



University of Cyprus
Department of Electrical and
Computer Engineering

Design and implementation of a Dual-Resistor String Digital to Analogue Converter.

MSc Thesis Electrical Engineering

Petros Karousios

A Thesis submitted to the Department of Electrical and Computer Engineering and the school of Postgraduate studies of the University of Cyprus, in partial fulfilment of the requirements for the degree of Master of Science in Electrical Engineering.

Department of Electrical and Computer Engineering

Nicosia, January 2021

Petros Karousios



University of Cyprus
Department of Electrical and
Computer Engineering

Design and implementation of a Dual-Resistor String Digital to Analogue Converter.

Petros Karousios

Supervisor:

Prof. Julius Georgiou

University of Cyprus, Associate Professor, Department of Electrical & Computer Engineering

Examination Committee

Prof. Julius Georgiou

ECE, Professor, Examiner

Prof. Chrysostomos Nikopoulos

ECE, Professor, Examiner

Prof. Chrysafis Andreou

ECE, Professor, Examiner

Petros Karousios

.....
Petros Karousios

MSc Candidate Electrical Engineer, University of Cyprus

Abstract

Analogue to Digital and Digital to Analogue converters have been around for many years, enabling engineers to manipulate electrical signals. These devices are in almost all electronic devices from washing machines to audio amplifiers. For example, in a biochemistry analyser there is need to convert the analogue voltage from a photodiode into digital signal, in order to represent the result into numbers. In recent years, the constant scaling down of the fabrication technologies, have enabled engineers to improve converters' characteristics such as area, power consumption and latency. By improving converters' characteristics, more demanding problems can be tackled easier.

This thesis work aims to introduce to the reader the different types of architectures and their operation alongside with their advantages and disadvantages. Additionally, it focuses on the design, implementation and testing of an 8-bit Dual-Resistor String Digital to Analogue Converter. The converter was designed using Cadence Virtuoso software and was fabricated by using Taiwan Semiconductor Manufacturing Company (TSMC) 180nm technology with 1.8V supply voltage. The purpose of this converter is to drive high impedance loads such as MOSFET-based varistors or varactors of programmable Metasurfaces. After fabrication and testing, the overall results are INL and DNL less than 0.25 LSB with power consumption less than 42.5 μ W.

Acknowledgments

During my master's degree I received great support and counselling from a lot of people in my circle.

First, I want to thank my supervisor, Professor Julius Georgiou who through his expertise, he guided me not only for my thesis report but also expanded my critical thinking and directed me into being more meticulous.

Additionally, I would like to acknowledge my colleagues at Holistic Electronics Research Lab at the University of Cyprus for their fantastic assistance whenever I needed it and their constant brainstorming during every lunch!

I would also like to thank my family for their patience and the sympathetic ear. Lastly, I want to thank my lovely wife who is always there for me and I could not have finished my Master's degree without her giving me a little push to overcome any obstacles in my way.

List of publications

- L. Petrou, P. Karousios, and J. Georgiou, "Asynchronous Circuits as an Enabler of Scalable and Programmable Metasurfaces," IEEE Int. Symp. Circuits Syst., 2018.
- K. M. Kossifos et al., "Toward the Realization of a Programmable Metasurface Absorber Enabled by Custom Integrated Circuit Technology," IEEE Access, vol. 8, 2020.

Table of Contents

Figures	10
Tables.....	13
I. Introduction	14
Digital to Analogue Converter (DAC)	14
Analogue to Digital Converter (ADC)	15
Motivation.....	16
II. Digital to Analogue Converter (DAC)	18
Characteristics	18
Differential Nonlinearity - DNL	19
Integral Nonlinearity - INL	20
Offset	20
Gain Error	21
Latency	21
Signal-to-Noise Ratio – SNR	21
Dynamic Range - DR.....	21
Architectures	21
Resistor String.....	22
R-2R Ladder DAC.....	25
Current Steering DAC.....	26
Capacitive DAC architecture.....	29
Cyclic DAC.....	31
Pipeline DAC	32
III. Analogue to Digital Converter	35
Introduction	35
Characteristics	36
Differential Nonlinearity	37
Integral Nonlinearity - INL	38
Missing Codes.....	38
Offset Error.....	39
Gain Error	39
Aliasing	40
Signal-to-Noise Ratio - SNR	40
Mixed-Signal Layout Issues	40
Architectures	41
Flash ADC	41

The Two-Step Flash ADC	43
Pipeline ADC	44
Single-Slope Architecture	45
Dual-Slope Architecture	47
The Oversampling Analogue to Digital Converter	48
SAR ADC	50
Design of Asynchronous SAR ADC	53
Proportional to Absolute Temperature - PTAT	54
Successive Approximation Register (SAR) Logic	58
Ring Oscillator [1]	61
Bandgap Reference.....	62
IV. Design and implementation of an 8-bit Digital to Analogue Converter	71
Varactor and Varistor [45].....	71
Transmission Gate	78
CMOS Inverter	80
NAND Gate	87
Simulation Results of the Converter.....	93
Characterization of Digital to Analogue Converter	104
V. Conclusions	110
Comparing same architectures	110
Improvements	110
Future work.....	112
VI. References.....	113
VII. Appendix I: MATLAB code for calculating DAC's parameters	117

Figures

Figure 1: Digital to Analogue Converter Block [1]	14
Figure 2: Analogue to Digital Converter Block [1]	15
Figure 3: 8 Channel ADC with 16-Bit Resolution [2]	15
Figure 4: The architecture of a HyperSurface [3]	16
Figure 5: Top level diagram of the ASIC placed on the HyperSurface, showing the 8 DACs used. [4] .	17
Figure 6: The ideal transfer curve of a 3-bit resolution DAC with normalized Y-axis [1].....	19
Figure 7: Differential Nonlinearity representation of a 3-bit non-ideal converter [1]	19
Figure 8: Differential Nonlinearity example of a 3-bit Converter [1]	19
Figure 9: DAC's transfer curve showing INL calculation example [1] Figure 10: An example of an INL error of a 3-bit DAC [1]	20
Figure 11: An Offset Voltage example of a 3-bit converter [1]	20
Figure 12: A Gain Error example of a 3-bit converter [1]	21
Figure 13: (a) Resistor String Architecture (b) Resistor String Architecture using binary switches to reduce output capacitance [1]	22
Figure 14: Dual-Resistor String using isolation buffers with 6-bit resolution [9]	23
Figure 15: Resistor String DAC showing linear gradient effect [9]	24
Figure 16: Poly Resistor Layout Design [9].....	25
Figure 17: An N-bit, R-2R digital to analogue converter architecture [1].....	25
Figure 18: Dummy Switches used to preserve R-2R relationship [1]	26
Figure 19: (a) N-bit current steering DAC architecture. (b) an N-bit current steering DAC with binary-weighted current sources [1]	26
Figure 20: They cause of nonlinear errors due to parasitic capacitance [10].....	28
Figure 21: Current steering architecture using Differential Pair of MOS transistors [10]	29
Figure 22: The output impedance of a current steering architecture and its effects [10]	29
Figure 23: Simple Capacitive DAC architecture [1].....	29
Figure 24: Corresponding Capacitive DAC Architecture when input word is 100..00 and the resulting output voltage is $V_{ref}/2$ [1]	30
Figure 25: Common-centroid scheme for the binary-weighted capacitor array architecture [1]	30
Figure 26: Split Array method on charge scaling DAC to reduce capacitance area [1]	31
Figure 27: Cyclic DAC architecture [1]	31
Figure 28: Pipeline DAC architecture [1]	32
Figure 29: A sample and hold circuit [28]	35
Figure 30: Errors that can be caused from the S/H circuit [1]	36
Figure 31: The ideal transfer curve (staircase) of a 3-bit resolution ADC with normalized X-axis [1]..	36
Figure 32: The ideal transfer curve (staircase) of a 3-bit resolution ADC shifted by 0.5LSB and the corresponding Quantization error [1]	37
Figure 33: An example of a 3-bit ADC and its corresponding Quantization error showing the DNL [1]	38
Figure 34: An example of a 3-bit ADC having DNL error above 0.5 LSB resulting in Missing Codes. [3]	38
Figure 35: An example of a 3-bit ADC transfer curve with Offset Error [1]	39
Figure 36: An example of a 3-bit ADC transfer curve with Gain Error [1]	39
Figure 37: An example of aliasing due to undersampling [1]	40
Figure 38: Flash ADC [1].....	42
Figure 39: Two step flash ADC [1]	43
Figure 40: Pipeline ADC [1]	44
Figure 41: Single Slope architecture [1].....	45

Figure 42: Single slope architecture output versus clock cycles [1]	46
Figure 43: Dual slope architecture [1]	47
Figure 44: Output of the dual-slope architecture versus clock cycles [1]	47
Figure 45: Regular Converter vs oversampling converter [1]	48
Figure 46: Sampling frequency of ordinary converter vs oversampling converter [1].....	48
Figure 47: First order oversampling converter [1]	49
Figure 48: Frequency response of the oversampling converter at first order [1].....	49
Figure 49: Second order $\Sigma\Delta$ modulator [1]	50
Figure 50: Example of noise shaping when comparing 1st, 2nd and 3rd order $\Sigma\Delta$ modulators [1]	50
Figure 51: Example of an 8-bit SAR ADC [1].....	51
Figure 52: Output of a 3-bit SAR ADC [1].....	51
Figure 53: (a) Discharging the capacitor array and automatic offset cancellation.(b) the voltage at the top of the capacitor array (c) the equivalent circuit while converting the MSB [1].....	52
Figure 54: Proposed System Design for Asynchronous ADC.....	53
Figure 55: Proposed high-slope PTAT Temperature Sensor [33]	54
Figure 56: Simulated (a) current and (b) voltage versus temperature [33].....	54
Figure 57: Schematics of (a) BJT PTAT and (b) CMOS PTAT [34]	55
Figure 58: Results before and after the BJT PTAT calibration [34].....	55
Figure 59: (a) PTAT system from [35] (b) PTAT system. (c) PTAT voltage and current with small resistances. (d) PTAT voltage and current with double the size resistances.....	58
Figure 61: Block diagram for a 10-bit SAR [38]	59
Figure 60: Finite State Machine describing the SAR logic of a 10-bit ADC. We can observe the 12 clock cycles needed for the conversion [38]	59
Figure 62: Timing diagram of SAR logic [38]	60
Figure 63: Design of DFF using transmission gates [38]	60
Figure 64: Ring Oscillator delay [1].....	61
Figure 65: Schematic design of a five-stage ring oscillator [1].....	61
Figure 66: Schematic of the current-starved voltage control oscillator [1].....	62
Figure 67: Reference voltages of PTAT and CTAT [1]	63
Figure 68: PTAT generator [35]	64
Figure 69: Circuit implementation of a bipolar bandgap circuit proposed by Kujik [40]	65
Figure 70: CMOS bandgap circuit with start-up circuit [40]	66
Figure 71: Comparator symbol and its operation [1]	67
Figure 72: Three stages of the comparator [1]	67
Figure 73: Preamplification stage of the comparator comprised by differential amplifier with active loads [1]	68
Figure 74: Decision circuit with cross-gate connection as positive feedback decision circuit [1]	69
Figure 75: Self-biased differential amplifier used as output buffer [1]	70
Figure 76: Comparator with rail-to-rail functionality [1]	70
Figure 77: The MOS transistor cross-section. Solid lines represent the movement of charge carriers when the transistor operates in strong and moderate inversion regions. The dashed lines show the movement of electrons represent the parasitic resistance. [45].....	71
Figure 78: Capacitance relation to Bulk-Gate voltage [45].....	72
Figure 79: The curve of capacitance vs voltage of a PMOS varactor, operating at accumulation and depletion region [45].....	72
Figure 80: The PMOS varactor diode [45].....	73
Figure 81: RF Varactor symbol from Cadence IC Design Software.....	73

Figure 82: (a) Linear relation of drain current and Source/Drain voltage in deep triode region. (b) MOS transistor as a varistor [39]	74
Figure 83: Dual-Resistor String Digital to Analogue converter architecture. [43.44].....	77
Figure 84: Transmission Gate schematic and symbol [1]	79
Figure 85: CMOS Transmission Gate Layout and Schematic	80
Figure 86: CMOS Inverter schematic and symbol [1]	81
Figure 87: CMOS Inverter DC transfer characteristics [1].....	81
Figure 88: CMOS Inverter transfer characteristic showing the switching point [1]	82
Figure 89: Propagation delays of the CMOS Inverter [1].....	83
Figure 90: CMOS Inverter digital model showing switching characteristics [1].....	83
Figure 91: CMOS Inverter layout design methods [1]	84
Figure 92: Cross-section showing a CMOS Inverter. The parasitic bipolar transistors and resistors can be seen, responsible for latch up condition. [1]	84
Figure 93: Schematic of the parasitics of the CMOS Inverter that create latch-up condition [1].....	85
Figure 94: Schematic (a) of CMOS Inverter and Layout (b), used in the system, using Cadence IC Design Software	85
Figure 95: Simulated propagation delay of the CMOS Inverter from low to high	86
Figure 96: Simulated propagation delay of the CMOS Inverter from high to low	86
Figure 97: Simulated switching point of the CMOS Inverter	87
Figure 98: Test bench used in simulations.....	87
Figure 99: Schematic of the NAND Gate and its symbol [1]	88
Figure 100: (a) Schematic and (b) layout of the NAND gate.....	89
Figure 101: Test bench for the NAND gate with 3 inputs to find the switching point.	90
Figure 102: Simulated switching point of the 3-input NAND Gate	90
Figure 103: Simulated propagation delay of the 3-input NAND Gate from high to low	90
Figure 104: Propagation delay of the 3-input NAND Gate low to high	91
Figure 105: (a) Schematic and (b) Layout of the NAND gate with 4 inputs	91
Figure 106: Test bench for simulating 4-input NAND Gate	91
Figure 107: Simulated switching point of the 4-input NAND Gate	92
Figure 108: Simulated propagation delay from high to low of the 4-input NAND Gate	92
Figure 109: Simulated propagation delay of the 4-input NAND Gate from low to high	92
Figure 110: Graphic representation of 200 Monte Carlo simulations showing worst case INL error..	94
Figure 111: Graphic representation of 200 Monte Carlo simulations showing worst case of DNL error	94
Figure 112: Graphic representation of 200 Monte Carlo simulations showing the worst case of Gain error.....	94
Figure 113: Graphic representation of 200 Monte Carlo simulations showing the worst case of Offset Voltage error	95
Figure 114: Simulated Settling Time from high to low	95
Figure 115: Simulated Settling Time from bit=01111111 to bit=10000000	96
Figure 116: Monte Carlo simulations showing the analogue output at bit=01111111	96
Figure 117: Monte Carlo simulations showing the analogue output at bit=1000000	97
Figure 118: Simulated INL error	97
Figure 119: Simulated DNL error.....	98
Figure 120: Monte Carlo simulations showing the analogue output at bit=01111111	98
Figure 121: Monte Carlo simulations showing the analogue output at bit=10000000	98
Figure 122: Simulated rise time	99
Figure 123: Simulated fall time	99

Figure 124: (a) MSB resistor schematic and (b) MSB resistor's layout.....	99
Figure 125: (a) LSB resistor schematic (b) and (c) layout	100
Figure 126: Dummy resistors for (a) LSB and (b) MSB.....	100
Figure 127: Pull down and pull up resistors used for biasing always on TG switches.....	100
Figure 128: MSB part of the converter	101
Figure 129: LSB part of the Converter	101
Figure 130:(a) Ripple Counter Layout and (b) D-latch schematic.....	101
Figure 131: Converter's Layout	102
Figure 132: Position of the 8x Converters on the chip.	103
Figure 133: Block diagram for testing propagation delay for the varistor and the varactor [L. Petrou, personal communication, January 18, 2021]	103
Figure 134: Testing the converter. Using different Power Sources for Digital and Analogue circuits.....	104
Figure 135: Output of the converter using Ripple Counter at 1MHz frequency.....	105
Figure 136: Probe's impedance that was used to test the converter	105
Figure 137: Total current dissipated by the chip	106
Figure 138: Pin-out of the chip varactor [L. Petrou, personal communication, December 2020].....	107
Figure 139: Ideal (purple) vs measured (blue) output of the converter.....	107
Figure 140: Measured INL error after Gain and Offset errors are removed	108
Figure 141: Measured DNL error.....	109
Figure 142: Common Centroid technique for used for matching four resistors [1]	111

Tables

Table 1: Digital to Analogue converters summary	34
Table 2: Analogue to digital converters summary [31]	53
Table 3: Device sizes.....	56
Table 4: Transmission Gate device sizes.....	80
Table 5: Transmission Gate resistance characteristics	80
Table 6: CMOS Inverter device sizes	86
Table 7: CMOS Inverter characteristics	86
Table 8: NAND Gate truth table [47]	88
Table 9: NAND 3-input and 4-input device sizes	92
Table 10: NAND characteristics	93
Table 11: Simulated Settling Time.....	95
Table 12: Converters measured settling times	106
Table 13: Summary of measured results	109
Table 14: Comparison between converters implemented by using similar architectures	110

I. Introduction

Digital to Analogue Converter (DAC)

In an analogue world, Data Converters are crucial for processing data. Digital to Analogue Converters are frequently used in electronics for converting digital signal into analogue, such as media playback or converting digital instructions to analogue. The output of the DAC is an analogue signal. The main inputs of the converter are the input word, which is in digital form. Additionally, the converter needs a reference voltage as input [1].

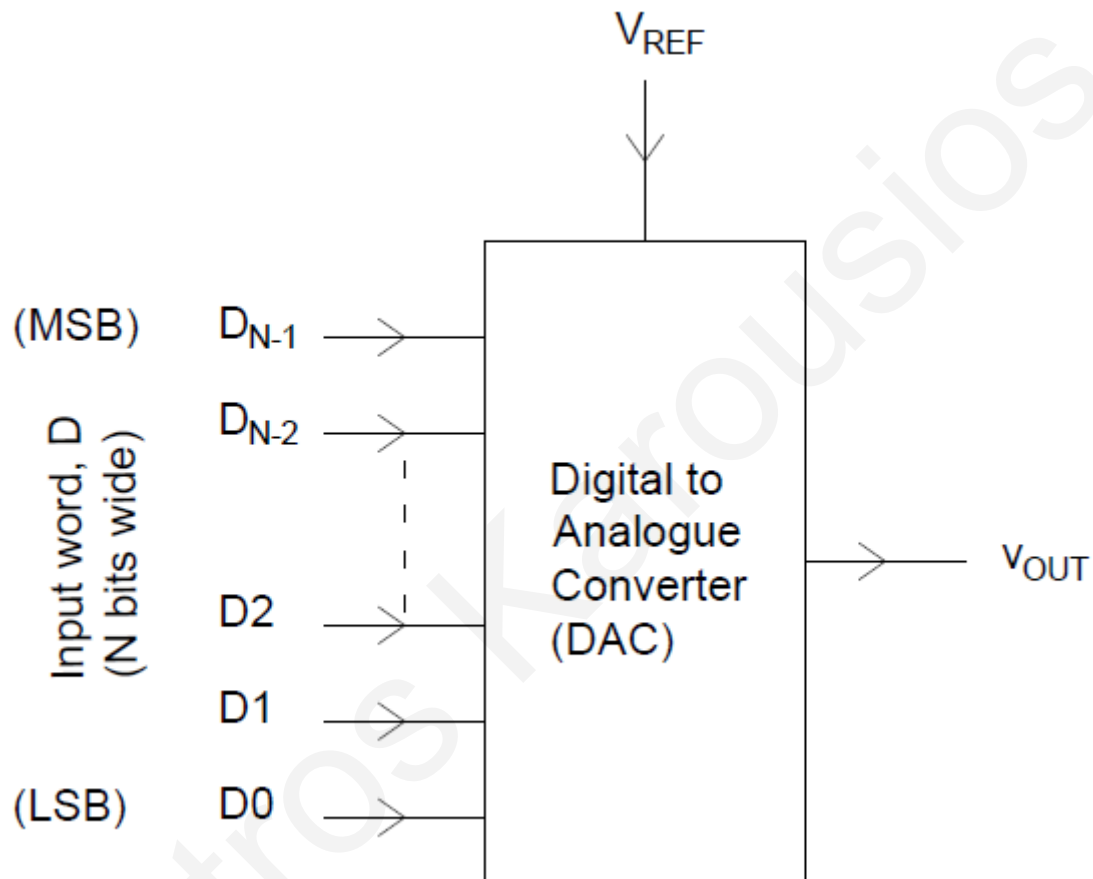


Figure 1: Digital to Analogue Converter Block [1]

Analogue to Digital Converter (ADC)

An ADC's purpose is to convert analogue signals to digital signals, or as the name describes, to digital signals. This is challenging because analogue signal can have an infinite number of values. With that being said, the converter must detect small changes in the analogue signal to convert them. In order to prevent aliasing, a Low Pass Filter (LPF) must be implemented to remove any noise or any other frequencies of higher order. A Sample and Hold system in many converter architectures is needed to sample and hold the value of the signal so the converter has sufficient time to complete the conversion. The converter has as an input an analogue signal, usually in voltage. Additionally, an input reference signal is needed. Figure 3 shows an 8-channel, 16-bit resolution ADC from Control Everything [2].

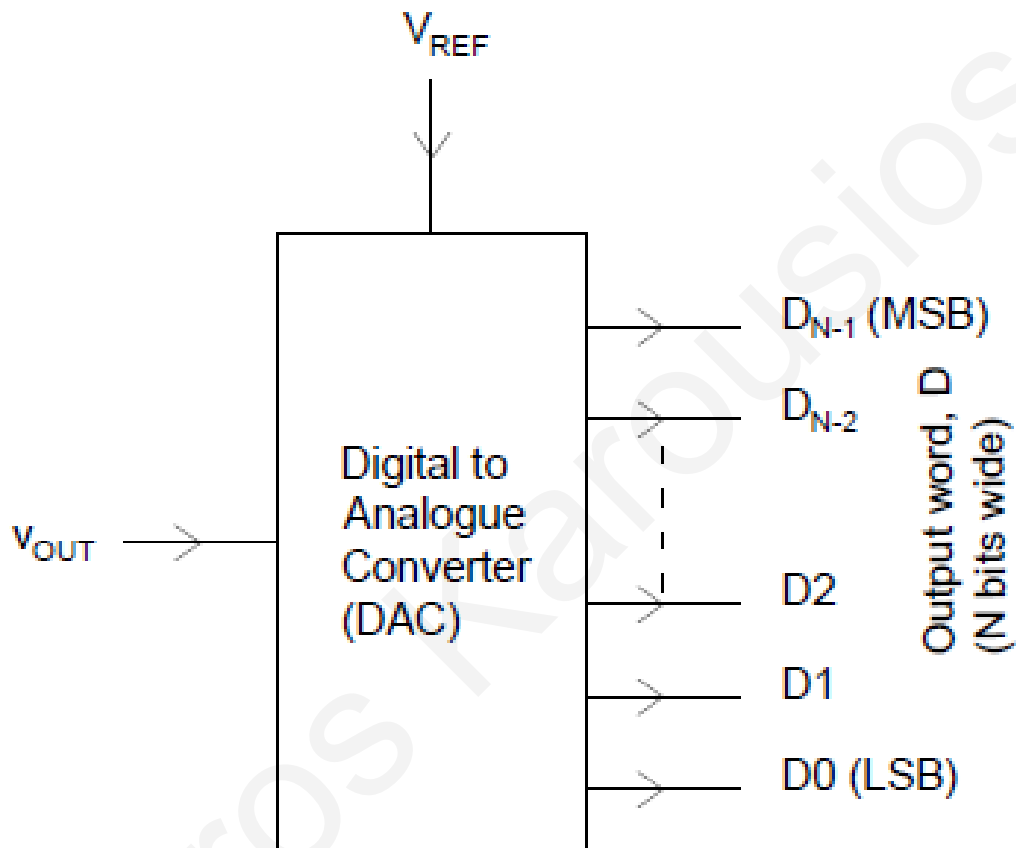


Figure 2: Analogue to Digital Converter Block [1]

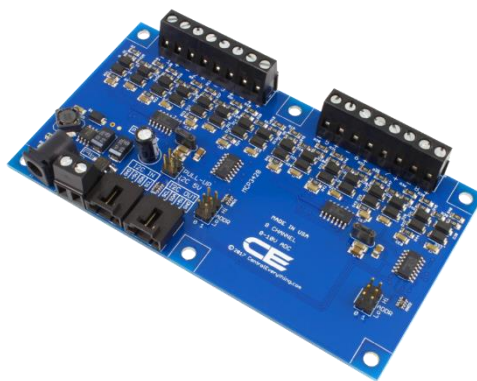


Figure 3: 8 Channel ADC with 16-Bit Resolution [2]

Motivation

This MSc Thesis is part of the coursework for completion of the Electrical Engineering Master's program of University of Cyprus. It is part of the European Funded FETOPEN project called "VISORSURF: A Hardware Platform for Software-driven Functional Metasurfaces". Its main objective is to manipulate electromagnetic (EM) waves through planar smart devices called HyperSurfaces, that can be interconnected and programmed to alter the EM behaviour of the incident EM waves. This objective can be achieved by using man-made structures called Metasurfaces. These structures have specific EM properties depending on their internal structure. The Metasurfaces change their structure by receiving commands from a Gateway outside the HyperSurfaces changing their EM properties [3].

A HyperSurface is comprised by tiles and each tile has a number of integrated circuits that controls the metamaterials. Each chip is asynchronous and uses handshake protocols to communicate from chip to chip, hence the low power dissipation of asynchronous circuits. This means that a clock is undesirable, due to large area that a crystal oscillator requires and due to high EM noise generation of the clock tree and lastly, as mentioned above, due to power dissipation.

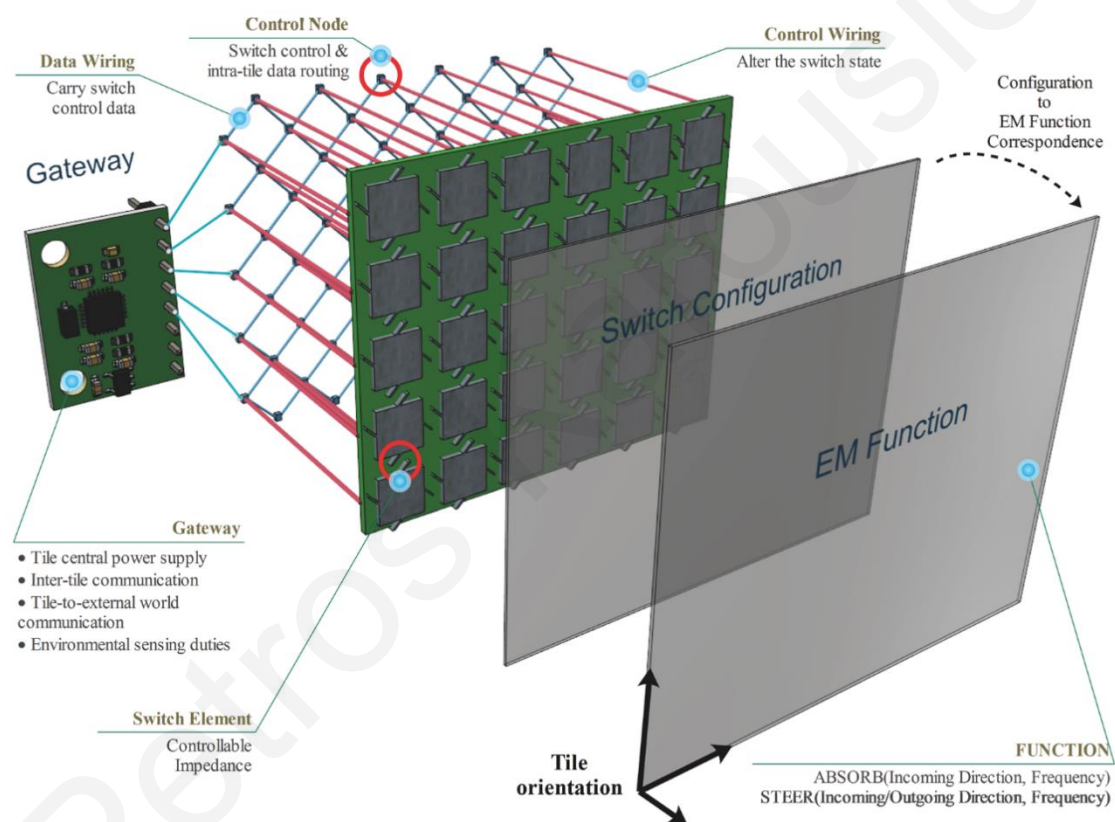


Figure 4: The architecture of a HyperSurface [3]

The digital signal sent from the gateway, needs to be converted into analogue voltages at each meta-atom. The analogue signal is then applied to the gate of a varistor or varactor, thus altering their resistance and capacitance respectively. By making this alteration, the Metasurface's properties change thus altering the angle of reflection of the incoming EM signal [4][5]. Additionally, by driving high impedance nodes makes the problem easier since the signal needs no buffering and power consumption is less. The following figure shows the position of the 8x 8-bit DACs inside the ASIC that is placed over one thousand times on a HyperSurface that has an area one square foot.

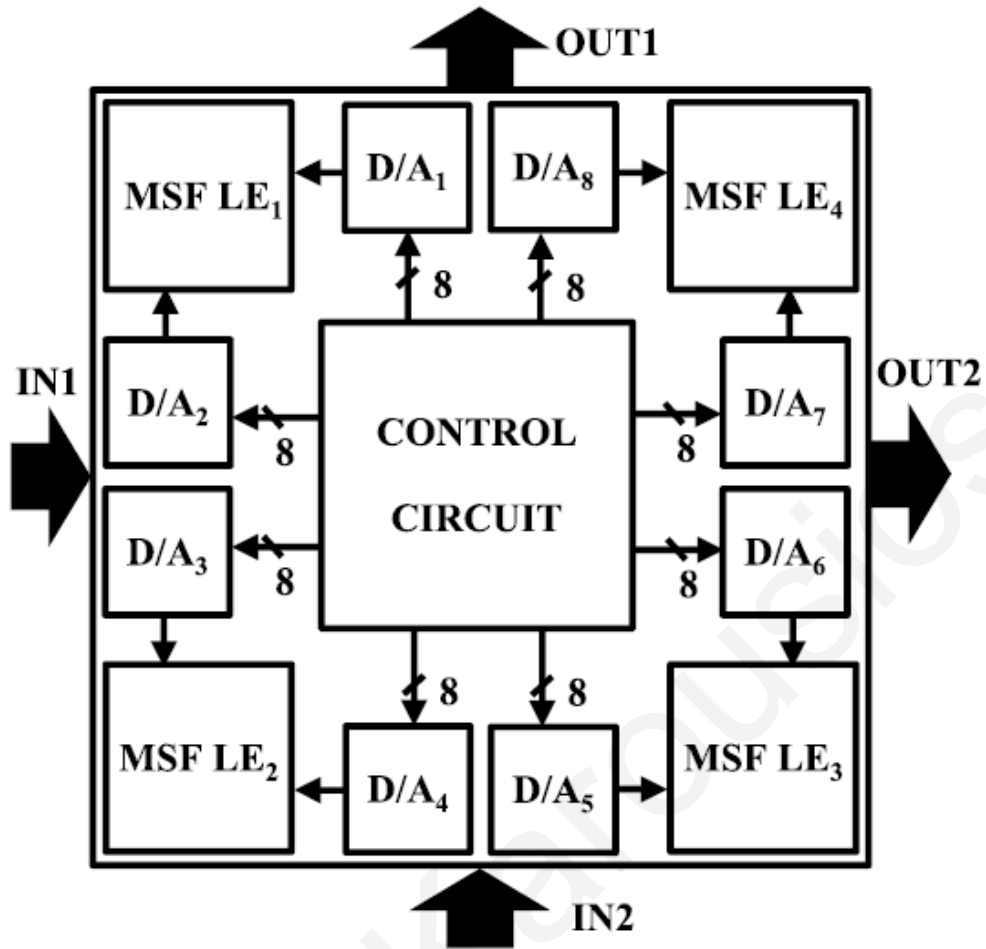


Figure 5: Top level diagram of the ASIC placed on the HyperSurface, showing the 8 DACs used. [4]

II. Digital to Analogue Converter (DAC)

Characteristics

Each DAC architecture has specific limitations. However, in order to be able to compare each DAC and later ADC, their specifications must be explained.

