# Coupled Oscillator Networks for Computation in the Beyond-CMOS Era

*Submitted in partial fulfillment of the*
*requirements for the degree of*

Doctor of Philosophy

*in*

Electrical and Computer Engineering

Yunus Emre Kesim

B.S., Electrical and Electronics Engineering, Bilkent University
M.S., Electrical and Electronics Engineering, Bilkent University
M.S., Electrical and Computer Engineering, Carnegie Mellon University

Carnegie Mellon University
Pittsburgh, PA

December 2019

# Acknowledgments

I would like to express my gratitude for my adviser Prof. Jeffrey Weldon for his guidance throughout this work. I also want to thank my committee members Prof. James A. Bain, Prof. Larry Pileggi (chair) and Dr. Abhishek Sharma. I would like to acknowledge the financial support from SONIC, Inteland the Department of Electrical and Computer Engineering. Special thanks to Prof. Pileggi for funding the CMOS tapeout. I would also like to acknowledge Axel Berny Graduate Fellowship and Ben Cook Graduate Fellowship.

I would like to thank my colleagues that I collaborated with. Abhishek Sharma, Darshil Gala, Thomas Jackson and Greg Slovin. Special thanks to Abhishek for being a great mentor in the earlier days of my studies and for serving as a committee member later on. I owe many thanks to Darshil for his friendship in and out of the laboratory. I would also like to thank Xiaoliang Li and Susnata Mondal for their help with the CMOS IC design.

I would like to thank members of the Weldon Group: Ece Isenbike Ozalp, Sungho Kim, Mohamed Darwish and Omer Ozdemir. Many thanks to Ece for being a close friend, all the great food she cooked and the fun times I spent with her two dogs, Taco and Yogi.

I would like to thank my friends Dr. Gurkan Bektas, Emre Gokce, Dr. Melik Kayikci and Dr. Selim Turan for their support that I felt across thousands of miles. Special thanks to Gurkan for checking in so often to make sure I was doing alright.

I would like to thank Etkin Akkaya and Dr. Burak Erbagci for their close friendship. Special thanks to Burak for many lessons I learned from him and to Etkin for his detail oriented reminders about things that I overlooked.

I would like to thank Dr. Oguz Cetin and Dr. Huseyin Gurkan, the duo that I started the PhD journey with back in Bilkent days. I am sure they will be great scholars and many

more people will be learning from them.

Finally, I would like to thank my parents and my brother for always supporting me. This work is dedicated to them.

# Abstract

CMOS technology and the aggressive roadmap outlined by the Moore's law enabled today's powerful computers and smartphones. Traditional scaling of CMOS technology is reaching an end due to a number of factors including cost and fundamental physical limits such as increased leakage current and high stochastic variation. In the meantime, the nature of computation also changed. The exchange and processing of graphical data increased, big data and IoT arrived. These more data-centric applications exacerbate the power consumption due to intense memory access required in the traditional von Neuman computation scheme. Therefore, to overcome these problems we need not only new devices or computation schemes; but a co-design of alternative compute paradigms and the device fabric that will enable them.

This work proposes using coupled oscillator networks for computation. These systems mimic the parallelism of the brain to overcome the von Neumann bottleneck. The device fabric we explore is the S-type negative differential resistance (S-NDR) oscillator, whose unique device properties lends itself to area and power efficient, BEOL compatible dense arrays. We explore the device-circuit relations of S-NDR devices, develop nano-oscillators and integrate them with the existing scaled CMOS technology. Using a compact circuit model and SPICE simulations, we show that capacitively coupled networks of S-NDR oscillators can solve image processing problems such as edge detection, stereo vision and image segmentation.

To verify the capabilities of oscillator networks on hardware, we design and tape-out a CMOS oscillator network. We analyze the capacitive coupling scheme of oscillators and extend Kuramoto's model to capture the properties of capacitive coupling. We finally demonstrate an image segmentation engine utilizing the oscillator network.

*...if you fail to attain self-knowledge,*

*what good is there in your studies?*

Yunus Emre (13th cent.)

*to my family...*

# Contents

**1 Introduction** **1**

1.1 S-type Negative Differential Resistance Oscillators . . . . . 4

1.2 Coupled Oscillator Networks . . . . . 5

1.3 Thesis Organization and Contributions . . . . . 6

**2 Oscillator Dynamics** **8**

2.1 First Order Model . . . . . 8

2.2 On the Stability of the Operating Points . . . . . 12

2.3 Oscillations in a First Order System . . . . . 18

2.4 Second Order Oscillator Systems . . . . . 20

2.5 Van der Pol Oscillator Using S-NDR Device . . . . . 25

**3 Developing S-NDR Nano-oscillators** **32**

3.1 Device Characterization . . . . . 33

3.2 Transistor Ballast for a Voltage-Controlled-Oscillator . . . . . 36

3.3 Integration with Scaled CMOS Transistors . . . . . 38

3.4 Oscillator Modelling . . . . . 39

3.5 Cross-coupled Transistor Pair as a Ballast . . . . . 41

3.6 Scaling the S-NDR Oscillators . . . . . 47

**4 Coupled Oscillator Networks for Computation** **49**

4.1 Kuramoto model . . . . . 50

4.2 Capacitively Coupled S-NDR Oscillators . . . . . 54

4.3 Forced Oscillations and Injection Locking . . . . . 57

4.4 Phase Based Computation . . . . . 59

4.5 Coupled Oscillator Networks . . . . . 63

4.6 Edge Detection . . . . . 64

4.7 Stereo Vision . . . . . 68

4.8 Scaling and Integration Projection . . . . . 71

**5 An all-CMOS Oscillator Network** **73**

5.1 Chip Overview . . . . . 74

5.2 Oscillator Network . . . . . 76

5.3 Chip Testing . . . . . 90

5.4 Single Oscillators . . . . . 92

5.5 Pairwise Coupled Oscillators . . . . . 94

5.6 Network Level Coupling . . . . . 98

5.7 Discussion . . . . . 105

**6 Conclusion and Suggestions for Future Research** **108**

**Appendix A CMOS Integration Fab Flow** **111**

**Appendix B Coupled Oscillator Network Image Segmentation Examples** **114**

**Bibliography** **119**

# List of Figures

1.1 Emerging computing architectures as outlined in International Roadmap for Devices and Systems White Paper [3] . . . . . 2

1.2 Image segmentation is a critical aspect of autonomous car technology. This is how the autonomous car *sees* the other cars, pedestrians, traffic signs and other objects around it. Such an application requires frequent read/write operations from/to memory, leading to long idle times for the processor core and high energy overheads. A power hungry computation scheme can reduce the fuel efficiency of an autonomous car by 11.5% [5]. Image adapted from [7]. . . . . 3

1.3 Two terminal S-NDR devices (a) show a bi-stable current-voltage relation. There are 2 stable states in the device I-V and an unstable state (b), which can be chosen as the region of operation using a resistor in series (c). . . . . 5

1.4 Schematic of the directly-coupled oscillator network for which the coupling element is a capacitor. Each oscillator is coupled to its 4 nearest neighbors. Only a portion of the network is shown. . . . . 6

2.1 (a) Ballast resistor ($R_s$) in series with the S-NDR device. The device voltage ($V_{out}$) will be the oscillator output throughout this thesis. (b) Current-voltage relationship of S-NDR devices overlaid with the ballast resistor's loadline. Depending on he resistance, different regions of operation can be chosen. (c) Piecewise linear, switching resistor model for the S-NDR device. (d) First order S-NDR oscillator model. . . . . 9

2.2 (a), (c), (e) shows the device IV overlaid with the ballast resistor loadline. (b), (d), (f) shows the RC charging/discharging behavior of the parasitic capacitance ($C_p$) for (a), (c), (e), respectively. Biasing the device in the NDR region (e) leads to oscillations (f). . . . . 11

2.3 The differential equation $\dot{x} = f(x)$ governs the time evolution of variable $x$ i.e. its flow on the x-axis. For $\dot{x} > 0$, $x$ grows and moves right on the x-axis. For $\dot{x} < 0$, $x$ shrinks and moves left. When $\dot{x} = 0$, $x$ doesn't change over time. These fixed points are marked with circles: an empty circle for the unstable fixed point and a filled circle for the stable fixed point. . . . . 13

2.4 Reduced first order circuit model for the OFF-state and the ON-state. . . . . 15

2.5 Plot of Equation 2.10 for $V_{DD} = 3V, R_s = 0.5M, R_d = 1M, C = 100p$. Note that the fixed point occurs at $V = 2$. This is the DC operating point of the circuit i.e. $V = V_{DD} \times R_d/(R_s + R_d)$ . . . . . 16

2.6 (a) Reduced first order circuit model and (b) the loadline for the NDR region. For the loadline to cross the NDR region, its slope should be larger than the NDR region's i.e. $-1/R_s > -m$. . . . . 17

2.7 Plot of Equation 2.12. The parameters are selected such that $V_{th} = 3$ and $V_h = 1$. Any point biased in the NDR region will move either towards $V_{th} or V_h$ and cause a switching. . . . . 18

2.8 Regions of operation . . . . . 19

2.9 Phase is a state variable that flows on a circle. . . . . 20

2.10 Harmonic oscillators (a) Simple harmonic oscillator i.e. nonzero damping (b) A positive damping causes the oscillations to die out. The source of damping can be the resistor for an LCR oscillator or friction for a mass-spring system. (c) A negative damping causes oscillation amplitude to grow to infinity. . . . . 22

2.11 Van der Pol oscillators recreated after [18] for (a) $\epsilon = 0.1$, (b) $\epsilon = 1$ and (c) $\epsilon = 10$. The nonlinear damping causes the amplitude to grow and saturate. The oscillations resemble a sinusoid for small $\epsilon$ and becomes a relaxation oscillation for large $\epsilon$. . . . . 24

2.12 (a) A third-order polynomial approximation of S-NDR IV. (b) The *primary* circuit as described by Shaw in [20]. The cubic IV in this circuit corresponds to the van der Pol's oscillator. . . . . 25

2.13 (a) Resistor ballasted S-NDR oscillator configuration and (b) the first order model considered earlier. In these configurations, there is no inductor. (c) Second order model with the inductor. . . . . 28

2.14 S-NDR device IV (blue) and the oscillation trajectory (red). The first order model without the inductor comes short in explaining the transitions between the OFF and ON states. During the transitions, there is a *gap* between the device IV and the oscillation trajectory. This $V_{gap}$ is captured by the inductor in the second order model. Physically, it captures the dynamics of the filamentation. . . . . 30

3.1 (a) Cross-sectional diagram of S-NDR devices. (b) Top view diagram of devices arranged in the crossbar configuration. (c) Microscope image of fabricated devices. . . . . 33

3.2 (a) DC characterization of the device is achieved with a resistor in series. (b) DC IV of the $GeTe_6$ devices pictured earlier. . . . . 34

3.3 (a) Dependence of the threshold voltage ($V_{th}$) on the device thickness. (b) Effect of contact resistance on the holding voltage ($V_h$). Resistive voltage division across the device reduces the potential across the filament ($V_f$) compared to $V_h$. . . . . 35

3.4 (a) Current (blue) and voltage (blue) oscillations observed in $GeTe_6$ devices. (b) Supply voltage dependence of the oscillation frequency in the resistor ballast configuration. Increasing the supply reduces the time required to charge the parasitic capacitance seen at the output hence the frequency increases. Frequency increases more steeply with supply when we remove the voltage probe which normally adds additional parasitic capacitance. . . 36

3.5 (a) A cartoon of device IV overlaid with a transistor's loadline in the transistor ballasted (1T1R) scheme shown in (b). Gate voltage of the transistor can be used to bias the device at various current levels within the NDR region and this configuration can be utilized as a compact voltage controlled oscillator. . . . . 37

3.6 (a) Current oscillations of the device when it is ballasted with a transistor operating in the sub-threshold region. (b) The dependency of oscillation frequency on the gate voltage of the transistor, $\Delta V_{GS} = |V_{GS}| - |V_{TH}|$ in the subthreshold region. The inset shows the transistor ballasted oscillator scheme. . . . . 38

3.7 SEM photo of a transistor on the 28nm CMOS chip. The terminals of the transistor are connected to metals going up to M7. On M7, there is a larger (10µm$\times$10µm) metal plate which contacts the pads of the fabricated device. 39

3.8 (a) Die photo showing S-NDR devices integrated with scaled CMOS transistors taped at 28nm process technology. The circuit schematic is overlaid for clarity. The pads are oriented to fit the SSG RF-probes we used for measurements. (b) Current and voltage waveforms of the oscillators. . . . . 40

3.9 (a) S-NDR element with reactive circuit components added to describe the AC behavior. The circuit on the left describes the behavior of a resistor connected in series with the S-NDR material stack. (b) To model DC behavior of the S-NDR device, a piecewise linear curve is fitted to the measured IV data. (c) The simulated and measured waveforms are in good agreement. . 42

3.10 (a) PMOS cross-coupled transistor pair. (b) Loadlines of a transistor and a cross-coupled pair overlaid with a piecewise-linear S-NDR device IV. . . . . 43

3.11 (a) Cross-coupled pair ballasted S-NDR oscillator scheme (b) Voltage waveforms at the output terminals. . . . . 44

3.12 A comparison of 1T1R and XCP-ballasted systems in terms of allowed W/L range and oscillation frequency. . . . . 45

3.13 PMOS cross-coupling of the two outputs enables higher frequencies compared to 1T1R system by facilitating (a) easier charging and (b) discharging of the outputs. The diagrams are drawn for the charging and discharging cycles of $V_1$. . . . . 46

3.14 (a) Cross-coupled pair ballasted VCO. In order to control the frequency, current injection into the devices should be tunable. The PMOS pair connected between the $V_{DD}$ and the sources of the XCP provides the control know to tune current. (b) Control Voltage vs. Frequency . . . . . 47

3.15 Oscillation waveforms of the fabricated XCP oscillator. While one of the outputs is oscillating (shown in red trace), the other one is not. This is due to the variation between the two S-NDR devices. In a more controlled process environment, the two devices should be more similar and oscillate similar to our simulations. . . . . 48

4.1 Phase is a state variable that flows on a circle. . . . . 50

4.2 (a) Solution of the Kuramoto equations for a coupled oscillator pair with natural frequencies $f_1^o = 50$ MHz and $f_2^o = 30$ MHz. The coupling strengths are $k_{i,j} = 2\pi \times 5$ MHz. The oscillators cannot synchronize because of large natural frequency mismatch. (b) Solution when $f_1^o = 50$ MHz and $f_2^o = 40$ MHz with the same coupling strengths. The two oscillators synchronize. (The differential equations are solved using 4$^{\text{th}}$-order Runge-Kutta algorithm on Matlab.) . . . . . 52

4.3 Synchronization behavior of Kuramoto oscillators. We solved the Kuramoto equation for values of $f_2^o$ between 30 MHz and 70 MHz while keeping $f_1^o = 50$ MHz. When detuning ($\Delta f^o \leq 10$ MHz, the two oscillators synchronize. This maximum allowed detuning is actually the sum of two coupling strengths $k_{i,j} = 2\pi \times 5$ MHz.) . . . . . 53

4.4 (a) S-NDR oscillator pair coupled with a capacitor. (b) The voltage oscillations of the coupled S-NDR oscillators show a 180° relative phase difference. Figures were adapted from [15]. . . . . 54

4.5 SPICE simulation of capacitively coupled S-NDR oscillator pair reveal that they couple in the same way Kuramoto oscillators do. . . . . 56

4.6 Sub-harmonic injection locking generates output signals ($\omega$) which oscillate at the half of the input AC signal's frequency ($2\omega$). This will happen only if $\omega \approx \omega_o$ where $\omega_o$ is the natural oscillation frequency of the oscillator, without the AC input. . . . . 57

4.7 (a) Sub-harmonic injection locking can be achieved by applying the AC signal to the ballast transistor's gate in the 1T1R configuration. (b) Experimental verification of SHIL in S-NDR oscillators. . . . . 58

4.8 The SPICE simulations of S-NDR oscillators confirm that SHIL results in two different phases at the output depending on the initial (pre-charged) voltage of the output node. The legends show these initial voltages. . . . . 58

4.9 (a) In traditional logic, boolean states are encoded in voltage levels. (b) In phase based logic paradigm, the states are encoded in the phase of an oscillating waveform. . . . . 59

4.10 a) MAJ gate design using $GeTe_6$ oscillator (b) The truth table. If the input A is 0, OUT = B AND C. For A = 1, OUT = B OR C. This demonstrates the functional completeness of {MAJ, NOT} . . . . . 60

4.11 (a) Full adder design. The XOR and AND gates were designed using MAJ gate. (b) resulting waveforms overlaid with logic 0 for reference. Input A is 0 for the whole time as opposed to signal B which is 1 for 25 ns and flips to 0 afterwards. Carry-in ($C_i$) input is 1. Output S is in phase with logic 0 for t $<$ 25 ns (S = 0) and out of phase with it for t $>$ 25 ns (S = 1, after a brief transient). Carry-out is out of phase with logic 0 and for t $<$ 25 ns ($C_o$ = 1) and in phase with it for t $>$ 25 ns ($C_o$ = 0, after the transient). . . . . 62

4.12 Schematic of the directly-coupled oscillator network for which the coupling element is a capacitor. Each oscillator is coupled to its 4 nearest neighbors. Only a portion of the network is shown. . . . . 63

4.13 (a) 14$\times$14 network used for the simulations. (b) shows the time evolution of the phases of the oscillators numbered 1-14 in (a). . . . . 64

4.14 (a) Graphical representation of the thresholding function. (b) Edge detection flowchart using coupled oscillator network. . . . . 66

4.15 (a) Input binary image (b) image is encoded in initial phases and (c) after the evaluation, the edges of the feature are detected. . . . . 66

4.16 Phase separation ($\Delta\phi$) and evaluation window ($\Delta t$) vs coupling capacitance ($C_c$). . . . . 67

4.17 The left and right images of the Tsukuba benchmark are used to create the disparity image (top right). The disparity image is fed into the network in the form of oscillation frequencies. The network settles to the disparity map (stereo map) shown on the bottom right. In this map, brighter colors represent higher frequency, thus are closer to the camera. Simulations designed by Abhishek A. Sharma [29]. . . . . 70

4.18 Comparison of stereo maps of (a) ground truth, (b) belief propagation using FPGA, and (c) oscillator network. The FPGA implementation [49] mispredicts the 8% of the pixels compared to the ground truth and the oscillator network's misprediction rate is 22%. Simulations designed by Abhishek A. Sharma [29]. . . . . 71

4.19 Simulated oscillation frequency and computation energy vs the parasitic capacitance, $C_p$ . . . . . 72

5.1 Synchronization behavior of two coupled CMOS ring oscillators, designed at a commercial 65n process technology and simulated in SPICE . . . . . 75

5.2 An overview of the chip layout. The dimension of our block is 1mm $\times$ 1mm. 77

5.3 Block diagram of a tile in the oscillator network. In each tile, there is one ring oscillator VCO, a DAC, transmission gates and capacitors that enable coupling, a buffer and 14 configuration scan flops. . . . . 78

5.4 Schematic of a 3-stage current-starved ring oscillator design with an enable NAND gate. When OSCEN signal is 0, the output of the NAND gate at at the first stage is always 1. This forces VOUT to be 1. If OSCEN is 1, the NAND gate acts like an inverter and hence the output oscillates. . . . . 80

5.5 Segmented DAC schematic. . . . . 83

5.6 The layout of the coupling elements. . . . . 85

5.7 Tile layout. . . . . 87

5.8 Read-out scheme of the oscillator network. 10 oscillators along each row are multiplexed (column select). Then 2 outputs from the 5 odd rows and 5 even rows are selected using 5-to-1 MUXes (row select). The two outputs (ODD_ROW and EVEN_ROW) are then buffered and connected to the IO pads. . . . . 88

5.9 Network layout. . . . . 89

5.10 Test setup. . . . . 90

5.11 A microphotograph of the chip. The dimensions are 1 mm$\times$1 mm. The wirebonded pads can be seen. The 6 pads in the upper left corner without wires are for probing purposes. They are connected to some of the test structures. . . . . 91

5.12 (a) The individual characterization of the 100 VCOs on chip reveal that there is a significant variation. Two oscillators (R:7 C:3 and R:8 C:3) are highlighted on purpose. These two oscillators are used for pairwise coupling later on. (b) Statistical distribution of the VCO frequency for DAC input $\langle 1111 \rangle$. . . . . 93

5.13 Chip measurement results showing the synchronization behavior of two coupled CMOS ring oscillators that were highlighted in Figure 5.12. Coupling capacitance was 50 fF. . . . . 95

5.14 Synchronization range as a function of the coupling capacitance. The linear dependence suggests that the coupling strength in capacitive coupling is proportional to the coupling capacitance ($C_c$). . . . . 96

5.15 Solution of the updated Kuramoto model in Equation 5.6 show that the synchronization range increases in proportion with the coupling capacitance. This result is in agreement with the experimental results we obtained earlier. 98

5.16 Synchronization of two coupled CMOS ring oscillators when coupling capacitance is 100 fF. The synchronization range not only widens, but its center also shifts to the left of the input of oscillator 1. In other words, it widens more to the left i.e. towards higher frequencies. This suggests that the coupling strength is a function of frequency as well. . . . . 99

5.17 Solution of the updated Kuramoto model (Equation 5.6) with $C_c = 100$ fF. The synchronization range widens into higher frequency ranges, very similar to what we have seen in our experimental results (Figure 5.16). . . . . 100

5.18 Image segmentation flow with the coupled oscillator network. The input image (a) and the network (b) has the same size. The input image is encoded into the network by mapping the pixel values of each pixel into natural oscillation frequencies. After the network settles (d) the oscillators in the same region of the image synchronize because of their similar natural frequencies. The output image (e) can be constructed by mapping the frequency information back to pixel values. . . . . 101

5.19 (a) Input image for image segmentation experiment using CMOS oscillator network. The numbers seen on the pixels corresponds to the natural oscillation frequencies corresponding to those pixels. (b) The output image after the network settles, the numbers show the oscillation frequencies after the network settles. The oscillators clustered into two groups corresponding to the regions in the input image. For this experiment, coupling strength was 70 fF. . . . . 104

5.20 An input image with 2 regions. Simple thresholding cannot classify the darker pixels within the lighter region and the lighter pixels in the darker region. . . . . 105

A.1 Fabrication process flow for the heterogeneous integration of S-NDR devices with the 28nm CMOS technology . . . . . 113

B.1 (a) Input image (b) Output image ($C_c$ = 50f) with noisy pixels. . . . . 116

B.2 (a) Output image ($C_c$ = 60f) (b) Output image ($C_c$ = 70f) . . . . . 117

B.3 (a) Input image with 3 regions (b) Output image segmented correctly . . . . 118

# List of Tables

5.1 A summary of the VDD domains on the chip. . . . . 76

5.2 Breakdown of the 14 configuration bits in each tile. . . . . 86

5.3 Comparison of S-NDR oscillator (this work and [16]) to CMOS oscillators. The numbers for 7 nm are generated using the Predictive Technology Model [59, 60]. The figures for the scaled S-NDR oscillator are based on our predictions discussed in Chapter 3. . . . . 107

# Chapter 1

# Introduction

CMOS technology and the aggressive roadmap outlined by the Moore's law enabled today's powerful computers and smartphones. Mere miniaturization of devices was initially enough to reduce the area and power requirements of transistors, yet at sub-100 nm technology nodes this was not enough. First metal gates and high-K oxides were introduced to improve the transistor performance [1]. This material level improvement eventually was reinforced by a structural change with the introduction of FinFETs. FinFETs allowed for the control of the channel from three sides. Recently 5nm gate-all-around transistors are announced, taking the channel control to the ultimate [2]. Nevertheless, despite the advancements in the transistor device and fabrication technologies, CMOS is facing physical barriers; we are approaching a limit where the transistor sizing is comparable to the size of a handful of atoms. This leads to significant leakage currents and lower yield due to higher variation. The problem translates to more power consumption and more expensive chips, totally reversing what Moore's law has been promising so far. At this point, a quest to find novel devices that will augment CMOS transistors started. Emerging memories (PCRAM, RRAM, STT-RAM) and new transistor technologies (TFET, CNTFET) are being studied to extend the 'good old-fashioned' Von Neumann computation architecture [3].

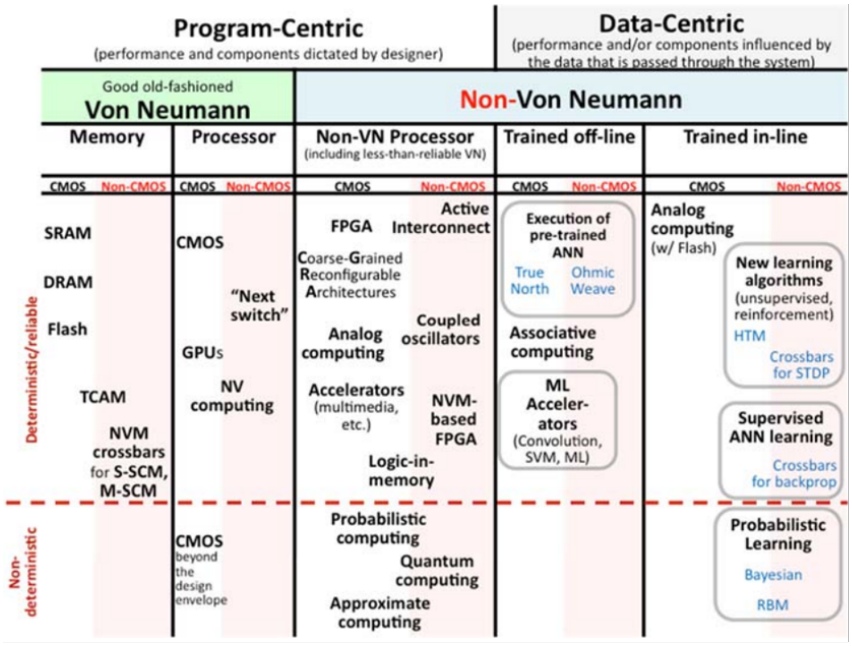



**Figure 1.1:** Emerging computing architectures as outlined in International Roadmap for Devices and Systems White Paper [3]

Device aspect of the research is aside, non-Von Neumann computation schemes are also being investigated to meet the requirements of data-centric applications. In the classical von Neuman computation scheme, data is fetched from the memory to the processor as it is processed. While data is being fetched from the memory, the processor core waits idle. The energy consumption of the active core can be as low as the 10% of total power consumption [4]. The majority of the power consumption arises from the idle waiting of thew core and the fetching of the data from the memory. This problem is exacerbated for more data centric applications such as image segmentation, which is at the core of the autonomous car technology. When the energy storage and the cooling overheads are taken into account, a power hungry computation scheme can reduce the fuel efficiency (or driving range) of an autonomous car by 11.5% [5]. Brain-inspired, non Von Neumann computation architectures such as neural networks started to draw more attention. Such systems mimic the parallelism of the brain. CMOS implementations of neural networks such as Google's Tensor Processing Unit offers 86$\times$ more computations per watt [6]. Even

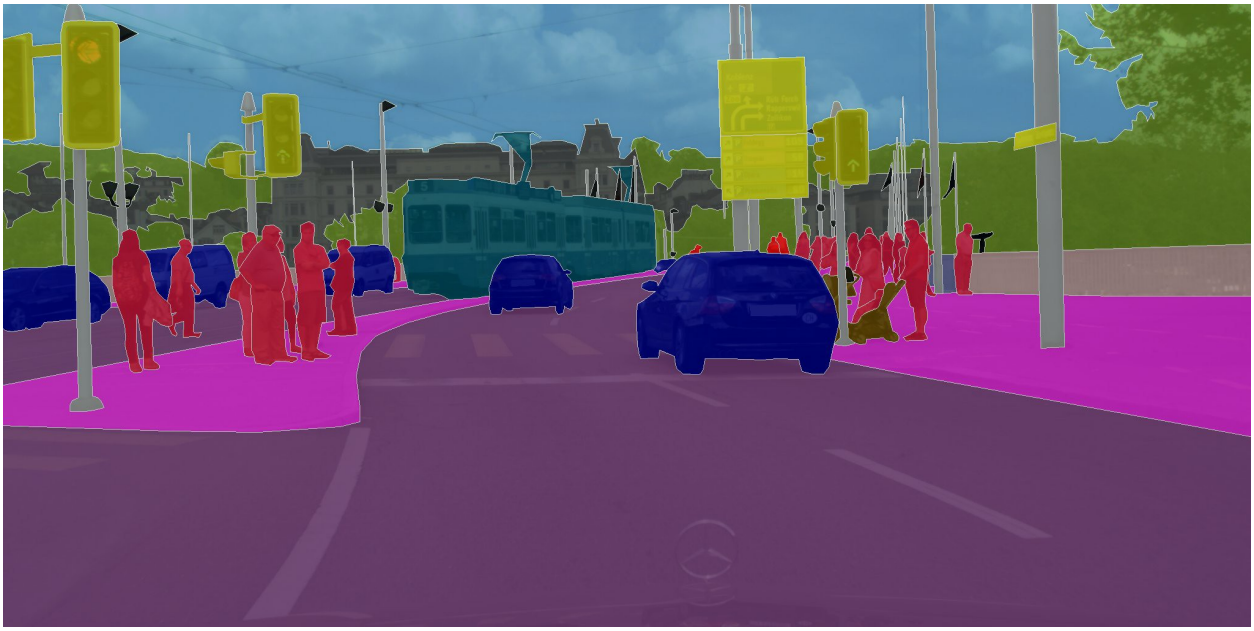


**Figure 1.2:** Image segmentation is a critical aspect of autonomous car technology. This is how the autonomous car *sees* the other cars, pedestrians, traffic signs and other objects around it. Such an application requires frequent read/write operations from/to memory, leading to long idle times for the processor core and high energy overheads. A power hungry computation scheme can reduce the fuel efficiency of an autonomous car by 11.5% [5]. Image adapted from [7].

though these systems are more power efficient compared to a CPU due to their architecture, the CMOS implementations will eventually face the problems that we described earlier.

Oscillator-based brain inspired neuromorphic computation systems such as oscillatory neural networks (ONNs) [8] and coupled oscillator networks [9] also emerged as strong candidates, especially for image processing applications. These are inspired by brains activity: neurons in human brain fire periodically [10] and synchronization of neurons might correspond to recognition of faces or spoken vowels [11, 12]. Typically, CMOS implementations of these systems are inefficient in terms of power and area. For example, oscillatory neural networks require VCOs and PLLs for their neurons [13], which are power and area hungry circuits. Moreover, a complex network of weighted synapses are required which further increases the chip area. Therefore, there is a need for: 1) dense networks that can be directly coupled, and 2) low power, scalable novel oscillators that will enable such net-

works.

The purpose of this work is exploring the unique attributes of S-type negative differential resistance (S-NDR) oscillators and directly-coupled oscillator networks to solve image processing problems. We will be discussing S-NDR oscillators, a class of oscillators that take advantage of fundamental physics of an unstable device. These devices are promising in terms of power consumption and smaller footprints. We will investigate the functionality of coupled oscillator networks. We aim to show that certain image processing tasks can be carried out using coupled oscillator networks. This approach can potentially offer better performance and scalability compared to CMOS. However, as we will discuss later in Chapter 6, there are still open research problems that needs to be solved before implementing a beyond-CMOS oscillator based hardware accelerator. Therefore, this research will serve as an investigation of the image processing abilities of coupled oscillator networks. For this purpose, we will first utilize compact circuit models of S-NDR devices and use SPICE to simulate the networks of oscillators. Finally, we will use CMOS oscillators as a test vehicle to verify the abilities of coupled oscillator networks in hardware.

## 1.1 S-type Negative Differential Resistance Oscillators

*S-type Negative Differential Resistance* (S-NDR) devices are two terminal devices (Figure 1.3 (a)) and named after their unique current-voltage properties. Figure 1.3 (b) shows the current-voltage relationship of an S-NDR device. Essentially, there are three branches in the curve and the device travels between these branches depending on the way it is biased; initially, the device is in the highly resistive OFF state. When the voltage drop exceeds a certain threshold voltage ($V_{th}$), a conductive filament is formed and the device switches to the low resistance ON state. A certain minimum current is required to sustain the filament, therefore, if the voltage in the ON state is below the corresponding holding voltage ($V_h$),

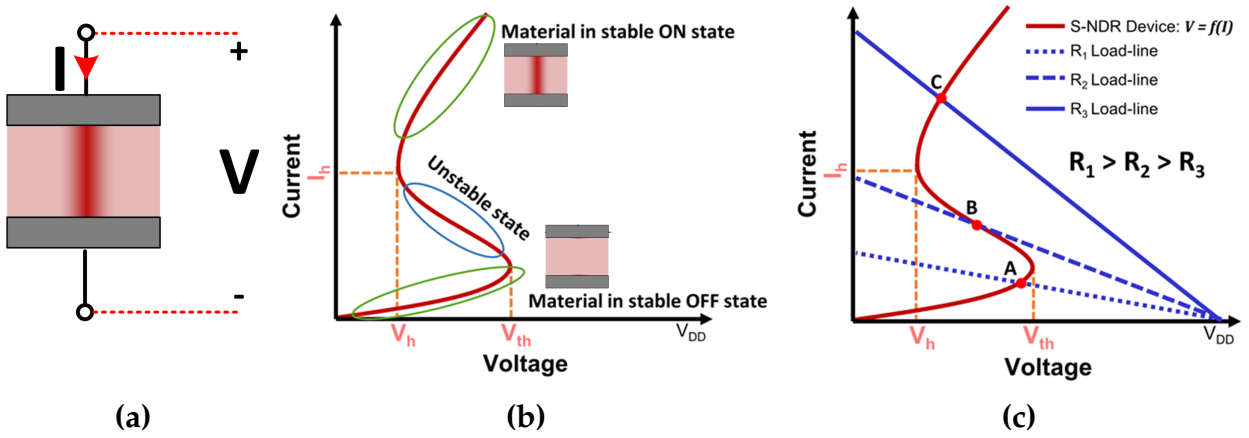



**Figure 1.3:** Two terminal S-NDR devices (a) show a bi-stable current-voltage relation. There are 2 stable states in the device I-V and an unstable state (b), which can be chosen as the region of operation using a resistor in series (c).

the device reverts back to the high resistance state. The switching from the OFF-state to the ON-state results in an unstable negative differential resistance (NDR) region. Therefore, the outline of the I-V graph roughly resembles an *S* letter, hence the name *S-NDR*.

## 1.2 Coupled Oscillator Networks

As it was previously mentioned, using simple elements for direct coupling among neurons is of importance in order to create dense networks that are efficient in terms of power and area. Inspired by the synchronization behavior of oscillators found in nature (such as human brain), we are proposing the nearest-neighbor capacitively-coupled oscillator network that is shown on Figure 1.4. Although pair-wise coupled S-NDR oscillators have been studied in the past [14,15], in this work we study a larger oscillator array where each oscillator is connected to its 4 nearest neighbors.

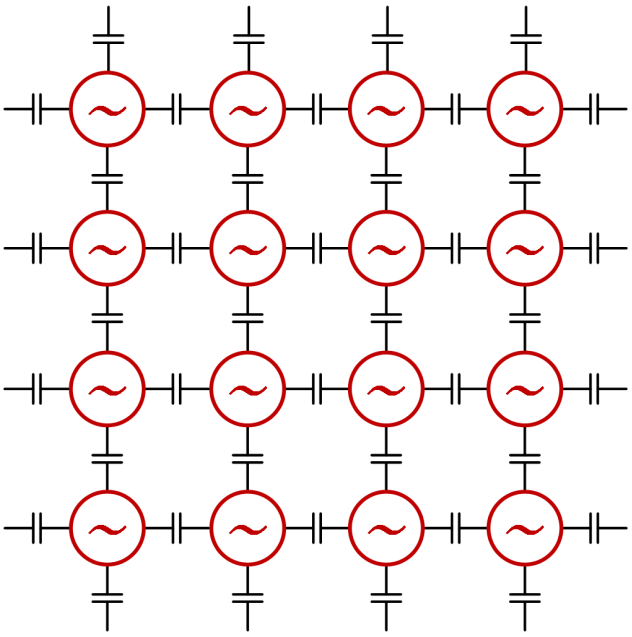


**Figure 1.4:** Schematic of the directly-coupled oscillator network for which the coupling element is a capacitor. Each oscillator is coupled to its 4 nearest neighbors. Only a portion of the network is shown.

## 1.3 Thesis Organization and Contributions

This research investigates the potential of capacitively coupled networks of S-NDR oscillators for image processing applications. Including this brief introduction chapter, there are 5 more chapters.

Chapter 2 provides a background on oscillator dynamics and discusses stability of S-NDR devices. This discussion establishes the biasing requirements for the devices to oscillate. Finally, we discuss how the oscillations can be modeled as relaxation oscillations within van der Pol's framework.

Chapter 3 presents our contributions at the device level: building onto Dr. Abhishek Sharma's exquisite work [16], we demonstrate the heteregenous integration of S-NDR oscillators with scaled CMOS technology. This step is crucial towards implementing systems where conventional CMOS is augmented by application specific beyond-CMOS compu-

tation engines. We also develop a compact model for S-NDR oscillators to explore device-circuit relations and system level functionality. We use the model to develop a novel biasing mechanisms, cross-coupled transistor ballast. This topology provides higher oscillation frequencies, design flexibility and robustness against variations.

Chapter 4 begins with a discussion of the dynamics of coupled oscillators. Using the oscillator model we built earlier, we demonstrate various coupled oscillator computation schemes. We show that edge detection and stereo vision (image segmentation) problems can be solved using capacitively coupled oscillator networks. We also show that phase based boolean computation is also achievable with S-NDR oscillators.

Because of limited fabrication capabilities, hardware demonstration of coupled oscillator networks we discuss in Chapter 4 is difficult. Therefore, in Chapter 5, we design a CMOS oscillator network. CMOS oscillators are chosen as a test vehicle to demonstrate the coupled oscillator network concept. To the best of our knowledge, this network is the first ever hardware demonstration of the computational capability of coupled oscillator networks. Analyzing our chip results, we extend Kuramoto's model for the capacitively coupled case and show that the coupling strength is a function of coupling capacitance and the oscillation frequency. We experimentally verify that image segmentation can be achieved using a network of capacitively coupled oscillators. The chapter concludes with a comparison of S-NDR oscillators to CMOS oscillators.

Chapter 6 discusses future research directions. Our experience of designing a CMOS oscillator network helped us understood that other than the oscillator and the coupling capacitors, there is a need for compact digital-to-analog converters and/or analog memory devices. Chapter 6 concludes with a summary of our work.

# Chapter 2

# Oscillator Dynamics

In the previous chapter, we stated that based on the ballast resistor's value, it is possible to bias the S-NDR device in the three branches, namely the OFF-state, the NDR region, and the ON-state. We further mentioned that the operating points on the ON-state and the OFF-state are stable, and an operating point in the NDR region is a unstable one which will lead to oscillations. In this chapter, we will be discussing the oscillation dynamics, stability of operating points and develop a descriptive circuit model for the oscillations observed with the S-NDR devices. We will begin with a discussion of a first order model to understand the basics of the oscillations.

## 2.1 First Order Model

Consider the diagram shown on Figure 2.1 (a) where a ballast resistor is connected in series with the S-NDR device. The current-voltage graph of the device overlaid with the loadline of the resistor is shown on Figure 2.1 (b) for different values of the ballast resistor $R_s$, $R_1 > R_2 > R_3$. For simplicity sake, we will approximate the DC IV behavior of the

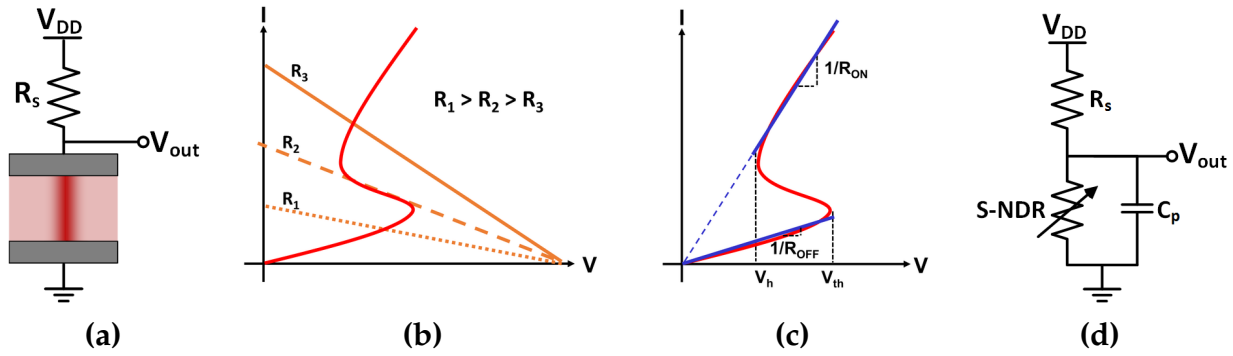



**Figure 2.1:** (a) Ballast resistor ($R_s$) in series with the S-NDR device. The device voltage ($V_{out}$) will be the oscillator output throughout this thesis. (b) Current-voltage relationship of S-NDR devices overlaid with the ballast resistor's loadline. Depending on he resistance, different regions of operation can be chosen. (c) Piecewise linear, switching resistor model for the S-NDR device. (d) First order S-NDR oscillator model.

