

# **Charge Pump for Ultrasonic Implant Stimulation**

**André Filipe Canas Agostinho**

Thesis to obtain the Master of Science Degree in

## **Electrical and Computer Engineering**

Supervisors: Prof. Doutor Marcelino Bicho dos Santos  
Prof. Doutor Jorge Manuel dos Santos Ribeiro Fernandes

### **Examination Committee**

Chairperson: Prof. Doutor Francisco André Corrêa Alegria  
Supervisor: Prof. Doutor Marcelino Bicho dos Santos  
Member of the Committee: Prof. Doutor João Manuel Torres Caldeiras Simões Vaz

**November 2021**



# Declaration

I declare that this document is an original work of my own authorship and that it fulfills all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa.



# Abstract

This thesis main objective is the design optimization of a fully integrated Multiple Output Switched Capacitor Converter (MOSCC) to be used in a neural stimulator implant developed in the context of the M4M European project. Initially, the state of the art modeling and analysis of a Switched Capacitor Converter (SCC) is presented. The output resistance is analysed based on the topology charge multipliers and component sizes by considering the converter in two ideal regimes: Slow Switching Limit (SSL) and Fast Switching Limit (FSL). As this analysis, as it is found in the literature, is only adequate for single output converters it was extended to the case of multiple outputs by using a transimpedance model. The most significant work on this thesis is the proposal of a method for optimally sizing a given MOSCC topology for a specific application based on the previously presented analysis. The method takes as inputs the topology's charge multipliers, the characteristics of the components used to implement it and the application specification, and calculates the optimal size of each component to minimize area and losses. The proposed method was automated using a Python script and used to size a MOSCC for the M4M application. The obtained circuit and the optimization process were validated using simulation.

## Keywords

Design of integrated circuits; power management; DCDC conversion; switched capacitor converter; multiple outputs; optimization.



# Resumo

O objetivo principal desta tese é otimização do projeto de um Conversor de Condensadores Comutados com Múltiplas Saídas (CCCMS) totalmente integrado a ser aplicado num estimulador neuronal implantável, no contexto do projeto Europeu M4M. Inicialmente, é apresentado o estado da arte em modelação e análise de Conversores de Condensadores Comutados (CCC). A impedância de saída é analisada com base nos multiplicadores de carga da topologia e nas características dos componentes considerando que o conversor se encontra em dois regimes hipotéticos: limite de comutação rápida e limite de comutação lenta. Como esta análise, tal como encontrada na literatura, é apenas adequada para descrever conversores com uma única saída, neste trabalho esta abordagem foi expandida para o caso de múltiplas saídas com recurso a um modelo de transimpedância. A contribuição mais significativa deste trabalho é a proposta de um método para o dimensionamento otimizado de uma dada topologia de CCCMS, para uma aplicação específica, baseado na análise apresentada anteriormente. O método recebe como entrada os multiplicadores de carga da topologia, as características dos dispositivos que são usados para a implementar e a especificação da aplicação, e otimiza o projeto de forma a minimizar a área e perdas do conversor. O método proposto foi automatizado num *script* em Python e utilizado para dimensionar um CCCMS para a aplicação do M4M. O circuito obtido e o processo de otimização foram validados por simulação.

## Palavras Chave

Projeto de circuitos integrados; gestão de energia; conversão DCDC; conversor de condensadores comutados; saídas múltiplas; otimização.





# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Motivation . . . . .	3
1.2	Objectives . . . . .	4
1.3	Thesis outline . . . . .	4
<b>2</b>	<b>SCC modeling and analysis</b>	<b>5</b>
2.1	SCC model . . . . .	6
2.2	SCC output resistance analysis . . . . .	7
2.2.1	Slow Switching Limit Impedance . . . . .	8
2.2.2	Fast Switching Limit Impedance . . . . .	8
2.3	MOSCC model . . . . .	9
2.4	MOSCC output resistance analysis . . . . .	11
2.4.1	Slow switching limit impedance . . . . .	11
2.4.2	Fast switching limit impedance . . . . .	12
2.5	SCC components analysis . . . . .	13
2.6	Losses model . . . . .	15
2.6.1	Capacitor parasitic losses . . . . .	15
2.6.2	Switch driving losses . . . . .	15
2.6.3	Other losses . . . . .	16
2.6.4	Total losses and efficiency . . . . .	16
2.7	SCC stages . . . . .	16
<b>3</b>	<b>MOSCC sizing method</b>	<b>19</b>
3.1	Proposed methodology . . . . .	20
3.2	Stage conductance distributions generation . . . . .	21
3.3	Total conductance sizing . . . . .	21
3.4	Intra stage optimization . . . . .	22
3.4.1	Capacitor sizing to meet SSL stage impedance . . . . .	22
3.4.2	Switch sizing to meet FSL stage impedance . . . . .	22

3.4.3	Optimization of SSL and FSL stage impedance distribution . . . . .	23
3.5	Choosing the final sizing solution . . . . .	27
3.6	Limitations . . . . .	28
<b>4</b>	<b>M4M application</b>	<b>29</b>
4.1	Specifications . . . . .	30
4.2	Technology device implementation and characterization . . . . .	30
4.2.1	Capacitors . . . . .	31
4.2.1.A	MOM capacitors . . . . .	31
4.2.1.B	MIM capacitors . . . . .	31
4.2.1.C	MOS capacitors . . . . .	32
4.2.2	Switches . . . . .	33
4.3	Topology choice . . . . .	33
<b>5</b>	<b>Results</b>	<b>41</b>
<b>6</b>	<b>Conclusions</b>	<b>45</b>
6.1	Contributions . . . . .	46
6.2	Future work . . . . .	46
	<b>Bibliography</b>	<b>46</b>
<b>A</b>	<b>Optimization algorithm source code</b>	<b>49</b>
<b>B</b>	<b>Simulation schematics</b>	<b>65</b>

# List of Figures

1.1	Block diagram of the UHF neural stimulator . . . . .	3
2.1	(a) Example Switched Capacitor Converter (SCC) topology (Dickson), capacitor connection during (b) phase $\Phi_1$ , (c) phase $\Phi_2$ . DC output capacitor omitted. . . . .	6
2.2	Model of a SCC. . . . .	6
2.3	Example SCC topology (Dickson) with two outputs with ideal ratios 1/3 and 2/3. DC output capacitors omitted. . . . .	10
2.4	Model of an output of a Multiple Output Switched Capacitor Converter (MOSCC). . . . .	11
2.5	Capacitor parasitics. . . . .	14
2.6	Single SCC stage. . . . .	17
2.7	3 stages connected to produce 3 outputs $V_{o1}$ , $V_{o2}$ , and $V_{o3}$ with nominal outputs $1/3V_{in}$ , $2/3V_{in}$ , and $2V_{in}$ , respectively. Each output must have its own decoupling capacitor (not represented). . . . .	18
3.1	Variation of the stage area and losses with the ratio $r$ . Minimum points are marked. . . . .	26
3.2	Variation of the stage total cost with the ratio $r$ for different $\lambda$ . Minimum points are marked. . . . .	26
3.3	Performance of optimally sized topology for a set of stage conductance distributions. $\lambda = 0.02 \text{ mm}^2/\text{mW}$ . . . . .	27
4.1	(a) NMOS accumulation structure (b) PMOS accumulation structure with high impedance bias. . . . .	32
4.2	Stage connections for TOP1. . . . .	34
4.3	Stage connections for TOP2. . . . .	35
4.4	Stage connections for TOP3. . . . .	36
4.5	Stage connections for TOP4. . . . .	37
4.6	Stage connections for TOP5. . . . .	37
4.7	Stage connections for TOP6. . . . .	37
4.8	Efficiency vs Power density for topologies 1 to 6. $f_{sw} = 32 \text{ MHz}$ , $\lambda = 0.02$ . . . . .	38

5.1 Output voltages of the simulated SCC. The outputs are loaded with 4 mA each at 20  $\mu$ s and unloaded at 30  $\mu$ s. . . . . 42

B.1 Schematic of the implemented topology TOP2. . . . . 66

B.2 Schematic of stage ST1. . . . . 67

B.3 Schematic of the ideal gate driver. . . . . 68

# List of Tables

1.1	Converter specifications for ultrasonic implant stimulation application. . . . .	4
4.1	Converter specifications for ultrasonic implant stimulation application. . . . .	30
4.2	Performance metrics of the capacitance structures considered. . . . .	31
4.3	Performance metrics of MOSFET available to be used as switches. . . . .	33
4.4	Voltage step and minimum capacitor and switch voltage rating for each stage of topology TOP1. . . . .	34
4.5	Optimal stage conductance distribution, $r$ and area for each of the stages of TOP1 for $f_{sw} = 32$ MHz and $\lambda = 0.02$ . . . . .	34
4.6	Voltage step and minimum capacitor and switch voltage rating for each stage of topology TOP2. . . . .	35
4.7	Optimal stage conductance distribution, $r$ and area for each of the stages of TOP2 for $f_{sw} = 32$ MHz and $\lambda = 0.02$ . . . . .	36
4.8	Voltage step and minimum capacitor and switch voltage rating for each stage of topology TOP3. . . . .	36
4.9	Voltage step and minimum capacitor and switch voltage rating for each stage of topology TOP6. . . . .	37
4.10	Optimal stage conductance distribution, $r$ and area for each of the stages of TOP6 for $f_{sw} = 32$ MHz and $\lambda = 0.02$ . . . . .	38
4.11	Cost and performance of each of the considered topologies with optimized sizing. . . . .	38
5.1	Specified, theoretical and simulated values for TOP2. . . . .	43
5.2	M4M solution characteristics. . . . .	44



# Listings

A.1	File topology.py . . . . .	49
A.2	File topologies.py . . . . .	58
A.3	File tech_devices.py . . . . .	63





# Acronyms

<b>DCDC</b>	DC to DC power converter
<b>FSL</b>	Fast Switching Limit
<b>KCL</b>	Kirchoff's Current Law
<b>LDO</b>	Low Dropout Output regulator
<b>M4M</b>	Moore4Medical
<b>MIM</b>	Metal Insulator Metal
<b>MOM</b>	Metal Oxide Metal
<b>MOS</b>	Metal Oxide Semiconductor
<b>MOSCC</b>	Multiple Output Switched Capacitor Converter
<b>MOSFET</b>	Metal Oxide Semiconductor Field Effect Transistor
<b>MRI</b>	Magnetic Resonance Imaging
<b>SCC</b>	Switched Capacitor Converter
<b>SIMO</b>	Single Input Multiple Output
<b>SOSCC</b>	Single Output Switched Capacitor Converter
<b>SSL</b>	Slow Switching Limit
<b>UHF</b>	Ultra High Frequency



# 1

## Introduction

### Contents

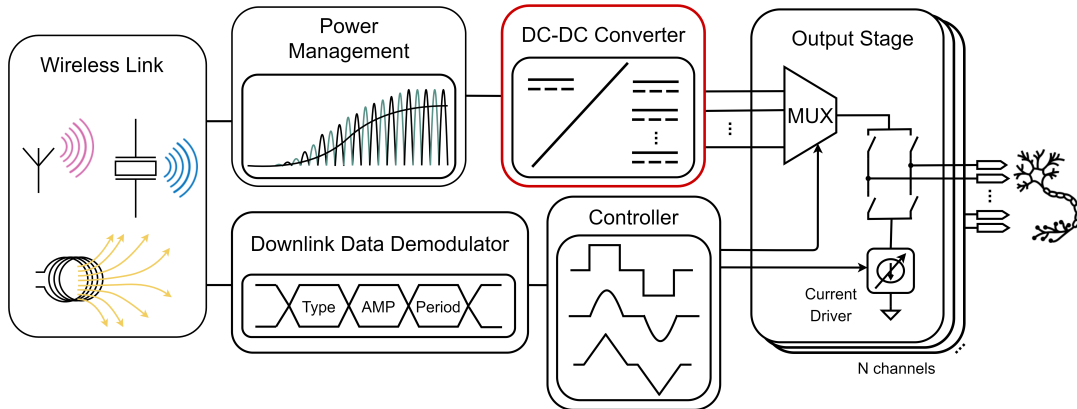
1.1	Motivation	3
1.2	Objectives	4
1.3	Thesis outline	4



## 1.1 Motivation

Moore4Medical (M4M) is a project funded by the European Union and headed by a consortium composed of multiple companies and universities throughout Europe. The project addresses emerging medical applications and technologies that offer significant new opportunities for patients as well as for industry including: bioelectronic medicines, organ-on-chip, drug adherence monitoring, smart ultrasound, radiation free interventions and continuous monitoring. The new technologies will help fighting the increasing cost of healthcare by reducing the need for hospitalisation, helping to develop personalized therapies, and realising intelligent point-of-care diagnostic tools [1].

Introduced in the project is a demonstrator for a implantable Ultra High Frequency (UHF) neural stimulator powered by an ultrasonic link whose architecture is illustrated in Figure 1.1. In the scope of M4M, this work focuses on the design of the Single Input Multiple Output (SIMO) DC to DC power converter (DCDC) converter which supplies the output stage with the multiple voltage rails required for efficient stimulation. This thesis' work was developed in cooperation with Silicongate Lda, one partner of the M4M consortium responsible for the development of the SIMO DCDC converter.



**Figure 1.1:** Block diagram of the UHF neural stimulator<sup>1</sup>.

The SIMO DCDC in Figure 1.1 is required to generate voltages both above and below the input voltage of 4.5 V with maximum total power in the order of 100 mW for outputs between 1.5 V and 9 V. Since up-conversion is required, Low Dropout Output regulator (LDO) regulators cannot be used. Inductive converters also face some challenges in this application. Being the application implantable, it must be designed to handle being inside Magnetic Resonance Imaging (MRI) machines which use radio frequency and magnetic fields that would interfere with any magnetic core inductor used, such that much bigger air core inductors would be needed. Switched Capacitor Converters (SCCs) turn out to be ideal for this type of applications due to the possibility of full integration in the silicon chip, that results in a small total area, absence of magnetic components and high efficiency. This Multiple Output Switched

<sup>1</sup>Figure drawn by Konstantina Kolovou Kouri (who is also part of the M4M project), reproduced with the author's permission

Capacitor Converter (MOSCC) topology must be chosen correctly and its components correctly sized to obtain a small occupied area and high efficiency while meeting the specifications. There is a lack of literature available about the subject of sizing MOSCCs and, to the best knowledge of the author, no method as been proposed that can be directly applied to solving this problem.

## 1.2 Objectives

In the scope of this master thesis, a method for optimally sizing the components of a MOSCC is to be developed. The method is then used to aid in the development of a MOSCC solution for the specific application of a UHF neural stimulator implant. The specifications for the MOSCC were provided by the partners of Silicongate in the M4M project and are listed in Table 1.1.

**Table 1.1:** Converter specifications for ultrasonic implant stimulation application.

Spec	Value
$V_{in}$	min: 4.5 V, typ: 4.6 V, max: 4.7 V
Ideal $V_{out}$ ratios	1/3, 2/3, 4/3, 5/3, 6/3
Ideal $V_{out}$ @ $V_{in} = 4.5$ V	1.5 V, 3.0 V, 6.0 V, 7.5 V, 9.0 V
$I_{out,max}$	4 mA per output
$\Delta V_{out,max}$	5% of $V_{out}$ @ 4.5 V
$V_{out,PP,max}$	2% of $V_{out}$ @ 4.5 V
Fabrication Technology	TSMC 0.18um BCD GenII
Integration	Fully integrated (no external capacitors)

## 1.3 Thesis outline

Chapter 2 presents the modeling and analysis of SCC and its extension to MOSCC including the modeling of losses in the converter. Chapter 3 presents the proposed method for optimization of the component sizing of a MOSCC to meet a given specification. Chapter 4 presents the design choices for the M4M application making use of the previously proposed method to size several topologies' and evaluate their performance to aid in the choice of the most advantageous solution. Chapter 5 presents the simulation results for the implemented SCC and their correlation to the theoretical analysis. Chapter 6 presents the conclusions and points at possible future improvements.

# 2

## SCC modeling and analysis

### Contents

---

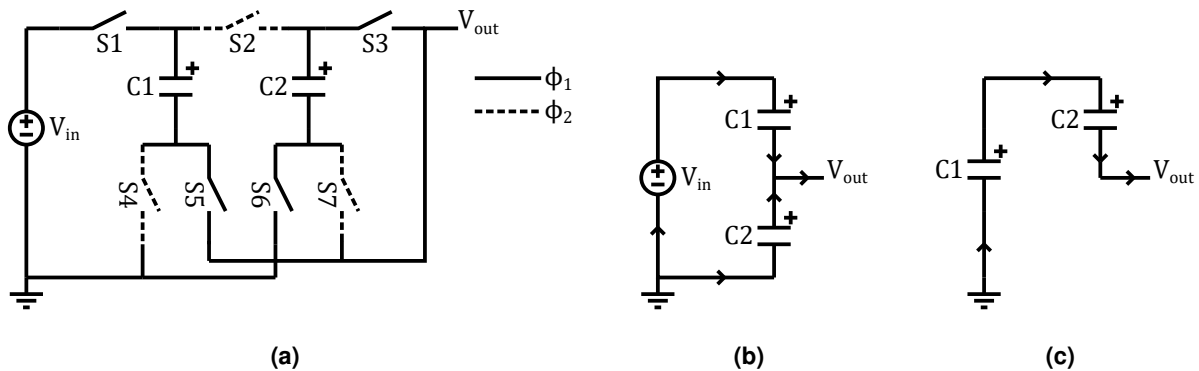
2.1	SCC model . . . . .	6
2.2	SCC output resistance analysis . . . . .	7
2.3	MOSCC model . . . . .	9
2.4	MOSCC output resistance analysis . . . . .	11
2.5	SCC components analysis . . . . .	13
2.6	Losses model . . . . .	15
2.7	SCC stages . . . . .	16

---

Extensive research has been published on the analysis and design of SCC in [2–7] to name a few. However, most published research on the topic focuses in Single Output Switched Capacitor Converter (SOSCC) converters with only a very small amount of research found on MOSCC. This chapter presents an approach to the analysis of SOSCC and its extension to MOSCC.

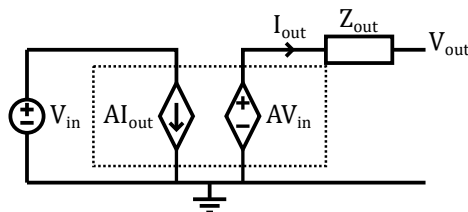
## 2.1 SCC model

SCC are composed of capacitors whose connections change according to the phase, being redefined by switches. By alternating cyclically between the different phases, voltage conversion ratios are realized between an input voltage and an output. Figure 2.1(a) shows an example two phase SCC using a Dickson topology [8] that multiplies the input voltage by  $1/3$ . In phase  $\Phi_1$  switches  $S1$ ,  $S3$ ,  $S5$  and  $S6$  are turned-on resulting in the capacitor connections in Figure 2.1(b), while in phase  $\Phi_2$  switches  $S2$ ,  $S4$  and  $S7$  are turned-on resulting in the capacitor connections in Figure 2.1(c). When the converter is unloaded,  $C1$  is charged to  $2/3V_{in}$  while  $C2$  is charged to  $1/3V_{in}$ . An output capacitor is required but omitted.