A DAC's output is a fraction of a reference voltage or current,

$$v_{OUT} = F \times V_{REF}$$

Where v_{OUT} is the converter's output, V_{REF} is the reference voltage of the converter and F is the fraction. The fraction F can be calculated by:

$$F = \frac{D}{2^N}$$

Where D is the digital input word converted to decimal and 2^N is the total number of combinations from the resolution N of the Converter.

An easy way to represent the output of a DAC, is to plot all the input words D versus the analogue output voltage v_{OUT} in a transfer curve. Then the y-axis can be normalised in order to represent the fraction F by using the above equation. The highest value of the input word D represented in the figure below for a 3-bit DAC is D=111 because the starting word is D=000. However, D=111 represents the $v_{OUTMAX} = \frac{7}{8}V_{REF}$ where V_{REF} is the total voltage but not the voltage output of the DAC, which is denoted as V_{FS} , Full Scale Voltage. It represents the full-scale voltage of the DAC, which can be calculated by:

$$V_{FS} = \frac{2^N - 1}{2^N} V_{REF}$$

The least significant bit is the smallest analogue output that can be represented. D_0 represents the LSB and can be calculated by:

$$1 \text{ LSB} = \frac{V_{REF}}{2^N}$$

In the case of a 5V reference and 3-bit DAC, LSB is equal to 0.625V, which is the level of quantization. This describes the resolution of the system because it denotes the smallest step that can be made, and it depends on the specifications of each application. For high resolution applications, the least significant bit LSB requires to be small, meaning that a DAC must be implemented by higher number of bits, N. As a result, when the number of bits is higher, the analogue representation of the LSB becomes smaller. For example, in an application that a converter will control a MOS varistor, we want to control the voltage that is applied to the gate of a MOSFET. The lowest voltage that can be applied is 0V and the highest, for example used in this thesis, 1.8V which is our V_{REF} . Therefore, the range is 1.8V. Thus, by using a DAC with 2^3 input combinations, the resulting resolution is $\frac{1.8}{8} = 0.225V$. However, a converter can never go as high as the Full Scale Voltage, because, in a 3-bit DAC, the word D=111 can output $7 \times 0.225 = 1.575 V$, as can be seen in fig. 6.

Two main ways to describe a DAC are Differential and Integral Nonlinearity that are explained in the following pages. These static characteristics are used to describe the performance of a DAC [1].

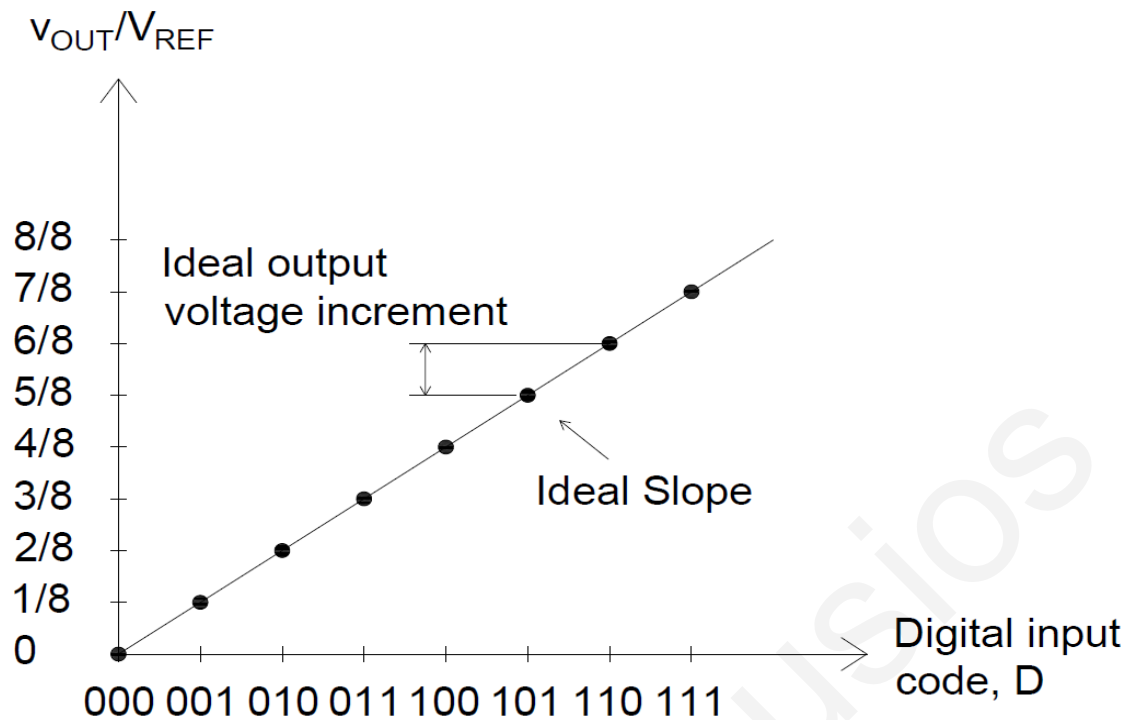


Figure 6: The ideal transfer curve of a 3-bit resolution DAC with normalized Y-axis [1]

Differential Nonlinearity - DNL

Differential Nonlinearity is a characteristic showing the DAC's ability to convert each increment of the digital word by a step of the same value, known as LSB. It's the difference of the ideal step value of a DAC to the actual step value and it is caused by nonideal components mainly due to mismatches. When a DAC has a DNL error larger than an $\pm 0.5\text{LSB}$ its resolution drops because it's no longer accurate. In addition, with DNL error larger than 1 LSB, the DAC loses its monotonicity and becomes nonmonotonic, meaning that the output voltage won't necessarily increase as the digital code increases [1].

$$DNL_n = \text{Actual step value } n - \text{Ideal value}$$

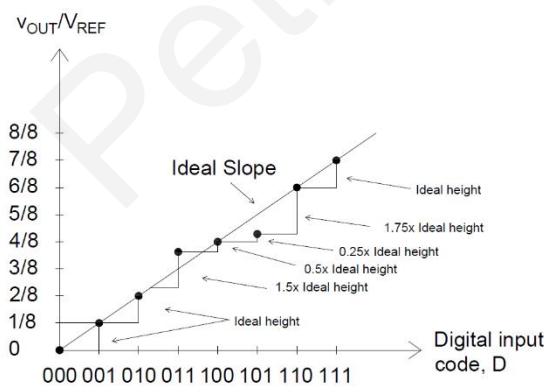


Figure 8: Differential Nonlinearity example of a 3-bit Converter [1]

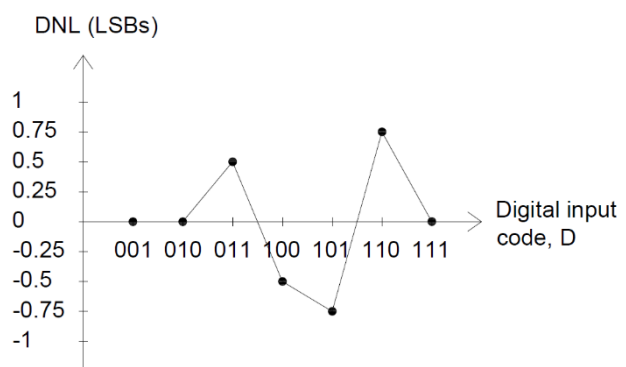


Figure 7: Differential Nonlinearity representation of a 3-bit non-ideal converter [1]

Integral Nonlinearity - INL

In order to fully visualize Integral Nonlinearity, a reference line must be drawn through the first and last output analogue values of the DAC. Then the INL can be calculated by:

$$INL_n = \text{Analogue output value for specific input} - \text{Analogue value of the reference line at that specific input}$$

For a proper DAC, as mentioned above, both DNL and INL must be below ± 0.5 LSB. This applies also to ADCs. In both cases, it denotes the maximum error of a data converter, analogue or digital. In case DNL or INL exceed ± 0.5 LSB, automatically the N-bit resolution decreases. It is worth mentioning that there are also other techniques to measure the INL that are also acceptable in the engineering community. One method is to compare the actual values with an ideal reference line but will include the offset voltage or gain error. The other method is to create a “best-fit” reference line that will minimize the INL and it is a subjective method [1].

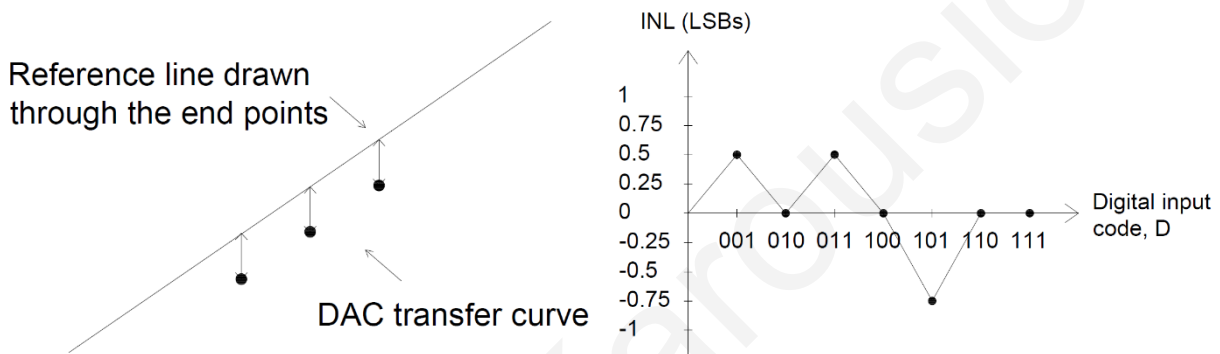


Figure 9: DAC's transfer curve showing INL calculation example [1] Figure 10: An example of an INL error of a 3-bit DAC [1]

Offset

Offset can be caused due to nonlinearities of the DAC's components. If the output voltage is not 0V at the word D=0, this means there is voltage offset or shift in the DAC's transfer curve [1].

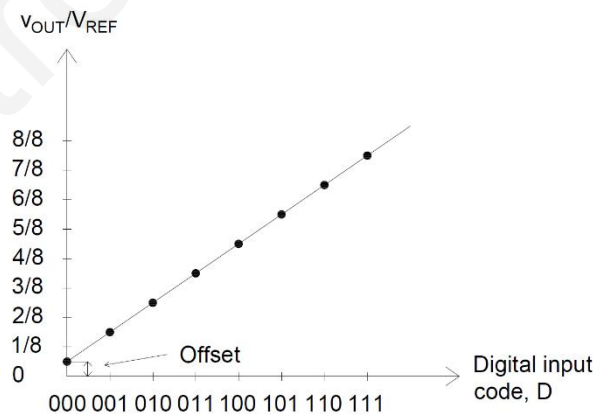


Figure 11: An Offset Voltage example of a 3-bit converter [1]

Gain Error

The Gain Error can be described as the difference between the ideal slope versus the actual slope [1].

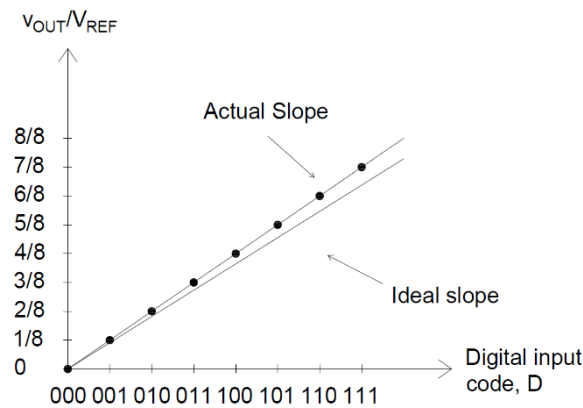


Figure 12: A Gain Error example of a 3-bit converter [1]

Latency

Latency is the time required from the moment the input word is changed to the moment the output voltage is settled is the latency, and it contains the settling time, which is the time required for the output to reach its value that is accepted by the requirements.

Signal-to-Noise Ratio – SNR

It is widely known that SNR represents the ratio of the Signal Power over the Noise Power at DAC's output. The test bench to calculate the SNR is to program the input bits in that manner as their output will represent a digital sinewave.

Dynamic Range - DR

Expresses the ratio between the output signals of highest and lowest value, and it is related to the resolution. For both DACs and ADCs, DR is approx. $6.02 \times N$ dB. For a DAC with its maximum output signal $D = 2^N - 1$ and minimum output signal $D = 1$, DR can be found by [1]:

$$DR = 20 \log \left(\frac{2^N - 1}{1} \right)$$

Settling Time

Is the total time it takes the output to respond to a step change of input. The input and output are at 0.1% tolerance [6].

Types of codes

There is several cases that the input code is not in binary form and can be in either thermometer code, gray code, two's compliment, BCD and other code, as can be seen in the following figure. If for example the engineer for some reason wants to change only one bit when going from one position to the next, thermometer code can be used. The type of code used is chosen by the engineer based on the application [1].

Architectures

The fast and major advances in engineering not only in devices that we use in our everyday life but also in industry and military, have increased the demand in use of Digital to Analogue Converters, which vary in speed, energy consumption and area. Their use in devices that can be found not only in the IoT but also in autonomous cars and in many other engineering fields that require electronics, has

distinguished Digital to Analogue Converters as an essential system. However, engineers around the world must examine thoroughly the different types of architecture and find the most suitable for their project [1].

Resistor String

This architecture is widely used due to its simplicity. It uses 2^N resistors of same value in series creating a string and 2^N identical switches. The analogue output is a result of voltage divider when the proper switch opens. A main drawback of this architecture is that it needs a decoder $N:2^N$ for the switches to be controlled. This architecture is best fit in applications where output current is not a requirement. In addition, good matching of the resistors is required for the output to be inside the chosen limits, thus ensuring monotonicity.

Resistor String DACs face two main issues: The first issue is parasitic capacitance due to the large number of OFF switches and just one on, and the second issue is the trade-off between energy consumption versus area.

The first problem mentioned above happens when the resolution is large, meaning that higher number of devices is used. Most of the switches are OFF, leaving just one switch on to transfer the voltage at the output. Then parasitic capacitance is gathered at the output node resulting in slower conversion. A solution proposed in [1], is to use a binary switch array, as can be seen in the following figure. The binary array will assist in the reduction of the parasitic capacitance because, instead of $2^N - 1$ number of OFF switches, now N OFF switches are connected to the output and N on.

The second problem mentioned above, challenges engineers because for high resolutions, large number of resistors and switches must be used thus area is increased. The tradeoff that must be taken into consideration is that, the resistors can be made smaller sacrificing power dissipation. Other passive components can be used, such as the n-well resistor which results in low resolution for the system.

Resistor matching is necessary for this type of DAC to have output within the required range. If resistors are not matched properly, this can lead to error in accuracy, INL and DNL error [1].

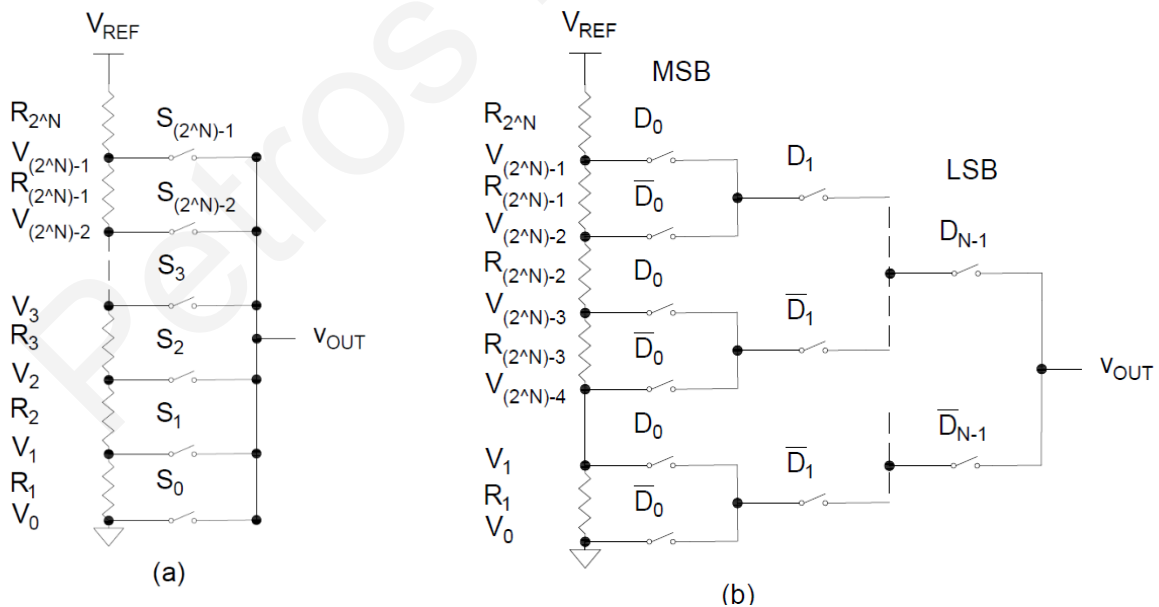


Figure 13: (a) Resistor String Architecture (b) Resistor String Architecture using binary switches to reduce output capacitance [1]

Yet another solution is to use two resistor strings, in a topology called dual-resistor string. This requires the use of unity gain buffers in order to isolate the two resistor strings and can be seen in the figure below. There is a reduction in the resistors used in this topology thus reducing the area and instead of 2^N resistors now $2 \times 2^{N/2}$ are used whilst the monotonicity is still achieved. However, the challenge moves to the amplifier design, which must not add any offset voltage and must be fast [7][8].

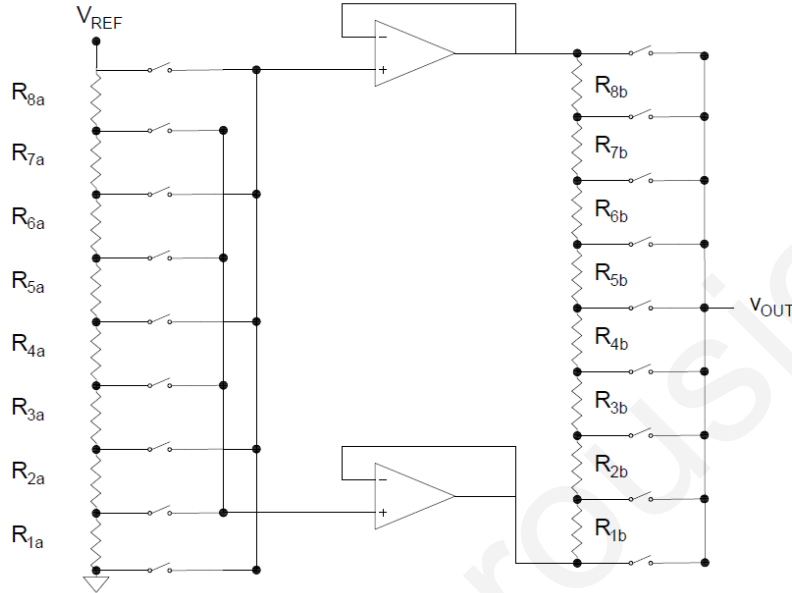


Figure 14: Dual-Resistor String using isolation buffers with 6-bit resolution [9]

INL and DNL related errors due to resistor mismatches

Resistor matching is a critical issue that may cause INL and DNL errors. By modeling the real resistor and not the ideal, total resistance is $R_i = R + \Delta R_i$, where R is the nominal resistance value and ΔR is the mismatch error. By assuming that the mismatch error is zero, the voltage of each node can be found by $V_{i,ideal} = \frac{(i)V_{REF}}{2^N}$, for $i=0,1,2,\dots,2^N - 1$. However, in real life mismatch error is never zero.

The mismatch error can be eliminated early in the design process, as can be seen later, by using time consuming Monte Carlo simulations. In [9], the authors found a way analytically model resistor's mismatch error by using MATLABTM, by using the resistor's geometry and statistical data of a 0.18 μ m technology design. The need for this modeling is due to the fact that resistor string DAC can be used in medium to high speed applications. In addition, it has reduced area and it is power efficient[1].

There are two possible causes of errors in a resistor string DAC. As stated in [9], one cause is the linear gradient effect due to voltage differences between the resistors and the other is random variations that can have a negative effect on the resistor's value. Engineers have found a way to reduce the DNL error by using a folded resistor string.

As can be seen in fig.15, the linear doping gradient is created due to variations to each node's voltage. INL error concludes to:

$$INL_j = \frac{j}{2^n} V_{REF} - \frac{jR + \frac{j(j-1)}{2} \Delta R}{2^n R + \frac{2^n(2^n-1)}{2} \Delta R} V_{REF}$$

and the maximum INL happens when $j = \frac{2^n}{2}$. This means that the worst-case scenario happens when the input word has MSB=1 and the rest of its bits equal to zero, meaning that half of the resistors are active, and the other half are not.

In addition to the INL error mentioned above, the DNL error concludes to:

$$DNL_j \approx \left(j - \frac{2^n - 1}{2}\right) \frac{\Delta R V_{REF}}{R 2^n}$$

and the maximum DNL happens when $j = 1$ or $j = 2^n - 1$ because ΔR is at its highest value, resulting in $DNL = V_{REF} \frac{\Delta R}{2R}$.

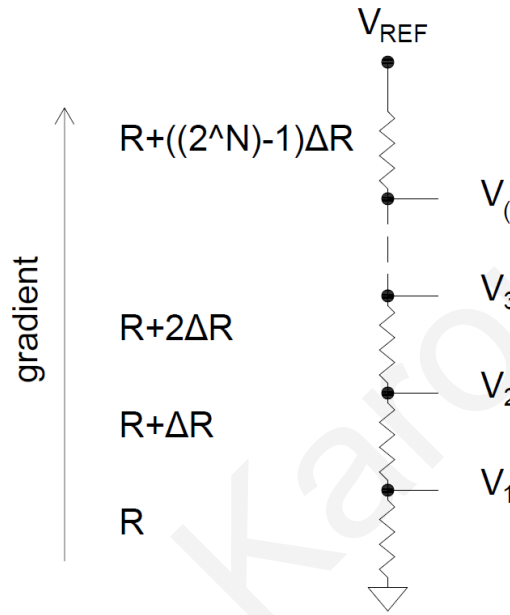


Figure 15: Resistor String DAC showing linear gradient effect [9]

The INL and DNL errors that occur due to random variations that happen during the photolithographic process and are related to the actual structure of the resistor, as can be seen in fig. 16. The ideal equation for a polysilicon resistor that describes fig. 16 is

$$R = \rho \frac{L}{Wt} + 2R_c$$

and it is widely known that L , W and t are length, width and thickness respectfully. In addition, ρ is, in this case polysilicon's resistivity and due to integrated resistors, R_c is the contact resistance. Due to photolithography errors, identical resistors cannot be made. Some of the possible mismatch errors might lead to resistivity errors, errors in geometry and random process variations, all resulting into altering the resistor's actual resistance value. By differentiating the above equation results in

$$\frac{\Delta R}{R} = \frac{\Delta \rho}{\rho} + \frac{\Delta L}{L} - \frac{\Delta W}{W} - \frac{\Delta t}{t} + 2 \frac{\Delta R_c}{R}$$

By inspecting the above equation, the numerator values cannot be controlled by the designer and only the denominator values can be controlled. A simple solution is to increase the geometry of the resistor by choosing large values of Length and Width, resulting in unwanted negative effects such as parasitic resistance between resistors and substrate. Another consideration is the large increase in area due to bigger geometry of the resistors.

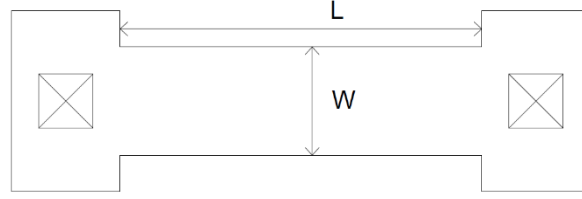


Figure 16: Poly Resistor Layout Design [9]

The authors in [9] created a 10 bit DAC with total area of 0.12mm^2 , $\text{INL} \leq 0.7\text{LSB}$ and $\text{DNL} \leq 0.87\text{LSB}$ using 180 nm technology. However, their work aimed to prove that by using sophisticated modeling of the resistors in MATLAB, the authors saved time versus the time-consuming Monte Carlo simulations.

R-2R Ladder DAC

This architecture that incorporates resistors of R and $2 \times R$ resistor values, uses less resistors than the Resistor String architecture and can be seen in the figure below. In this figure we can observe that the resistor connected to the switch, controlled by the digital input word has a value of $2R$ and this digital input word decides if the resistor is switched to the inverting input of the op-amp or to the ground. This voltage divider causes each node's voltage to be a binary-weighted fraction of the reference voltage. Since the switches connect the nodes either to ground or virtual ground, the total current running through each node is constant, resulting in constant voltage at each node [1].

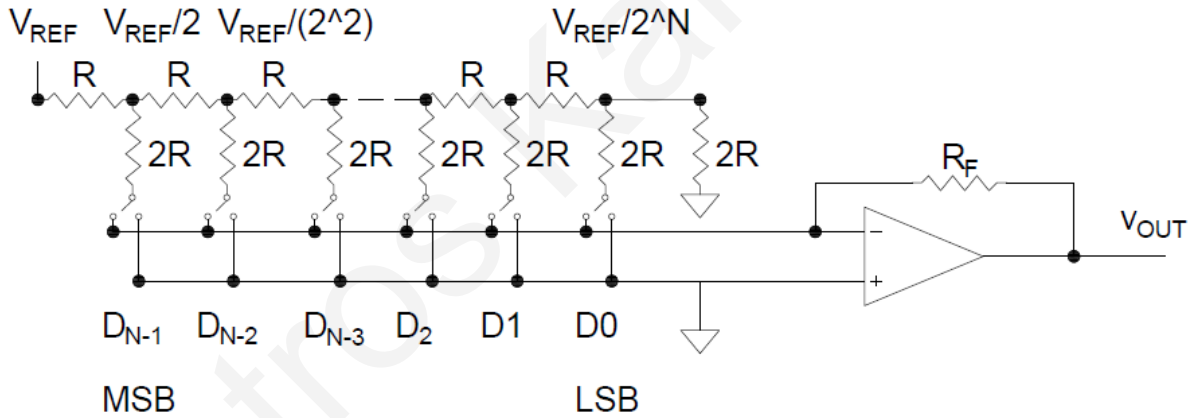


Figure 17: An N-bit, R-2R digital to analogue converter architecture [1]

The total current flows through the feedback resistor R_F and the corresponding voltage output is

$$v_{OUT} = -i_{TOT} \cdot R_F$$

The total current i_{TOT} running through the resistor depends on the input word, as can be seen from the figure above. The digital input word is responsible for transferring the voltage either to the inverting or the non-inverting input of the op-amp, meaning that each bit of the digital input word if it has a value of 1, node's voltage is transferred to the output. The total current transferred to the inverting input of the op-amp depends on the input bits and can be found by:

$$i_{TOT} = \sum_{k=0}^{N-1} D_k \cdot \frac{V_{REF}}{2^{N-k}} \cdot \frac{1}{2R}$$

One of the disadvantages of this architecture and every architecture that uses passive components is the required matching that is needed, like the Resistor String DAC architecture mentioned above. Current passes through digital switches that require to have negligible resistance otherwise the voltage divider will take into consideration this voltage drop. In order to eliminate this voltage drop is by using dummy switches with resistance half the value of the switches at the $2R$ node, next to resistors with R value. This way, the total resistance of the R node is now twice as the value of the $2R$ node. Another addition to the circuit is the dummy switch at the terminating resistor with resistance equal to that of the $2R$ node, in order to preserve the $R' - 2R'$ proportion [1].

$$R' = R + \frac{\Delta R}{2}$$

$$R = 2R + \Delta R$$

Dummy switch (always on)

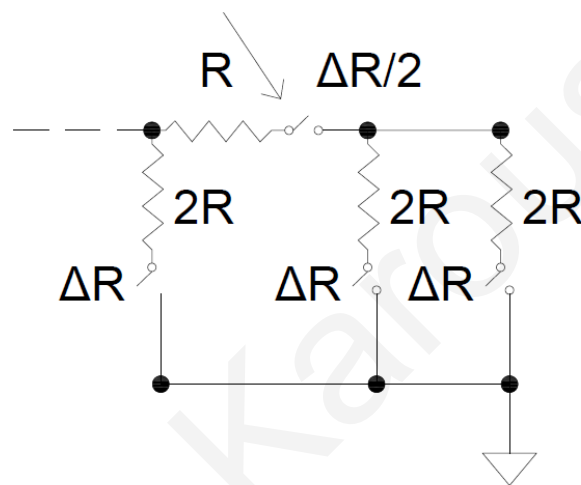


Figure 18: Dummy Switches used to preserve R - $2R$ relationship [1]

Current Steering DAC

This type of DAC architecture is comprised by current sources which sum up the current. By saying that, these current sources must be precise otherwise DNL and INL errors might occur.

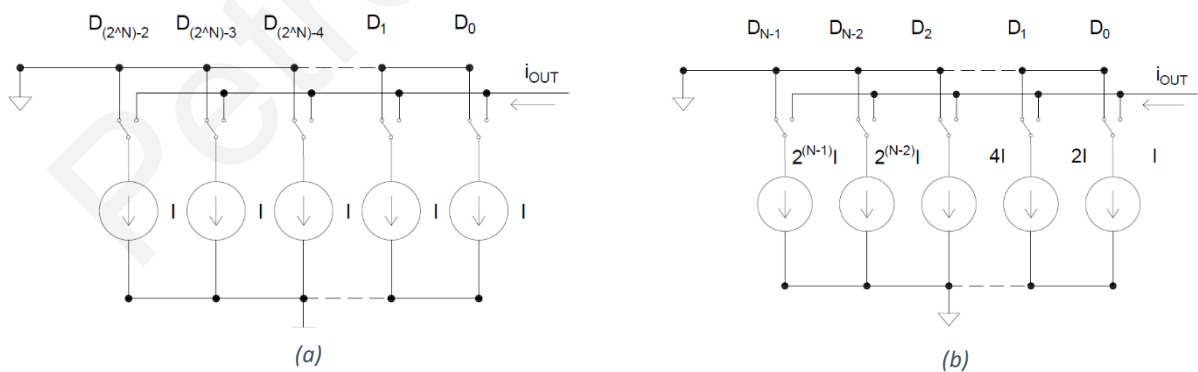


Figure 19: (a) N -bit current steering DAC architecture. (b) an N -bit current steering DAC with binary-weighted current sources [1]

Fig. 19a, shows a generic current steering DAC architecture that uses thermometer code, thus a thermometer encoder is used to convert binary data. Thermometer code chooses whether a current source is connected to the output or to another node. This architecture uses $2^N - 1$ current sources and each current source is identical to the other resulting in total output current in the range of

$$0 \leq i_{OUT} \leq (2^N - 1) \cdot I$$

In order to eliminate the need for a thermometer encoder, instead of using identical current sources some architectures use binary weighted current sources thus, instead of $2^N - 1$, only N devices will be used, and they have different sizes, as can be seen in fig. 19b [1].

There can be various advantages by using this type of architecture. Firstly, since that its output is a current, output buffers are eliminated from this circuit because are not required to drive resistive loads. As a result, current steering DACs have high speed and are used in high-speed applications like communication transmitters and consumer electronics. Another use of this architecture is in gigahertz frequency applications where transmission lines must be driven by the DAC [10].

As stated in [1], a drawback of this architecture is that requires great matching between the current sources otherwise errors might occur. For instance, in our case that we are using an 8-bit DAC, we should use $2^N - 1$ current sources resulting in total 255 devices. However, when choosing the binary weighted architecture and the current of the smallest device to be $10\mu A$, the current of the largest device should have been larger by a factor of 2^{N-1} or 128 meaning that the largest current source would be 1.28mA.