S-NDR device with a switching resistor model as shown on Figure 2.1 (c). Furthermore, to capture the AC behavior and to represent the parasitic capacitance at the output node, we include a capacitor in parallel with the switching resistor. Therefore, we end up with the model shown on Figure 2.1 (d).

When $R_s = R_1$, the system can be reduced to Figure 2.2 (a). In this case, the loadline of the resistor meets the devices IV curve at operating point $A$. Assuming the initial value of $V_{out} = 0$, the capacitor at the output will start slowly charging with a time constant

$$\tau_{ch} = (R_{OFF} \parallel R_s) \times C_p \tag{2.1}$$

towards a final voltage of

$$V_{f-ch} = \frac{R_{OFF}}{R_{OFF} + R_s} \times V_{DD} \tag{2.2}$$

which is less than the threshold voltage of the S-NDR device (Figure 2.2 (b)). Therefore, the device will stay in the stable OFF state.

When $R_s = R_3$, the output node will start charging with the same time constant in Equation 2.1 towards the same final voltage in Equation 2.2. This is shown with the green

trace on Figure 2.2 (d). However this time, because of the smaller value of the ballast resistor, the voltage drop on the device will be larger leading to a final voltage that is higher than the threshold voltage of the S-NDR device. As soon as the output voltage reaches the threshold voltage of the S-NDR device, the device will switch to the ON-state. Next, the low ON-state resistance of the device will lead to a discharge of the parasitic capacitance following a time constant

$$\tau_{dch} = (R_{ON} \parallel R_s) \times C_p \tag{2.3}$$

towards

$$V_{f-dch} = \frac{R_{ON}}{R_{ON} + R_s} \times V_{DD} \tag{2.4}$$

Because this final value of the device is larger than the holding voltage, the device will stay in the stable ON-state. The purple trace on Figure 2.2 (d) depicts this discharge behavior.

When $R_s = R_2$, the loadline of the resistor meets the device IV in the NDR branch. In this case, unlike the previous one, the final voltage that the parasitic capacitance charges towards, $V_{f-dch}$ that is less than the holding voltage ($V_h$) of the S-NDR device. As soon as the output voltage becomes less than $V_h$, the device switches back to the OFF-state. Next the device will start charging again with device voltage reaching the ${}_V th$, switching to On-state and start discharging till it switches back to the OFF-state. This behavior will repeat itself leading to oscillations at the output node. Based on this charging-discharging behavior, we can express the time it takes to charge the output from $V_h$ to $V_{th}$ and to discharge from $V_{th}$ to $V_h$ as follows:

$$t_{ch} = -\tau_{ch} \times \ln\left(\frac{V_{th} - V_{f-ch}}{V_h - V_{f-ch}}\right) \tag{2.5}$$

$$t_{dch} = -\tau_{dch} \times \ln\left(\frac{V_h - V_{f-dch}}{V_{th} - V_{f-dch}}\right) \tag{2.6}$$

where $t_{ch}$ is the charging time and $t_{dch}$ is the time to discharge. The parameters are as they

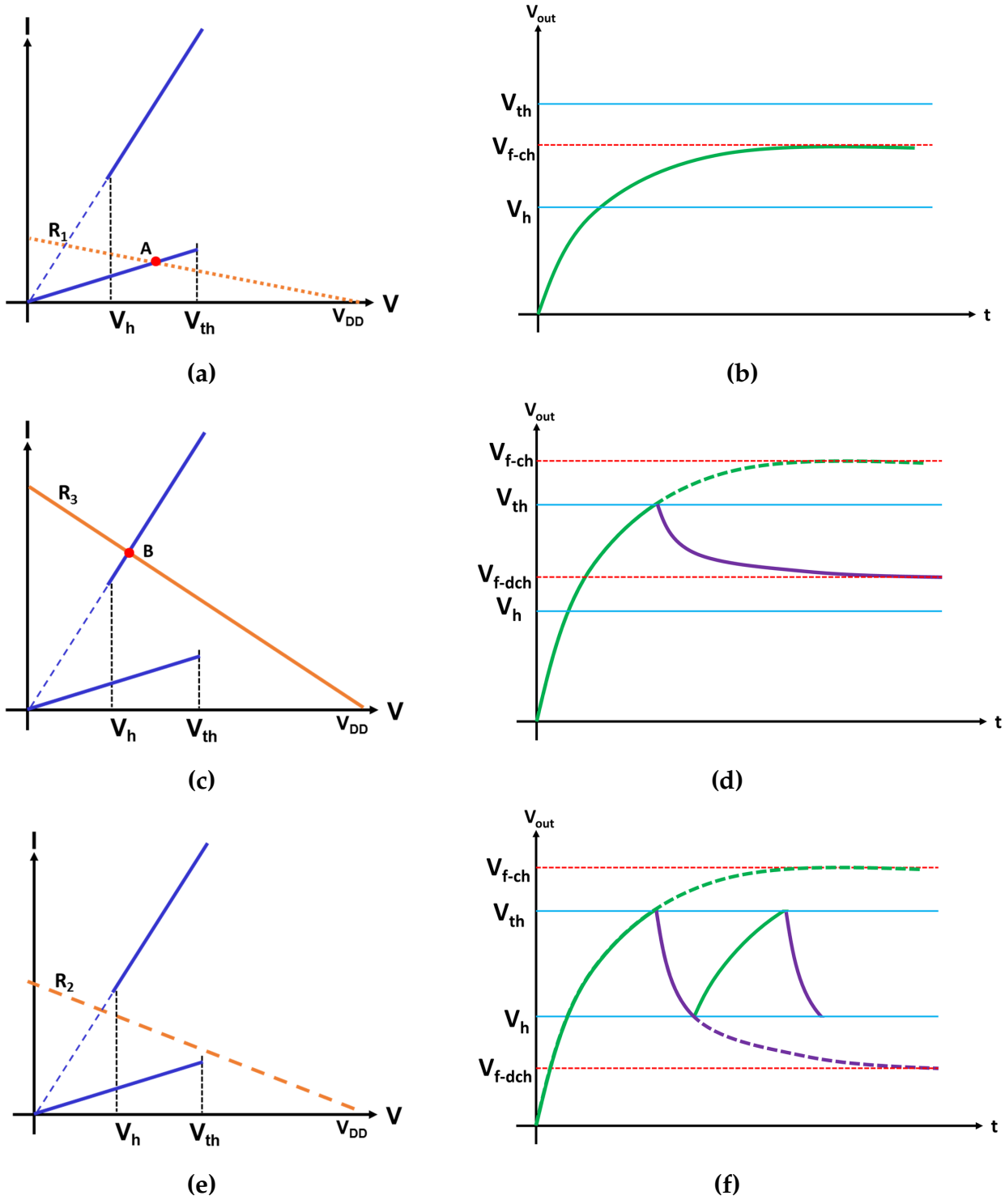



**Figure 2.2:** (a), (c), (e) shows the device IV overlaid with the ballast resistor loadline. (b), (d), (f) shows the RC charging/discharging behavior of the parasitic capacitance ($C_p$) for (a), (c), (e), respectively. Biasing the device in the NDR region (e) leads to oscillations (f).

were defined in Equations 2.1 - 2.4. Finally, the period of the oscillations can be calculated as:

$$T = t_{ch} + t_{dch} \tag{2.7}$$

In S-NDR oscillators, generally the OFF-state resistance is far greater than the ON-state resistance leading to $\tau_{ch} \gg \tau_{dch}$. For most practical bias conditions, this will lead to slow charging and quick discharging of the output node. Further, because both charging and discharging time constants are proportional to the parasitic capacitance seen at the output node, the frequency of the oscillations is inversely proportional to the parasitic capacitance.

Although rather intuitive this first order analysis is, it is only appropriate to establish a better understanding of the stability of the different branches of the IV curve of the S-NDR device. In the next section we will tackle this.

## 2.2 On the Stability of the Operating Points

In electrical engineering we often use pole-zero analysis to check the stability of circuits. However, in this work we will follow a geometric approach which provides a very simple yet elegant understanding of the dynamics.

We have seen in the previous section that the output voltage follows a time dependent RC charging-discharging behavior. This behavior can be expressed using a first order differential equation. Time dependency in any first order system can be expressed using a differential equations such as

$$\dot{x} = f(x) \tag{2.8}$$

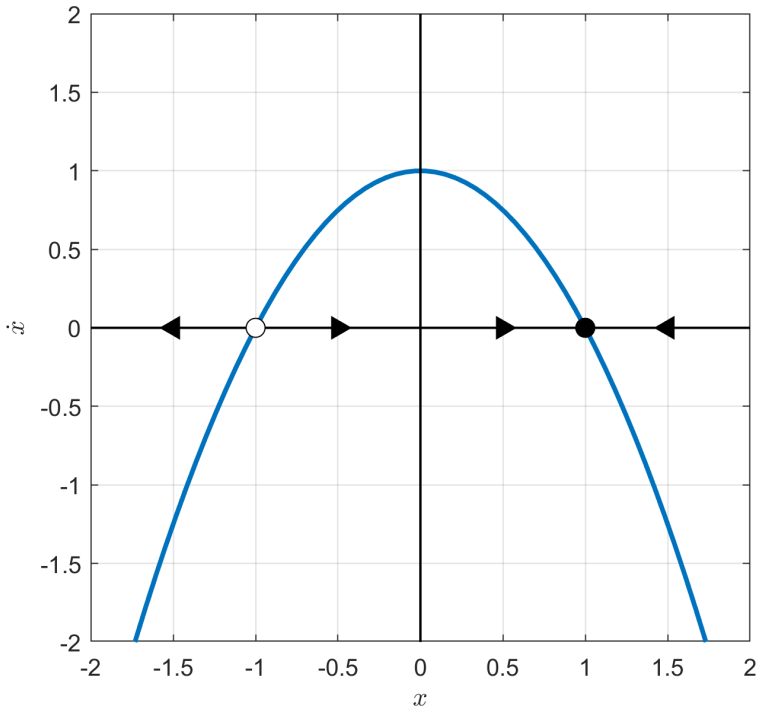



**Figure 2.3:** The differential equation $\dot{x} = f(x)$ governs the time evolution of variable $x$ i.e. its flow on the x-axis. For $\dot{x} > 0$, $x$ grows and moves right on the x-axis. For $\dot{x} < 0$, $x$ shrinks and moves left. When $\dot{x} = 0$, $x$ doesn't change over time. These fixed points are marked with circles: an empty circle for the unstable fixed point and a filled circle for the stable fixed point.

where

$$\dot{x} = \frac{\partial x}{\partial t} \tag{2.9}$$

In a first order differential equation like Equation 2.8, there is only the first order time derivative of the state variable $x$. Further, there is no time-dependent term on the RHS of the equation. Let's define an $\dot{x} = -x^2 + 0.5$ and plot it: When we look at Figure 2.3, we see that $\dot{x}$ is positive for $-1 < x < 1$ and it is negative for $x < -1$ and $x > 1$. $\dot{x} = 0$ at $x = -1$ and $x = 1$.

Physical meaning of $\dot{x}$ is its rate of change with respect to time. At any point on the x-axis, if $\dot{x} > 0$, it means that $x$ will be growing over time and the point will be moving right along the x-axis. This is the case for $-1 < x < 1$ for our case, hence the right-arrows on Figure 2.3. For a point where $\dot{x} < 0$, $x$ will be reducing with time and it will be moving

left along the x-axis. This is shown by the left-arrows on Figure 2.3 for $x < -1$ and $x > 1$.

For $x = -1$ and $x = 1$, $\dot{x} = 0$. If a system at a time exactly at one of these, it will stay there because the rate of change with time is zero. Such points are referred to as *fixed points*.

However, there is a difference between these two points. Looking at $x = 1$, we see that any point on the left or right of it will be moving towards the fixed point and eventually reach it and stay there. Note the two arrows facing towards $x = 1$ on the left and right of it. Even if any small perturbation moves the operation away from the fixed point, it will eventually be restored back to the fixed point. Such points are referred to as *stable operating points* because it is restored in the case of a perturbation. We mark the stable points with a filled dot.

$x = -1$ is different. The two arrows on its right and left are facing away from it. The points on its left will be shrinking towards $-\infty$ and the points to its right will be growing towards the fixed point at $x = 1$. If for $t = 0$, $x = -1$; it will stay there. However, any perturbation, such as noise, will not be restored and $x$ will move in the direction of the arrows. Points like this one are referred to as *unstable fixed points* since there is nothing that will restore it back.

In any real system, including electric circuits, perturbations are unavoidable due to noise. Therefore, unstable fixed points are not observed under normal operating conditions.

Having established this method, we can move on to analyzing the stability of the branches of the S-NDR device.

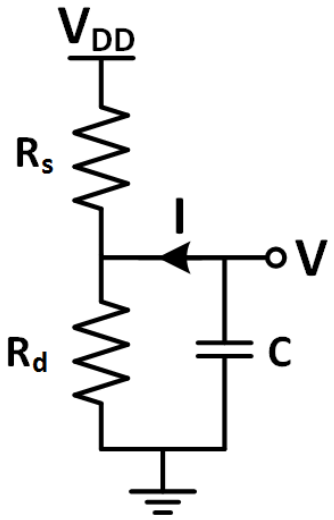



**Figure 2.4:** Reduced first order circuit model for the OFF-state and the ON-state.

### 2.2.1 Stability in the OFF and the ON states

The OFF and ON states are similar in the sense that in both cases the behavior of the S-NDR device resembles that of a resistor within the region that they are defined. Therefore, the system can be analyzed as an RC circuit as shown below. We will use the capacitor voltage, in this case named $V$ as the state variable. With a basic circuit analysis, we reach to the differential equation shown in Equiation 2.10:

$$\dot{V} = \frac{V_{DD}}{CR_s} - V \times \frac{1}{C \times (R_s \parallel R_d)} \tag{2.10}$$

Similar to the analysis that we conducted in the previous section, now we plot $\dot{V}$ as a function of $V$:

As we see in Figure 2.5, there is one fixed point in this system and it is a stable one. Therefore, if the device is biased in the OFF or the ON-state, no matter what its initial condition was, it will restore to this fixed point.

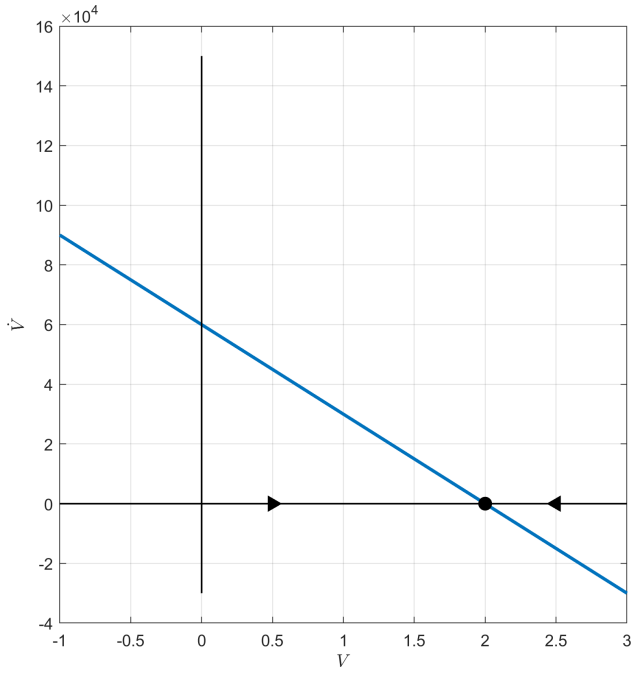



**Figure 2.5:** Plot of Equation 2.10 for $V_{DD} = 3V, R_s = 0.5M, R_d = 1M, C = 100p$. Note that the fixed point occurs at $V = 2$. This is the DC operating point of the circuit i.e. $V = V_{DD} \times R_d/(R_s + R_d)$

### 2.2.2 Stability in the NDR region

In the NDR region, the reduced circuit model looks like the one shown on Figure 2.6 (a) where the current-voltage behavior of the NDR element is given by the solid red line shown on Figure 2.6 (b). Note that this behavior is defined for $V_h < V < V_{th}$. This line is represented by the equation below:

$$I_d = -mV + n \tag{2.11}$$

where $I_d$ is the device current and $m$ and $n$ are positive constants which can be expressed in terms of $V_h$, $V_{th}$ and the device currents corresponding to these voltages. However this is not needed for the stability analysis.

Using circuit analysis, we reach at the following governing differential equation:

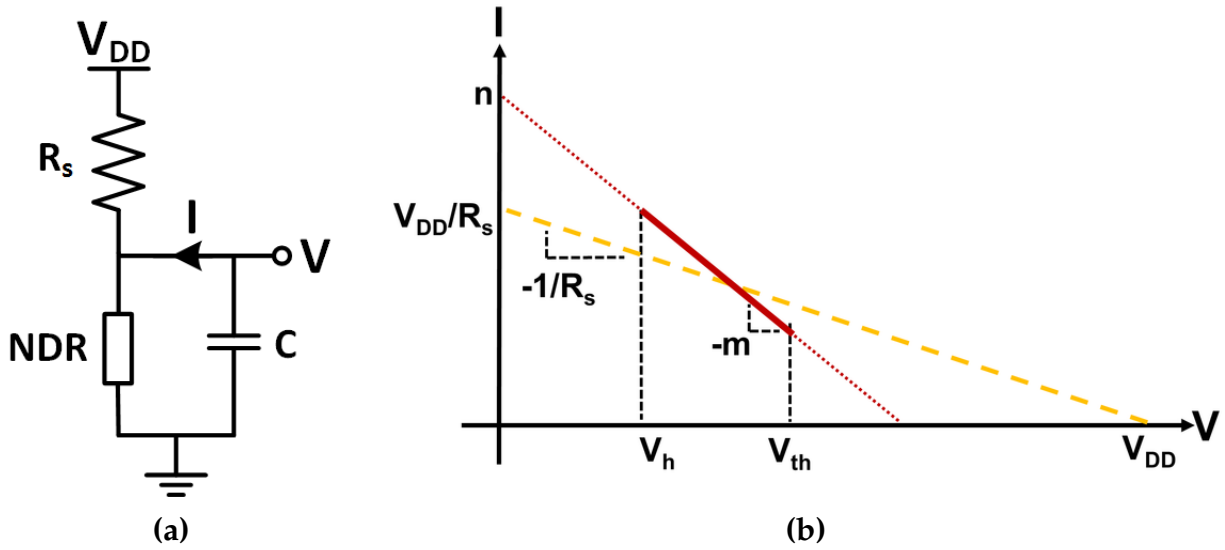



**Figure 2.6:** (a) Reduced first order circuit model and (b) the loadline for the NDR region. For the loadline to cross the NDR region, its slope should be larger than the NDR region's i.e. $-1/R_s > -m$.

$$\dot{V} = \frac{V_{DD}}{CR_s} - \frac{n}{C} + V \times \frac{1}{C} \times \left( m - \frac{1}{R_s} \right) \tag{2.12}$$

By definition, $n > V_{DD}/R_s$ and $m > 1/R_s$. The latter is a geometric implication of the loadline of the ballast resistor intersecting the NDR region. These can also be seen on Figure 2.6 (b). The line that corresponds to this equation is drawn below:

Because $m > 1/R_s$, the slope of the line in Figure 2.7 is positive. Even though there is a fixed point at $V = 2$, the positive slope leads to $V$ growing towards $V_{th}$ on the right of it and shrinking to $V_h$ on the left, making this an unstable point.

With this discussion, we have mathematically shown that biasing the S-NDR device in the OFF-state or the ON-state of its IV will lead to stable operating points. However, biasing it in the NDR region will lead to an unstable operating point and this is a required condition for oscillations in this system. Figure 2.8 summarizes the regions of operation: For a given supply voltage $V_{DD}$, only resistors with loadlines in the red-shaded region

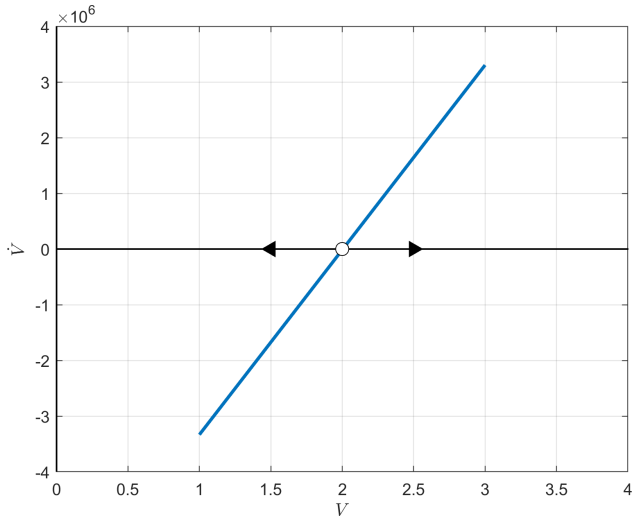



**Figure 2.7:** Plot of Equation 2.12. The parameters are selected such that $V_{th} = 3$ and $V_h = 1$. Any point biased in the NDR region will move either towards $V_{th} or V_h$ and cause a switching.

will lead to oscillations. Otherwise, the device will stay either in the OFF-state or in the ON-state.

## 2.3 Oscillations in a First Order System

Even though our intuitive discussion at the beginning of this chapter suggested that a first-order system could oscillate, a closer look at the dynamics reveal that it is actually impossible for a first-order system to oscillate if the state variable is defined on the real axis; such as current, voltage or charge in an electrical circuit. We have seen in Figures 2.3, 2.5 and 2.7 that there are fixed points -either stable or unstable; and the flow on the line as Strogatz puts it in [17], either converges to a stable fixed point or diverges to $\pm\infty$. For an oscillation to happen, the flow should change directions, but in a first order system there is always a fixed point between regions where flows is in the opposite direction. Hence, the fixed points continue to govern the operation. Despite this fundamental flaw, all the stability analyses and the calculations we made so far in this chapter are still accurate. In our analyses, all of our conclusions are local and accurate for the region that they are

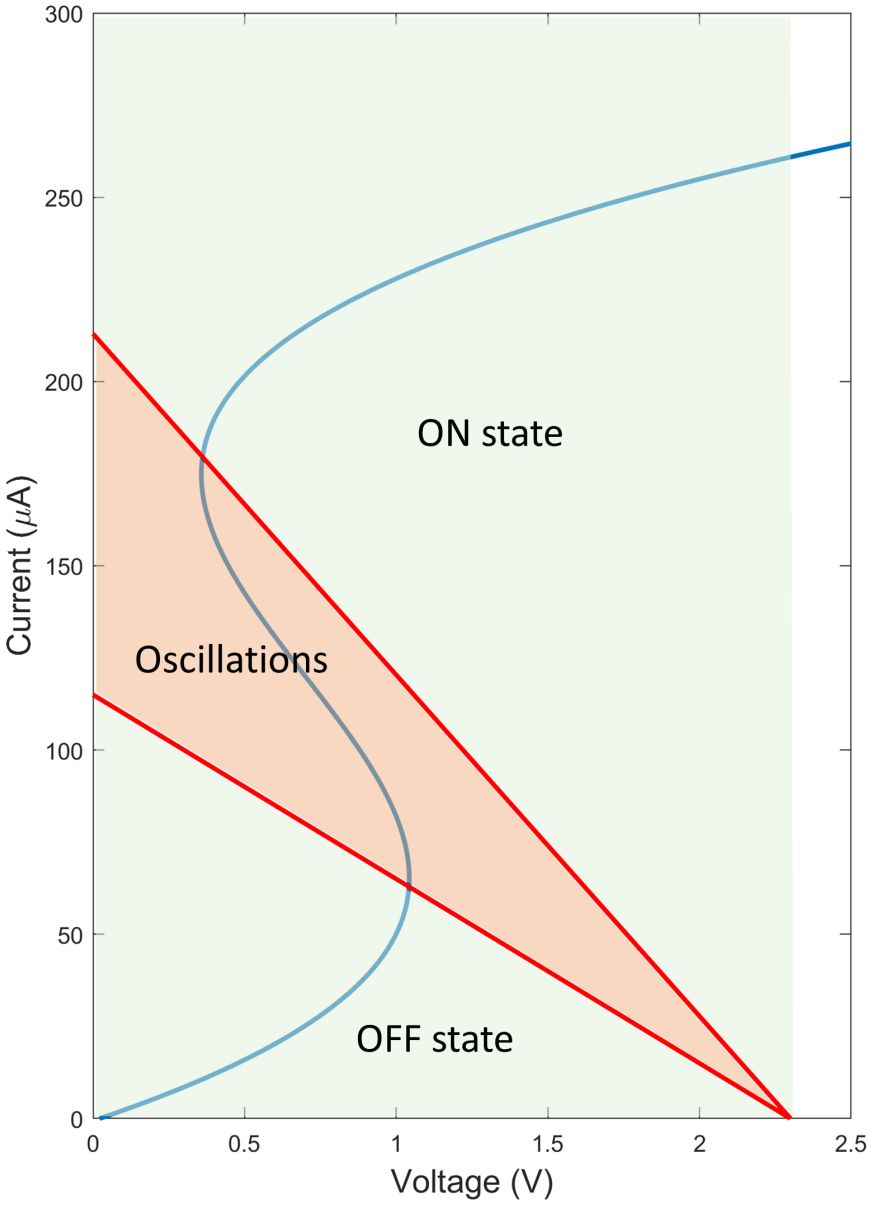



**Figure 2.8:** Regions of operation

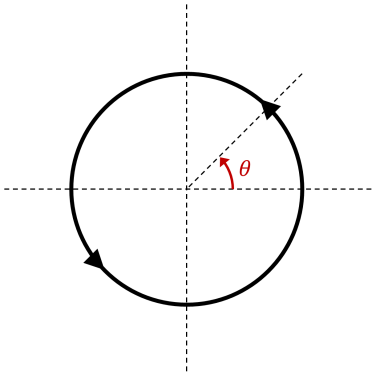



**Figure 2.9:** Phase is a state variable that flows on a circle.

defined in, such as the charging cycle and the discharging cycle. However, the first order model, for example, cannot capture the transitions between these cycles.

Finally, we should strictly note that a first order system will not oscillate *if the variable is defined on the real axis*. With a variable that is defined on a closed orbital, oscillations can happen with a first order system as you can come back to your starting point, even if you do not change the direction that you move. For example, phase is a state variable defined on a circle and a system like $\dot{\theta} = \omega$ describes the movement on a circle and corresponds to an oscillator. We will use this definition for an oscillator when we discuss the Kuramoto model in Chapter 4.

## 2.4 Second Order Oscillator Systems

In the previous section, we have seen that oscillations are impossible in a first order system where the state variable is defined on the real axis, such as current or voltage. In this section, we will look at second order systems that show oscillatory behavior. We will start by revisiting some of the basic oscillators.

### 2.4.1 The harmonic oscillator

The simplest second order system that will lead to oscillations is the harmonic oscillator which can be expressed by the following differential equation:

$$\ddot{x} + \omega^2 x = 0 \tag{2.13}$$

The solution of this equation is an oscillation of sinusoid form at angular frequency $\omega$. This type of behavior is observed in LC circuits or frictionless mass-spring systems. In those systems, there is no power dissipation: for the case of LC circuits, there is no resistor, for example. Because there is no energy dissipation, such systems are referred to as conservative systems.

However, the conservative systems are merely a simplification of the reality since some power loss is inevitable. For an LC circuit, this can be due to the leakage of the capacitor, the internal resistance of the inductor, the wire resistance or many other mechanisms that can lead to some parasitic resistance. The power dissipation due to the resistor in an RLC circuit, for example, can be expressed by adding a damping term into the simple harmonic oscillator of Equation 2.13:

$$\ddot{x} + \alpha\dot{x} + \omega^2 x = 0, \quad \alpha > 0 \tag{2.14}$$

where the damping term alpha is proportional to the resistance $R$. One solution of this equation is the damped sine, which is an oscillating waveform whose amplitude is decaying exponentially overtime following $e^{-\alpha t/2}$. Such systems where there there is some power dissipation are referred to as non-conservative systems.

Out of mathematical curiosity, and some practical reasons too, what would happen if the resistance was negative in an RLC circuit? In that case, instead of dying out, the amplitude of the oscillations would grow towards infinity as shown in Figure 2.10 (c).

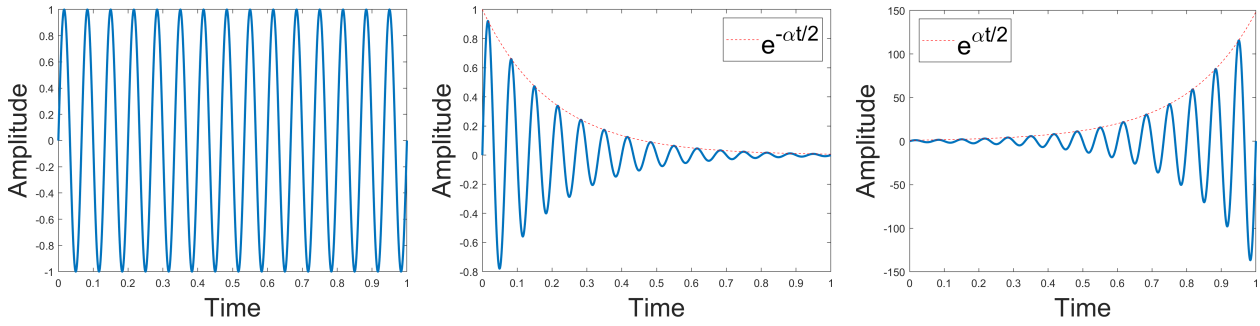



**Figure 2.10:** Harmonic oscillators (a) Simple harmonic oscillator i.e. nonzero damping (b) A positive damping causes the oscillations to die out. The source of damping can be the resistor for an LCR oscillator or friction for a mass-spring system. (c) A negative damping causes oscillation amplitude to grow to infinity.

Even though such oscillations can be observed locally i.e. for short time periods, they are not physically realizable for longer time periods. There is a need for an additional force that will limit the amplitude. In the next subsection, we will modify the damping term to achieve that purpose.

### 2.4.2 Van der Pol oscillator

In his 1926 paper, Dutch physicist Balthasar van der Pol suggested a non-linear damping term that would be negative for smaller amplitudes and positive for larger amplitudes [18]. Specifically, he recommended modifying the damping term of Equation 2.14 as follows:

$$\ddot{x} - (\alpha - 3\gamma x^2)\dot{x} + \omega^2 x = 0, \quad \alpha, \gamma > 0 \tag{2.15}$$

In this system, the oscillation amplitude will grow for small $|x|$ towards a steady state value. If we make the following change of variables:

$$t' = \omega t \tag{2.16}$$

$$x = \sqrt{\frac{\alpha}{3\gamma}}\nu \tag{2.17}$$

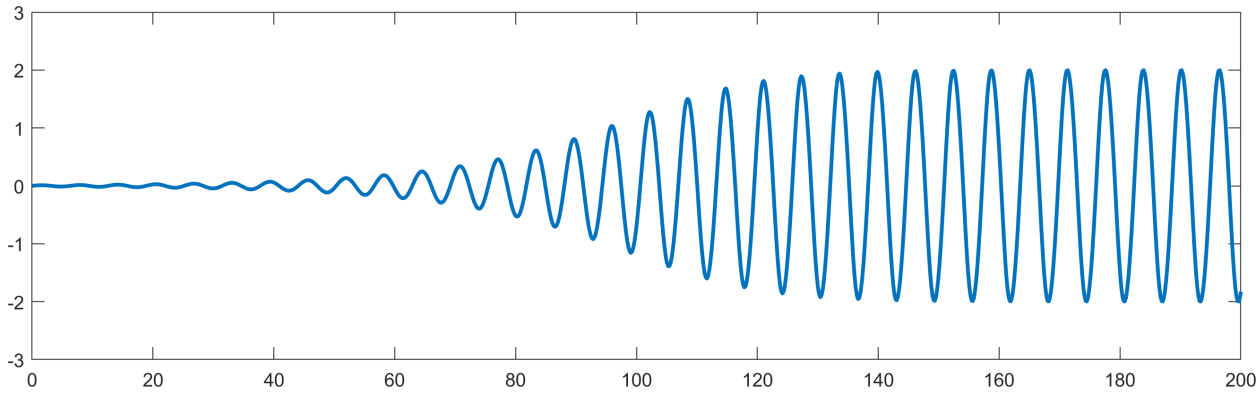
$$\epsilon = \frac{\alpha}{\omega} \tag{2.18}$$

Equation 2.15 can be written as:

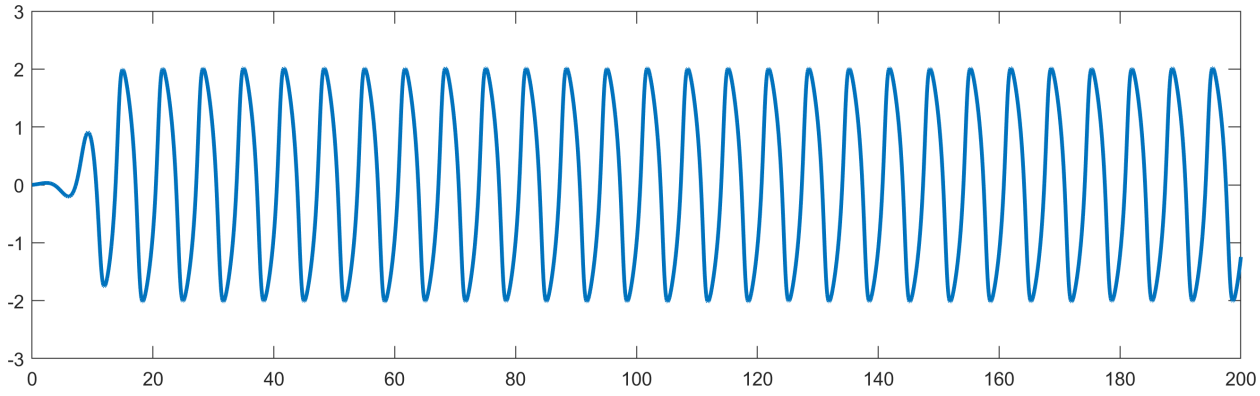
$$\ddot{\nu} - \epsilon(1 - \nu^2)\dot{\nu} + \nu = 0 \tag{2.19}$$

Equation 2.19 is the famous van der Pol equation. It describes a non-conservative system where the non-linear damping term leads to *self-sustained oscillations*. The damping factor, $\epsilon$, plays a critical role on the oscillation behavior. For very small values of $\epsilon$, the oscillations assume a sinusoidal shape and in fact when $\epsilon$ is $0$, what we end up with is the simple harmonic oscillator. As $\epsilon$ increases, the limit cycle turns into an ellipsoid and the oscillations are referred to as *relaxation oscillations* as each cycle is formed by a slow charging followed by a fast discharge. Figure 2.11 shows the oscillation waveforms for different values of $\epsilon$. For $\epsilon = 0.1$ in Figure 2.11 (a), we see that the amplitude is growing from $0$ to 2 and eventually saturates. Further, the waveform resembles a sinusoid. For $\epsilon = 1$ (Figure 2.11 (b)), the waveforms starts to deviate from a sinusoid. Finally, in Figure 2.11 (c), we see the waveform for a sufficiently large $\epsilon = 10$. In this case, the oscillations exhibit a slower cycle followed by a faster cycle.

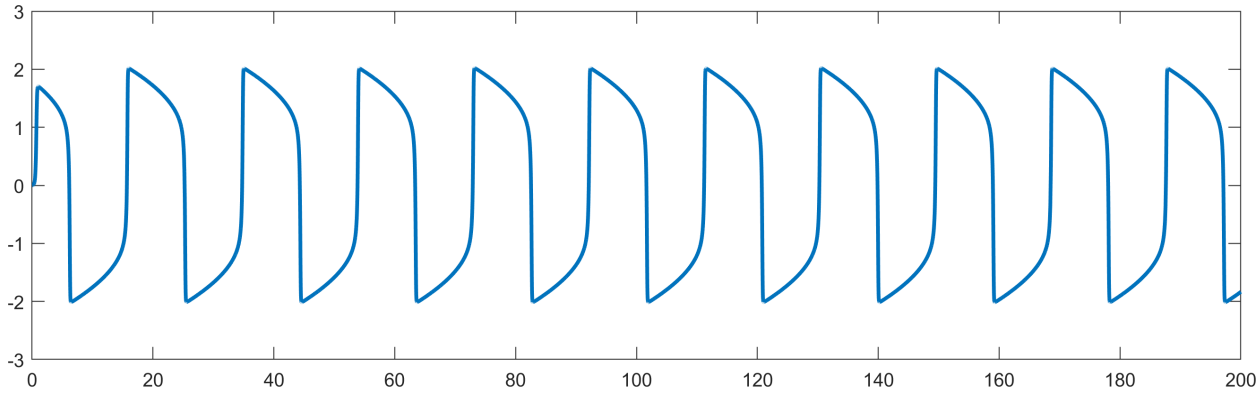
In that sense, van der Pol oscillator resembles the oscillations observed in the S-NDR oscillators where the output node was charging slowly along the OFF-state of the device and discharging quickly along the ON-state of the device IV. In the next section, we will see how the S-NDR oscillators can be modeled within the van der Pol framework.




**(a)**




**(b)**




**(c)**

**Figure 2.11:** Van der Pol oscillators recreated after [18] for (a) $\epsilon = 0.1$, (b) $\epsilon = 1$ and (c) $\epsilon = 10$. The nonlinear damping causes the amplitude to grow and saturate. The oscillations resemble a sinusoid for small $\epsilon$ and becomes a relaxation oscillation for large $\epsilon$.

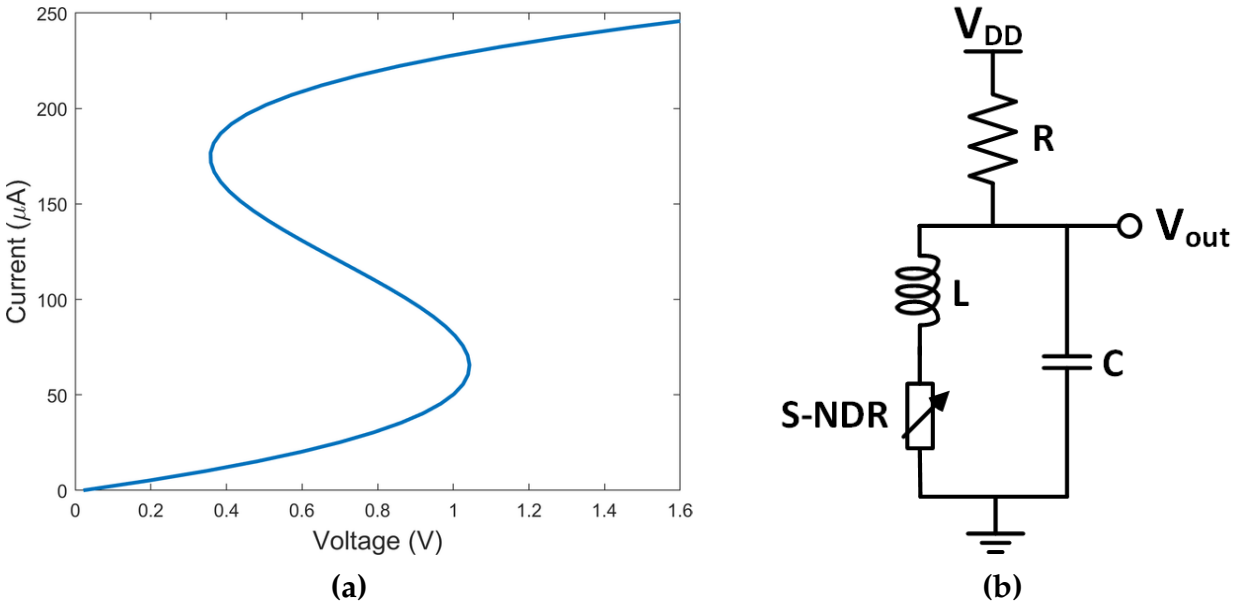



**Figure 2.12:** (a) A third-order polynomial approximation of S-NDR IV. (b) The *primary* circuit as described by Shaw in [20]. The cubic IV in this circuit corresponds to the van der Pol's oscillator.

