET

To study the power consumption of a TFET-based cell I have developed a SPICE model for the TFET (see Fig. 4.6a) that consists of an nMOSFET and a voltage controlled switch in series, with a RTD in parallel. Node $d$, $g$, $s$ are the drain, gate and source for the modeled TFET device. The RTD is modeled by a non-linear dependent current source and a diode in parallel. The current source is defined by a physics based compact current-voltage relationship described in [87]. The RTD conducts current only when a negative voltage is applied across $d$ and $s$. On the other hand the voltage controlled switch turns ‘on’ when $V_{ds}$ is positive. Therefore, for positive $V_{ds}$, only the nMOSFET conducts current and generates the right side of the TFET I-V (Fig. 4.6b). For negative $V_{ds}$, the RTD generates the Negative Differential Resistance (NDR) region and the diode-like behavior of the TFET I-V (Fig. 4.6b). The nMOSFET is modeled by the built-in SPICE NFET with $w = 1 \mu m$, $l = 15 \ nm$, $V_{TO} = -0.15 \ V$, and $TOX = 62 \ nm$. The RTD is characterized by $A = 305 \ A/cm^2$, $B = 0.05 \ V$, $C = 0.1 \ V$, $D = 0.01 \ V$, $n_1 = 1.98$, and an area of $10^{-8} \ cm^2$. The internal diode in the RTD is modeled by a generic SPICE diode with $N = 1.25$ and $R_s = 75 \ \Omega$. Fig. 4.6b demonstrates that there is good correlation between the SPICE model and the TFET characteristics with $V_{gs} = 0.3 \ V$.

4.5.2 TFET-based CNN vs. Conventional CNN

There are two types of energy associated with a CNN cell’s analog computation: (i) dynamic energy – the energy required for the state to make a transition from an initial state to a target equilibrium point (either a white or black pixel value in the HLD example), and (ii) static energy – the amount of energy the cell continues to dissipate after the state has been stable. In a CNN cell, the energy for the computation is supplied by the current sources and dissipated by the functioning device (i.e., the resistor/TFET) and the capacitor (assuming the transition starts from 0
Figure 4.6. (a) SPICE circuit model for the TFET; (b) TFET I-V characteristics for $V_{gs} = 0.3$ V [59] and the fitted SPICE model.

$V$ as the initial state). The power dissipation of the current sources would depend on their internal voltage supply and implementations. In this work, I calculate the power for a CNN cell with respect to the functioning device (resistor/TFET) and the capacitor, and show that the TFET-based cell is more power efficient than the conventional cell.

When considering power, it is assumed that both the TFET-based cell and the conventional cell complete a given transition within the same time duration (i.e., iso-performance). The value of the capacitance for both cells are $100\, fF$. For the TFET-based cell, a transition from a zero initial state voltage to a white pixel value ($-0.8\, V$) would require $I_{eff}$ to be $-288\, \mu A$ (from the TFET I-V for $V_{gs} = 0.3$ V shown in Fig. 4.1). SPICE simulation of this cell suggests a settling time (90%) of $T_s = 254\, ps$. To determine the required $I_{eff}$ for a conventional cell that would complete the transition in the same time, I use the voltage-time relationship of an $RC$ circuit:

$$v(t) = v_{max} \left( 1 - e^{-\frac{t}{RC}} \right)$$

(4.3)
where \( v(t) = 0.9V_{\text{max}} \) at \( T_s \). From Eq. 4.3 one can calculate the required value of resistance, \( R \). At equilibrium, the current through \( R \) is \( I_{\text{eff}} \), hence \( I_{\text{eff}} = -0.8/R \).

From this, \( R = 1104 \, \Omega \), and \( I_{\text{eff}} = -724 \, \mu \text{A} \) for the conventional cell. Simulation of this cell verifies that its settling time (90\%) is the same as that of the TFET-based cell. The following expressions is used to calculate the dynamic and static power of a cell:

\[
P_{\text{dynamic}} = \frac{I_{\text{eff}}}{T_s} \int_0^{T_s} x(t) \, dt \tag{4.4}
\]

\[
P_{\text{static}} = |V_{\text{max}} I_{\text{eff}}| \tag{4.5}
\]

The state voltage values, \( x(t) \), for different time points are generated through SPICE and then integrated numerically to compute dynamic power. Fig. 4.7a compares the power dissipation between a TFET-based cell with state voltage ranging from \(-0.8 \, V\) to \(0.1 \, V\), a conventional cell (like one in Fig. 2.3b) with \(-0.8 \, V\) to \(0.1 \, V\) range, and another conventional cell with \(-0.73 \, V\) to \(0.1 \, V\) range (0.73 V is the predicted supply voltage for the 15 \( \text{nm} \) CMOS technology node [72, 73]). From Fig. 4.7a one can see that a transition to a white pixel is invariably the most power dissipating event due to the asymmetric characteristics of the voltage ranges. A TFET-based cell always outperforms the conventional cell because the required \( I_{\text{eff}} \) for a transition to a white pixel in a conventional cell is always higher than that of a TFET-based cell, given both of them reach equilibrium within same time length, \( T \). This is illustrated in Fig. 4.8 where the current through the functioning device (resistor/TFT) is plotted against time. Since at any given time the current through the functioning device and the capacitor sums up to \( I_{\text{eff}} \), the annotated regions \( Q_T \) and \( Q_R \) (Fig. 4.8) represent the total charge needed for the TFET-based cell and the conventional cell, respectively,
to charge their capacitors up to the target equilibrium. To complete the transitions to the same stable voltages, $Q_R$ must be equal to $Q_T$ because the capacitance for the two cells are same. For these two areas to be same, $I_{eff}$ for the conventional cell must be higher than that of the TFET-based cell. As a result, the power dissipation of a conventional cell would be higher. Note that as $I_{eff}$ is supplied by the current sources, they would also dissipate more power in a conventional cell (assuming the current sources in both TFET-based and conventional cell are realized with the same technology).

4.5.3 TFET-based Cells with Different $V_{gs}$

In this subsection I investigate the effect of varying gate voltages on a TFET-based cell. The TFET characteristics for lower $V_{gs}$ yields reduced current drive (see Fig. 4.1). Hence, in Fig. 4.7b reduced power dissipation for lower gate voltages is observed. However, as a byproduct of reduced $I_{eff}$, the lower $V_{gs}$ TFET-based cells would have larger delays. For example, the settling times (90%) (for a transition to a white pixel) are 254, 266, and 275 ps for $V_{gs} = 0.30, 0.25,$ and $0.20$ V. Applications with flexible time constraints would benefit by choosing lower $V_{gs}$. Moreover, variable $V_{gs}$ would also provide some flexibility in terms of designing the templates for different applications and tuning the devices in response to process variations.

4.5.4 TFET-based Cells with Different Operating Ranges

So far TFETs with state voltages ranging only from $-0.8$ V to $0.1$ V is considered. Here, I study the impact of lower voltage ranges on the TFET-based cell’s performance. For this, TFET characteristics is modulated to generate two models operating between: (i) $-0.5$ V to $0.1$ V, and (ii) $-0.3$ V to $0.1$ V (see Fig. 4.9). In this case study, the current drives for these models are simply scaled downed linearly to achieve the same settling time as the cell operating between $-0.8$ V to $0.1$ V. For
example, $I_{\text{eff}}$ (at white pixel voltage) for the $-0.8$, $-0.5$, and $-0.3$ V TFET-based cells are $-288$, $-180$, and $-108$ $\mu$A. SPICE simulation of cells based on these models suggest similar settling times. The power dissipation for the cells is depicted in Fig. 4.7c. The $-0.3$ V TFET-based cell achieves more than $7x$ improvement in terms of power dissipation. These results suggest that carefully devised low voltage TFET devices (see [28] for an example of a TFET with lower voltage range) can be very useful for low power CNN applications without sacrificing performance.

4.6 Conclusion

I have shown that it is possible to build functional CNN systems using TFETs. The TFET-based CNN facilitates reduced hardware footprints along with significant power savings when compared to the conventional CNN. A recent study, [96] shows that steep-slope TFETs can also be used to implement ultra low-power, low-throughput operational transconductance amplifiers (OTA) – which could potentially allow for building CNN systems entirely out of TFETs.
Figure 4.7. Dynamic and static power estimation assuming different device operating characteristics for transitions to both white and black pixel values.
Figure 4.8. Charge required for TFET-based cell, $Q_T$ and conventional cell, $Q_R$. For $Q_R$ to be equal to $Q_T$, $|I_{\text{eff}}^R|$ needs to be higher than $|I_{\text{eff}}^T|$.

Figure 4.9. Assumed models for low voltage TFETs.
CHAPTER 5

MULTI-VALUED CNN

5.1 Overview

Multi-valued CNNs provide a powerful alternative to solve general classification problems [46]. Though it is possible to decompose a multi-valued problem with $n$ classes into a series of $n - 1$ binary classification steps, both operation time and energy dissipation would obviously increase with this approach. Also, reprogramming hardware between iterations (for different template operations) would introduce additional overheads. Therefore, a CNN cell capable of classifying multiple classes in a single step would be useful. Existing CNNs achieve multi-value functionality either by using output function that has several saturated levels [46] or by the discrete time version of CNN (DT-CNN) [40].