Another flaw of this architecture that produces a glitch, occurs when a switch is turned on and another switch is turned off. If the on or off switching happens simultaneously, this glitch can be seen at the output of the DAC.

The error due to mismatch is assumed to happen at the middle of the staircase, can be calculated by taking into consideration that half of the current sources produce the maximum positive mismatch and the other half the maximum negative mismatch. This concludes to:

$$I_{out} = \sum_{k=1}^{2^{N-1}} (I + \Delta I_k) = 2^{N-1} \cdot I + 2^{N-1} |\Delta I|_{max} = I_{out,ideal} + 2^{N-1} \cdot |\Delta I|_{max}$$

where I_{out} is the total output current, I is the ideal current, ΔI_k is the error in current due to mismatch and $|\Delta I_{max}|$ is the maximum positive or negative current due to mismatch. As mentioned in previous section, INL is the actual output voltage minus the ideal. In this case, it translates to the ideal current subtracted from the actual output current and the worst case of INL happens, as mentioned above, at the midscale, and concludes to

$$|INL|_{max} = 2^{N-1} \cdot |\Delta I|_{max,INL}$$

The above equation represents the INL error for the DAC architecture in fig. 19a, where each current source represents one bit or 1 LSB. The requirements of a DAC, demand that INL error is less than 0.5LSB meaning that the maximum tolerance for error is half an LSB for each current source, or $0.5I$, meaning that due to mismatches, the following equation is the maximum current error results in

$$|\Delta I|_{max,INL} = \frac{0.5I}{2^{N-1}} = \frac{I}{2^N}$$

and explains why this architecture is difficult to be used for high resolution applications. For example, if each current source has output current of $10\mu A$ and the input bits are set to 8, like in our case, then the maximum current source mismatch error required to keep the INL inside the required boundaries

will result to $|\Delta I|_{max,INL} = 19.5 \text{ nA}$. This result requires that for each current source, the current error must be $9.9805 \mu A \leq I_k \leq 10.0195 \mu A$.

Regarding the DNL error, can be easily derived from the transfer curve, by subtracting the output current of the previous value from the next value. The maximum DNL error can be found by subtracting any two consecutive current sources that have the largest error, resulting in

$$I_{out(x)} - I_{out(x-1)} = I_k + |\Delta I|_{max,DNL}$$

Having the requirement that the maximum DNL must be less than 0.5LSB, simplifies things resulting in $|\Delta I|_{max,DNL} = 0.5I$

The binary-weighted architecture in fig. 19b, an INL and DNL can be determine with a different approach. Due to the binary-weighted current sources, an assumption must be made that the current source responsible for the MSB has the highest positive mismatch error and the rest of the current sources have the negative mismatch error resulting in zero errors when they are added together. Therefore, the maximum INL error due to current mismatches results in

$$|INL|_{max} = 2^{N-1}(I + |\Delta I|_{max,INL}) - 2^{N-1} \cdot I = 2^{N-1} \cdot |\Delta I|_{max,INL}$$

Regarding the maximum DNL, like the other DAC architecture cases, happens when the input word changes from 0111..1 to 1000..0. As can be seen by the following equation, the boundaries for the maximum DNL error are more strict than the maximum INL error, and can be found by [1]

$$|\Delta I|_{max,DNL} = \frac{I}{2^{N+1} - 2}$$

In [10], Razavi describes the circuit in fig. 19b as a “simple binary-weighted current-switching DAC”. This architecture has dynamic errors due to the switches. When a switch is turned off, the node’s voltage turns to zero its capacitance as can be seen in the following picture turns also to zero. However, the next time the switch turns on, the capacitance charges up, resulting in nonlinear changes in the output current because the capacitor draws current. In addition, the periodical switching on and off of the switches causes the total output current to change. As a result, parasitic series inductance due to bond wires is introduced thus ground voltage does not remain constant.

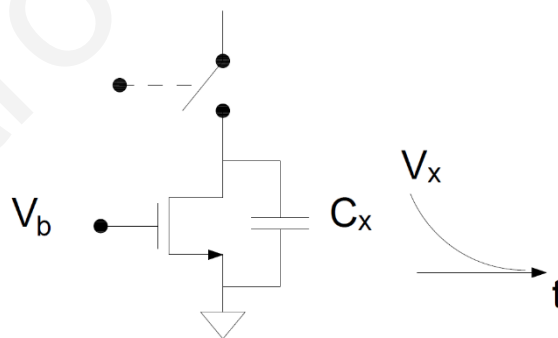


Figure 20: They cause of nonlinear errors due to parasitic capacitance [10]

The introduction of differential pair is introduced in this article in order to suppress both problems mentioned above. The differential pair draws less current to charge up its parasitic capacitance therefore there is small fluctuation at the output current. The problem regarding parasitic series inductance is suppressed because the total current of the array is constant. The main advantage of using differential pairs is the differential outputs that are introduced to the circuit.

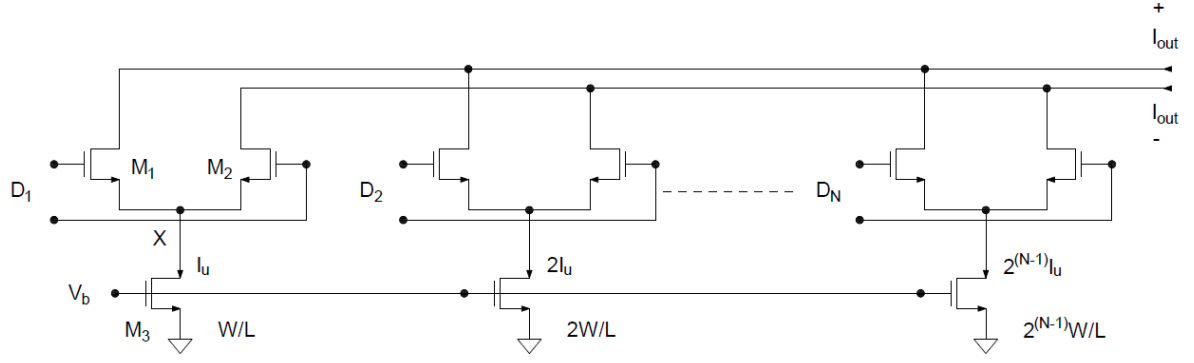


Figure 21: Current steering architecture using Differential Pair of MOS transistors [10]

Other static problems regarding the current steering architecture are the voltage drop due to long ground line and the output resistance of each current source. The first problem is created due to long ground lines, which in some cases, alter the voltage and instead of being zero can reach tens of millivolts. The other problem is related to the output resistance of the current sources and can be really be taken into consideration especially if the DAC will drive resistive loads. As can be seen by the following picture, each cell's resistance is $R_L || r_o$ and when all the cells are used, the total resistance results into $R_L || (r_o/M)$. In this case the INL can be found by:

$$\frac{MR_L}{4 \cdot r_o}$$

This means that for an $R_L = 50 \Omega$ and $M=256$, means that r_o must be at least $3.2 \text{ M}\Omega$ in order for the INL to remain under 0.1% .

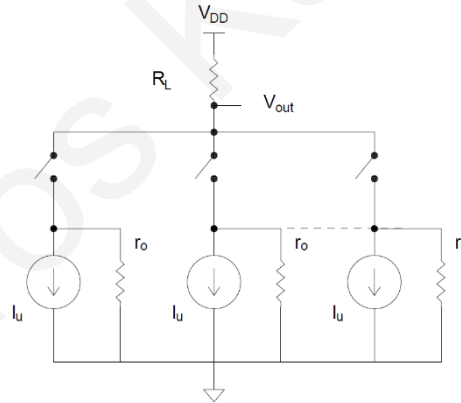


Figure 22: The output impedance of a current steering architecture and its effects [10]

Capacitive DAC architecture

The capacitive DAC architecture, also known as Charge-Scaling DACs are widely used in CMOS technology. Binary-weighted capacitors are used in parallel, as can be seen in fig. 23.

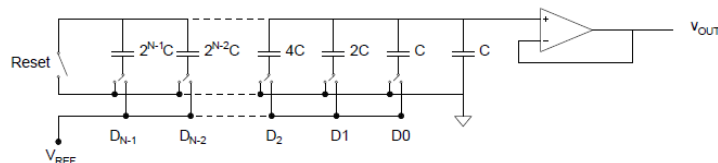


Figure 23: Simple Capacitive DAC architecture [1]

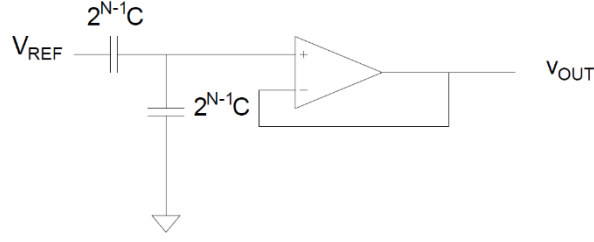


Figure 24: Corresponding Capacitive DAC Architecture when input word is 100..00 and the resulting output voltage is $V_{ref}/2$ [1]

This architecture uses 2^N Capacitors in parallel that are finally connected to an op-amp. The value C of the LSB's capacitor, is the scaling value to the other bits in the architecture. When this architecture starts its operation, must at first be discharged, connecting all capacitors to ground. After that, each switch that represents a bit in the binary word, connects the corresponding capacitor to either ground or V_{REF} . Therefore, a voltage division between the capacitors happens and is later transferred to the buffer. The analogue output for every digital input word can be found by:

$$v_{OUT} = \sum_{k=0}^{N-1} D_k 2^{k-N} V_{REF}$$

A main issue that this architecture has, is that the capacitance of the binary-weighted capacitors is changed due to parasitic capacitance caused by the op-amp. This parasitic capacitance occurs at the top of the plate of the capacitor array thus, discarding this architecture as a choice for high resolution applications. Regarding the static errors INL and DNL, are identical with the binary-weighted current steering architecture and can be derived by substituting the current I with the capacitance C and the error ΔI with ΔC .

There are some layout considerations regarding the architecture mentioned above. In order to have high accuracy in high resolutions, the ratio and the matching between the MSB and LSB capacitors must be precise. This is hard to achieve due to fabrication errors that, as can be seen by the following figure, the capacitors are smaller due to undercutting of the mask. The solution to this problem that lowers the INL and DNL errors is to lay out the capacitors in common-centroid scheme. As a result, the undercutting error is the same to all the errors and the oxide errors average resulting in being the same for each capacitor. In many cases, double polysilicon capacitors are used which have good matching accuracy.

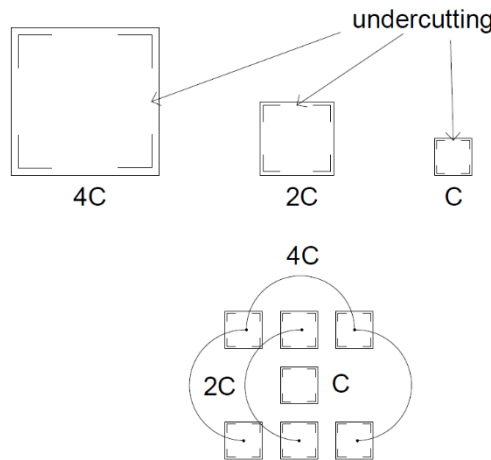


Figure 25: Common-centroid scheme for the binary-weighted capacitor array architecture [1]

However, in high resolutions, the capacitance of the MSB becomes enormous. This can be reduced by using a technique called split array, as can be seen in fig. 26. In this method, the charge-scaling DAC is separated by an attenuation capacitor into two arrays, the LSB and the MSB array. This architecture is quite simple and has good accuracy, which makes it a popular DAC architecture.

$$C_{atten} = \frac{\text{sum of the LSB array capacitors}}{\text{sum of the MSB array capacitors}} \cdot C$$

Where the sum of the MSB array can be found by summing all the LSB capacitors in the LSB capacitor array minus a standardize C capacitance, as can be derived from the figure below [1].

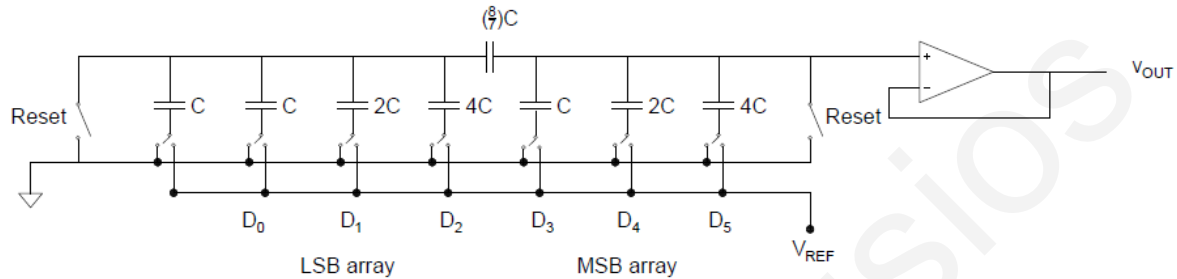


Figure 26: Split Array method on charge scaling DAC to reduce capacitance area [1]

Cyclic DAC

This is a simpler architecture compared to the other mentioned above and can be observed in fig.27. The input bits are converted from parallel to serial and control two switches, the first one adds the reference voltage to the feedback and the second one adds the ground to the feedback loop. A buffer with gain 0.5 feeds the summed voltage back to the feedback loop. The role of the 0.5 gain buffer is to ensure that the output voltage of every cycle is a fraction of the previous output voltage and can be derived by:

$$v_{OUT(n)} = (D_{n-1} \cdot V_{REF} + \frac{1}{2} \cdot v_A(n-1)) \cdot \frac{1}{2}$$

The above equation has the restriction that the initial output of the S/H circuit is zero otherwise offset will be added to the output. In addition, due to the requirement that the parallel bits must be converted to serial, this architecture requires N cycles to have final output.

A major drawback of this architecture is that its accuracy can be altered by many elements. One way that its accuracy might not be precise is the op-amp's gain, that instead of 0.5 is some other value. In addition, the summer and the S/H circuit must not have offset voltage [1].

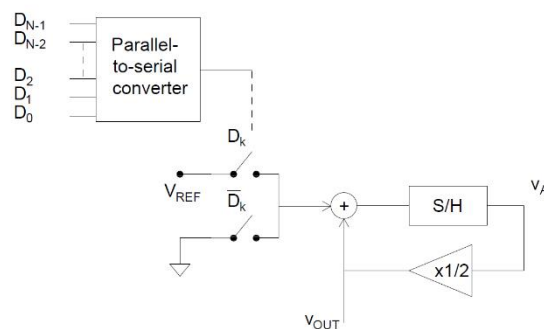


Figure 27: Cyclic DAC architecture [1]

Pipeline DAC

The main advantage of the pipeline DAC as well as the pipeline ADC, is that they require initially N clock cycles to convert the input signal and after that delay, for each clock cycle they can convert the next signal. This can happen, as you can see from the figure below, because there are N steps for converting the signal, one for each bit. As the signal passes to the next step, the previous step can start converting another input signal. Therefore, after the first N cycles, each cycle after that can output a conversion.

The conversion voltage of each stage can be found by:

$$v_{OUT(n)} = [D_{n-1} \cdot V_{REF} + v_{OUT(n-1)}] \cdot \frac{1}{2}$$

The above equation simply implies that if the input bit is equal to 0, the value of the previous voltage is passed to the next stage multiplied by $\frac{1}{2}$, otherwise the reference voltage is added to the voltage output of the previous stage, again multiplied by $\frac{1}{2}$.

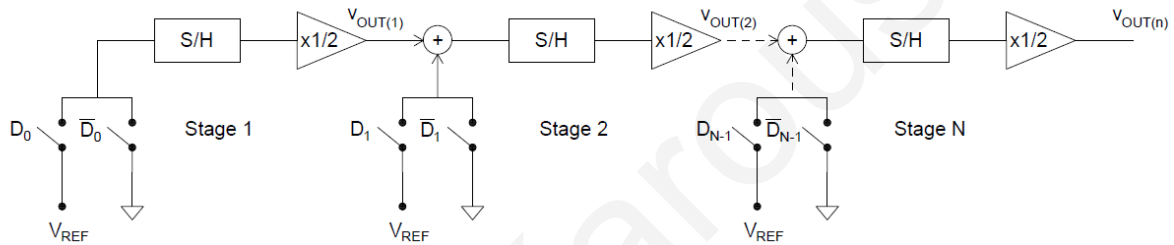


Figure 28: Pipeline DAC architecture [1]

The two major drawbacks of this architecture are firstly the high amount of circuits that are required for this system, which is N times more compared to the cyclic DAC. Secondly, the gain of each amplifier must be accurate in order to eliminate any errors, which requires good matching techniques between the elements of the amplifier. Finally, the S/H circuit must not have any offset that might lead to INL and DNL errors [1].

Comparing DACs

Some architectures have some inherent advantages. For example, regarding speed the most used architecture is the current-steering one which is used thoroughly in telecommunication applications. Nowadays, digital to analogue converters can be implemented using photons which are dominant to the CMOS architectures in clock speed, sampling, have wider bandwidth and their area is reduced. In some cases, can have a conversion rate of 80 MS/s. CMOS is the dominant semiconductor technology and with the rise of the System-on-Chip systems, DACs can be integrated alongside with the digital circuitry, especially for high resolution applications [11]. However, the current-steering architecture has some drawbacks regarding static and dynamic performance. This is mainly due to process variations which lead to mismatch regarding current, glitches and have increased area [12].

In [13], the authors propose DAC Design Guidelines which can aid in better performance of DAC architectures. A solution to the contact impedances from connecting the resistors is the increase the amount of contact holes or vias and place them in parallel. Another way to decrease the impedance is to increase the width of the wire thus its impedance decreases. In some cases the engineers achieved static power dissipation for resistor string architecture as low as $26.4\mu\text{W}$ in $0.13\mu\text{m}$ CMOS architecture [14]. Main effects that cause the DAC to deviate from the necessary INL and DNL values are element

mismatches, thermal noise and semiconductor noise all resulting in static error and noise generation [15].

Now regarding the Cyclic DAC architecture, the resolution and the accuracy depend on the fabrication technology. Many techniques such as dynamic element matching, or self-calibration are used in order to increase both resolution and accuracy, but they sacrifice area. In addition, the accuracy of the converter also depends on the gain of the S/H circuit. However, an advantage of this architecture is its simple design [16]. A modification of the cyclic DAC architecture called quasi-passive cyclic DAC can be found in [17][18] which uses switched-capacitors. In this design, due to process variation immunity, smaller capacitors can be used thus minimizing the area and power consumption. On the other hand, because cyclic DACs are sensitive to parasitic capacitance, additional circuitry is required so that the output of the DAC can be transferred unaltered to the next stages. The name quasi-passive means is given to this architecture because no amplifier is used thus eliminating the aforementioned error that the S/H circuit introduces to the system. In addition, this architecture has only two capacitors with the same size that need matching thus making the design even simpler and manage to achieve INL error less than $\pm 0.15\text{LSB}$ [17]. Additionally, in [18] they propose an active SC cyclic DAC, by using $0.18\mu\text{m}$ technology, with capacitor mismatch compensation and it is implemented with three capacitors and an amplifier. At the end, the authors achieved $\text{INL}=0.012\text{LSB}$.

The R-2R architecture is quite simple to implement and it is area efficient. A major drawback is the good resistor matching that is required to maintain not only linearity but also INL and DNL values within limits [19]. [20][21] Some engineers prefer to use $\Delta\Sigma$ techniques or current sources which are dynamically calibrated which are more linear and with higher resolution than the other architectures, sacrificing speed. Instead, other engineers prefer to use laser trimmed R-2R topologies, which are costly and area consuming. Otherwise, hybrid DACs or self-calibrated R-2R ladders can be used [20][21]. Engineers managed to reduce the total area of this architecture by implementing a CMOS-only R-2R ladder DAC with total area of 0.03mm^2 , 0.55mW power consumption for a full digital cycle, DNL and INL less than $\pm 1\text{LSB}$ and $\pm 0.5\text{LSB}$ respectively [22]. The authors in [23] proposed a nonlinear R-2R architecture, again using only transistors. They achieved INL and DNL errors less than $\pm 0.5\text{LSB}$. In addition, the terminal and threshold voltages alongside with the process parameters do not produce any errors. The system can only be affected by the aspect-ratio values of transistors that comprise the ladder. The R-2R architecture can be operated in both current and voltage mode. In the first mode, the output impedance is code dependent. In the second mode, the output impedance is not code dependent which makes the output op-amp unnecessary. In order to control the output a resistive load can be simply added. However, due to the fact that the voltage mode architecture's input is code dependent, it requires grounding and voltage reference to be connected outside of the chip which cause linearity errors. These errors are produced by the resistance in the wire which cause voltage drop [24].

Another topology identical to the R-2R, is the C-2C which instead of resistors uses capacitors. However, this architecture suffers from $\frac{kT}{C}$ noise when the switch to reset the capacitor opens. From the output node to the ground, the total capacitance is $2C$ and must be high in value so that the noise is negligible. Another drawback is the parasitic capacitances that all capacitors in this scheme have thus interfere with the linearity of the analogue output [25].

The current steering architecture is the fastest one among the Digital to Analogue Converters and it is preferred not only for high speed applications but also when the output of the DAC is a resistive load. As mentioned above, the current steering topology can be implemented either using thermometer code, which means that the current devices are identical, or using binary weighted devices [10]. Its major disadvantage is the element matching, which can be minimized by using techniques such as trimming, calibration and dynamic element matching. In addition, in order to improve INL and DNL errors, segmented DACs can be used [26]. An inherent advantage of the current steering architecture is their ability to drive resistive loads. The binary weighted architecture's advantage is the use of smaller

area due to lack of decoder. The downside is that monotonicity might not be achieved. Additionally, the glitch energy wasted is analogous to the number of switches used in the circuitry. Another structure, the thermometer DAC uses devices that have the same size. When there is only 1 LSB change in the input, only one device switches on or off, therefore monotonicity is certain. Additionally, the matching of the transistors is better because the current devices are the same. Finally, the last advantage is that the glitch energy wasted is analogous to 1 LSB, therefore is much less. The drawback of this architecture is the need for use of a decoder to convert binary to thermometer code resulting in using more area. The last architecture is the segmented DAC that combines the last two architectures. Thermometer code is used in the MSB section where accuracy is needed and on the LSB section binary code is used. Some segmented architectures use multisegmented techniques, like this one thus using less chip area [27].

Advantages of Resistor String DACs is that they are simple, they have intrinsic monotonicity and they have uninterrupted voltage output. When they are compared to Current Steering DACs, they are better in terms of area and power consumption at low speed. Regarding applications where something must be controlled, they are better when compared to Charge Scaling DACs because the latter have switches that produce noise and glitches [7][8].

DAC Architecture	Advantages	Disadvantages
Resistor String	Simple Design Inherent Monotonicity Fast	Needs good resistor matching Medium resolution Need buffering for low impedance loads
R-2R Ladder	Simple design	Medium resolution Medium speed Need buffering for low impedance loads
Current Steering DAC	Fastest High resolution No need for buffering for low impedance loads	Power hungry
Capacitive DAC	Simple design	Needs good capacitor matching Medium resolution Medium speed Need buffering for low impedance loads
Cyclic DAC	Simple design	Slow Medium resolution Need buffering for low impedance loads
Pipeline DAC	Slow but after first conversion, fast	Complicated design Medium resolution Need buffering for low impedance loads

Table 1: Digital to Analogue converters summary

III. Analogue to Digital Converter

Introduction

Analogue Signal

An analogue signal is continuous with respect to time, meaning that there are no interruptions. For the digitized signal to be accurate, firstly the number of samples taken need to be high. A simple way to achieve adequate sampling rate is the *Nyquist Criterion* which requires that the samples taken, also known as sampling rate must be at least twice as the highest frequency existing in the analogue signal. Secondly, the resolution must be high and can be achieved by setting low value quantization levels. These values are set by the requirements of each project [1].

Nyquist Criterion. f_{SAMPLING} is the required frequency in order to obtain adequate samples to represent the analogue signal at the output. f_{MAX} is the highest frequency that can be found in the analogue signal. [1]

$$f_{\text{SAMPLING}} \geq 2f_{\text{MAX}}$$

Sample and Hold

A critical circuit that accompanies ADCs converters is the Sample-and-Hold (S/H) circuit, which has 4 major components: Input Amplifier, a capacitor, an output buffer and the switch driver, fig.29. The input amplifier acts as a buffer. Its high input impedance keeps the signal from changing. In addition, provides current to the capacitor, which is an energy storage device, in order to charge and hold the input signal. This circuit has two states: Sample and Hold, as its name defines. On the first state, the circuit must track the analogue input and the capacitor has the same voltage as the input. In the second state, the switch opens. The capacitor has stored the analogue voltage from the input buffer and it is transferred to the output buffer. In this state, the capacitor must hold its energy as long as possible, thus the output buffer must have high impedance so the capacitor can maintain its voltage steady longer. This gives the ADC enough time to process the input signal. A S/H signal can limit the speed and accuracy of an ADC resulting in major errors regarding speed and accuracy [28].

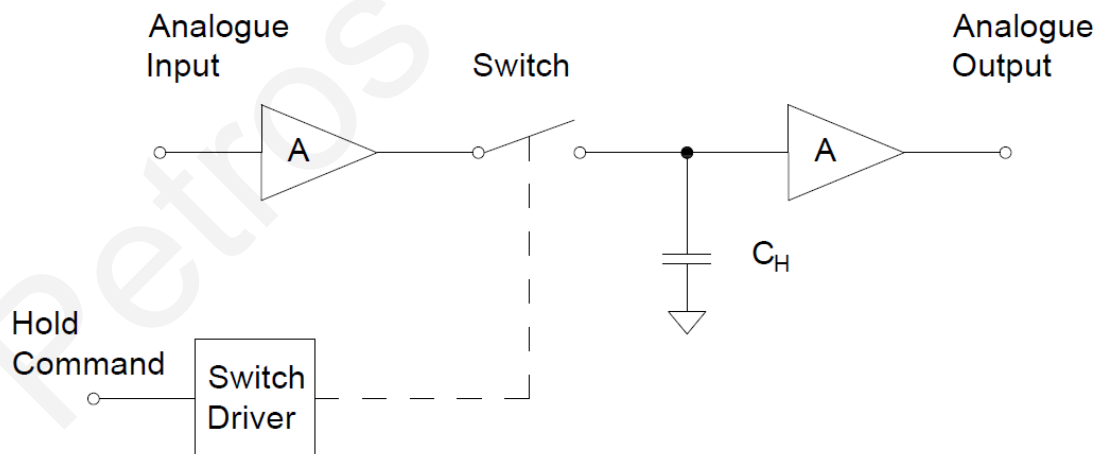


Figure 29: A sample and hold circuit [28]

S/H circuit starts to function when the sampling command is sent. Acquisition time is the total time required for the S/H to track the analogue signal to a value that is within the required margins. Acquisition's worst-case scenario is when the analogue signal must be sampled from zero voltage to its peak voltage, $V_{\text{IN(max)}}$. In a simple S/H circuit where a buffer is used, its slew rate, stability and offset voltage are limiting the overall performance of the S/H circuit. The overshoot can happen if the amplifier has a small phase margin which leads to instability, thus extending settling time. The offset

and gain error result in error in tolerance at the output of the S/H circuit. Ideally, the S/H must have gain=1 in order to recreate accurately the analogue signal and not alter its value [1].

Charge injection is one reason for the Pedestal error. Charge in the switch is distributed to the capacitor altering its voltage. In addition, clock feedthrough is another issue that causes Pedestal error. The gate/source or gate/drain overlap capacitance is coupled with the capacitor, thus altering the total capacitance and the total voltage. Parasitic impedances of the S/H capacitor and the switch's drain that forms a reverse-biased diode result in current leakage and are responsible for the Droop error. By reducing the drain area of the reverse-biased diode, the Droop error is improved. Leakage can also occur from the OFF state of the switch and its substrate. The key factor in order to reduce Droop error is by making large S/H capacitor, thus limiting the leakage current. However, a large S/H capacitor results in a slower response to the input signal [1].

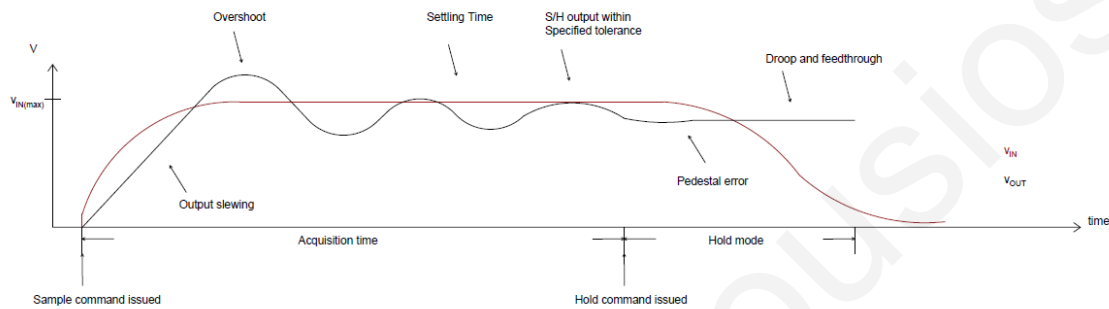


Figure 30: Errors that can be caused from the S/H circuit [1]

Characteristics

An ADC must be able to quantize the input analogue signal successfully, in respect to the Number of quantization levels which can be found by 2^N where N is the number of output bits. However, similarly with the DAC's input word case, the output word of the ADC will have maximum output 2^{N-1} [1].

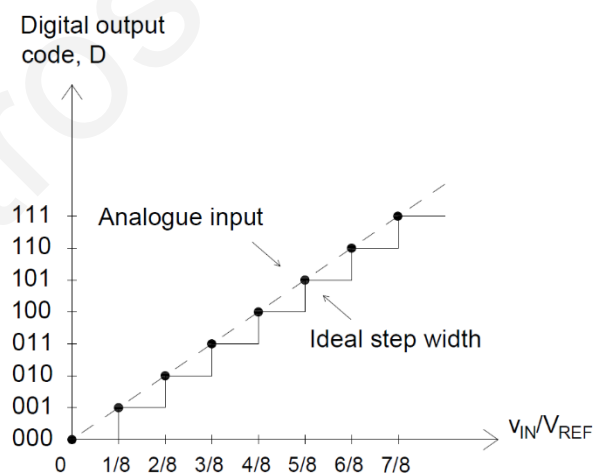


Figure 31: The ideal transfer curve (staircase) of a 3-bit resolution ADC with normalized X-axis [1]

The above figure represents the ideal digital output of an ADC versus its input. This graph shaped as a staircase of a 3-bit ADC has an ideal step width of $1/8$ due to its resolution. The LSB can be found by using the equation from [1],

$$1LSB = \frac{V_{REF}}{2^N}$$

Quantization Error - Q_e

As can be seen from the figure above, the analogue input is continuous versus the digital output, meaning that whilst the analogue signal continues to rise up, the digital signal cannot keep up.

$$Q_e = v_{IN} - V_{staircase}$$

$$V_{staircase} = D \cdot \frac{V_{REF}}{2^N} = D \cdot V_{LSB}$$

The above two equations are used in order to find the Quantization error. Similarly, with the DAC's case, in the above equations, D denotes the output word. Quantization Error must have value between 0 and 1 LSB. In order for the Q_e to be calculated, the best way is to shift the transfer curve of the ADC by 0.5LSB to the left as can be seen in fig.32. Then the Q_e can be calculated by subtracting the digital word from the analogue input. As a result, the actual Q_e will be $\pm 0.5\text{LSB}$, which is the maximum Quantization Error for an ADC [1].