**Figure 2.1:** (a) Example SCC topology (Dickson), capacitor connection during (b) phase  $\Phi_1$ , (c) phase  $\Phi_2$ . DC output capacitor omitted.

The most common equivalent circuit for a SCC [2, 9, 10] is an ideal DC transformer with voltage gain  $A$  and an output resistance  $Z_{out}$  as presented in Figure 2.2.



**Figure 2.2:** Model of a SCC.

The output resistance models the converter losses that depend on load current, which include charge



redistribution losses [11] between the capacitors and input voltage source, and resistance in capacitor charge/discharge loops. The output resistance is of central importance for the sizing of a SCC and, therefore, it will be analysed in detail in the next section.

## 2.2 SCC output resistance analysis

The output resistance will be analysed using the methods in [7] as it generates simple expressions that make component sizing optimization more practical and is expandable to MOSCC as will be seen in section 2.3. The converter is assumed to be loaded with a current source in parallel with a voltage source (zero output ripple) and analysed in two asymptotic limits the Slow Switching Limit (SSL) and the Fast Switching Limit (FSL). In SSL it is considered that full charge transfer occurs (zero parasitic resistances or zero switching frequency) so that losses are only due to charge redistribution in the capacitors, while in FSL the capacitor voltage is constant (infinite switching frequency) and losses are only due to switch resistance. An output resistance is derived for each regime  $R_{SSL}$  and  $R_{FSL}$  and combined using the following approximation [7]:

$$R_{out} \approx \sqrt{R_{SSL}^2 + R_{FSL}^2} \quad (2.1)$$

To calculate the  $R_{SSL}$  and  $R_{FSL}$  a set of charge multipliers are derived for the topology in study. Each charge multiplier  $a_{c,i}^j$  is defined as the ratio between the charge  $q_{c,i}^j$  flowing into capacitor  $i$  during switching phase  $j$  and the output charge during the switching period  $q_{out}$ . The charge multipliers can be obtained by inspection of the topology in each of the switching phases by applying Kirchoff's Current Law (KCL) and noting that the sum of capacitor charge flows during a switching period must be zero in steady state operation. Similarly, for switches, each  $a_{r,i}^j$  is defined as the ratio between the charge  $q_{r,i}^j$  flowing into switch  $i$  during switching phase  $j$  and the output charge  $q_{out}$  during the switching period.

For a two phase SCC the capacitor charge multipliers are symmetric between the two phases so  $a_{c,i}$  is defined as  $a_{c,i} = a_{c,i}^1 = -a_{c,i}^2$  grouped in the vector  $a_c$ :

$$a_c = [q_{c,1} \ \dots \ q_{c,n}]^T / q_{out} \quad (2.2)$$

while the switch charge multipliers  $a_{r,i}$  are defined to be equal to the charge multiplier of the phase in which they are turned ON and grouped in the vector  $a_r$ :

$$a_r = [q_{r,1} \ \dots \ q_{r,n}]^T / q_{out} \quad (2.3)$$

For the example, topology in Figure 2.1 these vectors are the following:

$$a_c = \left[ \frac{1}{3} \quad -\frac{1}{3} \right]^T \quad (2.4)$$

$$a_r = \left[ \frac{1}{3} \quad \frac{1}{3} \quad \frac{1}{3} \quad -\frac{1}{3} \quad \frac{1}{3} \quad -\frac{1}{3} \quad \frac{1}{3} \right]^T \quad (2.5)$$

### 2.2.1 Slow Switching Limit Impedance

In SSL the energy loss and output resistance are attributed solely to capacitor charge redistribution. The energy loss in SSL regime  $E_{SSL}$  in each switching period is given by [9]

$$E_{SSL} = \sum_{i \in caps} q_{c,i} \Delta v_{c,i} \quad (2.6)$$

where  $caps$  is the set of capacitors that compose the SCC,  $q_{c,i} = a_{c,i} q_{out}$  and  $\Delta v_i$  is capacitor  $i$  ripple computed from

$$\Delta v_{c,i} = q_{c,i} / C_i \quad (2.7)$$

where  $C_i$  is the capacitance of capacitor  $i$ . By dividing by  $q_{out}^2$  yields:

$$E_{SSL} = q_{out}^2 \sum_{i \in caps} \left( \frac{q_{c,i}}{q_{out}} \right)^2 \frac{1}{C_i} \quad (2.8)$$

Noting that  $q_{c,i}/q_{out} = a_{c,i}$ , multiplying by the switching frequency  $f_{sw}$  and substituting  $q_{out} f_{sw} = i_{out}$  yields the total average power loss in SSL  $P_{SSL}$ :

$$P_{SSL} = i_{out}^2 \sum_{i \in caps} \frac{a_{c,i}^2}{C_i f_{sw}} \quad (2.9)$$

Since the power loss is proportional to the output current squared the SSL output resistance  $R_{SSL}$  is given by:

$$R_{SSL} = \sum_{i \in caps} \frac{a_{c,i}^2}{C_i f_{sw}} \quad (2.10)$$

### 2.2.2 Fast Switching Limit Impedance

In FSL the capacitor ripple is zero and the current passing through a switch when it is ON is constant. In this case all energy losses are attributed to the switch resistance. The average current in each switch during the time it is ON is given by the charge flow through the switch  $q_{r,i}$  divided by the time it is ON

which is the product of the switching period  $1/f_{sw}$  by the switch duty cycle  $D$ :

$$i_{r,i} = q_{r,i} f_{sw} / D \quad (2.11)$$

Substituting  $q_{r,i} = a_{r,i} q_{out}$  and  $q_{out} = i_{out} / f_{sw}$  into 2.11 yields

$$i_{r,i} = a_{r,i} i_{out} / D \quad (2.12)$$

Computing the instantaneous power loss in each switch and multiplying by the duty cycle yields the total power loss in FSL  $P_{FSL}$ .

$$P_{FSL} = \sum_{i \in sws} D R_i (a_{r,i} i_{out} / D)^2 \quad (2.13)$$

where  $sws$  is the set of switches that compose the SCC,  $R_i$  is switch  $i$  on-state resistance. Simplifying the expression and substituting the switch resistance by the inverse of its conductance  $G_i$  yields

$$P_{FSL} = i_{out}^2 \sum_{i \in sws} \frac{a_{r,i}^2}{G_i D} \quad (2.14)$$

Since  $P_{FSL}$  is proportional to the output current squared the FSL output resistance is given by

$$R_{FSL} = \sum_{i \in sws} \frac{a_{r,i}^2}{G_i D} \quad (2.15)$$

## 2.3 MOSCC model

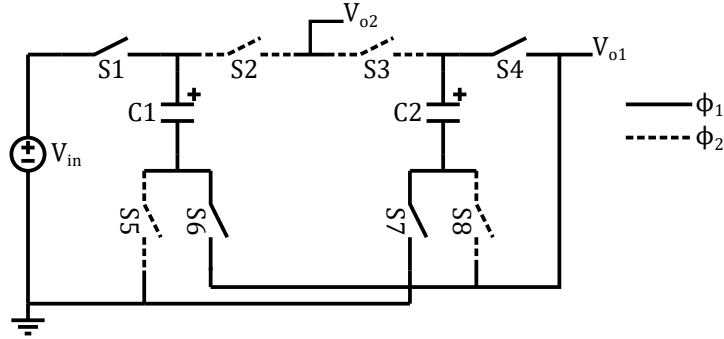
In a system where multiple output voltages are required, the traditional approach is to design an independent converter for each output. If the required output voltages are close, and load requirements on each output are similar, then a single converter can be designed and reused for each output. Nevertheless in systems requiring widely different output voltages, possibly including both step-up and step-down conversion, and/or widely different load currents, different converters must be designed for each output. This approach is not only impractical due to the number of different designs but it also introduces area overheads and power overheads that can hinder the total system power density and efficiency.

As analysed by [12], a SCC can be made to provide multiple output voltages without increasing the component count, providing gains in terms of circuit and design simplicity and achieving more compact and possibly more efficient solutions. Two methods can be employed to generate multiple output voltages, the first one consists of connecting internal nodes to outputs and the second one consists of adding switching phases. Both methods can be combined with the maximum number  $m$  of outputs given

by

$$m = n(s - 1) \quad (2.16)$$

where  $n$  is the number of capacitors and  $s$  is the number of switching phases. As stated previously, this work focuses on 2 phase SCC. Therefore the maximum number of outputs becomes equal to the number of capacitors  $n = m$ . An example topology with multiple outputs is shown in Figure 2.3. The topology is based on the single output example topology in Figure 2.1 but an extra switch allows an internal node to be connected to a second output with an ideal conversion ratio of 2/3. Each of the outputs requires an output capacitor (omitted from the figure) to control output voltage ripple.



**Figure 2.3:** Example SCC topology (Dickson) with two outputs with ideal ratios 1/3 and 2/3. DC output capacitors omitted.

Since, in a MOSCC, each capacitor and switch can be used in the charge transfer to multiple outputs a MOSCC presents coupling between the various outputs that manifest as a voltage drop (or rise) in one of the outputs when another output is loaded. Given this coupling, the simple output resistance model cannot be used with MOSCC. A transimpedance model is proposed by [10] which models the coupling between outputs using a transimpedance matrix  $\mathbf{Z}$  (2.17). This matrix is symmetric and has size  $[m \times m]$ , being  $m$  the number of outputs considered. Each entry  $z_{xy}$  relates the voltage drop in output  $x$  with the current drawn by output  $y$ . Each diagonal element  $z_{xx}$  is the output impedance of output  $x$  as defined for a single output SCC. The model of a single output is drawn in Figure 2.4, showing that each output is still modeled with a transformer with the gain for that output, while the output impedance is substituted by a transimpedance taking into account the currents on all outputs.

$$\mathbf{Z} = \begin{bmatrix} z_{11} & z_{21} & \dots & z_{1m} \\ z_{21} & z_{22} & \dots & z_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ z_{m1} & z_{m2} & \dots & z_{mm} \end{bmatrix} [\Omega] \quad (2.17)$$

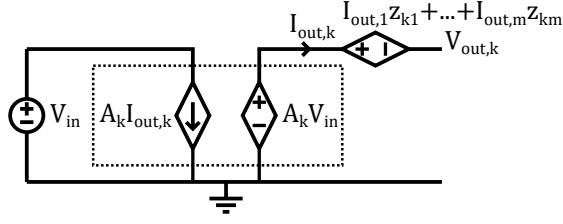


Figure 2.4: Model of an output of a MOSCC.

## 2.4 MOSCC output resistance analysis

The voltage drop on each output  $\Delta V_{out}$  and the total power loss due to output impedance  $P_{Rout}$  is given, respectively, by (2.18) and (2.19) where  $i_o$  is the output currents vector.

$$\Delta V_{out} = Z i_{out} \quad (2.18)$$

$$P_{Rout} = \Delta V_{out} i_{out} = \sum_{j=1}^m \sum_{i=1}^m z_{ij} i_{out,i} i_{out,j} \quad (2.19)$$

Since capacitors and switches are shared between multiple outputs, the charge transferred through each element, each cycle, is also dependent of the charge transferred to each output.

### 2.4.1 Slow switching limit impedance

The charge,  $q_{c,i}$ , transferred by the  $i^{th}$  capacitor, in each phase, is then given by a linear combination of the output charges  $q_{out,k} = i_{out,k} / f_{sw}$ :

$$q_{c,i} = \sum_{k \in outs} b_{c,ik} q_{out,k} \quad (2.20)$$

where  $outs$  is the set of outputs of the SCC and the charge multiplier  $b_{c,ik}$  is the coefficient relating the charge transferred through the  $i^{th}$  capacitor with the charge transferred to the  $k^{th}$  output. This coefficients can be calculated similarly to the single output case charge multipliers by applying KCL, solving for each  $q_{c,i}$  and letting all  $q_{out,k}$  as unknowns. The charge multipliers  $b_c$  for the example topology in Figure 2.3 are:

$$B_c = \begin{bmatrix} \frac{1}{3} & \frac{2}{3} \\ -\frac{1}{3} & \frac{1}{3} \end{bmatrix} \quad (2.21)$$

Similarly to the single output case the energy loss in each switching period by charge redistribution  $E_{SSL}$  is obtained by substituting (2.7) into (2.6):

$$E_{SSL} = \sum_{i \in caps} \frac{q_{c,i}^2}{C_i} \quad (2.22)$$

Substituting  $q_{c,i}$  by (2.20) and multiplying by  $f_{sw}$  yields

$$P_{SSL} = \sum_{i \in caps} \left[ \left( \sum_{k \in outs} b_{c,ik} i_{out,k} \right)^2 / (f_{sw} C_i) \right] \quad (2.23)$$

Expanding the square and rearranging the sums leads to

$$P_{SSL} = \sum_{l \in outs} \sum_{k \in outs} \left[ \left( \sum_{i \in caps} \frac{b_{c,ik} b_{c,il}}{f_{sw} C_i} \right) i_{out,k} i_{out,l} \right] \quad (2.24)$$

Similarly to (2.19)  $P_{SSL}$  is also given by

$$P_{SSL} = \sum_{j \in outs} \sum_{i \in outs} z_{SSL,ij} i_{out,i} i_{out,j} \quad (2.25)$$

where  $z_{SSL,xy}$  is an element of the matrix  $Z_{SSL}$  defined analogously to  $Z$  (2.17). The transimpedance matrix elements in (2.25) can be directly matched to the charge flow multipliers in (2.24) as

$$z_{SSL,kl} = \sum_{i \in caps} \frac{b_{c,ik} b_{c,il}}{f_{sw} C_i} \quad (2.26)$$

## 2.4.2 Fast switching limit impedance

Analogously to the capacitors, each switch transferred charge is also dependent on output currents as

$$q_{r,i} = \sum_{k \in outs} b_{r,ik} q_{out,k} \quad (2.27)$$

where the charge multiplier  $b_{r,ik}$  is the coefficient relating the charge transferred through the  $i^{th}$  switch with the charge transferred to the  $k^{th}$  output. The charge multipliers  $b_r$  for the example topology in Figure 2.3 are:

$$B_r = \begin{bmatrix} \frac{1}{3} & \frac{2}{3} \\ \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} \\ -\frac{1}{3} & \frac{1}{3} \\ -\frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{2}{3} \\ \frac{1}{3} & \frac{1}{3} \\ \frac{1}{3} & \frac{1}{3} \\ -\frac{1}{3} & \frac{1}{3} \\ -\frac{1}{3} & \frac{1}{3} \end{bmatrix} \quad (2.28)$$

The FSL loss  $P_{FSL}$  dependence on the switch charge multipliers is given by

$$P_{FSL} = \sum_{l \in outs} \sum_{k \in outs} \left[ \left( \sum_{i \in sws} \frac{b_{r,ik} b_{r,il}}{DG_i} \right) i_{out,k} i_{out,l} \right] \quad (2.29)$$

$P_{FSL}$  is also given by

$$P_{FSL} = \sum_{j \in outs} \sum_{i \in outs} z_{FSL,ij} \dot{i}_{out,i} \dot{i}_{out,j} \quad (2.30)$$

where  $z_{FSL,xy}$  is an element of the matrix  $Z_{FSL}$  defined analogously to  $Z$ . The transimpedance matrix elements in (2.30) can be directly matched to the charge flow multipliers in (2.29) as

$$z_{FSL,kl} = \sum_{i \in sws} \frac{b_{r,ik} b_{r,il}}{DG_i} \quad (2.31)$$

Following [10], the total transimpedance matrix elements can be calculated as

$$z_{kl} = \sqrt{z_{SSL,kl}^2 + z_{FSL,kl}^2} \quad (2.32)$$

from which follows

$$P_{rout} = \sum_{j \in outs} \sum_{i \in outs} z_{ij} \dot{i}_{out,i} \dot{i}_{out,j} \quad (2.33)$$

and

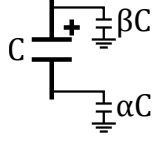
$$P_{rout} = \sqrt{P_{SSL}^2 + P_{FSL}^2} \quad (2.34)$$

## 2.5 SCC components analysis

The practical performance of any SCC is largely dependent on the performance of the components used to implement the circuit, therefore it is crucial to model and to access the performance of these devices and their influence in the overall converter performance.

Apart from the useful capacitance they provide, the real capacitors used in SCC, and their interconnects, present parasitic capacitances to other nodes. This capacitances are usually referred to a ground node and can be lumped in each capacitor bottom and top plate [6], as pictured in Figure 2.5.  $\alpha C$  and  $\beta C$  are, respectively, the bottom plate parasitic capacitance and the top plate parasitic capacitance. In designs with external flying capacitors, the parasitics are mostly caused by the interconnects to the capacitors (internal traces, wire bonds, package leads and PCB traces) and their proximity to other nets with different potentials. In fully integrated designs the parasitics are mostly caused by the actual capacitor plates proximity to the substrate, usually tied to the ground net. Real switches present a non-zero resistance when turned-on and require a given energy to be turned-on.

The performance of capacitors and switches can be evaluated using a range of metrics relating their value with their cost of implementation. [7] defines metrics adequate to describe very generally devices independent of technology and by assuming that all devices are predictably scalable with voltage rating. Capacitors are described using areal energy density and switches are described using their



**Figure 2.5:** Capacitor parasitics.

power handling capacity per area and per capacitive switching loss.

The assumption of predictable scalability with voltage rating is often not applicable due to technology limitations. This is specially true for integrated devices which are offered by each fabrication technology with quite different physical implementations to achieve different voltage ratings. In this work, the value of capacitors is its capacitance while the value of switches is their conductance in the on-state. Two costs are considered, area and losses, resulting in two metrics for each kind of device.  $m_{Ca}$  and  $m_{Cl}$  for capacitor and  $m_{Sa}$  and  $m_{Sl}$  for switches.  $m_{Ca}$  is the capacitor areal capacitance density given by

$$m_{Ca,i} = \frac{C_i}{A_i} \quad (2.35)$$

where  $C_i$  is the main capacitance and  $A_i$  is the occupied layout area.  $m_{Cl}$  is the ratio of main capacitance with total parasitic capacitance defined by

$$m_{Cl,i} = \frac{C_i}{\alpha_i C_i + \beta_i C_i} = \frac{1}{\alpha_i + \beta_i} \quad (2.36)$$

where  $\alpha_i$  and  $\beta_i$  are, respectively, the ratio between bottom and top plate parasitic capacitances to the main capacitance (see Figure 2.5).  $m_{Sa}$  is the switch conductance per area defined by

$$m_{Sa,i} = \frac{G_i}{A_i} \quad (2.37)$$

where  $G_i$  is the switch conductance or the inverse of the switch resistance  $R_i$  and  $A_i$  is the switch area.  $m_{Sl}$  is the ratio of switch conductance to energy required to drive the switch

$$m_{Sl,i} = \frac{G_i}{E_{drv,i}} \quad (2.38)$$

where  $E_{drv,i}$  is the energy required to drive the switch a single switching cycle. These metrics are more adequate to describe the performance of specific technology devices, for specific voltage levels, but cannot be extrapolated for other voltage ratings. In fact, for a specific technology, the performance and cost of each device has be evaluated individually.