In this chapter, I explore how emerging devices can be utilized to output more than two stable states (i.e., multi-value) in a CNN architecture. I demonstrate that devices such as graphene transistors [13] could be used successfully in CNN architectures to solve quaternary classification problems. Alternative solution through a binary CNN requires multiple iterations. Also, while “single device solutions” (e.g., via graphene transistors) for multi-valued problems are obviously attractive, it should also be noted that the likelihood of device-to-device mismatch, etc. would likely increase, which would in turn results in varying cell-to-cell behavior. Given this, I have also used a TFET-based circuit that exhibit useful properties (for multi-valued problems) in their output transfer characteristics.
To determine the utility of these approaches, as a case study, a tactile sensing problem from [50] is considered, where the goal is to detect and identify slippage of an object from a robotic arm. Architectural level simulations for this problem demonstrate that TFET circuit-based CNNs are capable of efficiently solving ternary classification problems. When compared to the conventional CNN, a TFET-based ternary CNN requires fewer computational steps and obviates the necessity of a complete datapath – thus reducing hardware by half.

5.2 Desired Characteristics

Here I discuss what properties of non-linearity are expected to solve both binary, and multi-valued problems in general. To perform binary classification, it is essential to have some form of non-linear relationship between $I_{eff}$ (see Fig. 2.3b) and the output voltage, which is provided by the non-linear output transfer function. It is also possible to introduce non-linearity to the state node’s behavior by replacing the linear resistor with a non-linear resistive component or device. Among the potential non-linear devices, devices with steep slopes and that operate at low voltages are especially interesting. Devices with steep current growth regions in their I-V characteristics can accommodate a wide range of currents within a very narrow window of voltages. As such, a number of different values of $I_{eff}$ corresponding to a specific output class would result in stable voltages within a very close range. The given output class can be represented by this narrow voltage window, and the steep-slope of the device itself ensures saturating state voltages. Therefore, the non-linear output transfer function is no longer required; i.e., the state voltage and output becomes equivalent to each other. Hence, by employing steep-slope devices in the CNN cell design, it should be possible to reduce the area of each cell. The low voltage operability of the potential devices would also improve power dissipation. In chap. 4 and [39, 65, 74], devices like TFET, and Resonant Tunneling Diodes (RTD) – that demonstrate the
aforementioned characteristics – are shown to solve binary classification problems with CNN architectures.

Now I discuss how a CNN architecture can be used to solve multi-valued problems in a single step by enforcing some characteristics in a non-linear device’s I-V. An example of a desirable device I-V characteristic is shown in Fig. 5.1a. More specifically, a device (or equivalent circuit) should exhibit more than two current growth regions as each growth region can represent an output level. Furthermore, very steep current growth regions are desired as this ensures that stable state voltages would be bounded within very narrow windows, and this would further eliminate the need for analog to digital conversion at the output. Also, it is desirable that the growth regions of a device are reasonably well separated, as this should lead to more robust operation in the presence of noise. Ideally, the slopes of the I-V characteristics in the separation regions should not be positive, as this could effectively merge two neighboring output windows. Interestingly, recent work on double-layer graphene transistors [13] suggests that a single device (Fig. 5.1b) might be able to deliver desired I-V characteristics (like those illustrated in Fig. 5.1a).

5.3 Example Multi-valued CNN

Assuming a piece-wise linear approximation (Fig. 5.1b) of the graphene transistor (for simplicity), I now demonstrate how CNN cells based on such characteristics can perform quaternary classifications. An example problem is used to illustrate quaternary CNN’s functionality. Given an image of an object as input (see Fig. 5.3a), the problem is to identify four distinct features, e.g., outliers, corners, edges, and cores. Each cell processes a single pixel of the input image. The input nodes of the cells are provided with the corresponding input image pixel (−1 V or +1 V for a white or black pixel, respectively). The cell states can be initialized with any random values. The template values for this problem are:
Figure 5.1. (a) Example characteristics of a CNN device for multi-valued problems; (b) I-V characteristics of the double-layer graphene device at room temperature \[13\], and its Piece-wise linear (PWL) approximation.
As the values in the feedback template are equal to zero, the DP component becomes the negated version of the device’s I-V. From Fig. 5.2, one can see that each steep-slope current growth region of the device is responsible for identifying one particular feature of the problem. For a given feature, the programmed templates offset the DP component by the proper amount of current such that the responsible steep-slope (negated in DP) intersects with the x-axis and results in an equilibrium state voltage. The ranges of stable voltages for the features are always finite, and mutually exclusive. Thus, with proper template design graphene transistor based cells can reach desired output states.

The cell array dynamics is verified via simulation. Fig. 5.3 illustrates the simulation results of the quaternary problem solved by both the graphene transistor based CNN (in a single computational step, Fig. 5.3a), and the conventional CNN (requires three computational steps, Fig. 5.3b-d). The resistor value in the conventional CNN is set to 36 $M\Omega$ such that the settling time for a binary step is similar to that of graphene transistor based CNN. During each step, the conventional approach eliminates one particular feature, and passes the remaining features to the succeeding step. Moreover, step 1 requires pre-processing to initialize the state nodes with the input image. Altogether, simulation suggest that the conventional CNN takes 59 ns to complete the quaternary classification whereas the graphene transistor based CNN needs only 16 ns.
Figure 5.2. DP plots for the representative cases of the four classes with \( k = 7 \times 10^{-9} \) and \( \varepsilon = 3 \times 10^{-9} \). The state equilibrium points are marked with open circles.

5.4 Circuits for Multi-valued Problems

While it is demonstrated that graphene transistors are useful, it is still in the early stages of development. Additional studies and experiments are required before it can be used in complex systems such as CNNs. In this section, I use a circuit\(^1\) realized via a more mature beyond-CMOS technology – an InAs TFET\(^7\) – to deliver multi-valued characteristics. The TFETs have a channel length of 20 nm, channel thickness of 7 nm and oxide thickness of 2.5 nm. The transfer characteristics of the circuit is shown in Fig. 5.4a. The four bias voltages \( V_{B1-4} \), can be used to tune the characteristics of the circuit. Examples of such tunings are shown in Fig. 5.5. This feature provides versatility for a CNN to perform different applications. Furthermore, this circuit allows the voltage swing to be as low as 0.5 V. The circuit also demonstrates a higher noise margin when compared to a CMOS-based circuit that shows similar non-linear characteristics\(^75\).

\(^1\)The circuit is designed by Behnam Sedighi from the University of Notre Dame.
5.5 Case Study

As a case study, I consider how CNNs that employ ternary classification circuitry can be used to solve a tactile sensing problem. Initially described in [50], CNNs can be used to process vertical sheer stress – which is an indicator of slippage of an
Figure 5.4. (a) Circuit for a ternary cell. (b) Its transfer characteristic with $V_{B1} = V_{B4} = 0.1V$, $V_{B2} = V_{B3} = -0.1V$, $V_{DD} = -V_{SS} = 0.25V$. The width of all TFETs is 1 $\mu$m. (This figure appears in [75].)

Figure 5.5. (a) Sweeping $V_{B1}$ and $-V_{B2}$ from 50 mV to 150 mV, (b) Sweeping $V_{B4}$ from 50 mV to 150 mV. (Other parameters are similar to those of Fig. 5.4b. This figure appears in [75].)
object grasped by a two-fingered robotic hand. Fig. 5.6 (from [50]) describes the time-evolution of the vertical shear stress (denoted as $T_y$) and its gradient when an object starts to slip out of the robotic grasp as a result of increased weight. Here, the task of the CNN system is to measure the duration of region ‘b’, and ‘c’ locally, which conveys information about how much weight is applied to the object and how fast it is slipping. Based on this data, the next level of computation (which may not be located with sensors) can make a decision about exactly how much grasping force should be applied to stop the slippage without causing any harm to the object.

Employing the TFET circuit-based ternary CNN, one can solve the slippage detection problem in 3 steps as shown in Fig. 5.7. The input to the system is a $64 \times 64$ gray scale image, where each column corresponds to a given sensor, and the image data reflect the normalized $T_y$ profile over time (as in Fig. 5.6a). To account for the 64 sensors in the system, data for each time step (i.e., a row in the input image) is generated by a normal distribution with the value from Fig. 5.6a (for the given time step) as the mean, and 10% of the value as variance. The first step calculates the gradient of $T_y$ with respect to time (similar to the data in Fig. 5.6b). The output of the gradient function shows the positive and negative bulges in the gradient profile (i.e., region ‘b’ and ‘c’) as black and white horizontal strips, respectively.