However, shifting the transfer curve, results in another problem, as can be seen in the figure below. The last code word represented by the ADC, has $Q_e=1$, which exceeds the tolerance mentioned above. This problem is considered negligible and it must be placed into consideration only if

$$\frac{v_{IN}}{V_{REF}} \geq \frac{2 \cdot 2^N - 1}{2 \cdot 2^N}$$

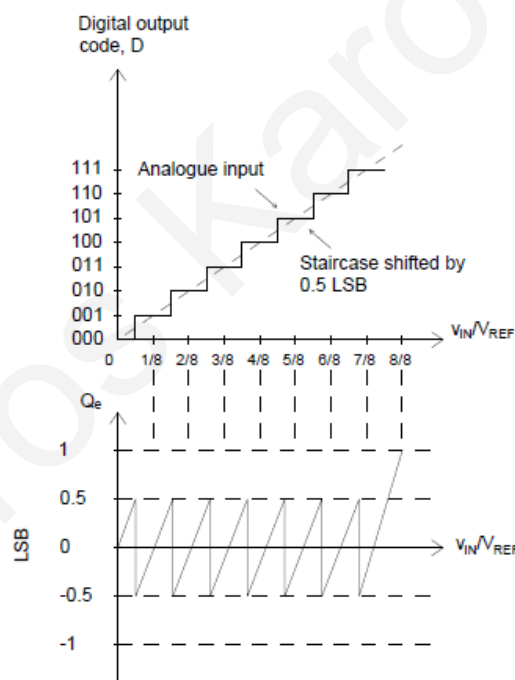


Figure 32: The ideal transfer curve (staircase) of a 3-bit resolution ADC shifted by 0.5LSB and the corresponding Quantization error [1]

Differential Nonlinearity

Similarly with DNL characteristic that we described earlier for the DAC, for an ADC is the reverse, meaning that in this case, DNL is the result the width of the ideal step subtracted from the width of the actual step. The DNL can be also represented in LSBs, which can help with the proper characterization of an ADC, which normally $\text{DNL} = \pm 0.5\text{LSB}$. The Ideal Step value is actually 1 LSB or $\frac{V_{REF}}{2^N}$ Volts [1].

$$\text{DNL}_n = \text{Value of Actual Step} - \text{Value of Ideal Step}$$

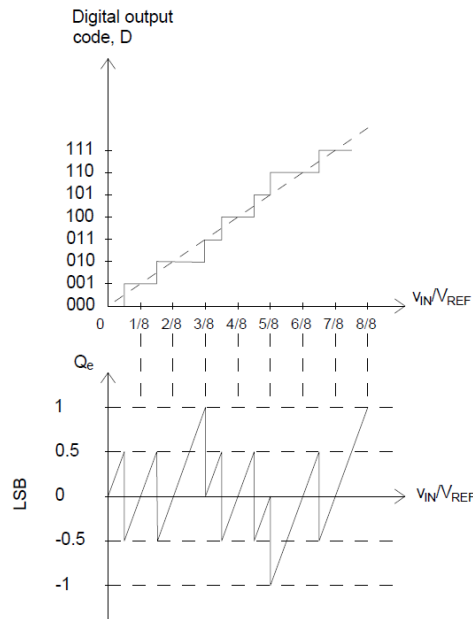


Figure 33: An example of a 3-bit ADC and its corresponding Quantization error showing the DNL [1]

Integral Nonlinearity - INL

In order to identify the Integral Nonlinearity Errors, similarly with the DAC case, a straight line must be drawn through the first and last input word when the transition happens, and all other errors are set to zero.

$$INL_n = \text{Value of Straight Line at Transition} - \text{Ideal Value When Transition Happens}$$

Missing Codes

Regarding the DNL error mentioned above, being greater than $\pm 0.5\text{LSB}$ results into missing output words thus reducing the resolution of the ADC [1].

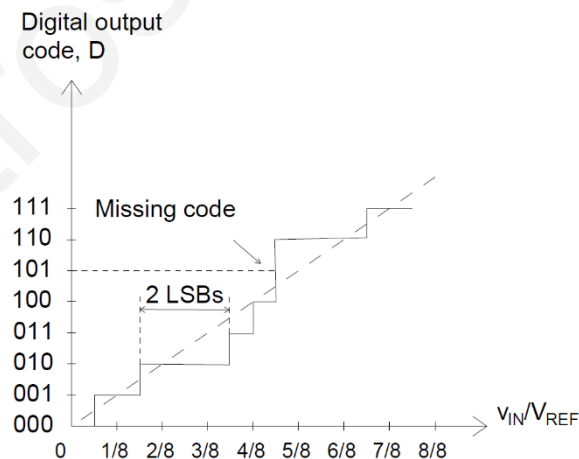


Figure 34: An example of a 3-bit ADC having DNL error above 0.5 LSB resulting in Missing Codes. [3]

Offset Error

Again, those errors are similar with the DAC's case. The Offset Error can be easily described as a shift in the transfer curve of the ADC. It can also affect the Q_e by shifting its corresponding output, as can be seen in the figure below. In the [1] describes that "Offset Error occurs when there is a difference between the value of the first code transition and the ideal value of $1/2$ LSBs".

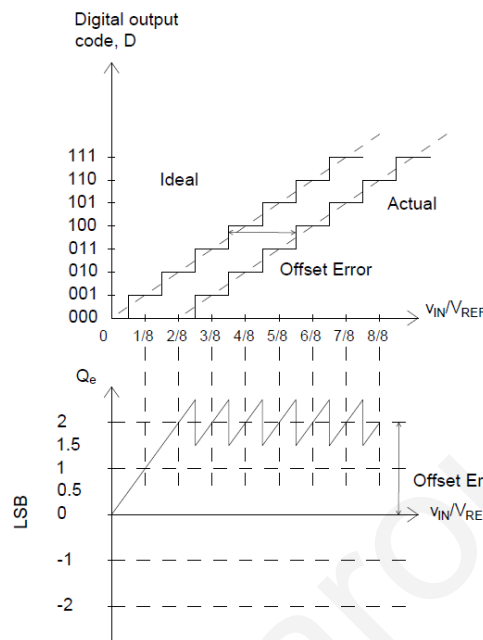


Figure 35: An example of a 3-bit ADC transfer curve with Offset Error [1]

Gain Error

Regarding the Gain Error, requires the ideal straight line mentioned in INL and another line drawn on the actual values. The Gain error is the difference between the slopes of the two lines. In addition, Gain Error is also called Scale Factor [1].

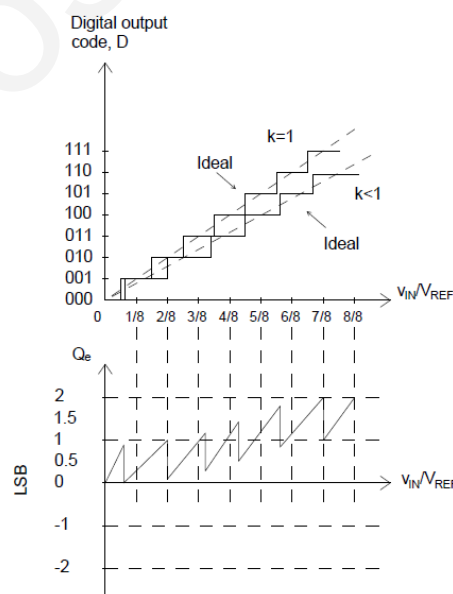


Figure 36: An example of a 3-bit ADC transfer curve with Gain Error [1]

Aliasing

Happens when the Nyquist Criterion mentioned above is not satisfied, resulting in losing information of the input signal, as can be seen in the figure below where the analogue input is sampled at lower frequency than the actual frequency that exists in the signal. In order to eliminate Aliasing, the frequency that the input signal is being sampled must be increased, as per the Nyquist Criterion. In addition, again according to the Nyquist Criterion, by using low pass filter eliminating frequencies that have higher values than the half of the sampling frequency, can assist with the proper sampling of the input signal [1].

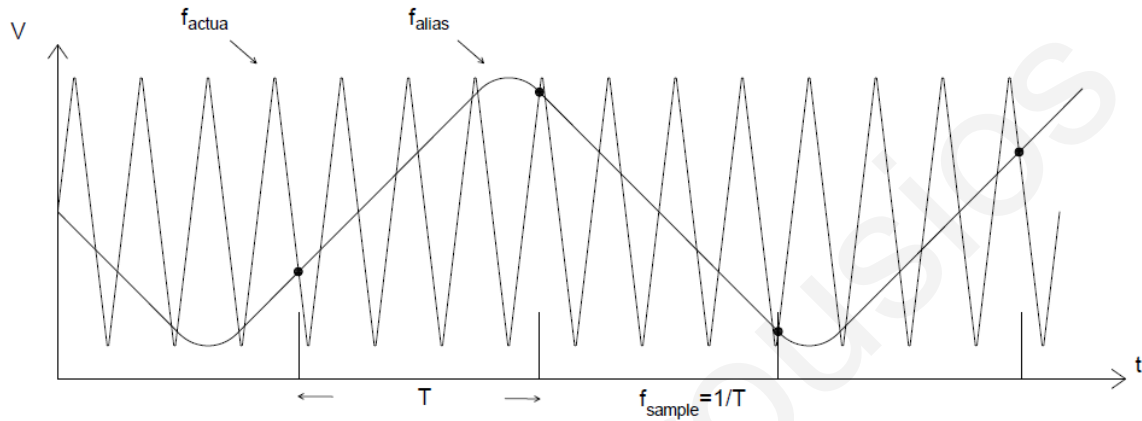


Figure 37: An example of aliasing due to undersampling [1]

Signal-to-Noise Ratio - SNR

In order to accurately describe an ADC, the dynamic nonlinearity must be expressed. This is done as described in [29] by finding the Signal-to-Noise Ratio of the ADC by applying a peak-to-peak sinusoidal input wave to the ADC. Taking into considerations glitches, INL and sampling-time uncertainty, the SNR for an N-bit Analogue to Digital converter is

$$SNR(in\ dB) = N \times 6.02 + 1.76$$

Mixed-Signal Layout Issues

Analogue Integrated Circuits have increased sensitivity to noise versus digital Integrated Circuits. Engineers have found solutions to minimize the affection of noise to the analogue elements in an IC. Shielding the sensitive analogue nodes as well as grounding techniques have been established, alongside with power supply alternative routings. The main consideration for mixed-signal layout design is the floorplanning. Since in mixed-signal ICs analogue and digital circuits co-exist, their grounds and power supply lines must be connected separately. Finally, matching analogue devices is a critical issue alongside with guard rings and shielding [1].

Floorplanning

This step must be carefully planned since sensitive analogue nodes, such as weak signals or nodes with high impedance must be inside guard ring or shielded from noise, especially if they relate to input signals. In addition, digital circuits must undergo detail planning taking into consideration their speed and function. Since the digital output drivers are attached to capacitive loads with parasitics, they must be placed as far as possible from the analogue devices. Furthermore, sensitive analogue devices must be placed as far as possible from the output drivers. Between those two, analogue devices with rail-to-rail analogue output are placed, alongside with low and high-speed digital circuitry [1].

Grounding and Power Supply lines

Proper grounding and power supply routing secure that no noise will travel from the digital devices to the analogue devices. Noise can travel from the digital lines to the analogue lines and can alter the function of the analogue devices. The best solution is to separate ground and power lines, by connecting each one to separate pad. In addition, cable's induction from the signal generator and the pad's resistance must be taken into consideration, especially in analogue devices that can be affected by an alteration in resistance and inductance [1].

Architectures

There are many types of ADCs, each one designed to serve specific purpose. Such ADCs are pipeline, flash ADCs, successive approximation and oversampled ADCs. The drawback of the ADCs is that, the input analogue signal can have full analogue range thus making it hard to translate its full range to digital. As a result, for each ADC we examine the analogue input that the converter can convert.

Flash ADC

This type of ADC is used where speed is a necessity. The Flash ADC as seen in the figure below, converts the analogue input signal in parallel by using one comparator for each quantization level and resistors. This means that the total count of comparators is $2^N - 1$ and the total count of resistors that are used as a resistor-string DAC is 2^N . The reference voltage is split into 2^N values and each comparator compare a portion of the divided reference voltage with the input voltage resulting into a thermometer digital output which requires a $2^N - 1: N$ decoder to convert the thermometer code to an N-bit word. The thermometer's output is zero when the analogue input voltage is less than the analogue output of the resistor string, that divides the reference voltage. As mentioned above, the main advantage of this architecture is its speed because for each clock pulse, there is a digital output.

However, this type of ADC uses much more area and power than other architectures, mainly due to the switching of comparators. In addition, if the number of bits for the required resolution increase by 1, from 8-bit to 9-bit, the number of comparators needed from 255 jumps up to 511. As a result, this type of ADCs has limited use of up to 8-bits, using 255 comparators with speeds up to 40Ms/s [1].

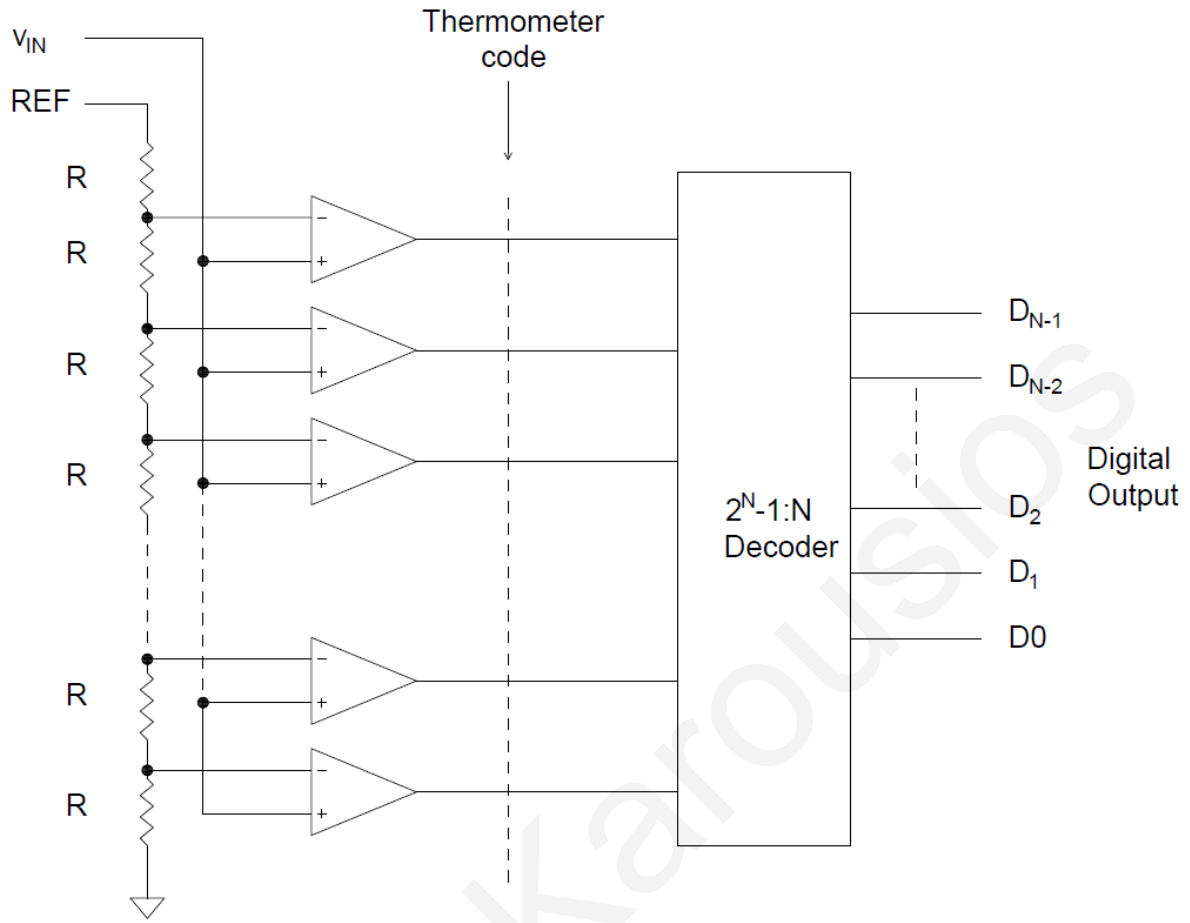


Figure 38: Flash ADC [1]

As in every design, there are some flaws in this architecture. In this case, the accuracy depends on the resistor string and the matching among its resistors. Another issue is the offset of the input voltage that is fed to the comparators. As a result, instead of switching when $v_+ = v_-$, the comparator takes into consideration the offset voltage V_{os} . The output of the comparator is 1 when $v_+ \geq v_- + V_{os}$ otherwise its output is $v_o = 0$. The resistor's DAC voltage at any position can be found by:

$$V_t = V_{i,ideal} + \frac{V_{REF}}{2^N} \sum_{k=1}^i \frac{\Delta R_k}{R}$$

where $V_{i,ideal}$ is the ideal voltage for the i -th resistor, ΔR_k is the mismatch error. With that being said, the maximum INL can be found by:

$$|INL|_{max} = \frac{V_{REF}}{2} \cdot \left| \frac{\Delta R_k}{R} \right|_{max} + |V_{os,j}|_{max}$$

where $V_{os,j}$ is the offset voltage of the input of the i -th comparator. The maximum negative INL happens at the upper half of the resistor DAC and the maximum positive at the lower half. Due to the input referred offset voltage, this architecture has limitations in high-speed applications because it messes up with accuracy. Regarding the DNL this can be found by assuming that the maximum offset voltage at the middle of the string, in both directions is symmetrical [1]. The equation as described in [1] is:

$$|DNL|_{max} = \frac{V_{REF}}{2^N} \cdot \left| \frac{\Delta R_i}{R} \right|_{max} + 2|V_{os}|_{max}$$

The Two-Step Flash ADC

Another architecture is the Two-step Flash ADC. This type is also parallel with feed-forward control as can be seen in the figure below. There are two flash converters, one roughly estimating the input value and the other one converting the values with precision. This architecture uses less comparators than the Flash ADC mentioned above and instead of using $2^N - 1$ comparators, it uses $2 \times (2^{N/2} - 1)$. This means reduced area consumption. However, the conversion is done in two steps instead of one thus adding to the conversion time. In addition, the conversion time depends on the bandwidth and the settling time of the residue amplifier and the subtractor. The function of the Two-Step Flash ADC is not as simple as the Flash ADC. Firstly, the analogue signal is sampled by a S/H circuit. Then the MSBs are converted by the rough ADC. The linearity of the MSB ADC is responsible for the accuracy of the system. A DAC converts the output of the MSB ADC back to analogue signal as to be subtracted from the original signal. This output, also called residue is then fed into an amplifier with gain $2^{N/2}$. The amplified analogue output is then pushed into the second ADC which converts the LSBs, thus making the fine conversion. The amplification's purpose is to ensure that both the ADCs are the same. With that being said, the MSB ADC must have the accuracy of double the bits than the LSB ADC. As a result, the worst case INL and DNL error of the MSB ADC, even though it is a 4-bit converter, must be less than $\pm 0.5 \text{ LSB}$ of an 8-bit Flash converter. This requires proper resistor and comparator matching for the MSB ADC [1].

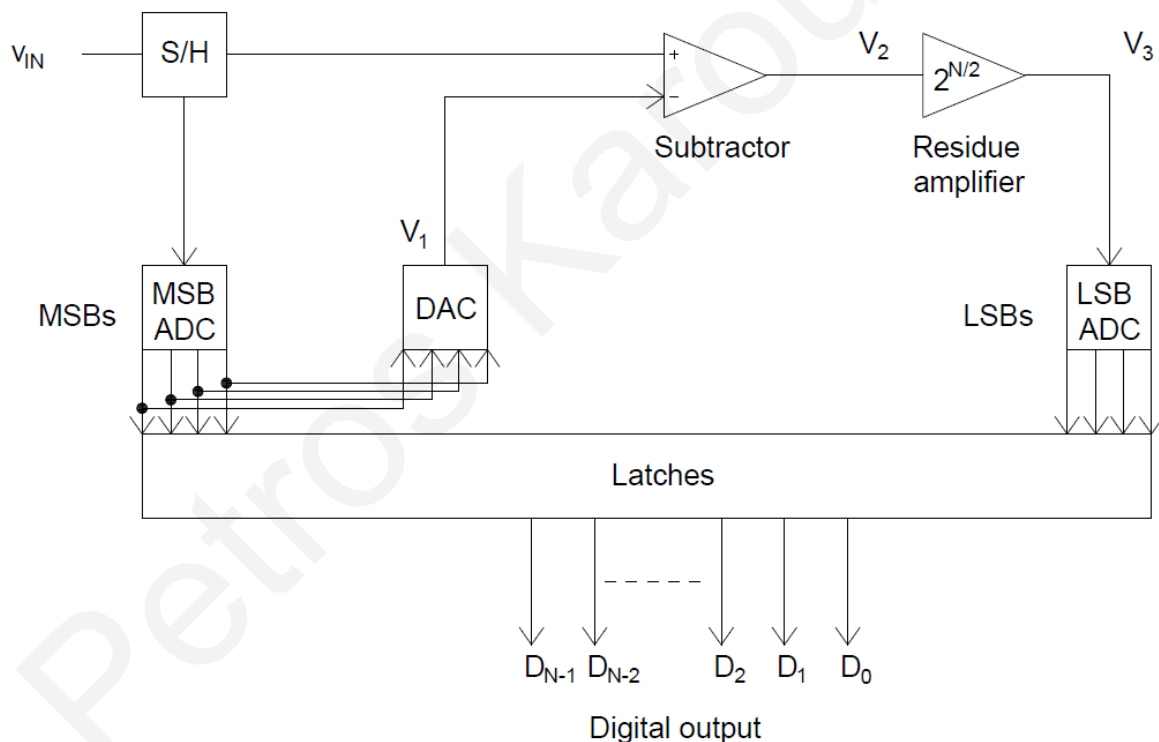


Figure 39: Two step flash ADC [1]

As mentioned above, the addition of the subtractor and the Residue Amp creates some issues to the system, especially if their offset is not as low as $\pm 0.5 \text{ LSB}$. A difficulty that designers face is the problem that the amplifier must multiply the signal by a factor of two. The difficulty in this case, is the open-loop gain, which the amplification of the signal depends on. For high resolution applications it is required to be high resulting in designers using two-step Flash ADCs for resolutions up to 12-bits. Another issue that this design faces is the amplifier's linearity. The range of the range of the input signal that can be amplified, must have error less than 0.5 LSB [1].

Pipeline ADC

The Pipeline ADC is an N-step converter, which converts one bit every step. Its purpose is to convert analogue signals with high resolution and quite fast. Each step includes a S/H circuit, a comparator that serves as an ADC, a summer and an amplifier with gain=2, as can be seen in the figure below. Firstly, the Voltage Reference must be divided by 2 and the analogue input signal is sampled and hold. After the S/H, the v_{IN} is compared to $\frac{V_{REF}}{2}$ in every comparator. The digital output is the bit conversion for that state. If $\frac{V_{REF}}{2}$ is lower than v_{IN} , the summer subtracts $\frac{V_{REF}}{2}$ from the S/H signal and it is pushed to the amplifier and the comparator's output is set to 1. If $\frac{V_{REF}}{2}$ is larger than the input signal, the comparator's output is set to 0 and the sampled signal is passed to the amplifier. An amplifier doubles the result of the summation and pushes it to the next step [1].

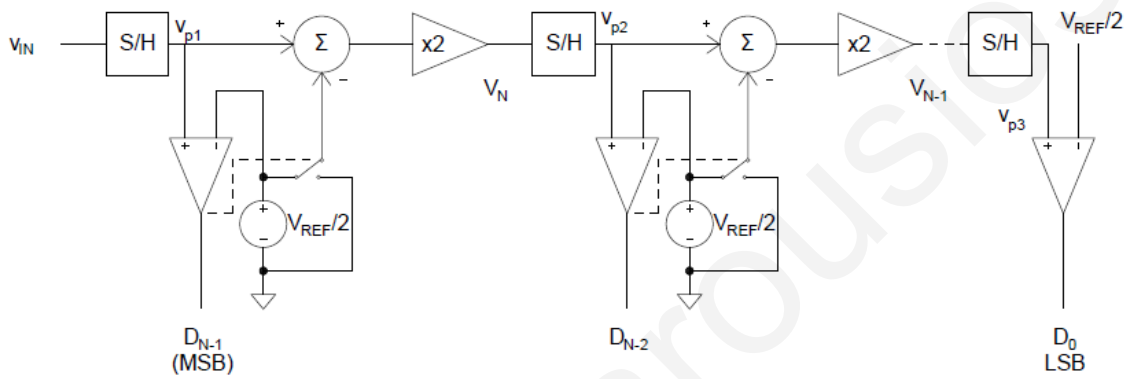


Figure 40: Pipeline ADC [1]

As mentioned above, the Pipeline ADC needs N-steps to make a full conversion. However, when the conversion is finished for the 1st comparator, it is then ready to receive the next input signal and start the conversion. This results in high throughput as one conversion can be done in one clock cycle. The disadvantage is the N-step delay at the start of the first conversion. Pipeline ADC requires careful design because if an error occurs in a step, it is then passed to the next one [1].

The Integral Nonlinearity of the Pipeline ADC can be found be:

$$INL_N = \frac{1}{2} \cdot D_{N-2} \cdot V_{REF} \cdot \left(\frac{1}{A} - \frac{1}{2} \right) + \frac{1}{2} \cdot D_{N-3} \cdot V_{REF} \cdot \left(\frac{1}{A^2} - \frac{1}{4} \right) + \dots + \frac{1}{2} \cdot D_1 \cdot V_{REF} \cdot \left(\frac{1}{A^{N-2}} - \frac{1}{2^{N-2}} \right) + \frac{1}{2} \cdot V_{REF} \cdot \left(\frac{1}{A^{N-1}} - \frac{1}{2^{N-1}} \right) + \frac{V_{COS,N}}{A^{N-1}} - \sum_{k=1}^N \frac{V_{SOS,K}}{A^{k-1}}$$

In the above equation, A is the gain of the amplifiers, D_{N-x} is the output of the comparator, either 1 or 0 and x is the step number. $V_{COS,x}$ and $V_{SOS,x}$ are the comparator's and the S/H circuit offset voltage.

The Pipeline converter suffers from high INL when the value of the offsets is higher than 0.5LSB. In addition, it can be observed that as we go to the larger steps, tend to be less prone to errors because they are divided with high gain value thus their design does not have to be as accurate as the design of the first steps. Additionally, amplifier's and S/H offset can introduce nonlinearities to the conversion [1].

The maximum DNL error can be found by:

$$DNL_{max} = \frac{1}{2} \cdot V_{REF} \left(1 - \sum_{K=1}^{N-1} \frac{1}{A^K} \right) + V_{COS,1} - \frac{V_{SOS,K}}{A^{N-1}} + \sum_{K=2}^N \frac{V_{SOS,K}}{A^{k-1}} - \frac{V_{REF}}{2^N}$$

and must be less than 0.5LSB to achieve the desired accuracy. As one can observe, the comparator's offset dominates the equation [1].

Single-Slope Architecture

It is used in applications that require high resolution conversion where speed is not a concern and budget must be kept low. This type of converter uses a counter, which counts the clock pulses from start to finish of the conversion. When the integrated value of the reference voltage is compared to the input signal and they are found equal, the counter stops, and its product is the digital output.

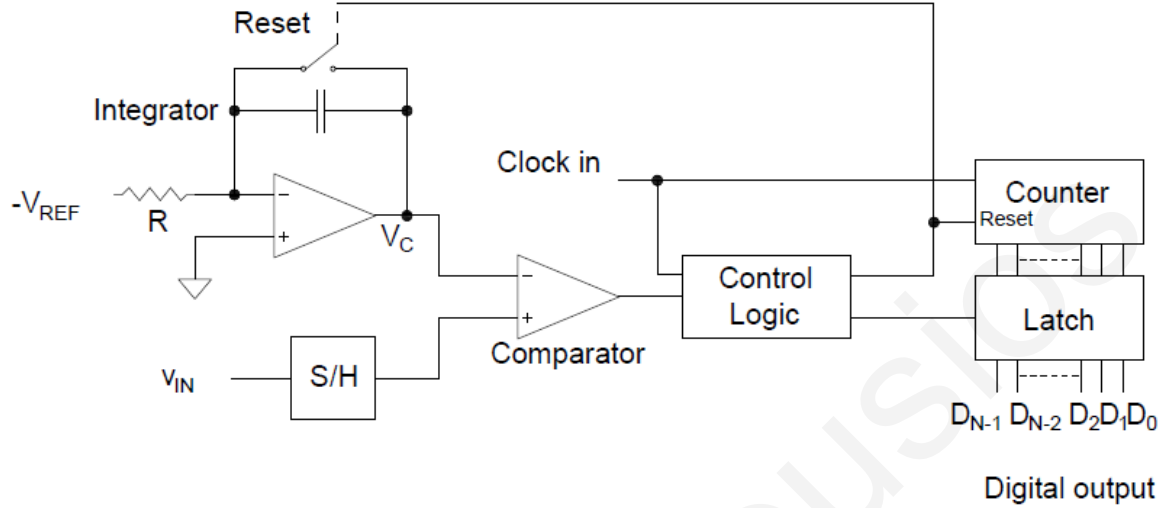


Figure 41: Single Slope architecture [1]

The designer must take into consideration that the reference voltage must be of negative value to enter the integrator. The comparator compares the signal from the S/H circuit, which is the input signal with the integrated signal. When the integrator's output is higher in value than the input signal, the comparator's state changes thus the control logic latches the value of the counter. Another function of the control logic is to reset the system as to prepare for the next sample [1].

One drawback of this architecture is that in order to accurately convert the input analogue signal, the bandwidth of the analogue signal must be quite bigger than the clock's frequency. The conversion time can be found by:

$$t_c = \frac{v_{IN}}{V_{REF}} \cdot 2^N \cdot T_{CLK}$$

And the sampling rate can be found by:

$$f_{sample} = \frac{V_{REF}}{V_{IIN} \cdot 2^N} \cdot f_{CLK}$$

In these two equations, T_{CLK} is the clock period in seconds [1].