## 2.6 Losses model

Losses derived from charge redistribution and switch resistance are lumped in the output impedance/transimpedance of the converter and have been analysed in section 2.4. SCCs present additional losses due to parasitics in capacitors, switches and interconnects which represent a significant part of the total losses, specially, in fully integrated implementations.

### 2.6.1 Capacitor parasitic losses

Since the flying capacitors terminals are switched between two different potentials during the switching cycle the parasitic capacitances get charged and discharged in a lossy way. The energy wasted in a cycle by the parasitic capacitances of the  $i^{th}$  is given by [7]:

$$E_{cpar,i} = (\alpha_i + \beta_i) C_i (v_{cb,i})^2 \quad (2.39)$$

where  $\alpha_i$  and  $\beta_i$  are the bottom and top plate capacitor parasitic fractions and  $v_{cb,i}$  is the magnitude of the voltage swing in the capacitor bottom plate (or top plate), which depends on the placement of the capacitor in the topology and the input voltage.  $E_{cpar}$  can also be calculated using the capacitor's loss metric defined previously in (2.36):

$$E_{cpar,i} = \frac{C_i (v_{cb,i})^2}{m_{Cl,i}} \quad (2.40)$$

The total power wasted in the parasitic capacitors  $P_{cpar}$  can then be calculated by summing the losses on all capacitors and multiplying by  $f_{sw}$ :

$$P_{cpar} = f_{sw} \sum_{i \in caps} \frac{C_i (v_{cb,i})^2}{m_{Cl,i}} \quad (2.41)$$

Although each capacitor voltage swing is dependent on output loading and, to a lesser extent, on  $\alpha$  and  $\beta$  of each of the converter capacitors [6], those effects are disregarded. The model may then present deviations when the SCC is operated with high output voltage drop or capacitors with very high parasitics.

### 2.6.2 Switch driving losses

The only losses considered in the switches are a consequence of the power necessary to drive them  $P_{sdrv}$ . In the case of a switch implemented with a Metal Oxide Semiconductor Field Effect Transistor (MOSFET), which accounts for most practical applications, these losses are the power dissipated in charging and discharging the gate capacitance. The losses caused by the switches on-state resistance

are already accounted for in the output resistance losses  $P_{rout}$ . The driving losses in a MOSFET switch scales linearly with the required switch conductance. Since gate drive voltage influences the MOSFET conductance and drive losses simultaneously and in a non-linear way, the energy required to drive the MOSFET  $E_{sdrv,ref}$  can instead be measured for a given transistor size and drive voltage and then scaled to meet the required switch conductance  $G_i$  yielding

$$E_{sdrv,i} = E_{sdrv,ref} \times \left( \frac{G_i}{G_{ref}} \right) \quad (2.42)$$

Using the switch loss metric defined previously in (2.38),  $E_{sdrv,i}$  can be calculated as

$$E_{sdrv,i} = \frac{G_i}{m_{sl,i}} \quad (2.43)$$

Multiplying  $E_{sdrv,i}$  by the switching frequency  $f_{sw}$  and summing over all switches yields the total power dissipated driving the switches  $P_{sdrv}$ :

$$P_{sdrv} = f_{sw} \sum_{i \in sws} \frac{G_i}{m_{sl,i}} \quad (2.44)$$

### 2.6.3 Other losses

In practice there are other losses in a system due to the resistivity of interconnects, static and dynamic consumption of auxiliary circuits, leakage, etc. Yet, these are not considered since some of them are constant, independent of the SCC topology, and others are too complex to model, being relatively small and providing little use in numeric modeling and analysis of different SCC topologies.

### 2.6.4 Total losses and efficiency

The total SCC losses is given by

$$P_{loss} = P_{rout} + P_{cpar} + P_{sdrv} \quad (2.45)$$

where  $P_{rout}$ ,  $P_{cpar}$  and  $P_{sdrv}$  are given by (2.34), (2.41) and (2.44), respectively.

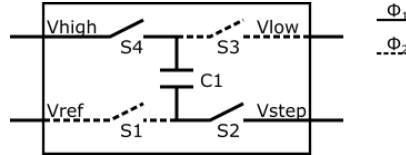
The total converter efficiency is then given by

$$\eta = \frac{P_{out}}{P_{in}} = \frac{P_{out}}{P_{out} + P_{loss}} = \frac{\sum_{k \in outs} V_{out,k} I_{out,k}}{\sum_{k \in outs} V_{out,k} I_{out,k} + P_{loss}} \quad (2.46)$$

## 2.7 SCC stages

To streamline the analysis and design of two phase SCC it is common to construct topologies based on stages of similar composition [6, 13, 14]. Figure 2.6 shows the composition of one such stage with

one capacitor and four switches that allow switching each of the capacitor's terminals to two different voltages. On clock phase  $\Phi_1$  switches S2 and S4 are driven ON, connecting the capacitor between  $V_{high}$  and  $V_{step}$ , while on clock phase  $\Phi_2$  S1 and S3 are driven ON, connecting the capacitor between  $V_{ref}$  and  $V_{low}$ . Although this stage configuration allows the capacitor terminals to be connected to any two node voltages on a circuit, thus allowing any two phase topology to be realized, it may result in redundant switches when implementing some SCC topologies.



**Figure 2.6:** Single SCC stage.

Considering an unloaded converter, the capacitor voltage is constant in the two phases such that

$$V_{high} - V_{step} = V_{low} - V_{ref} \quad (2.47)$$

Defining  $V_{\Delta} = V_{step} - V_{ref}$  then  $V_{high}$  is given by

$$V_{high} = V_{low} + (V_{\Delta}) \quad (2.48)$$

In terms of voltage ratings the capacitor in each stage must handle a voltage of  $V_C = V_{low} - V_{ref}$  while all four switches must be rated to handle  $V_{\Delta}$ .

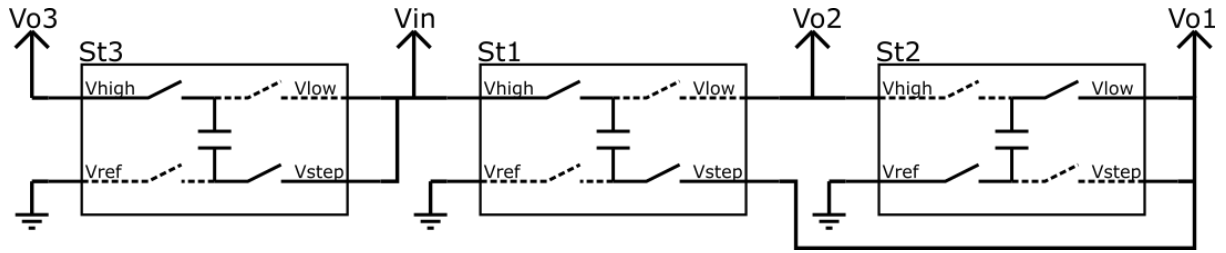
Since the switches are in series with the capacitor in the phase they are turned-on the four switches have charge multipliers  $b_{s,ik}$  equal to the capacitor charge multipliers  $b_{c,ik}$ . This allows the definition of a set of stage's charge multipliers  $b_{st,ik}$  as equal to  $b_{c,ik}$ .

This stages can be used to describe both step-up conversion if  $V_{low}$  is the input or step-down conversion if  $V_{high}$  is the input. Figure 2.7 shows an example converter that provides 3 outputs using 3 stages. Stages 1 and 2 perform down-conversion with  $V_{\Delta} = V_{o1} - 0 = 1/3V_{in}$ , resulting in outputs with voltages  $2/3V_{in}$  and  $1/3V_{in}$ . Stage 3 performs up-conversion with  $V_{\Delta} = V_{in} - 0 = V_{in}$ , resulting in a doubler output  $2V_{in}$ .

The charge multipliers for this stages are

$$\mathbf{B}_{st} = \begin{bmatrix} \frac{1}{3} & \frac{2}{3} & 0 \\ -\frac{1}{3} & \frac{1}{3} & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.49)$$

where it can be noted that stages  $ST1$  and  $ST2$  present null charge multipliers  $b_{st,13}$  and  $b_{st,23}$ , that will result in the elements  $z_{13}, z_{31}, z_{23}$  and  $z_{32}$  of the transimpedance matrix being also null. This causes



**Figure 2.7:** 3 stages connected to produce 3 outputs  $V_{o1}$ ,  $V_{o2}$ , and  $V_{o3}$  with nominal outputs  $1/3V_{in}$ ,  $2/3V_{in}$ , and  $2V_{in}$ , respectively. Each output must have its own decoupling capacitor (not represented).

outputs  $V_{o1}$  and  $V_{o2}$  to be independent from output  $V_{o3}$ , such that loading the output  $V_{o3}$  causes no voltage drop on outputs  $V_{o1}$  and  $V_{o2}$  and loading outputs  $V_{o1}$  and  $V_{o2}$  causes no voltage drop on output  $V_{o3}$ .

# 3

## MOSCC sizing method

### Contents

---

3.1	Proposed methodology . . . . .	20
3.2	Stage conductance distributions generation . . . . .	21
3.3	Total conductance sizing . . . . .	21
3.4	Intra stage optimization . . . . .	22
3.5	Choosing the final sizing solution . . . . .	27
3.6	Limitations . . . . .	28

---

The naive approach to sizing of the SCC is to size every stage with the same capacitance and switch conductance. Depending on the topology, technology and load distribution between the various outputs, this sizing might result in very poor performance. A method is proposed by [7] to optimize capacitor and switch relative sizing based on the device's charge multiplier and performance metrics which results in highly optimized component sizing on single output charge pumps. This method cannot be used with multiple output converters since the charge multipliers are dependent on output current distribution. A similar method is applied by [10] to multiple output converters by considering a single operating point for which the circuit is to be optimized and calculating the charge multipliers for that specific case. Still this method fails to ensure that the voltage drop on each output is lower than a specification in the entire load distribution range and it does not take into account the device performance metrics which might result in a low performance solution if, for example, capacitor technologies with widely different performance are used in different stages.

In the following sections a method is presented to optimally size the capacitors and switches of a MOSCC topology to adhere to a given specification with the objective of minimizing the implementation costs of the SCC, area and power losses.

### 3.1 Proposed methodology

The proposed method consists on the following steps:

1. Generate a set of stage conductance distributions.
2. For each conductance distribution:
  - (a) Find the total conductance required to keep all outputs above the maximum voltage drop at maximum load.
  - (b) Optimize the distribution of  $Z_{st,SSL}$  and  $Z_{st,FSL}$  within each stage to minimize the cost each stage.
  - (c) Calculate the total solution cost taking into account total area and losses.
3. Choose the conductance distribution with the lowest cost as the optimal solution.

The optimal stage conductance is found by evaluating a set of solutions in a search space and choosing the best solution. This is necessary due to the extra constraints caused by the maximum voltage drop specifications for each output which do not allow a general expression to be found for the optimal stage conductance.

This algorithm was implemented in a Python program that takes as inputs the topology characteristics (charge multipliers), the technology device characteristics (device performance metrics) and, the

application specifications and outputs the optimized topology sizing and its performance. The Python code listings are presented in Appendix A.

The following sections will present the above mentioned steps.

### 3.2 Stage conductance distributions generation

The search space is created by generating all permutations (with repetition) of integers from 1 to  $res$  (the resolution of the sizing), normalizing each number by dividing by the total such that they sum to 1 and assigning the normalized values  $h_i$  to each of the topology stages. The conductance of each stage is given as

$$G_{st,i} = h_i G_{st,tot} \quad (3.1)$$

where  $G_{st,tot}$  is the sum of all stages conductance and needs to be calculated to meet the voltage drop specification on each of the outputs.

On a SCC with  $n$  stages the search space is composed of  $res^n$  solutions (conductance distributions). Increasing the sizing resolution may provide better optimized solutions but may increase the search space considerably to the point it may become impractical.

### 3.3 Total conductance sizing

Assuming a given stage conductance distribution, the required  $G_{st,tot}$  needs to be found. Given the load distribution, maximum drop specification, and the coupling between outputs, there may be outputs that require a small stage conductance to achieve a voltage drop below the spec while other outputs may require higher conductance to meet spec. The circuit must be sized such that all outputs are within spec and, therefore, the output that requires higher total conductance must be found.

Multiplying the transimpedance matrix elements by  $G_{st,tot}$  yields a matrix  $\zeta$  with elements given by

$$\zeta_{kl} = z_{kl} G_{st,tot} = \sum_{i \in stages} \frac{b_{st,ik} b_{st,il}}{h_i} \quad (3.2)$$

$\zeta_{st}$  describes how  $Z$  scales with  $G_{st,tot}$  for a given stage conductance distribution  $h$ . Expanding (2.18) and substituting (3.2):

$$\Delta V_{out,k} = \sum_{l \in outs} z_{kl} i_{out,l} = \frac{1}{G_{st,tot}} \sum_{l \in outs} \zeta_{kl} i_{out,l} \quad (3.3)$$

The total conductance required for each output to meet the maximum voltage drop specification with

full load on every output can be obtained from (3.4).

$$G_{st,req,k} = \sum_{l \in outs} \frac{\zeta_{kl} i_{out,max,l}}{\Delta V_{out,max,l}} \quad (3.4)$$

Since all outputs must meet the specification at the same time the higher  $G_{st,req,k}$  is taken as  $G_{st,tot}$ :

$$G_{st,tot} = \max_{k \in outs} \{G_{st,req,k}\} \quad (3.5)$$

To ensure that the maximum drop in a given output is obtained for the maximum loading on every output all elements of the matrix  $Z$  (or  $\zeta$ ) must be positive. This way, an increase in current in a given output always causes a voltage drop on other outputs, never a rise. Any conductance distributions that generate negative transimpedance elements are discarded.

### 3.4 Intra stage optimization

With the total conductance already calculated, the conductance of each stage can be calculated using (3.1) and each stage can be optimized by itself to achieve the required stage conductance, while optimizing area and power losses. The SSL, FSL and total output resistance of each stage (considering the charge multipliers are unitary) are given by equations (3.6) to (3.8).

$$Z_{st,SSL} = \frac{1}{f_{sw} C_1} \quad (3.6)$$

$$Z_{st,FSL} = \sum_{i=1}^4 \frac{1}{DG_i} \quad (3.7)$$

$$Z_{st,tot} = \sqrt{Z_{st,SSL}^2 + Z_{st,FSL}^2} \quad (3.8)$$

#### 3.4.1 Capacitor sizing to meet $Z_{st,SSL}$

Since each stage has only one capacitor, the capacitance required to achieve a given SSL output resistance is directly given by (3.9).

$$C_1 = \frac{1}{f_{sw} Z_{st,SSL}} \quad (3.9)$$

#### 3.4.2 Switch sizing to meet $Z_{st,FSL}$

These stages are often implemented with different MOSFETs for each switch, commonly 2 NMOS and 2 PMOS of the smallest voltage rating above the  $V_{\Delta}$  of the stage, but possibly even devices of different



voltage ratings are used due to startup considerations, for example. The relative sizing of this switches within each stage can then be optimized based on their differing performance metrics using the method on [9]. The optimization is performed to minimize a cost, either switch area or driving losses depending on which parameter is more important in the design power density or efficiency. The cost considered is the switch driving losses as it is the most relevant for fully integrated designs or semi-integrated designs aiming for maximum efficiency, yet the same method can easily be applied to other metrics. The total switch driving energy is obtained by dividing each switch conductance  $G_i$  by its loss metric  $m_{SL,i}$  and summing (3.10).

$$E_{tot} = \sum_{i=1}^4 \frac{G_i}{m_{SL,i}} \quad (3.10)$$

The result of the optimization yields the optimized switch conductance (3.11) and the resulting optimized FSL output resistance (3.12).

$$G_i = \sqrt{m_{SL,i}} \frac{E_{tot}}{\sum_{k \in sws} 1/\sqrt{m_{SL,k}}} \quad (3.11)$$

$$Z_{st,FSL} = \frac{1}{DE_{tot}} \left( \sum_{i=1}^4 \frac{1}{\sqrt{m_{SL,i}}} \right)^2 \quad (3.12)$$

Combining (3.11) and (3.12) yields each switch conductance related to the stage FSL output resistance (3.13).

$$G_i = \frac{\sqrt{m_{SL,i}}}{DZ_{st,FSL}} \left( \sum_{k=1}^4 \frac{1}{\sqrt{m_{SL,k}}} \right) \quad (3.13)$$

### 3.4.3 Optimization of $Z_{st,SSL}$ and $Z_{st,FSL}$ distribution

Both  $Z_{st,SSL}$  and  $Z_{st,FSL}$  contribute to the total output impedance  $Z_{st,tot}$  according to the relationship (3.8). The same  $Z_{st,tot}$  can then be obtained using different distributions of  $Z_{st,SSL}$  and  $Z_{st,FSL}$  which results in different capacitor and switch sizes according to equations (3.9) and (3.13), respectively. Combining (2.35) and (3.9) yields the stage's capacitor area and its relationship to  $Z_{st,SSL}$  given by (3.14), where  $K_{Acap}$  is given by (3.15).

$$A_{cap} = \frac{C_1}{m_{Ca,1}} = \frac{1}{Z_{st,SSL}} \frac{1}{f_{sw}m_{Ca,1}} = \frac{K_{Acap}}{Z_{st,SSL}} \quad (3.14)$$

$$K_{Acap} = \frac{1}{f_{sw}m_{Ca,1}} \quad (3.15)$$

Combining (2.41) and (3.9) and noting the stage's  $V_{\Delta}$  is equal to the the capacitor's terminal voltage

swing  $v_{cb,i}$  yields the stage's capacitor losses and its relationship to  $Z_{st,SSL}$  given by (3.16), where  $K_{P_{cpar}}$  is given by (3.17).

$$P_{cpar} = \frac{C_1 V_{\Delta}^2 f_{sw}}{m_{Cl,1}} = \frac{1}{Z_{st,SSL}} \frac{V_{\Delta}^2}{m_{Cl,1}} = \frac{K_{P_{cpar}}}{Z_{st,SSL}} \quad (3.16)$$

$$K_{P_{cpar}} = \frac{V_{\Delta}^2}{m_{Cl,1}} \quad (3.17)$$

Combining (2.37) and (3.13) and summing over the 4 switches yields the stage's total switch area and its relationship to  $Z_{st,FSL}$  given by (3.18), where  $K_{A_{sw}}$  is given by (3.19).

$$A_{sw} = \sum_{i=1}^4 \frac{G_i}{m_{Sa,i}} = \frac{1}{Z_{st,FSL}} \sum_{i=1}^4 \left[ \frac{\sqrt{m_{Sl,i}}}{D m_{Sa,i}} \left( \sum_{k=1}^4 \frac{1}{\sqrt{m_{Sl,k}}} \right) \right] = \frac{K_{A_{sw}}}{Z_{st,FSL}} \quad (3.18)$$

$$K_{A_{sw}} = \sum_{i=1}^4 \left[ \frac{\sqrt{m_{Sl,i}}}{D m_{Sa,i}} \left( \sum_{k=1}^4 \frac{1}{\sqrt{m_{Sl,k}}} \right) \right] \quad (3.19)$$