The next step, referred to as thresholding, classifies the output data of Step 1 into three distinct classes: positive (black) and negative (white) bulges, and no-change (gray) (see Fig. 5.7). The I-V characteristic of the TFET based-circuit (Fig. 5.4b) represents a black, white, and gray pixel by $(0.20, 0.25) V$, $(-0.25, -0.20) V$, and $(-0.05, 0.05) V$, respectively. The functionality of this step is explained in Fig. 5.8a, where the three stable state voltages (marked with open circles) are separated by the two meta-stable points $p$ and $q$. Any initial state voltage in the ranges of $(-\infty, p)$, $(p, q)$, and $(q, \infty)$ would eventually settle to one of the three stable voltages that reside in that range. Thus, the output of the threshold function contains pixels of
Figure 5.6. (a) Profile of the vertical component of sheer stress, $T_y$, from [50]. In region ‘a’, nothing is grasped. $T_y$ increases proportionally with the weight, in region ‘b’. During ‘c’ the object starts to slip out of the robot’s grasp. In ‘d’, $T_y$ becomes stable as it equal with the kinetic friction force; (b) The gradient profile of $T_y$.

only three distinct values.

The final step is the connected component detection (CCD) function that projects the presence of a black/white pixel in a horizontal line, to the leftmost column by the same colored pixel. As a result, by counting the number of black and white pixels in the first column, the lengths of region ‘b’ and ‘c’ are obtained. The information is processed by the next level for making further decisions. Note that, CCD requires its input pixels to be strictly stratified to one of the class levels for its correct functionality. This is why the threshold step is required.

As a comparison, a solution of the same problem based on a conventional binary CNN requires 5 steps and 2 sets of hardware (see [50] for more detail). This additional cost in both hardware and time is due to the fact that a binary classifier would only be able to retain information about one bulge in an image as one of the two stable output levels needs to be reserved for representing the empty space. This fact becomes evident also from Fig. 5.8b, where one can find at most two stable output levels for
any of the curves. Consequently, the solution of the problem through conventional CNN requires two different sets of hardware. Thus, TFET circuit-based CNN requires fewer computational steps (3 vs. 5), and reduces hardware by half.

5.6 Conclusion

Apart from overcoming the challenges in digital computation, emerging devices have a great promise for enabling new and enhanced functions for CNN-like non-von Neumann computing paradigms. Moreover, circuits built from emerging device technologies could also bring clear benefits over existing approaches. Initial study shows that at the cell level, multi-valued classifications improve the energy dissipation by an order of magnitude. Detail analysis of the energy, power, and performance – considering other system components (e.g., the current multipliers) – would be considered in future work.
Figure 5.7. The Universal Machines on Flows (UMF) description of the TFET circuit-based CNN for detecting slippage.

Figure 5.8. DP components for (a) TFET circuit-based CNN with the center matrix element of the feedback template, \( a_{00} = 7 \times 10^{-6} \); (b) Conventional CNN with 1 Ω resistance.
CHAPTER 6

MODELING CNN ENERGY DISSIPATION

6.1 Overview

As energy dissipation has become a first-class design consideration, it is important to be able to investigate the impact of various design options on energy. Existing analysis of CNN power dissipation is mostly performed and reported at the chip level (e.g., \cite{81,35}). This typically includes dissipations in the sensors, actuators, and accompanying digital components as well. Furthermore, such chip-level power dissipation do not explain the impact of various system parameters (e.g., the synaptic weights, resistive elements’ characteristics, etc.), and hence do not provide much insight as to how emerging technologies improve the status quo. An analytical model for calculating a CNN’s settling time was proposed in \cite{38} assuming a linear resistive element within cells, and hence cannot be applied to a non-linear CNN. Reliable analytical power and performance models are becoming even more important as the number of cells increases.

In this chapter, I propose analytical models to calculate the power dissipation and performance (in terms of settling times) for CNN applications. These models can be used for both linear and non-linear CNNs. The power model captures dissipations of both the cell core (i.e., the neurons) and the VCCSs (i.e., the synapses). It is shown that power dissipation of a CNN cell can be estimated in a normalized form from the application templates. When comparing energy/power between different CNN systems, the effect of the normalizing factor can be ignored to simplify comparisons.
If the exact energy dissipation of the system is of interest, the normalizing factor can also be deduced either: (i) directly, by providing some implementation specific parameters for the VCCSs, or (ii) indirectly, by simulating the power dissipation of a single VCCS. Applying the performance model, I show that more convex non-linear characteristics can enable faster signal processing by allowing higher current through the cell capacitor. This might be a critical feature when considering potential emerging devices in the context of non-linear CNN architectures.

I also have validated the analytical models against electrical circuit simulation, and found that the models can predict power and performance very reliably with errors less than 1% and 3%, respectively. Tactile sensing and pattern recognition problems are considered as case studies to compare power and performance between (i) non-linear CNNs that employs TFET circuits\textsuperscript{1} and (ii) conventional linear CNNs.

6.2 Model Derivation

In this section, I present the analytical models for power and performance. First, it is shown that a CNN dissipates energy at a constant rate (i.e., constant power) throughout the computation, and the rate can be calculated from its template values. I also discuss how to calculate the settling time of a CNN cell with arbitrary non-linear characteristics.

6.2.1 Model for Power Dissipation

Power dissipation of a CNN cell can vary based on a number of factors, e.g., transconductances of the VCCSs (application dependent), combinations of analog input signals, characteristics of the resistive component of neurons, etc. I show that among these various factors, the transconductances of the VCCSs captures the

\textsuperscript{1}The models can be applied to evaluate CNNs based on other emerging devices as well.
Figure 6.1. Assumed CMOS-based OTA driven CNN cell circuit.

major trend in power. Therefore, it can be reliably used to approximate the power dissipation. To explain the power model, a simple CNN cell with a VCCS realized by a CMOS-based operational transconductance amplifier (OTA) circuit (see Fig. 6.1) is assumed. In this circuit, the required power for computation is supplied by the voltage sources $V_{dd}$ and $V_{ref}$. Hence, the cell power dissipation is

$$P = V_{dd}i_{gnd} + (V_{dd} - V_{ref})i_{out} \tag{6.1}$$

As, $i_{dd} = i_{gnd} + i_{out}$, (6.1) can be expressed as

$$P = V_{dd}i_{dd} - V_{ref}i_{out} \tag{6.2}$$
The last term in (6.2) depends on the input signals provided to the OTA. In a CNN cell, the reference voltage $V_{\text{ref}}$ is usually selected halfway between $V_{dd}$ and ground such that the state at node $x$ can be differentially interpreted as both positive and negative voltages for different instances. Moreover, in an OTA circuit, the output current $i_{out}$ is significantly smaller than $i_{dd}$. As such, the last term in (6.2) is much smaller than the first term, and can be safely ignored for power modeling (corroborative simulation results are shown later). In a CNN array, the values of the last term for different cells assume positive or negative values with a resultant close to zero. Ignoring the last term also simplifies the power model by avoiding the use of the input signals as parameters.

The OTA in Fig. 6.1 consists of two current mirrors and one differential pair. The output current of the OTA can be captured by the following expression:

$$i_{out} = M_r I_b \tanh \left( \frac{\kappa v_{\text{diff}}}{2} \right)$$

(6.3)

where $M_r$ is the ratio of the current mirrors, $I_b$ is the bias current, $\kappa$ is a technology specific constant that represents the transistors’ effectiveness in controlling the gate energy barrier, and $v_{\text{diff}} = v_{in-} - v_{in+}$ is the differential input voltage of the OTA. The sigmoid-like hyperbolic tangent function provides a range of voltages $V_{\text{diff}}$, in which the relationship between $i_{out}$ and $v_{\text{diff}}$ is linear for a fixed $I_b$. By setting the value of $I_b$, the slope of this linear curve can be controlled and thus different transconductances ($G_m$ values) for the OTA can be programmed.

Utilizing the relationship in (6.3), I now show that all the cells in a CNN dissipate energy at a constant rate. For any voltage $v \in V_{\text{diff}}$, $i_{out}$ should be equal to $v G_m$. Hence, the corresponding $I_b$ for a given $G_m$ can be determined by

$$I_b = \frac{G_m}{M_r} v \coth \left( \frac{\kappa v}{2} \right)$$

(6.4)
Given a target $G_m$ and fixed $M_r$, all $v \in V_{diff}$ result in similar $I_b$ values. As $I_b$ is always unidirectional, both $\pm G_m$ are realized via the same $I_b$. For $-G_m$, the nodal voltages $v_{in+}$ and $v_{in-}$ are interchanged resulting in a negative $v_{diff}$, and a negative $I_{out}$ as per (6.3). Hence, $I_b \propto |G_m|$.