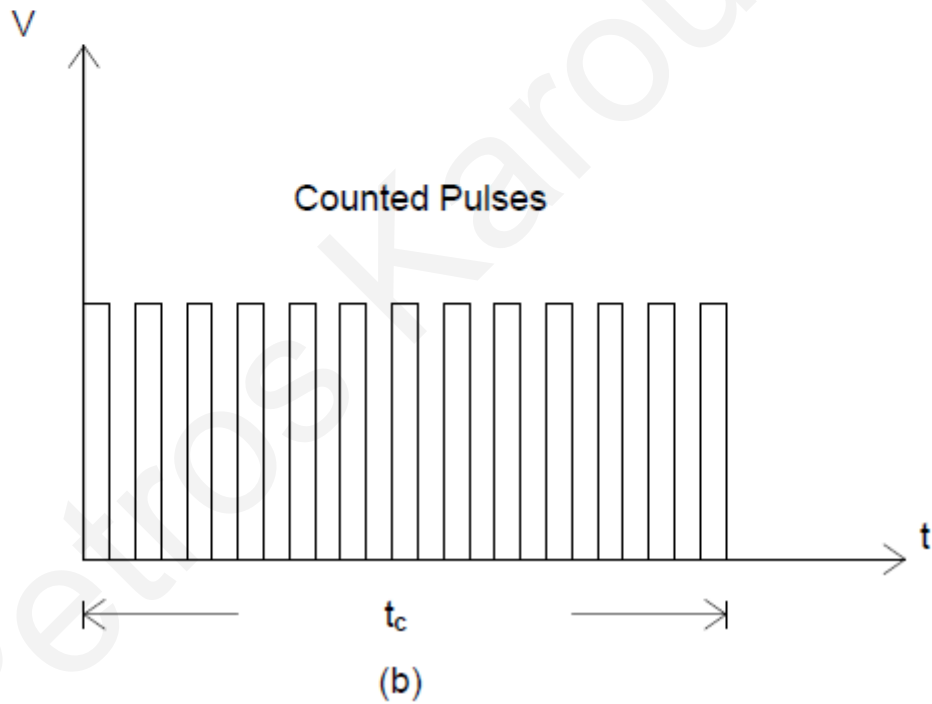
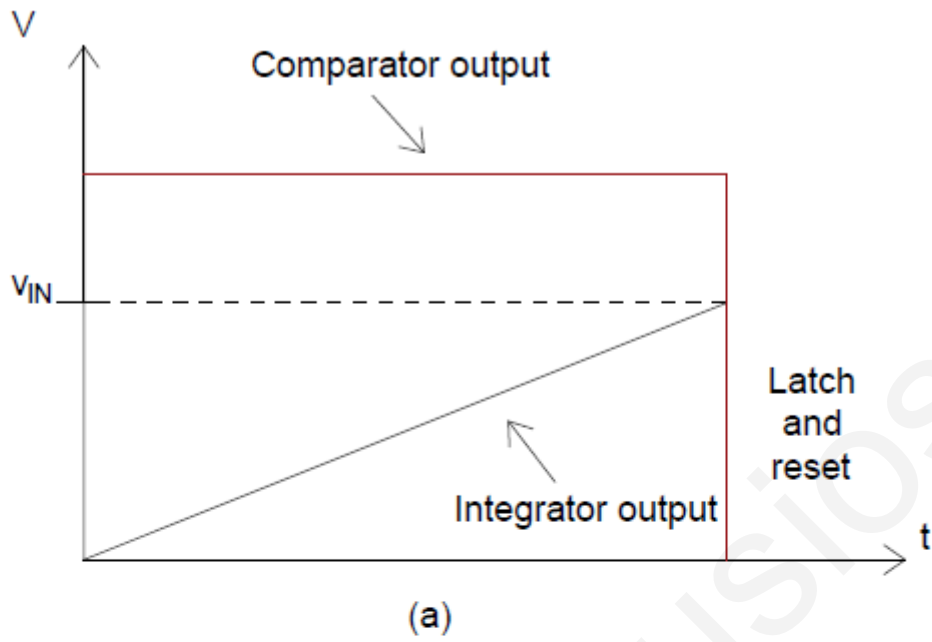


Figure 42: Single slope architecture output versus clock cycles [1]

This architecture is prone to errors due to the fact that many components are involved in the conversion, as can be observed by the following equation: $V_C = \frac{1}{C} \int_0^{t_c} \frac{V_{REF}}{R} dt$. This equation assumes no initial condition for the capacitor and after combining it with the above equation the result we get is $V_C = \frac{2^N \cdot v_{IN}}{f_{CLK} \cdot RC}$. With just a glimpse, one can understand why in this architecture many things can go wrong. Firstly, if there is any offset in the input voltage this will alter the integration. In addition, resistor's linearity or the capacitor's charge leakage might affect the result [1].

Dual-Slope Architecture

This architecture is more advanced than the Single-slope ADC thus eliminating the problems that the latter one was facing. As mentioned in its name, two integrations are performed. However, in this case, only the analogue input voltage needs to be negative. Firstly, the input signal is integrated in a fixed integration period called T_1 , as can be seen in the figure below, which is for a 3-bit architecture. This creates the first slope which has upward direction, charging the capacitor. After the counter is full and is reset, a switch connects the reference voltage to the input of the integrator, which is of positive value thus discharging the capacitor. This time is called Variable integration period T_2 . The counter measures the time needed for the capacitor to discharge hence the digital output is generated [1].

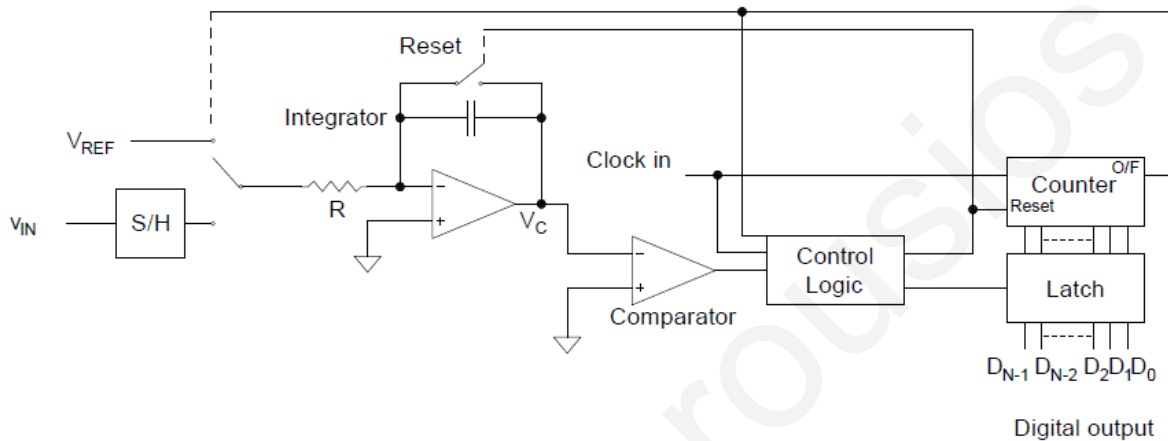


Figure 43: Dual slope architecture [1]

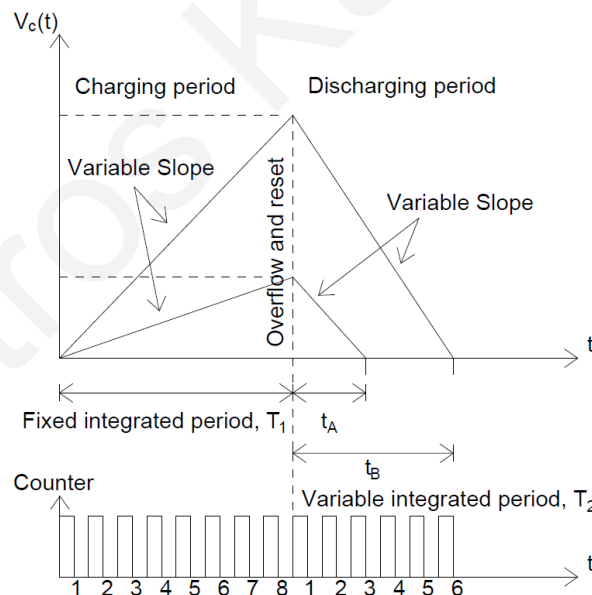


Figure 44: Output of the dual-slope architecture versus clock cycles [1]

The reason that Dual-Slope architecture is preferred over the Single-Slope architecture, is mainly due to the fact that the same components, integrator and clock, are used for the input and the reference signal thus any errors that might occur when integrating the input signal will be cancelled when the reference signal is integrated. The equation that describes the system is $\frac{D}{2^N} = \frac{|v_{IN}|}{V_{REF}}$ and one can

observe that the non-linearities regarding the Capacitor and the resistor are eliminated. In this equation, D is the output of the counter, which is also the digital output of the system [1].

The Oversampling Analogue to Digital Converter

As mentioned in its name, this type of converters uses sampling rate f_N many times higher than the bandwidth of the signal F . This is an overkill of the Nyquist criterion, which states that $f_N = 2F$ resulting in much higher resolution. One compromise for using oversampling ADCs, is the use of complex Digital Signal Processing instead of complex analogue circuits thus eliminating errors produced by S/H circuits and resistor or capacitor mismatches. The main drawback for this type of architectures is the time required to oversample the input signal, thus requiring more time to output the digital word when compared to Nyquist rate ADCs. The block diagrams of the two different type of ADCs can be seen in the following figure, in which one can observe that no anti-aliasing filter or S/H circuit is needed. Instead, an $\Sigma\Delta$ modulator is used alongside with a digital filter which functions as an encoder [1].

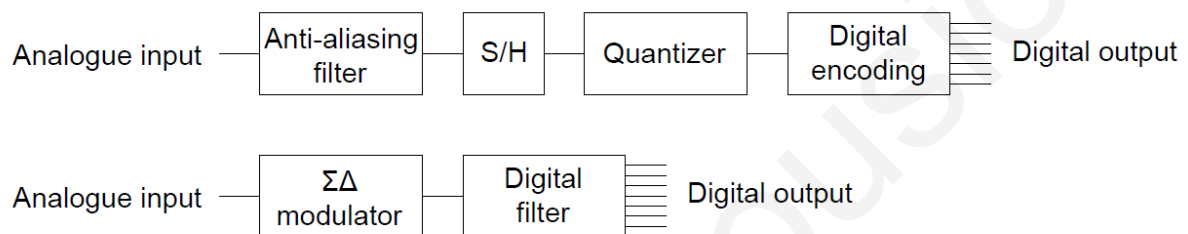


Figure 45: Regular Converter vs oversampling converter [1]

As mentioned above, one main difference that oversampling ADCs have is the elimination of the anti-aliasing filter. As one can observe from fig.46, due to the fact that the sampling takes place many times in the signal's bandwidth, the aliasing error is eliminated. Eliminating the S/H circuit can be achieved by using switched capacitor circuits which can also hold the signal. Then it is pushed into the modulator which quantizes the signal. The output is in the form of a pulse and its density represents the average value of the signal during a certain period. The digital filter is used to filter out quantization noise and its output is then sampled down to the Nyquist rate and it's the ADCs output. For an oversampling ADC, its performance can be verified by calculating the Signal-to-Noise ration and its dynamic range [1].

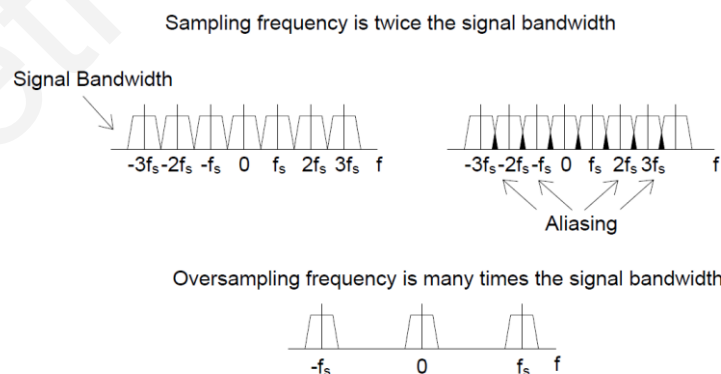


Figure 46: Sampling frequency of ordinary converter vs oversampling converter [1]

As mentioned above, the main advantage of the $\Sigma\Delta$ ADC architecture is the oversampling of the input signal resulting in high resolution data. The block diagram of the first order $\Sigma\Delta$ modulator is shown in fig.47. This architecture uses a feedback loop. In the forward path the integrator and the 1-bit ADC, which is basically a comparator, can be found. In the feedback loop there is a 1-bit DAC and its

function is to determine if positive or negative voltage reference is to be added to the summer. The letter “T” represents the time, which is the inverse of frequency and k is an integer [1].

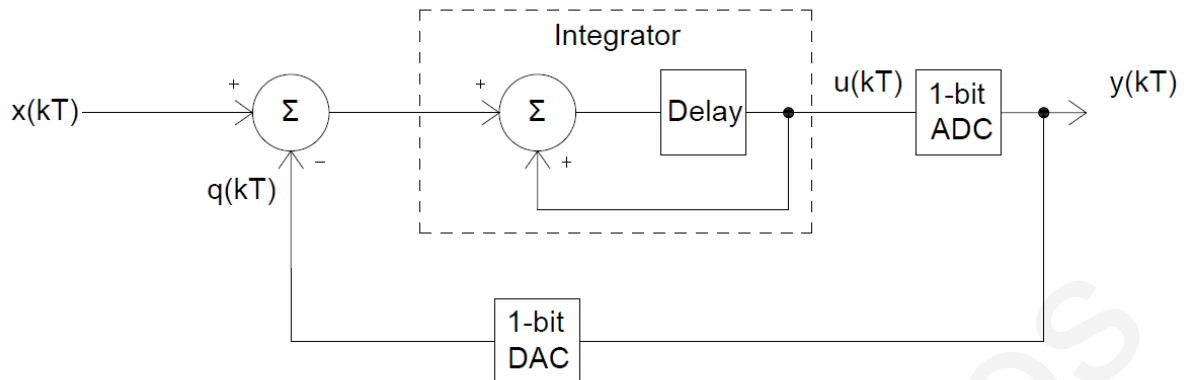


Figure 47: First order oversampling converter [1]

Another advantage of the $\Sigma\Delta$ architecture is that the quantization noise Q_e cancels itself out and this can be observed by the output equation of the block diagram seen above, yielding that $y(kT) = x(kT - T) + Q_e(kT) - Q_e(kT - T)$. In this equation, $x(kT - T)$ is the input signal with one period delay time and $Q_e(kT) - Q_e(kT - T)$ is the difference in quantization error between the previous and present values. As a result, the quantization noise Q_e cancel itself out. By converting the above block diagram to the frequency domain as can be seen in the figure below, one can confirm that the quantization noise is cancelled. The transfer function that describes the system is:

$$v_{OUT}(s) = Q_e(s) \cdot \frac{s}{s+1} + v_{IN}(s) \cdot \frac{1}{s+1}$$

The Q_e transfer functions resembles a high pass filter and the transfer function of the input signal that of a low pass filter. The signal of interest, as can be seen in the figure below, is located in the lower portion of frequencies where the noise is low, meaning that the noise is pushed out of picture, in higher bandwidth and this technique is known as noise shaping. By implementing a low pass digital filter, the quantization noise is filtered out resulting in a noise-free signal. This can be very useful for electronic designers which want to convert analogue inputs of very low values [1].

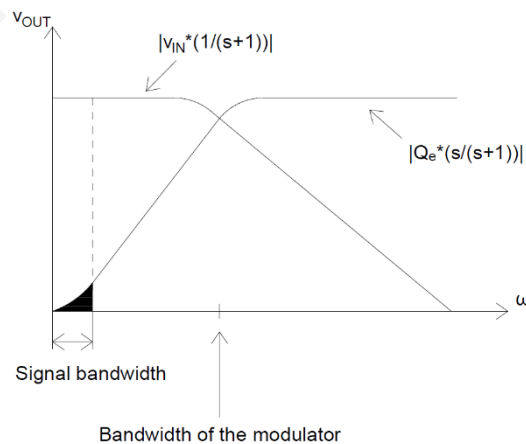


Figure 48: Frequency response of the oversampling converter at first order [1]

In some applications, higher order $\Sigma\Delta$ modulators are needed which can shape the noise even more. A second order modulator that is shown in the figure below, shapes not only the current conversion's and previous conversion's quantization noises, but also the noise of the conversion before that. Even higher order of $\Sigma\Delta$ modulators can be used such as third order. In addition, by using higher order modulators, not only the quantization noise is eliminated but also the resolution increases due to an increase in the sampling ratio [1].

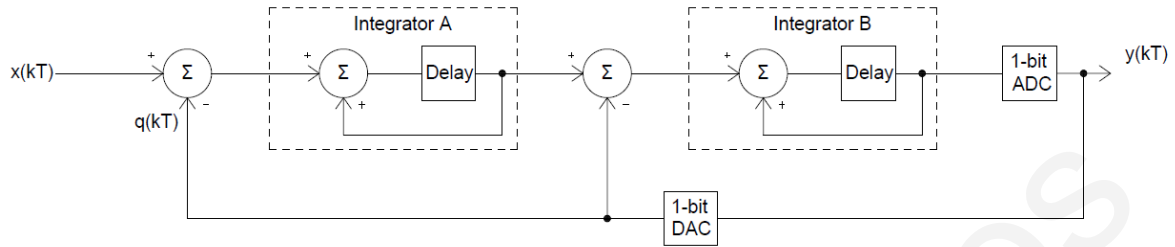


Figure 49: Second order $\Sigma\Delta$ modulator [1]

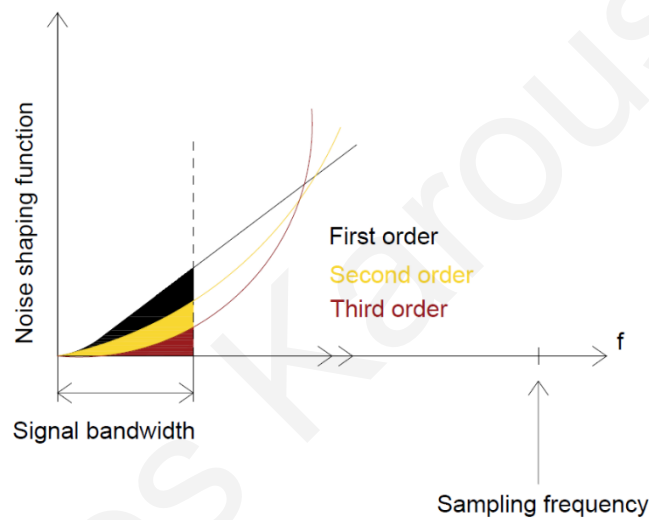


Figure 50: Example of noise shaping when comparing 1st, 2nd and 3rd order $\Sigma\Delta$ modulators [1]

SAR ADC

The Successive Approximation Register Analog to Digital Converter (SAR ADC) is a critical system in an IC design. It has medium – to high – resolution and can provide up to 5Msps sampling rates. Their resolution varies from 8 to 18 bits and they require small power consumption. The basic architecture of an 8-bit SAR ADC can be seen in Fig. 51 [30]. The 8-bit register controls the timing of the conversion. The voltage that needs conversion, V_{IN} , is sampled and compared with the output of the DAC. The Comparator's output is then fed back to the Successive Approximation Register, which its output is the actual digital conversion. The SAR algorithm for an 8-bit architecture is as follows:

- i. When the conversion starts, 1 is applied to the Shift Register and for each bit that is converted, the 1 is shifted to the right, meaning at the start of the conversion, $B[7]=1$ and $B[6..0]=0$.
- ii. $D[7]$, which is the MSB is set to 1 and $D[6..0]$ are zero.
- iii. Then the bits of the SAR, are transferred to the 8-bit DAC which are converted into analog value. The first value of the DAC, ideally, is $\frac{V_{REF}}{2}$.
- iv. The next step, is for the comparator to compare the analog output value of the DAC with the analog value that we want to convert. If $\frac{V_{REF}}{2} > V_{IN}$, the comparator sets its output to zero and

it resets D[7] to zero. If $\frac{V_{REF}}{2} < V_{IN}$, the comparator sets its output to 1 and the D[7] remains as is. D[7] is the actual MSB of the final digital output code.

- v. The 1 is shifted one bit to the right, setting B[6] to 1.
- vi. D[6] is then set to 1, having the rest of the bits equal to zero except from D[7], which has the value previously decided by the comparator. Then, the SAR output is transferred to the DAC for conversion and repetition of the previous steps [1].

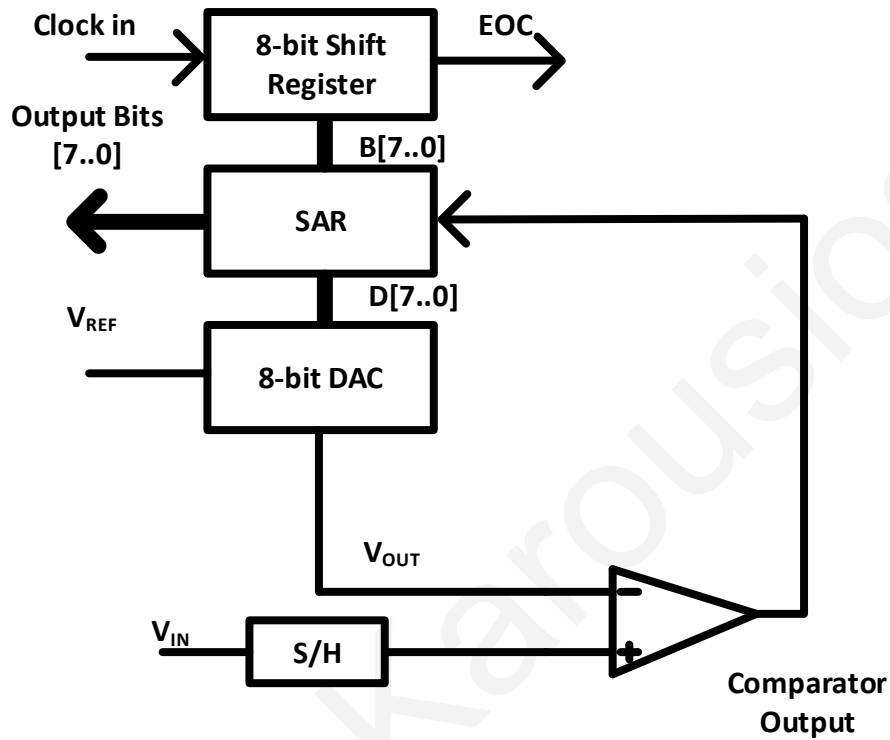


Figure 51: Example of an 8-bit SAR ADC [1]

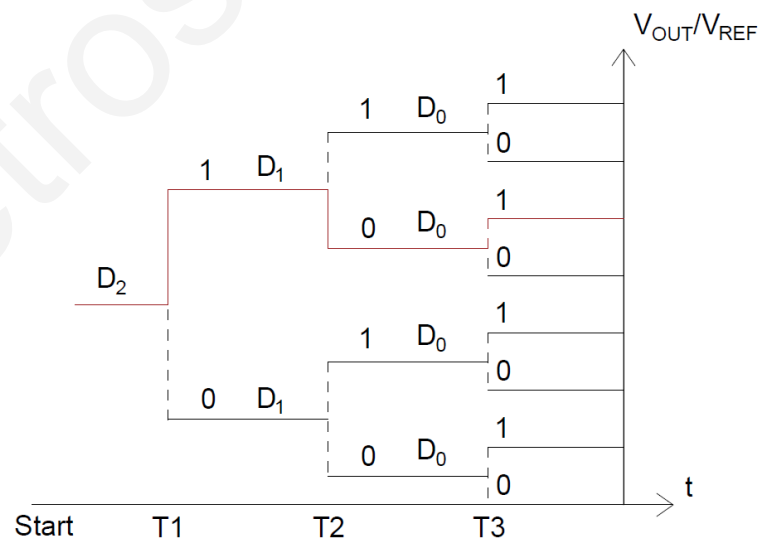


Figure 52: Output of a 3-bit SAR ADC [1]

SAR ADC, as all analog designs, has its own flaws, disadvantages. It requires N comparison periods and it cannot perform another conversion until the previous one is done. They require less power and area. However, their conversion speed and resolution are limited. An advantage of this system is

that power dissipation scales with the sample rate in contrast with flash or pipelined ADCs, which have constant power dissipation [30].

The Digital to Analog Converter and the Comparator, are the two critical components of the system. First things first. The DAC is a critical block in the SAR ADC for various reasons. One reason is the settling time of the DAC. It must settle within a resolution in time for the next digital to analog conversion, in order to maintain monotonicity. Another reason is that, for the SAR ADC to be monotonic, the DAC must be monotonic.

Many SAR ADC designs, use capacitive DACs and they are named charge-redistribution successive approximation ADC. As a result, the Sample and Hold circuit is unnecessary because capacitive DACs have an inherent track/hold function [30]. Their design consists of an array of N capacitors with binary weighted values, plus one “dummy LSB” capacitor. This charge redistribution can be seen in Fig. 53. The first step, is to discharge the capacitor array through a reset switch. When the reset switch closes, the comparator behaves like a unity gain buffer. As a result, the capacitors charge to the offset voltage of the comparator thus performing automatic offset cancellation. Next, the capacitor array samples the input voltage, V_{IN} . Then the reset switch opens and the bottom plates of each capacitor are connected to ground. As a result, the voltage at the top plate of the array is $V_{OS} - V_{IN}$. The conversion sequence will start when the SAR logic switches the bottom plate of the MSB capacitor to V_{REF} . The comparator, decides as to how the bottom plate of the MSB capacitor will be connected. If its output is low, the bottom plate is connected to ground. Otherwise, it remains connected to V_{REF} . Note that the accuracy of the DAC limits the accuracy of the ADC because the latter, depends on the DAC to provide the necessary analogue voltage to compare with the input voltage [1].

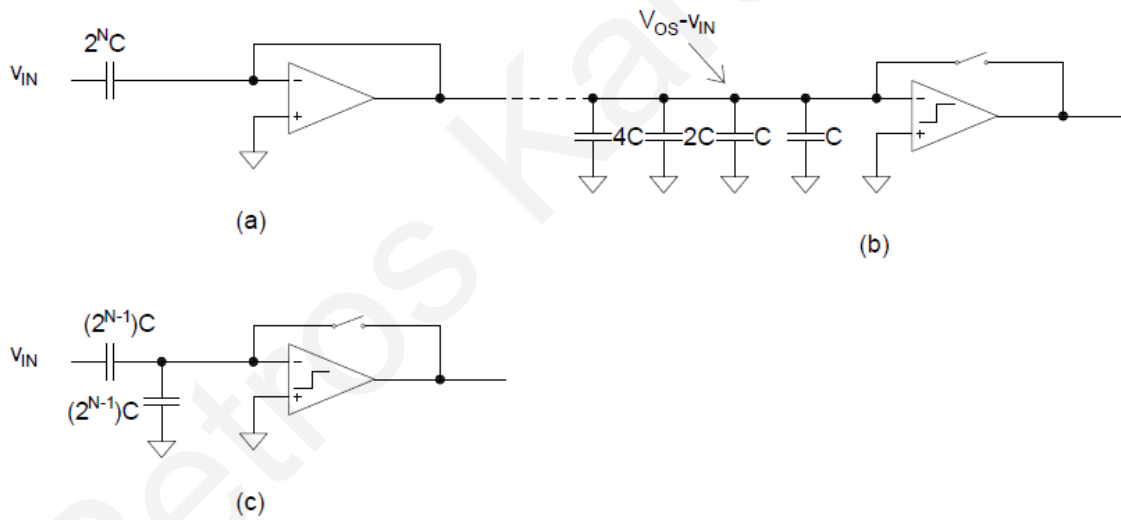


Figure 53: (a) Discharging the capacitor array and automatic offset cancellation. (b) the voltage at the top of the capacitor array (c) the equivalent circuit while converting the MSB [1]

The main issue that causes INL and DNL error, is the capacitor mismatches. Therefore, the tolerance for the ADC's accuracy is the INL being less than $\pm 0.5LSB$ [1].

$$|INL|_{max} = \frac{V_{REF}}{2} \cdot \frac{|\Delta C|_{max, INL}}{C} = \frac{V_{REF}}{2^{N+1}} = 0.5LSB$$

Which results in $|\Delta C|_{max, INL} = \frac{C}{2^N}$

Regarding DNL_{max} , again will occur when the mismatch between the capacitors is at its highest resulting in

$$DNL_{max} = \frac{(2^N - 1) \cdot V_{REF} |\Delta C|_{max,DNL}}{2^N \cdot C} = \frac{V_{REF}}{2^{N+1}} = 0.5LSB$$

Which results in $|\Delta C|_{max,DNL} = \frac{C}{2^{N+1}-2} [1]$

[31] ADC Type	Advantages	Disadvantages
Flash ADC	Fastest	Low resolution
Pipeline ADC	Fast	Low resolution
Dual-Slope ADC	High resolution Low power	Slow
$\Delta\Sigma$ ADC	Noise shaping Oversampling High resolution	Slow
SAR ADC	Medium to high resolution Medium to high speed	Require anti-aliasing filter

Table 2: Analogue to digital converters summary [31]

Design of Asynchronous SAR ADC

This SAR ADC is designed to be used in an asynchronous circuit. Its purpose is to convert the analogue temperature of the chip into digital signal to be transferred to the gateway. However, the SAR ADC must be active only when the gateway needs to know the temperature. As a result, when the gateway asks for the temperature by sending the temperature bit=1, the ADC will start to function by activating the Ring Oscillator (RO) which is needed to enable the digital logic of the converter. The chip is asynchronous; therefore, no clock will be on the chip.

The Asynchronous SAR ADC won't need to be rail-to-rail, because the temperature sensor cannot be rail-to-rail due to the limitation of voltage that the transistors require to function properly. The system's architecture is comprised by the SAR ADC, temperature Sensor, Ring Oscillator and Voltage Reference. The SAR ADC is comprised by a shift register, a Successive Approximation Register, an N-bit Digital to Analogue Converter, a comparator and a Sample and Hold Circuit [1]. The Bandgap reference voltage provides stable voltage at the RO, which, when activated, the SAR Logic of the ADC starts to function thus the conversion starts. The PTAT circuit provides the chip's temperature that will be converted. Finally, the ADC will output the required bits to be transferred to the Gateway.

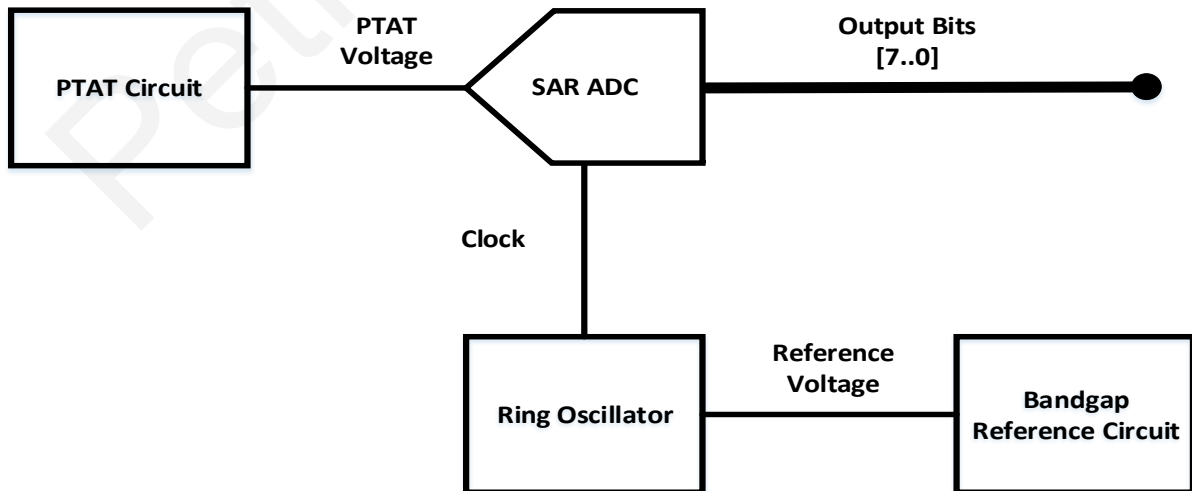


Figure 54: Proposed System Design for Asynchronous ADC

Proportional to Absolute Temperature - PTAT

The PTAT circuit provides the temperature variation of the chip. The voltage is transferred to the SAR ADC, which then converts the Analog signal to Digital.

The PTAT circuit can be implemented in various techniques. One technique is to directly connect the PTAT voltage output from the Bandgap Reference Circuit (BGR) to the Analog input of the SAR ADC. However, by implementing this technique, current might be drawn thus altering the Reference Voltage of the BGR. In addition, in the BGR, the PTAT voltage range is large enough as a PTAT temperature sensor itself. Another technique is to create a PTAT circuit, independent from the BGR circuit. Thus, the Reference Voltage of the BGR will not be altered.

A temperature sensor has some parameters, such as temperature range, sensitivity, output range, linearity, accuracy and noise immunity. [32]

Another reason as to why is better to use this technique of Chaparro et al. [33]. In this paper, they propose a high-slope PTAT temperature Sensor which has a larger range and higher slope. The higher slope is obtained by subtracting a CTAT current from a PTAT current and uses one-point calibration at 25 °C thus reducing the power and area consumption of the SAR ADC. The voltage slope of the PTAT, affects the error requirement of the ADC. In order to have monotonicity, the INL and DNL errors must be less than $\frac{LSB}{2}$. The error of the ADC is based on the constrains on the Comparator and the DAC, therefore, if the temperature sensor has higher slope, the constrains on the two previous components of the SAR ADC are relaxed. Also, the low power consumption can be achieved when the output of the sensor is close to zero voltage.