## 2.5 Van der Pol Oscillator Using S-NDR Device

To express the current-voltage behavior of the S-NDR devices, let's assume the simplest continuous function: a 3rd order polynomial:

$$V = -a(I - I_o) + b(I - I_o)^3 + V_o, \quad a, b, I_o, V_o > 0 \tag{2.20}$$

where $a$ and $b$ are reflecting the steepness of the 3 branches of the IV curve and $I_o$ and $V_o$ are simply shifting the curve along both voltage and current axes (Figure 2.12 (a)). Shaw et. al. [19] considers the circuit shown in Figure 2.12 (b) which contains the S-NDR device and reactive circuit elements. Unlike the first order model that we discussed earlier, this model has an inductor in series with the S-NDR device.

If we analyze this circuit for the S-NDR device current, we can write:

$$LCR\frac{\partial^2 I}{\partial t^2} - [aCR - L - 3bRC(I - I_o)^2]\frac{\partial I}{\partial t} + IR - a(I - I_o) + b(I - I_o)^3 + V_o = V_{DD} \quad (2.21)$$

Defining $I' = I - I_o$, we reach:

$$\frac{\partial^2 I'}{\partial t^2} - \left(\frac{aCR - L}{LCR} - \frac{3bRC}{LCR}I'^2\right)\frac{\partial I'}{\partial t} + \frac{R - a}{LCR}I' + \frac{b}{LCR}I'^3 + \frac{I_oR + V_o - V_{DD}}{LCR} = 0 \quad (2.22)$$

Doing the following change of variables

$$\alpha = \frac{aCR - L}{LCR} \quad (2.23)$$

$$\gamma = \frac{3bRC}{LCR} \quad (2.24)$$

$$\omega^2 = \frac{R - a}{LCR} \quad (2.25)$$

$$H(I') = \frac{b}{LCR}I'^3 + \frac{I_oR + V_o - V_{DD}}{LCR} \quad (2.26)$$

we reach:

$$\frac{\partial^2 I'}{\partial t^2} - \left(\alpha - 3\gamma I'^2\right)\frac{\partial I'}{\partial t} + \omega^2 I' + H(I') = 0 \quad (2.27)$$

We define:

$$t' = \omega t \quad (2.28)$$

$$\nu = I'\sqrt{\frac{3\gamma}{\alpha}} \quad (2.29)$$

$$H' = \frac{H}{\omega^2\sqrt{\frac{\alpha}{3\gamma}}} \quad (2.30)$$

and reach:

$$\frac{\partial^2 \nu}{\partial t'^2} - \frac{\alpha}{\omega}(1 - \nu^2)\frac{\partial \nu}{\partial t'} + \nu + H'(\nu) = 0 \quad (2.31)$$

By dropping the superscripts and defining

$$\epsilon = \frac{\alpha}{\omega} \tag{2.32}$$

we finally conclude:

$$\ddot{\nu} - \epsilon(1 - \nu^2)\dot{\nu} + \nu + H(\nu) = 0 \tag{2.33}$$

Equation 2.33 is a generalized version of the van der Pol equation that we have seen in Equation 2.19. Although there are higher order terms, they should not affect the structural stability of this equation [21] and the outcome is still a van der Pol oscillator unless $H(\nu)$ breaks the unstable behavior of the oscillations. This means that, the biasing conditions that will stop the oscillations (such as $R$ not being in the right region described in Figure 2.8) are captured by this function. Indeed, we see that the bias parameters such as the ballast resistor $R$ and $V_{DD}$ are in Equation 2.26 where we first defined $H$.

By substituting Equations 2.23 and 2.25 into Equation 2.32, the damping parameter, $\epsilon$ can be rewritten as

$$\epsilon = \frac{aCR - L}{\sqrt{LCR}} \frac{1}{\sqrt{R - a}} \tag{2.34}$$

For $aCR \gg L$, this reduces to

$$\epsilon = a\sqrt{\frac{R}{R - a}}\sqrt{\frac{C}{L}} \tag{2.35}$$

Therefore

$$\epsilon \propto \sqrt{\frac{C}{L}} \tag{2.36}$$

We had discussed earlier that the van der Pol equation describes a relaxation oscillation for $\epsilon \gg 1$ and the waveform approaches a sinusoid as $\epsilon$ approaches 0. Therefore, looking at Equation 2.36, we can conclude that as $L$ grows, the oscillation waveforms will approach a sinusoid.

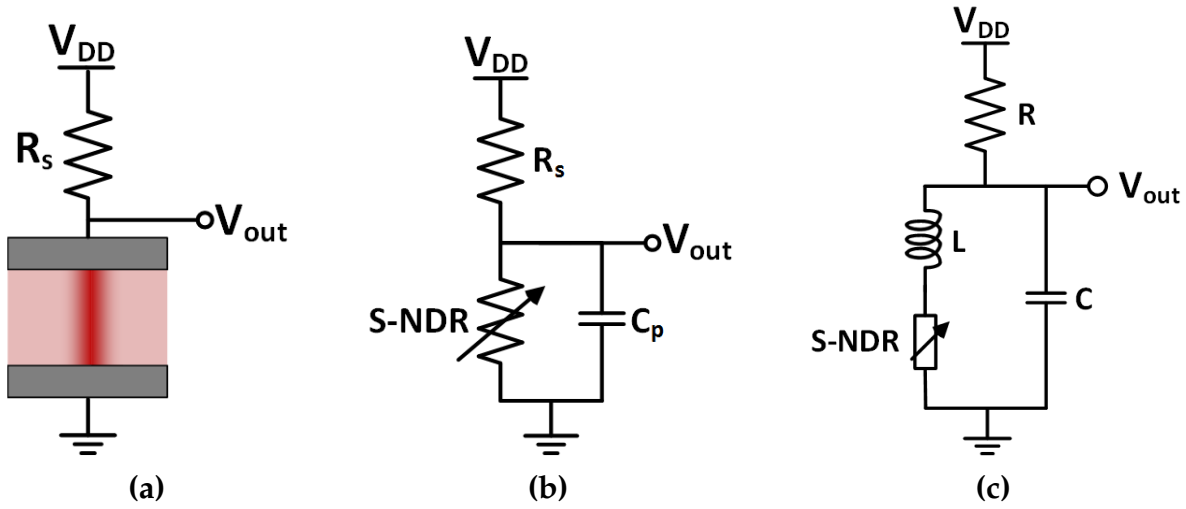



**Figure 2.13:** (a) Resistor ballasted S-NDR oscillator configuration and (b) the first order model considered earlier. In these configurations, there is no inductor. (c) Second order model with the inductor.

### 2.5.1 The case for the inductor

In our resistor ballasted S-NDR device system of Figure 2.13 (a), there is no inductor. Hence, we didn't include one in the first order model in Figure 2.13 (b). However, when we derived the van der Pol oscillator using the S-NDR device, we used a circuit with an inductor in series with the device. As we discussed earlier in Section 2.3, this was at the very least necessary to create a second order system. In this section, we will further discuss why is it required and what physical processes it might be capturing.

In the first order model, the parasitic capacitance $C_p$ is charged when the device is in the OFF-state. During this charging, the oscillation waveform follows the device IV and reaches the threshold voltage. We have drawn the oscillation trajectory on the device IV on Figure 2.14 with red dashed lines. At this point, switching occurs. Because the capacitor requires the continuity of the output voltage, during the switching the trajectory jumps vertically to the ON-state. The capacitors starts discharging, again along the device IV and switches back to the OFF-state. It once again switches vertically for the same reason. In

both of these switching events, the device current changes instantaneously. This would require an infinite current supply into the device which is physically not realizable.

Furthermore, Figure 2.13 (b) implies that the device voltage shoud be equal to the capacitor's voltage. Yet, looking at the trajectory in Figure 2.14 we see that there is a gap between the device voltage and the capacitor's voltage during the switching events. Therefore, a circuit element that is in series with the S-NDR device is required. Hence, an inductor in series with the S-NDR device makes mathematical sense to

1. produce a second order system,
2. ensure the current continuity of the device during switching,
3. account for the discrepancy between the device voltage and the capacitor voltage.

While discussing Abraham & Bloch's multivibrator described in [22, 23], van der Pol argues that the inductance should be arising from the parasitic inductance of the wires in the circuit [18]. Even though some parasitic inductance is inevitable, there are other physical reasons to include an inductor in the circuit model originating from the nature of the switching event. Our current understanding is that the conduction in the OFF state is homogeneous meaning the current density is constant across the device cross-section. This assumption ignores the surface effects. As the device voltage reaches the $V_{th}$, a more conductive filament forms. This leads to an increase in the device current while shunting the voltage across. If there is a dynamic filament formation, it makes sense assume that there must be a finite-time required to create the filament. This is true whether the switching event is a predominantly thermal or electronic (athermal). In chalcogenides, the switcing process is athermal and the associated time constants are in the order of ps [24]. In $TiO_x$, the switching process is predominantly thermal and the time constants are in the range of 1 ns [25]. In [26], Pickett states that the time-continuous enthalpy within the devices exhibiting thermal switching requires device current to be continuous and this causes an inductor-like behavior of the filament. Therefore, it makes sense to include an

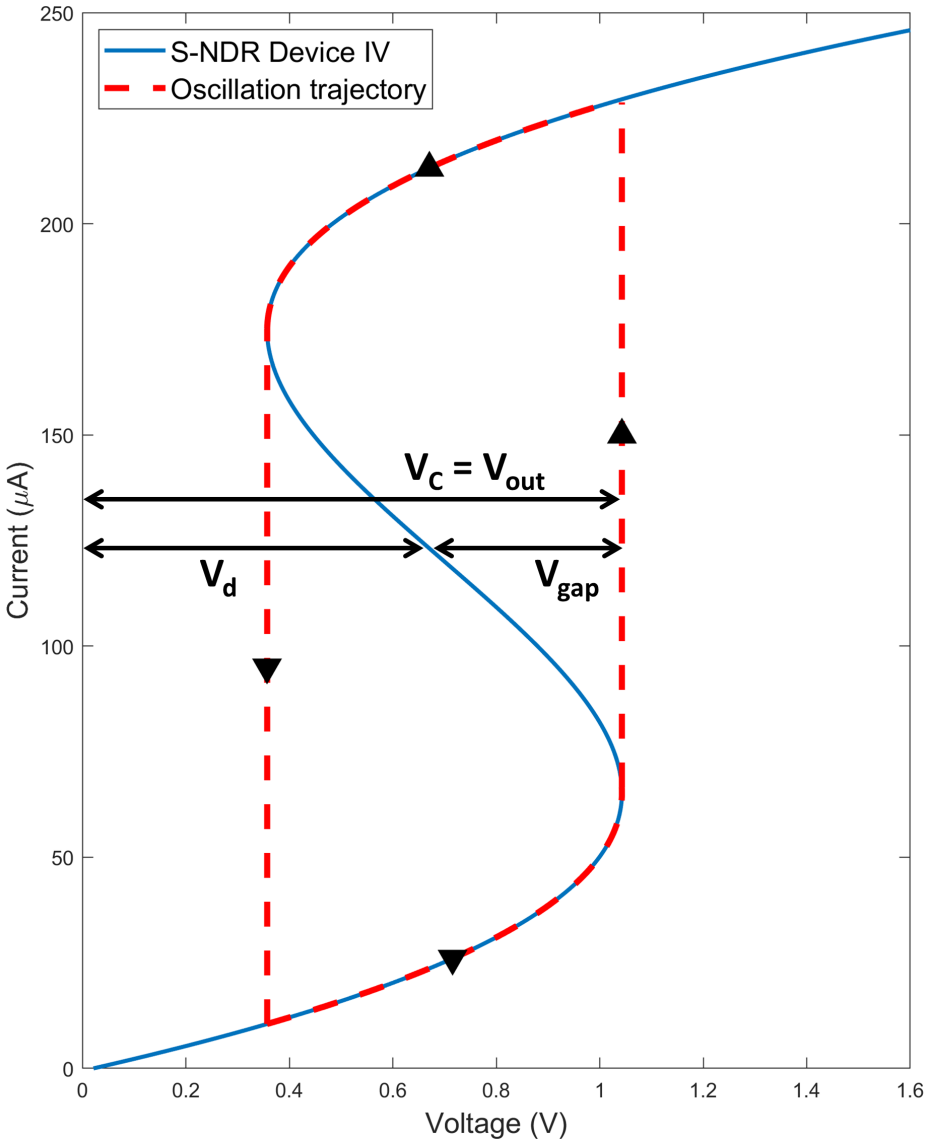



**Figure 2.14:** S-NDR device IV (blue) and the oscillation trajectory (red). The first order model without the inductor comes short in explaining the transitions between the OFF and ON states. During the transitions, there is a *gap* between the device IV and the oscillation trajectory. This $V_{gap}$ is captured by the inductor in the second order model. Physically, it captures the dynamics of the filamentation.

inductor in the circuit model in series with the S-NDR device to capture the transients of the switching.

# Chapter 3

# Developing S-NDR Nano-oscillators

The S-NDR devices studied in this work are crossbar configured metal-insulator-metal (MIM) or metal-semiconductor-metal (MSM) stacks, where the insulator/semiconductor is a threshold switching material (Figure 3.1). Chalcogenide glasses has been known to exhibit threshold switching properties and oscillators are demonstrated using them since 1970s [27]. Sub-stoichiometric phases of transition-metal-oxides such as $Ta_2O_5$ also show reversible threshold switching properties [28]. Other materials studied commonly are $VO_2$ [14] and $NbO_2$ [26]. However, recent efforts at Carnegie Mellon University has shown that composites of chalcogenide glasses and transition-metal-oxides (TMO) take advantage of the best properties of each material family: chalcogenide glasses such as $GeTe_6$ show fast switching and high endurance. Oxides such as $HfTaO_x$, on the other hand, has lower leakage in the OFF state and has higher transition temperatures [29].

Therefore, the work proposed here will not focus on the further optimization of materials. We have found it easier to deposit $GeTe_6$ showing threshold switching characteristics compared to TMOs. Therefore, $GeTe_6$ threshold switches and oscillators are used as a test vehicle in this study. A photo of a typical device is provided in Figure 3.1 (c). The bottom

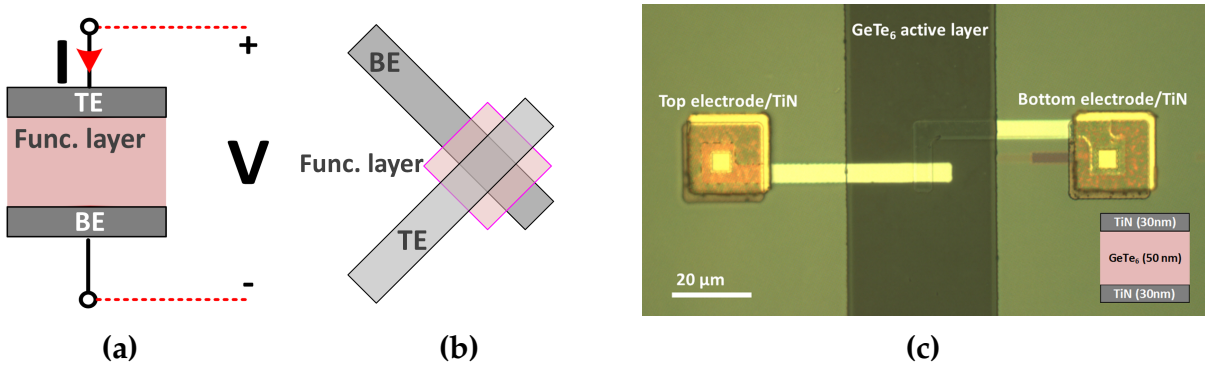



**Figure 3.1:** (a) Cross-sectional diagram of S-NDR devices. (b) Top view diagram of devices arranged in the crossbar configuration. (c) Microscope image of fabricated devices.

electrode is a stack of 10 nm of Pt and 30 nm of TiN and the top electrode is 30 nm of TiN. $GeTe_6$ was deposited by co-sputtering Ge and Te and the rates were optimized to ensure the accurate composition. The dimensions of this device is 5 µm × 4 µm and the thickness of the $GeTe_6$ layer is 50 nm.

## 3.1 Device Characterization

The device characterization starts with a DC sweep. For DC sweep, the device is connected in series with a resistor ($R_s$), similar to the resistor ballast scheme described in the previous chapter. Because of the nature of the device's current-voltage (IV) behavior, it makes sense to sweep current/voltage up and down across the device. Further, larger $R_s$ is generally desirable (within the limits of the measurement instruments) to be able to trace the IV.

The IV of the devices shown earlier are shown in Figure 3.2 (b). The unstable negative-differential-resistance (NDR) region is annotated. Note that the NDR region is not captured during the measurements as the system is unstable in that region, and the device jumps from the stable OFF state to the ON state once the threshold voltage is reached. As soon as the device voltage reaches threshold voltage ($V_{th}$), a conductive filament is formed which results in a lower-resistance ON state. The filament and therefore the low-resistance

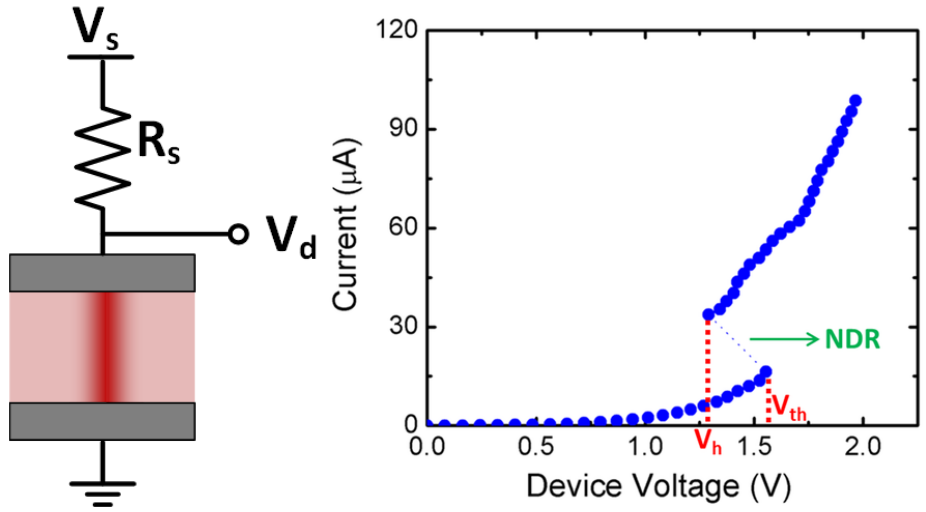



**Figure 3.2:** (a) DC characterization of the device is achieved with a resistor in series. (b) DC IV of the $GeTe_6$ devices pictured earlier.

ON state is preserved as long as the device voltage is above the holding voltage, $V_h$. For this particular device, $V_{th}$ is around 1.6 V and the holding voltage $V_h$ is around 1.3 V.

The exact mechanism behind the filamentation is still debated, but the relation between the threshold voltage and the device thickness suggests an electric field driven mechanism. We fabricated devices with varying functional layer thicknesses and measured the IV characteristics. The dependence of $V_{th}$ on the device thickness is shown in Figure 3.3 (a). There is a linear relation between the two for thinner devices ($< 60$ nm). Therefore, it is possible to design for a certain threshold voltage by controlling the device thickness.

The holding voltage is shown to be a function of the electrode material work function [28]. We also believe that the contact resistance between the electrodes and the functional layer can affect the holding voltage. As depicted in Figure 3.3 (b), if a minimum critical potential across the filament ($V_f$) is required to sustain it i.e. keep the device in the ON-state, then $V_f$ will be a function of the device voltage and the contact resistances. The measured $V_h$ therefore can be reduced by engineering the contacts for lower contact resistance. The oscillation amplitude is predominantly dictated by $V_{th}$-$V_h$ and lower $V_h$ is

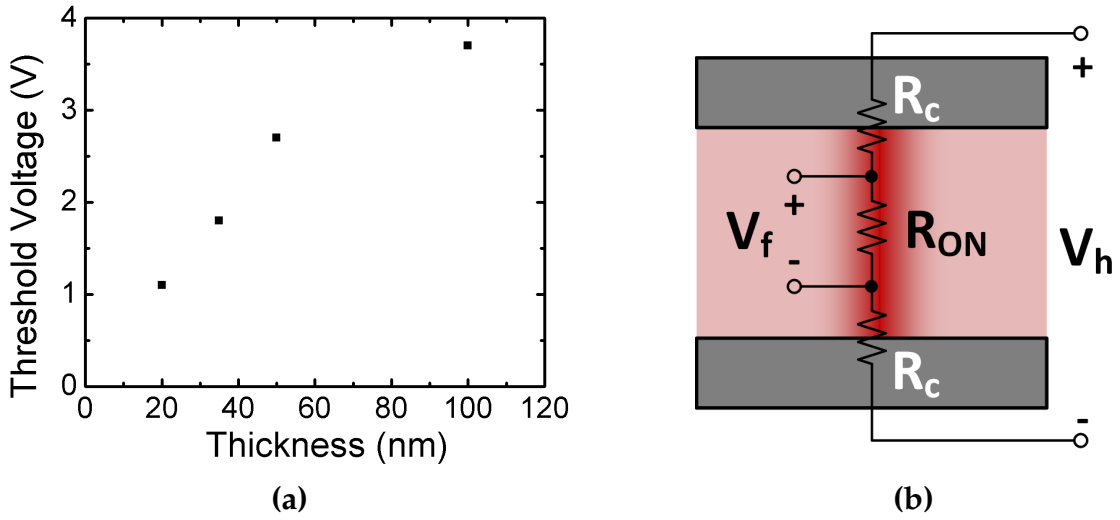



**Figure 3.3:** (a) Dependence of the threshold voltage ($V_{th}$) on the device thickness. (b) Effect of contact resistance on the holding voltage ($V_h$). Resistive voltage division across the device reduces the potential across the filament ($V_f$) compared to $V_h$.

required for larger amplitudes.

As described in the previous chapter, the biasing conditions of the resistor-device system shown in Figure 3.2 (a) can lead to sustained oscillations. When the device is biased in the NDR region using a lower $R_s$ and appropriate $V_s$, it is possible to generate oscillations. Figure 3.4 (a) shows the current and voltage oscillations with such biasing. The device voltage exhibits relaxation oscillations i.e. slow charging followed by a quick discharge of the parasitic capacitance seen at the output node. the oscillation frequency, in this case, depends on the supply voltage, $V_s$. As $V_s$ increases, more current is supplied and the parasitic capacitance charges faster. Figure 3.4 (b) shows the dependence of oscillation frequency on the input voltage ($V_s$). Also, we observe the effect of the reduced parasitic capacitance on the oscillation frequency. When the voltage probe at the device output node is removed, oscillation frequency increases. In this case, we only measured the device current and frequency is measured using the current waveform.

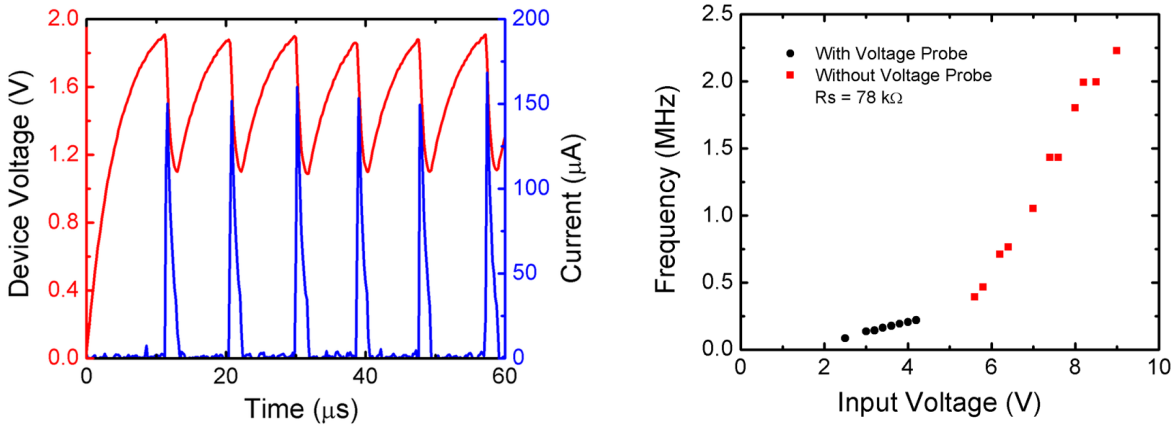



**Figure 3.4:** (a) Current (blue) and voltage (blue) oscillations observed in $GeTe_6$ devices. (b) Supply voltage dependence of the oscillation frequency in the resistor ballast configuration. Increasing the supply reduces the time required to charge the parasitic capacitance seen at the output hence the frequency increases. Frequency increases more steeply with supply when we remove the voltage probe which normally adds additional parasitic capacitance.

## 3.2 Transistor Ballast for a Voltage-Controlled-Oscillator

Instead of the ballast resistor, a transistor can also be used as a ballast element. Unlike a resistor, a transistor allows the loadline to be tuned through the gate voltage applied, even for a constant supply voltage. The stability conditions outlined in the previous chapter are still valid: the circuit will have stable operating points if the transistor loadline meets S-NDR I-V at the two stable branches. Thanks to the gate voltage control, the operating point can be changed and the S-NDR device can be programmed for high or low resistance operation. This type of threshold switching behavior is useful in resistive memory applications. What is more interesting, an S-NDR device in series with a transistor can operate as a voltage controlled oscillator (VCO) when the loadline is intersecting the device I-V at the NDR region. Because oscillations are caused by the charging and the discharging of the parasitic capacitor, oscillation frequency increases with lower gate voltages on the PMOS transistor (Figure 3.5 (a)). The scheme shown on Figure 3.5 (b) is referred to as a 1T-

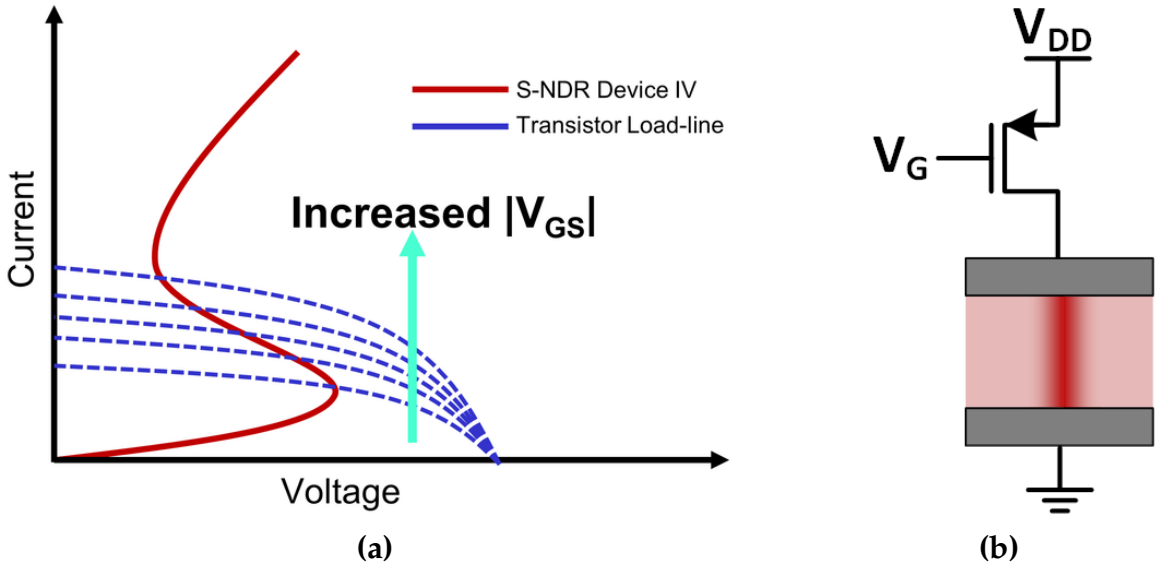



**Figure 3.5:** (a) A cartoon of device IV overlaid with a transistor's loadline in the transistor ballasted (1T1R) scheme shown in (b). Gate voltage of the transistor can be used to bias the device at various current levels within the NDR region and this configuration can be utilized as a compact voltage controlled oscillator.

1R (1 transistor-1 resistor) system, which serves as a very compact, highly scalable VCO. A VCO is the unit cell of a coupled oscillator network, therefore, the scalable, low-power S-NDR oscillators proposed in this work lend themselves naturally to such applications.

In 1T-1R S-NDR oscillators, gate voltage serves as the control knob of the VCO. The higher the gate-to-source voltage is, the higher the oscillation frequency is due to the increased current supply. However, this also increases the power consumption of the oscillator. For low power oscillators, the sub-threshold region of the transistor can be utilized. Figure 3.6 (a) shows the oscillations when a transistor biased in the sub-threshold region was used. The gate voltage still can be used to adjust the oscillation frequency (Figure 3.6 (b)).

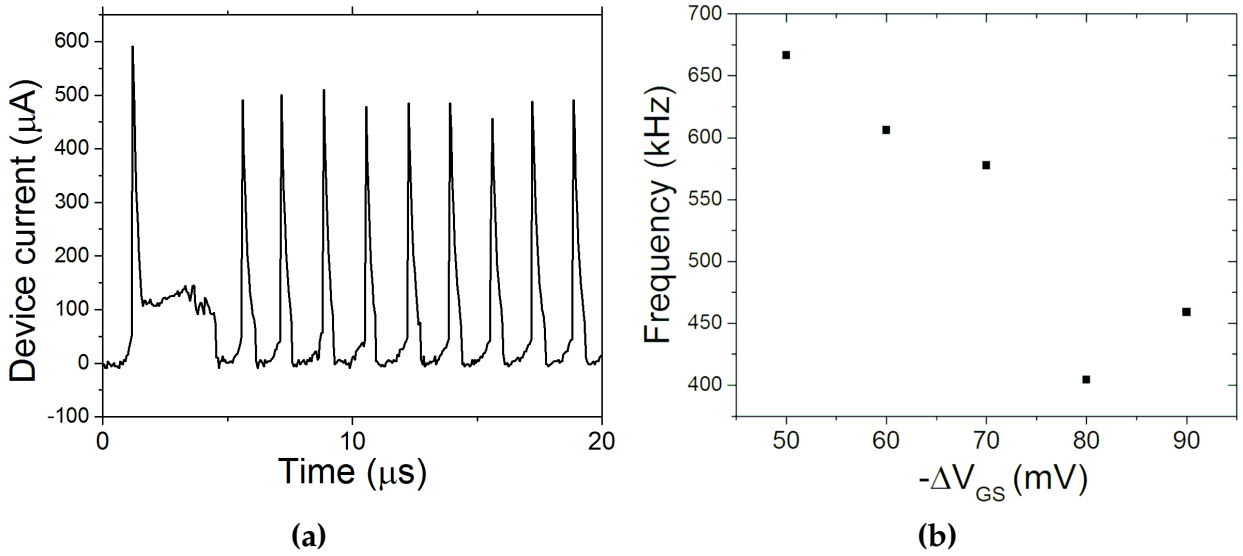



**Figure 3.6:** (a) Current oscillations of the device when it is ballasted with a transistor operating in the sub-threshold region. (b) The dependency of oscillation frequency on the gate voltage of the transistor, $\Delta V_{GS} = |V_{GS}| - |V_{TH}|$ in the subthreshold region. The inset shows the transistor ballasted oscillator scheme.

## 3.3 Integration with Scaled CMOS Transistors

The ballast elements discussed so far are discrete circuit components wired to the fabricated devices through the measurement instruments. This leads to an increased parasitic capacitance at the output node. The on-chip integration of S-NDR devices with scaled CMOS technology is crucial to prove compatibility with the existing CMOS technology. To demonstrate such integration, a CMOS chip was taped out at a commercial foundry at 28 nm CMOS process technology. The CMOS fabrication process was cut short at metal 7 layer to allow for integration. On this chip, we have single PMOS transistors whose gate, source and drain connections are taken up to metal 7 layer. An SEM photo of the area where the transistor lies can be seen in Figure 3.7.

We developed a fabrication process, fabricated and tested nano-oscillators integrated onto a CMOS chip. The details of the fabrication process is provided in Appendix 1. A photo of the fabricated device along with the circuit schematic is shown below in Figure

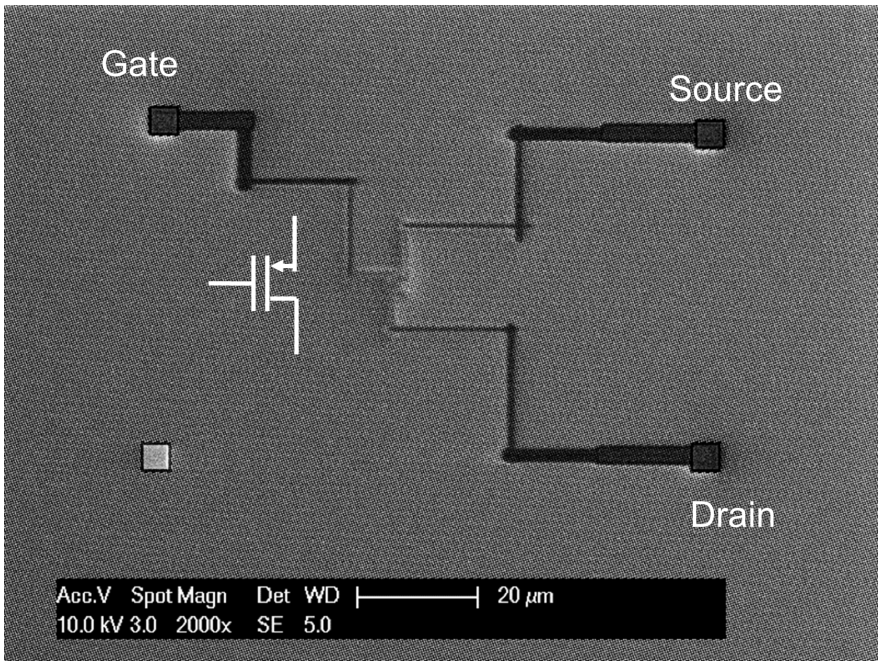
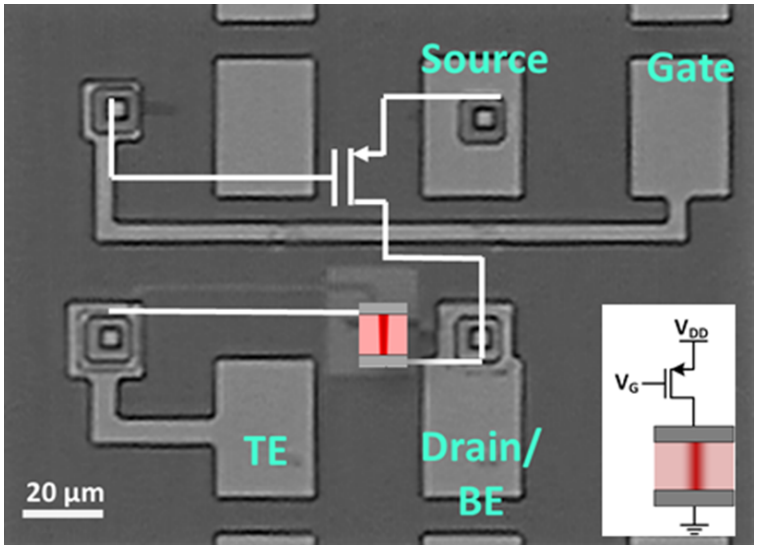



**Figure 3.7:** SEM photo of a transistor on the 28nm CMOS chip. The terminals of the transistor are connected to metals going up to M7. On M7, there is a larger (10µm×10µm) metal plate which contacts the pads of the fabricated device.

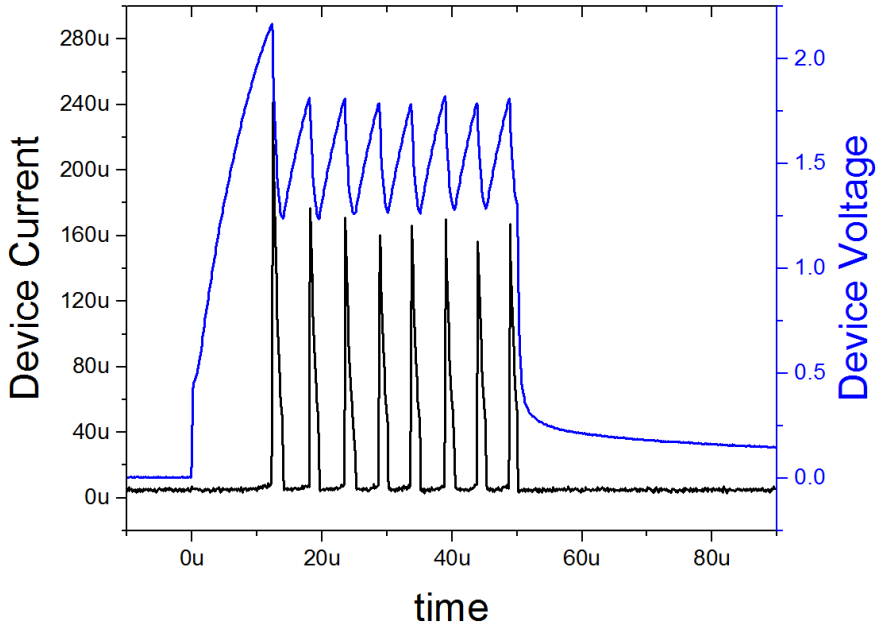
3.8 (a). Figure 3.8 (b) shows the output voltage and the current waveforms of the integrated oscillator.

## 3.4 Oscillator Modelling

A SPICE circuit model is helpful in terms of understanding the effects of circuit parameters such as parasitic capacitances and it can be used to develop novel circuit topologies and for system level simulations using coupled oscillators. Following the van der Pol formalism discussed in detail in the previous chapter, we developed a SPICE model. In order to simulate van der Pol's oscillator, we created the circuit shown in Figure 3.9 in SPICE. In order to implement the S-shaped current-voltage relationship of the S-NDR stack, we assumed a piecewise-linear I-V curve. It was fitted to the measured data as shown in Figure 3.9 (b). This DC I-V behavior was implemented in Verilog-A. Then, the circuit shown




**(a)**




**(b)**

**Figure 3.8:** (a) Die photo showing S-NDR devices integrated with scaled CMOS transistors taped at 28nm process technology. The circuit schematic is overlaid for clarity. The pads are oriented to fit the SSG RF-probes we used for measurements. (b) Current and voltage waveforms of the oscillators.

in Figure 3.9 (a) was simulated using SPICE. The inductance and the capacitance in the circuit was chosen such that the simulations and the measured voltage oscillations are in good agreement (Figure 3.9 (c)). The model captures the threshold switching and oscillations properties of the S-NDR oscillators, and therefore was used for the simulations in this work.

## 3.5 Cross-coupled Transistor Pair as a Ballast

It is noted previously that the DC operating point should be chosen in the NDR region for oscillations. In the previous chapter, we showed that the resistance of the ballast resistor should be larger than the differential resistance of the S-NDR device in the NDR region. We expressed that this is a geometric implication of biasing the device in the NDR region. For the case of the transistor ballast, this condition is similar. Oscillations can be observed only if $|NDR| < r_o$, where $r_o$ is the output impedance of the transistor. Therefore, using a transistor as a ballast element poses a problem because of the finite output impedance ($r_o$) of transistors. The problem becomes more prominent as the ON-state current reduces since the allowed window for oscillations narrows down. (Lower ON-state current is desirable for lower power.) This puts a restriction on transistor sizing.