The power dissipation of the cell in Fig. 6.1 can be derived from (6.2) (ignoring the last term) as:

$$P = V_{dd} (M_r + 1) I_b$$

(6.5)

As $P \propto I_b$, it can be rewritten as:

$$P = \eta |G_m|$$

(6.6)

where,

$$\eta = V_{dd} \left(1 + \frac{1}{M_r}\right) \left(v \coth \frac{\kappa V}{2}\right)$$

(6.7)

In a CNN, each feedback, control, and constant current (i.e., the corresponding currents for template $A$, template $B$, and $Z$) is realized by a VCCS. Thus, for a given application, the power dissipation of cell $C_{ij}$ (see Fig. 2.3b) is:

$$P_{cell} = \eta \left[ |Z| + \sum_{C_{kl} \in N_{ij}} (|a_{ij,kl}| + |b_{ij,kl}|) \right]$$

(6.8)

where, the fixed bias $Z$ of the cell is implemented by an OTA with $G_m = Z$. The differential input of this OTA is permanently fixed to the maximum possible input signal. Cell power quantified by (6.8) includes dissipations for both the ‘neurons’, and ‘synapses’. Note that as the transconductances of the OTAs do not change over time, a CNN cell dissipates a fixed power over the course of a given computation.
Also, in a CNN array the transconductances of the OTAs are space invariant. Hence, the total power dissipation of the network is:

\[ P_{\text{cnn}} = MN P_{\text{cell}} \]  \hspace{1cm} (6.9)

By using (6.8) and (6.9), one can estimate the power dissipation of a CNN system (with both linear/non-linear resistive element in the neuron) for a given application. Furthermore, the model could be used to compare power dissipation across different CNN systems (assuming the same OTA technologies across systems) without knowing the details of the OTA implementation, i.e., without having to calculate \( \eta \). Power dissipation of OTAs built from beyond-CMOS technologies (e.g. a TFET-based OTA in [96]) can also be modeled by identifying the relationship between the OTA output currents and the input voltages (like (6.3)). Based on such relationships, similar strategy outlined here can be adopted to model the overall power.

### 6.2.2 Model for Settling Time

Besides power, it is also important to be able to predict the settling time for a CNN array. The settling time signifies that all the cells in the network have converged to corresponding equilibrium states, the output can be sampled, and new inputs can be provided. The settling time \( t_s \), for a CNN array is defined as the time required by the slowest cell in the network to reach a stable equilibrium voltage. As the cells dissipates energy at a fixed rate (as per (6.8)), the total energy dissipation of the system is:

\[ E_{\text{cnn}} = MN P_{\text{cell}} t_s \]  \hspace{1cm} (6.10)

An existing model for calculating CNN settling times [38] is applicable only for linear resistor based conventional CNNs. In this section, I present a general model
that is capable of calculating CNN settling time given an arbitrary resistive element (linear/non-linear). To determine the settling time of a CNN cell for a trajectory from an initial state voltage $x_0$ to a target equilibrium voltage $x_n$, the resistive element’s transfer characteristics is divided into $n$ equal voltage segments, where $n$ is a large positive integer. Within each of these segments the characteristic is assumed to be linear. Let the required time for the state to transition from $x_i$ to $x_{i+1}$ be $t_i$. Furthermore, let the corresponding piecewise linear (PWL) segment of the characteristic be represented as $h(x) = mx + b$ for $x_i \leq x < x_{i+1}$. As such, the linearity of the DP plot within the range $[x_i, x_{i+1}]$ can be expressed as $(\alpha - m)x + (w - b)$ (as per (4.2), assuming uncoupled CNN templates, e.g., $a_{ij,kl} = 0$ for $kl = ij$), where the self feedback parameter $a_{ij,ij} = \alpha$ (see Fig. 6.2). The ODE described by (4.2) can be
analytically solved to find $t_i$:

$$t_i = \frac{C}{m-\alpha} \ln \frac{x_i - \frac{w-b}{m-\alpha}}{x_{i+1} - \frac{w-b}{m-\alpha}}$$

(6.11)

Given this, (6.11) can be simplified as:

$$t_i = \frac{C}{m-\alpha} \ln \frac{x_i - L}{x_{i+1} - L} = -\frac{C}{m-\alpha} \ln \frac{L - x_{i+1}}{L - x_i}$$

$$= -\frac{C}{m-\alpha} \ln \frac{k_{i+1}}{k_i} = -\frac{C}{m-\alpha} \ln \left(1 - \frac{\Delta k}{k_i}\right)$$

(6.12)

where $k_i$ represents the capacitor current when the state voltage is $x_i$ (see Fig. 6.2).

Since $\frac{\Delta k}{k_i} \ll 1$, one can safely consider only the first term from the Taylor expansion of the logarithmic function to get:

$$t_i = \frac{C}{m-\alpha} \frac{\Delta k}{k_i} = C\Delta x \frac{1}{k_i}$$

(6.13)

As such, the total time $t$, for the transition from $x_0$ to $x_n$ can be expressed as:

$$t = \sum_{i=0}^{n-1} t_i = C\Delta x \sum_{i=0}^{n-1} \frac{1}{k_i}$$

(6.14)

Physically (6.14) can be interpreted as:

$$t = \frac{Cn\Delta x}{\left(\sum_{i=0}^{n-1} \frac{1}{k_i}\right)^n}$$

(6.15)

where the numerator represents the total change in capacitor charge due to the state transition from $x_0$ to $x_n$, and the denominator represents the average current (harmonic mean) through the capacitor. As there might be many combinations of initial
and final states for a cell, the settling time $t_s$ for a CNN is the longest of all such possible transitions.

6.3 Model Validation

In this section, the estimations made by the proposed models are validated against HSPICE circuit simulation results.

6.3.1 Model Validation for Power

To validate the proposed power model, I simulate a linear resistor based CNN in HSPICE for two image processing applications: (i) Optimal edge detection, and (ii) Convex corner detection (see Fig. 7.9). An OTA circuit based on 14 nm PTM CMOS devices (see Fig. 6.4a) is used as VCCSs for the cells.

The optimal edge detection problem is defined by the following templates [2]:

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} -0.11 & 0 & 0.11 \\ -0.28 & 0 & 0.28 \\ -0.11 & 0 & 0.11 \end{bmatrix}, \quad Z = 0$$

To implement this application, 6 individual OTAs are realized in each cell. Four of them are programmed with $G_m = 1.1 \ \mu A/V$, and the remaining two with $G_m = 2.8 \ \mu A/V$. A negative $G_m$ is programmed by interchanging the nodal voltages $v_{in+}$ and $v_{in-}$. Note that the standard $G_m$ values defined in the aforementioned templates assume a linear resistor value of 1 $\Omega$ in the cell neuron. As it is not practical to design OTA circuits with such high transconductances, the programmed $G_m$ values are compensated by using $R = 100$ $K\Omega$. For all the OTAs, the supply voltage $V_{dd} = 1$ $V$. Black and white pixels in the input images are represented by 0.9 $V$ and 0.1 $V$ respectively. One of the differential inputs of the OTAs is tied to $V_{ref} = 0.5$ $V$. 

78
Thus, the OTAs can recognize differential voltages in the range of $[-0.4, 0.4] \, V$.

A single such OTA was simulated (following the settings described in Fig. 6.1) to determine its power dissipation (via HSPICE) for an input differential voltage, $v_{\text{diff}} = 0V$. From the power dissipation, the value of $\eta$ for the OTA can be obtained using (6.6). For the OTA circuit in Fig. 6.4a, $\eta = 0.85 \, V^2$. From (6.8), and (6.9), for a $5 \times 6$ array:

$$P_{\text{cell}} = 0.85 \left[4 \times 1.1 + 2 \times 2.8\right] \times 10^{-6} = 8.5 \, \mu W$$

$$P_{\text{cnn}} = 5 \times 6 \times 8.5 = 255 \, \mu W$$
The template values for the convex corner detection problem [2] are given below:

$$A = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}, \quad Z = -8.5$$

This application requires 11 OTAs with four different $G_m$ values: 1 $\mu A/V$, 2 $\mu A/V$, 8 $\mu A/V$, and 8.5 $\mu A/V$. The corresponding value of $R = 1 \, M\Omega$. The estimated
power dissipation for convex corner detection problem are:

\[ P_{\text{cell}} = 0.85 \times (8 \times 1 + 2 + 8 + 8.5) \times 10^{-6} = 22.5 \, \mu W \]
\[ P_{\text{cnn}} = 5 \times 6 \times 22.5 = 675 \, \mu W \]

To verify the estimated power dissipations by the model, I simulate a 5×6 network for 10,000 random input images where the pixel values are drawn from a uniform distribution between [0.1, 0.9]V. For each input image, the power dissipation of the applications are obtained via HSPICE, and plotted in histograms (Fig. 6.5). The peaks of the distributions in the histograms for optimal edge detection and convex corner detection tasks, are situated at the bins centered at 257 \, \mu W and 681 \, \mu W, respectively. Hence, the power values obtained by the analytical models result in 0.8\% and 0.9\% errors, respectively.

6.3.2 Model Validation for Settling Time

To validate the analytical model for settling time, I consider a number of hypothetical non-linear transfer characteristics as depicted in Fig. 6.6. The settling time of a single cell for a given transition from 0 \, V to 0.25 \, V for each of the non-linear curves is calculated (assuming \( C = 100 \, fF \)) via both the analytical approach (Eqn. 6.14) and via HSPICE simulation (Table 6.1). The non-linear characteristics are realized in HSPICE via look-up table based Verilog models. As can be seen from Table 6.1, the settling times calculated by the analytical model are in good agreement with the settling times derived via simulation, with maximum error of 3\%. Also note that more convex\(^2\) non-linear characteristics result in faster settling time. This is due to the fact that convex characteristics allow higher capacitor current, and thus

\(^2\)The term “convex” is used in a rather loose sense where a more convex curve simply refers to a curve having a smaller area between the curve and the x-axis.
Figure 6.5. Histograms of ten thousands simulated power dissipations for (a) Optimal edge detection, and (b) Convex corner detection problems.

enables faster charging or discharging. Hence, when studying potential emerging devices for non-Boolean CNN computing, more convex non-linear characteristics might be desirable.