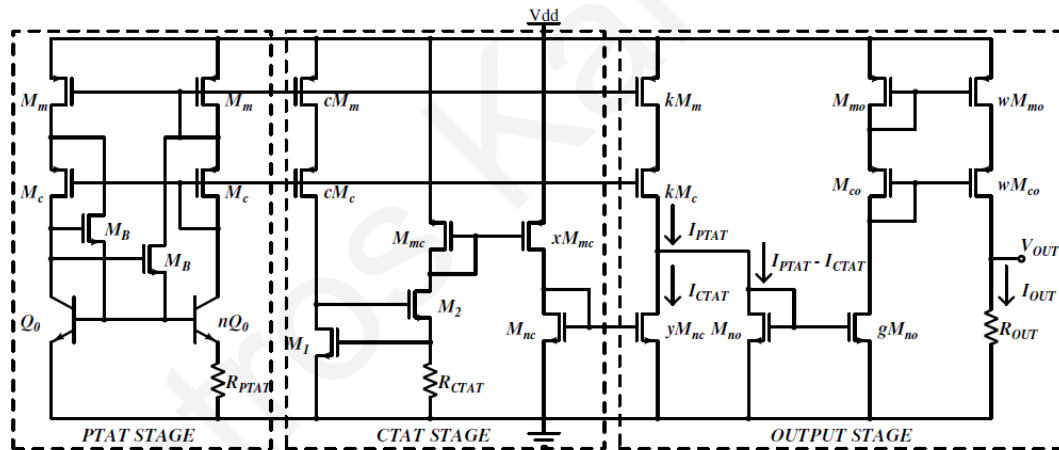


Figure 55: Proposed high-slope PTAT Temperature Sensor [33]

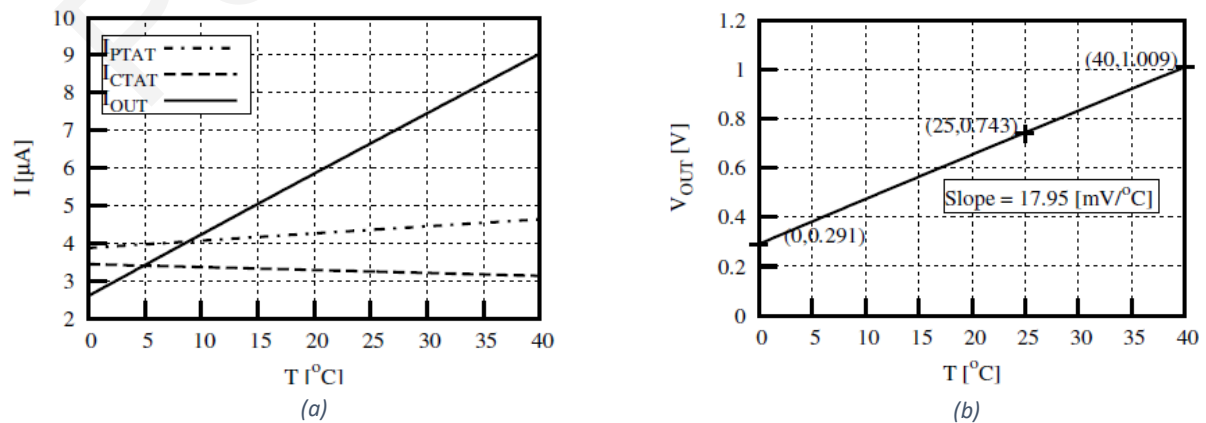


Figure 56: Simulated (a) current and (b) voltage versus temperature [33]

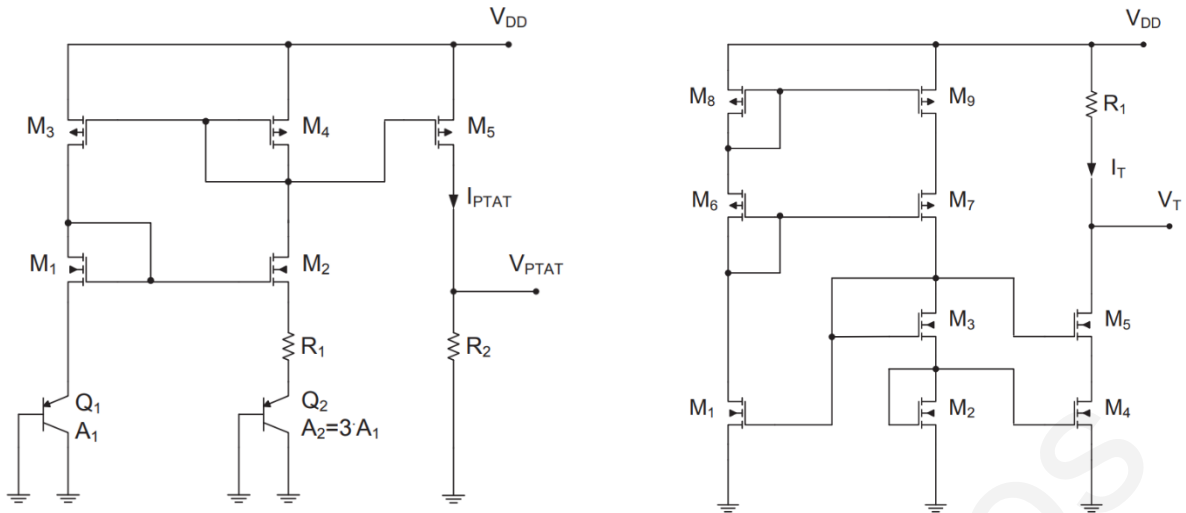


Figure 57: Schematics of (a) BJT PTAT and (b) CMOS PTAT [34]

The disadvantage of the above technique, is that in the paper the authors only test this device from 0-40 degrees Celsius. Therefore, no other data can be found for this design about variations in temperature.

The authors from [34], implemented two types PTAT sensors, one with Bipolar transistors and one with MOS transistors only. In the latter design, the technique they used relies on the thermal dependencies of the MOSFETs. In the bipolar method [35] that they implemented, the design had 5.31 mV/K average sensitivity and it was significantly smaller than that of the second design which had 9.23 mV/K average sensitivity. However, comparing the BJT and the CMOS designs, the latter was not as linear as the first design. The authors, implemented a calibration circuit therefore increasing the sensitivity of the BJT temperature sensor, making it similar to the CMOS design. In their implementation, they used a calibration technique that improved the slope of the PTAT voltage, making it similar to that of the CMOS PTAT sensor. The drawback of their design is that it requires larger area than other designs.

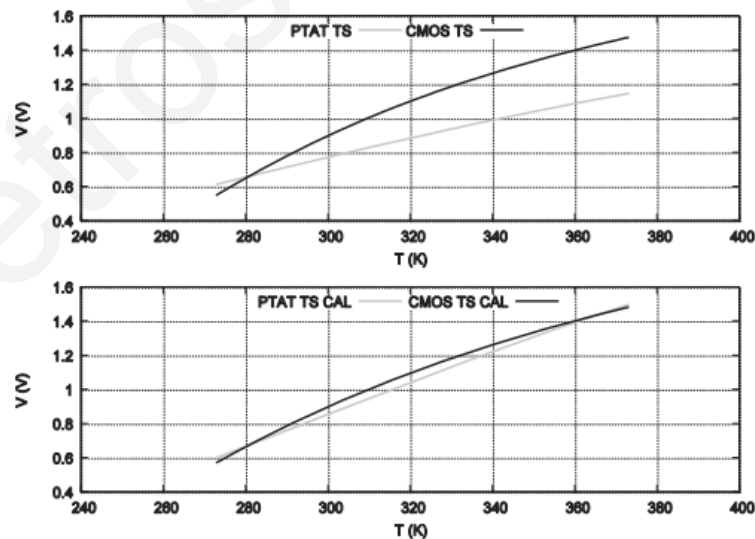


Figure 58: Results before and after the BJT PTAT calibration [34]

The conventional PTAT sensor can be found in [35],[32],[36] and in the following figure. [32]The BJTs are connected as diodes. They have different areas and conduct the same current. The principle of operation of the conventional PTAT sensor is based on the phenomenon that the difference

in the voltage of the above BJTs is proportional to absolute temperature. As can be seen from fig. 59a, the difference in the voltage is converted to current and it is amplified by means of a current mirror. This design has good accuracy and small area. However, it does not contain operational amplifier because it will need more space.

Regarding the following figure, transistors M1, M2, M3 and M4 are identical as to have the same current in each branch. If V_X and V_Y have the same voltage potential, then the following equation states the drain current of M1 and M2

$$I_1 = I_2 = \frac{V_T \cdot \ln(n)}{R_1} \quad (1)$$

Where

$$V_T = \frac{kT}{q} \quad (2)$$

k is the Boltzmann's constant, q is the electron charge, n is the ratio between p-n junction areas and T is the absolute temperature. The PTAT current is

$$I_{PTAT} = I_5 = \frac{W_5}{W_4} \cdot I_2 = \frac{W_5}{W_4} \cdot \frac{k \cdot T \cdot \ln(n)}{q \cdot R_1} \quad (3)$$

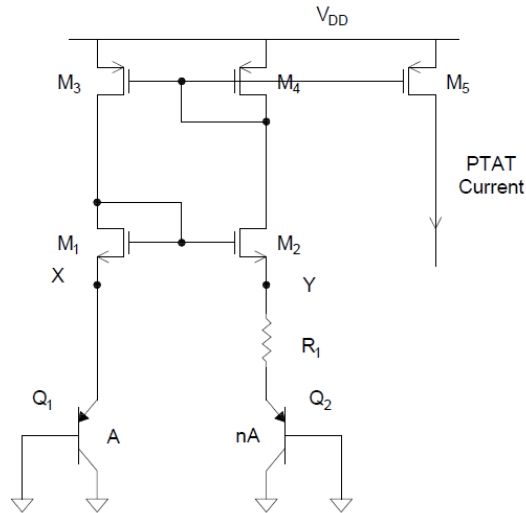
$$V_{PTAT} = I_5 \cdot R_2 = \frac{W_5}{W_4} \cdot \frac{k \cdot T \cdot \ln(n) \cdot R_2}{q \cdot R_1} \quad (4)$$

$$P = (2 + \frac{W_5}{W_4}) \cdot I_2 \cdot V_{DD} \quad (5)$$

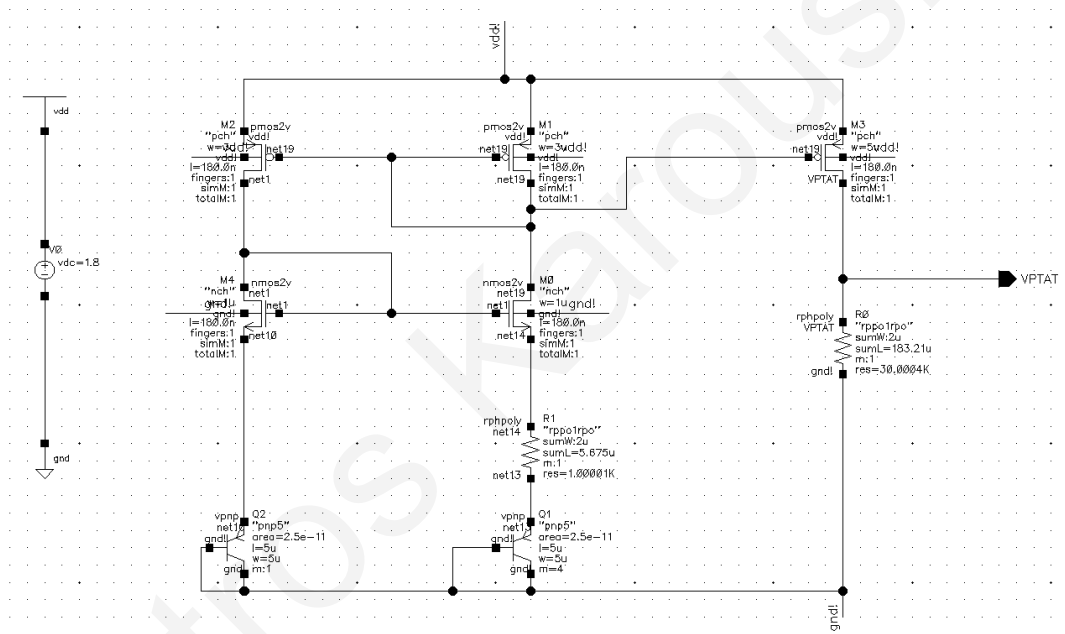
Trying out the design in fig. 59b, I obtained voltages from 285.45mV to 1.4504V, 1.165V range and maximum current at 110.91μA. However, if each one of the resistances is doubled, therefore increasing the area, the current is decreased to 70μA and the range is increased to 1.225V, 435.55mV to 1.6608V.

Device	Width (μm)	Length (μm)
M3, M4	4	0.5
M2, M1	1	0.5
Q1	5	5
Q3 (8xQ1)	5x8	5x8
R1 - 10KΩ	2	60.77
R2 - 130KΩ	2	795.39
M5	5.5	0.5

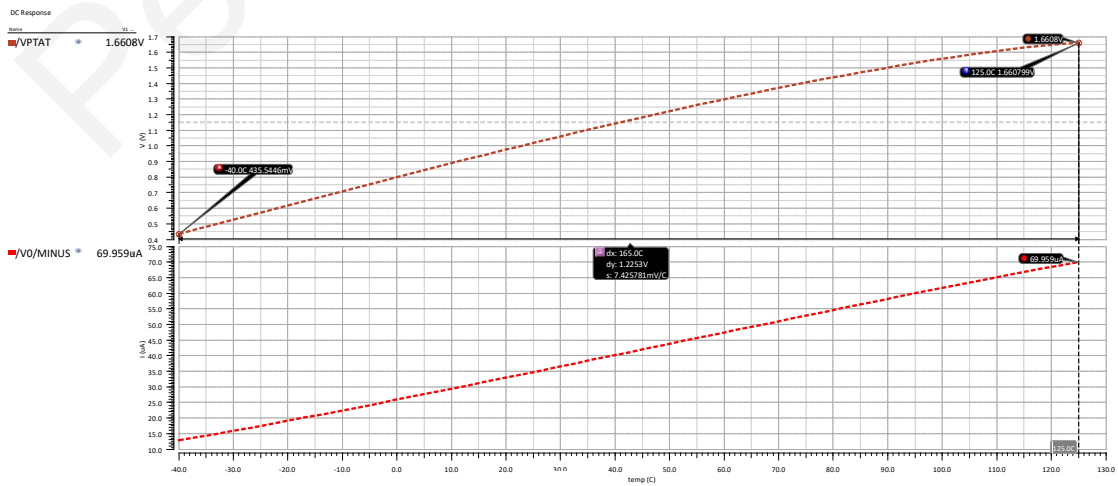
Table 3: Device sizes



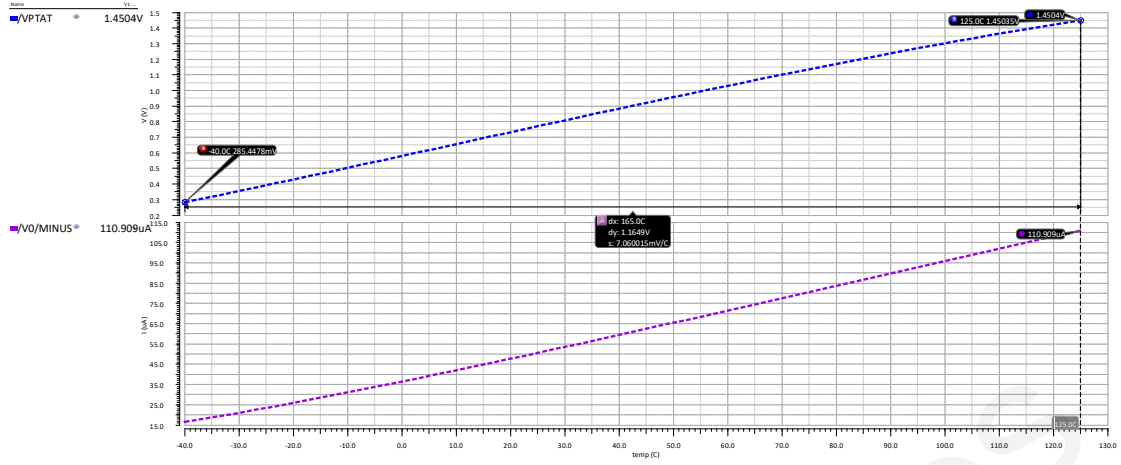
(a)



(b)



(c)



(d)

Figure 59: (a) PTAT system from [35] (b) PTAT system. (c) PTAT voltage and current with small resistances. (d) PTAT voltage and current with double the size resistances.

Successive Approximation Register (SAR) Logic

The SAR logic controls the timing of the input signal to be converted and its corresponding output. It is implemented by using D flip-flops (DFFs) in two rows. In the top row, the DFFs operate as a ring counter whilst the DFFs in the bottom row are responsible for not only to control the DAC switches, but also to store the result from the comparator during the conversion process [37].

There are two different ways of implementing the SAR logic. The first one, as mentioned above, is implemented by using DFFs in two rows, in which the first row the DFFs act as ring counter and in the second row the DFFs act as a code register. In this type of SAR logic, $2N$ DFFs are used. In the other approach, N DFFs are used additionally with some other logic gates, with N representing the bits of conversion. In this thesis, the first approach is explained, because, not only it can be implemented easier but also has lower power consumption with higher frequencies or higher supply voltages[38].

In order for a full conversion to take place, $N + 2$ clock cycles are needed. During the first cycle, SAR logic is reset whilst all its outputs are set to zero. The next N cycles are used for the conversion of the input analogue signal. Finally, the last cycle stores the results of the conversion. A Finite State Machine can describe the ADC's logic operation, which has as an output the result of the comparator [38].

Cycle	Sample	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	Comp
0	1	0	0	0	0	0	0	0	0	0	0	-
1	0	1	0	0	0	0	0	0	0	0	0	A9
2	0	A9	1	0	0	0	0	0	0	0	0	A8
3	0	A9	A8	1	0	0	0	0	0	0	0	A7
4	0	A9	A8	A7	1	0	0	0	0	0	0	A6
5	0	A9	A8	A7	A6	1	0	0	0	0	0	A5
6	0	A9	A8	A7	A6	A5	1	0	0	0	0	A4
7	0	A9	A8	A7	A6	A5	A4	1	0	0	0	A3

Ring Oscillator [1]

In a clockless circuit, a way to emulate the clock signal is by implementing a Ring oscillator, which is sticking an odd number of identical inverters in the circuit forming a closed loop. It provides a positive feedback and its frequency can be found by

$$f_{osc} = \frac{1}{n \cdot (t_{PHL} + t_{PLH})}$$

Whereas t_{PHL} and t_{PLH} are the delay time between the 50% points of the input and output. Their name depends as to whether the output changes from high to low or from low to high. Additionally, n is the number of the inverters in the circuit.

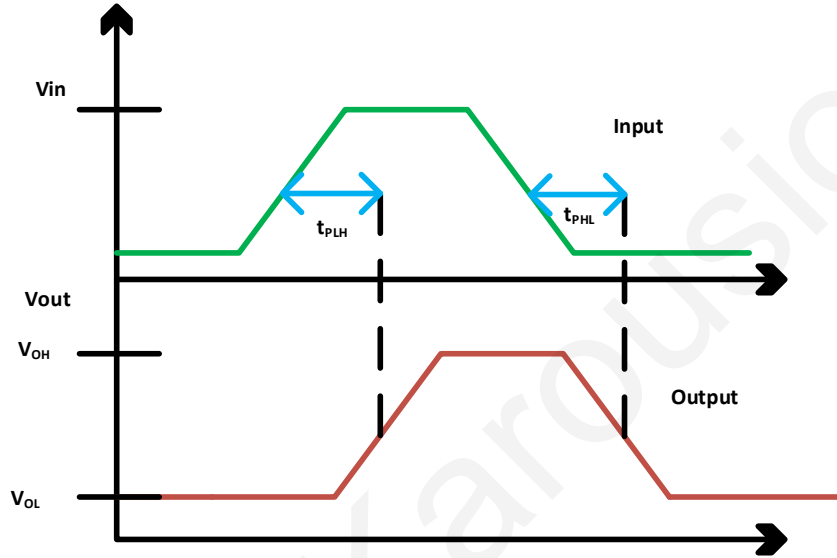


Figure 64: Ring Oscillator delay [1]

The delay of the Ring Oscillator is

$$t_{PLH} + t_{PHL} = 0.7(R_n + R_p) \cdot C_{tot}$$

Where

$$C_{tot} = C_{oxp} + C_{oxn} + \frac{3}{2} \cdot (C_{oxp} + C_{oxn}) = \frac{5}{2} \cdot (C_{oxp} + C_{oxn})$$

The PMOS devices are responsible for the power dissipation of the ring oscillator when they operate. Considering that the input of the inverter has as an input, a pulse with period T and frequency f_{clk} , the average power dissipation of the inverter can be found by [1]:

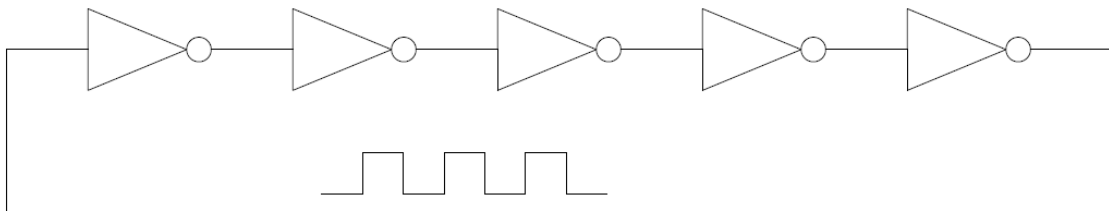


Figure 65: Schematic design of a five-stage ring oscillator [1]

$$P_{avg} = VDD \cdot I_{avg} = \frac{C_{tot} \cdot VDD^2}{T} = C_{tot} \cdot VDD^2 f_{clk}$$

The current-starved VCO, is a technique which designers choose to implement in their designs. Its main advantage is the limitation in current consumption. As can be seen from Fig.66, MOSFETS M2 and M3 operate as an inverter. M1 and M4 operate as current sources, which control and eventually limit the current through the inverter making it starve for current. The current of M5 and M6, is the same and it is set by the input voltage and it is mirrored in each inverter. The system operates when the input voltage $V_{inVCO} > V_{THN}$, meaning that the oscillation frequency is zero [1]. The total capacitance on the drains of M2 and M3 is:

$$C_{tot} = \frac{5}{2} \cdot C_{ox}(W_p \cdot L_p + W_n \cdot L_n)$$

The current-starved Voltage Controlled Oscillator comprised by a number of N devices (remember, must be an odd number, higher than 5 devices) has an oscillation frequency that can be found by:

$$f_{osc} = \frac{I_D}{N \cdot C_{tot} \cdot VDD}$$

Where N is the odd number of devices and $I_D = I_{D1} = I_{D4}$

The average power dissipated by the voltage-controlled oscillator, not including the power dissipated by the MOSFETs M5 and M6 and the equation to describe it from [1] is:

$$P_{avg} = VDD \cdot I_{avg} = N \cdot VDD \cdot C_{tot} \cdot f_{osc}$$

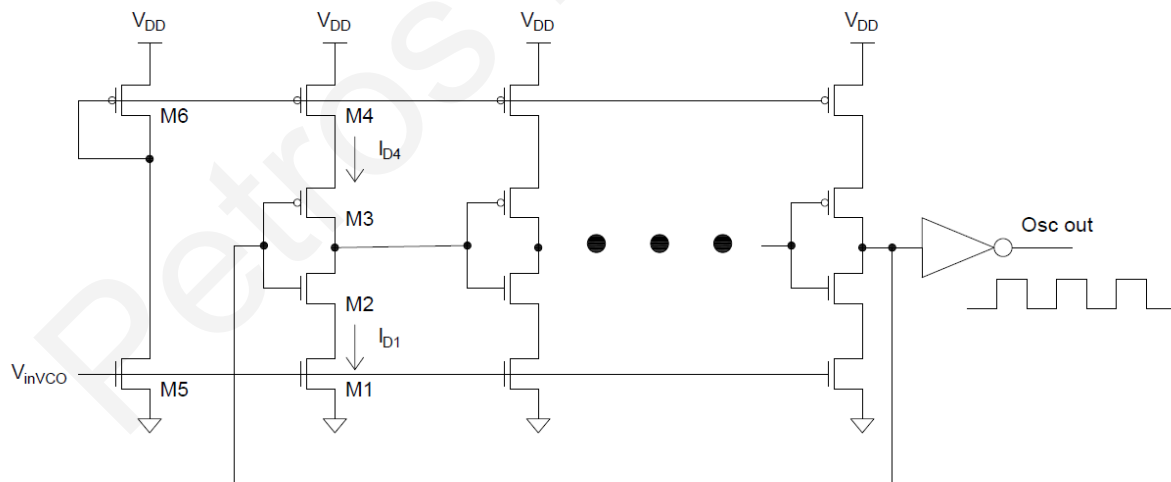


Figure 66: Schematic of the current-starved voltage control oscillator [1]

Bandgap Reference

Voltage Reference is a circuit that provides a fixed voltage, which cannot be altered by variations either in supply voltage VDD, temperature or process variations. However, a designer sometimes prefers a circuit that the voltage changes according to the temperature variation. There are two types of reference voltages: the first type, is called PTAT, which is the acronym to Proportional To Absolute

Temperature, the reference voltage increases when the temperature rises. The second type of reference voltage is called CTAT or Complementary To Absolute Temperature. Bandgap reference voltage is created when these two reference voltages are put together, meaning that the temperature will not vary the slightest and can be achieved by using parasitic diodes. Due to the fact that in a CMOS process, at the manufacturing stage, the electrical characteristics of the parasitic diode cannot be controlled, the reference voltages are comprised by using MOSFETs connected as resistors [1].

The voltage that is independent of process, temperature and power supply variations, it is a simple combination of PTAT circuit as described in [39], with voltage V_1 and CTAT circuit, with voltage V_2 . This means that since each circuit has a temperature coefficient (TC) opposing to the other circuit, the total TC is zero. The coefficients α_1 and α_2 are chosen as to satisfy the equation

$$\frac{\alpha_1 \partial V_1}{\partial T} + \frac{\alpha_2 \partial V_2}{\partial T} = 0$$

Which for zero TC, the above equation concludes that the reference voltage becomes

$$V_{REF} = \alpha_1 V_1 + \alpha_2 V_2$$

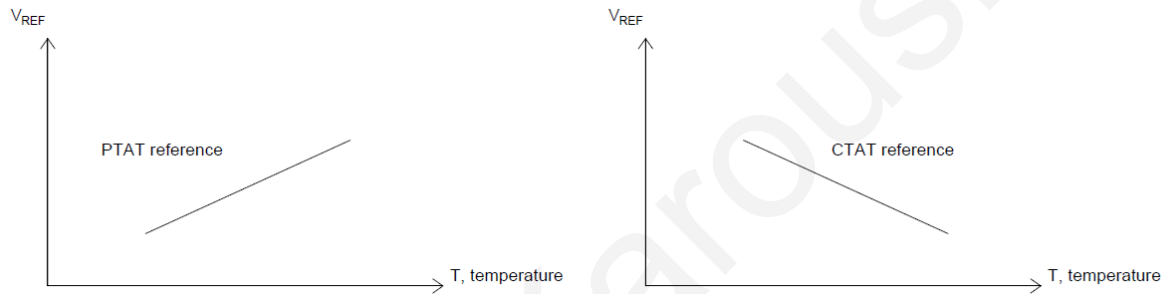


Figure 67: Reference voltages of PTAT and CTAT [1]

For the negative TC, designers can use the forward voltage of a p-n junction diode, which can be found in the base-emitter bipolar transistors. The current for a bipolar device is:

$$I_C = I_S \exp\left(\frac{V_{BE}}{V_T}\right)$$

Where I_S is the saturation current proportional to $\mu k T n_i^2$. The μ represents the minority carriers' mobility, n_i is the concentration of silicon's intrinsic minority carriers and k is the Boltzmann's constant. Additionally, the thermal voltage V_T ($\approx 26\text{mV}$ at room temperature) can be found by:

$$V_T = \frac{kT}{q}$$

By keeping I_C constant, the derivative of the Base-Emitter voltage can be found by:

$$\frac{\partial V_{BE}}{\partial T} = \frac{V_{BE} - (4 + m) \cdot V_T - \frac{E_g}{q}}{T}$$

Where T is the absolute temperature, q is the electron charge, $m \approx -\frac{3}{2}$ and the bandgap energy of silicon is $E_g \approx 1.12\text{ eV}$. By solving this equation, one can find the temperature coefficient of the base-emitter voltage at a specific temperature. One can identify that the voltage of the Base-Emitter of the bipolar transistor has can alter the temperature coefficient of the device. Assuming $T = 300^\circ\text{K}$ and Base/Emitter voltage $V_{BE} = 750\text{mV}$, the temperature coefficient $\frac{\partial V_{BE}}{\partial T} \approx -1.5\text{ mV}/^\circ\text{K}$ [39].

Regarding the Positive Temperature Coefficient Voltage mentioned in [39][40], this can be achieved by placing two identical transistors as can be seen in the following figure, with different biasing collector current of nI_0 and I_0 . Keeping in mind that the base currents are of no importance, the difference between the two Base/Emitter voltages can be found by

$$\begin{aligned}\Delta V_{BE} &= V_{BE1} - V_{BE2} \\ &= V_T \ln \frac{nI_0}{I_{S1}} - V_T \ln \frac{I_0}{I_{S2}} \\ &= V_T \ln(n)\end{aligned}$$

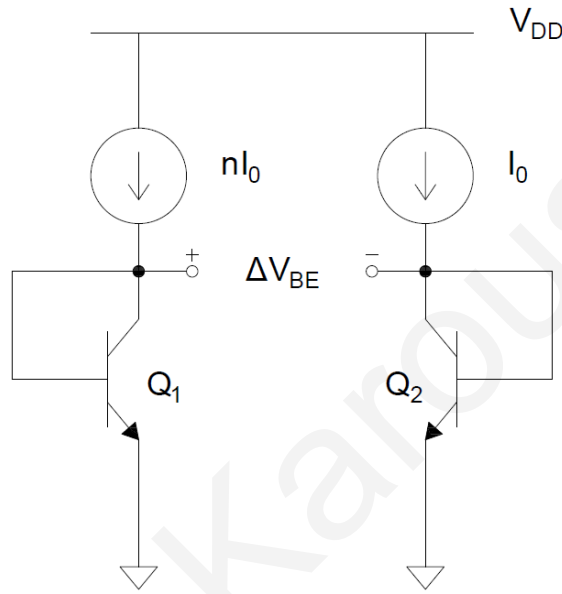


Figure 68: PTAT generator [35]

By differentiating in parts, the above equation, we can find that the temperature coefficient is independent of temperature and collector currents and results into:

$$\frac{\partial \Delta V_{BE}}{\partial T} = \frac{k}{q} \ln(n)$$

The above equation, when solved gives result of:

$$\frac{\partial \Delta V_{BE}}{\partial T} = +0.087 \cdot \ln(n) \text{ mV}/^\circ\text{K}$$

The reference voltage of the PTAT is an addition of the Base/Emitter voltage and the voltage variation

$$V_{REF} = V_{BE} + V_T \cdot \ln(n)$$

Considering that the negative temperature coefficient has value of $\frac{\partial V_{BE}}{\partial T} = -1.5 \text{ mV}/^\circ\text{K}$, in order for the positive temperature coefficient mentioned above to reach the aforementioned value in order to cancel each other, the value of $\ln(n)$ must be 17.2, which results into $n \approx 2.95 \times 10^7$. This factor is large and impractical to be implemented in a circuit design. Therefore, by substituting the above value for n and the value of $V_{BE} \approx 750 \text{ mV}$ into the equation shown below from [39], results into:

$$V_{REF} = V_{BE} + 17.2 \cdot V_T \approx 1.25 \text{ V}$$

Another equation from [40] for the reference voltage is:

$$V_{REF} = \frac{E_g}{q} + (4 + m) \cdot V_T$$

In order to partially solve the problem of $n = 2.95 \times 10^7$ is to amplify $\Delta V_{BE} = V_T \cdot \ln(n)$ before it is added to V_{BE} . The solution can be seen in the following figure, in which the non-inverting amplifier has a gain of $1 + \frac{R_2}{R_3}$. The Base/Emitter voltage V_{BE1} of the first diode connected bipolar transistor is amplified by that gain, and the Base/Emitter voltage V_{BE2} is amplified by $gain = -\frac{R_2}{R_3}$. In the configuration below, $R_1 = R_2$ in order for the two BJTs to have equal current [40]. The output of the circuit becomes

$$V_{OUT} = V_{BE1} + \frac{R_2}{R_3} \cdot V_T \cdot \ln(n) = V_{BE2} + (1 + \frac{R_2}{R_3}) \cdot V_T \cdot \ln(n)$$

Now it is easier to choose values in order to satisfy $(1 + \frac{R_2}{R_3}) \cdot V_T \cdot \ln(n) = 17.2$. However, it is wiser to reduce the value of 17.2 to 16 since the collector current is not actually constant but rather is a PTAT circuit. By reducing the value as mentioned above, this mitigates that issue. The use of BJTs is preferred over the use of MOSFETs despite the popularity of CMOS technology, since the latter devices have higher threshold mismatches, resulting in non-negligible error [40].