Combining (2.44) and (3.13) and summing over the 4 switches yields the stage's total stage's switch driving losses and its relationship to  $Z_{st,FSL}$  given by (3.20), where  $K_{P_{sdrv}}$  is given by (3.21).

$$P_{sdrv} = \sum_{i=1}^4 \frac{G_i f_{sw}}{m_{Sl,i}} = \frac{1}{Z_{st,FSL}} \sum_{i=1}^4 \left[ \frac{f_{sw}}{D \sqrt{m_{Sl,i}}} \left( \sum_{k=1}^4 \frac{1}{\sqrt{m_{Sl,k}}} \right) \right] = \frac{K_{P_{sdrv}}}{Z_{st,FSL}} \quad (3.20)$$

$$K_{P_{sdrv}} = \sum_{i=1}^4 \left[ \frac{f_{sw}}{D \sqrt{m_{Sl,i}}} \left( \sum_{k=1}^4 \frac{1}{\sqrt{m_{Sl,k}}} \right) \right] \quad (3.21)$$

Defining  $Z_{st,FSL}$  as  $Z_{st,SSL}$  multiplied by a constant  $r$  (3.22) allows obtaining  $Z_{st,tot}$  as (3.23). Using the definition of  $r$ , the relationship between  $Z_{st,SSL}$  and  $Z_{st,tot}$  is given by (3.24) while the relationship between  $Z_{st,FSL}$  and  $Z_{st,tot}$  is (3.25).

$$Z_{st,FSL} = r Z_{st,SSL} \quad (3.22)$$

$$Z_{st,tot} = \sqrt{1 + r^2} Z_{st,SSL} \quad (3.23)$$

$$Z_{st,SSL} = \frac{1}{\sqrt{1 + r^2}} Z_{st,tot} \quad (3.24)$$

$$Z_{st,FSL} = \frac{r}{\sqrt{1 + r^2}} Z_{st,tot} \quad (3.25)$$

The total stage area  $A_{st}$  can now be obtained by summing the capacitor and switch areas while the total stage power losses  $P_{st}$  is obtained by summing the capacitor parasitics and switch driving

losses. By using the relationships (3.24) and (3.25) the dependencies of  $A_{st}$  and  $P_{st}$  on  $Z_{st,tot}$  and  $r$  are obtained as (3.26) and (3.27), respectively.

$$A_{st} = \frac{K_{Acap}}{Z_{st,SSL}} + \frac{K_{Asw}}{Z_{st,FSL}} = \frac{1}{Z_{st,tot}} \left( \sqrt{1+r^2} K_{Acap} + \frac{\sqrt{1+r^2}}{r} K_{Asw} \right) \quad (3.26)$$

$$P_{st} = \frac{K_{Pcpar}}{Z_{st,SSL}} + \frac{K_{Psdrv}}{Z_{st,FSL}} = \frac{1}{Z_{st,tot}} \left( \sqrt{1+r^2} K_{Pcpar} + \frac{\sqrt{1+r^2}}{r} K_{Psdrv} \right) \quad (3.27)$$

Since there may be a trade-off between the stage's occupied area and losses it cannot be simultaneously optimized for both quantities. A cost function  $f$  (3.28) and a design parameter  $\lambda$  are introduced to account for the trade-off.  $f$  includes both the area and losses cost with the losses portion affected of parameter  $\lambda$  which denotes the relative importance of the losses in relation to the occupied area.  $\lambda$  choice is in charge of the designer which should tune it to match the application characteristics.

$$f = A_{st} + \lambda P_{st} \quad (3.28)$$

The optimal ratio between  $Z_{st,SSL}$  and  $Z_{st,FSL}$  is found by minimizing  $f$ . The derivative of  $f$  is set to zero (3.29) and solved for  $r$  yielding the optimized  $r$  (3.30).

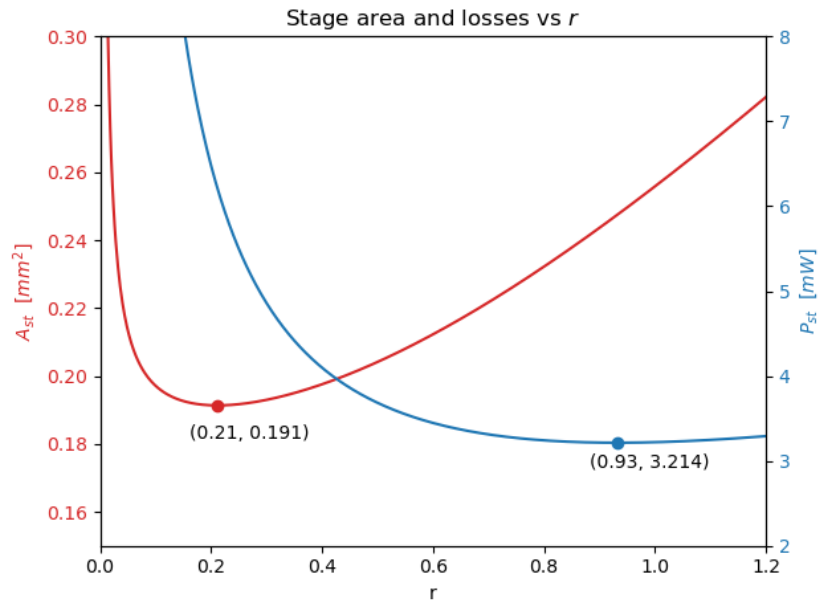
$$\frac{\partial f}{\partial r} = \frac{1}{Z_{st,tot}} \left[ \frac{r}{\sqrt{1+r^2}} (K_{Acap} + \lambda K_{Pcpar}) - \frac{1}{r^2 \sqrt{1+r^2}} (K_{Asw} + \lambda K_{Psdrv}) \right] = 0 \quad (3.29)$$

$$r_{opt} = \sqrt[3]{\frac{K_{Asw} + \lambda K_{Psdrv}}{K_{Acap} + \lambda K_{Pcpar}}} \quad (3.30)$$

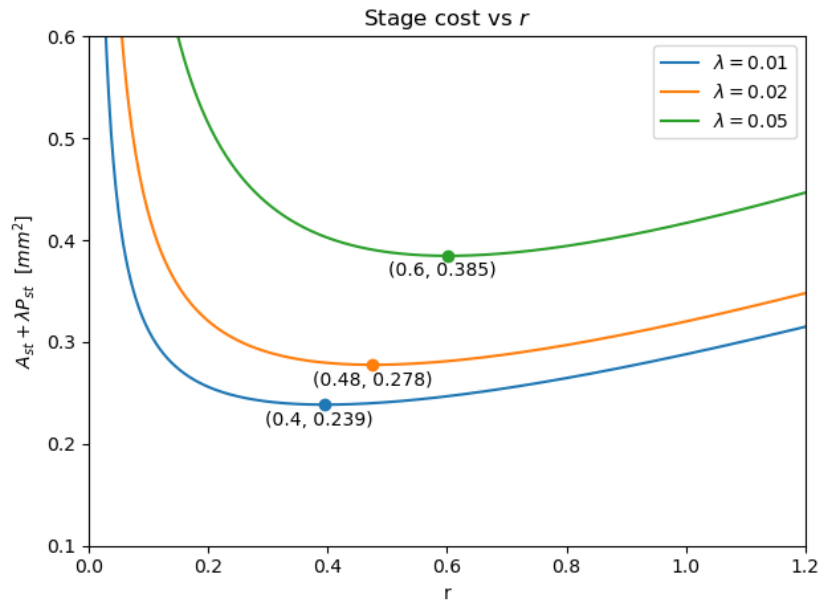
$r_{opt}$  will be smaller than 1 if the design is mostly capacitor constrained as is usually the case of fully integrated designs and it will be higher than 1 for designs switch constrained as is usually the case for designs using external capacitors.

Figure 3.1 shows how the stage area  $A_{st}$  and losses  $P_{st}$  vary with the ratio  $r$  for an example stage. If the stage was optimized only for minimum area the optimal  $r$  would be 0.21 while if it was optimized for minimum losses the optimal  $r$  would 0.93. Figure 3.2 shows the plot of the cost function (3.28) with respect to  $r$  for three different  $\lambda$ . The optimal  $r$  is always between the  $r$  for minimum area and the  $r$  for minimum losses with the parameter  $\lambda$  governing the tradeoff between the two. When  $\lambda$  is small  $r_{opt}$  is close to the optimal  $r$  for minimum area, while for higher  $\lambda$ , meaning a bigger importance on losses,  $r_{opt}$  gets closer to the optimal  $r$  for minimum losses.

With  $r$  calculated the stage component values can be calculated. To calculate the stage capacitance,  $Z_{st,SSL}$  needs to be found using (3.24) and then  $C_1$  is obtained with (3.9). To calculate the conductance of each of the stage's switches,  $Z_{st,FSL}$  is found using (3.25) and then  $G_i$  is obtained for each switch



**Figure 3.1:** Variation of the stage area and losses with the ratio  $r$ . Minimum points are marked.

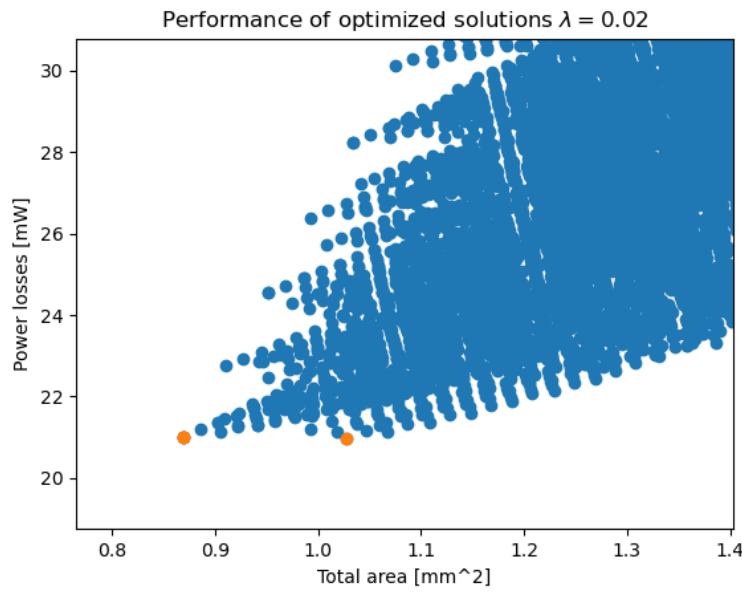


**Figure 3.2:** Variation of the stage total cost with the ratio  $r$  for different  $\lambda$ . Minimum points are marked.

with (3.13).

### 3.5 Choosing the final sizing solution

Each stage conductance distribution will generate a solution with optimized stages. Figure 3.3 shows the total area and power loss of an example topology with 5 stages with optimally sized stages for each stage conductance distribution generated.  $\lambda$  is set to 0.02 and a sizing resolution  $res$  of 10 was used (total of  $10^5 = 100000$  solutions calculated).



**Figure 3.3:** Performance of optimally sized topology for a set of stage conductance distributions.  $\lambda = 0.02 \text{ mm}^2/\text{mW}$ .

To find the overall optimal solution, a cost function based on the same parameter  $\lambda$  as the stage optimization cost function (3.28) is used. This cost function is given by (3.31) where  $A_{tot}$  is the sum of each stage area and  $P_{tot}$  is the sum of each stage power losses. If  $A_{tot}$  is expressed in  $[\text{m}^2]$  and  $P_{tot}$  in  $[\text{W}]$  then  $\lambda$  has units of  $[\text{m}^2/\text{W}]$  such that it expresses how much extra area is occupied to reduce a given amount of losses.

$$F = A_{tot} + \lambda P_{tot} \quad (3.31)$$

The solution with the minimum cost is the optimal sizing for the topology (and correspondent device technology assignment), switching frequency  $f_{sw}$ ,  $\lambda$  and application specs considered. The topology, switching frequency and  $\lambda$  can be varied by the designer to find the most advantageous solution for the application.

## 3.6 Limitations

A number of limitations of this method are listed below.

- The method only sizes the flying capacitors and switches. It does not take into account how different switching frequencies and different topologies require different sized output capacitors on each output to achieve the specified ripple and the how much the output capacitors contribute to the total solution size.
- The method assumes there is a maximum output current and drop specification as if the SCC is being used mainly as a transformer. Regulation is not taken into account in any way.
- The method assumes the wanted topology can be represented using the stages described in section 2.7. Although any two phase topology can be represented using this stages, for some topologies like the ladder it generates redundant switches that would reduce the circuit performance significantly. It should be possible to adapt the method to this cases.
- The method assumes output voltage drop does not influence gate drive voltage and capacitor parasitics voltage swing which is a good approximation for small specified output voltage drops but might have significant error for bigger ones.
- The optimal conductance search is performed what is essentially brute force such that the algorithm run time presents exponential growth with problem size (number of stages in the topology).
- The method completely disregards any practical aspects of the implementation of circuit implementation like startup, level shifting and gate driving circuits, etc. The method cannot be directly used to choose a topology, only to guide the choice by analysing the theoretical performance of each considered solution.

# 4

## M4M application

### Contents

---

4.1	Specifications . . . . .	30
4.2	Technology device implementation and characterization . . . . .	30
4.3	Topology choice . . . . .	33

---

In this chapter the previously presented method is used to aid in the choice and sizing of the topology to be used in the M4M UHF stimulation implant.

## 4.1 Specifications

To support the multiple simultaneous stimulation channels with both high and low voltage rails, voltages both bellow and above the input voltage need to be generated with high efficiency. From a regulated input voltage of 4.5 V a total of 6 supply rails, including the input voltage, are to be generated in steps of  $(1/3)V_{in}$ . Each output  $V_{ox}$  provides a nominal conversion ratio of  $(x/3), x \in \{1, \dots, 6\}$  and should be sized to supply 4 mA at a maximum drop of 5% from the nominal conversion ratio. The specifications where defined in chapter 1 and they are reproduced in Table 4.1 for the comfort of the reader.

**Table 4.1:** Converter specifications for ultrasonic implant stimulation application.

Spec	Value
$V_{in}$	min: 4.5 V, typ: 4.6 V, max: 4.7 V
Ideal $V_{out}$ ratios	1/3, 2/3, 4/3, 5/3, 6/3
Ideal $V_{out}$ @ $V_{in} = 4.5$ V	1.5 V, 3.0 V, 6.0 V, 7.5 V, 9.0 V
$I_{out,max}$	4 mA per output
$\Delta V_{out,max}$	5% of $V_{out}$ @ 4.5 V
$V_{out,PP,max}$	2% of $V_{out}$ @ 4.5 V
Fabrication Technology	TSMC 0.18um BCD GenII
Integration	Fully integrated (no external capacitors)

## 4.2 Technology device implementation and characterization

The converter is to be implemented in TSMC 180nm BCD GenII process with the following options:

- 1.8V core devices, 5V IO devices, 6V to 70V high voltage devices (including laterally diffused MOS)
- 6 Metal layers with 30kA Ultra Thick Top Metal
- 2fF MIM capacitors between the two top metal layers

The process makes use of four LV wells, two HV wells and an N+ Buried Layer (NBL) for device construction and insulation from the substrate. No Deep NWell is available. To characterize the available technology devices representative units where drawn in schematic and layout, their parasitics where extracted and were simulated using the models provided by the foundry.



### 4.2.1 Capacitors

Three types of capacitors are possible of being implemented in this technology: Metal Oxide Semiconductor (MOS), Metal Oxide Metal (MOM) and Metal Insulator Metal (MIM). Their performance metrics for each of the considered capacitance structures are presented in table 4.2. The implementation of each of the capacitor types is described on the following sections.

**Table 4.2:** Performance metrics of the capacitance structures considered.

Structure	$m_{SA}$ [nF/mm <sup>2</sup> ]	$\alpha$	$\beta$	$m_{SL}$
NMOS 1V8	8.9	2%	0	50
NMOS 5V	4.1	4%	0	25
PMOS 1V8	8.9	0	1%	100
PMOS 5V	4.1	0	2%	50
MOM M2-M5 70V	0.8	2.3%	2.3%	22
MIM 5V	1.55	0.7%	0	143

#### 4.2.1.A MOM capacitors

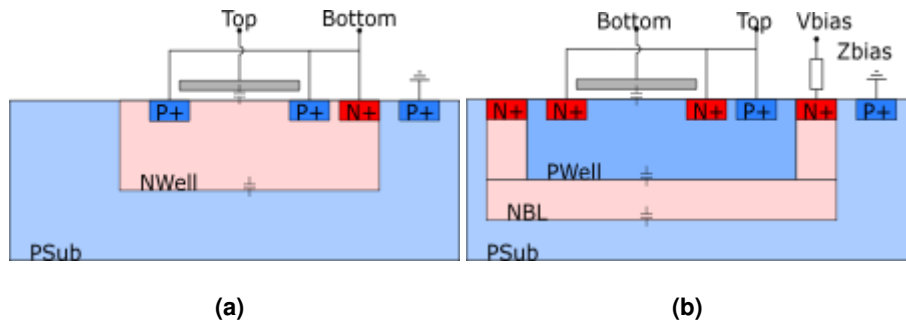
MOM capacitors are implemented simply by using capacitance created between metal lines separated by the regular layer insulation oxide. The main contribution the capacitance is the lateral capacitance between metal fingers in the same metal layer. The vertical capacitance between fingers placed in adjacent metal layers also provides contribution to the total capacitance but in a small portion due to higher distances between metal layers and smaller parallel area. The capacitance density of MOM capacitors is constrained by the minimum spacing between metal lines. Being this an old technology the spacings between metals are big and thus this is the capacitor type with the smallest capacitance density. This type of capacitors also present high parasitic capacitance in both plates due to the proximity of the lower metal layers to the substrate. MOM capacitors are rated for 70 V so they are the only devices available to implement capacitors with more than 5 V rating.

#### 4.2.1.B MIM capacitors

MIM capacitors are implemented in the top of the metal stack using two specially closely spaced metal plates separated by a special dielectric. This arrangement provides more capacitance density than MOM capacitors while presenting the lowest possible parasitic capacitance due to the increased distance from the bottom plate to the substrate. MIM capacitors are only rated for 5 V.

#### 4.2.1.C MOS capacitors

MOS capacitors are implemented using the gate capacitance of the available MOSFETs which achieves higher capacitance densities than other types of capacitors. 1.8 V devices provide considerably higher capacitance density than 5 V devices due to the thinner gate oxide but can only be used to implement capacitors with an applied voltage up to 1.8 V while the 5 V devices are rated for a gate voltage of 5 V due to the usage of a thicker gate oxide. MOS capacitors also present high parasitic capacitance due to their proximity to the substrate. MOS capacitors can be implemented using different structures and biasing techniques aimed at reducing the parasitic capacitance to the substrate, the two structures considered in this work are pictured in figure 4.1(a) and figure 4.1(b)



**Figure 4.1:** (a) NMOS accumulation structure (b) PMOS accumulation structure with high impedance bias.

The structure in 4.1(a) is a NMOS accumulation structure implemented using a PMOS with source, drain and bulk (NWell) terminals connected to the bottom terminal (such that the simulation model for the PMOS can be used). This configuration results in a structure similar to [6]. This structure is compact while still providing insulation from the substrate but presents quite high parasitic capacitance on the bottom terminal. It is best suited to implement DC capacitors. The structure in 4.1(b) is a PMOS accumulation structure implemented using a NMOSFET with source, drain and bulk (PWell) terminals connected to the top terminal (such that the simulation model for the NMOS can be used). This configuration results in a structure similar to [15]. The high impedance bias is also realized like suggested by [15] by the usage of a PMOS and a NMOS configured as back to back diodes connected to a voltage of  $2V_{in}$ .