In his 2014 article, Razavi declares cross-coupled transistor pair (XCP) as a *beautiful* circuit as it finds applications in sense amplifiers, regenerative latches and LC oscillators [30]. The cross-coupled pair shown on Figure 3.10 (a) exhibits a negative resistance of $-2/g_m$ between the drains in the small-signal regime. In memories and digital circuits, it is used as a bistable element with no static power dissipation. We decided to explore XCP as a ballast element for S-NDR devices. Figure 3.10 (b) shows its loadline along with a transistor's loadline at different gate voltages. The loadlines are overlaid on the piecewise linear IV of a hypothetical (yet reasonable) S-NDR device. Judging by this simple figure, XCP

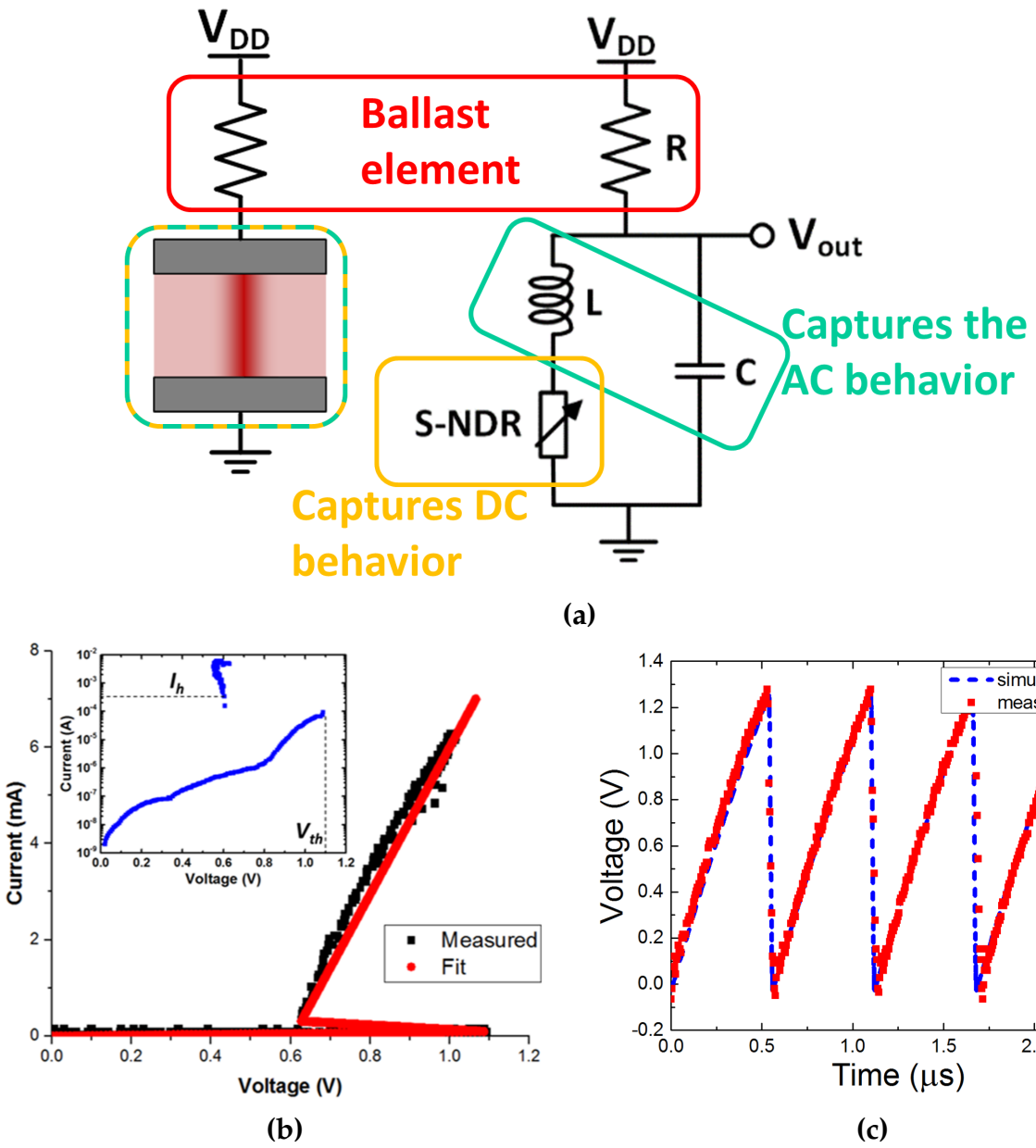



**Figure 3.9:** (a) S-NDR element with reactive circuit components added to describe the AC behavior. The circuit on the left describes the behavior of a resistor connected in series with the S-NDR material stack. (b) To model DC behavior of the S-NDR device, a piecewise linear curve is fitted to the measured IV data. (c) The simulated and measured waveforms are in good agreement.

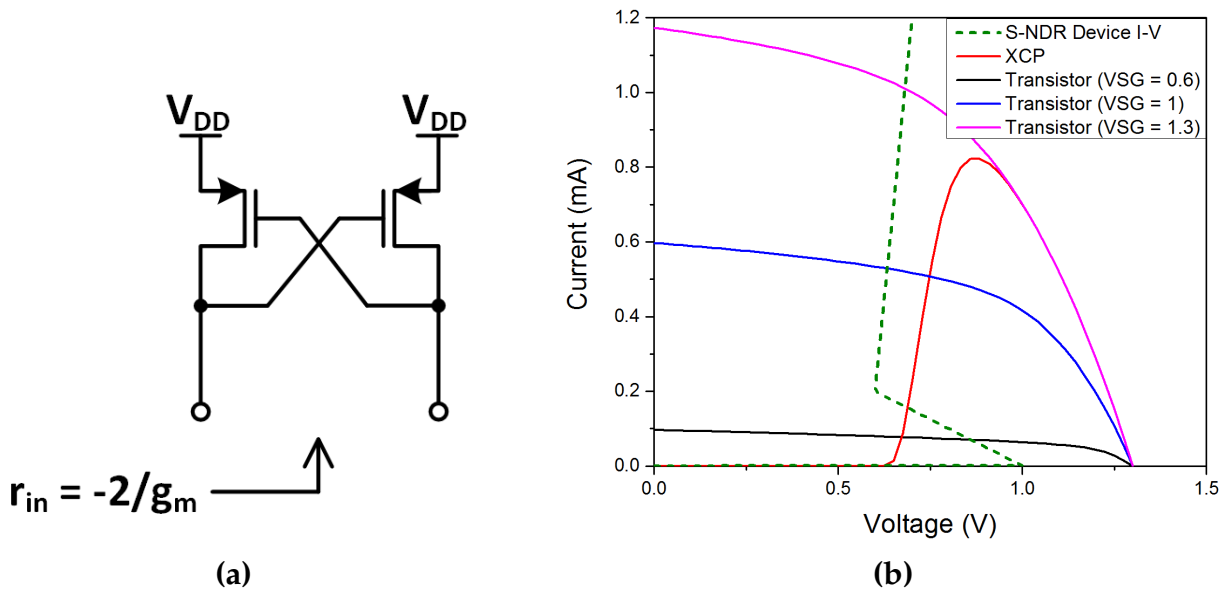



**Figure 3.10:** (a) PMOS cross-coupled transistor pair. (b) Loadlines of a transistor and a cross-coupled pair overlaid with a piecewise-linear S-NDR device IV.

looks promising in terms of biasing S-NDR devices, as it doesn't have output impedance characteristic like a transistor's.

To utilize the cross-coupled pair as a ballast element, we connected two S-NDR devices to the drains of the two PMOS transistors as shown in Figure 3.11 (a). The oscillation outputs of this scheme are at the drains and there are two outputs. Using the model we developed, we simulated this circuit on SPICE. Figure 3.11 (b) shows the voltage waveforms at the two outputs. As the gates of the PMOS transistors are connected to the outputs, the two voltage waveforms are out of phase. The availability of two out of phase outputs is potentially useful for phase-based neural networks and phase-based logic applications.

Next, we compare 1T-1R (transistor ballast) and XCP-ballasted systems in terms of the transistor sizing. We investigated the allowed W/L ranges of the transistors for both systems. A larger W/L range translates to design flexibility for oscillator networks. We ran simulations for constant transistor length (L = 100 nm) and varying width. Our simulations showed that 1T-1R system does not oscillate for W>1.7 µm as the loadline now

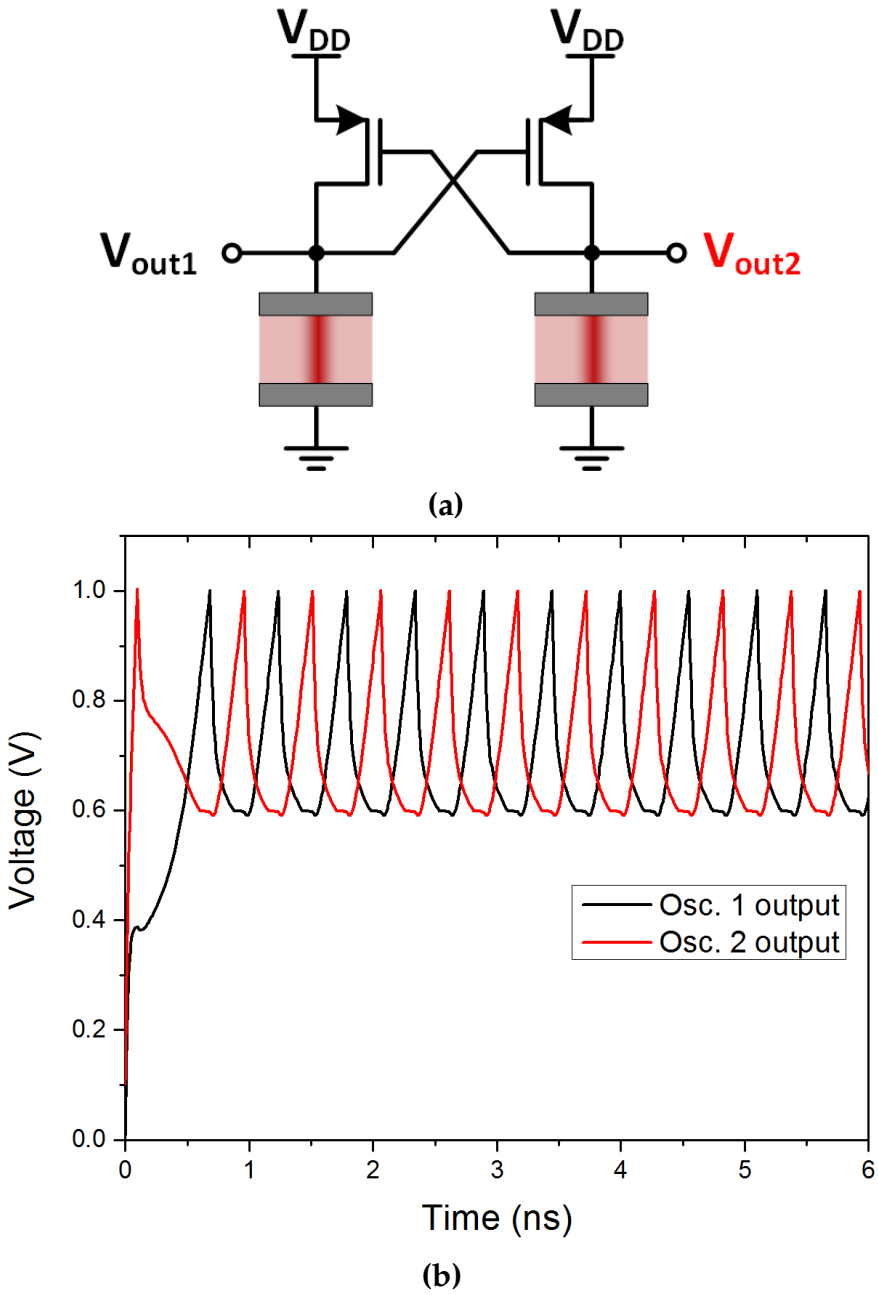



**Figure 3.11:** (a) Cross-coupled pair ballasted S-NDR oscillator scheme (b) Voltage waveforms at the output terminals.

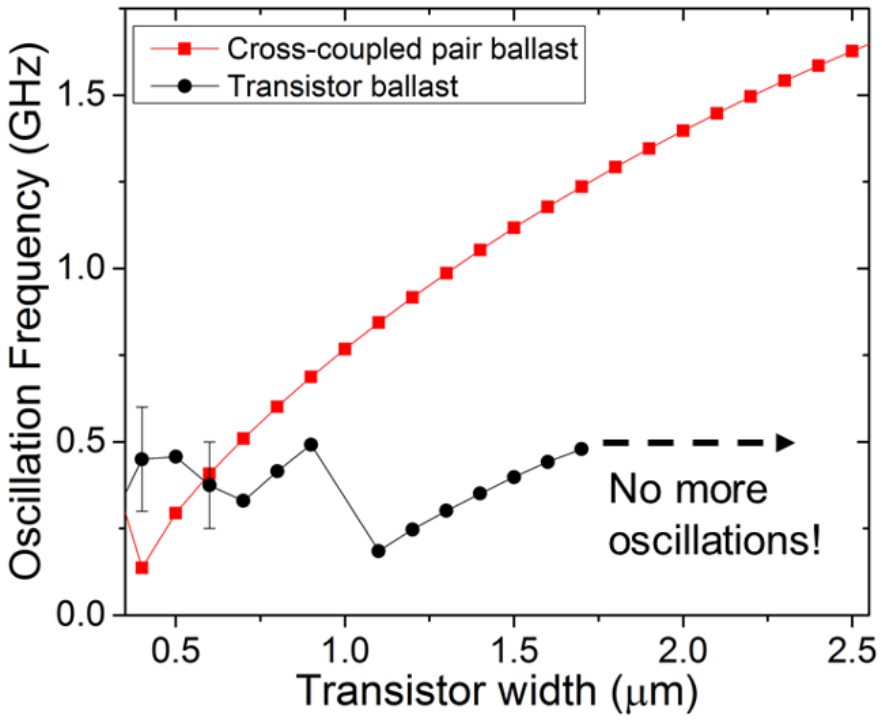



**Figure 3.12:** A comparison of 1T1R and XCP-ballasted systems in terms of allowed W/L range and oscillation frequency.

intersects the upper stable branch of the S-NDR IV assumed here and the device stays at ON state. On the other hand, XCP-ballasted oscillators not only oscillate even for larger W, but the oscillation frequency is larger compared to the 1T-1R system. These results are shown in Figure From the applications point of view, higher oscillation frequency can be desirable: coupled oscillator networks converge to a solution after certain number of oscillation cycles [31], therefore achieving a higher frequency is important as the computation time reduces.

One other observation from Figure 3.12 is the higher oscillation frequency of XCP-ballasted oscillators. The higher oscillation frequency arises from the fact that the two oscillators are coupled and they are out of phase. When oscillator 1 shown in Figure 3.13 is charging, oscillator 2 is discharging and therefore $V_2$ decreases quickly to lower values (Figure 3.13 (a)). This increases the current flowing into oscillator 1 ($I_1$) as the gate voltage of its transistor ($M_1$) is low. Higher source current reduces the charging time. In the

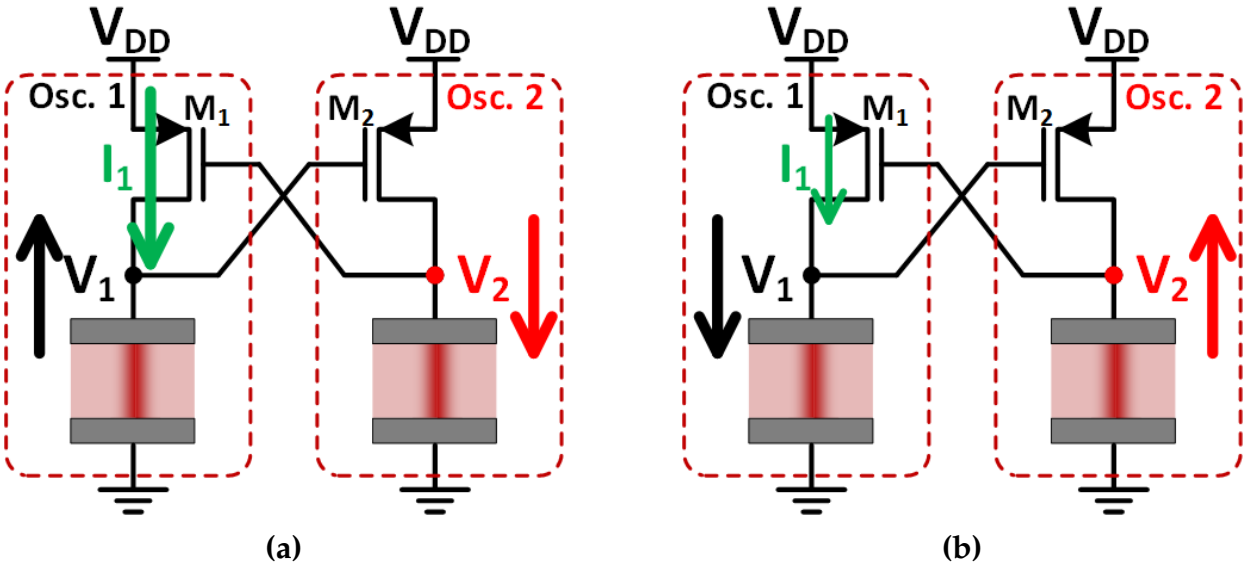



**Figure 3.13:** PMOS cross-coupling of the two outputs enables higher frequencies compared to 1T1R system by facilitating (a) easier charging and (b) discharging of the outputs. The diagrams are drawn for the charging and discharging cycles of $V_1$.

discharge cycle of oscillator 1, the operation is reversed (Figure 3.13 (b)). As oscillator 2 is now charging, $V_2$ is increasing. Therefore, the current supplied to oscillator 1 is reducing. Lower source current allows for a quicker discharge. Therefore, the two oscillators in the cross-coupled scheme helps each other in terms of quicker charging and discharging, which leads to the higher oscillation frequency seen on Figure 3.12.

One obvious disadvantage of the XCP-ballast is that it is not a VCO and there is no knob to tune the frequency other than $V_{DD}$. Voltage tunability is important in oscillator networks where the information is encoded into frequency. In order to make the cross-coupled pair a VCO, we can add a transistor pair to the top which can restrict the current flow from the VDD using the gate voltage control. Fig. 8 shows this circuit and how the control voltage affects the oscillation frequency.

Finally, we fabricated the XCP-ballasted oscillator shown in Figure 3.11 (a) using the process flow that we developed for the CMOS integrated oscillators. The two outputs are shown in Figure 3.15. While one of the outputs is oscillating (shown in red trace), the other

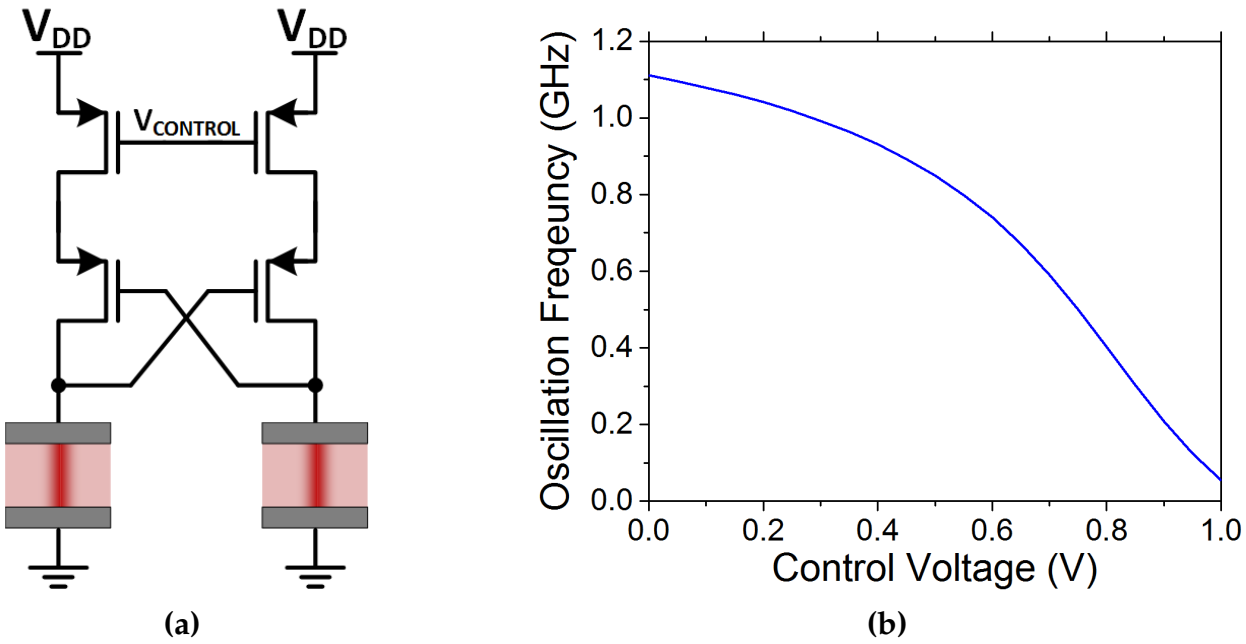



**Figure 3.14:** (a) Cross-coupled pair ballasted VCO. In order to control the frequency, current injection into the devices should be tunable. The PMOS pair connected between the $V_{DD}$ and the sources of the XCP provides the control know to tune current. (b) Control Voltage vs. Frequency

one is not. This is due to the variation between the two S-NDR devices. In a more controlled process environment, the two devices should be more similar and oscillate similar to our simulations.

## 3.6 Scaling the S-NDR Oscillators

The DC-IV character shown in Figure 3.2 belongs to a 5 µm×5 µm device. We believe that the conduction is uniform in the OFF-state, therefore we can assume that the OFF-state current will scale down with area. If the device dimensions are reduced to 50 nm×50 nm, the current at the swithcing point ($I_{th}$) will go down to 1.5 nA. $V_{th}$ is mainly a function of the device thickness, so it should stay constant. Therefore, the switching power will become 2.4 nW. Therefore, the device power can be as low as 2.4 nW for sustained oscillations.

If we assume 10fF parasitic capacitance for the scaled devices and 0.5 V oscillation

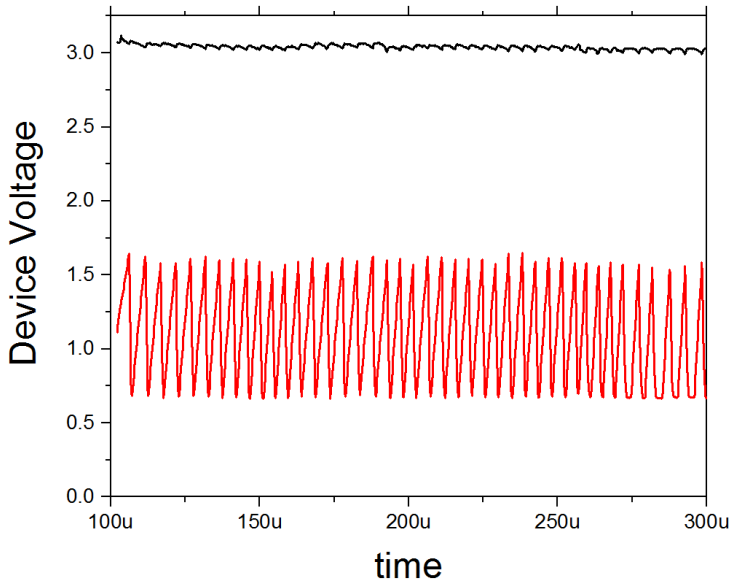



**Figure 3.15:** Oscillation waveforms of the fabricated XCP oscillator. While one of the outputs is oscillating (shown in red trace), the other one is not. This is due to the variation between the two S-NDR devices. In a more controlled process environment, the two devices should be more similar and oscillate similar to our simulations.

amplitude, the frequency of oscillations go up to a MHz for a 2.5 nW oscillator. The frequency can become a GHz for an oscillator biased at 2.5 µW. Therefore, we can conclude that S-NDR oscillators are potentially low power and scalable, lending themselves nicely for dense network applications that we will be discussing in the next chapter.

# Chapter 4

# Coupled Oscillator Networks for Computation

In the first chapter, we briefly mentioned that coupled oscillators synchronize when their natural oscillation frequencies are close. In this chapter, we will discuss this behavior in more detail. Furthermore, we will present coupled S-NDR oscillator networks that achieve image processing problems thanks to this synchronization behavior.

Let's consider a point moving on the unit circle as in Figure 4.1. The $x$ and $y$ positions of the point can be expressed as $y = sin(\omega_o t + \theta_o)$ and $x = cos(\omega_o t + \theta_o)$. These x and y coordinates will be oscillating with constant frequency $\omega_o$. We will call the operands in these equations as *phase*:

$$\theta(t) = \omega_o t + \theta_o \tag{4.1}$$

$\theta_o$ is the initial phase at $t = 0$. The time dependence of Equation 4.1 can be expressed by a differential equation:

$$\frac{\partial \theta}{\partial t} = \omega_o, \quad \theta(0) = \theta_o \tag{4.2}$$

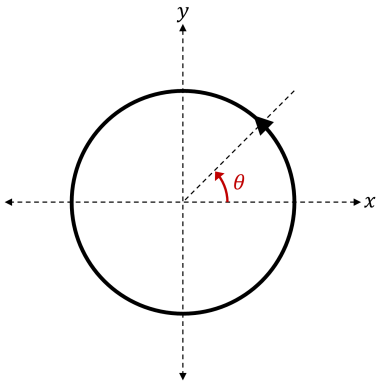



**Figure 4.1:** Phase is a state variable that flows on a circle.

As long as $\theta$ refers to the movement on a circle (or another closed trajectory), Equation 4.2 defines the simplest oscillator [17]. It describes a single, free-running oscillator without any perturbation to its movement. To model the synchronization behavior of coupled oscillators, we need to consider the contribution of interaction due to coupling.

## 4.1 Kuramoto model

Let's consider two oscillators with different natural frequencies, $\omega_1^o > \omega_2^o$. The difference between these natural frequencies is referred to as *detuning*: $\Delta\omega^o = \omega_2^o - \omega_1^o$. Since $\omega_1^o > \omega_2^o$, $\theta_1 > \theta_2$ as t increases. For two oscillators to synchronize, in case of coupling, oscillator 1 needs to slow down. This should appear as an addition of a minus term to this oscillator's differential equation i.e. its rate of change of phase:

$$\dot{\theta_1} = \omega_1^o - k_{1,2} f(\theta_1, \theta_2) \tag{4.3}$$

where $k_{1,2} > 0$ is a constant representing the ability of oscillator 1 to slow down. It can also be interpreted as a measure of interaction (coupling strength) from oscillator 2 to oscillator 1. If the coupling is stronger, the interaction oscillator 1 can slow down more. $f(\theta_1, \theta_2) > 0$ is a function describing the nature of the slowdown. It needs to be $2\pi$ periodic

since its operands are phase. Following the same nature of slow down, oscillator 2 should be accelerating to catch up with oscillator 1 i.e. a positive term in its governing differential equation. We can write this equation, similar to Equation 4.3:

$$\dot{\theta_2} = \omega_2^o + k_{2,1} f(\theta_1, \theta_2) \tag{4.4}$$

If we select an odd function for $f$, we can drop the $+$ and $-$ signs since $f(x) = -f(-x)$ if $f$ is odd. Equations 4.3 and 4.4 reduce to:

$$\dot{\theta_i} = \omega_i^o + k_{i,j} f(\theta_j, \theta_i) \tag{4.5}$$

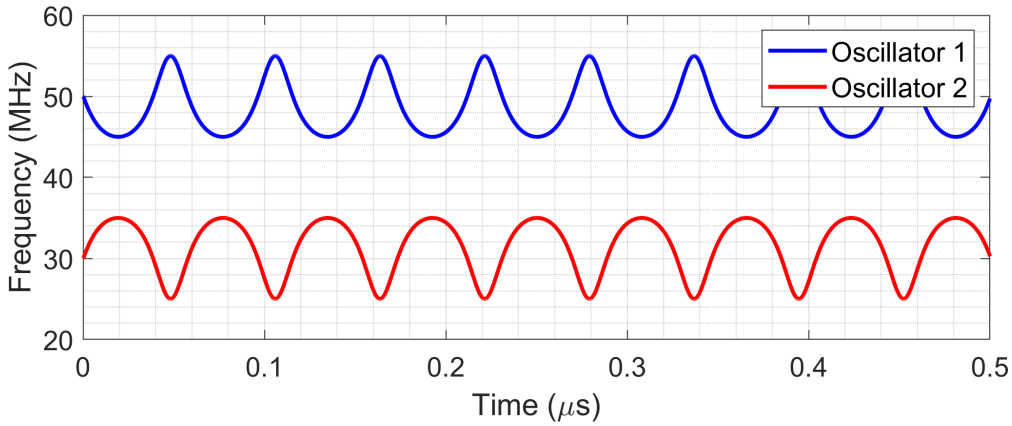
The simplest $2\pi$ periodic odd function for $f$ would be $sin(\theta_2 - \theta_1)$ [32]. Therefore, we reach:

$$\begin{cases} \dot{\theta_1} = \omega_1^o + k_{1,2} sin(\theta_2 - \theta_1) \\ \dot{\theta_2} = \omega_2^o + k_{2,1} sin(\theta_1 - \theta_2) \end{cases} \tag{4.6}$$

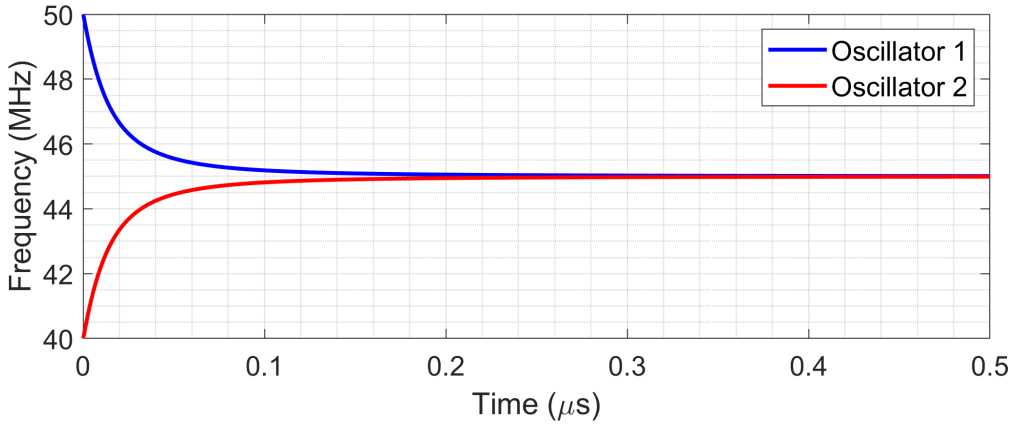
Equation 4.6 is the famous Kuramoto model that describes the synchronization of two oscillators. $\theta_i$ and $\omega_i^o$ are the phase and natural frequency of the two oscillators, respectively. $k_{i,j}$ is the coupling strength from oscillator $j$ to oscillator $i$. This model can also be generalized to multiple oscillators:

$$\dot{\theta_i} = \omega_i^o + \sum_{j}^{N} k_{i,j} sin(\theta_j - \theta_i) \tag{4.7}$$

Note that the time derivative of phase gives the instantaneous frequency of oscillators as a function of time i.e. $\omega = \dot{\theta}$. To get a feeling of how coupling works for different oscillation frequencies, we will provide solutions of Equation 4.6. We solved this differential equation system using MATLAB's built-in differential equation solver (ode45). Figure 4.2 depicts how the frequency of two coupled oscillators change over time for two cases. In




**(a)**




**(b)**

**Figure 4.2:** (a) Solution of the Kuramoto equations for a coupled oscillator pair with natural frequencies $f_1^o = 50$ MHz and $f_2^o = 30$ MHz. The coupling strengths are $k_{i,j} = 2\pi \times 5$ MHz. The oscillators cannot synchronize because of large natural frequency mismatch. (b) Solution when $f_1^o = 50$ MHz and $f_2^o = 40$ MHz with the same coupling strengths. The two oscillators synchronize. (The differential equations are solved using $4^{\text{th}}$-order Runge-Kutta algorithm on Matlab.)

both cases, the coupling strengths are $k_{1,2} = k_{2,1} = 2\pi \times 5$ MHz. Figure 4.2 (a), $f_1^o = 50$ MHz and $f_2^o = 30$ MHz. We see that the frequencies of the oscillators fluctuate around their natural frequencies, but they do not synchronize since the detuning is too large. In the second example (Figure 4.2 (b)), $f_2^o = 40$ MHz. During a brief transient, oscillator 1 slows down and oscillator 2 accelerates. They eventually synchronize at 45 MHz. Note that for the simulations and the graphs we used linear frequency ($f$) because it's more intuitive.

We repeated this experiment with different values of $f_2^o$ between 30 MHz and 70 MHz.

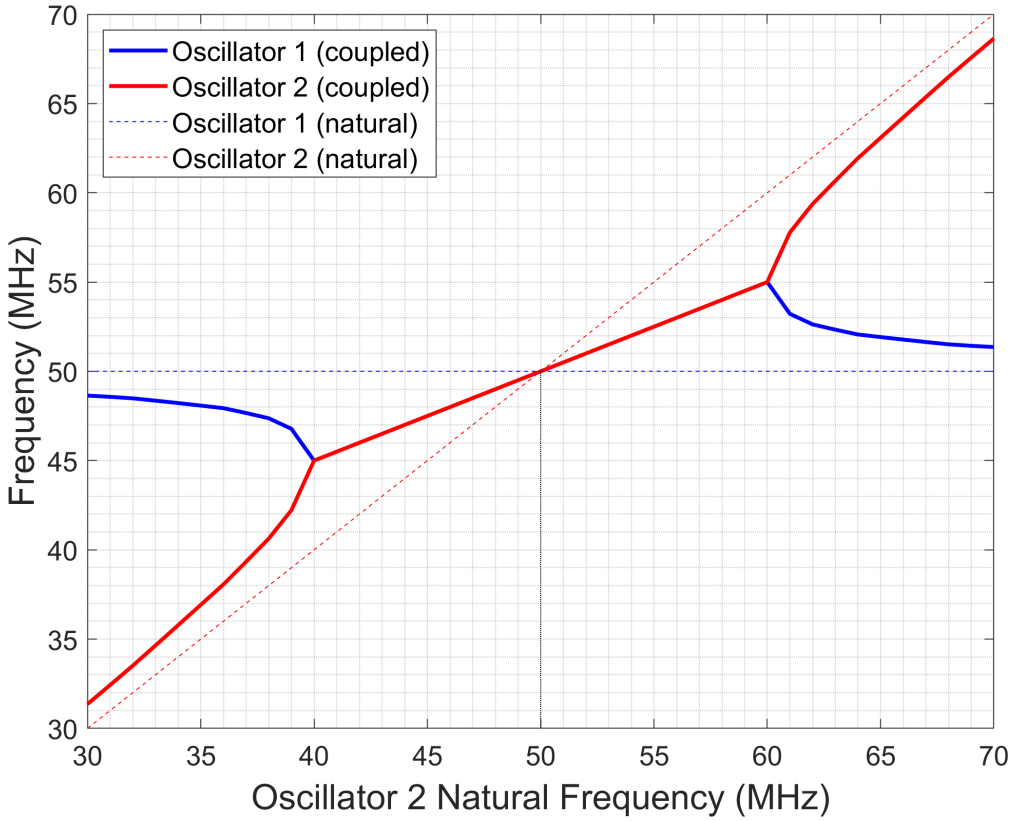



**Figure 4.3:** Synchronization behavior of Kuramoto oscillators. We solved the Kuramoto equation for values of $f_2^o$ between 30 MHz and 70 MHz while keeping $f_1^o = 50$ MHz. When detuning ($\Delta f^o \leq 10$ MHz, the two oscillators synchronize. This maximum allowed detuning is actually the sum of two coupling strengths $k_{i,j} = 2\pi \times 5$ MHz.)

The steady state mean frequencies of the oscillators are plotted in Figure 4.3. In this graph, we see the behavior that we described earlier: if the discrepancy between the natural frequencies are large, the oscillators do not synchronize. As the two frequencies get close to each other, synchronization occurs within a certain range. This window is referred to as *synchronization range* [33].

If we reconsider Equation 4.6, for synchronization i.e. for $\omega_1 = \omega_2$, $\dot{\theta}_1 = \dot{\theta}_2$. In this case, we can write

$$\Delta\omega^o = \omega_2^o - \omega_1^o = (k_{1,2} + k_{2,1})sin(\theta_2 - \theta_1) \tag{4.8}$$

The maximum and minimum values that the sine term on the RHS attain are $\pm 1$ for $\Delta\theta = \pm 90°$. Therefore, maximum detuning allowed for synchronization becomes $\Delta\omega^o = \pm k_{1,2} +$

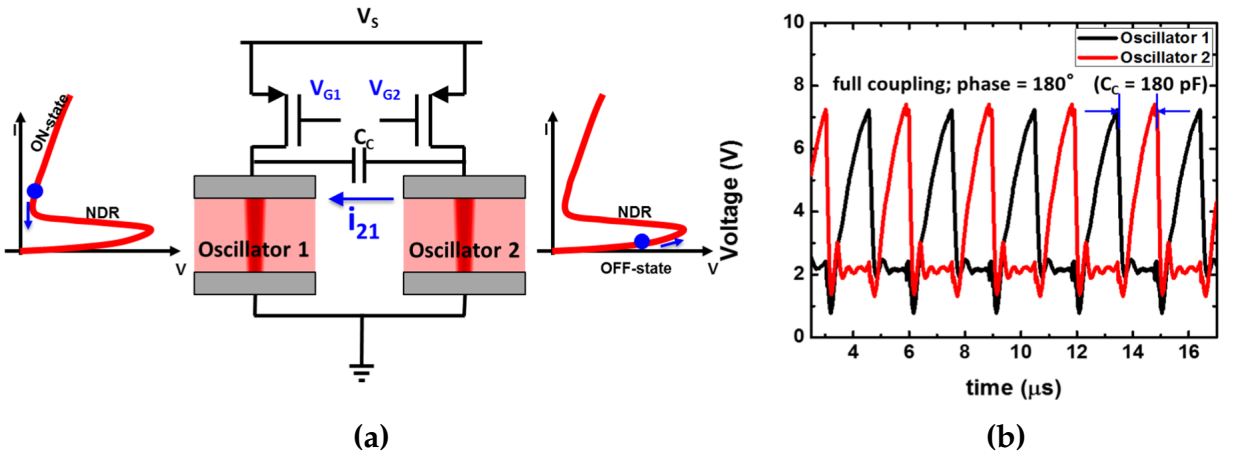



**Figure 4.4:** (a) S-NDR oscillator pair coupled with a capacitor. (b) The voltage oscillations of the coupled S-NDR oscillators show a 180° relative phase difference. Figures were adapted from [15].

$k_{2,1}$ and the phase difference in these cases becomes $\Delta\theta = \pm 90°$. If we solve Equation 4.8 for $sin(\theta_2 - \theta_1)$ and substitute that back into Equation 4.6, we can show that the frequency the oscillators will *agree* is a weighted average of $\omega_1$ and $\omega_2$ with the weights $k_{2,1}$ and $k_{1,2}$, respectively:

$$\omega = \frac{\omega_1 k_{2,1} + \omega_2 k_{1,2}}{\omega_1 + \omega_2} \tag{4.9}$$

## 4.2 Capacitively Coupled S-NDR Oscillators

With Kuramoto's model, we established an understanding of oscillator couplings. Next we consider coupling the S-NDR oscillators introduced in the previous section. At Carnegie Mellon, Abhishek Sharma demonstrated first experimental results of coupled S-NDR oscillators by coupling them through capacitors [15]. When the output nodes of two S-NDR oscillators connected through a capacitor as shown in Figure 4.4 (a), the two oscillators synchronize by assuming a phase difference of 180°. The output voltages of the two oscillators are shown in Figure 4.4 (b). The physics of the two devices lead to the stable 180° phase difference: Let's assume that at some time $t_o$, oscillator 1 and 2 are both in the OFF-state with some phase difference such that $V_1^{out} > V_2^{out}$ i.e. $\theta_1^o > \theta_2^o$ and $\Delta\theta^o < 180°$. In the

OFF-state, both outputs are charging and the displacement current across the coupling capacitor is small. However, oscillator 1 will reach the threshold voltage earlier (given its head-start) and switch to the ON-state. As soon as this switching happens, its output 1 will start discharging rapidly. This leads to a large displacement current from oscillator 2 to oscillator 1. The extra current supplied by the coupling capacitor continues to keep oscillator 1 in the ON-state. This trend changes once oscillator 2 reaches the threshold voltage and switches. The rapid decrease of $V_{out}^2$ leads to the flow of displacement current into oscillator 2 and oscillator 1 can finally switch to the OFF-state. Therefore, because of the capacitive coupling, the oscillators switch at the same time yet in opposite directions. This leads to the 180° phase difference. The behavior is similar to *kick synchronization* with $\epsilon < 0$ discussed in [34]. The switching of one oscillator creates an impulse of current through the coupling capacitor (kick) which causes a delay in the phase of the second oscillator ($\epsilon < 0$). This behavior can also be generalized to other types of relaxation oscillators and its mathematical foundations are discussed by Kopell and Somers in [35].

We repeated the natural oscillation frequency sweep experiment with S-NDR oscillators using SPICE simulations. Using the gate voltage of the ballast transistor in the van der Pol circuit model described earlier, we swept the natural frequency of one oscillator while keeping the other one constant. Even though the stable phases are different than those of Kuramoto model, the overall coupling behavior is similar. As depicted in Figure 4.5, there is a synchronization range and the size of this region depends on the coupling capacitance.