I also use the analytical model to estimate the settling times for the optimal edge detection problem. Settling times for different initial states to target state transitions are estimated, and the longest of all transitions is selected as the final settling time, $t_s$, which is 22.97 $ns$. To validate, a $32 \times 30$ CNN is simulated via HSPICE for the optimal edge detection task. From simulation, the settling time of the slowest cell is found to be 22.67 $ns$ – resulting in 1.3% error.

6.4 Case Studies

To demonstrate how the proposed analytical models can be used to compare power and performance between different CNN systems, two applications are considered as
case studies: (i) a tactile sensing problem, and (ii) a pattern recognition problem.

In a tactile sensing problem [50], the goal is to detect and prevent the slippage of an object held by a robotic arm. As already discussed in chap. 5.5, a TFET-based circuit [75] can provide a non-linear characteristic that can solve a ternary classification problem when used as a non-linear resistor in a CNN. By using such ternary CNN, the tactile sensing problem can be solved in three computational steps (see chap. 5.5), whereas a conventional binary CNN requires five steps employing two parallel hardware data paths [50] (see Fig. 6.7). The calculated power values and settling times for the two systems are reported in Table 6.2. Note that under the assumption that OTA technology across the systems do not change, calculating the exact value of $\eta$ (see (6.8)) is not required to compare power. Hence, for simplicity let $\eta = 1 \ V^2$. (6.8) is used to calculate the power dissipation of a cell that includes

3A CMOS-based approach that could enable ternary outputs has also been considered in [75]. However, it has been shown in [75] that its noise margins – to obtain reliable output values – are inferior to that of the TFET-based approach. As such, the CMOS based approach is not considered here.
TABLE 6.1

SETTLING TIMES FOR TRANSITIONS FROM 0 V TO 0.25 V

<table>
<thead>
<tr>
<th>Non linearity</th>
<th>Settling time (90%) from simulation (ns)</th>
<th>Settling time (90%) from model (ns)</th>
<th>Difference (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>123.88</td>
<td>123.00</td>
<td>1.0</td>
</tr>
<tr>
<td>2</td>
<td>105.11</td>
<td>104.00</td>
<td>1.1</td>
</tr>
<tr>
<td>3</td>
<td>62.65</td>
<td>61.10</td>
<td>2.5</td>
</tr>
<tr>
<td>4</td>
<td>8.60</td>
<td>8.41</td>
<td>2.2</td>
</tr>
<tr>
<td>5</td>
<td>7.95</td>
<td>7.71</td>
<td>3.0</td>
</tr>
<tr>
<td>6</td>
<td>7.34</td>
<td>7.14</td>
<td>2.8</td>
</tr>
<tr>
<td>7</td>
<td>6.86</td>
<td>6.69</td>
<td>2.4</td>
</tr>
<tr>
<td>8</td>
<td>6.88</td>
<td>6.82</td>
<td>1.0</td>
</tr>
</tbody>
</table>

dissipation of both the ‘neuron’ and ‘synapses’. I also consider all possible transitions for a cell, and among those the slowest one corresponds to the settling time, $t_s$. For propagating type problems (e.g., CCD), the worst case settling time is multiplied by 63 to account for the fact that within each column of the $64 \times 64$ input image, the computation might need to propagate from one end to other. For the tactile sensing problem, the proposed models show that the TFET circuit based CNN dissipates $4x$ less energy when compared to a linear, conventional CNN. Note that the linear resistor value in the conventional CNN is chosen such that the two systems require similar settling times (i.e, iso-performance). However, changing the linear resistor value scales the power and settling time by the same factor but in the opposite direction. Hence, the total energy dissipation of a conventional CNN cell remains constant irrespective of the value of the resistor.
TABLE 6.2

COMPARING ENERGY DISSIPATION FOR TACTILE SENSING

<table>
<thead>
<tr>
<th>steps</th>
<th>Settling time (ns)</th>
<th>Power/cell (µW)</th>
<th>Energy/cell (pJ)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ternary Init to 0</td>
<td>20.6</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Ternary Gradient</td>
<td>58.5</td>
<td>56.51</td>
<td>3.31</td>
</tr>
<tr>
<td>Ternary Threshold</td>
<td>60.1</td>
<td>15.0</td>
<td>0.90</td>
</tr>
<tr>
<td>Ternary CCD</td>
<td>4340.0</td>
<td>12.2</td>
<td>52.90</td>
</tr>
<tr>
<td>Total</td>
<td>4479.2</td>
<td>12.8</td>
<td>57.10</td>
</tr>
<tr>
<td>Conv Gradient</td>
<td>33.1</td>
<td>31.56</td>
<td>1.04</td>
</tr>
<tr>
<td>Conv ADD left</td>
<td>66.1</td>
<td>6.94</td>
<td>0.50</td>
</tr>
<tr>
<td>Conv INV left</td>
<td>66.1</td>
<td>3.47</td>
<td>0.23</td>
</tr>
<tr>
<td>Conv Threshold left</td>
<td>54.1</td>
<td>15.95</td>
<td>0.86</td>
</tr>
<tr>
<td>Conv CCD left</td>
<td>4260.0</td>
<td>26.36</td>
<td>112.00</td>
</tr>
<tr>
<td>Conv SUB right</td>
<td>66.1</td>
<td>6.94</td>
<td>0.50</td>
</tr>
<tr>
<td>Conv Threshold right</td>
<td>54.1</td>
<td>15.95</td>
<td>0.86</td>
</tr>
<tr>
<td>Conv CCD right</td>
<td>4260.0</td>
<td>26.36</td>
<td>112.00</td>
</tr>
<tr>
<td>Left path delay</td>
<td>4479.4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Right path delay</td>
<td>4413.3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>4479.4</td>
<td>51.00</td>
<td>228.00</td>
</tr>
<tr>
<td>Ratio</td>
<td>1.00</td>
<td>4.00</td>
<td>4.00</td>
</tr>
</tbody>
</table>
Figure 6.7. The Universal Machines on Flows (UMF) [82] description for detecting slippage via: (a) Conventional linear CNN; and (b) TFET circuit based ternary CNN. Each horizontal bar represent individual step of a given CNN work flow.
The aforementioned TFET-based circuit can also be used to solve a pattern recognition problem in a single step, where the task is to identify three distinct features (the outliers, the edges, and the cores) from a given input image by assigning different levels of grey at the output (Fig. 6.8). A linear, conventional CNN can solve the same problem in two consecutive steps. The power and settling times for this problem are reported in Table 6.3. For the pattern recognition problem, the energy dissipation of the TFET circuit based CNN cell improves by 1.47x when compared to the linear conventional cell. The higher energy benefit in the case of the tactile sensing problem results mostly from the fact that a conventional linear CNN requires a single CNN hardware for the pattern recognition problem, whereas for the tactile sensing problem it requires two sets of hardware that dissipate energy simultaneously.

One key factor in the proposed analytical model for power is the dissipation by VCCSs in a CNN cell. To investigate the impact on power when implementing a VCCS with TFET and CMOS devices, a TFET-based operational transconductance amplifier (OTA) is also considered with transconductance, $G_m = 2\mu A/V$, and a linear range of $[-250mV, 250mV]$ (Fig. 6.4b). The OTA uses InAs homojunction TFET devices [7]. Detail device characteristics are discussed in [89]. The TFET-based OTA dissipates 1.6X less power, and runs 1.15X faster when compared to the 14nm PTM CMOS based OTA implementation shown in Fig. 6.4a (see Table 6.4). As such, a CNN system built exclusively from TFET devices for both the ‘neurons’, and ‘synapses’, can enable further energy savings.