Since the MOS capacitor implementations only occupy layers up to the first metal layer all remaining metal layers can be used to implement MOM and MIM structures in the same area. The stacking of the 3 technologies allows the capacitance density to be increased with no penalty on the parasitics since the coupling to the substrate has the same area. This composite structure allows for about 30% and 90% increase on capacitance the density of 1.8 V and 5 V MOS structures, respectively.

## 4.2.2 Switches

This technology provides MOSFETs up to 70 V voltage rating. However, considering that the input voltage is 4.5 V, and the maximum output voltage is 9 V, only devices with ratings of 1.8 V (core), 5 V (IO) and 12 V (LDMOS) are considered. Core devices use thin gate oxide with a gate-source voltage rating of 1.8 V while IO and LDMOS devices use thick gate oxide with a gate-source voltage rating of 5 V. Both NMOS and PMOS are available for each voltage rating and means of insulating transistor bulks from the substrate with enough withstand voltage are provided by the usage of a N+ buried layer. With this insulation layer, all MOSFETs can be implemented in any topology without suffering body effect even if their source is connected to a potential higher than their voltage rating. The switch performance metrics are presented in table 4.3.

**Table 4.3:** Performance metrics of MOSFET available to be used as switches.

Switch	$m_{SA}$ [mS/ $\mu\text{m}^2$ ]	$m_{SL}$ [mS/pJ]
NMOS 1V8	0.97	440
PMOS 1V8	0.36	120
NMOS 5V	0.29	17
PMOS 5V	0.11	5.5
NMOS 12V	0.11	7.4
PMOS 12V	0.023	1.8

## 4.3 Topology choice

There are several approaches to generate the five required voltage rails. The traditional method would be to design five independent SCC and optimize each one using the methods proposed by [7]. This approach requires a minimum of 11 capacitors [4] and more than 30 switches. This creates a complex circuit, requiring more auxiliary circuits and creating layout area overheads.

Since the application requires linearly spaced voltage ratios a simple approach would be a linear topology with each stage generating a step of  $1/3V_{in}$  similar to [13] and [16]. This topology's (TOP1) stage arrangement is shown in Figure 4.2. The minimum capacitor and switch voltage rating and device technology used for each stage are presented in Table 4.4. Each stage is assumed to be implemented with two NMOS and two PMOS for switches. TOP1 stages ST4 and ST5 require capacitors with a voltage rating of at least 6.3 V and 7.9, respectively. For this voltage ratings only MOM capacitors can be used which present very low capacitance density and very high parasitic capacitance.

Inputting this topology's characteristics and the application specifications in the algorithm proposed in chapter 3 and considering a switching frequency of 32 MHz and  $\lambda = 0.02$  yielded the optimal stage conductance,  $r$  and area for each of the topology's stages. This values appear in Table 4.5. Although

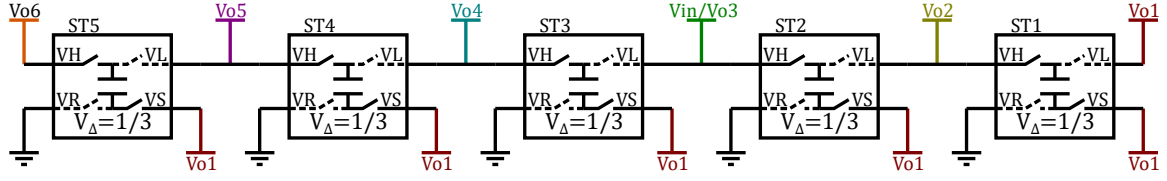


Figure 4.2: Stage connections for TOP1.

TOP1	$V_{\Delta}$	$V_{cap}$	cap tech	$V_{sw}$	sw tech
ST1	$1/3V_{in}$	1.6 V	MOS 1V8	1.6 V	2x NMOS 1V8 + 2x PMOS 1V8
ST2	$1/3V_{in}$	3.2 V	MOS 5V	1.6 V	2x NMOS 1V8 + 2x PMOS 1V8
ST3	$1/3V_{in}$	4.7 V	MOS 5V	1.6 V	2x NMOS 1V8 + 2x PMOS 1V8
ST4	$1/3V_{in}$	6.3 V	MOM 70V	1.6 V	2x NMOS 1V8 + 2x PMOS 1V8
ST5	$1/3V_{in}$	7.9 V	MOM 70V	1.6 V	2x NMOS 1V8 + 2x PMOS 1V8

Table 4.4: Voltage step and minimum capacitor and switch voltage rating for each stage of topology TOP1.

the two upper stages are attributed the lowest conductances they occupy the biggest area derived from the usage of MOM capacitors that present low capacitance density. The  $r$  values are also smaller for those two stages since the optimization process tried to minimize the capacitor size by increasing the switch size to achieve the required conductance.

Stage	$h_i$	$G_{st,i}$ [mS]	$r_i$	$A_{st,i}$ [mm <sup>2</sup> ]
ST1	0.29	164	0.34	0.64
ST2	0.29	164	0.26	1.32
ST3	0.25	143	0.26	1.16
ST4	0.11	61	0.16	2.46
ST5	0.07	41	0.16	1.66

Table 4.5: Optimal stage conductance distribution,  $r$  and area for each of the stages of TOP1 for  $f_{sw} = 32$  MHz and  $\lambda = 0.02$ .

With optimal sizing, this topology achieves a power density of  $14.4 \text{ mW/mm}^2$  and an efficiency of 70.2% which is a very poor performance. The performance, capacitor and switch area, and losses are discriminated in Table 4.11 where it can be compared with the other topologies proposed bellow.

The two biggest factors hindering the performance of this topology are: the, already identified, use of MOM capacitors and the inefficiency in the generation of  $V_{o6}$  that can be observed in the topology's charge multipliers in (4.1). All elements on the last column of the matrix (4.1) are non-zero, meaning all capacitors are involved in the charge transfer from the input to the fifth output  $V_{o6}$  resulting in high required conductances, and consequently high area and losses, to meet the specification.

$$B_{st, TOP1} = \begin{bmatrix} -\frac{1}{3} & \frac{1}{3} & -\frac{1}{3} & -1 & 1 \\ \frac{1}{3} & \frac{2}{3} & \frac{1}{3} & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.1)$$

To address the problems of TOP1, a number of other topologies are proposed. TOP2 stage arrangement is shown in Figure 4.3 and the required stage voltage ratings and technology attribution is shown in Table 4.6. In this topology ST1, ST2 and ST3 are connected exactly the same way as TOP1 to generate  $V_{o1}$ ,  $V_{o2}$  and  $V_{o4}$ .  $V_{o5}$  is generated with a step of  $2/3V_{in}$  above  $V_{in}$  and  $V_{o6}$  is generated with a step of  $3/3V_{in}$  above  $V_{in}$ . In this configuration no stage requires MOM capacitors with ST4 using a 5V MOS capacitor and ST5 using a MIM capacitor. A MIM capacitor is used in ST5 because this stage has a bigger step equal to  $3/3V_{in}$ , causing significantly higher losses in the parasitic capacitances as studied in section 2.6.1.  $V_{o6}$  is now much more efficiently generated as it is generated independently using ST5. The resulting charge multipliers, presented in (4.2), are in general lower and show how both  $V_{o5}$  and  $V_{o6}$  are generated with charge transfer through a lower number of stages (more zeros in the fourth and fifth columns).

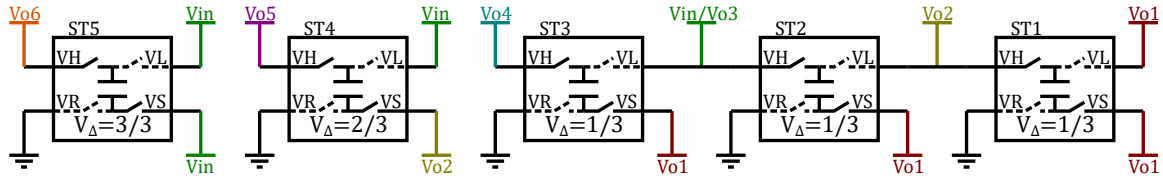


Figure 4.3: Stage connections for TOP2.

TOP2	$V_{\Delta}$	$V_{cap}$	cap tech	$V_{sw}$	sw tech
ST1	$1/3V_{in}$	1.6 V	MOS 1V8	1.6 V	2x NMOS 1V8 + 2x PMOS 1V8
ST2	$1/3V_{in}$	3.2 V	MOS 5V	1.6 V	2x NMOS 1V8 + 2x PMOS 1V8
ST3	$1/3V_{in}$	4.7 V	MOS 5V	1.6 V	2x NMOS 1V8 + 2x PMOS 1V8
ST4	$2/3V_{in}$	4.7 V	MOS 5V	3.2 V	2x NMOS 5V + 2x PMOS 5V
ST5	$3/3V_{in}$	4.7 V	MIM 5V	4.7 V	2x NMOS 5V + 2x PMOS 5V

Table 4.6: Voltage step and minimum capacitor and switch voltage rating for each stage of topology TOP2.

$$B_{st,TOP2} = \begin{bmatrix} -\frac{1}{3} & \frac{1}{3} & -\frac{1}{3} & \frac{1}{3} & 0 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.2)$$

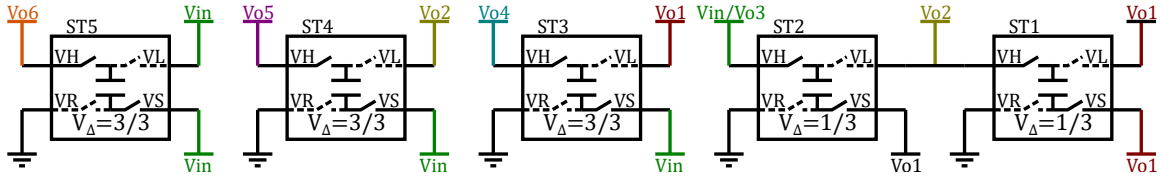
Passing this topology through the sizing algorithm yields much smaller stages as can be observed in Table 4.7. All stages are significantly smaller due both to smaller required conductances (better charge multipliers), and the usage of capacitors with better capacity density. This results in a power density of 118 mW/mm<sup>2</sup> and an efficiency of 83%, a performance much superior to TOP1.

TOP3 is another alternative topology whose stage arrangement is shown in Figure 4.4 and the required stage voltage ratings and technology attribution is shown in Table 4.8.  $V_{o1}$ ,  $V_{o2}$  and  $V_{o6}$  are generated exactly the same as TOP2, the difference between TOP2 and TOP3 is stages ST3 and ST4

Stage	$h_i$	$G_{st,i}$ [mS]	$r_i$	$A_{st,i}$ [mm <sup>2</sup> ]
ST1	0.18	18	0.34	0.07
ST2	0.36	36	0.26	0.28
ST3	0.18	18	0.26	0.15
ST4	0.18	18	0.6	0.16
ST5	0.09	9	0.48	0.21

**Table 4.7:** Optimal stage conductance distribution,  $r$  and area for each of the stages of TOP2 for  $f_{sw} = 32$  MHz and  $\lambda = 0.02$ .

both using a step of  $V_{in}$  to generate  $V_{o4}$  and  $V_{o5}$  respectively. TOP3 has the same charge multipliers as TOP2 and the only differences in terms of performance is that stages ST4 and ST5 require lower capacitor voltage ratings but require higher switch voltage ratings and present higher voltage swings on the capacitors, that may increase total loss.



**Figure 4.4:** Stage connections for TOP3.

TOP3	$V_{\Delta}$	$V_{cap}$	cap tech	$V_{sw}$	sw tech
ST1	$1/3V_{in}$	1.6 V	MOS 1V8	1.6 V	2x NMOS 1V8 + 2x PMOS 1V8
ST2	$1/3V_{in}$	3.2 V	MOS 5V	1.6 V	2x NMOS 1V8 + 2x PMOS 1V8
ST3	$3/3V_{in}$	1.6 V	MOS 1V8	4.7 V	2x NMOS 5V + 2x PMOS 5V
ST4	$3/3V_{in}$	3.2 V	MIM 5V	4.7 V	2x NMOS 5V + 2x PMOS 5V
ST5	$3/3V_{in}$	4.7 V	MIM 5V	4.7 V	2x NMOS 5V + 2x PMOS 5V

**Table 4.8:** Voltage step and minimum capacitor and switch voltage rating for each stage of topology TOP3.

Another two topologies are presented, TOP4 and TOP5, that consist of mixing stages ST3 and ST4 from TOP2 and TOP3. The stage arrangements are shown in Figure 4.5 and Figure 4.6. The charge multipliers are equal in all four topologies TOP2, TOP3, TOP4 and TOP5, so their performance differences are only due to different combinations of component ratings and voltage swings in stages ST3 and ST4.

A different approach from the other topologies is taken in TOP6. Instead of using step-up stages to generate  $V_{o4}$  and  $V_{o5}$  from a combination of  $V_{in}$  and the lower voltages,  $V_{o4}$  and  $V_{o5}$  are generated by stepping-down  $V_{o6}$  and using  $V_{in}$  as the reference voltage for ST3 and ST4.  $V_{o6}$  is generated using stage ST5 with the same connections as the previous topologies. The stage arrangement for TOP6 is shown in Figure 4.7 while the required stage voltage ratings and technology attribution is shown in Table 4.9. In this topology, since  $V_{o4}$  and  $V_{o5}$  are not generated using  $V_{o1}$  and  $V_{o2}$ , the step-up outputs

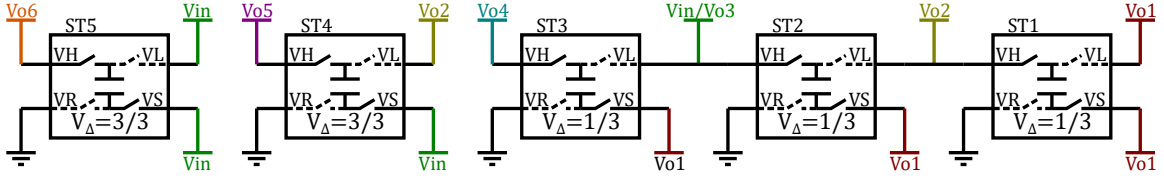


Figure 4.5: Stage connections for TOP4.

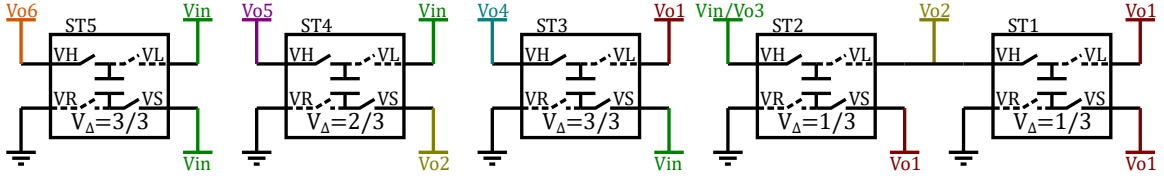


Figure 4.6: Stage connections for TOP5.

are independent from the step-down outputs as can be seen from the charge multipliers in (4.3). The result is that passing this topology through the sizing algorithm yields smaller stages ST1 and ST2 but a bigger ST5 as can be observed in Table 4.10. This topology has the advantage that all stages except ST5 use a step of  $1/3V_{in}$  reducing the capacitor parasitics loss while requiring the same device voltage ratings.

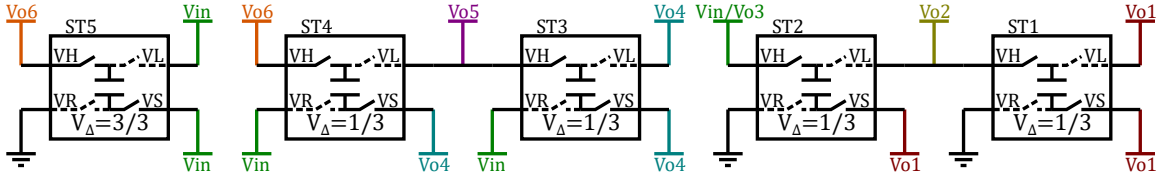


Figure 4.7: Stage connections for TOP6.

TOP6	$V_{\Delta}$	$V_{cap}$	cap tech	$V_{sw}$	sw tech
ST1	$1/3V_{in}$	1.6 V	MOS 1V8	1.6 V	2x NMOS 1V8 + 2x PMOS 1V8
ST2	$1/3V_{in}$	3.2 V	MOS 5V	1.6 V	2x NMOS 1V8 + 2x PMOS 1V8
ST3	$1/3V_{in}$	1.6 V	MOS 1V8	1.6 V	2x NMOS 1V8 + 2x PMOS 1V8
ST4	$1/3V_{in}$	3.2 V	MOM 5V	1.6 V	2x NMOS 1V8 + 2x PMOS 1V8
ST5	$3/3V_{in}$	4.7 V	MIM 5V	4.7 V	2x NMOS 5V + 2x PMOS 5V

Table 4.9: Voltage step and minimum capacitor and switch voltage rating for each stage of topology TOP6.

$$B_{st, TOP6} = \begin{bmatrix} -\frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & \frac{2}{3} & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{3} & \frac{1}{3} & 0 \\ 0 & 0 & \frac{1}{3} & \frac{2}{3} & 0 \\ 0 & 0 & \frac{1}{3} & \frac{1}{3} & 1 \end{bmatrix} \quad (4.3)$$

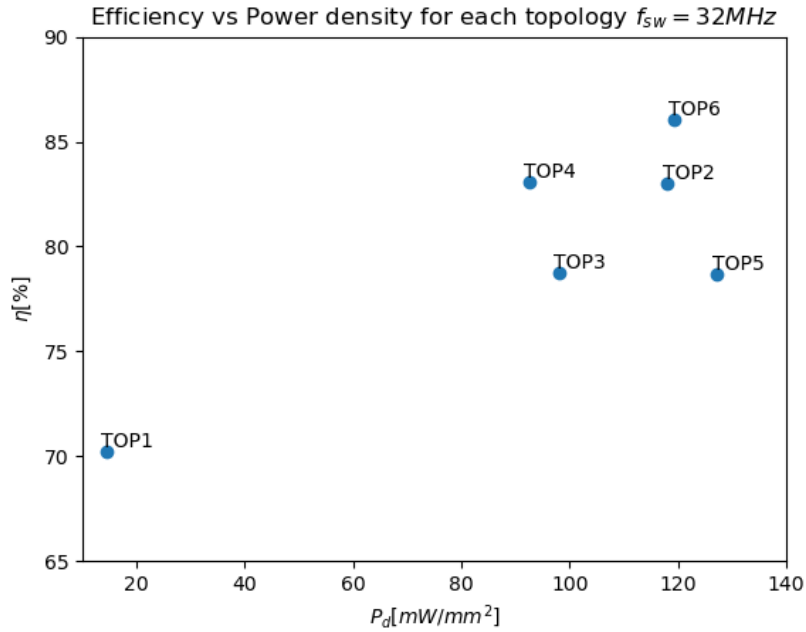
The sizing of all six topologies was optimized for a switching frequency of 32 MHz and  $\lambda = 0.02$ . The resulting cost and performance is summarized in Table 4.11. A plot of efficiency versus power density of the 6 topologies is shown in figure 4.8.