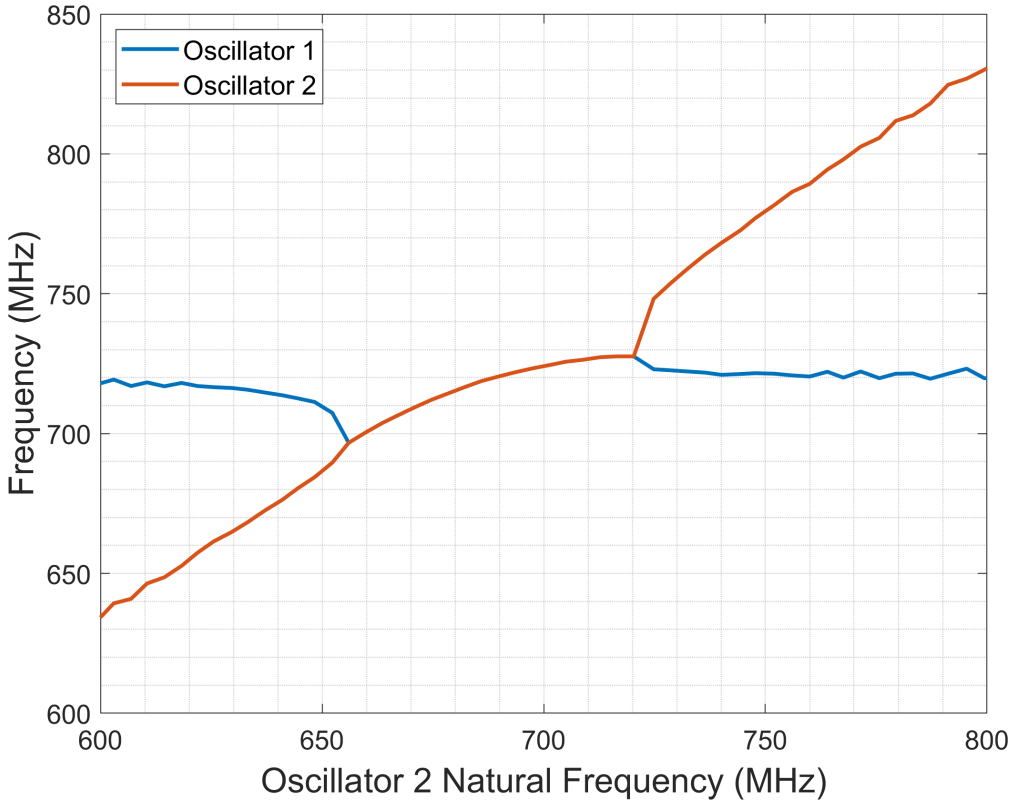



**Figure 4.5:** SPICE simulation of capacitively coupled S-NDR oscillator pair reveal that they couple in the same way Kuramoto oscillators do.

## 4.3 Forced Oscillations and Injection Locking

So far we considered coupling two free running oscillators where the coupling worked in both directions (non-zero $k_{i,j}$). Synchronization can also occur if coupling is in one direction. For example, most mammals have a circadian rhythm 24 hours, locked onto the periodic sunlight exposure. On a similar note, we can consider an oscillator driven by an AC signal whose frequency is close to the natural frequency of the oscillator. In that case, the oscillator can lock to the AC signal and synchronize. Higher order synchronizations can also occur: suppose the driving AC signal's frequency is $2\omega$ where $\omega$ is close to the natural oscillation frequency of the oscillator. In this case, the oscillator can lock to frequency $\omega$ i.e. exactly half of the driving AC signal's frequency (Figure 4.6). This phenomenon is referred to as sub-harmonic injection locking (SHIL) [36]. In fact, synchronizations of orders *n:m* can be observed where $n/m$ is the ratio of driving AC frequency to natural oscillation frequency, for integer values of n and m [33]. Sub-harmonic injection locking is often used in frequency synthesizers [37,38].

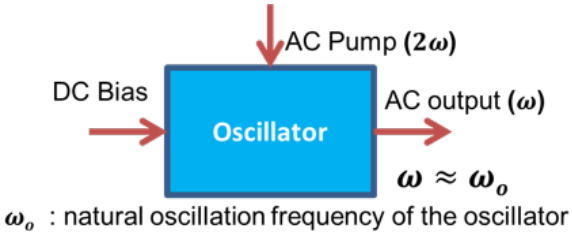



**Figure 4.6:** Sub-harmonic injection locking generates output signals ($\omega$) which oscillate at the half of the input AC signal's frequency ($2\omega$). This will happen only if $\omega \approx \omega_o$ where $\omega_o$ is the natural oscillation frequency of the oscillator, without the AC input.

We experimentally demonstrated 2:1 frequency locking with S-NDR oscillators. When the gate of the ballast transistor is driven at 28 MHz, we observed oscillations at 14 MHz (Figure 4.7).

What is more interesting, when sub-harmonic injection locking of order 2:1 occurs,

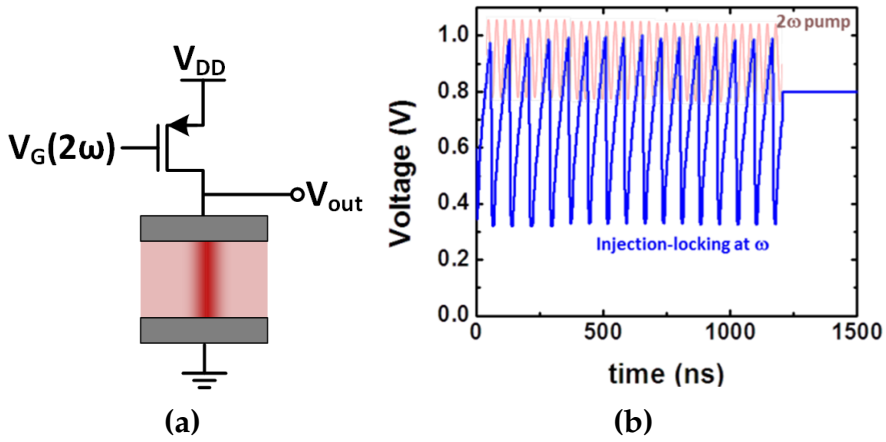



**Figure 4.7:** (a) Sub-harmonic injection locking can be achieved by applying the AC signal to the ballast transistor's gate in the 1T1R configuration. (b) Experimental verification of SHIL in S-NDR oscillators.

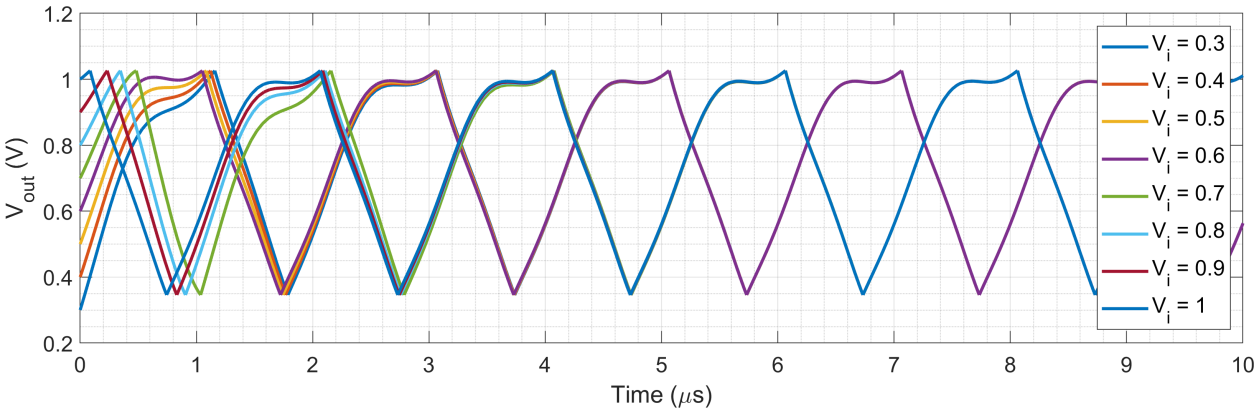



**Figure 4.8:** The SPICE simulations of S-NDR oscillators confirm that SHIL results in two different phases at the output depending on the initial (pre-charged) voltage of the output node. The legends show these initial voltages.

there are two stable out of phase outputs are available. Depending on the initial phase of the output, the output phase can be either 0° or 180°. For S-NDR oscillators, the initial phase can be set by precharging the output node of the oscillator. Fig. 4.8 presents the SPICE simulation results showing that any initial voltage of this node will result in locking to one of the stable phases. When the initial voltage is between 0.3 and 0.6 volts, the oscillator will lock the one phase, and it will lock to the other allowed phase if the initial voltage is 0.7-1 V. This interesting property will be utilized in our phase-based computation and edge detection applications to encode bit values or pixel values.

## 4.4 Phase Based Computation

In traditional level based logic, logic 1 and 0 are encoded in voltage levels which are generally VDD and VSS (0 V) of the circuit, respectively. In the phase based logic, the information is encoded into the phase of an oscillating waveform, 0Âř for logic 0° and 180° for logic 1 (Figure 4.9). The required phases can be generated out of oscillators thanks to a nonlinear phenomenon named sub-harmonic injection locking (SHIL).

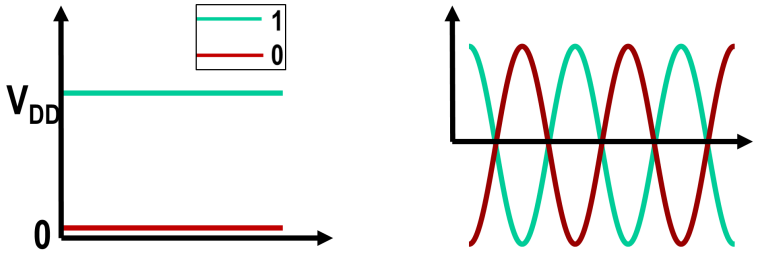
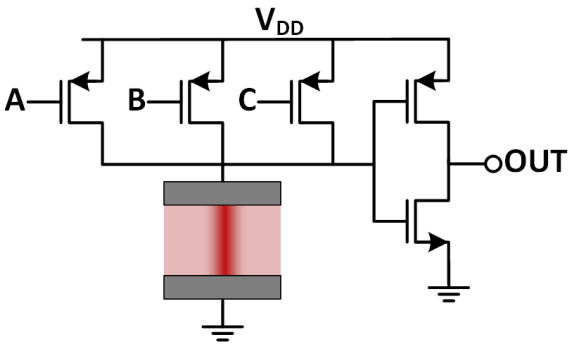



**Figure 4.9:** (a) In traditional logic, boolean states are encoded in voltage levels. (b) In phase based logic paradigm, the states are encoded in the phase of an oscillating waveform.

The idea of phase based logic was developed independently by two scientists around the same time, in 1950s. John von Neumann developed the idea and proposed oscillatory circuits that make use of nonlinear capacitors and inductors [39]. Eiichi Goto proposed the same idea and built computers that are based on the phase based computation paradigm [40]. These computers were named *Parametron* and nearly half of the computers in Japan were actually parametrons [40]. At that time, parametrons were superior to vacuum tube based computers since they were more reliable and robust. Eventually, Silicone CMOS technology not only eliminated the vacuum tube but also the parametrons.

In order to implement any truth table in Boolean logic, a set of functionally complete gates is required. In phase based logic, the most suitable set of gates is MAJ, NOT where MAJ represents a three input gate that returns Boolean value that occurs more than once among its inputs [41]. MAJ gate relies on the analog summation of input signals: if the




(a)

| A | B | C | OUT |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

(b)

**Figure 4.10:** a) MAJ gate design using $GeTe_6$ oscillator (b) The truth table. If the input A is 0, OUT = B AND C. For A = 1, OUT = B OR C. This demonstrates the functional completeness of {MAJ, NOT}

two of its inputs are different, they effectively cancel each other since summation of two out of phase signals is DC. If all three inputs are the same, then they add up. In this case, some form of amplitude limiting is required [41]. S-NDR nano-oscillators lend themselves to phase-based computation naturally because: 1) the ballast transistors can be used to convert input voltage signals to current signals which can be summed easily at a common node and 2) the amplitude of the oscillations are already limited by the threshold and holding voltages, independent of the input.

To achieve the MAJ gate using S-NDR oscillators, we connected 3 ballast transistors to an oscillator instead of 1 as shown in Figure 4.10 (a) [42]. The input signals (A-C) are applied to the gates of the transistors. A CMOS inverter is connected to the output of the oscillator which serves 2 purposes: 1) as discussed above, the output needs to be quantized to prevent phase shifts and 2) because of the inverting property of the PMOS ballast in the oscillators, the output of the oscillator shows a MIN characteristic, instead of MAJ. The CMOS inverter is required for MAJ operation. The NOT gate of the phase-based logic is implemented using a regular CMOS inverter. Bit flip can be achieved by momentarily slowing down the oscillations at the output node using a control signal.

Once the MAJ gate is achieved, all the basic gates can be generated using MAJ and

NOT. Figure 4.10 (b) shows the truth table. Notice that when one of the inputs is 0, the output is the AND of the other two inputs. If one of the inputs is 1, the output is the OR of the other inputs. Using AND, OR and NOT; NAND and NOR can be built. An XOR gate can be achieved using 4 NAND gates. This shows that {MAJ, NOT} is a functionally complete set of gates.

For demonstration of a functional block, a full adder was created and simulated (Figure 4.11). Input A was 0 during the simulation time, whereas input B was 1 for $0 < t < 25$ ns. For $t > 25$ ns, input B is 0. The carry-in input ($C_i$) was 1. Figure 4.11 (b) overlays the output signals, S and carry-out ($C_o$) on top of logic 0 signal for reference. Output S is in phase with logic 0 for $t < 25$ ns (S = 0) and out of phase with it for $t > 25$ ns (S = 1, after a brief transient). Carry-out is out of phase with logic 0 for $t < 25$ ns (Co = 1) and in phase with it for $t > 25$ ns (Co = 0, after the transient).

The average power of the phase-based full-adder was estimated to be around 750 µW. For comparison, we also simulated a regular CMOS full-adder implemented at 65 nm technology node and the average power was 1 µW. However, for a better comparison, we need to consider the power scaling of the nano-oscillators as they scale down in size. In a scaling scenario, the OFF-state current scales down linearly with device area. The scaling of the ON current, on the other hand, is not straightforward as conduction in the ON-state is not uniform, but filamentary. If we assume uniform conduction for simplicity, for device sizes around 100 nm $\times$ 80 nm (comparable to minimum transistor sizing at 65 nm technology node), dimensions scale by 50 and power scales down by 2500 times, bringing the full adder power to 150 nW.

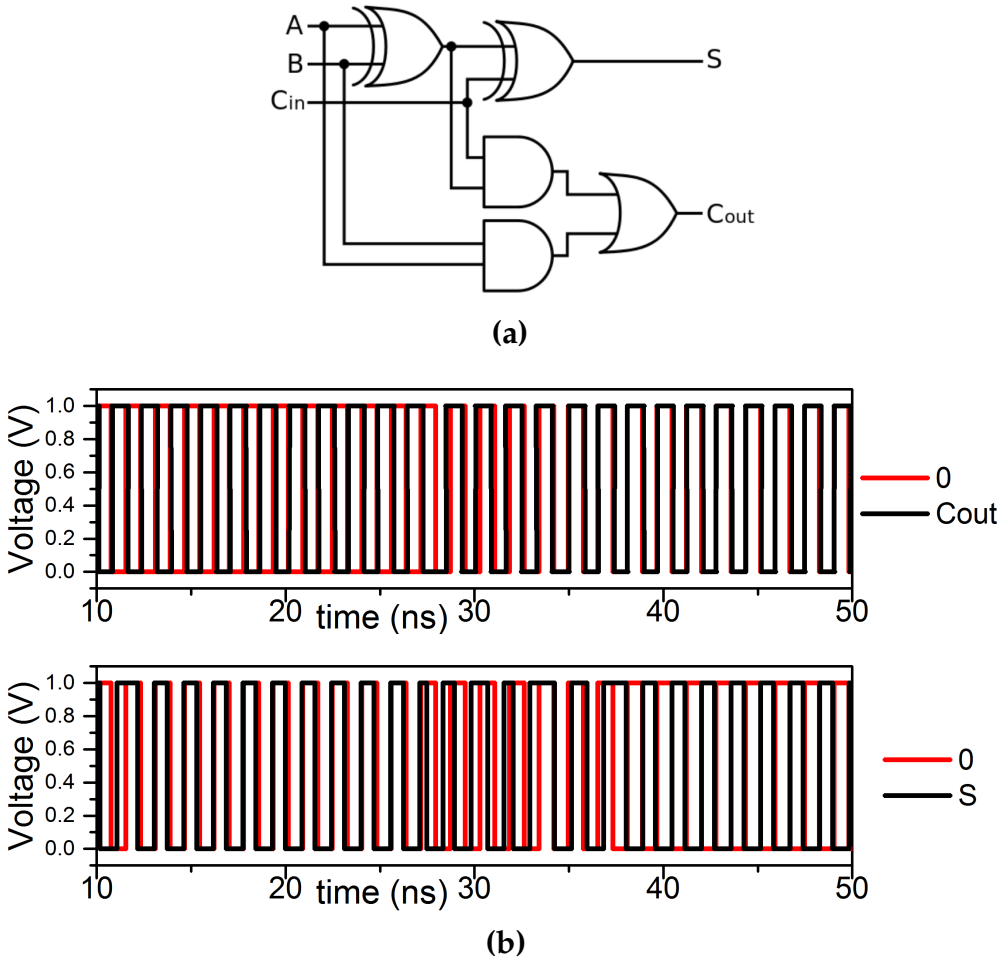



**Figure 4.11:** (a) Full adder design. The XOR and AND gates were designed using MAJ gate. (b) resulting waveforms overlaid with logic 0 for reference. Input A is 0 for the whole time as opposed to signal B which is 1 for 25 ns and flips to 0 afterwards. Carry-in ($C_i$) input is 1. Output S is in phase with logic 0 for $t < 25$ ns ($S = 0$) and out of phase with it for $t > 25$ ns ($S = 1$, after a brief transient). Carry-out is out of phase with logic 0 and for $t < 25$ ns ($C_o = 1$) and in phase with it for $t > 25$ ns ($C_o = 0$, after the transient).

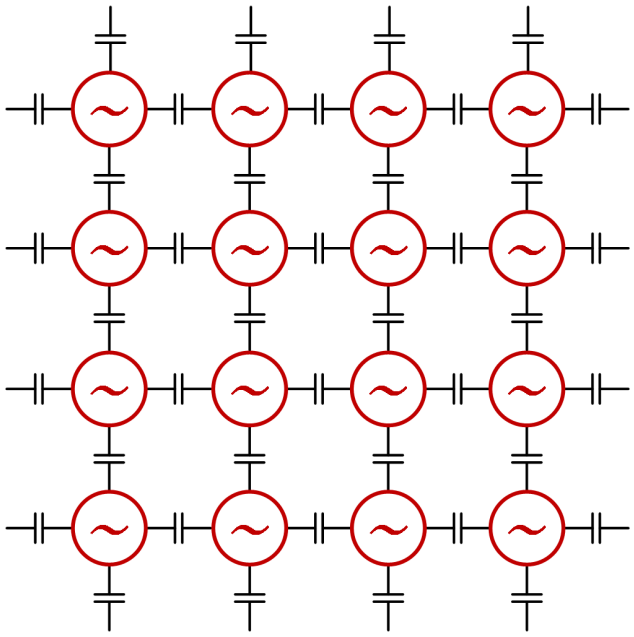


**Figure 4.12:** Schematic of the directly-coupled oscillator network for which the coupling element is a capacitor. Each oscillator is coupled to its 4 nearest neighbors. Only a portion of the network is shown.

## 4.5 Coupled Oscillator Networks

In this section, we will start investigating the functionality of coupled S-NDR oscillator networks. Specifically, we will be looking at solving two image processing problems: edge detection and stereo vision. Inspired by the synchronization behavior of oscillators found in nature, we are proposing the nearest-neighbor capacitively-coupled oscillator network that is shown on Figure 4.12. The network is simulated in SPICE and shown to solve for edge detection and stereo vision problems which will be presented in the following sections.

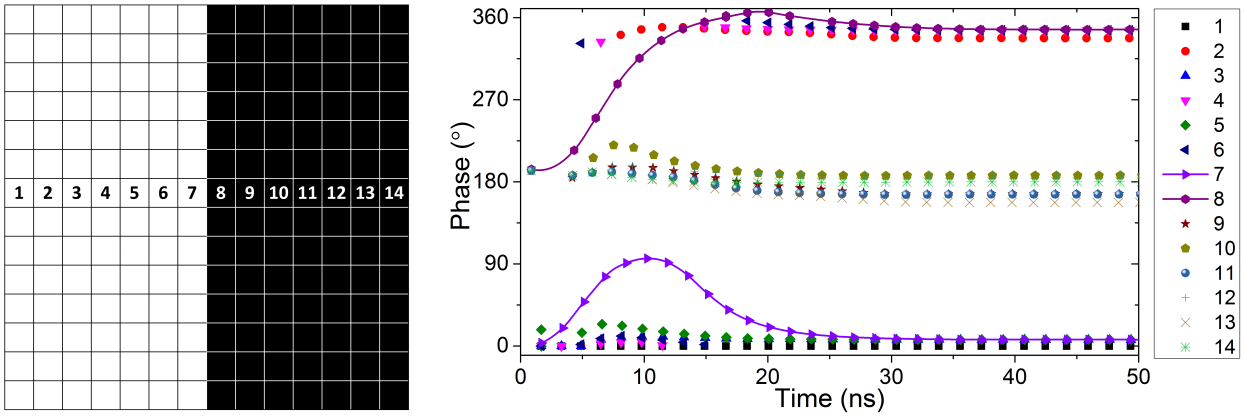



**Figure 4.13:** (a) 14×14 network used for the simulations. (b) shows the time evolution of the phases of the oscillators numbered 1-14 in (a).

## 4.6 Edge Detection

Edge-detection is a graphics processing method where the goal is finding the boundaries of objects in a scene. Similar to other image processing problems, image processing too suffers from the need for frequent memory access. The parallel nature of coupled oscillator networks is therefore promising for edge detection. In fact, [43] and [44] reports networks of single-electron tunneling junction oscillators solving edge detection problems. Both of these works are simulations of capacitively 4-nearest neighbor coupled oscillator networks, similar to the one shown in Figure 4.12. We have used the same approach to simulate a network of coupled S-NDR oscillators in SPICE, using the model we developed.

In this edge detection implementation, each pixel in the input image corresponds to an oscillator in the network. In order to encode image into the network, we are using sub-harmonic injection locking (SHIL). The pixel value of the input image is translated to the oscillator network as the initial phases of the oscillators (phase-shift-keying). For simplicity, we have chosen a binary image as shown on Figure 4.13 (a) where the left half of the image is white and the right is black. For the white pixels, the initial voltage of

the oscillators in the 14×14 network are set to 1 V, and for the black pixels, the initial voltage is 0.1 volts. After the image is encoded into the network, a transient simulation is conducted. Before each oscillator in the network settles to one of the two stable phases (0° and 180°), the oscillators corresponding to the edge pixels are clearly differentiable in terms of their phase evolution from the rest of the circuit due to the loading effect from the neighbors. For oscillators within the regions i.e. far from the cells, the initial phases of the neigboring oscillators are the same. Therefore, the displacement current passing through the coupling capacitor ($C_c$) between them is limited and the two oscillators do not interact with each other. In this case, the loading effect between the oscillators is limited and the phase evolution of the oscillators follow their natural course. For oscillators at the edges, the case is different. Because the two oscillators are in different regions. they have different initial phases. When one oscillator is charging, the other one will be discharging which will result in a large $dV/dt$ across the coupling capacitor. This leads to a large displacement current across $C_c$ and the edge oscillators experience a loading effect. This effect will cause the oscillator to momentarily slow-down and the oscillators will experience a lag during their phase evolution. Due to this lag, the edge cells become distinguishable. Figure 4.13 (b) shows the phases of 14 pixels along a row of the 14×14 network that is given in Figure 4.13 (a). Pixels numbered 1-7 and 8-14 belongs to different segments of the image and pixels 7 and 8 are the neighbors at the boundary. While oscillators settle to their final phases around 0 or 180, the process is slower for the edge pixels (7,8) and they become distinguishable. The output image can be constructed by converting the phase values to pixel values using a threshold function similar to Equation 4.10:

$$f(\phi) = \begin{cases} 0, & \text{if } -45 < \phi \leq 45 \text{ or } 135 < \phi \leq 225 \\ 1, & \text{if } 45 < \phi \leq 135 \text{ or } 225 < \phi \leq 315 \end{cases} \tag{4.10}$$

This function will convert the edge cells to binary 1 and non-edge cells to 0. Figure 4.14

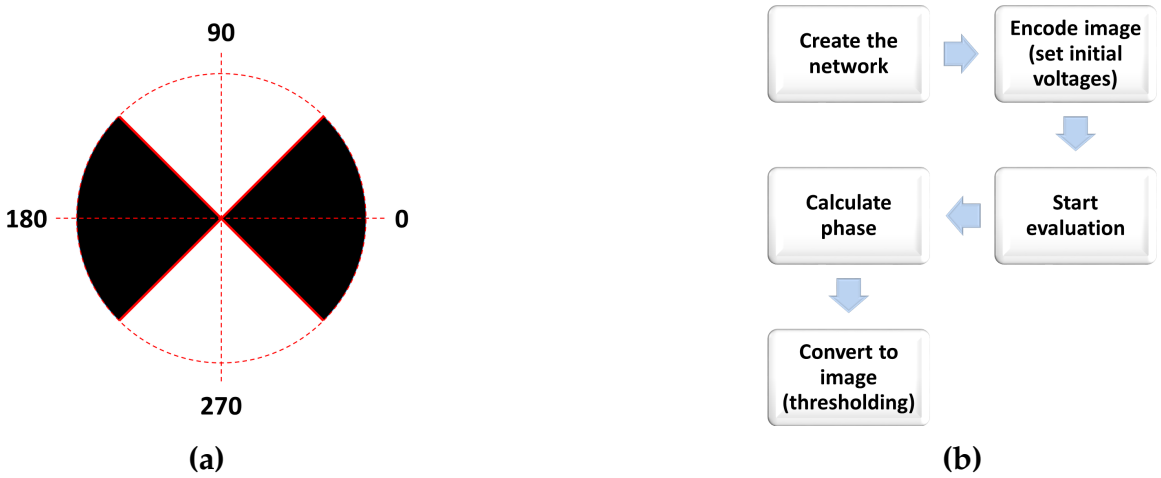



**Figure 4.14:** (a) Graphical representation of the thresholding function. (b) Edge detection flowchart using coupled oscillator network.

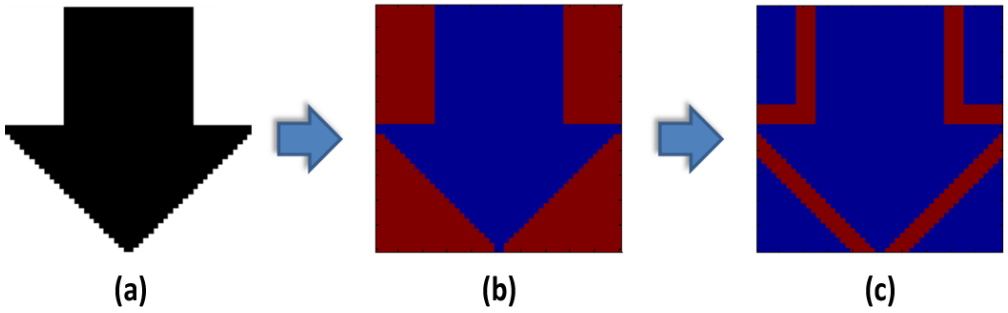



**Figure 4.15:** (a) Input binary image (b) image is encoded in initial phases and (c) after the evaluation, the edges of the feature are detected.

(a) shows a graphical representation of the function. Figure 4.14 (b) summarizes the edge detection application using coupled oscillators.

Figure 4.15 shows a 50×50 image that we processed using the oscillator network. To process color images, each color component (RGB) can be processed separately.

The coupling strength, $C_c$, has a critical role in the oscillator network based edge detection technique. If $C_c$ is very small, the coupling is too weak even if the neighboring oscillators are out-of-phase. If $C_c$ is very large, the coupling becomes too strong and the oscillators cannot lock to the driving AC signal anymore. Therefore, the coupling capacitance determines the strength of interaction between the oscillators. This affects how

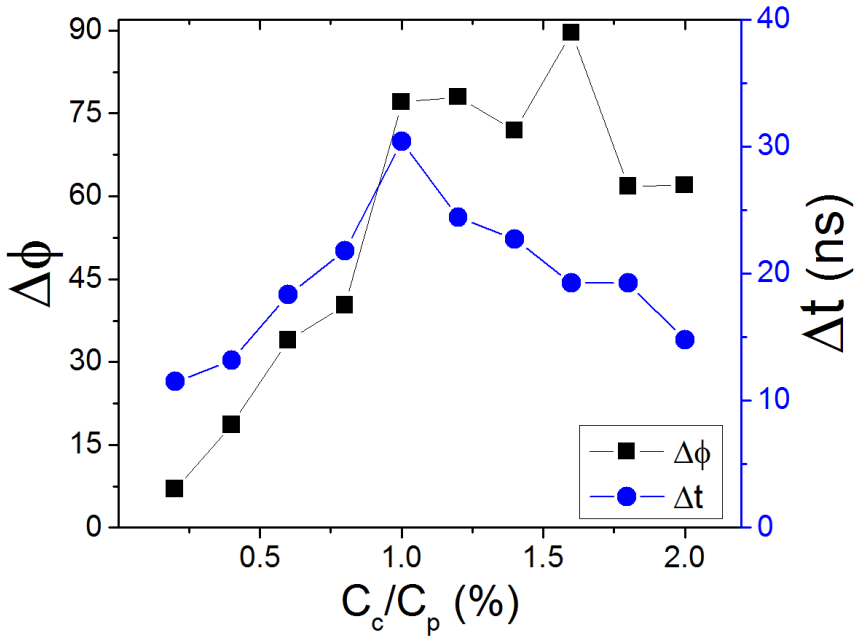



**Figure 4.16:** Phase separation ($\Delta\phi$) and evaluation window ($\Delta t$) vs coupling capacitance ($C_c$).

much the edge cells are separated from the others in terms of phase ($\Delta\phi$). $C_c$ also determines how fast the networks settles and reach synchronization. Therefore, it also affects the evaluation window ($\Delta t$). For the case pictured in Figure 4.13, the evaluation should take place between 5 and 15 ns. Figure 4.16 shows how the coupling capacitance affects the phase difference ($\Delta\phi$) and evaluation window ($\Delta t$). The results indicate that the coupling capacitor ($C_c$) should be small compared to the parasitic capacitance ($C_p$) of the oscillators for weak coupling, a $C_c/C_p$ ratio of 1% being the most advantageous in terms of $\Delta\phi$ and $\Delta t$.

Based on Figure 4.13, we can state that edge detection can be carried out within 10 oscillation cycles. Considering that the leakage current go down as the devices scale and the parasitics are reduced (discussed in more detail in Section V), the evaluation time can be as low as 3.5 ns (assuming 3 GHz operation) with an energy expense of a few pJ. Compared to the FPGA implementation discussed in [45], the oscillator network can provide energy efficiency up to 6 orders of magnitude. However, itś noteworthy that we calculate the energy of the oscillator network assuming that it is only on during the edge

detection operation. Moreover, the power consumption of the other circuits that will be used to program the network and the read-out circuitry is not simulated.

## 4.7 Stereo Vision

Stereo vision (3D-vision) is the process of extracting depth information from images. Comparing two images of a scene taken from slightly different angles, stereo vision algorithms in computer vision can create a depth map of the objects in the scene. Efficient stereo implementations are especially drawing interest these days as autonomous robotic systems including self-driving cars are being widely studied [46,47].

Conventional solutions to stereo vision problem in computer vision involves a process of belief propagation where each node (pixel) conveys the depth information to a neighboring node [48]. An iteration consists of a raster scan that is required to update the depth information of all the pixels. After a number of iterations, the depth information does not change beyond a threshold and the algorithm converges. There are two factors determining the convergence: the data cost and the smoothness cost. The data cost determines how important any given pixel is, and the smoothness cost determines how much of the noise will be left out. The algorithm converges quicker for lower data cost and higher smoothness cost, yet, more information is lost. The main challenge in implementing these algorithms manifests itself as the speed reduction and power consumption due to the frequent need to access memory for the raster scan and the multiple iterations [49].

The coupled-oscillator network solution that we are proposing for stereo vision makes use of the same capacitively-coupled network shown in Figure 4.12. The network was designed and simulated in collaboration with Abhishek A. Sharma. A disparity image, that is the color difference between the two cameras (right and left) used for stereo vision, is

created. The objects that are closer to the cameras will have more discrepancy and each oscillator in the network corresponds to one pixel of the disparity image. The pixel values of the disparity image is encoded into the network in the form of natural oscillation frequencies (frequency-shift-keying). To achieve that, the pixel values between 0-255 are mapped on to the gate voltage $V_G$ over 0-1 V using the transistor-ballasted oscillator scheme. In this case, oscillators belonging to the same object will have similar gate voltages and natural oscillation frequencies. Neighboring oscillators with similar natural oscillation frequencies tend to synchronize as the network settles. Groups of synchronized oscillators will define the segments in the image and the unsynchronized oscillators at the boundary of neighboring segments form the edges. Thus, the oscillators that correspond to the same object will lock together and oscillate at the same frequency. As the objects closer to the camera are encoded using higher $V_{GS}$ because of larger discrepancy, those oscillators will oscillate at a higher frequency compared to the objects further away. After the network settles, the frequencies of the oscillators are converted back to pixel values to create the disparity map. Figure 4.17 shows the left and right images and the raw disparity image of the Tsukuba benchmark from the Middlebury set [48]. The pixel values are fed into a $128 \times 128$ network and the network settles to the disparity map shown on the bottom right in Figure 4.17.

In this network implementation, the data cost is represented by the oscillation frequency and the smoothness cost is determined by the coupling capacitor, $C_c$. For larger values of $C_c$, the network settles quicker but the features are smoothened out as higher values of $C_c$ will create a stronger coupling. For an average oscillation frequency of 1 GHz and $C_c$ = 10 fF, the network converges in 24 oscillation cycles.

When compared to the ground truth shown in Figure 4.18 (a), the oscillator network implementation results in a pixel misprediction below 22% whereas the FPGA implementation in [49] shows 8% pixel misprediction. The oscillator networks provides a compute

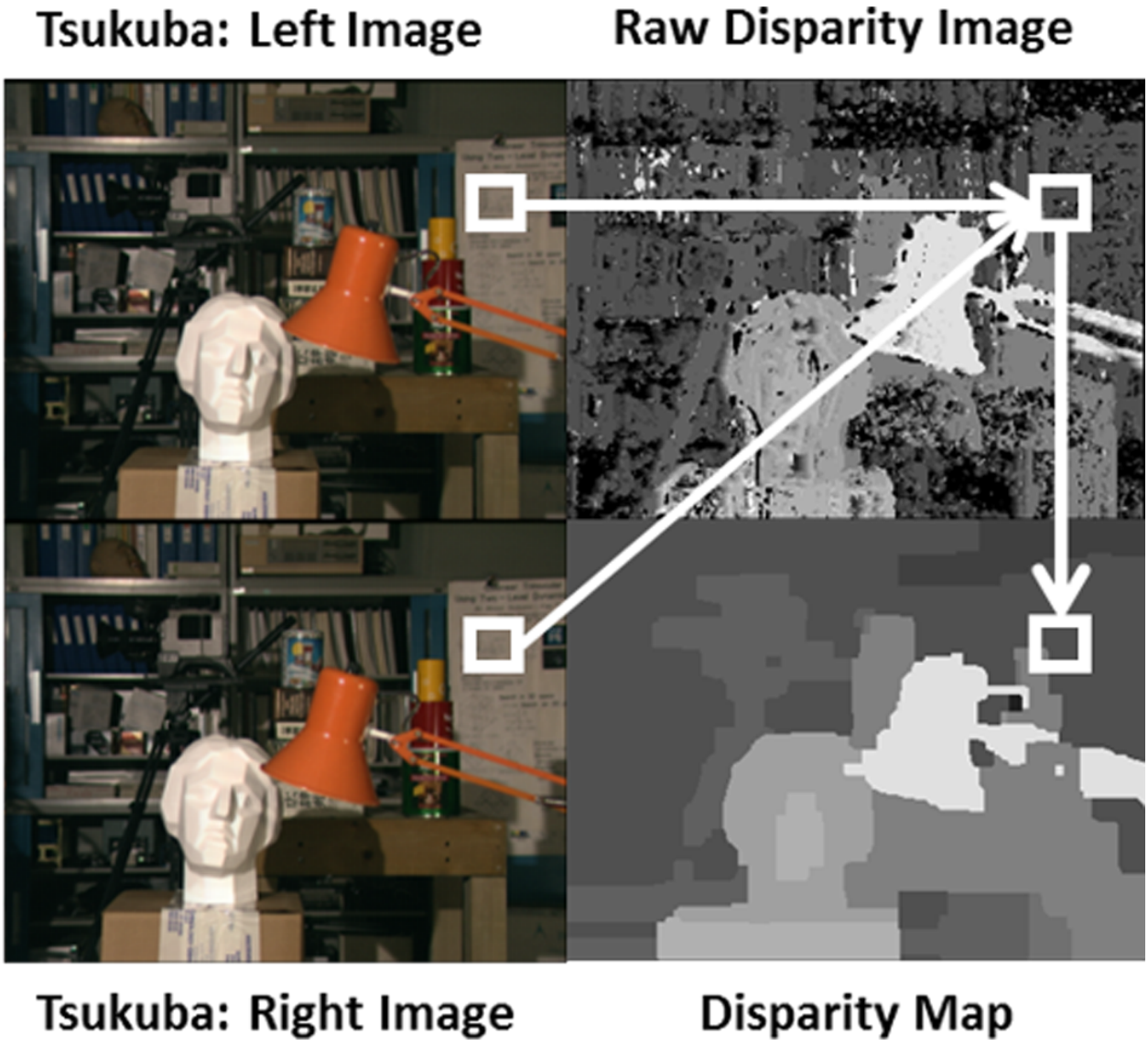



**Figure 4.17:** The left and right images of the Tsukuba benchmark are used to create the disparity image (top right). The disparity image is fed into the network in the form of oscillation frequencies. The network settles to the disparity map (stereo map) shown on the bottom right. In this map, brighter colors represent higher frequency, thus are closer to the camera. Simulations designed by Abhishek A. Sharma [29].

(a) (b) (c)

**Figure 4.18:** Comparison of stereo maps of (a) ground truth, (b) belief propagation using FPGA, and (c) oscillator network. The FPGA implementation [49] mispredicts the 8% of the pixels compared to the ground truth and the oscillator network's misprediction rate is 22%. Simulations designed by Abhishek A. Sharma [29].

speed-up of 128x and energy reduction of 100x compared to [49]. Note that, the energy calculations only span the oscillator network itself, excluding any read-out circuitry that is not simulated.

## 4.8 Scaling and Integration Projection

As previously discussed, the oscillator frequency is largely determined by the parasitic capacitance. While the frequency of the oscillations are at a few MHz level when off-chip, discrete ballast elements are used, it can go as high as 600 MHz when the structures are integrated on-chip [28]. Also, one would expect that the capacitance would reduce further once the device is scaled down. It was shown previously that this class of oscillators can scale down to a few tens of nanometers in size [50]. Using our model and simulations, we project that if parasitic capacitance can be reduced to 10 fF, oscillations at a frequency of a few GHz are possible (Figure 4.19).