6.5 Conclusions

Similar to designing von Neumann architectures, being able to quickly estimate power and performance of CNN is critical for exploring different design options. I proposed analytical models for estimating the power and performance of a CNN sys-
### TABLE 6.3

**COMPARING ENERGY DISSIPATIONS FOR THE PATTERN RECOGNITION PROBLEM**

<table>
<thead>
<tr>
<th>Steps</th>
<th>Settling time</th>
<th>Power/cell</th>
<th>Energy/cell</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(ns)</td>
<td>(µW)</td>
<td>(pJ)</td>
</tr>
<tr>
<td>Ternary Initialization to 0</td>
<td>18.3</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Ternary single step</td>
<td>106.0</td>
<td>18.9</td>
<td>2.00</td>
</tr>
<tr>
<td>Total</td>
<td>124.3</td>
<td>16.1</td>
<td>2.00</td>
</tr>
<tr>
<td>Conv Initialization by input</td>
<td>30.3</td>
<td>7.6</td>
<td>0.23</td>
</tr>
<tr>
<td>Conv step1</td>
<td>47.1</td>
<td>23.5</td>
<td>1.11</td>
</tr>
<tr>
<td>Conv step2</td>
<td>47.1</td>
<td>34.0</td>
<td>1.60</td>
</tr>
<tr>
<td>Total</td>
<td>124.5</td>
<td>23.6</td>
<td>2.94</td>
</tr>
<tr>
<td>Ratio</td>
<td>1.00</td>
<td>1.47</td>
<td>1.47</td>
</tr>
</tbody>
</table>

### TABLE 6.4

**COMPARING TFET AND CMOS BASED OTA**

<table>
<thead>
<tr>
<th>Parameters</th>
<th>TFET OTA</th>
<th>CMOS OTA</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_{DD}$ (V)</td>
<td>0.8</td>
<td>1.0</td>
</tr>
<tr>
<td>Power (µW)</td>
<td>1.0</td>
<td>1.6</td>
</tr>
<tr>
<td>Delay (ps)</td>
<td>19.8</td>
<td>22.9</td>
</tr>
</tbody>
</table>
(a) Single step of TFET circuit based CNN

(b) Step 1 of conventional CNN

(c) Step 2 of conventional CNN

Figure 6.8. Computational steps for the pattern recognition problem.

tem. The models are capable of handling arbitrary non-linear characteristics in the CNN cell dynamics. Applying the models, I show that convex non-linear characteristics result in faster CNN operations. The models are also utilized to compare energy dissipations between a TFET circuit based non-linear CNN and a conventional linear resistor based CNN for a tactile sensing, and a pattern recognition problem.
CHAPTER 7

MIXED SIGNAL ARCHITECTURE FOR CNN

7.1 Overview

The ubiquitous, CMOS-based von Neumann (i.e., stored program) architecture excels for high precision arithmetic operations. However, when considering the more complex processing requirements of high volume, spatio-temporal information, the same architecture performs poorly. Examples of such applications include image processing, video analytics, pattern recognition/formation, etc. Furthermore, the scalability of the underlying CMOS technology is approaching its physical limits giving rise to issues like increased device mismatch, power densities, etc. As such, exploring non-von Neumann computing architectures as well as new device technologies for efficient realization of spatio-temporal applications are of heightened interest.

In this chapter, I propose a non-von Neumann, mixed signal (MS) architecture which is augmented from a TFET-based architecture described in [90]. The mixed signal architecture in [90] performs CNN based template operations in a restricted setting. The proposed MS architecture overcomes its limitations, and can address a much wider range of interesting and complex applications.

The architecture in [90] specifically exploits the unique characteristics of TFET devices. TFETs are a promising candidate for realizing energy efficient digital circuits in the post-CMOS era. Their on-current ($I_{on}$) to off-current ($I_{off}$) ratio can be made large [88, 62, 95]. Subthreshold swings as low as 21 mV/dec have been observed experimentally [95]. Previous publications have examined the merits of
TFET-based digital circuits, and have shown that at low supply voltages, TFET-based digital circuits can be more energy efficient when compared with conventional CMOS designs [85, 71, 58, 94]. Turning to analog applications/circuits, researchers have also considered how the higher $g_m/I_D$ of TFETs in the subthreshold region could be employed to design low-power amplifiers [91, 96, 60, 61].

The proposed MS architecture processes analog input signals and produces a digital output. For many applications, this built-in analog-to-digital (A-to-D) conversion step is critical to reduce the amount of data sent to digital processors, and hence the overall system energy dissipation. The MS architecture eliminates the need for voltage controlled current sources (VCCSs) which are needed to realize feedback and feedforward templates [21], and are the dominant source of power dissipation in a CNN array as explained in chap. 6.2.1. Instead, VCCSs are replaced with simple comparators that can be efficiently realized with TFETs given their high intrinsic gain. The architecture converts input voltage to pulse-width, and measure pulse width with the aid of a high frequency clock. (There are some similarities to single-slope A-to-D converters that have been partially explored in other efforts [29, 68].)

In essence, this work is analogous to highly parallel processing platforms similar to single-instruction-multiple-data (SIMD) processors [32, 76, 16], cellular neural networks (CNNs) [80, 56, 29], or vision chips [67, 98, 57, 23]. In contrast to existing CNN architectures, the proposed MS architecture approaches the driving point dynamics of CNN cells, and accordingly follows predefined algorithmic steps to identify the final state equilibrium. The performance comparisons for many of the CMOS processors referenced above will be summarized and discussed later in this chapter in order to place this work in the proper context. Specifically, mega-operations per cell (MOPS), and giga-operations per second per Watt (GOPS/W) are used as measures of performance efficiency, and power efficiency, respectively. (In both instances, higher numbers are more desirable). The architecture demonstrates more
than 2X improvement in power efficiencies over state-of-the-art architectures that assume MOSFET/FinFET technology and seek to accomplish similar information processing tasks.

7.2 Architecture for Zero-feedback CNN Applications

In this section, I briefly discuss an initial TFET-based MS architecture from [90] that is capable of executing zero-feedback CNN templates, where all elements in the feedback template \( A \) is zero. Later in Sec. 7.3, I demonstrate how this architecture can be augmented to execute more general uncoupled CNN templates with non-zero feedbacks.

In an analog CNN, the VCCSs implemented by OTAs suffers from issues like transistor mismatches and process variation that prevent having well-defined gains and introduce offsets. Moreover, at small supply voltages, maintaining linearity for a large input range is more difficult. To overcome these issues, A TFET-based mixed signal architecture for CNN-like signal processing is presented in [90] (see Fig. 7.1), which unlike the conventional analog CNN employs a single, comparator circuit within each cell to realize the multiplication operations of the VCCSs in a time multiplexed manner. The non-ideal effects in a comparator may cause input-referred offset. However, as gain-error and non-linearity are not relevant in a comparator, it is relatively easier to attain robustness.

The architecture in [90] consists of a homogeneous array of processing cells. Each cell receives an analog signal as its input, communicates with its neighbors, and produces a digital output. To facilitate multiplication operations, a frequency synthesizer is used to generate variable clock frequencies. Also, the architecture employs a ramp generator to generate a ramp signal \( V_{ramp} \), and provides it as one input to the comparator in each cell. A control unit sets the frequency of the synthesizer, starts and stops the ramp, and applies proper settings for the cells. The comparator
Figure 7.1. (a) CNN architecture for zero-feedback applications [90]; (b) a cell in \( i^{th} \) row and \( j^{th} \) column. (This figure appears in [90].)

Figure 7.2. Voltage to pulse-width conversion and measurement. (This figure appears in [90].)
and the AND gate together generates a pulse, $p_{i,j}$ whose width is proportional to the cell’s analog input voltage $u_{i,j}$ (see Fig. 7.2). The cell logic unit routes this pulse, and other pulses generated by the neighboring cells to the clock gate in a time multiplexed manner. A digital counter, Y can measure the pulse-width $T_{i,j}$ (see Fig. 7.2). By counting down for a reference voltage $V_m$ during a given ramp period, and counting up for an input signal $u_{i,j}$ during the following ramp period (two such consecutive ramp periods constitute a ramp pair), the net change of the counter value becomes:

$$\Delta y_{i,j} = (T_{i,j} - T_m) f_{CLK} = (u_{i,j} - V_m) f_{CLK} / s_{ramp}. \quad (7.1)$$

where $s_{ramp}$ is the slope of the ramp signal (in V/s), $f_{CLK}$ is the applied clock frequency. Alternation between $u_{i,j}$ and $V_m$ cancels any offset introduced by the comparator. Per (7.1), the counter accumulates a digital value proportional to an applied differential input voltage ($u_{i,j} - V_m$). The multiplication/scaling factor can be controlled by changing the value of $f_{CLK}$ that drives the counter, and thus different template values can be realized.

7.3 MS Architecture for Uncoupled CNN Applications

Many applications that can benefit from CNN-based spatio-temporal signal processing, utilize templates with non-zero feedback values. The architecture introduced in Sec. 7.2 cannot solve such problems. Also, it cannot realize the output transfer function (Eq. 2.2), hence resulting digital state might overshoot the range of voltages defined for input signals. In this section, I present an augmented MS architecture that overcomes these limitations. Specifically, the augmented MS architecture can solve uncoupled CNN templates where, $a_{ij,kl} = 0$ for $kl \neq ij$, in Eq. 2.1 and it also realizes the output transfer function by not allowing the state to grow beyond a threshold voltage in either the positive or negative direction.
The augmented MS architecture determines the output for a given template by analytically solving a corresponding driving point (DP) plot. For uncoupled CNNs, the cell equation described in (Eq. 2.1) can be rewritten as:

\[ C \frac{dx_{ij}(t)}{dt} = -\frac{x_{ij}(t)}{R} + \alpha y_{ij}(t) + w \]  

(7.2)

where, \( w = \sum_{c_{kl} \in N_{ij}} b_{ij,kl} u_{kl} + Z \), and the self feedback parameter \( a_{ij,ij} = \alpha \). Fig. 7.3 shows the DP plots for different values of \( \alpha \). For a simple case of \( \alpha = 0 \) (the dotted straight line in Fig. 7.3), the slope of the DP plot becomes \(-1\). This is why the accumulated offset value, \( w \) for the architecture in Sec. 7.2 directly represents the output. For uncoupled problems (i.e., \( \alpha \neq 0 \)), the relationship between \( w \) and the state equilibrium assumes different forms at different piecewise linear (PWL) segments of the corresponding DP plot. Hence, for a given value of \( \alpha \) and calculated offset \( w \), one has to know the corresponding operating point in the DP plot, and determine the state equilibrium and final output accordingly. For different cases of \( \alpha \), Table 7.1 summarizes the required checks to determine the operating point within DP, and the corresponding required actions to calculate the output \( y \).