Stage	$h_i$	$G_{st,i}$ [mS]	$r_i$	$A_{st,i}$ [mm <sup>2</sup> ]
ST1	0.12	9	0.34	0.03
ST2	0.23	17	0.26	0.14
ST3	0.08	6	0.34	0.02
ST4	0.31	23	0.26	0.19
ST5	0.27	21	0.48	0.21

**Table 4.10:** Optimal stage conductance distribution,  $r$  and area for each of the stages of TOP6 for  $f_{sw} = 32$  MHz and  $\lambda = 0.02$ .

	TOP1	TOP2	TOP3	TOP4	TOP5	TOP6
$A_{cap}$ [mm <sup>2</sup> ]	7.10	0.84	1.01	1.08	0.78	0.85
$A_{sw}$ [mm <sup>2</sup> ]	0.14	0.02	0.02	0.02	0.02	0.02
$A_{tot}$ [mm <sup>2</sup> ]	7.21	0.87	1.05	1.11	0.81	0.87
$P_{cpar}$ [mW]	28.3	8.0	10.6	7.1	11.6	5.5
$P_{sdrv}$ [mW]	11.7	7.6	11.7	8.4	10.9	6.6
$P_{rout}$ [mW]	4.0	5.4	5.4	5.4	5.4	4.6
$P_{loss}$ [mW]	44.1	21.0	27.7	20.9	27.8	16.7
$P_{out}$ [mW]	104.0	102.6	102.6	102.6	102.6	103.3
$P_d$ [mW/mm <sup>2</sup> ]	14	118	98	92	127	119
$\eta$ [%]	70	83	79	83	79	86

**Table 4.11:** Cost and performance of each of the considered topologies with optimized sizing.



**Figure 4.8:** Efficiency vs Power density for topologies 1 to 6.  $f_{sw} = 32$  MHz,  $\lambda = 0.02$



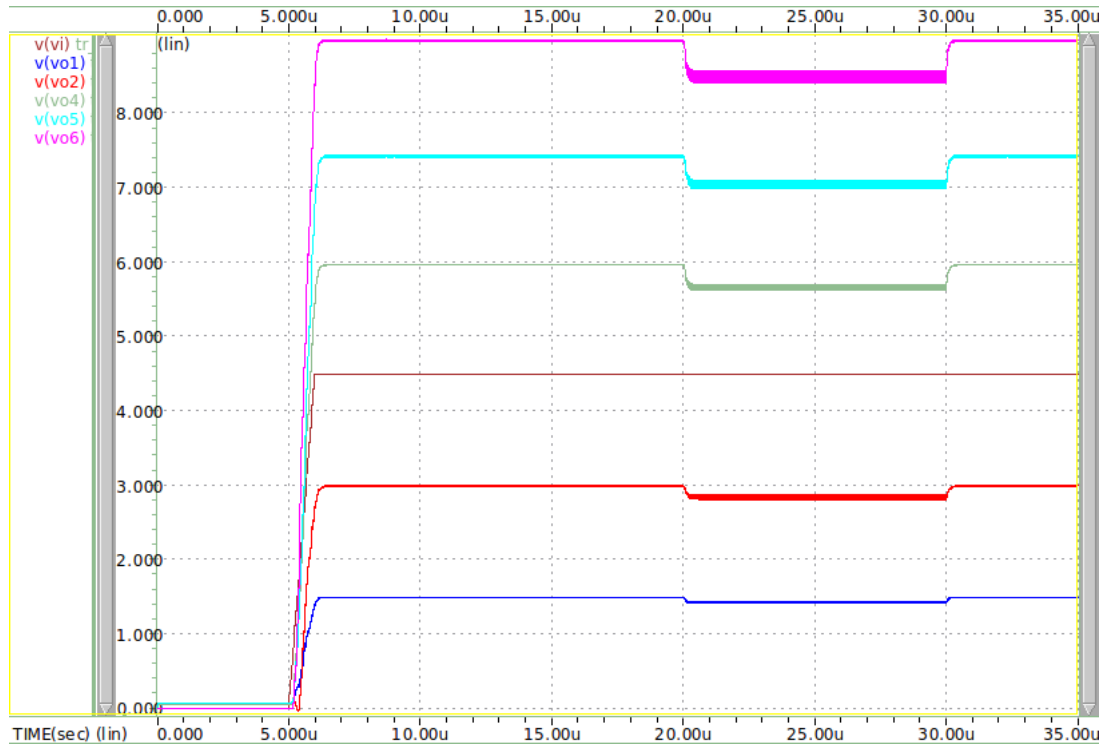
Topology TOP1 is inferior to the other solutions by a big margin. Topology TOP5 is the most power dense topology but presents significantly lower efficiency than TOP2, TOP4 and TOP6. Only considering power density and efficiency TOP6 seems to be the most advantageous topology. Nevertheless TOP6 presents extra implementation challenges related to startup and switch driving due to the reference of stages ST3 and ST4 being  $V_{in}$  instead of ground. This challenges considerably increase the design effort of topology TOP6. Another aspect that can be considered is the possibility of independent regulation of outputs. In TOP2 the outputs  $V_{o4}$ ,  $V_{o5}$  and  $V_{o6}$  can be independently regulated and/or shut down, because stages ST3, ST4 and ST5 transfer charge to a single output, such that clock gating those stages allows independent regulation of each of the step-up outputs. This is advantageous in this application because it allows some stages and all of its auxiliary circuits to be turned off when the upper outputs are not required, reducing the static consumption of those circuits. In TOP6 the three upper outputs are coupled such that loading outputs  $V_{o4}$  or  $V_{o5}$  implies that all three stages ST3, ST4 and ST5 must be clocked to maintain the output voltage in the loaded output even though only one of the top outputs is being used. Considering the extra design complexity of topology TOP6 and the small performance gains it provides in relation to TOP2, the latter was chosen to be implemented for the M4M project.



# 5

## Results

The chosen topology (TOP2) was designed in Cadence® Virtuoso® schematic and simulated using the HSPICE® simulator with MOSFETs and capacitor models from TSMC 0.18  $\mu\text{m}$  BCD GenII technology. Ideal gate drivers were used to drive the MOSFETs as switches, generating the required voltages correctly referenced to the source of each MOSFET while drawing the energy required to drive the gate capacitance from the input source. Using multiple small switch and capacitor units, each component was implemented with a size as close as possible to the sizing algorithm result. The schematics of the implemented topology are shown in appendix B. The input voltage and switching frequency are the same passed to the sizing algorithm, 4.5 V and 32 MHz, respectively. Since the sizing algorithm does not take into account ripple and output capacitors, those were sized such that the simulated ripple was below the specification. The output capacitors occupy a total area of 2.06  $\text{mm}^2$ . The output voltages of the simulated topology are shown in Figure 5.1.



**Figure 5.1:** Output voltages of the simulated SCC. The outputs are loaded with 4 mA each at 20  $\mu\text{s}$  and unloaded at 30  $\mu\text{s}$ .

The specification, theoretical (obtained through the sizing algorithm) and simulated values for each output, as well as the error between the theoretical and simulated values, are presented in Table 5.1. The average output voltages  $V_{out,k}$  are measured with all outputs fully loaded with 4 mA. The gain of each output  $A_k$  is calculated by dividing the unloaded output voltage by the input voltage. The simulated output voltage values have less than 1% error to the theoretical values. Similar deviations are observed in the gain of each output. This deviations can be explained by the effects of capacitor parasitics in

the converter voltage gain and output impedance (or transimpedance) [6] that were not considered in the proposed sizing method. The power loss, and therefore the efficiency, present much larger errors, with the simulated loss being 16.1% higher than expected, with resulting 2.8% lower efficiency. One possible cause for the extra losses include MOSFET capacitances that were not taken into account and create extra loss when the capacitor terminals are switched. Another cause is the fact that the MOS capacitors' parasitics to the substrate are formed by diodes which present a voltage dependent junction capacitance. The parasitics were measured with a constant bias voltage in relation to the capacitor bulk while in the circuit this bias is different for different stages and at each phase causing some deviation in the effective  $\alpha$  and  $\beta$  of the capacitor.

Quantity	Spec Value	Theo Value	Sim Value	Unit	Error
$V_{out1}$	1.425	1.425	1.422	V	-0.2%
$A_1$	0.333	0.333	0.332	1	-0.3%
$V_{pp1}$	30	-	27	mV	-
$V_{out2}$	2.850	2.850	2.835	V	-0.5%
$A_2$	0.667	0.667	0.664	1	-0.4%
$V_{pp2}$	60	-	53	mV	-
$V_{out4}$	5.700	5.700	5.667	V	-0.6%
$A_4$	1.333	1.333	1.327	1	-0.5%
$V_{pp4}$	120	-	93	mV	-
$V_{out5}$	7.125	7.129	7.056	V	-1.0%
$A_5$	1.666	1.666	1.653	1	-0.8%
$V_{pp5}$	150	-	118	mV	-
$V_{out6}$	8.550	8.550	8.523	V	-0.3%
$A_6$	2.000	2.000	1.997	1	-0.1%
$V_{pp6}$	180	-	163	mV	-
$P_{out}$	-	102.6	102.0	mW	-0.6%
$P_{loss}$	-	21.0	24.4	mW	16.1%
$\eta$	-	83.0	80.7	%	-2.8%

**Table 5.1:** Specified, theoretical and simulated values for TOP2.

The final characteristics of the designed solution, including the output capacitors, are presented in Table 5.2.

	Value
Number of outputs	5
Voltage ratios	1/3, 2/3, 4/3, 5/3, 2
Integration	Fully integrated
Clock frequency	32 MHz
Silicon area	2.93 mm <sup>2</sup>
Technology	TSMC 0.18 $\mu$ m BCD GenII
Total output power	102 mW
Power density	34.8 mW/mm <sup>2</sup>
Efficiency	80.7 %

**Table 5.2:** M4M solution characteristics.

# 6

## Conclusions

### Contents

---

6.1 Contributions . . . . .	46
6.2 Future work . . . . .	46

---

In this work, a MOSCC was successfully sized for the M4M application with the aid of an original method for sizing the components of the MOSCC. The method is generic enough that it was applied to six different candidate topologies. The method was implemented in Python and was validated through simulation that it is able to generate a correct sizing of the MOSCC components. The simulated output characteristics of the sized MOSCC were very close to the specifications passed to the algorithm, with errors up to 1%. However the simulation results show considerable error on the losses calculated from the presented losses model.

## 6.1 Contributions

The main contribution of this work was the development of a method to optimally size the components of a MOSCC. The method allows the sizing of any two phase MOSCC with any level of integration (discrete, integrated switches, fully-integrated) as long as it is described as a connection of multiple stages. The proposed method provides a helpful contribution to the design of a real application, a neural stimulation implant, which is being designed at the time of writing. It will also be useful for any other future application requiring multiple supply rails.

## 6.2 Future work

Sizing the SCC is only one of the first steps required in the development of the M4M application. A number of auxiliary circuit still need to be designed, including but not limited to output voltage regulation circuits that enable power saving for light loads, gate drivers, clock generation circuits and inrush current limiter. After designing the circuits, the silicon layout of the entire system with capacitors, switches and all auxiliary circuits needs to be done.

The algorithm for determining the best stage conductance distribution is essentially brute force and is not scalable for converters with many stages. Other optimization algorithms incorporating, for example, gradient descent, genetic or even machine learning algorithms could be investigated to make the method scalable.

The method does not take into account the output ripple and how the topology characteristics, switching frequency and output capacitor sizing influence the total solution size and performance. Introducing this considerations in the method could lead to more optimized solutions.



# Bibliography

- [1] Moore4medical website. Last accessed 11/10/2021. [Online]. Available: <https://moore4medical.eu/>
- [2] S. Ben-Yaakov and M. Evzelman, "Generic and unified model of Switched Capacitor Converters," in *2009 IEEE Energy Conversion Congress and Exposition*, no. 2. IEEE, sep 2009, pp. 3501–3508.
- [3] M. Evzelman and S. Ben-Yaakov, "Optimal switch resistances in Switched Capacitor Converters," in *2010 IEEE 26-th Convention of Electrical and Electronics Engineers in Israel*. IEEE, nov 2010, pp. 000 436–000 439.
- [4] M. S. Makowski and A. Kushnerov, "Canonical switched capacitor converters. Comments, complements, and refinements," in *2017 European Conference on Circuit Theory and Design (ECCTD)*. IEEE, sep 2017, pp. 1–4.
- [5] J. W. Kimball and P. T. Krein, "Analysis and design of switched capacitor converters," *Conference Proceedings - IEEE Applied Power Electronics Conference and Exposition - APEC*, vol. 3, pp. 1473–1477, 2005.
- [6] Y. Allasasmeh and S. Gregori, "High-Performance Switched-Capacitor Boost–Buck Integrated Power Converters," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 65, no. 11, pp. 3970–3983, nov 2018.
- [7] M. D. Seeman, "A Design Methodology for Switched-Capacitor DC-DC Converters," *Spring*, vol. 9, no. 4, pp. 981–984, 2009.
- [8] J. Dickson, "On-chip high-voltage generation in mnos integrated circuits using an improved voltage multiplier technique," *IEEE Journal of Solid-State Circuits*, vol. 11, no. 3, pp. 374–378, 1976.
- [9] M. D. Seeman and S. R. Sanders, "Analysis and Optimization of Switched-Capacitor DC–DC Converters," *IEEE Transactions on Power Electronics*, vol. 23, no. 2, pp. 841–851, mar 2008.
- [10] J. Delos, T. Lopez, E. Alarcon, and M. A. M. Hendrix, "On the modeling of switched capacitor converters with multiple outputs," in *2014 IEEE Applied Power Electronics Conference and Exposition - APEC 2014*. IEEE, mar 2014, pp. 2796–2803.

- [11] Wing-Hung Ki, Feng Su, and Chi-Ying Tsui, "Charge Redistribution Loss Consideration in Optimal Charge Pump Design," in *2005 IEEE International Symposium on Circuits and Systems*, vol. 2, no. 8. IEEE, 2005, pp. 1895–1898.
- [12] W. Hong, Z. Zhou, B. Nguyen, N. Tang, J. Kim, P. P. Pande, and D. Heo, "Analysis and Design Method of Multiple-Output Switched-Capacitor Voltage Regulators With a Reduced Number of Power Electronic Components," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 67, no. 9, pp. 3234–3247, sep 2020.
- [13] A. Rashidi, N. Yazdani, and A. M. Sodagar, "Fully-implantable, multi-channel, microstimulator with tracking supply ribbon and energy recovery," *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, vol. 2016-Octob, pp. 1818–1821, 2016.
- [14] R. Pelliconi, D. Iezzi, A. Baroni, M. Pasotti, and P. L. Rolandi, "Power efficient charge pump in deep submicron standard CMOS technology," *European Solid-State Circuits Conference*, vol. 38, no. 6, pp. 73–76, 2001.
- [15] N. Butzen and M. S. J. Steyaert, "Design of Soft-Charging Switched-Capacitor DC–DC Converters Using Stage Outphasing and Multiphase Soft-Charging," *IEEE Journal of Solid-State Circuits*, vol. 52, no. 12, pp. 3132–3141, dec 2017.
- [16] A. Rashidi, N. Yazdani, and A. M. Sodagar, "Fully-Integrated, High-Efficiency, Multi-Output Charge Pump for High-Density Microstimulators," in *2018 IEEE Life Sciences Conference (LSC)*. IEEE, oct 2018, pp. 291–294.



# Optimization algorithm source code

**Listing A.1:** File topology.py

```
1 # -*- coding: utf-8 -*-
2 """
3 Created on Thu Oct  7 11:03:46 2021
4
5 @author: aagostin
6
7 Definition of a topology object and the methods for calculating total
8     solution
9 performance and optimization of component sizing.
10 """
11 import numpy as np
12 import math
13
14 class topology_performance(object):
```

```

15     def __init__(self):
16         self.Vin = 0 #[V]
17         self.Iin = 0 #[mA]
18         self.Pin = 0 #[W]
19         self.Vout = 0 #[V]
20         self.Iout = 0 #[mA]
21         self.Pout = 0 #[mW]
22         self.Rout = 0 #[Ohm]
23         self.Pcap = 0 #[mW]
24         self.Psw = 0 #[mW]
25         self.PSSL = 0 #[mW]
26         self.PFSL = 0 #[mW]
27         self.Prout = 0 #[mW]
28         self.Ploss = 0 #[mW]
29         self.fsw = 0 #[MHz]
30         self.eff = 0 [%]
31         self.Acap = 0 #[mm^2]
32         self.Asw = 0 #[mm^2]
33         self.Atot = 0 #[mm^2]
34         self.Pdensity = 0 #[mW/mm^2]
35         self.reg_factor = 0 #[ratio]
36         self.reg_lim_out = 0
37
38     class topology(object):
39
40         def __init__(self):
41             # topology characteristics
42             self.n_cap = 1
43             self.n_sw = 4
44             self.n_out = 1
45             self.vout_ratio = np.ones([self.n_out]) # ideal Vout/Vin ideal
46             self.coeff_b = np.zeros([self.n_cap,self.n_out]) # cap charge
47                 multiplier coeffs
48             self.coeff_r = np.zeros([self.n_sw,self.n_out]) # switch charge
49                 multiplier coeffs
50             self.vcb_ratio = np.zeros([self.n_cap]) # cap bot plate swing ratio
51                 to Vin

```

```

49     self.vsg_ratio = np.zeros([self.n_sw]) # sw gate drive swing ratio to
        Vin
50
51     # technology device assignement
52     self.caps = [] # capacitor tech
53     self.sws = [] # switch tech
54
55     # component sizing
56     self.Ccap = np.zeros([self.n_cap]) #[F]
57     self.Ccap_tot = 0 #[F]
58     self.Gsw = np.array([0]) #[S]
59     self.Gsw_tot = 0 #[S]
60     self.r = np.zeros([self.n_cap]) # FSL/SSL of each stage
61
62     # areas
63     self.Acap = np.array([self.n_cap]) #[m^2]
64     self.Acap_tot = 0 #[m^2]
65     self.Asw = np.array([self.n_sw]) #[m^2]
66     self.Asw_tot = 0 #[m^2]
67     self.Atot = 0 #[m^2]
68
69     def calc_loss_noreg(self, Vin, Iout, fsw, duty):
70         # calculate transimpedance matrixes
71         ZSSL = np.zeros([self.n_out, self.n_out])
72         ZFSL = np.zeros([self.n_out, self.n_out])
73         Ztot = np.zeros([self.n_out, self.n_out])
74         for i in range(self.n_out):
75             for j in range(self.n_out):
76                 for c in range(self.n_cap):
77                     ZSSL[i][j] += 1/(self.Ccap[c]*fsw) * self.coeff_b[c,i]*
                        self.coeff_b[c,j]
78         for i in range(self.n_out):
79             for j in range(self.n_out):
80                 for s in range(self.n_sw):
81                     ZFSL[i][j] += 1/(self.Gsw[s]*duty) * self.coeff_r[s,i]*
                        self.coeff_r[s,j]
82         for i in range(self.n_out):
83             for j in range(self.n_out):