Thanks to scaling and engineering the functional material, the power consumption can be lowered, too. The leakage current or the OFF-state current is a non-linear function of the voltage. Reducing the threshold voltage through material and interface engineering there-

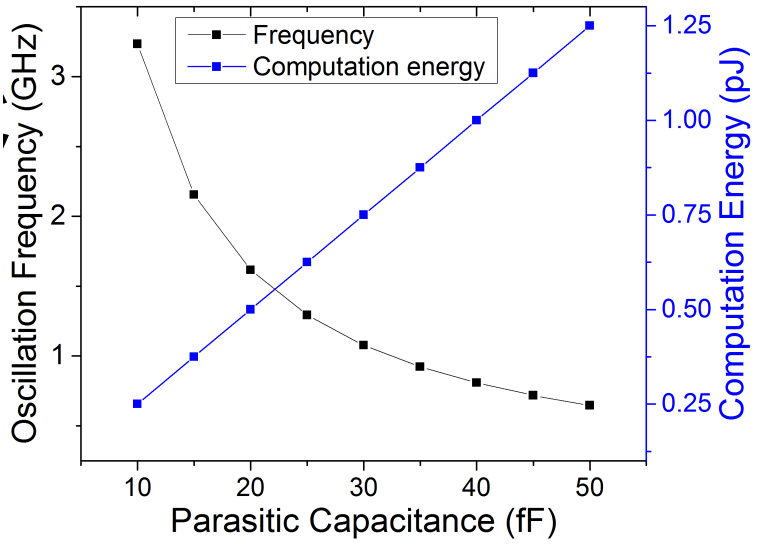



**Figure 4.19:** Simulated oscillation frequency and computation energy vs the parasitic capacitance, $C_p$

fore reduces the leakage current. As the OFF-state conduction is uniform, it goes down linearly with the device area. It was shown previously that this class of oscillators can scale down to a few tens of nanometers in size and sub-100 fJ in switching energy [26]. The effect of scaling and integration on the computation energy budget can be approximated as follows: energy stored in the parasitic capacitance and discharged into the ground in each oscillation cycle is given by . For oscillation amplitude of 1 V, when the parasitic capacitance is in the fF range, say 40 fF, the energy expenditure during one period is 40 fJ. Assuming it takes 25 cycles for a 1000×1000 network to settle to the solution of the image processing problems discussed here, a 1 MP image could be processed spending only 1 pJ energy. This number scales further down for lower parasitic capacitance values as shown on Figure 4.19.

# Chapter 5

# An all-CMOS Oscillator Network

Our S-NDR oscillator network is inspired from the coupled oscillators that are found in nature, and it has similar coupling properties such as synchronization. Similarly, we should be able to apply the concepts that we learned from our simulations to CMOS oscillators as well. Therefore, our goal in designing an all-CMOS oscillator network is to verify these concepts in hardware. By all means we are not aiming to design an oscillator network optimized for area or power efficiency, our chip is designed to provide the flexibility that we would need while investigating the coupled oscillator networks for computation. For example, we are using a digital-to-analog (DAC) converter to bias the voltage controlled oscillators in the network. The power and area requirement of a DAC can easily surpass those of a VCO's. Yet, it offers a reliable solution that can be easily configured to provide the control voltage while we investigate the oscillator network at various frequencies. On a similar note, we used a MIMCAP capacitor bank to provide the coupling between the oscillators, yet MIMCAP is not a dense capacitor, especially when it's compared to a MOS capacitor. However, it rules out the some of the second order effects that would be faced with a MOS capactior, such as the nonlinear capacitance-voltage behavior. This allows us to focus on and understand the fundamentals of the capacitive coupling.Finally, it should

also be noted that we are not proposing a CMOS oscillator network for computation. Our effort is only for a proof-of-concept demonstration of coupled oscillator networks, explore their properties that can be extended to networks of beyond-CMOS oscillators, such as the S-NDR oscillators we discussed earlier.

Each oscillator in our coupled network is a VCO, and in the CMOS network we will use current starved ring oscillator for its simpler design. We designed a ring oscillator at a commercial 65nm process technology and conducted simulations. We created a circuit where two ring oscillator VCOs are coupled through a coupling capacitor. When the control voltage of the VCO is 0.55 V, its natural oscillation frequency (uncoupled) is around 19.8 MHz. After the two oscillators are coupled, the control voltage of oscillator 2 is swept from 0.3 V to 1 V, i.e. sweeping its natural frequency from 4 MHz to 44.8 MHz. The control voltage of oscillator 1 was kept constant at 0.55 V. As the natural oscillation frequency of oscillator 2 approaches that of oscillator 1, shown on Figure 5.1 (b), the two oscillators agree on an average oscillation frequency and they do synchronize.

The synchronization behavior seen in Figure 5.1 is the property we are looking for in the coupled oscillator network applications. Similarity of this coupling behavior in CMOS ring oscillators to that of S-NDR oscillators gives us confidence that two systems are similar in nature. Therefore, the conclusions we draw from CMOS ring oscillators applies to S-NDR oscillators as well.

## 5.1 Chip Overview

We designed and taped-out a CMOS chip at a commercial 65nm process node. The nominal VDD in this process is 1 V. A top level layout of the design is provided in Figure 5.2. Our design is 1mm $\times$ 1mm and has 48 I/O pads, 6 of which is reserved for probing with

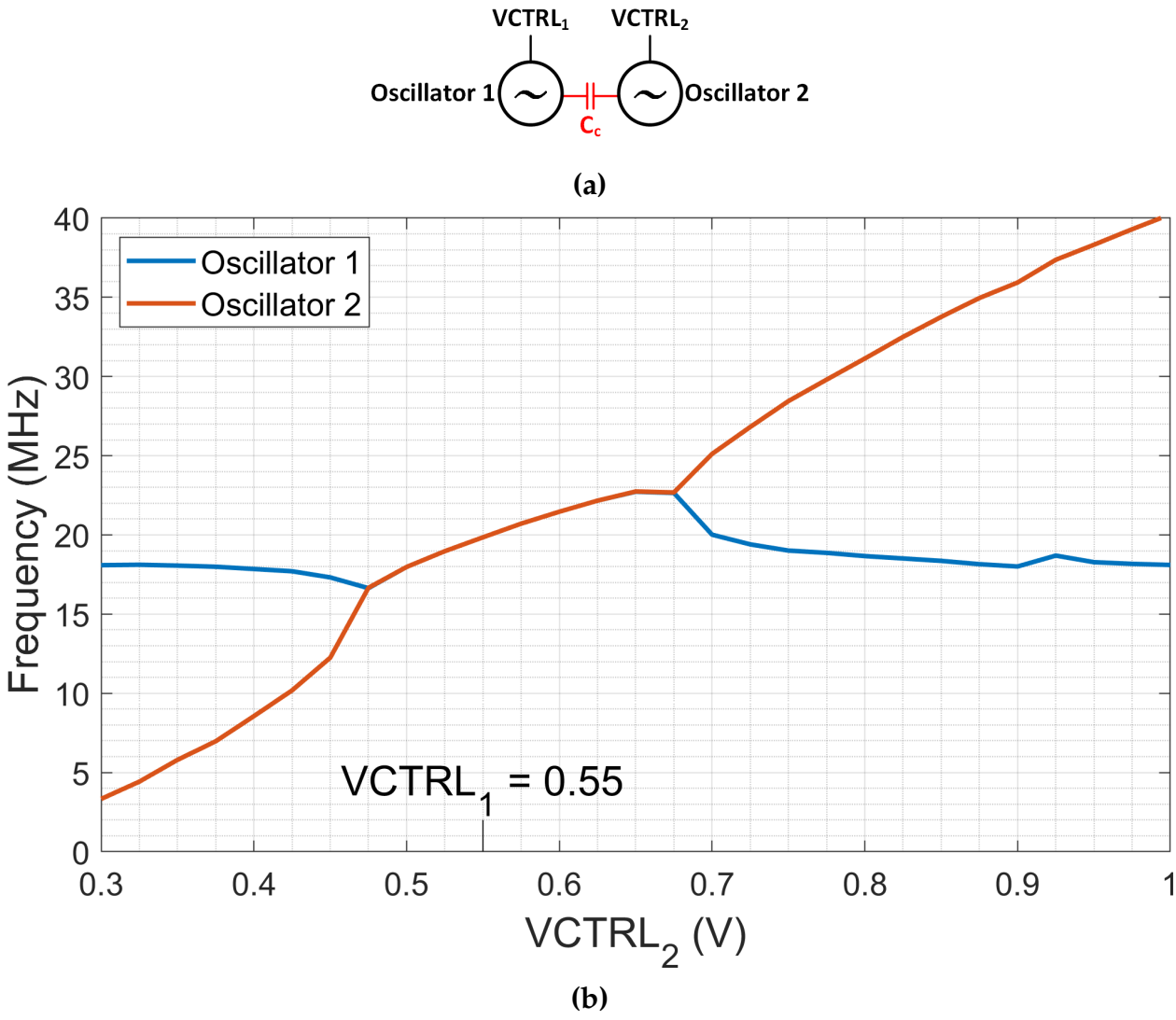



**Figure 5.1:** Synchronization behavior of two coupled CMOS ring oscillators, designed at a commercial 65n process technology and simulated in SPICE

RF probes. The rest of the pads are to be wire bonded onto a package.

On the chip, we have a 10×10 oscillator network and some other test structures. The test structures include single oscillators and pairwise coupled oscillators.

Except one pad that provides the reference current for the digital-to-analog converters, all other signal pads are connected to digital signals.

We used separate VDD domains for isolation purposes and there are 5 of them. In our design, they are all 1 V nominal. The breakdown of the voltage domains are provided in Table 5.1.

| Signal name | Powers: |
|---|---|
| VDDOSC | Oscillators in the network |
| VDDDAC | DACs and current mirrors generating reference currents |
| VDDTEST | Digital blocks such as buffers, scan flops and multiplexers |
| VDDOSCTEST | Oscillators among the test structures |
| VDDE | I/O pads |

**Table 5.1:** A summary of the VDD domains on the chip.

## 5.2 Oscillator Network

In this work, we aim to design the oscillator network such that it can solve the image segmentation (stereo vision) problem we described in the previous chapter. This requires a network where each oscillator is coupled to its 4 nearest neighbors through capacitors. In the image segmentation problem, the input image is encoded into network by mapping the pixel values into the natural oscillation frequency of the oscillators. Therefore, each oscillator needs to be a VCO, and we also need a digital-to-analog converter (DAC) to be able to configure the VCO control voltage.

A 4 nearest neighbor, capacitively coupled oscillator network is a repetition of a unit

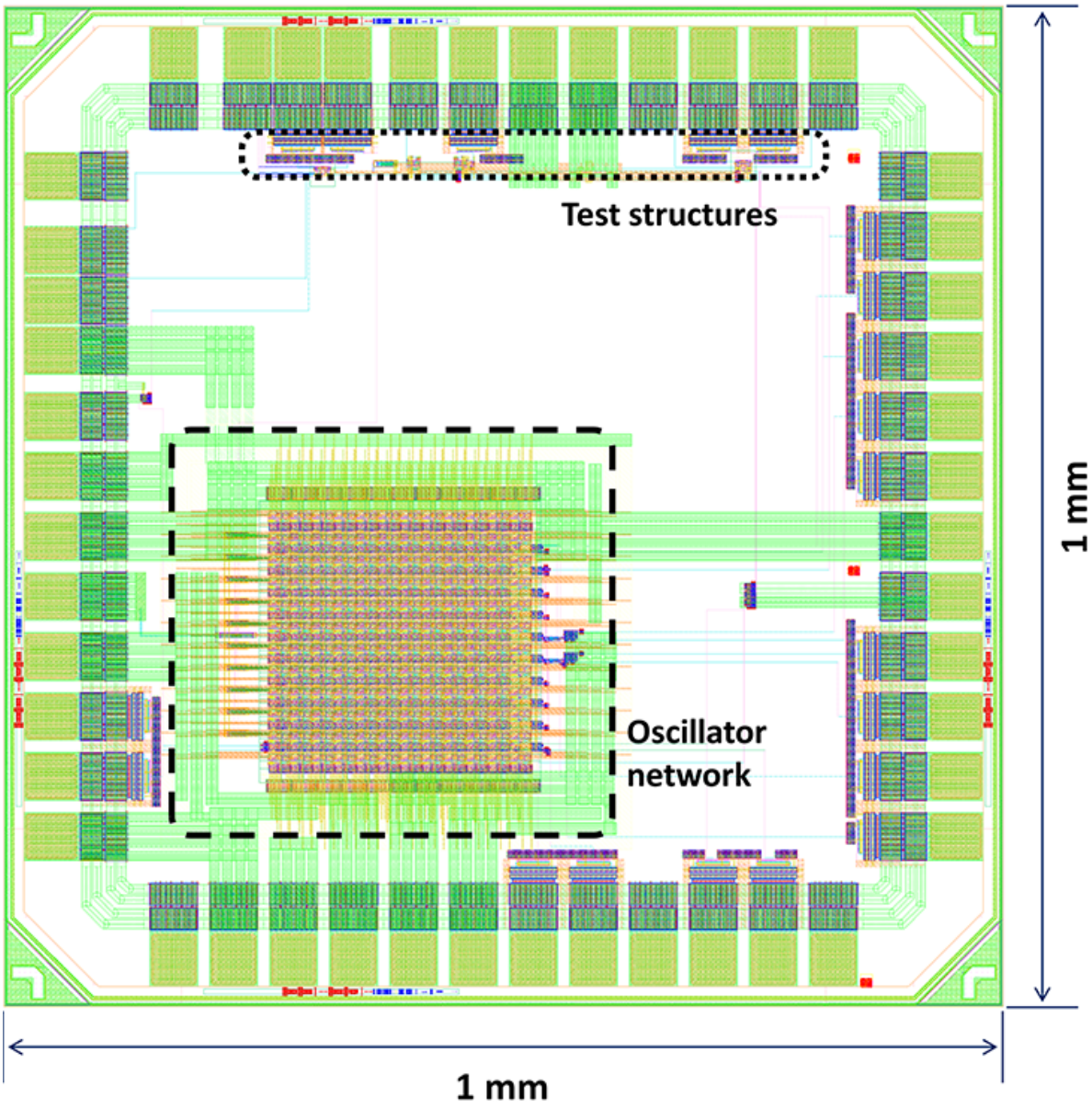



**Figure 5.2:** An overview of the chip layout. The dimension of our block is 1mm $\times$ 1mm.

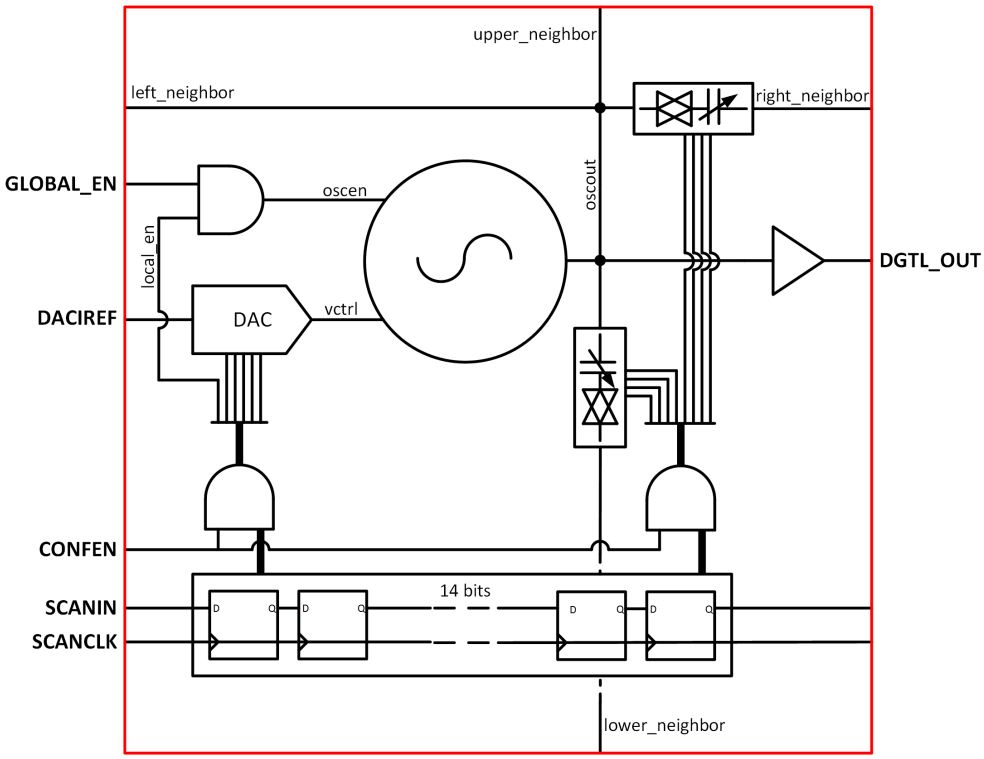



**Figure 5.3:** Block diagram of a tile in the oscillator network. In each tile, there is one ring oscillator VCO, a DAC, transmission gates and capacitors that enable coupling, a buffer and 14 configuration scan flops.

cell that consists a VCO, DAC, configuration blocks and two capacitor banks. One of these capacitors connects the oscillator to its neighbor on the right, and the other one connects the oscillator to its neighbor on the bottom. Once such a unit cell is designed, a matrix of these cells creates the network. We will call this unit cell a *tile* and a functional block diagram is provided in Figure 5.3. In the following subsections, we will go over the design of each block in the tile.

### 5.2.1 Current-starved ring oscillator

Current-starved ring oscillator is our VCO choice for its simplicity. A ring oscillator is an odd number of inverters connected in a ring fashion, hence each inverter output oscillates between VSS and VDD. In the current-starved version, each inverter has two additional transistors. A PMOS is connected between the inverter's PMOS and VDD, and n NMOS is

connected between the inverter's NMOS and VSS. By controlling the gate voltages of these two *current-starving* transistors, the current flowing into the inverter can be adjusted. This affects how quickly the output of the inverter can be charged (or discharged) and therefore affects the oscillation frequency of the ring oscillator. Two additional transistors, an NMOS and a diode-connected PMOS can be used as a biasing circuit to generate the gate voltages of the current-starving transistors from a single control voltage, VCTRL.

A schematic of the ring oscillator used in this work is provided in Figure 5.4. In our design, we replaced the inverter in the first stage of the ring oscillator with a NAND gate to provide an enable input to our oscillator. One input of the NAND gate is connected to OSCEN signal, and the input is connected to the output of the last inverter to complete the loop. OSCEN signal is 0, the output of the NAND gate at at the first stage is always 1. This forces VOUT to be 1. If OSCEN is 1, the NAND gate acts like an inverter and hence the output oscillates. There are two main advantages of this approach. Depending on the configuration, we can turn any oscillator in the network on/off while studying smaller portions of the network, such as a coupled oscillator pair. The second advantage is the ability to start all of the oscillations at the same time with same initial conditions. Even though the network is robust against the initial conditions to the best of our knowledge, this enables us study the evolution of the coupling dynamics.

Because in this study we are aiming a proof-of-concept demonstration, we decided to slow down the oscillators to sub-100 MHz regime for simpler testing. As we have seen from our discussion of the Kuramoto model, the coupled oscillator concept should work at any frequency as long as the coupling strength is scaled appropriately. Even though additional stages is the natural choice to reduce the oscillation frequency, we decided to use 3 stages. With more stages, the oscillator output approaches a square wave, and this means additional high-frequency components. Even though the coupling still works, we decided to stick with the 3 stage design to eliminate high frequency effects while studying

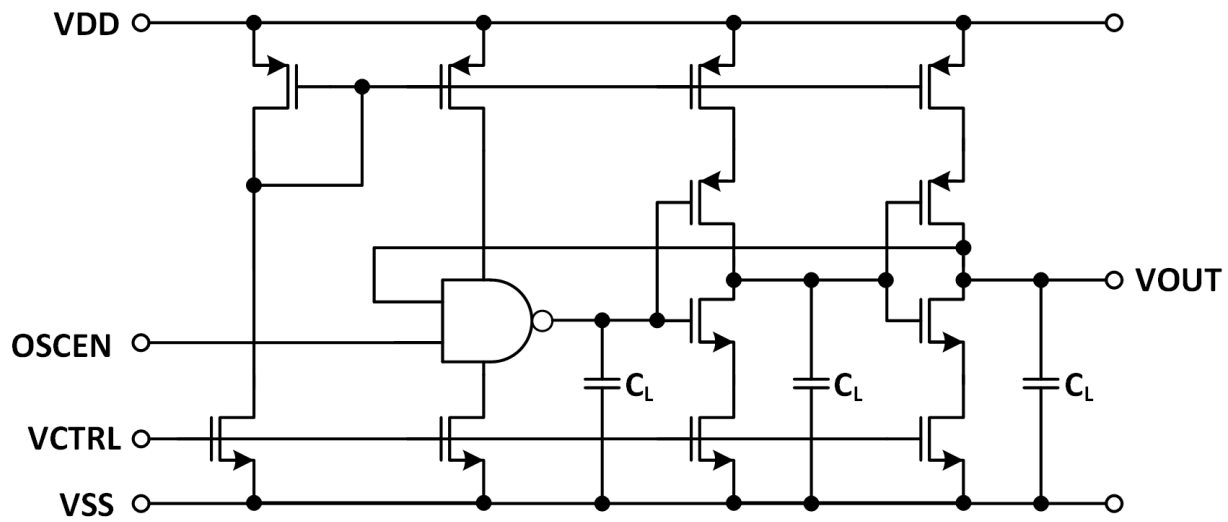



**Figure 5.4:** Schematic of a 3-stage current-starved ring oscillator design with an enable NAND gate. When OSCEN signal is 0, the output of the NAND gate at at the first stage is always 1. This forces VOUT to be 1. If OSCEN is 1, the NAND gate acts like an inverter and hence the output oscillates.

the coupling. In order to slow down the oscillators, we loaded each stage with MOSCAP capacitors. We also designed the inverters using high threshold voltage devices to increase the inverter delay.

The linearity of the oscillator is another concern while designing a VCO. Basically, the oscillation frequency should have a linear dependence on the control voltage which is helpful while mapping the pixel values into oscillation frequency. In the current-starved ring oscillator topology, oscillator linearity depends on the linearity of the drain current in the biasing circuit with respect to the control voltage as the current flowing into each stage are essentially copies of this current. We sized the transistors such that the NMOS in the biasing circuit stays in the SAT region. Because of velocity saturation [51], the saturation current has a linear dependence on the gate voltage (Equation 5.1). Furthermore, the diode connected PMOS should be much stronger than the NMOS to ensure that the NMOS is in deep saturation as the control voltage approaches VDD [52]. For the biasing circuit, we used low threshold voltage transistors to maximize the tuning range of the VCO.

$$I_{Dsat} \cong ZC_o(V_{GS} - V_{th})\nu_{sat} \tag{5.1}$$

With these measures, the oscillation frequency of our design can be tuned between 4 MHz and 44 MHz for control voltages 0.3 V and 1 V, respectively. The power of the oscillator can go up to 52 $\mu W$ depending on the frequency.

### 5.2.2 Digital-to-analog converter

We need a digital-to-analog converter (DAC) to provide the anaalog voltage level for the VCO. We decided to implement a 5-bit *current-steering* DAC (Figure 5.5). In the current-steering DAC, there are curremt sources that connects to a common node ($V_{OUT}$ in Figure 5.5) through CMOS switches. A load resistor ($R_L$) is connected between this node and VDD. Depending on the digital input to the DAC, some of the switches are activated such that total current pulled from the output node is proportional to the decimal DAC input. This current leads to a voltage drop on the resistor and hence the output voltage has a linear dependence on the DAC input.

$$V_{OUT} = V_{DD} - INPUT \times I_o \times R_L \tag{5.2}$$

While designing the current sources, there are two main approaches: binary weighted current sources vs. non-weighted current sources. When binary weighted current sources are used, the binary input to the DAC can be directly used to select these current sources. This is a very simple and area-efficient approach. However, it has disadvantages, too. In the case of process variation, the mismatch between the current sources can lead to large nonlinearity error, even non-monotonous outputs [53]. This becomes more of a concern for more significant bits. For example, for digital input switching from $\langle 0111 \rangle$ (MSB-first)

to $\langle 1000 \rangle$, 3 LSB current sources that corresponds to nominal $7I_o$ are turned off, and the MSB source with nominal $8I_o$ is turned on. If the MSB current source is 12.5% slower than nominal because of process mismatch, during this switching from $\langle 0111 \rangle$ to $\langle 1000 \rangle$, the output current actually reduces. On the other hand, the required mismatch is 100% to cause monotonicity while switching from $\langle 0001 \rangle$ to $\langle 0010 \rangle$.

The other approach is using non-weighted (unary) current sources. In this case, the digital input is converted to a thermometer coded signal where the signal has ones that corresponds to the decimal value of the digital input. For example, the thermometer coded version of $\langle 110 \rangle$ (MSB-first) is $\langle 1111110 \rangle$. With this coding scheme, non-weighted current sources can be used. Unlike the previous case, now when the DAC input is increasing from $\langle 0111 \rangle$ to $\langle 1000 \rangle$, there is no current source that is being turned off. Simply, a new one is being turned on. This, inherently makes the system more robust against process mismatch and guarantees monotonicity. The disadvantage is the requirement for a binary-to-thermometer decoder.

Another approach is using a segmented DAC architecture where the more significant bits are coded by thermometer coding and less significant bits are achieved with binary weighted current sources [54]. This provides a trade-off between the advantages of the aforementioned two approaches. In our design, we went for a 3+2 segmentation where the 2 LSB are achieved with weighted current sources and the 3 MSB are provided by non-weighted current sources. A 3-to-7 thermometer decoder is implemented using basic digital gates. The schematic of our design can be seen in Figure 5.5.

The current sources in our DAC design are NMOS transistors biased in the SAT region. A reference input current is copied into the current sources by current mirrors. This reference input itself is a copy of a reference current provided from one of the I/O pads. The reference current and the load resistor are chosen to provide output voltages between 0.3 and 1.0 V, in accordance with our VCO design.

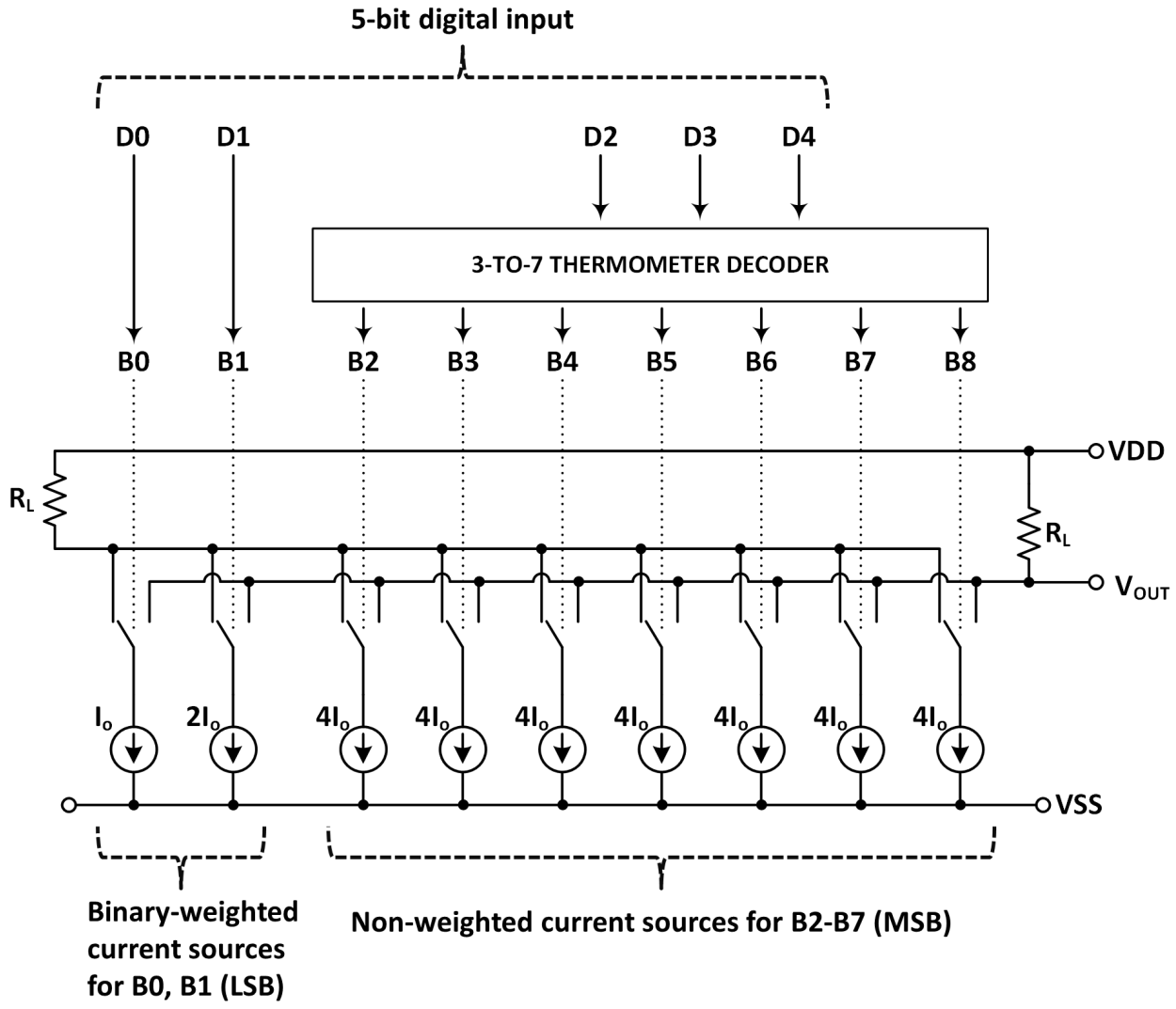



**Figure 5.5:** Segmented DAC schematic.

While designing the current sources and the current mirrors, we utilized higher overdrive voltages (∼0.3 V), common-centroid layout design and dummy transistors to reduce mismatch [55, 56]. During Monte Carlo simulations, we have also found that placing the current mirrors away from the oscillator network core (∼40 µm) helped with mismatch.

### 5.2.3 Coupling capacitors

To couple the oscillators, we decided to use a 4-bit switch capacitor bank. We connected transmission gates in series with the capacitors to enable or disable them. A 4-bit scheme provides enough resolution to study the effect of coupling strength. Further, if all of the transmission gates are turned off, the connection between the two neighbors is broken. This way, we can isolate portions of the network from the rest of it. For example, while studying pairwise coupling, we enable only the one capacitor bank between the pair and disable the rest.

In terms of the capacitor of choice, the options we had were MOSCAP, MIMCAP, and MOMCAP. MOSCAP is the most dense one i.e. has the highest capacitance per area. However, MOSCAPs are nonlinear in nature, their capacitance depends on it voltage [51]. We decided not to use MOSCAP because of this nonlinearity as we wanted a simpler scheme without nonlinear artifacts during this early study of coupled oscillator networks. Yet, our simulations show that MOSCAPs can also be used for coupling and this option can be investigated in future.

Even though MOMCAP can be more dense than MIMCAP, it uses the metalization layers that are needed for routing [57].

MIMCAP is available in the process technology that we used with dedicated metal layers for the bottom and top electrodes of the capacitor. Besides its linearity, another advantage of MIMCAP is the utilization of the area under it. Transistors and the routing metalization can be placed underneath the capacitor to reduce the overall area requirements. We designed our coupling capacitors with this approach. In each tile, there are two capacitor banks and the associated transmission gates to provide coupling to the neighbor on the right and to the one on the bottom. Each capacitor bank can provide capacitance

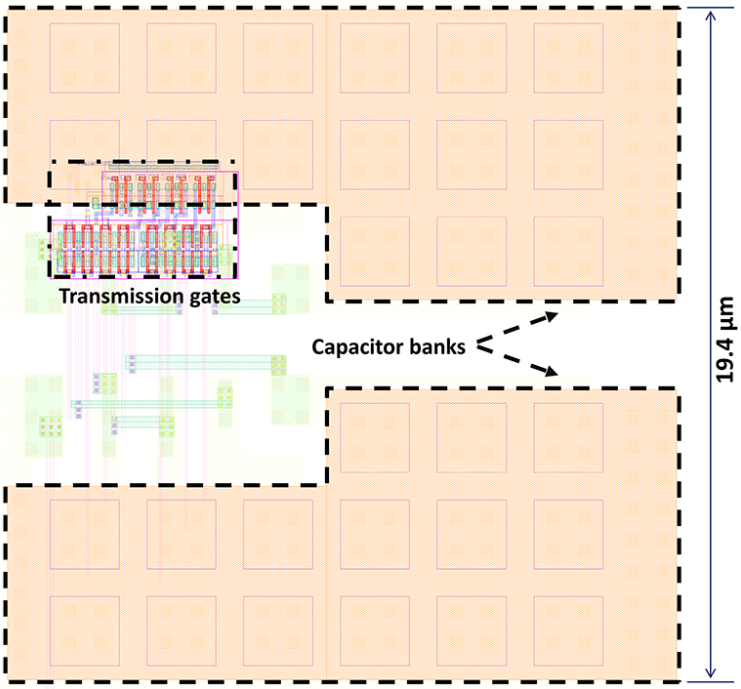



**Figure 5.6:** The layout of the coupling elements.

between 0-150 fF. The layout of these coupling elements is provided in Figure 5.6. The other functional blocks in the tile (Figure 5.7) fits nicely underneath the capacitors.

### 5.2.4 Buffer

The outputs of the oscillators are buffered using digital inverters before they are fed the multiplexers. The inverters are sized to provide enough drive strength to carry the signals across up to 220 µm wiring and the input capacitance of the multiplexers.

### 5.2.5 Scan flops

We have 14 configuration bits in each tile, the breakdown is provided on Table 5.2 below. To provide these scan bits, we constructed a scan chain using flip-flops. The output of each flip-flop ANDed with an external configuration enable signal (CONFEN) that is provided from one of the I/O pads.

Considering the 100 tiles in the 10×10 network, the total length of the network configuration scan chain is 1400 bits. The network is programmed through this scan chain.

The scan flops complete the functional blocks in the tile. The layout of the tile is provided in Figure 5.7. There was some empty space that we filled with some decap capacitors connected between the VDD of the oscillators (VDDOSC) and the VSS.

| Signal name | # of bits | Purpose |
|---|---|---|
| OSCEN | 1 | Enable/disable the oscillator in the tile |
| DACCTRL | 5 | Digital input of the DAC |
| TILE_CPL_RIGHT | 4 | Config. bits for coupling to the right neighbor |
| TILE_CPL_DOWN | 4 | Config. bits for coupling to the bottom neighbor |

**Table 5.2:** Breakdown of the 14 configuration bits in each tile.

### 5.2.6 Read-out Circuitry

In our network, the state variable is the oscillation frequency. We map the pixel values into natural oscillation frequencies and set the DAC configuration bits accordingly. The output we would like to read is also the frequency of the oscillators after the network settles. For this purpose, we could use on-chip frequency detectors. However, we are also interested in the dynamics of the coupling as the network settles and this requires reading the oscillator outputs. For example, in order to investigate the number of oscillation cycles it takes for the network to settle, we might want to check the output signals. Also, as we have seen from the Kuramoto model, the phase difference between pairwise coupled oscillators is an important parameter that needs to be monitored to understand the coupling. Therefore, we decided to MUX all the buffered oscillator outputs and feed to the I/O pads.

We used a 10-to-1 multiplexer at the end of each row in the network and connected all the outputs from that row to this multiplexer. These are our *column-select* multiplexers and there are 10 of them. Then, the outputs of the column-select MUXes are multiplexed again using two 5-to-1 multiplexers, which we will call the *row-select* multiplexers. We connected

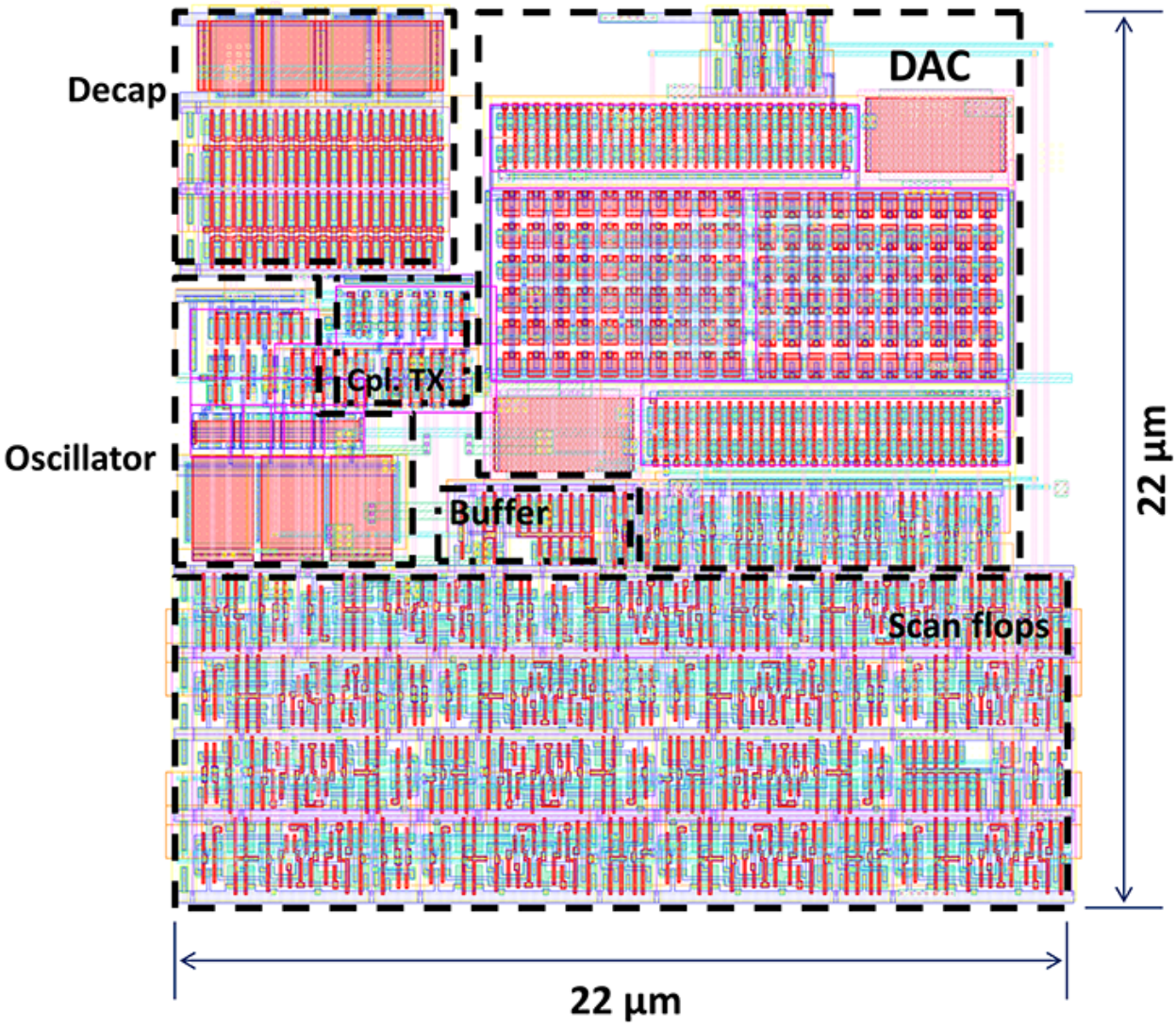



**Figure 5.7:** Tile layout.

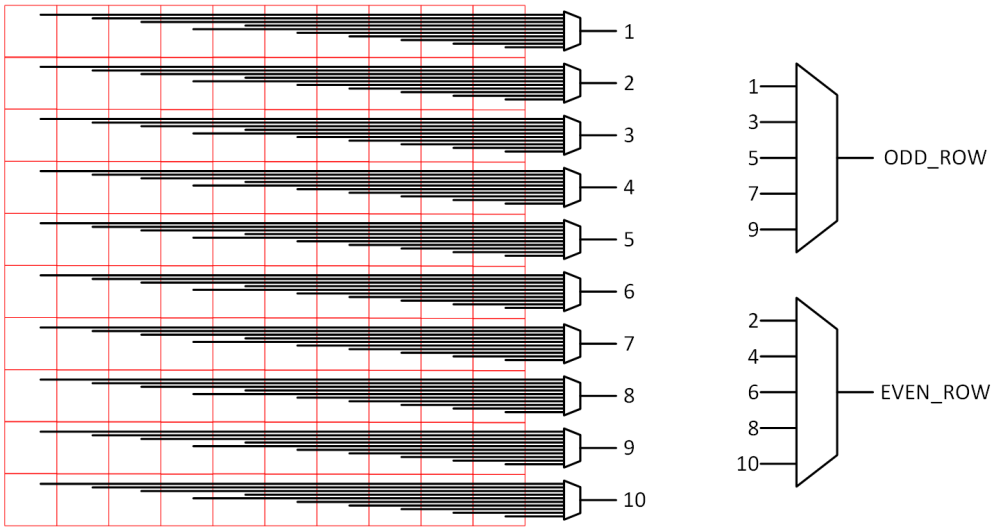



**Figure 5.8:** Read-out scheme of the oscillator network. 10 oscillators along each row are multiplexed (column select). Then 2 outputs from the 5 odd rows and 5 even rows are selected using 5-to-1 MUXes (row select). The two outputs (ODD_ROW and EVEN_ROW) are then buffered and connected to the IO pads.

the odd-numbered rows and even-numbered rows to different row-select multiplexers and their outputs are connected to two I/O pads. The select inputs of the MUXes provided by a dedicated scan chain. Thanks to this dedicated scan chain, we are able to read the output of any oscillator in the network without disturbing the operation of the network. We can also concurrently measure two oscillators if one of them on an odd-numbered row and the other one on an even-numbered row. This is particularly useful while studying pairwise coupled oscillators. The signals are measured using an oscilloscope and the frequency is calculated off-chip.