Fig. 7.3 illustrates that the first derivative of the DP plots become discontinuous at \( x = \pm x_{\text{max}} \) for an \( \alpha \neq 0 \). The DP plot’s slope within the range \((-x_{\text{max}}, x_{\text{max}})\) can be determined from (Eq. 7.2) as \((\alpha - 1)\). Hence, the corresponding threshold along the y-axis, where the discontinuity triggers can be expressed as \( y_{\text{th}} = |\alpha - 1|x_{\text{max}} \).

The four different cases from Table 7.1 are discussed as follows.

1. \( \alpha > 1 \): If \( w \) crosses \( \pm y_{\text{th}} \) (i.e., \(|w| \geq y_{\text{th}}\)), then \( y \) is either \( x_{\text{max}} \) or \(-x_{\text{max}}\) depending on the direction of crossing (see the equilibriums along the red lines in Fig. 7.3). If \( w \in (-y_{\text{th}}, y_{\text{th}}) \) (i.e., \(|w| < y_{\text{th}}\)), the cell computes the DP plot’s intersection with the x-axis, \( x_m \) as \( w/(1 - \alpha) \). Note that \( x_m \) is a metastable state (as the DP plot intersects the x-axis with a positive slope). The output \( y \) depends on the
Figure 7.3. Uncoupled CNN cell’s Driving point plot for different values of \( \alpha \) (R = 1 \( \Omega \)). \((-x_{\text{max}}, x_{\text{max}})\) defines the range of input voltages.
TABLE 7.1

CONDITION CHECKING AND CORRESPONDING ACTIONS FOR
DIFFERENT CASES OF $\alpha$

<table>
<thead>
<tr>
<th>Case</th>
<th>Check1</th>
<th>Action1</th>
<th>Check2</th>
<th>Action2</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\alpha &gt; 1$</td>
<td>$</td>
<td>w</td>
<td>\geq y_{th}$</td>
<td>$w \geq 0$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$</td>
<td>w</td>
<td>&lt; y_{th}$</td>
</tr>
<tr>
<td>$\alpha = 1$</td>
<td>$w \geq 0$</td>
<td>$y = x_{max}$</td>
<td>$w &lt; 0$</td>
<td>$y = -x_{max}$</td>
</tr>
<tr>
<td>$\alpha = 0$</td>
<td>$</td>
<td>w</td>
<td>\geq y_{th}$</td>
<td>$w \geq 0$</td>
</tr>
<tr>
<td></td>
<td>$</td>
<td>w</td>
<td>&lt; y_{th}$</td>
<td>$y = w$</td>
</tr>
<tr>
<td>$\alpha &lt; 1 \land \alpha \neq 0$</td>
<td>$</td>
<td>w</td>
<td>\geq y_{th}$</td>
<td>$w \geq 0$</td>
</tr>
<tr>
<td></td>
<td>$</td>
<td>w</td>
<td>&lt; y_{th}$</td>
<td>$y = w/(1 - \alpha)$</td>
</tr>
</tbody>
</table>
initial state voltage \( x_0 \). If \( x_0 \in (x_m, \infty) \), \( y \) is \( x_{\max} \), otherwise it is \(-x_{\max}\) (see the equilibriums along the blue lines in Fig. [7.3]).

2. \( \alpha = 1 \): As the DP plot’s slope within the range \((-x_{\max}, x_{\max})\) is zero, \( y \) is either \( x_{\max} \) or \(-x_{\max}\) depending on the sign of \( w \). If \( w \geq 0 \), \( y \) is \( x_{\max} \), otherwise it is \(-x_{\max}\).

3. \( \alpha = 0 \): In this case \( y_{th} = x_{\max} \). If \( w \) crosses \( \pm y_{th} \), then \( y \) is either \( x_{\max} \) or \(-x_{\max}\) depending on the sign of \( w \). If \( w \in (-y_{th}, y_{th}) \), then the offset \( w \) directly represents the output, i.e., \( y = w \).

4. \( \alpha < 1 \land \alpha \neq 0 \): If \( w \) crosses \( \pm y_{th} \), then \( y \) is either \( x_{\max} \) or \(-x_{\max}\) depending on the sign of \( w \). If \( w \in (-y_{th}, y_{th}) \), the cell computes output \( y \) as \( w/(1 - \alpha) \).

To perform the checkings, and the actions described in Table 7.1, the cell structure is augmented as shown in Fig. 7.5. Each cell in the proposed MS architecture uses a Direct Digital Synthesizer (DDS) to perform the multiplication operations as opposed to the zero-feedback architecture, where a single DDS is used globally for all the cells. Each time, to reprogram template values, a new value of \( N \) is applied to the DDS (see Fig. 7.4). The DDS output (the most-significant bit (MSB) of the accumulator), goes to all cells. The output frequency is \( f_{\text{REF}}N/2^k \), where \( k \) is the word-length of the accumulator. \( N \) can be any integer in the \([0, 2^{k} - 1]\) range. \( k \) depends on the desired accuracy of the calculations and the ratio between the largest and smallest output frequencies (i.e., the largest and smallest weights) is \( 2^k - 1 \). As such, (7.1) can be expressed as:

\[
\Delta y_{i,j} = (u_{i,j} - V_m) \frac{N f_{\text{REF}}}{2^k s_{\text{ramp}}} = u_{i,j}^{\text{diff}} NK
\]  

(7.3)

Thus, the cell is capable of generating a product of (i) a differential analog voltage \( u_{i,j}^{\text{diff}} \), and (ii) a digitally programmed constant \( N \), with a normalizing factor \( K = f_{\text{REF}}/2^k s_{\text{ramp}} \), and accumulating the product to a given \( Y \). For calculating the
contribution of a template value of 1, if register $N$ is loaded with $N^1$, then for an arbitrary template value $b$, $N$ should be loaded with $bN^1$ to have the desired effect.

The steps from Table 7.1 are executed as follows. To assign $y = x_{max}$, (i) the counter $Y$ is reset and set to count ‘UP’, (ii) the register $N$ is loaded with $N^1$, and (iii) the input voltage $x_{max}$ is applied to the cell. The final counter value is a normalized version of $x_{max}$. In a similar manner, $y$ can be assigned with $-x_{max}$ just by flipping the count direction. Flip-flop, $DF$ is used as a flag to store the sign of the counter value. As the calculated offset $w$ is stored in the counter which assumes 2’s complement numbering, checking its sign via $DF$ is straightforward.

To check whether $|w| \geq y_{th}$, the cell adds $y_{th}$ to $w$ if $w$ is negative, otherwise $y_{th}$ is subtracted from $w$. (In more detail, to subtract $y_{th}$ (which is $|\alpha - 1|x_{max}$) from $w$ (i) the counter is set to count ‘DOWN’, (ii) the register $N$ is loaded with $|\alpha - 1|N^1$, and (iii) the input voltage $x_{max}$ is applied. Similarly, addition can be performed.) Another flip-flop $PF$ is used to store the sign of this addition/subtraction result. If the sign of this result agrees with the original sign of $w$, the comparison evaluates to ‘true’.

To calculate the value of $w/(1 - \alpha)$, (i) the absolute value of $w$ (stored in the counter) is loaded to register $N$, (ii) the counter $Y$ is reset, (iii) a fixed analog voltage $1/K(1 - \alpha)$ is provided to the comparator differentially, and (iv) the resulting product is added to $Y$ if $w \geq 0$, or subtracted otherwise. The final counter value per (7.3) becomes $w/(1 - \alpha)$.

7.4 Evaluation

All TFET-based circuit simulations in this chapter, assume InAs homo-junction TFETs (HomJTFETs) [7] as this device is one of the most mature and well-studied TFETs. An analytical TFET model described in [63] is used for circuit-level simulations.
Figure 7.4. Direct Digital Synthesizer (DDS). (This figure appears in [90].)

Figure 7.5. Augmented MS cell circuit for uncoupled CNN templates.
7.4.1 Benefits of TFETs in MS Architecture

The proposed architectures benefit from two unique characteristics of TFETs: (i) higher $g_m/I_{DS}$ in sub-threshold region, and (ii) higher output resistance in the saturation region. The input differential transistors of the voltage comparator (see Fig. 7.6a) operate in subthreshold region. As TFETs have a higher $g_m/I_{DS}$ in sub-threshold region than that of CMOS, power dissipation of TFET-based comparator will be lower than a CMOS-based one. The ramp generator circuit (Fig. 7.6b) utilizes a current mirror to generate a fixed current $I_r$ to charge up a capacitor $C_{int}$ via a current mirror. The higher output resistance of TFETs (i.e., more flat plateau in saturation region) helps to realize simplified and robust current mirrors.