```

```

84         Ztot[i][j] = math.sqrt(ZSSL[i][j]**2+ZFSL[i][j]**2)
85
86         # calculate cap and sw losses
87         Pcap = [self.Ccap[i]/self.caps[i].mCl * (self.vcb_ratio[i]*Vin)**2 *
88                 fsw for i in range(self.n_cap)]
89         Pcap_tot = np.sum(Pcap)
90         Psw = [self.Gsw[i]/self.sws[i].mSl * fsw for i in range(self.n_sw)]
91         Psw_tot = np.sum(Psw)
92
93         VSSL = (ZSSL @ Iout[None].T).T
94         PSSL = np.sum(Iout[None].T @ Iout[None] * ZSSL, axis=1)
95         PSSL_tot = np.sum(Iout[None].T @ Iout[None] * ZSSL)
96
97         VFSL = (ZFSL @ Iout[None].T).T
98         PFSL = np.sum(Iout[None].T @ Iout[None] * ZFSL, axis=1)
99         PFSL_tot = np.sum(Iout[None].T @ Iout[None] * ZFSL)
100
101         PRout_tot2 = np.sum(Iout[None].T @ Iout[None] * Ztot)
102
103         VRout = np.sqrt(VSSL**2+VFSL**2)
104         PRout = np.sqrt(PSSL**2+PFSL**2)
105         PRout_tot = np.sqrt(PSSL_tot**2 + PFSL_tot**2)
106
107         #Pout = self.vout_ratio * Vin * Iout - PRout
108         Pout = (self.vout_ratio * Vin - VRout) * Iout
109         Pout_tot = np.sum(Pout)
110
111         Ploss_tot = Pcap_tot + Psw_tot + PRout_tot
112
113         Pin = Pout_tot + Ploss_tot
114
115         Pdensity = Pout_tot/self.Atot
116         eff = Pout_tot/(Ploss_tot+Pout_tot)
117
118         perf = topology_performance()
119         perf.Vin = Vin # [V]
120         perf.Iin = Pin/perf.Vin * 1000 # [mA]
121         perf.Pin = (Pout_tot + Ploss_tot) * 1000 # [mW]

```

```

121     perf.Vout = self.vout_ratio * Vin - VRout #[V]
122     perf.Iout = Iout * 1000 #[mA]
123     perf.Pout = Pout * 1000 #[mW]
124     perf.Rout = VRout/Iout #[Ohm]
125     perf.Pcap = Pcap_tot *1000 #[mW]
126     perf.Psw = Psw_tot *1000 #[mW]
127     perf.PSSL = PSSL *1000 #[mW]
128     perf.PFSL = PFSL *1000 #[mW]
129     perf.Prout = PRout_tot *1000 #[mW]
130     perf.Prout2 = PRout_tot2 *1000 #[mW]
131     perf.Ploss = Ploss_tot *1000 #[mW]
132     perf.fsw = fsw /1e6 #[MHz]
133     perf.eff = eff * 100 #[%]
134     perf.Acap = self.Acap_tot *1e6 #[mm^2]
135     perf.Asw = self.Asw_tot *1e6 #[mm^2]
136     perf.Atot = self.Atot *1e6 #[mm^2]
137     perf.Pdensity = Pdensity *1e3/1e6 #[mW/mm^2]
138     return perf
139
140 def calc_loss_reg(self, Vin, Iout, fsw, duty, spec_maxdrop):
141     ZSSL = np.zeros([self.n_out,self.n_out])
142     ZFSL = np.zeros([self.n_out,self.n_out])
143     for i in range(self.n_out):
144         for j in range(self.n_out):
145             for c in range(self.n_cap):
146                 ZSSL[i][j] += 1/(self.Ccap[c]*fsw) * self.coeff_b[c,i]*
147                     self.coeff_b[c,j]
148     for i in range(self.n_out):
149         for j in range(self.n_out):
150             for s in range(self.n_sw):
151                 ZFSL[i][j] += 1/(self.Gsw[s]*duty) * self.coeff_r[s,i]*
152                     self.coeff_r[s,j]
153
154     VSSL = (ZSSL @ Iout[None].T).T
155     VFSL = (ZFSL @ Iout[None].T).T
156     VRout = np.sqrt(VSSL**2+VFSL**2)
157
158     drop_factor = VRout/spec_maxdrop

```

```

157     reg_factor = np.max(drop_factor)
158     lim_out = np.argmax(drop_factor)
159
160     Pcap = np.array([self.Ccap[i]/self.caps[i].mCl * (self.vcb_ratio[i]*
        Vin)**2 * fsw for i in range(self.n_cap)]) * reg_factor
161     Pcap_tot = np.sum(Pcap)
162     Psw = np.array([self.Gsw[i]/self.sws[i].mSl * fsw for i in range(
        self.n_sw)]) * reg_factor
163     Psw_tot = np.sum(Psw)
164
165     VSSL = (ZSSL @ Iout[None].T).T / reg_factor
166     PSSL = np.sum(Iout[None].T @ Iout[None] * ZSSL, axis=1) / reg_factor
167     PSSL_tot = np.sum(Iout[None].T @ Iout[None] * ZSSL) / reg_factor
168
169     VFSL = (ZFSL @ Iout[None].T).T / reg_factor
170     PFSL = np.sum(Iout[None].T @ Iout[None] * ZFSL, axis=1) / reg_factor
171     PFSL_tot = np.sum(Iout[None].T @ Iout[None] * ZFSL) / reg_factor
172
173     VRout = np.sqrt(VSSL**2+VFSL**2)
174     PRout = np.sqrt(PSSL**2+PFSL**2)
175     PRout_tot = np.sqrt(PSSL_tot**2 + PFSL_tot**2)
176
177     Pout = (self.vout_ratio * Vin - VRout) * Iout
178     Pout_tot = np.sum(Pout)
179
180     Ploss_tot = Pcap_tot + Psw_tot + PRout_tot
181
182     Pin = Pout_tot + Ploss_tot
183
184     Pdensity = Pout_tot/self.Atot
185     eff = Pout_tot/(Ploss_tot+Pout_tot)
186
187     perf = topology_performance()
188     perf.Vin = Vin #[V]
189     perf.Iin = Pin/perf.Vin * 1000 #[mA]
190     perf.Pin = (Pout_tot + Ploss_tot) * 1000 #[mW]
191     perf.Vout = self.vout_ratio * Vin - VRout #[V]
192     perf.Iout = Iout * 1000 #[mA]

```



```

193     perf.Pout = Pout * 1000 #[mW]
194     perf.Rout = VRout/Iout #[Ohm]
195     perf.Pcap = Pcap_tot *1000 #[mW]
196     perf.Psw = Psw_tot *1000 #[mW]
197     perf.PSSL = PSSL *1000 #[mW]
198     perf.PFSL = PFSL *1000 #[mW]
199     perf.Prout = PRout_tot *1000 #[mW]
200     perf.Ploss = Ploss_tot *1000 #[mW]
201     perf.fsw = fsw /1e6 #[MHz]
202     perf.eff = eff * 100 #[%]
203     perf.Acap = self.Acap_tot *1e6 #[mm^2]
204     perf.Asw = self.Asw_tot *1e6 #[mm^2]
205     perf.Atot = self.Atot *1e6 #[mm^2]
206     perf.Pdensity = Pdensity *1e3/1e6 #[mW/mm^2]
207     perf.reg_factor = reg_factor
208     perf.reg_lim_out = lim_out
209     return perf
210
211     def calc_areas(self):
212         self.Acap = [self.Ccap[i]/self.caps[i].mCa for i in range(self.n_cap)
213                     ]
214         self.Acap_tot = np.sum(self.Acap)
215         self.Asw = [self.Gsw[i]/self.sws[i].mSa for i in range(self.n_sw)]
216         self.Asw_tot = np.sum(self.Asw)
217         self.Atot = self.Acap_tot + self.Asw_tot
218
219     def optimize_component_sizing(self, Vin, spec_maxI, spec_maxdrop, fsw, dc
220     , lamb, n_val):
221         """
222         Optimizes the topology component sizing for a given specification
223         """
224         # Generate sizing combinations
225         n_comb=n_val**self.n_cap
226         val=np.linspace(1, n_val, n_val)
227         combinations=np.zeros([n_val**self.n_cap, self.n_cap])
228         for i in range(self.n_cap):
229             for j in range(n_val**self.n_cap):
230                 k=(j//((n_val**(self.n_cap-1-i))))%n_val

```

```

229         combinations[j,i]=val[k]
230     for i in range(n_comb):
231         combinations[i,:]=combinations[i,:]/sum(combinations[i,:])
232
233     zeta=np.zeros([n_comb,self.n_out,self.n_out]) # output transimpedance
           factor matrix
234     vo_coeff=np.zeros([n_comb,self.n_out]) # total output impedance
           factor
235     valid_sol=np.full([n_comb],True) # valid combinations
236     limit_out=np.zeros([n_comb],dtype=int) # limiting output for each
           combination
237
238     Gsttot=np.zeros([n_comb]) # total FSL stage conductance for each
           combination
239     st_r=np.zeros([n_comb,self.n_cap]) # each stage FSL/SSL ratio
240
241     capC = np.zeros([n_comb,self.n_cap]) # capacitance of each capacitor
242     swG = np.zeros([n_comb,self.n_sw]) # conductance of each switch
243     Ctot=np.zeros([n_comb]) # total capacitance for each combination
244     Gtot=np.zeros([n_comb]) # total FSL stage conductance for each
           combination
245
246     Atot=np.zeros([n_comb])
247     Ploss=np.zeros([n_comb])
248     Pdensity = np.zeros([n_comb])
249     eff = np.zeros([n_comb])
250
251     # calculate all combination performance
252     for comb, Gst_frac in enumerate(combinations):
253
254         ## Calculate zeta coefficients
255         for i in range(self.n_out):
256             for j in range(self.n_out):
257                 for c in range(self.n_cap):
258                     zeta[comb][i][j] += 1/(Gst_frac[c]) * self.coeff_b[c,
                                   i]*self.coeff_b[c,j]
259
260

```

```

261     ## Size total stage conductance
262     valid_sol[comb] = np.all(zeta[comb]>=0)
263     #vo_coef[comb] = (zeta[comb] @ spec_maxI[None].T).T
264     #limit_out[comb]=np.argmax(vo_coef[comb]/(spec_maxdrop), axis=0)
265     Greq=((zeta[comb] @ spec_maxI[None].T).T)/spec_maxdrop
266     Gsttot[comb]=np.max(Greq)
267     limit_out[comb]=np.argmax(Greq)
268
269     ## Intra stage optimization
270     for st in range(self.n_cap):
271         Gst = Gsttot[comb]*Gst_frac[st]
272         Rst = 1/Gst
273
274         ### find r
275         KAcap = 1/(fsw*self.caps[st].mCa)
276         KPcpar = (self.vcb_ratio[st]*Vin)**2/self.caps[st].mCl
277         st_mSl_sum = np.sum(np.array( [1/np.sqrt(self.sws[a].mSl) for
278             a in range(st*4,(st+1)*4)] ))
279         KAsw = np.sum(np.array( [np.sqrt(self.sws[a].mSl)/(dc*self.
280             sws[a].mSa)*st_mSl_sum for a in range(st*4,(st+1)*4)] ))
281         KPsdrv = np.sum(np.array( [fsw/(dc*np.sqrt(self.sws[a].mSl))*
282             st_mSl_sum for a in range(st*4,(st+1)*4)] ))
283         st_r[comb,st] = ((KAsw + lamb/1000*KPsdrv)/(KAcap + lamb/100
284             0*KPcpar))**(1/3)
285
286         ### calc C
287         st_rssl = 1/(1+st_r[comb,st]**2)**0.5 * Rst
288         capC[comb,st] = 1/(fsw*st_rssl)
289
290         ### calc Gi
291         st_rfsl = st_r[comb,st]/(1+st_r[comb,st]**2)**0.5 * Rst
292         swG[comb,st*4:(st+1)*4] = np.array([np.sqrt(self.sws[a].mSl)
293             /(dc*st_rfsl)*st_mSl_sum for a in range(st*4,(st+1)*4)])
294
295     Ctot[comb] = np.sum(capC[comb])
296     Gtot[comb] = np.sum(swG[comb])
297
298     # calculate performance

```

```

294         self.Ccap = capC[comb] #[F]
295         self.Ccap_tot = Ctot[comb] #[F]
296         self.Gsw = swG[comb] #[S]
297         self.Gsw_tot = Gtot[comb] #[S]
298         self.r = st_r
299         self.calc_areas()
300         perf = self.calc_loss_noreg(Vin, spec_maxI, fsw, 0.5)
301         Atot[comb]=perf.Atot
302         Ploss[comb]=perf.Ploss
303         Pdensity[comb] = perf.Pdensity
304         eff[comb] = perf.eff
305
306         # determine best solution (for now its the most power dense)
307         opt_solution_p = np.argmax(Pdensity*valid_sol)
308         opt_solution_e = np.argmax(eff*valid_sol)
309         cost=Atot+Ploss*lamb
310         opt_solution=np.argmin(cost+(1-valid_sol)*np.max(cost))
311
312         # assign the best solution to the topology
313         self.Ccap = capC[opt_solution] #[F]
314         self.Ccap_tot = Ctot[opt_solution] #[F]
315         self.Gsw = swG[opt_solution] #[S]
316         self.Gsw_tot = Gtot[opt_solution] #[S]
317         self.r = st_r
318         self.calc_areas()
319         perf = self.calc_loss_noreg(Vin, spec_maxI, fsw, 0.5)
320
321         return perf

```

**Listing A.2:** File topologies.py

```

1  # -*- coding: utf-8 -*-
2  """
3  Created on Sun Oct 24 10:58:56 2021
4
5  @author: aagostin
6
7  Definitions of the topologies to evaluate
8  """

```

```

9 import numpy as np
10 import topology as tp
11 import tech_devices as tech
12
13 # top 1 - linear
14 top1 = tp.topology()
15 top1.n_cap=5
16 top1.n_sw=4*5
17 top1.n_out=5
18 top1.vout_ratio = np.array([1/3, 2/3, 4/3, 5/3, 6/3])
19 top1.coeff_b=np.array([
20     [ -1/3,  1/3, -1/3,  -1,  -1],
21     [  1/3,  2/3,  1/3,   1,   1],
22     [   0,   0,   1,   1,   1],
23     [   0,   0,   0,   1,   1],
24     [   0,   0,   0,   0,   1]
25 ])
26 top1.coeff_r=np.repeat(top1.coeff_b,4,axis=0) # each 4 switches as same
        coeffs as cap
27 top1.vcb_ratio = np.array([1/3, 1/3, 1/3, 1/3, 1/3])
28 #top1.vsg_ratio = np.array([1/3]*top1.n_sw)
29 top1.caps=[tech.cap_mos2v, tech.cap_mos5v, tech.cap_mos5v, tech.cap_mom70v,
        tech.cap_mom70v]
30 top1.sws=[tech.sw_nmos2v, tech.sw_nmos2v, tech.sw_pmos2v, tech.sw_pmos2v,
31     tech.sw_nmos2v, tech.sw_nmos2v, tech.sw_pmos2v, tech.sw_pmos2v,
32     tech.sw_nmos2v, tech.sw_nmos2v, tech.sw_pmos2v, tech.sw_pmos2v,
33     tech.sw_nmos2v, tech.sw_nmos2v, tech.sw_pmos2v, tech.sw_pmos2v,
34     tech.sw_nmos2v, tech.sw_nmos2v, tech.sw_pmos2v, tech.sw_pmos2v]
35
36
37 # top2 - nonlinear 1 ( $Vo4=Vi+(Vo1-0)$ ,  $Vo5=Vi+(Vo2-0)$ ,  $Vo6=Vi+(Vi-0)$ )
38 top2 = tp.topology()
39 top2.n_cap=5
40 top2.n_sw=4*5
41 top2.n_out=5
42 top2.vout_ratio = np.array([1/3, 2/3, 4/3, 5/3, 6/3])
43 top2.coeff_b=np.array([
44     [ -1/3,  1/3, -1/3,  1/3,  0],

```

```

45     [ 1/3, 2/3, 1/3, 2/3, 0],
46     [ 0, 0, 1, 0, 0],
47     [ 0, 0, 0, 1, 0],
48     [ 0, 0, 0, 0, 1]
49     ])
50 top2.coeff_r=np.repeat(top2.coeff_b,4,axis=0) # each 4 switches as same
        coeffs as cap
51 top2.vcb_ratio = np.array([1/3, 1/3, 1/3, 2/3, 3/3])
52 #top2.vsg_ratio = np.array([1/3]*top2.n_sw)
53 top2.caps=[tech.cap_mos2v, tech.cap_mos5v, tech.cap_mos5v, tech.cap_mos5v,
        tech.cap_mim5v]
54 top2.sws=[tech.sw_nmos2v, tech.sw_nmos2v, tech.sw_pmos2v, tech.sw_pmos2v,
55         tech.sw_nmos2v, tech.sw_nmos2v, tech.sw_pmos2v, tech.sw_pmos2v,
56         tech.sw_nmos2v, tech.sw_nmos2v, tech.sw_pmos2v, tech.sw_pmos2v,
57         tech.sw_nmos5v, tech.sw_nmos5v, tech.sw_pmos5v, tech.sw_pmos5v,
58         tech.sw_nmos5v, tech.sw_nmos5v, tech.sw_pmos5v, tech.sw_pmos5v]
59
60
61 # top3 - nonlinear 2 (Vo4=Vo1+(Vi-0), Vo5=Vo2+(Vi-0), Vo6=Vi+(Vi-0))
62 top3 = tp.topology()
63 top3.n_cap=5
64 top3.n_sw=4*5
65 top3.n_out=5
66 top3.vout_ratio = np.array([1/3, 2/3, 4/3, 5/3, 6/3])
67 top3.coeff_b=np.array([
68     [-1/3, 1/3, -1/3, 1/3, 0],
69     [ 1/3, 2/3, 1/3, 2/3, 0],
70     [ 0, 0, 1, 0, 0],
71     [ 0, 0, 0, 1, 0],
72     [ 0, 0, 0, 0, 1]
73     ])
74 top3.coeff_r=np.repeat(top3.coeff_b,4,axis=0) # each 4 switches as same
        coeffs as cap
75 top3.vcb_ratio = np.array([1/3, 1/3, 3/3, 3/3, 3/3])
76 #top3.vsg_ratio = np.array([1/3]*top3.n_sw)
77 top3.caps=[tech.cap_mos2v, tech.cap_mos5v, tech.cap_mos2v, tech.cap_mim5v,
        tech.cap_mim5v]
78 top3.sws=[tech.sw_nmos2v, tech.sw_nmos2v, tech.sw_pmos2v, tech.sw_pmos2v,