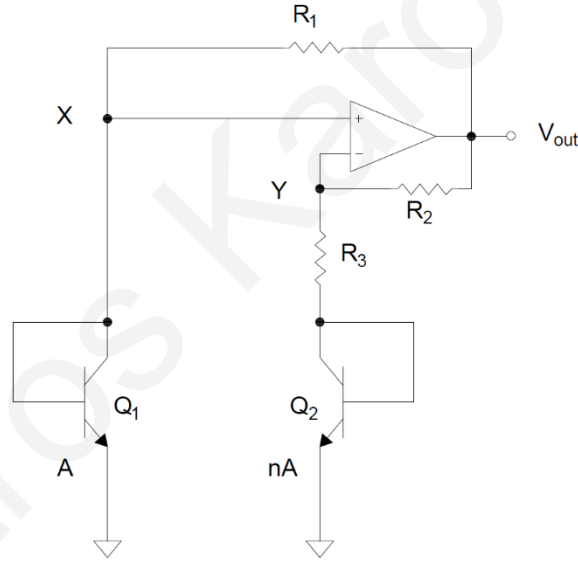


Figure 69: Circuit implementation of a bipolar bandgap circuit proposed by Kujik [40]

The above design has three issues: firstly, the designer of the operation amplifier has to deal with tradeoffs between power, stability and gain. Secondly, the implementation of Bipolar transistors with collectors not connected to ground is difficult to be implemented. This is due to the fact that in CMOS process the p-substrate usually is connected to ground and the collector of a PNP transistor is implemented by using p-substrate. Finally, the offset of the operation amplifier can alter the voltage output of the circuit [40].

The above problems can be faced by implementing the circuit below. Transistors M_1 and M_2 are exactly the same thus the equations for ΔV_{BE} and V_{OUT} can be used. One of the obstacles in this design, is the channel length modulation that introduces current mismatch error. This can be tackled by introducing resistors $R_2' = R_2$ which ensure that $V_{DS1} = V_{DS2}$. Another problem mentioned above, is

the operation's amplifier offset, which can be improved by creating a circuit with a high n value, in the vicinity of 20-30 or by altering the CMOS transistors' width. To elaborate, if $(\frac{W}{L})_1 = m(\frac{W}{L})_2$ the branch of the M_1 has current density larger by a factor of m than the current density of the other branch. As a result, the output voltage can be found by:

$$V_{out} = V_{BE2} + (1 + \frac{R_2}{R_3}) \cdot V_T \cdot \ln(n \cdot m)$$

The above equation can make a designer's life easier by choosing smaller gain for the operation amplifier, thus ameliorating the offset voltage [40].

The above circuit exhibits some unwelcomed effects despite being robust. One of them is the flicker and thermal noise that is introduced by the MOSFETs as well as by the operational amplifier. The solution is to implement large MOSFETs. Additionally, this BGR circuit cannot drive large capacitive loads because the output becomes unstable. Finally, if the output of the op-amp has high output, the circuit will not power up, leaving the two brunches off. This can be resolved by adding a start-up brunch, as can be seen in the above figure. If the two brunches can be imagined as a seesaw balancing on the op-amps output, they need a push to start in order to balance. As a result, the start-up circuit will produce an imbalance at the op-amp's output, pushing it to go low. The branch that it is implemented by using two MOSFET devices, which is connected at node "X", turns off only after the circuit has a stable output [40].

The authors in [41] proposed a novel bandgap reference circuit implemented by using 0.35 CMOS technology. The proposed circuit has $0.08 ppm/^{\circ}C$ with reference voltage of $600 mV$. In this paper [42], the authors achieved a $2 ppm/^{\circ}C$ temperature compensation for a range of $160^{\circ}C$ by using a 4-bit trimming circuit, in order to compensate for process variation.

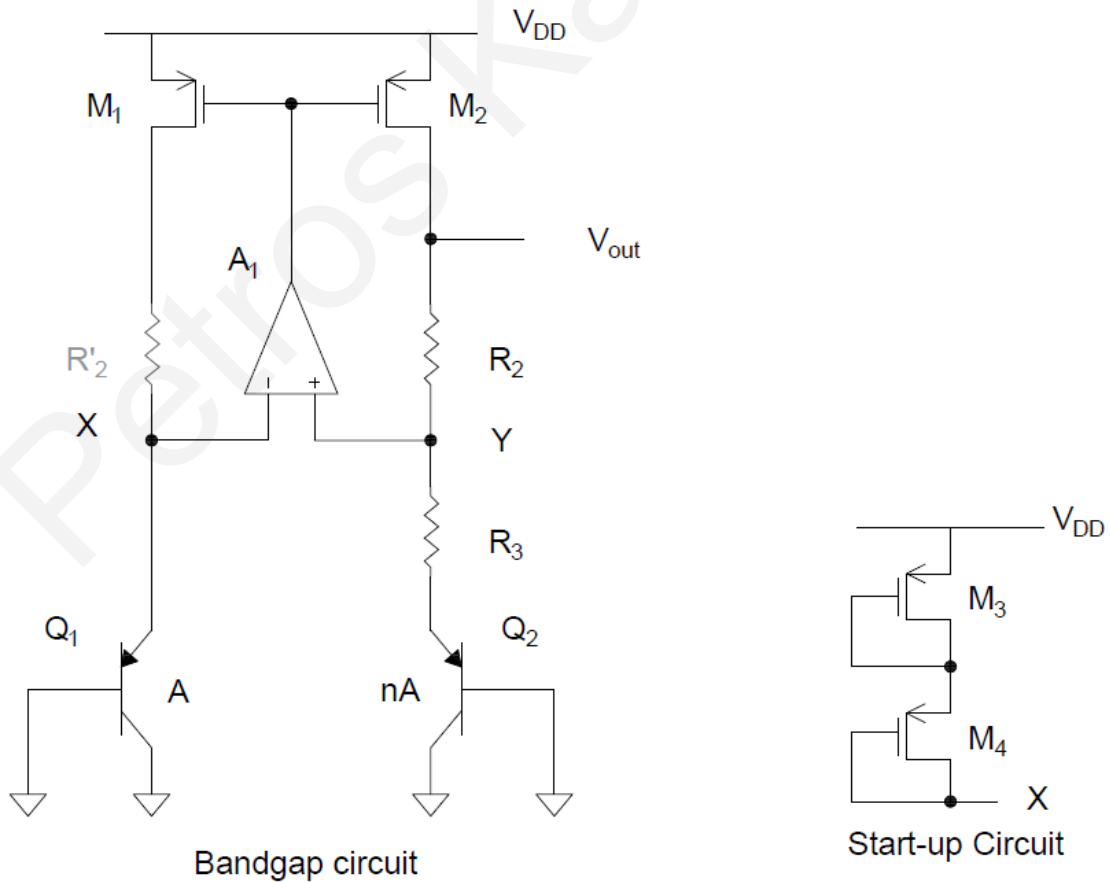


Figure 70: CMOS bandgap circuit with start-up circuit [40]

[1] Comparator

The comparator is used to determine if the input voltage of the ADC is of higher voltage than the output of the DAC that is integrated in the ADC. If the input voltage at v_p is of higher voltage than the input voltage at v_m , the output of the comparator is “1”. Otherwise, it is “0”. This can be seen in the following figure, which shows the comparator’s symbol and its operation.

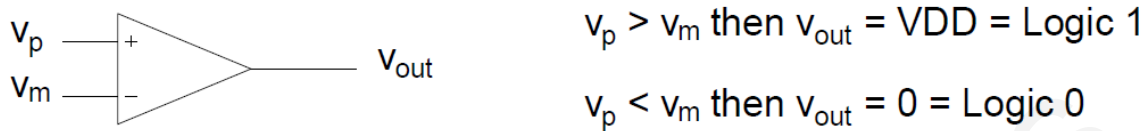


Figure 71: Comparator symbol and its operation [1]

A comparator is comprised by three stages: preamplification, decision circuit, also called positive feedback and postamplification. A designer has to face challenges when designing a comparator such as power dissipation, propagation delay and its gain. The first stage of the system is used to amplify the input signal thus improving the comparator’s sensitivity. Additionally, shields the input of the comparator from switching noise that is produced from the second stage, also known as kickback noise. This noise is pushed or “kicked” in the input of the circuit when the state of the latches changes. The second stage, as per its name, is the decision stage of the comparator. Finally, the third stage is responsible for amplifying and passing out the output signal.

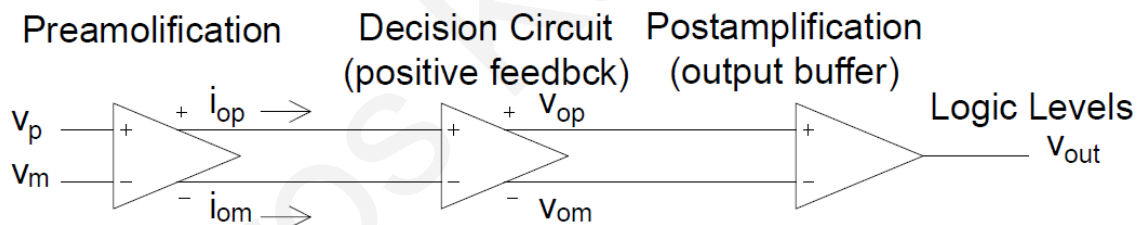


Figure 72: Three stages of the comparator [1]

The preamplification stage is comprised by using a differential amplifier with active loads. The gain of this stage is determined by the transconductance of the differential amplifier, g_m , thus transistors M1 and M2 sizes are set accordingly. Additionally, these transistors are responsible for the input capacitance of this stage, which, in addition with the low resistive nodes in this circuit, can determine the speed of the stage. The equation of this stage is:

$$i_{op} = \frac{g_m}{2}(v_p - v_m) + \frac{I_{SS}}{2} = I_{SS} - i_{om}$$

Whereas i_{op} is the positive small signal AC current of the circuit, i_{om} is the negative small signal AC current, v_p and v_m are the positive and negative input signals. [39] The transconductance describes the change of the current I_D when there is a change in the Gate/Source voltage V_{GS} , and the transconductance of the NMOS device can be found by:

$$g_m = \mu_n C_{ox} \frac{W}{L} (V_{GS} - V_{TH}) = \sqrt{2\mu_n C_{ox} \frac{W}{L} I_D} = \frac{2I_D}{V_{GS} - V_{TH}}$$

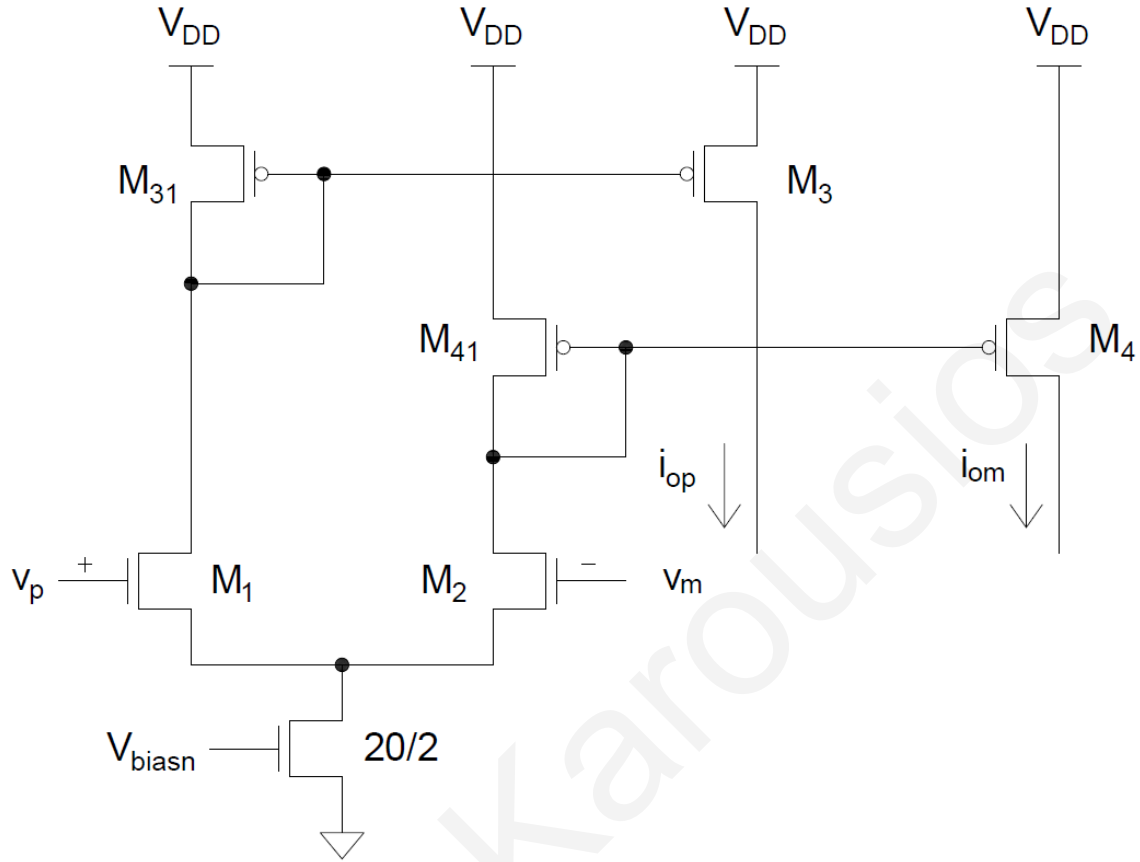


Figure 73: Preamplification stage of the comparator comprised by differential amplifier with active loads [1]

The second stage of the comparator which is the decision circuit is responsible for comparing the analogue inputs and determining which one is of highest voltage. Additionally, this circuit uses positive feedback by cross connecting the gates of M6 and M7, as can be seen in the figure below, thus increasing the gain of this stage [1].

The analyses of this circuit start by assuming that the small signal AC currents $i_{op} \gg i_{om}$ meaning that $v_{om} = 0$, devices M5 and M7 are ON whilst devices M6 and M8 are OFF. This is because the $V_{GS8} = V_{GS6} = 0$ and $V_{GS5} = V_{GS7}$ which are higher than V_{THN} . Another condition as can be seen in the figure, is that transistors M5 and M8 (represented as A) are identical and M6 and M7 (represented as B) are identical. The analogue small signal output v_{op} can be found by:

$$v_{op} = \sqrt{\frac{2 \cdot i_{op}}{\beta_A}} + V_{THN}$$

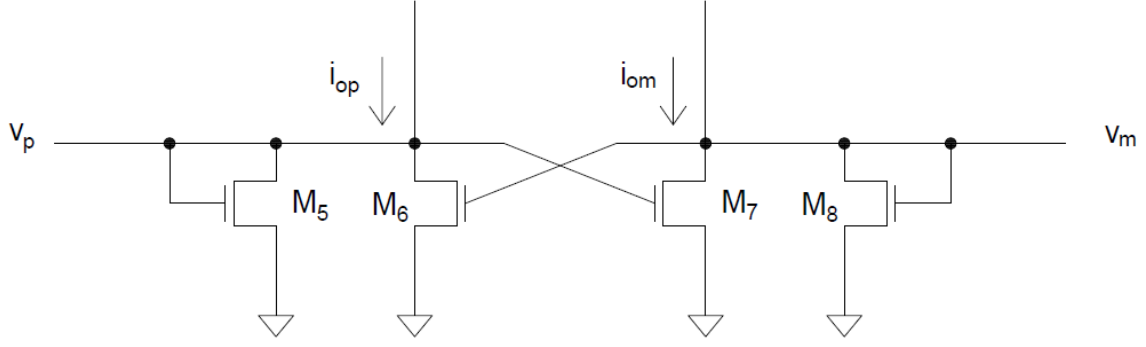


Figure 74: Decision circuit with cross-gate connection as positive feedback decision circuit [1]

The circuit functions as a seesaw. When i_{op} decreases, i_{om} increases and the Gate/Source voltage of the M8 increases to V_{THN} thus its state changes to ON. Additionally, M6 starts to switch on resulting in drawing current and M7 turns OFF. The currents of the M7 and M5 respectively, when M8 and M6 are not operating, can be found by:

$$i_{om} = \frac{\beta_B}{2} \cdot (v_{op} - V_{THN})^2$$

$$i_{op} = \frac{\beta_A}{2} \cdot (v_{op} - V_{THN})^2$$

$$i_{op} = \frac{\beta_A}{\beta_B} \cdot i_{om}$$

The switching of the decision circuit occurs when the small signal currents are equal, if the transconductance parameters are $\beta_A = \beta_B$. The switching point of the circuit can be calculated by:

$$V_{SPH} = -V_{SPL} = v_p - v_m = \frac{I_{SS}}{g_m} \cdot \frac{\frac{\beta_B}{\beta_A} - 1}{\frac{\beta_B}{\beta_A} + 1} \text{ for } \beta_B \geq \beta_A$$

The final stage of the comparator is the post-amplification circuit, that converts the output of the decision circuit to a logic state, 0 or VDD. The following circuit is a self-biased differential amplifier and is used as an input buffer and can be seen in the following figure. The circuit is self-biased due to the fact transistors M6 and M3 have their gates tied together. Assuming $V_{inp} > V_{inm}$ the current flowing in M2 device is larger than the current flowing in M1 device because it has larger Gate/Source voltage. Its current is mirrored by M4 through M3. Consequently, the current of M2 is higher than the current of M4 resulting the output node to have output logic LOW thus forcing the output of the inverter to be HIGH. This condition changes when the devices M2 and M4 have equal currents. Its operation is to amplify the difference between the two inputs. The operation of this buffer can be summed up by:

$$V_{inp} > V_{inm} \text{ output logic HIGH}$$

$$V_{inp} < V_{inm} \text{ output logic LOW}$$

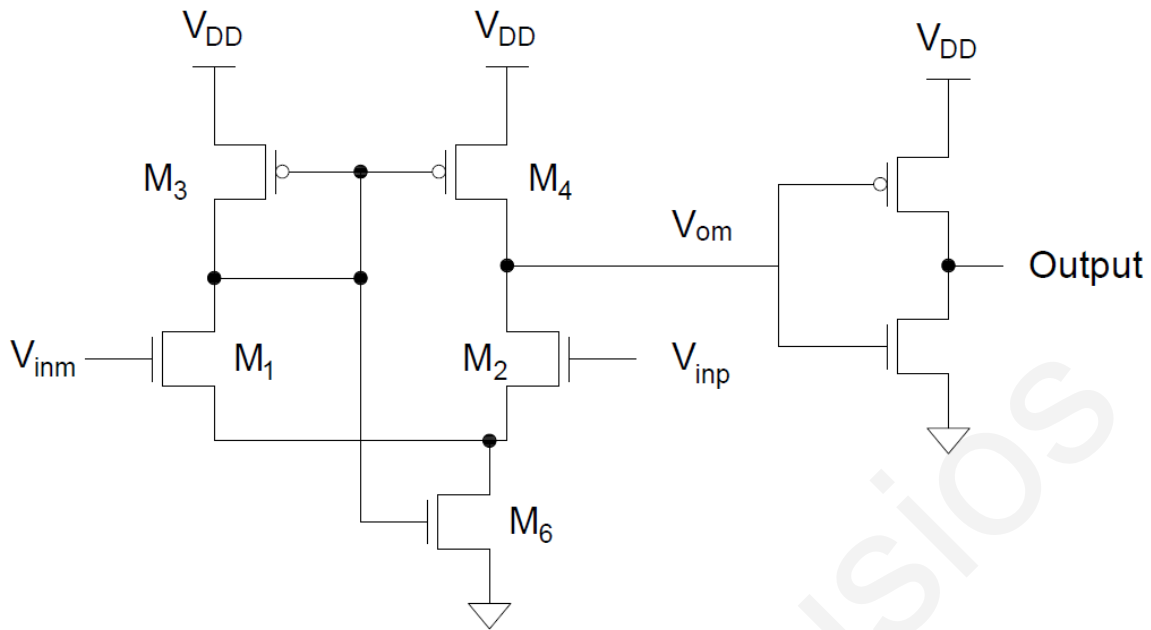


Figure 75: Self-biased differential amplifier used as output buffer [1]

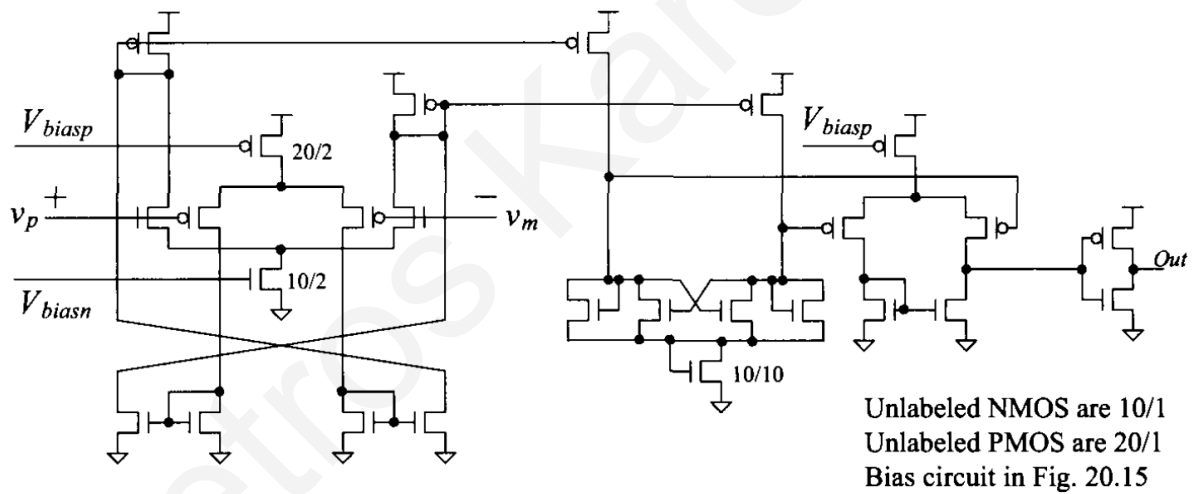


Figure 76: Comparator with rail-to-rail functionality [1]

The above circuit is a comparator used for general purposes. It uses as a preamplification circuit both PMOS and NMOS differential amplifiers in order to achieve rail-to-rail functionality. In this circuit, the output buffer is of PMOS type that drives the output inverter. The reason that a PMOS type of output buffer is used is because it has better operation when facing lower input level signals, with an improvement regarding delays [1].

IV. Design and implementation of an 8-bit Digital to Analogue Converter

The proposed DAC architecture was the dual-resistor string. This architecture was chosen for a variety of reasons. First, this architecture can achieve rail-to-rail output using simple components with high accuracy. With that being said, the second reason is the simplicity of its design. They require no active circuitry; they have simple design and they are monotonic [43][44]. Additionally, this design does not require the use of an op-amp at the output because the system will drive a resistive load of high value, such as the gate of the transistor, which theoretically has infinite resistance value. Other DAC types, such as current steering or R-2R architectures cannot drive such resistive loads because there is no current flow through the gate of a MOSFET. Thus, this application specific device will be controlling the transistor, which is used either as a varistor or as a varactor, as to alter the electromagnetic properties of the metamaterials.

Varactor and Varistor [45]

The varactor cross-section can be seen in the following figure and its purpose is to form a diode. The charge in this case is formed in the gate and in the semiconductor under the oxide. Source and Drain terminals are connected. The device is controlled by applying voltage at the Gate of the transistor, since the Source/Drain connection is connected to bulk. If the MOS varactor is a p-type MOS transistor, when the bulk-gate voltage V_{bg} is greater than the threshold voltage V_{TH} a channel will be formed [45].

The following figure shows the cross section of the MOS varactor. The topology of the device is as follows: A metal oxide is formed on a lightly doped n-well diffusion layer. The total capacitance of the device is formed by connecting oxide capacitance C_{ox} in series with the two other capacitances formed in this device, C_b and C_i , which are connected in parallel. The first capacitance occurs due to lack of carriers in the depletion region whilst the latter capacitance is formed by the inversion layer. The total capacitance of the device it is shown in fig.78, where it is drawn versus bulk-gate voltage [45].

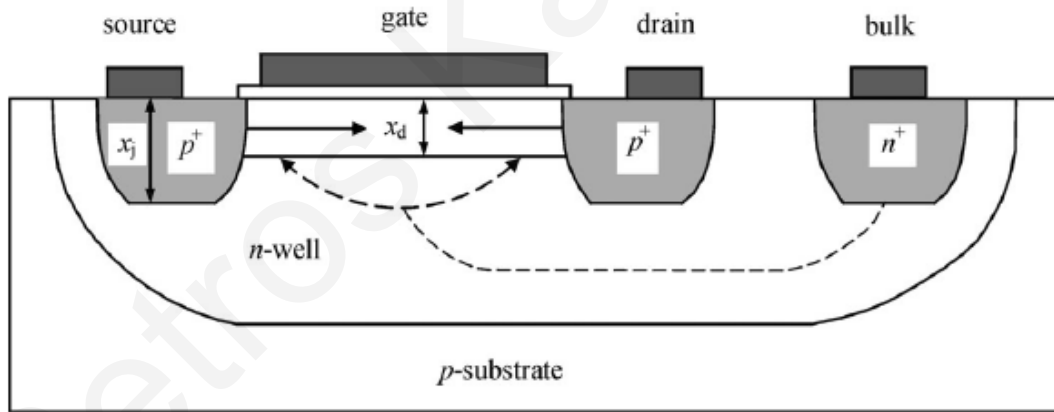


Figure 77: The MOS transistor cross-section. Solid lines represent the movement of charge carriers when the transistor operates in strong and moderate inversion regions. The dashed lines show the movement of electrons represent the parasitic resistance. [45]

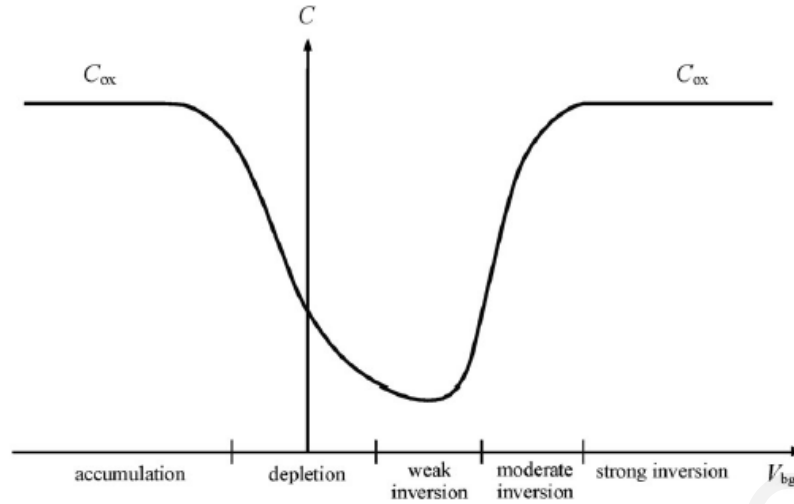


Figure 78: Capacitance relation to Bulk-Gate voltage [45]

In the moderate inversion region where V_{bg} is near the threshold voltage, the capacitance of the inversion layer is far greater than of the bulk capacitance, even though the number of holes decreases in the oxide interface. In the weak inversion region where there is modulation of the depletion region and hole injection, the capacitances involved, C_i and C_b are equal. In the depletion region, where there is more charge collected there than in the previous region mentioned above, $C_i < C_b$. Finally, in the strong inversion and accumulation regions, the total capacitance C_v is equal to the oxide capacitance. This happens because at high V_{bg} , there is the larger number of carriers. However, due to the large number of carriers, parasitic resistances are introduced, resulting into losses of electrons. They tend to move from the bulk metal contact to the depletion region. This parasitic resistance can be reduced by using smaller technology that have smaller gate length L . Additionally, another advantage of using smaller technology, is that the thickness of the oxide is reduced thus increasing the oxide capacitance. This increase improves the quality factor Q of the device and introduces wider tuning range [45].

Engineers, operate the MOS device (in varactor mode) with the bulk terminal connected to the supply voltage, being the highest DC voltage in the circuit. This prevents the device entering the accumulation region, thus the relation capacitance to voltage is now monotonic. However, the use of p-type device as varactor in accumulation and depletion regions, reduces the parasitic resistances due to slower mobility of holes versus electrons, thus the quality factor increases alongside with the tuning range [45].

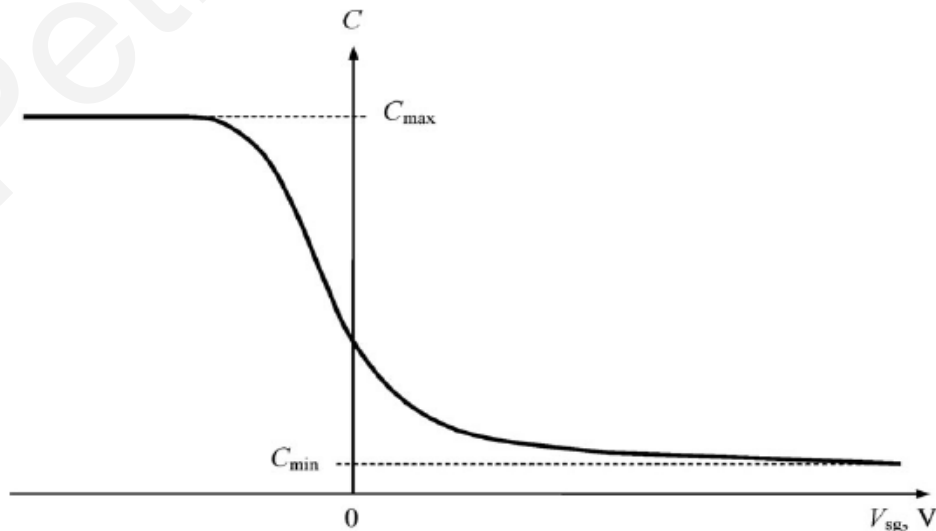


Figure 79: The curve of capacitance vs voltage of a PMOS varactor, operating at accumulation and depletion region [45]

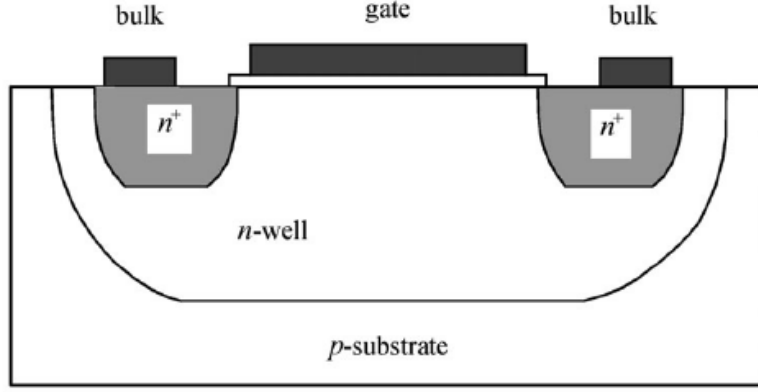


Figure 80: The PMOS varactor diode [45]

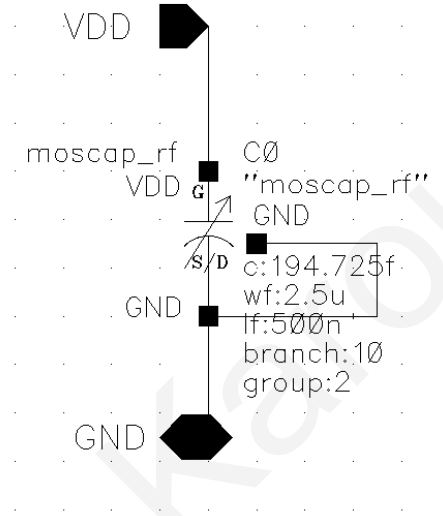


Figure 81: RF Varactor symbol from Cadence IC Design Software

As can be seen in the above figure 79, in order for a PMOS device varactor to have a monotonic relation between capacitance and voltage, the device must operate in the accumulation and depletion regions. This can be achieved by terminating the formation of the inversion layer in any of its forms. In order for this to happen, the injection of holes into the MOS channel must be stopped and can be accomplished by replacing S/D p^+ doped layers with Bulk n^+ doped layers, actually forming a varactor diode [45].