This completes our network design. An overall look to the network's layout can be seen in Figure 5.9 on the next page.

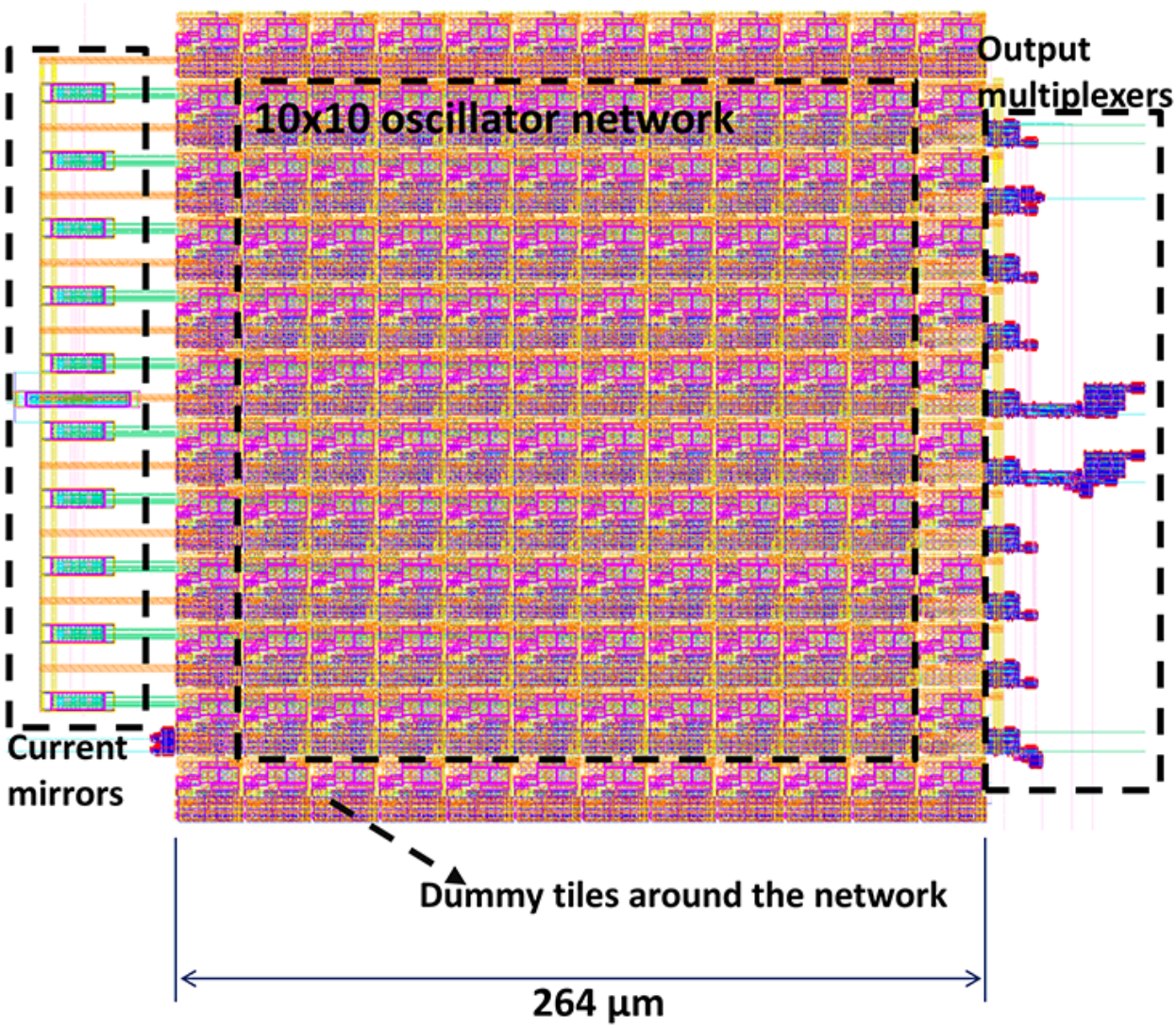



**Figure 5.9:** Network layout.

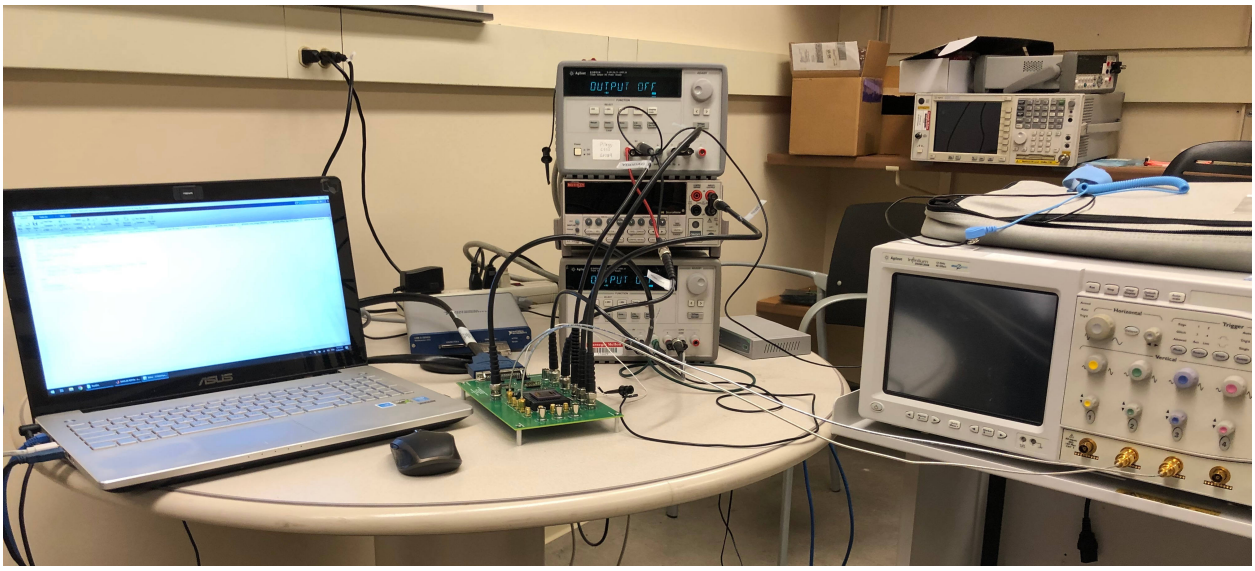



**Figure 5.10:** Test setup.

## 5.3 Chip Testing

Our block was taped out in April 2019. The chips are wirebonded onto a ceramic packaging. The package is placed on a socket and the latter was soldered on a PCB. Power connections are provided through BNC coax cables using Agilent DC Power Supply. The reference current for the DACs are provided through a BNC coax cable using a Keithley 2400 SourceMeter. The two outputs are connected to an Agilent DSO81204B Infiniium High Performance Oscilloscope using SMA cables. The digital input signals are provided using a National Instruments USB-6363 Multifunction I/O Device (NI-DAQ). A photo of the test setup is shown in Figure 5.10. A microphotograph of the wirebonded chip is shown in Figure 5.11.

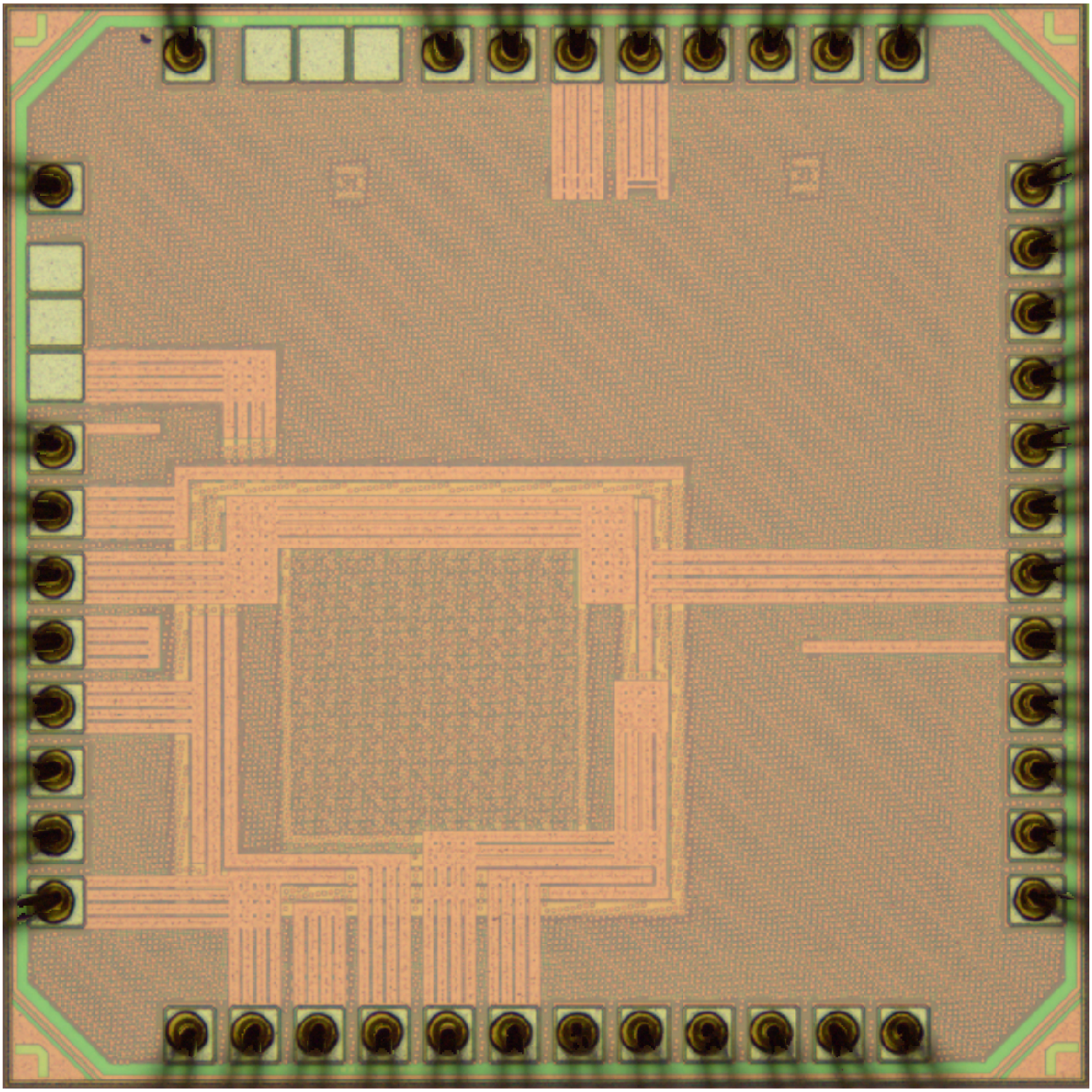


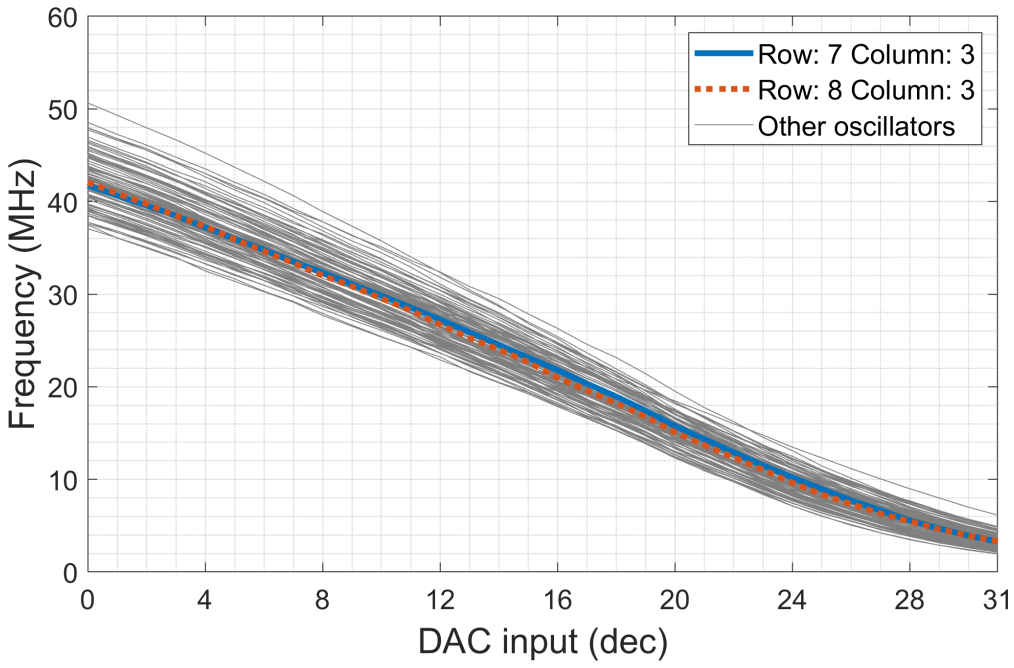
**Figure 5.11:** A microphotograph of the chip. The dimensions are 1 mm×1 mm. The wirebonded pads can be seen. The 6 pads in the upper left corner without wires are for probing purposes. They are connected to some of the test structures.

## 5.4 Single Oscillators

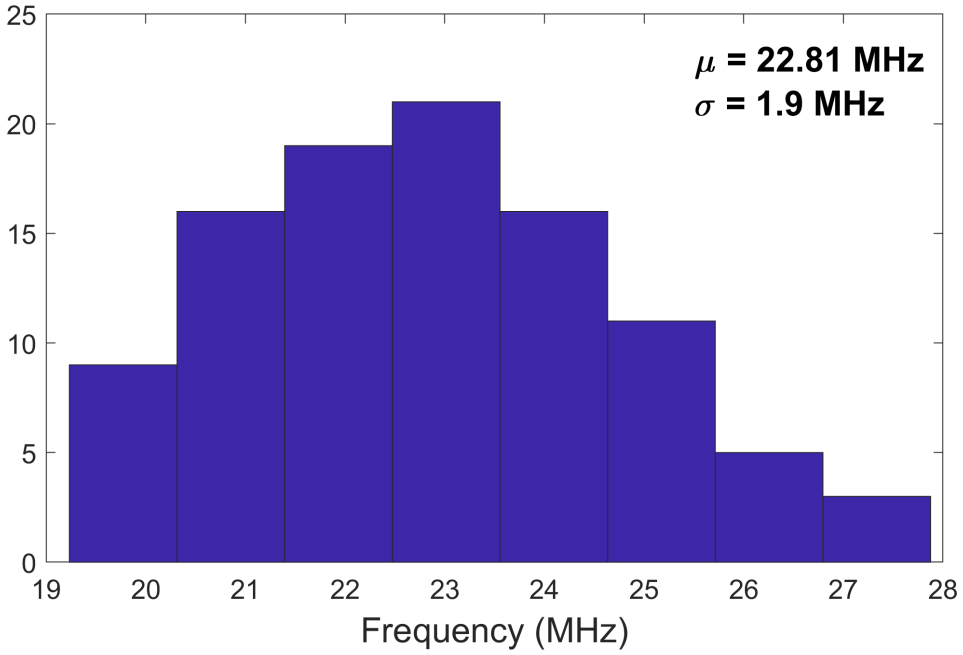
We began testing by characterizing each oscillator in the network to extract their natural oscillation behavior. To do this, we configured the network such that all coupling capacitors are disabled. We also disabled the oscillators except the one we are interested in testing. Then, we measured oscillator output for DAC inputs corresponding to decimal values between 0 and 31. The frequency values that we measured will be treated as the natural (uncoupled) oscillation frequencies during the coupled oscillator experiments. We haven't run into any monotonicity issue, so our DAC design efforts paid off. The linearity of the oscillators are also reasonable.

The DAC input vs frequency plot of all 100 oscillators is provided in Figure 5.12 (a). We particularly highlighted two oscillators to exemplify the behavior: the blue trace corresponds to the oscillator at row 7, column 3; and the orange trace corresponds to row 8, column 3. These two are neighbors that show similar behavior, and we can measure them concurrently. For these reasons, they will be the subjects of the pairwise coupling scheme.

The profiling of the oscillators revealed that the process variations are significant. The gray traces in Figure 5.12 (a) show the other 98 oscillator's behavior and they can deviate quite a bit from the two that we highlighted. To emphasize this variation, we created a histogram plot of the oscillation frequencies of all 100 oscillators for DAC value $\langle 1111 \rangle$ in Figure 5.12 (b). The mean frequency is 22.8 MHz and the standard deviation is 1.9 MHz. This variation between oscillators needs to be addressed while mapping images into network in the image segmentation problem. We will discuss this later.




**(a)**




**(b)**

**Figure 5.12:** (a) The individual characterization of the 100 VCOs on chip reveal that there is a significant variation. Two oscillators (R:7 C:3 and R:8 C:3) are highlighted on purpose. These two oscillators are used for pairwise coupling later on. (b) Statistical distribution of the VCO frequency for DAC input $\langle 1111 \rangle$.

## 5.5 Pairwise Coupled Oscillators

As mentioned earlier, we decided to use the two oscillators at row 7 and row 8 of column 3 for coupled pair experiments. We configured the network to disable all other oscillators except these two. Also, we disabled all the coupling capacitors except the one between these two oscillators. This configuration effectively reduces the network to the diagram shown in the inset in Figure 5.13.

To examine coupling behavior, we set the coupling capacitance to 50 fF and the DAC input of Osc. 1 to $\langle 1111 \rangle$ (decimal 15). Then we measured the frequencies of the two oscillators for Osc. 2 DAC inputs between 0-31 (decimal). The measured frequencies are plotted in Figure 5.13 along with the natural oscillation frequencies measured earlier. Similar to our earlier simulations, coupled ring oscillators exhibit Kuramoto-like coupling behavior. When the difference between the natural oscillation frequencies (detuning) is large, the coupled oscillators oscillate close to their natural frequencies. When the detuning is smaller, however, the two oscillators synchronize. We define *synchronization range* as the difference between the minimum and the maximum natural oscillation frequency of Oscillator 2 for which it can synchronize with Oscillator 1.

We investigated the coupling behavior for different coupling capacitances. Figure 5.14 shows how the synchronization range changes with respect to the coupling capacitance. 0 fF refers to coupling being disabled. We see that there is a linear relationship between the synchronization range and the coupling capacitance.

### 5.5.1 Revisiting the Kuramoto model

The synchronization range, in fact, is the sum of maximum allowed detuning for $f_1 > f_2$ and $f_2 > f_1$.The original Kuramoto model suggest that the maximum detuning allowed is

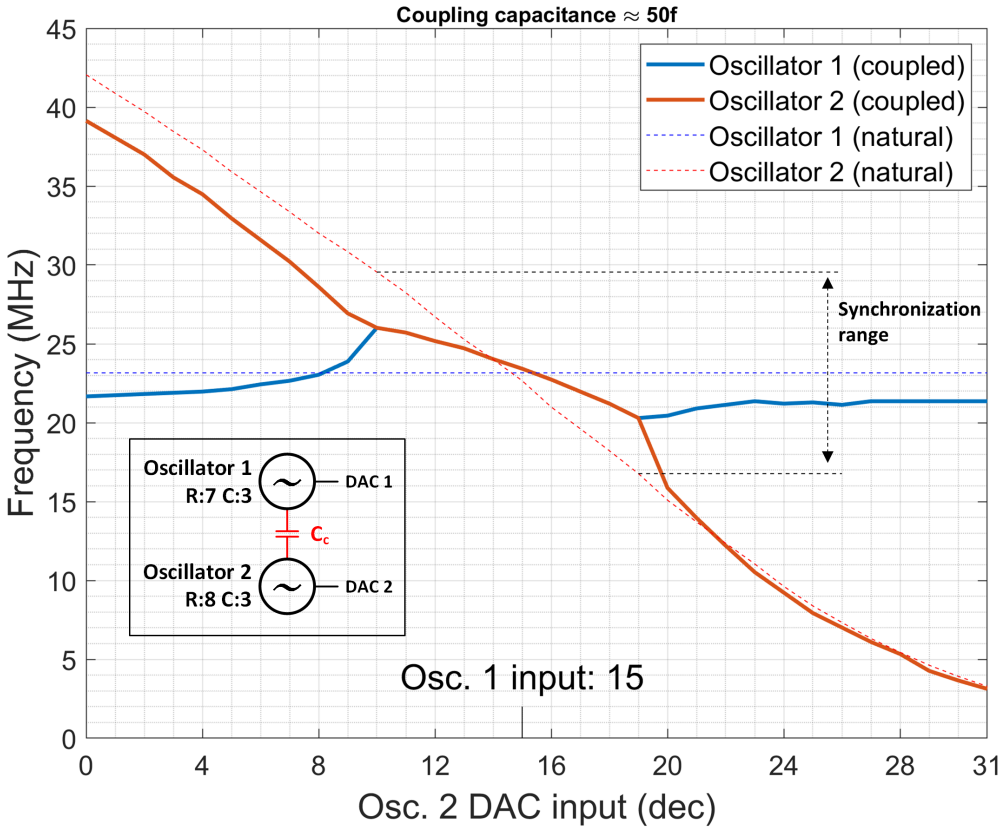



**Figure 5.13:** Chip measurement results showing the synchronization behavior of two coupled CMOS ring oscillators that were highlighted in Figure 5.12. Coupling capacitance was 50 fF.

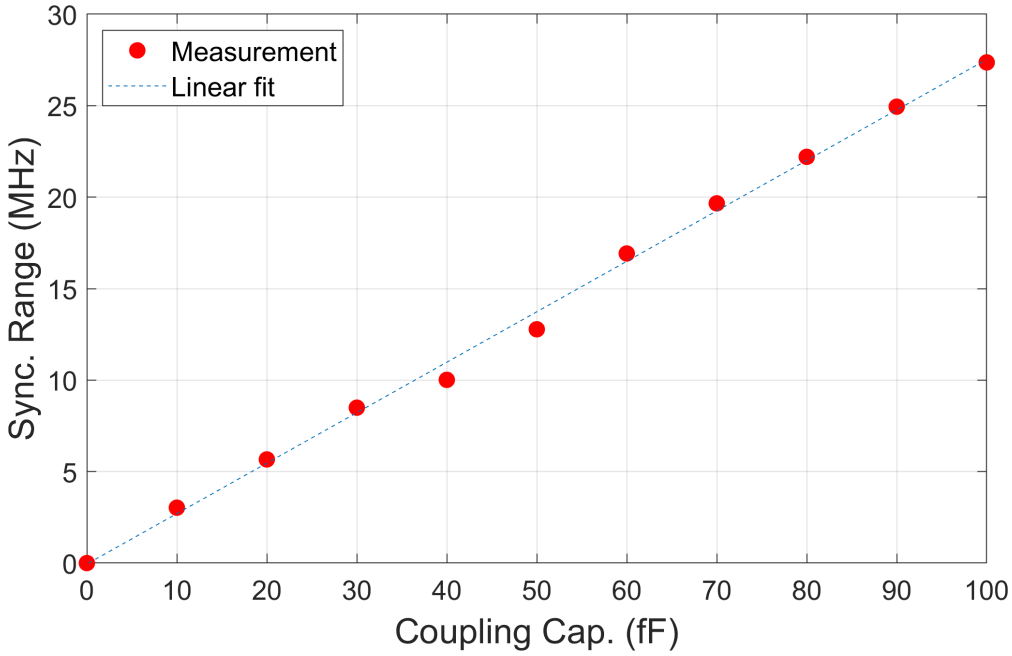



**Figure 5.14:** Synchronization range as a function of the coupling capacitance. The linear dependence suggests that the coupling strength in capacitive coupling is proportional to the coupling capacitance ($C_c$).

the sum of the coupling strengths between the two oscillators. The synchronization range therefore should be a function of the coupling strength. The measurement results that we see in Figure 5.14 suggest that the coupling strength in the capacitive coupling scheme should be proportional to the coupling capacitance ($C_c$).

This result is not surprising. When the two oscillators are capacitively coupled, the current flowing through the capacitor ($I_c$) should be the main drive that forces the two oscillators to synchronize. The fact that there is no synchronization for $C_c = 0$ (or in fact just two disconnected oscillators) supports this hypothesis. $I_c$ should have a linear dependence on $C_c$, as it is inversely proportional to the impedance of the capacitor, $Z_c$.

$$I_c \propto \frac{1}{Z_c} \quad Z_c = \frac{1}{j\omega C_c} \tag{5.3}$$

Therefore, for capacitive coupling case, we propose to update the coupling strength in the

original Kuramoto model as follows:

$$k_{i,j} = K\omega_j C_c \tag{5.4}$$

$k_{i,j}$ is the coupling strength from oscillator $j$ to oscillator $i$. $K$ is a fit parameter with units $1/F$. $C_c$ is the capacitive coupling between the two oscillators. With these modifications, Kuramoto model for capacitive coupling can be expressed as

$$\begin{cases} \dot{\theta}_1 = \omega_1^o + K\omega_2 C_c sin(\theta_2 - \theta_1) \\ \dot{\theta}_2 = \omega_2^o + K\omega_1 C_c sin(\theta_1 - \theta_2) \end{cases} \tag{5.5}$$

Once we substitute $\omega_i = \dot{\theta}_i$ and $\phi = \theta_2 - \theta_1$, this equation reduces to

$$\begin{cases} \dot{\theta}_1 = \dfrac{\omega_1^o + KC_c\omega_2^o sin(\phi)}{1 + K^2 C_c^2 sin^2(\phi)} \\ \dot{\theta}_2 = \dfrac{\omega_2^o - KC_c\omega_1^o sin(\phi)}{1 + K^2 C_c^2 sin^2(\phi)} \end{cases} \tag{5.6}$$

We solved Equation 5.6 using 4$^{\text{th}}$ order Runge-Kutta method on MATLAB R2018a. As we expected, the synchronization range has a linear dependence on the coupling capacitance. The results of this simulation is provided in Figure 5.15.

The frequency dependence of the coupling strength in Equation 5.5 suggests that for larger natural frequencies, the coupling should be stronger. This is difficult to observe in Figure 5.13 because of the narrowness of the synchronization range, but it becomes more significant for larger values of $C_c$. In Figure 5.13, the two oscillators synchronize when the DAC input of oscillator 2 is between 10 and 19. This window is almost symmetrical around 15, the input of oscillator 1. When $C_c$ is 100 fF, in Figure 5.16, the synchronization range extends further for lower DAC inputs i.e. larger natural frequencies for oscillator 2. There is a clear asymmetry favoring the higher frequencies. Figure 5.17 is the prediction

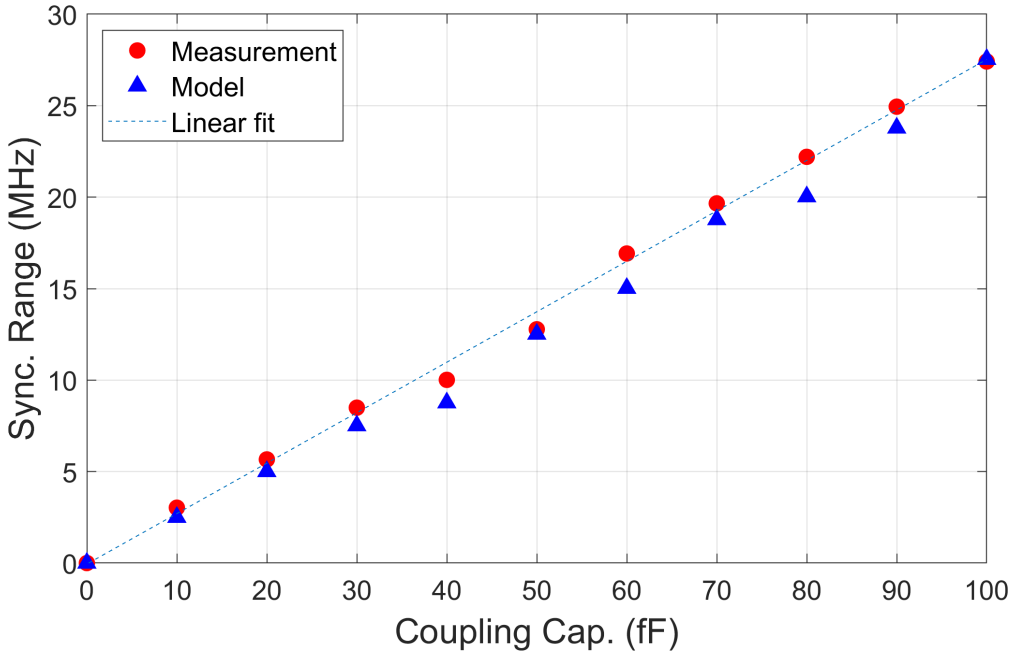



**Figure 5.15:** Solution of the updated Kuramoto model in Equation 5.6 show that the synchronization range increases in proportion with the coupling capacitance. This result is in agreement with the experimental results we obtained earlier.

of Equation 5.6. Similar to the measurement results, an asymmetry is observed.

## 5.6 Network Level Coupling

So far our efforts with the coupled CMOS oscillators were restricted to pairwise coupled oscillators. With this primitive block, we have seen that coupled oscillators can synchronize and their synchronization behavior follows the one described by the Kuramoto model. Further, the coupling strength is proportional to the coupling capacitance.

Next we extend our efforts to a coupled network of oscillators by enabling all the coupling capacitances between the oscillators. We demonstrate an image segmentation application by fully utilizing the network, which is an extension of the stereo vision application discussed in the previous chapter. In image segmentation, the goal is to assign labels to each pixel where the label represents the region that pixel belongs to. in the stereo vi-

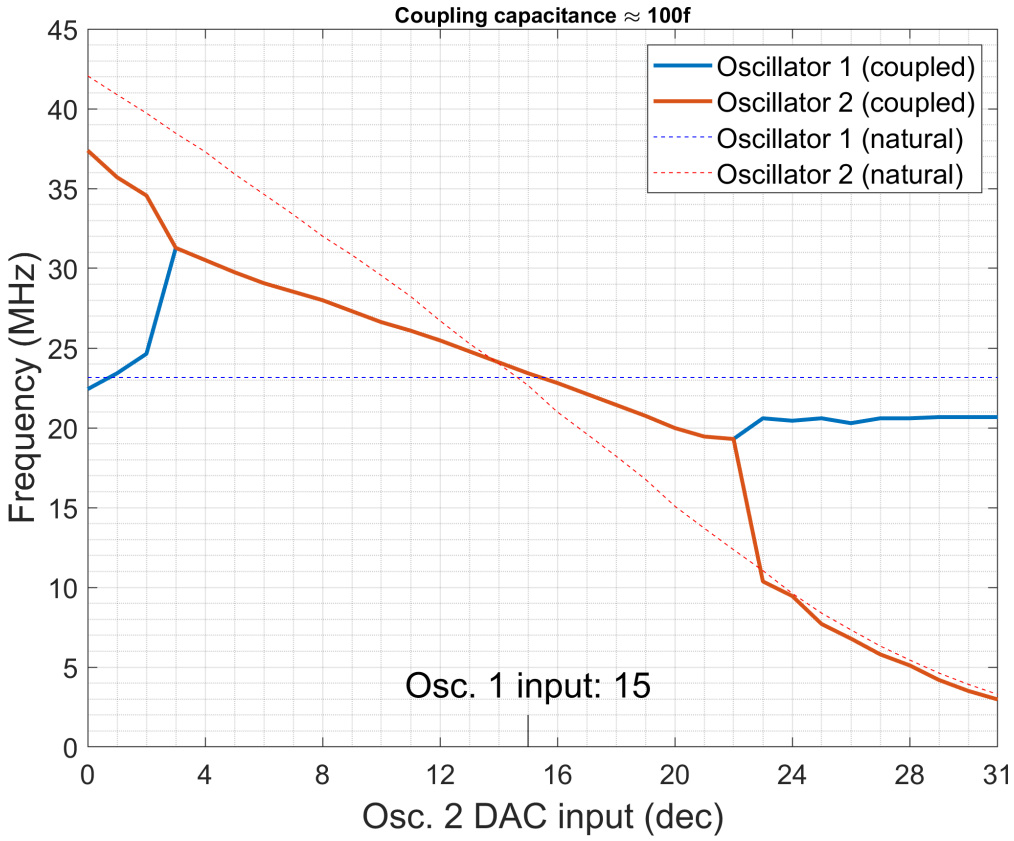



**Figure 5.16:** Synchronization of two coupled CMOS ring oscillators when coupling capacitance is 100 fF. The synchronization range not only widens, but its center also shifts to the left of the input of oscillator 1. In other words, it widens more to the left i.e. towards higher frequencies. This suggests that the coupling strength is a function of frequency as well.

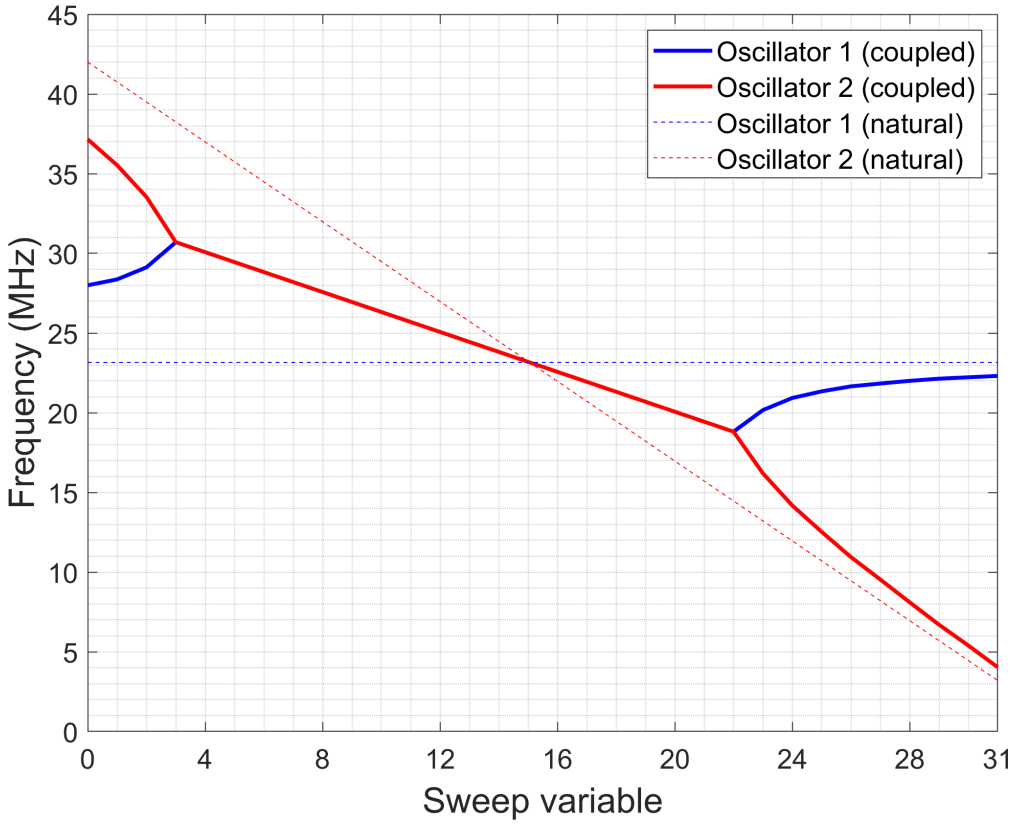



**Figure 5.17:** Solution of the updated Kuramoto model (Equation 5.6) with $C_c = 100$ fF. The synchronization range widens into higher frequency ranges, very similar to what we have seen in our experimental results (Figure 5.16).

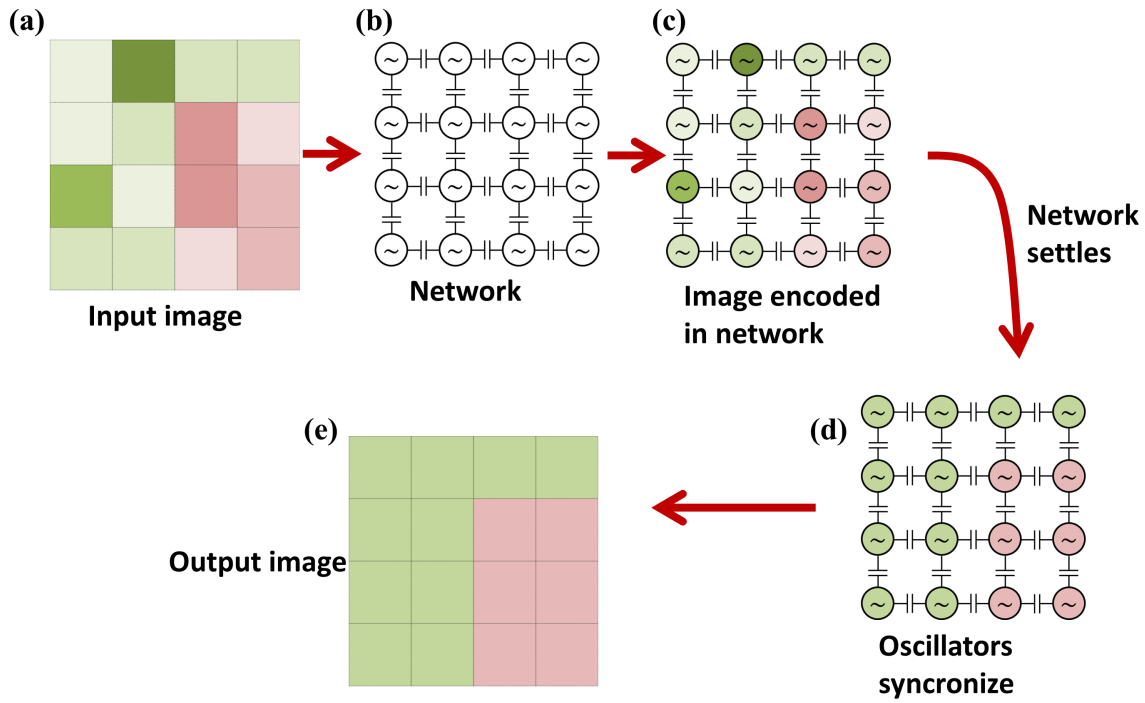



**Figure 5.18:** Image segmentation flow with the coupled oscillator network. The input image (a) and the network (b) has the same size. The input image is encoded into the network by mapping the pixel values of each pixel into natural oscillation frequencies. After the network settles (d) the oscillators in the same region of the image synchronize because of their similar natural frequencies. The output image (e) can be constructed by mapping the frequency information back to pixel values.

sion application, the input image that is segmented is the disparity image and the regions are the objects in the scene. The labels are the oscillation frequencies of the oscillators corresponding to the pixels which represents the distance of the objects to the cameras.

To generalize, in image segmentation application using oscillator networks we start with a noisy input image. We create a network at the same size of the image and map the pixel information into the natural oscillation frequencies of the oscillators. Then we enable the oscillators and let the network settle. As the network settles, the oscillators corresponding to pixels in the same region synchronize because of their similar natural oscillation frequencies. After the network settles, the frequencies of the oscillators are read, converted to pixel values and the output image is constructed. Figure 5.18 summarizes this flow.

Because of extremely limited size of our CMOS oscillator network, it's not realistic to process real images. Even though an image can be processed by breaking it down to smaller tiles, processing and stitching afterwards [29], a 10×10 network is still too small. Therefore, for our tests, we created sample input images using random number generators on Matlab. For our experiments, the pixel values are chosen in the frequency range that our oscillators can span, therefore the mapping from pixel values to frequency is 1:1. To process a real image, such as an 8-bit RGB coded one, the pixel values between 0-255 can be linearly mapped onto the oscillator's frequency range for each color component.

The next step is selecting appropriate DAC inputs. Because of the process variation between the oscillators that we described earlier, mapping the pixel value directly to DAC inputs causes errors. Therefore, the pixel values should first be mapped to natural oscillation frequencies. After that, the DAC value that gives the closest natural oscillation frequency for the oscillator that corresponds to the specific pixel should be chosen. This requires a profiling of the oscillators in the network and saving their natural oscillation frequencies to be used while programming.

An input image to the network is shown in Figure 5.19 (a). The numbers in each pixel of this 10×10 image represents the pixel value and therefore the corresponding natural oscillation frequency in MHz. In the image, we see that there are two main regions with darker and lighter colors. Darker colors are encoded with lower oscillation frequencies and lighter colors are represented by higher frequency oscillations. Within each region, there is a distribution of pixel values with a nonzero mean, similar to a real life image. After the appropriate DAC values are chosen, the network is programmed. All of the coupling capacitances in the network are activated and configured to the same value. This provides a uniform coupling strength across the network. After the oscillations are enabled, the oscillators settled to the frequency values shown in Figure 5.19 (b). We see that the oscillators clustered into two groups corresponding to the regions in the input image.