```

```

79         tech.sw_nmos2v, tech.sw_nmos2v, tech.sw_pmos2v, tech.sw_pmos2v,
80         tech.sw_nmos5v, tech.sw_nmos5v, tech.sw_pmos5v, tech.sw_pmos5v,
81         tech.sw_nmos5v, tech.sw_nmos5v, tech.sw_pmos5v, tech.sw_pmos5v,
82         tech.sw_nmos5v, tech.sw_nmos5v, tech.sw_pmos5v, tech.sw_pmos5v]
83
84
85 # top4 - nolinear 3 (Vo4=Vi+(Vo1-0), Vo5=Vo2+(Vi-0), Vo6=Vi+(Vi-0))
86 top4 = tp.topology()
87 top4.n_cap=5
88 top4.n_sw=4*5
89 top4.n_out=5
90 top4.vout_ratio = np.array([1/3, 2/3, 4/3, 5/3, 6/3])
91 top4.coeff_b=np.array([
92     [-1/3, 1/3, -1/3, 1/3, 0],
93     [ 1/3, 2/3, 1/3, 2/3, 0],
94     [ 0, 0, 1, 0, 0],
95     [ 0, 0, 0, 1, 0],
96     [ 0, 0, 0, 0, 1]
97 ])
98 top4.coeff_r=np.repeat(top4.coeff_b,4,axis=0) # each 4 switches as same
coeffs as cap
99 top4.vcb_ratio = np.array([1/3, 1/3, 1/3, 3/3, 3/3])
100 #top4.vsg_ratio = np.array([1/3]*top4.n_sw)
101 top4.caps=[tech.cap_mos2v, tech.cap_mos5v, tech.cap_mos5v, tech.cap_mim5v,
102            tech.cap_mim5v]
103 top4.sws=[tech.sw_nmos2v, tech.sw_nmos2v, tech.sw_pmos2v, tech.sw_pmos2v,
104           tech.sw_nmos2v, tech.sw_nmos2v, tech.sw_pmos2v, tech.sw_pmos2v,
105           tech.sw_nmos5v, tech.sw_nmos5v, tech.sw_pmos5v, tech.sw_pmos5v,
106           tech.sw_nmos5v, tech.sw_nmos5v, tech.sw_pmos5v, tech.sw_pmos5v]
107
108 # top5 - nolinear 4 (Vo4=Vo1+(Vi-0), Vo5=Vi+(Vo2-0), Vo6=Vi+(Vi-0))
109 top5 = tp.topology()
110 top5.n_cap=5
111 top5.n_sw=4*5
112 top5.n_out=5
113 top5.vout_ratio = np.array([1/3, 2/3, 4/3, 5/3, 6/3])
114 top5.coeff_b=np.array([

```

```

115     [ -1/3,  1/3, -1/3,  1/3,  0],
116     [  1/3,  2/3,  1/3,  2/3,  0],
117     [   0,   0,   1,   0,  0],
118     [   0,   0,   0,   1,  0],
119     [   0,   0,   0,   0,  1]
120     ])
121 top5.coeff_r=np.repeat(top5.coeff_b,4,axis=0) # each 4 switches as same
        coeffs as cap
122 top5.vcb_ratio = np.array([1/3, 1/3, 3/3, 2/3, 3/3])
123 #top5.vsg_ratio = np.array([1/3]*top5.n_sw)
124 top5.caps=[tech.cap_mos2v, tech.cap_mos5v, tech.cap_mos2v, tech.cap_mos5v,
        tech.cap_mim5v]
125 top5.sws=[tech.sw_nmos2v, tech.sw_nmos2v, tech.sw_pmos2v, tech.sw_pmos2v,
126         tech.sw_nmos2v, tech.sw_nmos2v, tech.sw_pmos2v, tech.sw_pmos2v,
127         tech.sw_nmos5v, tech.sw_nmos5v, tech.sw_pmos5v, tech.sw_pmos5v,
128         tech.sw_nmos5v, tech.sw_nmos5v, tech.sw_pmos5v, tech.sw_pmos5v,
129         tech.sw_nmos5v, tech.sw_nmos5v, tech.sw_pmos5v, tech.sw_pmos5v]
130
131 # top6 - buck boost ( $Vo_4=Vo_5-(Vo_4-V_i)$ ,  $Vo_5=Vo_6-(Vo_4-V_i)$ ,  $Vo_6=V_i+(V_i-0)$ )
132 top6 = tp.topology()
133 top6.n_cap=5
134 top6.n_sw=4*5
135 top6.n_out=5
136 top6.vout_ratio = np.array([1/3, 2/3, 4/3, 5/3, 6/3])
137 top6.coeff_b=np.array([
138     [ -1/3,  1/3,   0,   0,  0],
139     [  1/3,  2/3,   0,   0,  0],
140     [   0,   0, -1/3,  1/3,  0],
141     [   0,   0,  1/3,  2/3,  0],
142     [   0,   0,  1/3,  2/3,  1]
143     ])
144 top6.coeff_r=np.repeat(top6.coeff_b,4,axis=0) # each 4 switches as same
        coeffs as cap
145 top6.vcb_ratio = np.array([1/3, 1/3, 1/3, 1/3, 3/3])
146 #top6.vsg_ratio = np.array([1/3]*top6.n_sw)
147 top6.caps=[tech.cap_mos2v, tech.cap_mos5v, tech.cap_mos2v, tech.cap_mos5v,
        tech.cap_mim5v]
148 top6.sws=[tech.sw_nmos2v, tech.sw_nmos2v, tech.sw_pmos2v, tech.sw_pmos2v,

```



```

149         tech.sw_nmos2v, tech.sw_nmos2v, tech.sw_pmos2v, tech.sw_pmos2v,
150         tech.sw_nmos2v, tech.sw_nmos2v, tech.sw_pmos2v, tech.sw_pmos2v,
151         tech.sw_nmos2v, tech.sw_nmos2v, tech.sw_pmos2v, tech.sw_pmos2v,
152         tech.sw_nmos5v, tech.sw_nmos5v, tech.sw_pmos5v, tech.sw_pmos5v]

```

**Listing A.3:** File tech\_devices.py

```

1  # -*- coding: utf-8 -*-
2  """
3  Created on Thu Oct 7 17:45:48 2021
4
5  @author: aagostin
6
7  Technology device performance definitions
8  """
9  class tech_cap(object):
10
11      def __init__(self):
12          self.mCa = 0 #[F/m^2] area metric
13          self.par_a = 0 #[ratio] metric bottom plate parasitics
14          self.par_b = 0 #[ratio] metric bottom plate parasitics
15          self.mCl = 0 #[F/F=ratio] loss metric (refers to parasitic
16                      capacitance)
17
18  class tech_sw(object):
19
20      def __init__(self):
21          self.mSa = 0 #[S/m^2] area metric
22          self.mSl = 0 #[S/F] loss metric (refers to gate capacitance)
23
24  # CAPS
25  cap_mos2v = tech_cap()
26  cap_mos2v.mCa = 8.9/1e3 #[F/m^2]
27  cap_mos2v.par_a = 0.01 #[ratio]
28  cap_mos2v.par_b = 0.00 #[ratio]
29  cap_mos2v.mCl = 100 #[ratio]
30
31  cap_mos5v = tech_cap()
32  cap_mos5v.mCa = 4.1/1e3 #[F/m^2]

```

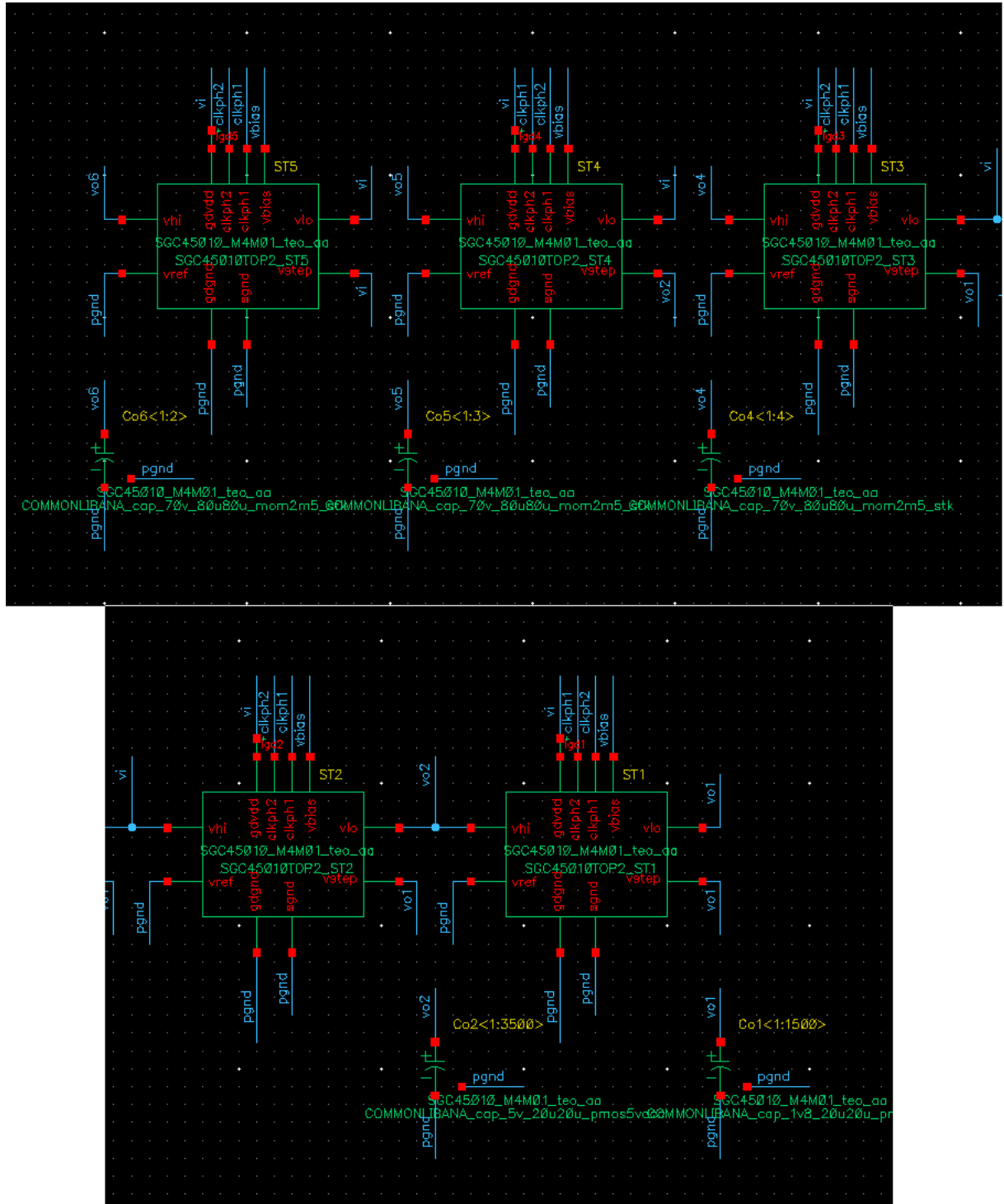
```

32 cap_mos5v.par_a = 0.02 #[ratio]
33 cap_mos5v.par_b = 0.00 #[ratio]
34 cap_mos5v.mCl = 50 #[ratio]
35
36 cap_mim5v = tech_cap()
37 cap_mim5v.mCa = 1.55/1e3 #[F/m^2]
38 cap_mim5v.par_a = 0.007 #[ratio]
39 cap_mim5v.par_b = 0.00 #[ratio]
40 cap_mim5v.mCl = 143 #[ratio]
41
42 cap_mom70v = tech_cap()
43 cap_mom70v.mCa = 0.8/1e3 #[F/m^2]
44 cap_mom70v.par_a = 0.022 #[ratio]
45 cap_mom70v.par_b = 0.022 #[ratio]
46 cap_mom70v.mCl = 23 #[ratio]
47
48 # SWS
49 sw_nmos2v = tech_sw()
50 sw_nmos2v.mSa = 0.97*1e9 #[S/m^2]
51 sw_nmos2v.mSl = 440*1e9 #[S/J]=[mS/pJ]*1e9
52
53 sw_pmos2v = tech_sw()
54 sw_pmos2v.mSa = 0.36*1e9 #[S/m^2]
55 sw_pmos2v.mSl = 120*1e9 #[S/J]=[mS/pJ]*1e9
56
57 sw_nmos5v = tech_sw()
58 sw_nmos5v.mSa = 0.29*1e9 #[S/m^2]
59 sw_nmos5v.mSl = 17*1e9 #[S/J]=[mS/pJ]*1e9
60
61 sw_pmos5v = tech_sw()
62 sw_pmos5v.mSa = 0.11*1e9 #[S/m^2]
63 sw_pmos5v.mSl = 5.5*1e9 #[S/J]=[mS/pJ]*1e9

```

# B

## **Simulation schematics**



**Figure B.1:** Schematic of the implemented topology TOP2.

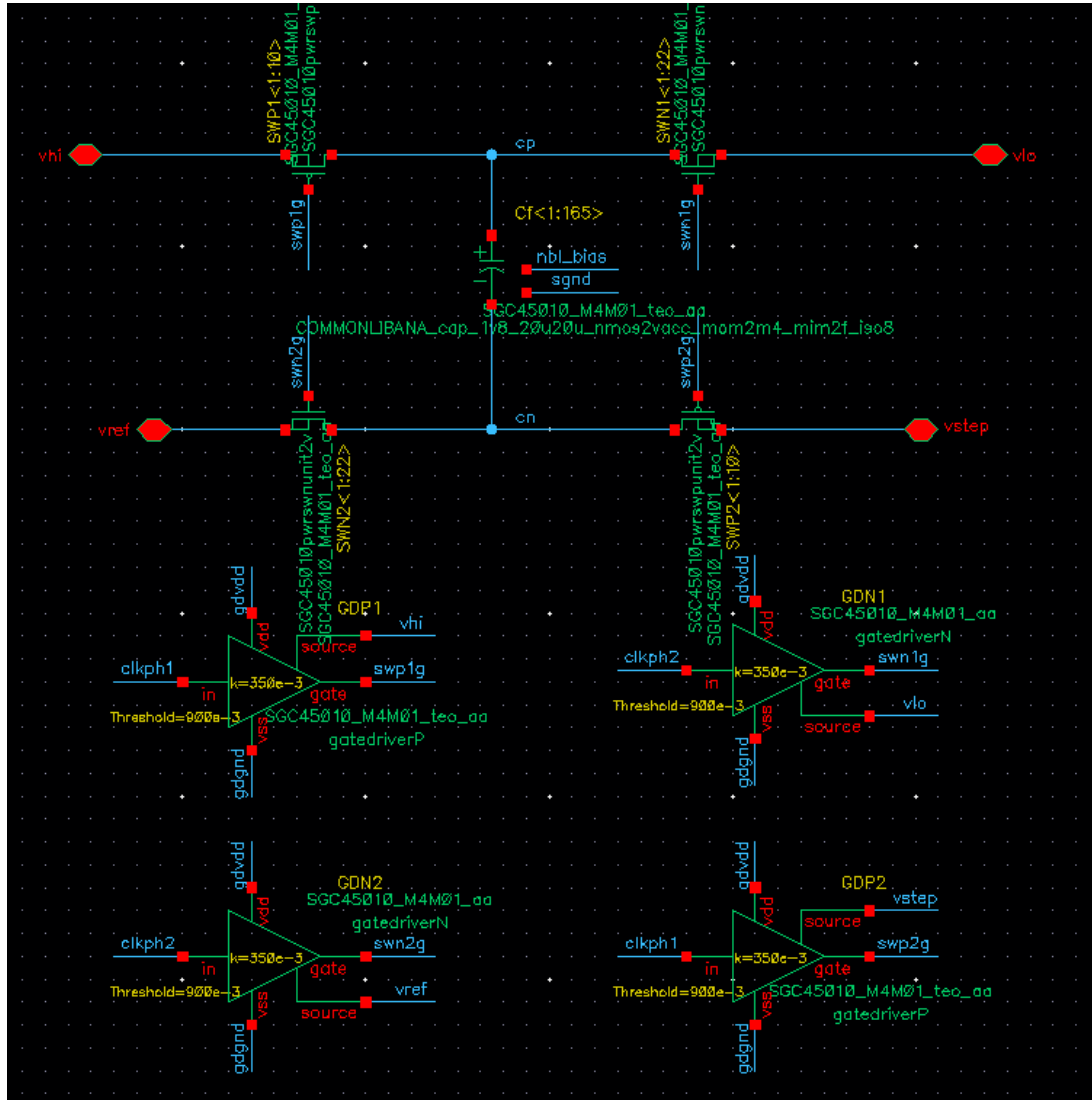


Figure B.2: Schematic of stage ST1.

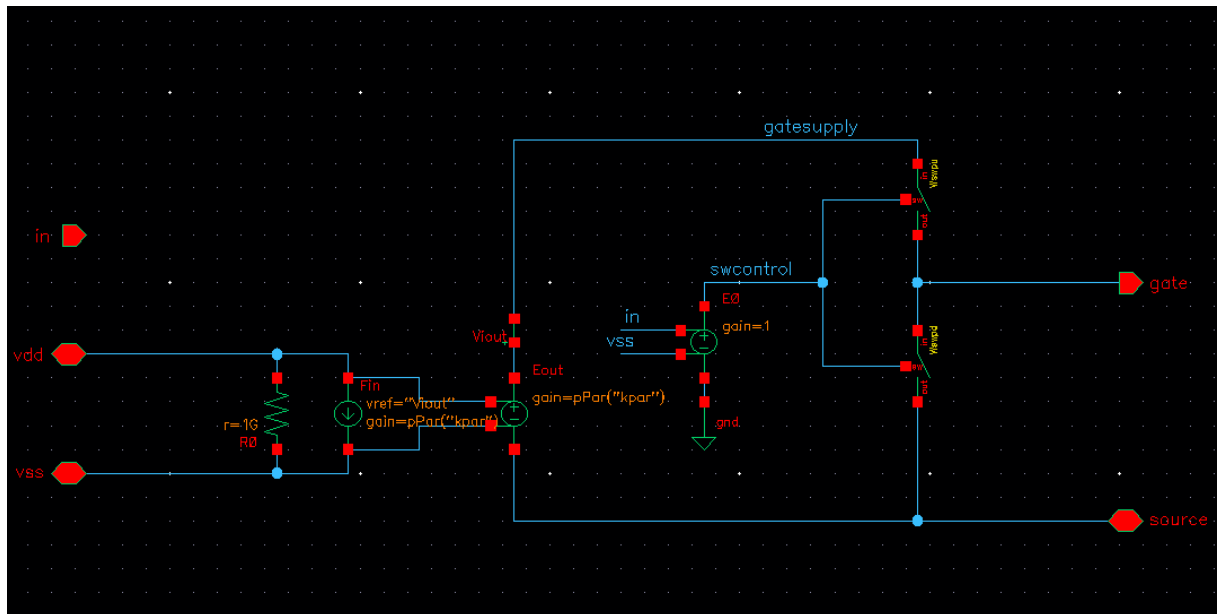


Figure B.3: Schematic of the ideal gate driver.