In Fig. 7.7, the TFET-based ramp generator’s output is compared to that of a CMOS-based (14 nm PTM CMOS LV) ramp generator that utilizes the same topology as in Fig. 7.6b. The TFET-based ramp generator performs better simply because of the fact that TFETs help to build better current mirrors. Note that it is possible to design a similar performing CMOS current mirror for a CMOS-based ramp generator. The power dissipation of such circuit will still remain same as long as the programmed bias current $I_r$ and supply voltage are kept same, however such a current mirror would require higher transistor counts effectively increasing the area. Moreover, additional processing of the comparator generated pulse ($p_{i,j}$) are transferred to the digital domain, where low-power circuits can be designed with TFETs.

The architectures utilize digital counters, and frequency synthesizers (DDS) within each cell. To demonstrate TFET’s benefit over CMOS in architectural context, these components are also benchmarked in Fig 7.8. Note that, in addition to being more power hungry, CMOS LV based counter and DDS also reach maximum operating frequency at 160 MHz.
7.4.2 Evaluating MS Architecture

Power dissipation of a TFET-based uncoupled MS cell is measured via transistor-level simulation in HSPICE. The supply voltages for the analog and digital components of the cell are 0.8 V, and 0.4 V, respectively. A ‘black’/‘high’, and ‘white’/‘low’ input signal is represented by 0.5 V, and 0.3 V, respectively. The reference voltage $V_m$ is set to 0.4 V. A ramp of $1 V/\mu s$ is used that raises the voltage from 0 V to 0.6 V over a period of $0.6 \mu s$. The clock $f_{REF}$ is set at 256 MHz. The size of the counter (Y) is 8 bits. As the register $N$, and $DDS$ store only positive numbers, their size is set to 7 bits. The $N$ register is loaded with a value of 10 for reflecting a $B$ template value of 1. For implementing different template values, $N$ is accordingly loaded with linearly scaled values. A TFET-based MS cell is simulated to operate on a Logical AND template. Correct operations are observed. The measured total power dissipation of the cell is $275 nW$. 

Figure 7.6. TFET-based (a) Comparator circuit, and (b) Ramp generator from [90].
Figure 7.7. Simulated ramp voltage and its error for TFET-based and CMOS-based ramp generators.

Figure 7.8. Simulation results for power dissipation of 8 bit counter and DDS block. All supply voltage is 0.4V.
The proposed cell is also simulated assuming 14nm PTM CMOS LV technology. All the simulation parameters are same as for the TFET-based cell simulation. The measured power dissipation of this CMOS “drop-in replacement” is 330nW. As such, HomJTFET demonstrates 1.2X improved power dissipation when compared to CMOS LV.

To verify the functionality of the uncoupled MS architecture, it is also simulated via Verilog-AMS at RTL behavioral level. The verilog simulation is device independent, and focuses on architectural functionality. A FSM controller to dictate the cell operations is also modelled. An MS architecture with 30x32 cells is simulated for two image processing tasks, namely: (i) optimal edge detection, and (ii) convex corner detection. Correct operations are observed (see Fig. 7.9).

The TFET-based architectures are also compared with other CMOS-based architectures from the literature. For the zero-feedback CNN architecture, an optimal edge detection template is considered. For the uncoupled CNN architecture, four CNN templates, namely – optimal edge detection, diagonal line removal, vertical line removal, and binary edge detection are considered. Quantitatively, when compared with other CMOS-based architectures, the TFET-based uncoupled MS architecture has modest processing ability (MOPS), and superior power efficiency (GOPS/W). Per Fig. 7.10 (which plots GOPS/W as a function of MOPS), the TFET-based MS architecture approaches the desired corner of the graph – where both performance and power efficiency are maximized. For the measurements, a ramp of 1 V/µs is used. A faster ramp requires GHz clock frequencies for similar accuracy (8-bit output), whereas a slower ramp duration improves the accuracy but lowers the throughput. Although the presented architecture loses some flexibility when compared to a digital processor, it has the advantage of having a built-in A/D conversion and compact hardware.
7.5 Delay-based Operation via MS Architecture

In this section, it is demonstrated that the MS architecture can also be used to detect motion of objects that moves slower than a threshold velocity. For this task, [51] has mathematically formalized a delay-based CNN model that tracks the position of an object not only during a given instance but also during a pre-specified moment back in the past. By using these spatial information during two distinct temporal co-ordinates, the motion of the object (if moving slower than the threshold velocity) can be detected throughout its trajectory.

For a diagonally moving object, for each cell, the templates monitor the input pixel at the current time $t$, and also the input of the top-left neighboring cell at time $(t - \tau)$, where $\tau$ is the time required for the object (travelling at the threshold velocity, $V_{th}$) to move from the top-left neighbor cell to the given cell. If an object is detected both by a given cell at time $t$, and by its top-left neighbor at time $(t - \tau)$, we can conclude that the object is moving at a velocity less than or equal to $V_{th}$. To realize the templates using conventional analog CNN architecture, we would need a reliable delay element in the circuit that can defer a continuous signal for a precise period. As the required delay might be very long (order of ms), it is hard to design...
Figure 7.10. Comparisons between the TFET-based architectures to different CMOS digital and analog processors (GOPS/W vs. MOPS). ▽ and ◦ indicate existing digital and analog implementations, respectively.

an energy efficient, precise delay element in analog paradigm.

Unlike conventional CNN, the MS architecture multiplexes each of the associated template functions in time domain, and accumulates the function results in a built-in memory (an 8 bit counter). Hence for regular CNN templates (i.e., without delay), all input signals must be sampled and held until all the template functions are completed (see Fig. 7.11a). All template operations in this case are functions of inputs sampled during the same time. However, delay-based templates include functions that require signals from both time \( t \) and \( (t - \tau) \). If we sample the inputs at \( \tau \) sampling interval, the functions of the delay-based motion detection template can be organized as shown in Fig. 7.11b such that the function associated with top-left cells operation \( f_1 \) gets its input during time \( (t - \tau) \), and the function associated with the given cells operation \( f_2 \) gets its input during time \( t \). Note that the MS architecture leverages its internal memory to achieve this goal. Even under the assumption of inputs being sampled at interval \( \tau \), a conventional CNN cannot perform the computation in this setting.
because the result of function $f1$ will be lost as there is no way to store it in a conventional cell.

To verify the MS architectures capability to detect motion, we simulate five diagonally arranged cells in HSPICE for the following three cases: (i) the object moves at a velocity $V_{th}$, i.e., the pixel moves to the next cell exactly after $\tau$ time, (ii) the object moves at a velocity slower than $V_{th}$, i.e., the pixel moves to the next cell at least after $2\tau$ time, and (iii) the object moves at velocity faster than $V_{th}$, i.e., after $\tau$ time the pixel moves to the second next or even further. The simulated cells state are shown in Fig. 7.12. For case (a), and (b), the motion of the object is detected in the output image which is represented by a ‘black’ pixel. In the case of (c), where the object moves faster than threshold velocity, we cannot detect the trajectory.

7.6 Conclusions

In this chapter, a TFET-based MS architecture is proposed that can solve a wide range of CNN templates. The proposed architecture approaches the driving point dynamics of CNN cells, and accordingly follows predefined algorithmic steps to identify the final state equilibrium. Initial projections suggest that power efficiencies of more than 2,000 GOPS/W could be achievable with this architecture. This represents an improvement of more than 2X over state-of-the-art architectures that assume MOSFET/FinFET technology.
Figure 7.11. Operation of MS architecture for a template with two functions $f_1$ and $f_2$. (a) For regular CNN template (no delay) computation, (b) for delay-based template computation.

Figure 7.12. Simulation outputs of the delay-based motion detection task for an object moving (a) at a given threshold speed, (b) slower than the threshold, and (c) faster than the threshold. Object is identified by the presence of ‘black’ pixel.
CHAPTER 8

CONCLUSIONS

In this work, I have presented an analysis technique for NML circuits to determine the necessary conditions for a given magnet to remain in or transition to a desired magnetization state. Using this analysis, I have developed a design methodology for generating working NML layouts. The methodology is further augmented to support layouts at room temperature. I have also demonstrated that low voltage, steep-slope devices like TFETs can be successfully employed in a CNN system to solve various image processing tasks with wins in performance, power, and area. To help guide future device development, I have outlined a number of desired properties of non-linear characteristics in the context of CNN functionality. I also show how emerging, beyond-CMOS devices could help to enhance the capabilities of CNNs for solving tasks with non-binary outputs. To estimate the energy dissipation of a CNN system with arbitrary non-linear/linear resistance, I have proposed analytical models for estimating power, and performance for a given CNN application. Also, I propose a TFET-based mixed signal architecture that can perform a wide range of CNN applications in an energy efficient manner when compared to existing von Neumann architectures performing similar tasks.