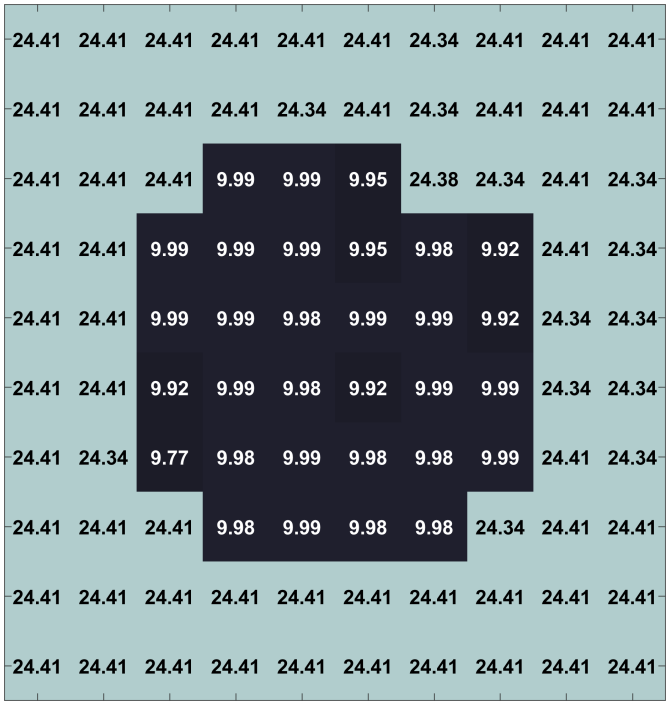
In the output image (Figure 5.19 (b)), we see that there is one mispredicted pixel. This misprediction is likely to be a result of quantization error and network's coupling dynamics. Because of limited DAC resolution, the actual natural frequency will not be exactly same as the input pixel's frequency value. This error should reduce with increasing DAC resolution. The next contributing factor is the relative strength of coupling seen from the neighbors oscillating at higher frequencies. We earlier established that in the case of capacitive coupling the coupling strength is a function of frequency and it favors signals at higher frequencies. Therefore, while an oscillator is being pulled by multiple neighbors, the neighbors with higher frequencies has an upper hand.

An image as simple as Figure 5.19 (a) could have been segmented simply by thresholding. For example, applying a binary thresholding filter with threshold set to 20 MHz would correctly classify all the pixels. However, more complicated or noisy images where pixels with different colors present within a region cannot be segmented by thresholding. Consider the input image shown in Figure 5.20. In this image there are 2 regions, a darker one on the left and a lighter one on the right. There are also darker pixels within the lighter region and the lighter pixels in the darker region. Those pixels cannot be classified correctly by thresholding. In the oscillator network however, neighbors of these pixels can help smooth out the image. Kuramoto's model for multiple neighbors coupled by capacitors can be generalized to Equation 5.7.

$$\dot{\theta}_i = \omega_i^o + \sum_{j=1}^{N} K\omega_j C_c sin(\theta_j - \theta_i) \quad i, j \text{ are neighbors} \tag{5.7}$$

In this case, contributions from multiple neighbors can affect an oscillator's frequency and can help it synchronize, causing the features to be smoothed. Because the coupling capacitance ($C_c$) determines the coupling strength, it also determines the *smoothness cost*


**(a)**




**(b)**

**Figure 5.19:** (a) Input image for image segmentation experiment using CMOS oscillator network. The numbers seen on the pixels corresponds to the natural oscillation frequencies corresponding to those pixels. (b) The output image after the network settles, the numbers show the oscillation frequencies after the network settles. The oscillators clustered into two groups corresponding to the regions in the input image. For this experiment, coupling strength was 70 fF.

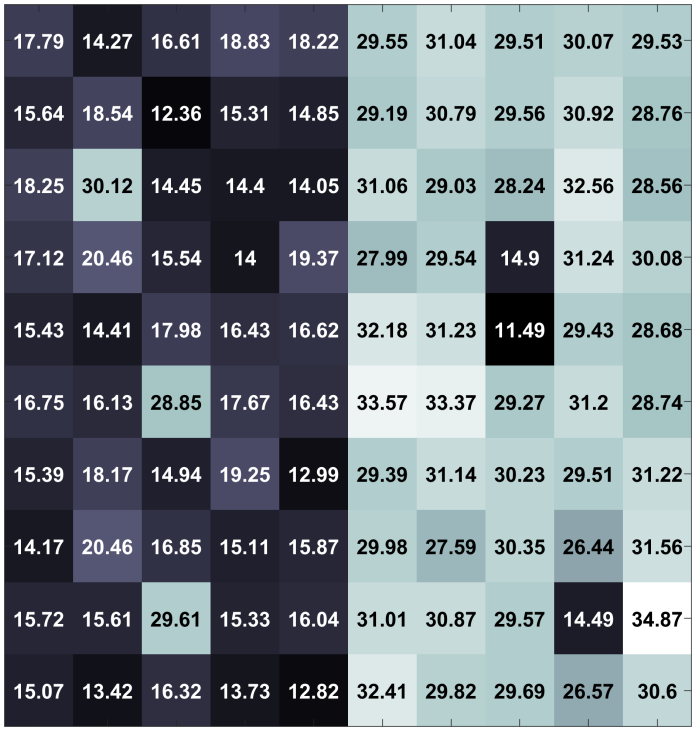



**Figure 5.20:** An input image with 2 regions. Simple thresholding cannot classify the darker pixels within the lighter region and the lighter pixels in the darker region.

[29]. For higher $C_c$, the problematic pixels such as the ones shown in Figure 5.20 can be classified correctly, but this can also lead to losing some finer features as well. This will be the programmer's choice of design in an actual implementation. The outputs of the network with varying degrees of coupling strength ($C_c$) are provided in Appendix 2 for input image shown in Figure 5.20. There are also further examples such as images with 3 regions.

## 5.7 Discussion

The analysis of capacitively coupled oscillator pairs revealed that the coupling behavior is similar to that described by the Kuramoto model. Further, we showed that the coupling strength depends not only on the coupling capacitance but also the operation frequency (Equation 5.8). Kuramoto's original model is scale invariant [33], in other words scaling

the natural frequencies and the coupling strength by the same factor does not affect the dynamics and results in a equivalent behavior [10]. Therefore, there is a striking conclusion that we should draw from Equation 5.8: in a capacitively coupled network, coupling strength automatically scales with the frequencies. Once a capacitively coupled oscillator network is implemented, migrating across process nodes or even different types of oscillator topologies should be seamless, without the need to redesign the couplings.

$$\dot{\theta}_i = \omega_i^o + K\omega_j C_c sin(\theta_j - \theta_i) \tag{5.8}$$

We also showed that CMOS oscillators and S-NDR oscillators exhibit similar coupling properties. Therefore this work also confirms that S-NDR oscillator networks can be used for image processing and verifies our simulation work in the previous chapter. This also poses a question of whether to use CMOS oscillators or S-NDR oscillators for a coupled oscillator network system. Table 5.3 provides a comparison between S-NDR oscillators and CMOS oscillators at 65 nm and 7 nm technology node. The substantial superiority of CMOS oscillators in frequency is a result of the reduced parasitics due to scaling. Independent of the technology, an oscillator's power will be lower bounded by the charging and discharging of the parasitic capacitors. Therefore it's safe to assume that the S-NDR oscillators can provide higher frequency oscillations once scaled and integrated. In fact, the first order frequency calculation model that we developed in Chapter 2 predicts approximately 10 GHz oscillation frequency for parasitic capacitance of $C_p = 10$ fF. Also, in Chapter 3 we discussed how the power consumption of miniaturized S-NDR oscillators can potentially scale down to µW levels. This scenario is also included in the table for reader's convenience. The low power and low frequency operation of S-NDR oscillators resembles the nature of the human brain [58].

Another important advantage of S-NDR oscillators is the area requirement. To con-

| | S-NDR Osc. [49] | Scaled S-NDR | CMOS (65 nm) | CMOS (7 nm) |
|---|---|---|---|---|
| Power | 50 µW | 2.4 µW | 50.8 µW | 12.8 µW |
| Frequency | 600 MHz | 1 GHz | 25.8 GHz | 72.5 GHz |
| Transistors in FEOL | 1 | 1 | 14 | 14 |

**Table 5.3:** Comparison of S-NDR oscillator (this work and [16]) to CMOS oscillators. The numbers for 7 nm are generated using the Predictive Technology Model [59, 60]. The figures for the scaled S-NDR oscillator are based on our predictions discussed in Chapter 3.

struct a current-starved ring oscillator, 14 transistors are needed. On the other hand, an S-NDR oscillator with the 1T1R configuration requires only one transistor. The reduction in the precious FEOL area is important in terms of cost reductions.

# Chapter 6

# Conclusion and Suggestions for Future Research

In order to answer the exponentially growing data processing demand, we need to come up with more efficient ways of computation. Towards this end, CMOS Application Specific Integrated Circuits (ASICs) started to become the norm even in smartphones: Apple's A12 SoC has a dedicated neural engine for machine learning and image processing tasks. Tesla Inc announced their very first self-driving car chip with neural network processors. Even though these are steps in the right direction within today's technology, they will eventually hit the physical barrier any CMOS system will reach. For the future computation systems, we shouldn't be solely focusing on new architectures as in the case with ASICs; but we should also be looking for new device fabrics that are geared towards specific applications because of their fundamental device properties. In this work we demonstrated that S-NDR oscillators and brain-inspired coupled oscillator network system can be viable option to augment CMOS computation, especially for image processing tasks.

Our main contribution in Chapter 2 was application of nonlinear dynamics into novel

S-NDR devices developed. Although the theory behind van der Pol oscillator is old and the threshold switching devices have been around for quite a while, the connection between the two was weak as far as our awareness goes, especially among the recently developing emerging devices community. Chapter 3 built on our understanding of dynamics and explored the device-circuit interrelations. We explored novel methods of ballasting and demonstrated the integration of S-NDR devices with the CMOS technology, which we see as a critical step towards the implementation of systems where the CMOS is augmented by the beyond-CMOS. In Chapter 4 we showed that coupled oscillator networks can solve edge detection and stereo vision problems in an efficient manner. The contribution in Chapter 5 was not only an experimental demonstration of coupled oscillator network based image processing engine, but also a detailed study of capacitive coupling. We extended Kuramoto's model to capture the frequency dependent effects observed in capacitive coupling.

Although our main goal in Chapter 5 was a demonstration of the concept, my experience of designing a CMOS circuit revealed some of the problems that I was not aware of as a device engineer. For example, when we advertised 1T1R configuration as a compact VCO, we often overlooked how the analog voltage levels required for the oscillator network could be provided. The design of the CMOS DAC showed that it's both a power and area hungry circuit. These overheads increase with the DAC resolution, speed and linearity. Although speed is not a requirement for our application and nonlinearity can be mitigated while mapping the image, larger resolution is required especially for complicated images with more features. The need for a large CMOS DAC can nullify the advantages of S-NDR oscillator.

One BEOL-compatible option for replacing the DAC can be using graphene neurons developed in [61]. The integration of graphene with S-NDR oscillators hasn't been demonstrated yet.

A more interesting replacement for DAC can be pulse programmed analog non-volatile memories [62]. Resistive Random Acces Memories (RRAM) are a class of emerging memories which share a number of properties with S-NDR oscillators such as structure, process steps and material families [16]. Their resistance can be programmed to analog levels and hence can be used as a replacement for the DAC. In this case, we would end up with a RRAM ballasted S-NDR oscillator, further improving the area overhead. This scheme is actually used in RRAM implementations where S-NDR device is utilized as a threshold switch.

Even though in this study we focused on image segmentation, oscillator networks can be used to convolve vectors and matrices [63]. This opens more potential application areas especially in the field of computer vision where images are convolved with filters specific to application. The synchronization property of the oscillators can also be used to calculate the Euclidian distance between two vectors which can be interpreted as the covariance or similarity of these two vectors. This in turn can be viewed as the dot product which itself is a memory heavy compute and even graphene based nanofunctions are proposed [64]. Especially the integration of S-NDR oscillators with RRAM memory can pave the way to in/near memory computation of dot product and can provide a hardware accelerator.

In conclusion, we have investigated investigated directly coupled oscillator networks applied to image processing applications. We have shown a proof-of-concept for image processing but the circuits and devices presented were not optimized for performance. We also explored the circuit-device relations in S-NDR oscillators and developed a circuit model accordingly. This model enabled us investigate new ballasting mechanisms and oscillator networks before fabricating them. The initial results shown in this thesis are promising but much more work is required before coupled networks of S-NDR oscillators become a viable technology.

# Appendix A

# CMOS Integration Fab Flow

Figure A.1 summarizes the fabrication process flow for the heterogeneous integration of S-NDR devices with the scaled CMOS technology. This process was developed in collaboration with Dr. Greg Slovin.

1. The CMOS chips were taped out at a commercial 28 nm process node. The chip was taken out of the process line after metal 7 interconnect layer. The source, drain and gate terminals of the transistors are connected to interconnects on M7. The fabrication steps are outlined below:

2. Reactive-ion-etching of SiCN passivation layer involves using an $SF_6$ based recipe.

3. The bottom electrode is deposited using sputtering and patterned using photolithography.

4. The materials for the functional layer (Ge and Te, for example) are deposited by co-sputtering. The correct stoichiometry is achieved by adjusting for the power levels of the targets. The deposition rates of each target are measured by rate runs that was done earlier. The patterning was done using photolithography.

5. The top electrode is deposited using sputtering and patterned using photolithography.

6. The pads are deposited and patterned.

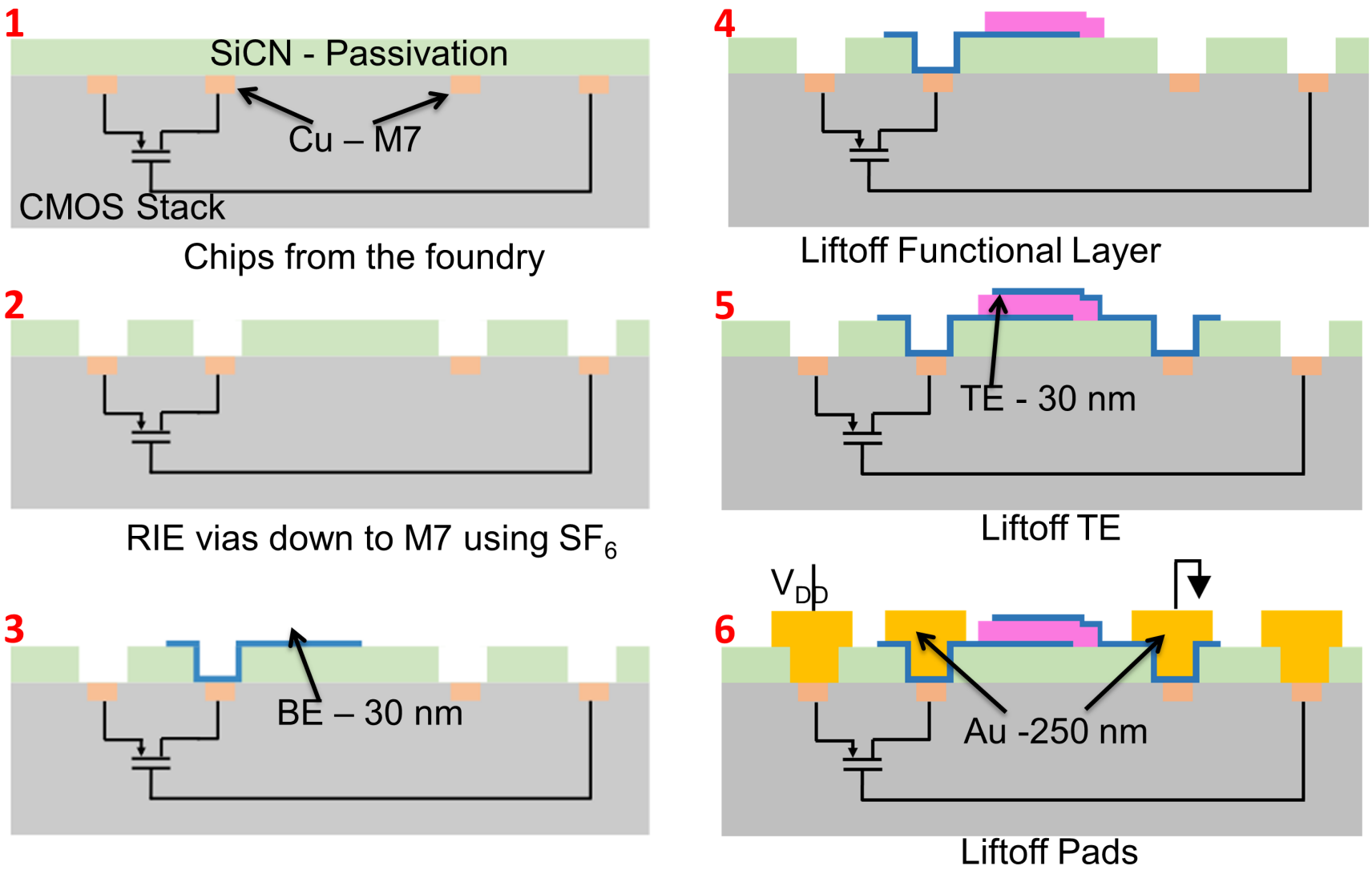



**Figure A.1:** Fabrication process flow for the heterogeneous integration of S-NDR devices with the 28nm CMOS technology

# Appendix B

# Coupled Oscillator Network Image Segmentation Examples

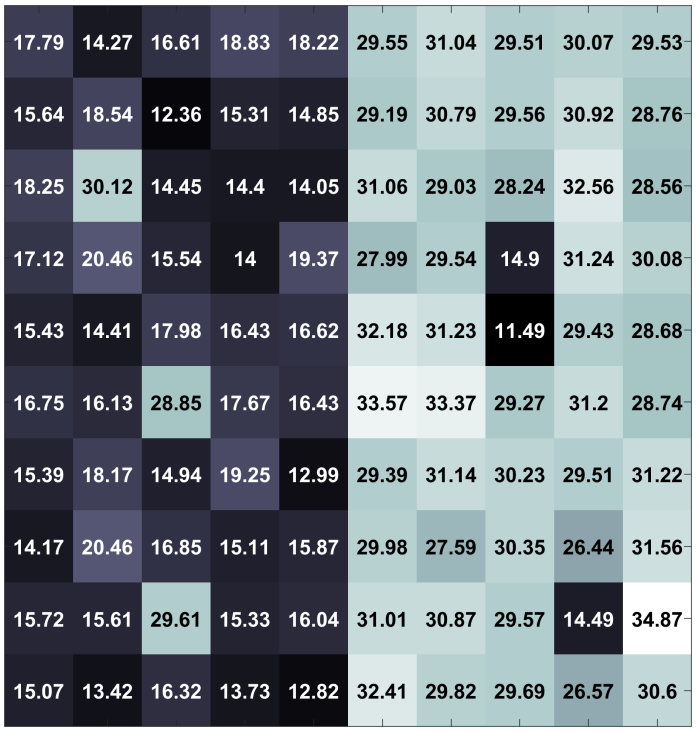
Figure B.1 (a) shows the noisy input image that we discussed earlier in Chapter 5. In an image segmentation application, we ideally we would like to get rid of the noise pixels. A simple thresholding function cannot eliminate those, this is why image segmentation becomes a difficult problem. An image segmentation method should be able to smooth out the noise based on the pixel values of the neighbors. In the coupled oscillator network, the coupling from the neighbors provide this smoothing. The value of the coupling capacitance will play a key role in this, as the $C_c$ is larger, the synchronization window is larger. Therefore, larger $C_c$ values lead to more smooth images.

Figure B.1 (b) shows the output image when $C_c = 50f$. The noisy pixels on the right have been eliminated, but the ones on the left are still there. Because the average frequency is higher in the bright region (right), the effective coupling is stronger. Therefore it is easier to smooth out such features at a relatively low value of $C_c$.

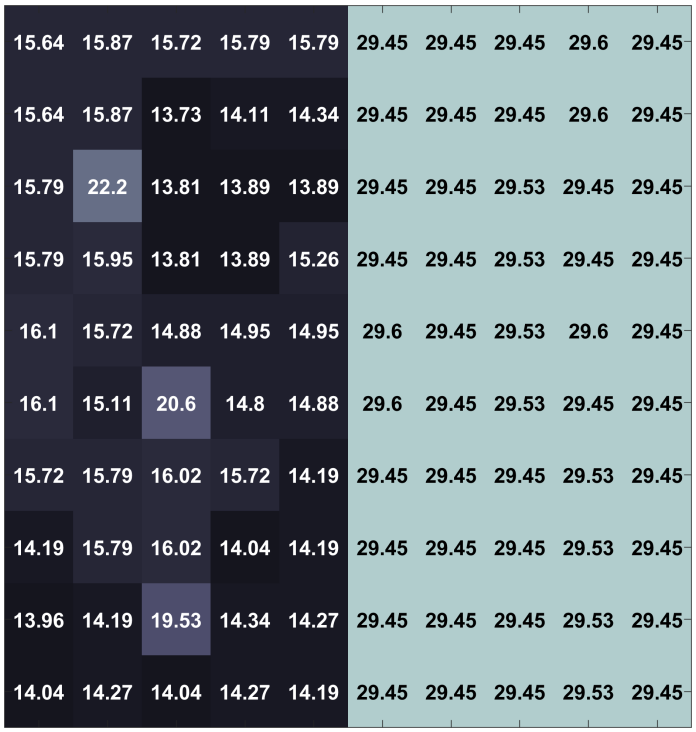
Figure B.2 (a) and (b) shows the output image when $C_c = 60f$ and $C_c = 70f$, respectively.

We see that increasing $C_c$ helps smoothing the pixels on the darker region as well.

Figure B.3 (a) and (b) shows the input and the output images with 3 regions.

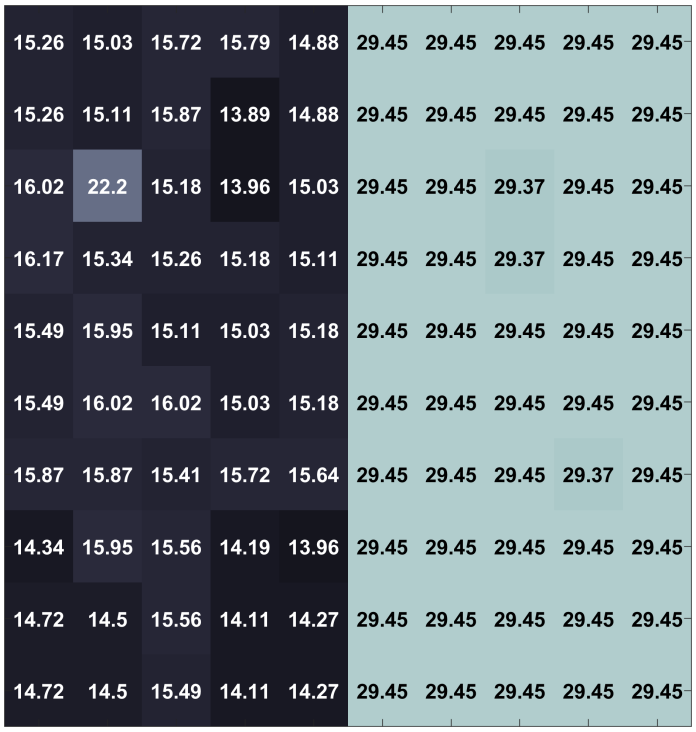



**(a)**

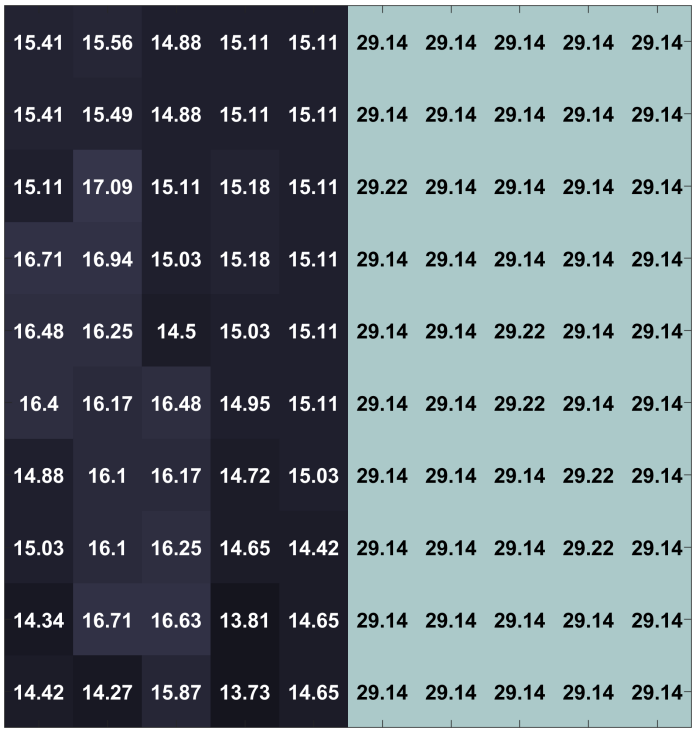



**(b)**

**Figure B.1:** (a) Input image (b) Output image ($C_c$ = 50f) with noisy pixels.




**(a)**




**(b)**

**Figure B.2:** (a) Output image ($C_c$ = 60f) (b) Output image ($C_c$ = 70f)


**(a)**


**(b)**

**Figure B.3:** (a) Input image with 3 regions (b) Output image segmented correctly

# Bibliography


[1] K. Mistry, C. Allen, C. Auth, B. Beattie, D. Bergstrom, M. Bost, M. Brazier, M. Buehler, A. Cappellani, R. Chau, *et al.*, "A 45nm logic technology with high-k+ metal gate transistors, strained silicon, 9 cu interconnect layers, 193nm dry patterning, and 100% pb-free packaging," in *2007 IEEE International Electron Devices Meeting*, pp. 247–250, IEEE, 2007.

[2] N. Loubet, T. Hook, P. Montanini, C.-W. Yeung, S. Kanakasabapathy, M. Guillom, T. Yamashita, J. Zhang, X. Miao, J. Wang, *et al.*, "Stacked nanosheet gate-all-around transistor to enable scaling beyond finfet," in *2017 Symposium on VLSI Technology*, pp. T230–T231, IEEE, 2017.

[3] T. I. R. for Devices and Systems, "International technology roadmap for semiconductors 2.0 2015 edition beyond cmos," tech. rep., 2015.

[4] M. M. S. Aly, M. Gao, G. Hills, C.-S. Lee, G. Pitner, M. M. Shulaker, T. F. Wu, M. Asheghi, J. Bokor, F. Franchetti, *et al.*, "Energy-efficient abundant-data computing: The n3xt 1,000 x," *Computer*, vol. 48, no. 12, pp. 24–33, 2015.

[5] S.-C. Lin, Y. Zhang, C.-H. Hsu, M. Skach, M. E. Haque, L. Tang, and J. Mars, "The architectural implications of autonomous driving: Constraints and acceleration," in *ACM SIGPLAN Notices*, vol. 53, pp. 751–766, ACM, 2018.

[6] N. Jouppi, C. Young, N. Patil, and D. Patterson, "Motivation for and evaluation of the first tensor processing unit," *IEEE Micro*, vol. 38, no. 3, pp. 10–19, 2018.

[7] D. Karunakaran, "Semantic segmentation - udaity's self-driving car engineer nanodegree." https://medium.com/intro-to-artificial-intelligence/semantic-segmentation-udaitys-self-driving-car-engineer-nanodegree-c01eb6eaf9d. [Online; accessed 31-August-2019].

[8] F. C. Hoppensteadt and E. M. Izhikevich, "Pattern recognition via synchronization in phase-locked loop neural networks," *IEEE Transactions on Neural Networks*, vol. 11, no. 3, pp. 734–738, 2000.

[9] D. Wang and D. Terman, "Locally excitatory globally inhibitory oscillator networks," *IEEE transactions on neural networks*, vol. 6, no. 1, pp. 283–286, 1995.

[10] D. Vodenicarevic, *Rhythms and oscillations: a vision for nanoelectronics*. PhD thesis, Paris-Sud University, 2017.

[11] M. Romera, P. Talatchian, S. Tsunegi, F. A. Araujo, V. Cros, P. Bortolotti, J. Trastoy, K. Yakushiji, A. Fukushima, H. Kubota, *et al.*, "Vowel recognition with four coupled spin-torque nano-oscillators," *Nature*, vol. 563, no. 7730, p. 230, 2018.

[12] M. B. Sachs, H. F. Voigt, and E. D. Young, "Auditory nerve representation of vowels in background noise," *Journal of neurophysiology*, vol. 50, no. 1, pp. 27–45, 1983.

[13] T. C. Jackson, A. A. Sharma, J. A. Bain, J. A. Weldon, and L. Pileggi, "Oscillatory neural networks based on tmo nano-oscillators and multi-level rram cells," *IEEE journal on Emerging and Selected Topics in Circuits and Systems*, vol. 5, no. 2, pp. 230–241, 2015.

[14] N. Shukla, A. Parihar, E. Freeman, H. Paik, G. Stone, V. Narayanan, H. Wen, Z. Cai, V. Gopalan, R. Engel-Herbert, *et al.*, "Synchronized charge oscillations in correlated electron systems," *Scientific reports*, vol. 4, p. 4964, 2014.

[15] A. A. Sharma, J. A. Bain, and J. A. Weldon, "Phase coupling and control of oxide-based oscillators for neuromorphic computing," *IEEE Journal on Exploratory Solid-State Computational Devices and Circuits*, vol. 1, pp. 58–66, 2015.

[16] A. A. Sharma, *S-type Negative Differential Resistance: Emerging Memory and Oscillators for Next-Generation Computation*. PhD thesis, Carnegie Mellon University, 2015.

[17] S. H. Strogatz, *Nonlinear dynamics and chaos: with applications to physics, biology, chemistry, and engineering*. Perseus Book, 1994.

[18] B. Van der Pol, "Lxxxviii. on âĂIJrelaxation-oscillationsâĂİ," *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, vol. 2, no. 11, pp. 978–992, 1926.

[19] M. Shaw, H. Grubin, and I. Gastman, "Analysis of an inhomogeneous bulk" s-shaped" negative differential conductivity element in a circuit containing reactive elements," *IEEE Transactions on Electron Devices*, vol. 20, no. 2, pp. 169–178, 1973.

[20] H. L. Grubin, V. V. Mitin, E. Schöll, and M. P. Shaw, *The physics of instabilities in solid state electron devices*. Springer Science & Business Media, 2013.

[21] Z. Sun, S. Yin, X. Wang, J. Cao, Y. Wang, and Y. Wang, "Self-sustained current oscillations in superlattices and the van der pol equation," *Applied Physics Letters*, vol. 87, no. 18, p. 182110, 2005.

[22] H. Abraham and E. Bloch, "Mesure en valeur absolue des périodes des oscillations électriques de haute fréquence," 1919.

[23] J.-M. Ginoux, "Van der pol's lectures: Towards the concept of relaxation oscillations," in *History of Nonlinear Oscillations Theory in France (1880-1940)*, pp. 109–130, Springer, 2017.

[24] A. Sharma, Y. Kesim, M. Shulaker, C. Kuo, C. Augustine, H.-P. Wong, S. Mitra, M. Skowronski, J. Bain, and J. Weldon, "Low-power, high-performance s-ndr oscillators for stereo (3d) vision using directly-coupled oscillator networks," in *2016 IEEE Symposium on VLSI Technology*, pp. 1–2, IEEE, 2016.

[25] M. D. Pickett, J. Borghetti, J. J. Yang, G. Medeiros-Ribeiro, and R. S. Williams, "Coexistence of memristance and negative differential resistance in a nanoscale metal-oxide-metal system," *Advanced Materials*, vol. 23, no. 15, pp. 1730–1733, 2011.

[26] M. D. Pickett and R. S. Williams, "Sub-100 fj and sub-nanosecond thermally driven threshold switching in niobium oxide crosspoint nanodevices," *Nanotechnology*, vol. 23, no. 21, p. 215202, 2012.

[27] S. R. Ovshinsky, "Reversible electrical switching phenomena in disordered structures," *Physical Review Letters*, vol. 21, no. 20, p. 1450, 1968.

[28] A. Sharma, T. Jackson, M. Schulaker, C. Kuo, C. Augustine, J. Bain, H.-S. Wong, S. Mitra, L. Pileggi, and J. Weldon, "High performance, integrated 1t1r oxide-based oscillator: Stack engineering for low-power operation in neural network applications," in *2015 Symposium on VLSI Technology (VLSI Technology)*, pp. T186–T187, IEEE, 2015.

[29] A. Sharma, Y. Kesim, M. Shulaker, C. Kuo, C. Augustine, H.-P. Wong, S. Mitra, M. Skowronski, J. Bain, and J. Weldon, "Low-power, high-performance s-ndr oscillators for stereo (3d) vision using directly-coupled oscillator networks," in *2016 IEEE Symposium on VLSI Technology*, pp. 1–2, IEEE, 2016.

[30] B. Razavi, "The cross-coupled pair-part i [a circuit for all seasons]," *IEEE Solid-State Circuits Magazine*, vol. 6, no. 3, pp. 7–10, 2014.

[31] N. Shareef, D. L. Wang, and R. Yagel, "Segmentation of medical images using legion," *IEEE transactions on medical imaging*, vol. 18, no. 1, pp. 74–91, 1999.

[32] Y. Kuramoto, *Chemical oscillations, waves, and turbulence*. Courier Corporation, 2003.

[33] A. Pikovsky, J. Kurths, M. Rosenblum, and J. Kurths, *Synchronization: a universal concept in nonlinear sciences*, vol. 12. Cambridge university press, 2003.

[34] A. Mauroy, P. Sacré, and R. Sepulchre, "Kick synchronization versus diffusive synchronization," in *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, pp. 7171–7183, IEEE, 2012.

[35] N. Kopell and D. Somers, "Anti-phase solutions in relaxation oscillators coupled through excitatory interactions," *Journal of mathematical biology*, vol. 33, no. 3, pp. 261–280, 1995.

[36] A. Neogy and J. Roychowdhury, "Analysis and design of sub-harmonically injection locked oscillators," in *2012 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pp. 1209–1214, IEEE, 2012.

[37] S. Dal Toso, A. Bevilacqua, M. Tiebout, S. Marsili, C. Sandner, A. Gerosa, and A. Neviani, "Uwb fast-hopping frequency generation based on sub-harmonic injection locking," *IEEE Journal of Solid-State Circuits*, vol. 43, no. 12, pp. 2844–2852, 2008.

[38] M. Acar, D. Leenaerts, and B. Nauta, "A wide-band cmos injection-locked frequency divider," in *2004 IEE Radio Frequency Integrated Circuits (RFIC) Systems. Digest of Papers*, pp. 211–214, IEEE, 2004.

[39] R. Wigington, "A new concept in computing," *Proceedings of the IRE*, vol. 47, no. 4, pp. 516–523, 1959.

[40] E. Goto, "The parametron, a digital computing element which utilizes parametric oscillation," *Proceedings of the IRE*, vol. 47, no. 8, pp. 1304–1316, 1959.

[41] J. Roychowdhury, "Boolean computation using self-sustaining nonlinear oscillators," *Proceedings of the IEEE*, vol. 103, no. 11, pp. 1958–1969, 2015.

[42] Y. E. Kesim, D. Gala, J. A. Bain, and J. A. Weldon, "Phase based boolean computation using gete6 oscillators," in *2017 IEEE 17th International Conference on Nanotechnology (IEEE-NANO)*, pp. 850–854, IEEE, 2017.

[43] T. Yang, R. A. Kiehl, and L. O. Chua, "Tunneling phase logic cellular nonlinear networks," *International Journal of Bifurcation and chaos*, vol. 11, no. 12, pp. 2895–2911, 2001.

[44] X. Lai and J. Roychowdhury, "Fast simulation of large networks of nanotechnological and biochemical oscillators for investigating self-organization phenomena," in *Proceedings of the 2006 Asia and South Pacific Design Automation Conference*, pp. 273–278, IEEE Press, 2006.

[45] H. S. Neoh and A. Hazanchuk, "Adaptive edge detection for real-time video processing using fpgas," *Global Signal Processing*, vol. 7, no. 3, pp. 2–3, 2004.

[46] S. Mattoccia, P. Macrí, G. Parmigiani, and G. Rizza, "A compact, lightweight and energy efficient system for autonomous navigation based on 3d vision," in *2014 IEEE/ASME 10th International Conference on Mechatronic and Embedded Systems and Applications (MESA)*, pp. 1–6, IEEE, 2014.

[47] G. Sibley, C. Mei, I. Reid, and P. Newman, "Vast-scale outdoor navigation using adaptive relative bundle adjustment," *The International Journal of Robotics Research*, vol. 29, no. 8, pp. 958–980, 2010.

[48] D. Scharstein and R. Szeliski, "A taxonomy and evaluation of dense two-frame stereo correspondence algorithms," *International journal of computer vision*, vol. 47, no. 1-3, pp. 7–42, 2002.

[49] A. A. Sharma, K. Neelathalli, D. Marculescu, and E. Nurvitadhi, "Hardware-efficient stereo estimation using a residual-based approach," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 2693–2696, IEEE, 2013.

[50] J. Kim, C. Ko, A. Frenzel, S. Ramanathan, and J. E. Hoffman, "Nanoscale imaging and control of resistance switching in vo 2 at room temperature," *Applied Physics Letters*, vol. 96, no. 21, p. 213106, 2010.

[51] R. Pierret, *Semiconductor Device Fundamentals*. Addison-Wesley, 1996.

[52] G. Rajahari, Y. A. Varshney, and S. C. Bose, "A novel design methodology for high tuning linearity and wide tuning range ring voltage controlled oscillator," in *VLSI Design and Test*, pp. 10–18, Springer, 2013.

[53] A. Van den Bosch, M. A. Borremans, M. S. Steyaert, and W. Sansen, "A 10-bit 1-gsample/s nyquist current-steering cmos d/a converter," *IEEE Journal of Solid-State Circuits*, vol. 36, no. 3, pp. 315–324, 2001.

[54] T. Miki, Y. Nakamura, M. Nakaya, S. Asai, Y. Akasaka, and Y. Horiba, "An 80-mhz 8-bit cmos d/a converter," *IEEE Journal of Solid-State Circuits*, vol. 21, no. 6, pp. 983–988, 1986.

[55] A. H. Van Roermund, A. Baschirotto, and M. Steyaert, *Nyquist AD converters, sensor interfaces, and robustness: advances in analog circuit design, 2012*. Springer Science & Business Media, 2012.

[56] P. G. Drennan and C. C. McAndrew, "Understanding mosfet mismatch for analog design," *IEEE Journal of solid-state circuits*, vol. 38, no. 3, pp. 450–456, 2003.

[57] P.-Y. Chiu and M.-D. Ker, "Metal-layer capacitors in the 65 nm cmos process and the application for low-leakage power-rail esd clamp circuit," *Microelectronics Reliability*, vol. 54, no. 1, pp. 64–70, 2014.

[58] T. Frank, A. Daffertshofer, C. Peper, P. Beek, and H. Haken, "Towards a comprehensive theory of brain activity:: Coupled oscillator systems under external forces," *Physica D: Nonlinear Phenomena*, vol. 144, no. 1-2, pp. 62–86, 2000.

[59] S. Sinha, G. Yeric, V. Chandra, B. Cline, and Y. Cao, "Exploring sub-20nm finfet design with predictive technology models," in *DAC Design Automation Conference 2012*, pp. 283–288, IEEE, 2012.

[60] K. Cao, "Predictive technology model." `http://ptm.asu.edu/`. [Online; accessed 4-September-2019].

[61] M. Darwish, *Graphene Devices for Beyond-CMOS Heterogeneous Integration*. PhD thesis, Carnegie Mellon University, 2017.

[62] S. B. Eryilmaz, E. Neftci, S. Joshi, S. Kim, M. BrightSky, H.-L. Lung, C. Lam, G. Cauwenberghs, and H.-S. P. Wong, "Training a probabilistic graphical model with resistive switching electronic synapses," *IEEE Transactions on Electron Devices*, vol. 63, no. 12, pp. 5004–5011, 2016.

[63] D. M. Chiarulli, B. Jennings, Y. Fang, A. Seel, and S. P. Levitan, "A computational primitive for convolution based on coupled oscillator arrays," in *2015 IEEE Computer Society Annual Symposium on VLSI*, pp. 125–130, IEEE, 2015.

[64] N. C. Wang, S. K. Gonugondla, I. Nahlus, N. R. Shanbhag, and E. Pop, "Gdot: A graphene-based nanofunction for dot-product computation," in *2016 IEEE Symposium on VLSI Technology*, pp. 1–2, IEEE, 2016.