Regarding the varistor as described in [1][39], basically it is a transistor used in the triode regions, which serves as a variable resistor. If we consider the Taiwan Semiconductor Manufacturing Company (TSMC) 0.18 μm to be a long channel technology, [39] Razavi's equations for long channel MOS models can be used. When the device operates in the deep triode region, meaning that $V_{DS} \ll 2(V_{GS} - V_{TH})$, there is a linear relation between the transistor's current and Drain/Source Voltage which is described in the following equation:

$$I_D \approx \mu_n C_{ox} \frac{W}{L} (V_{GS} - V_{TH}) V_{DS}$$

I_D is the drain current, μ_n is the electron mobility, C_{ox} is the capacitance of the oxide of the gate, W and L are the width and length of the device respectively. V_{GS} is the Gate/Source voltage, V_{DS} is the Drain/Source voltage and V_{TH} is the threshold voltage of the device.

The above equation can be reformed to represent the channel resistance

$$R_{on} = \frac{1}{\mu_n C_{ox} \frac{W}{L} (V_{GS} - V_{TH})}$$

At the glimpse of an eye at the above equation, the only variable is the Gate/Source voltage, because the other parameters cannot be altered, and they are depended on the device's technology. Therefore, by adjusting the Gate/Source voltage while the transistor operates in the triode region, the device functions as a varistor [39].

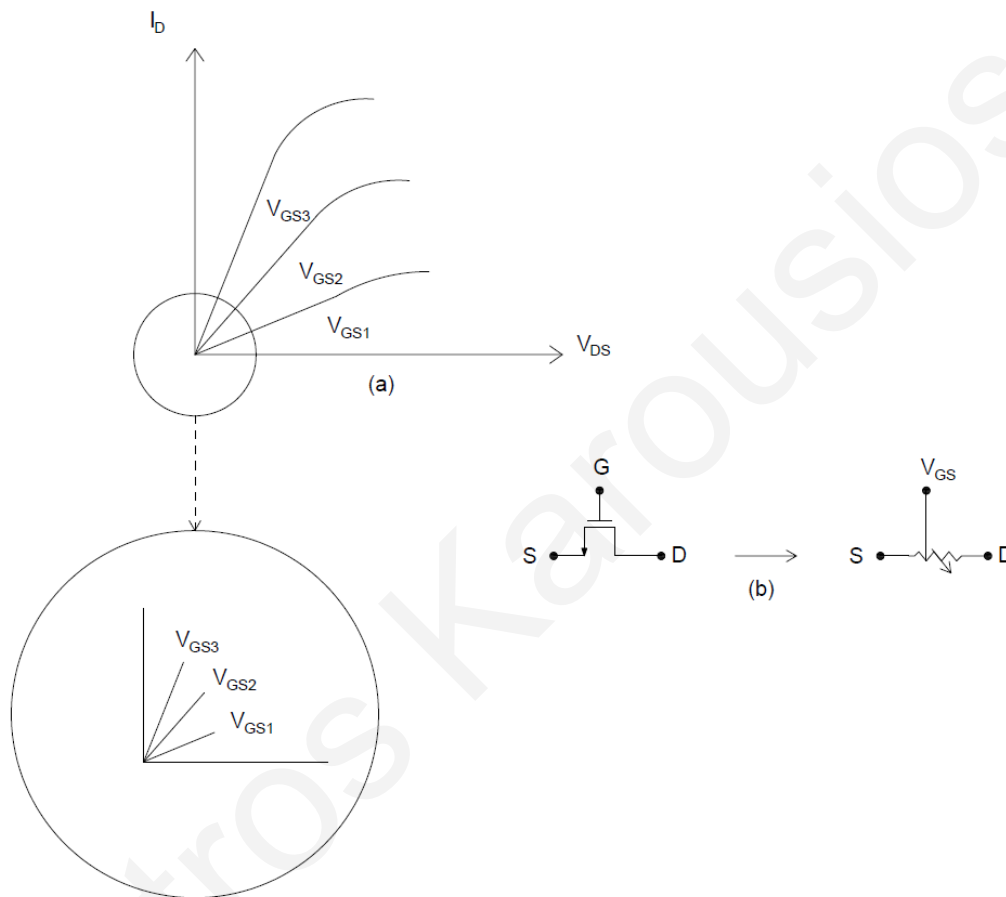


Figure 82: (a) Linear relation of drain current and Source/Drain voltage in deep triode region. (b) MOS transistor as a varistor [39]

Specifications

A rail-to-rail Dual-resistor string Digital to Analogue converter uses less resistors than a normal resistor string, specifically, instead of using 2^N resistors, now uses $2 \times 2^{N/2}$. This is a significant reduction in devices which translates to a reduction in area used. Another advantage of using less devices, is the downsize of the parasitic capacitance and the decoder's complexity. However, despite the fact that the reduction of resistive devices reduces the parasitic capacitance, the large number of switches actually produce parasitic capacitance and consume area [43][44].

This architecture uses two ladders to make the conversion. The first ladder, which is the MSB ladder or coarse resistor string, converts N_c bits. Additionally, the second ladder, which is the LSB ladder or fine resistor string, converts the remaining N_f bits. Thus, the complexity of the system is reduced to $2^{N_c} + 2^{N_f}$. An advantage of this architecture is that, each ladder is read separately which aids in reducing the complexity of the digital circuitry. A drawback of using two ladders is that, the

second ladder draws static current from the first ladder, causing voltage drop thus increasing the DNL error. Different techniques have been introduced to eliminate this voltage drop. One technique is the use of two large common mode buffers at the fine ladder end points. This requires additional use of area and the offset must be trimmed to match the DAC's characteristics. Another solution is passive compensation which is used in this thesis. It uses dummy switches between the resistors of the LSB ladder as a countermeasure for the voltage drop that occurs due to static current flowing through the secondary ladder. These dummy switches are identical as the switches used in the MSB ladder and every step in the LSB ladder has a resistor and a switch [43][44].

Each resistor string is comprised of 16 resistors, identical for each string. Regarding the Coarse Resistor String, which, as mentioned before has 16 resistors, the voltage $\Delta V = V_{REFH} - V_{REFL}$ is divided into sixteen segments resulting in $\frac{\Delta V}{16}$. V_{REFH} is the high voltage reference and V_{REFL} is the low voltage reference. In the coarse resistor string, there are 16 nodes numbered n_0 to n_{16} . Each node is connected to a switch numbered from CS_0 to CS_{16} . These 16 switches serve as the 1st multiplexer which is responsible for bits D_4 to D_7 . The switches are controlled by the 4-bit decoder which decodes the 4 MSBs. All nodes are connected to two side to side switches, except n_0 and n_{16} which are connected to only one switch. When a pair of switches closes, the corresponding voltage of the resistor connected to the switches is transferred to the end points of the fine resistor string, the high voltage at V_H and the low voltage at V_L . As a result, the sixteen resistors located in the fine resistor string are connected in parallel with the resistor mentioned above. This results in ideally dividing the voltage at the fine resistor string by 16 additional resolution increments resulting into $\frac{\Delta V}{256}$. Then, a second multiplexer, MUX2, which is controlled by a 4-bit decoder, is responsible for the bits D_0 to D_4 . Finally the output is transferred to the load [44].

For example, if the digital input is 01000100, then the DAC will separate the input into 2 segments. For the MSBs, the string will decode the binary number 4 and close switch pairs CS_4 . As a result, the fine resistor string will be connected in parallel with CR_4 . As mentioned above, the voltage transferred is $\frac{\Delta V}{16}$, which will be additionally divided into sixteen segments. Furthermore, MUX1 not only transfers $\frac{\Delta V}{16}$ to the fine resistor string but also passes an offset to the voltage, through n_4 to V_L . Node V_H is connected to n_5 , which is the higher voltage whilst V_L to n_4 , which is the lowest. The fine resistor will decode the 4 LSBs like it was the binary number 4, closing FS_4 . The total voltage then becomes $V_{DAC} = \Delta V \times \frac{4}{16} + \Delta V \times \frac{4}{256} = \Delta V \times \frac{68}{256}$. In order to obtain LSB zero, the switch 0,1 is closed and SW_x is open. This architecture even though uses less resistors and switches, Integral and Differential Nonlinearity are introduced.

Integral Nonlinearity: It is introduced in the system by the way this architecture is designed. As mentioned above, the coarse resistor string provides to the fine resistor string not only the voltage drop across one of the coarse resistors but also the voltage offset. The latter is used to sum up all the voltage drops across the coarse resistors that are located below the coarse resistor that is selected, at the node V_L . All coarse resistors are identical meaning that the voltage drop across each one of them is the same and constant. Due to the fact that the fine resistor string is connected in parallel with the selected coarse resistor, its resistance now differs of that of the rest of the coarse resistors. Thus, the current flowing through the coarse resistor string changes. According to Ohm's Law, the voltage at the ends of a resistor is proportional to the current flowing through the resistor and the resistance value, $V = I \times R$. Therefore, the voltage across each of the coarse resistor changes due to changes in the current. Additionally, the selected coarse resistor does not have the same voltage drop as the rest of the coarse resistors because its resistance value is not equal as the rest of the resistors. The solution to this problem is to make the resistors at the fine resistor string have greater value than the resistors located at the coarse resistor string. However, this introduces latency and slows the conversion speed [44].

Differential Nonlinearity: This error is introduced by the current that flows through the selected analog switches that connect the two resistor strings and their “ON” resistance. The solution to minimize this error is to create analog switch pairs that have resistance less than of that of the fine resistors. This requires switches with bigger transistors thus increasing their capacitances, eventually introducing latency to the system. Another way to reduce the DNL error from the selected coarse switches stated in [43], it to place dummy switches between the fine resistors. By making these switches identical with the coarse switches, then every step at the fine resistor string includes a resistor and a dummy switch, thus compensating for the voltage drop occurred by the coarse switches.

By making the fine resistances as large as possible, when connected in parallel with the coarse resistors, their resistance will be almost insignificant. The condition that needs to be met in order to achieve this, is to keep the voltage drop of the coarse resistor below a certain fraction α of an LSB [43].

$$\frac{V_{ref}}{2^{N_c} \cdot R_c} (R_c - R_c) \left| (2^{N_f} \cdot R_f) \right| < \alpha \cdot \frac{V_{ref}}{2^N}$$

Which can be simplified to:

$$R_f > \frac{R_c}{\alpha}$$

In our case, the coarse resistors are $R_c = 250.6 \Omega$ and the fine resistors are $R_f = 8017 \Omega$. Why these values? The value of the coarse resistors was chosen that low, in order to save up in area. The value of the fine resistors was chosen that high in order to eliminate INL errors and helps with the resistor matching. As a result, the fraction α is 0.0313 which satisfies the above equation.

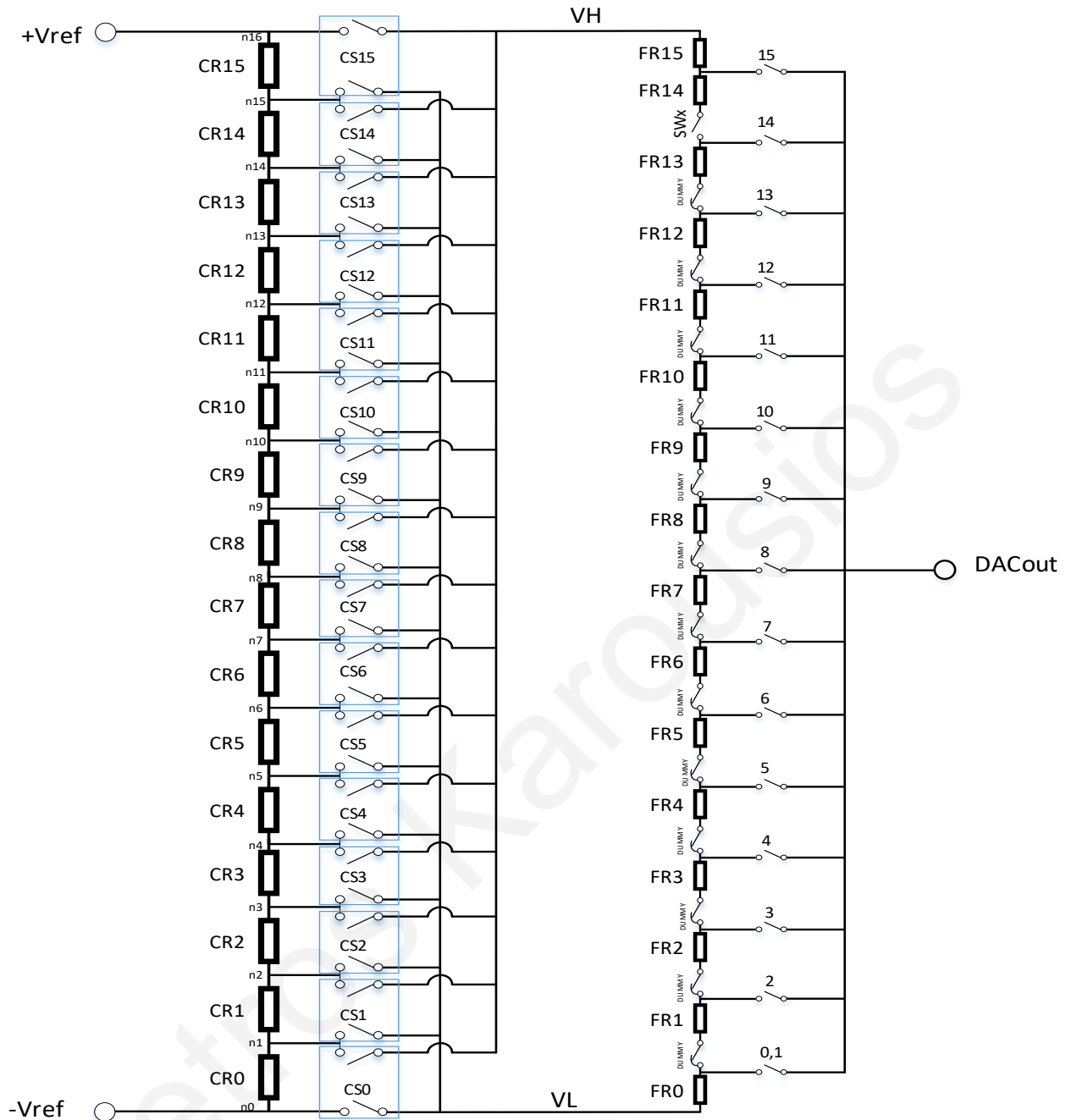


Figure 83: Dual-Resistor String Digital to Analogue converter architecture. [43.44]

The converter was designed using polysilicon resistors for various reasons. Firstly, this type of resistor is not located in the bulk area of the chip like other type of resistors, which, due to formation of pn-junction they have an increase in their voltage coefficient thus diminishing their precision. Additionally, since poly resistors have not only better matching features, but also better temperature performance is preferred in analogue systems such as data converters. The resistors were implemented by using fingers, which are smaller resistors of equal value that when connected together make a full resistor, aiming to reduce the area of the converter [1]. Furthermore, dummy resistors were used to minimize the etch effects during fabrication. These resistors did not need to have full width as the normal resistors but the same length. Additionally, they were connected to ground to reduce any static electrical discharge [46].

There are various ways that the final product of a design might be altered. Regarding resistors, some are the random mismatches due to dimension alterations, doping, irregularities in oxide thickness, nonuniform current flow and other, were statistically modelled in order to be predicted and eventually avoided by simply following some rules to tackle mismatched [46].

In [46], the author proposes 23 rules for resistor matching. Some of them, that were used in the implementation of the converter are:

1. The resistors that to be matched must be implemented by using the same material.
2. The resistors that to be matched must not have different width.
3. The resistors that to be matched must have same geometries.
4. The resistors that to be matched must be placed in the same direction.
5. The resistors that to be matched must be placed as close as possible.
6. Dummy resistors must be placed on both sides of the resistor array.
7. Short segments have higher mismatches and must be avoided due to contact resistance.
8. Resistors must be placed as close to the center of the die to avoid mechanical stress.
9. Must be placed far from power devices due to overheating.
10. The use of poly resistors instead of diffusion resistors is preferred because they are constructed to be narrower and therefore long thus smaller width results in small increase in mismatch.
11. Use guard ring around the resistor array with grounding.

Resistors, as all devices suffer from noise. One type of noise that resistors experience is the thermal noise, which is produced by electron movement which in conductor is performed randomly. As a result, the voltage suffers from variations. The frequency spectrum that thermal noise affects the resistor is analogous to the absolute temperature.

Thermal noise of a resistor can be modelled by placing in series a noiseless resistor R and a voltage source. The equation that calculates the spectral density can be found by:

$$S_v(f) = 4 \times k \times T \times R, \quad f \geq 0$$

Whereas the Boltzmann's constant $k = 1.38 \times 10^{-23} \text{ J/K}$, R is the resistor's value and T is the temperature.

Due to the fact that the spectral density has units V^2/Hz , the "noise voltage" $\overline{V_n^2} = 4 \times k \times T \times R$. For example, in our case $R_c = 250.6 \Omega$ and $R_f = 8017 \Omega$ which, at $T = 300^\circ\text{K}$, the thermal noise is $41.5 \times 10^{-19} \text{ V}^2/\text{Hz}$ and $1327 \times 10^{-19} \text{ V}^2/\text{Hz}$ respectively. The result is then converted to a more "friendly" form of noise figure by taking the square root resulting in $2.037 \text{ nV}/\sqrt{\text{Hz}}$ and $11.52 \text{ nV}/\sqrt{\text{Hz}}$. By examining spectral density one can observe that the noise is white and flat for frequencies up to 100 THz.

Transmission Gate

The switches are implemented as shown in [1] using Transmission Gates, TG, which is comprised by an NMOS transistor, which passes logic "0" well, and a PMOS connected in parallel, which passes logic "1" well, therefore the resistance from the input to the output can be calculated as $R_n || R_p$. The control signal selects the state of the TG. When $S = 1$, the TG is "ON" and the input signal is transferred to the output. One can observe that the input and output of the TG are interchangeable.

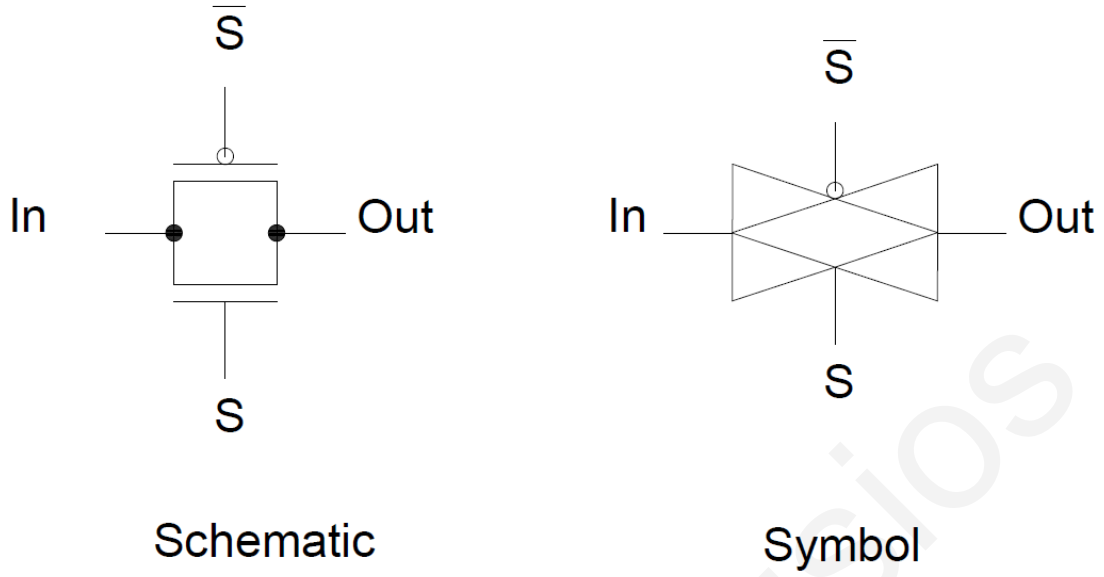


Figure 84: Transmission Gate schematic and symbol [1]

The resistance of the TG, as stated above, is comprised by two resistances in parallel. R_n is the switching resistance of the NMOS transistor and R_p is the switching resistance of the PMOS transistor. The resistance can be found by the following equation, when $VDD = VDS$ and $VGS = VDD$

$$R_n = \frac{VDD}{\frac{KP_n}{2} \cdot \frac{W}{L} \cdot (VDD - V_{TH})^2} = R'_n \cdot \frac{L}{W}$$

Where VDD is the highest voltage in the circuit, KP_n is the transconductance parameter in A/V^2 (sometimes can be seen as $\mu_n C_{ox}$), W and L are the width and length of the transistor, and finally V_{TH} is the threshold voltage of the transistor. However, it is not recommended to use the above equation because it does not take into account the reduction in mobility, it is best to use measured data, resulting in:

$$R_n = R'_n \cdot \frac{L}{W} = \frac{VDD}{I_{D,sat}} \cdot \frac{L}{W}$$

One has to take into account that the electron's mobility is about 2.5 times larger than the mobility of holes, resulting into the switching resistance of the NMOS device being 3 times smaller than the one of the PMOS device [1]. The calculated results for $L_{n,p} = 0.18\mu m$, $W_n = 1\mu m$ and $W_p = 3\mu m$ are:

$$R'_n = \frac{1.8}{571.77 \mu A} = 3.148 k\Omega$$

$$R_n = R'_n \cdot \frac{L}{W} = 566.64 \Omega$$

$$R'_p = \frac{1.8}{747.487 \mu A} = 2.408 k\Omega$$

$$R_p = R'_p \cdot \frac{L}{W} = 188.88 \Omega$$

The total resistance of the TG can be calculated by $R_n || R_p = 141.66 \Omega$

The delay of the TG can be calculated by

$$t_{delay} = 0.7 \cdot (R_n || R_p) \cdot C_L$$

Where C_L is the load capacitance. C_{ox} capacitance is not taken into consideration due to the fact that it is quite smaller than the load capacitance, therefore is neglected. One can observe that if the total resistance of the TG is reduced, the propagation delay is also decreased. However, this requires more area [1].

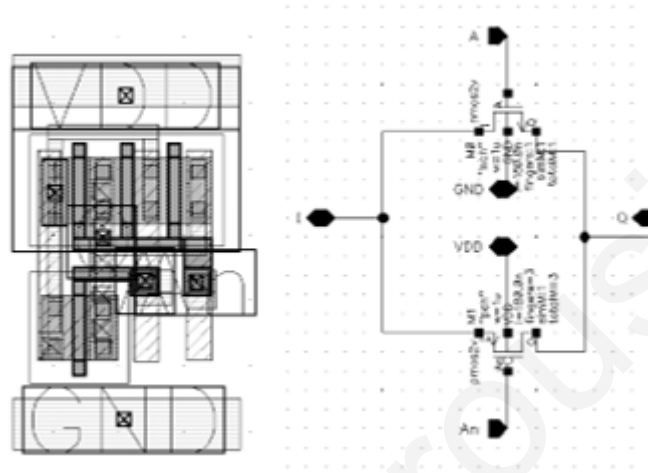


Figure 85: CMOS Transmission Gate Layout and Schematic

Device	Width (μm)	Length (μm)
PMOS	3	0.18
NMOS	1	0.18

Table 4: Transmission Gate device sizes

Characteristics	Result
R_n (calculated)	3148 Ω
R_p (calculated)	2408 Ω

Table 5: Transmission Gate resistance characteristics

CMOS Inverter

The CMOS inverter is a basic block in every digital circuit. Its operation is to invert the input signal A to output signal A not. The task is performed by two transistors connected in series, a PMOS and an NMOS. For example, if the input signal is ground, the output is driven to VDD by the PMOS, and similarly, when the input signal is connected to VDD, the output is driven to ground by the NMOS device. In both cases, when the transistor responsible for the output signal operates, the other transistor is OFF. Major advantages, among other, include the fact that the CMOS inverter has practically zero power dissipation and by changing the size of the device the switching threshold can change [1].

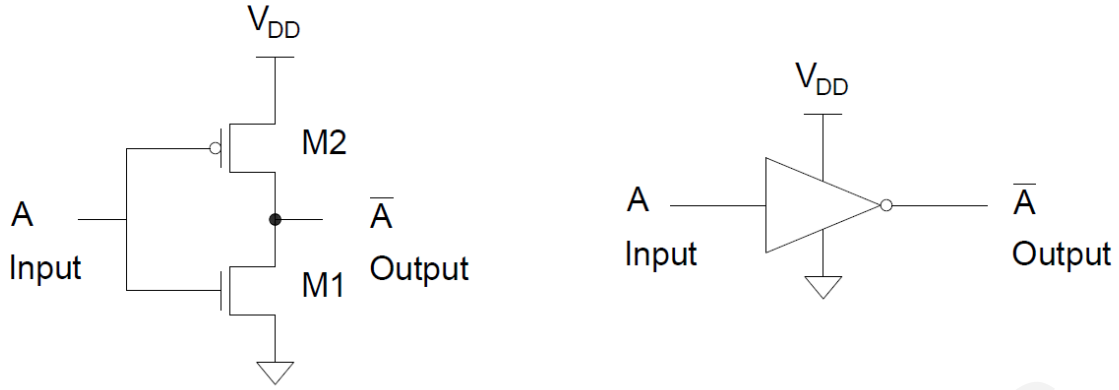


Figure 86: CMOS Inverter schematic and symbol [1]

The CMOS Inverter operates in three regions. At the first one, we assume the input voltage is low, less than the threshold voltage of the NMOS transistor (M1) in order for the NMOS device to be OFF. Additionally, for the PMOS transistor (M2) the voltage is $V_{SG} \gg V_{THP}$ which is required for the PMOS transistor to be ON. In the second region, we assume that the input voltage increases resulting in M1 switching ON, therefore both transistors operate in this region. Additional increase in the voltage will eventually turn off the M2 and the CMOS Inverter operates in the third region. In the figure below, V_{OH} is the maximum output voltage of the inverter, V_{OL} is the lowest output voltage. Ideally, V_{IL} and V_{IH} are zero. In this region there is no defined logic level [1].

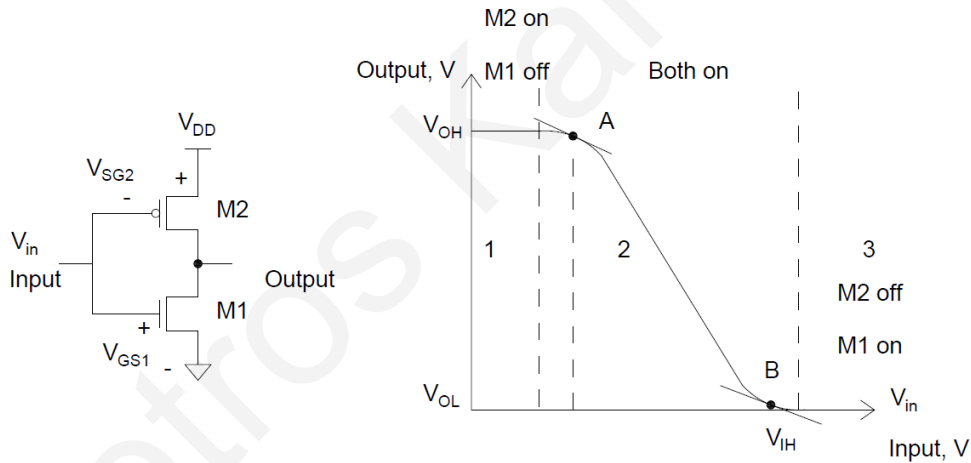


Figure 87: CMOS Inverter DC transfer characteristics [1]

The inverter's switching point, V_{SP} , as can be seen in the following figure, is the point where the input voltage is the same as the output voltage. The ideal switching point is $V_{DD}/2$. Both devices are in saturation region, and since their voltage is the same, the following equation from [1] arises:

$$\frac{\beta_n}{2} (V_{SP} - V_{THN})^2 = \frac{\beta_p}{2} (V_{DD} - V_{SP} - V_{THP})^2$$

Which concludes to:

$$V_{SP} = \frac{\sqrt{\frac{\beta_n}{\beta_p}} \cdot V_{THN} + (V_{DD} - V_{THP})}{1 + \sqrt{\frac{\beta_n}{\beta_p}}}$$

Where β_n is the $\mu_n C_{OX} \frac{W}{L}$ of the NMOS device and β_p is $\mu_p C_{OX} \frac{W}{L}$ of the PMOS device. Due to the fact that the mobility of electrons is three times faster than the mobility of holes, the PMOS device must be 3 times larger in order to have balanced inverter [1].

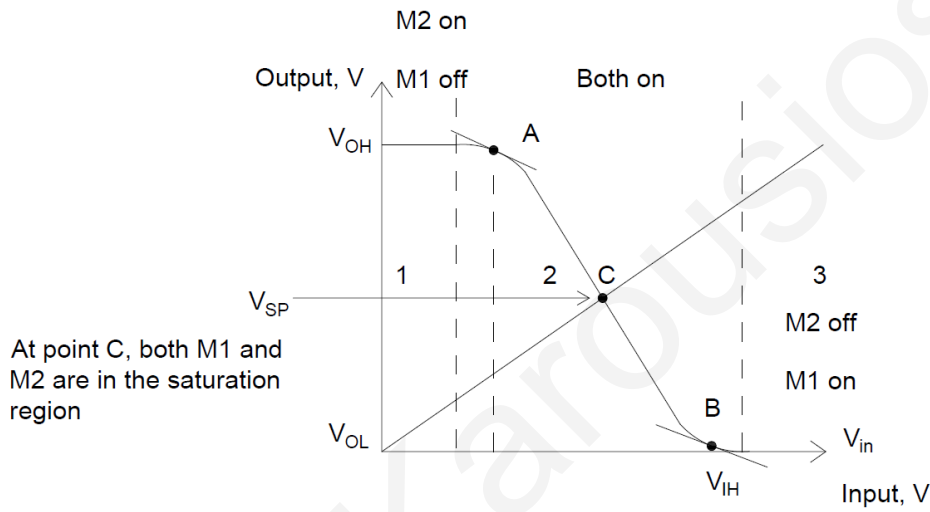


Figure 88: CMOS Inverter transfer characteristic showing the switching point [1]

The parasitic capacitances can be seen in the following figure and the delays can be calculated by:

$$t_{PLH} = 0.7 \cdot R_{p2} \cdot (C_{out} + C_{load})$$

The above equation shows the propagation delay of the Inverter from low to high, where R_{p2} is the switching resistance of the PMOS device, C_{load} is the capacitance of the load and

$$C_{out} = C_{outn} + C_{outp} = C_{ox2} + C_{ox1}$$

The propagation delay from high to low can be calculated by:

$$t_{PHL} = 0.7 \cdot R_{n1} \cdot (C_{out} + C_{load})$$

Where R_{n1} is the switching resistance of the NMOS device [1].

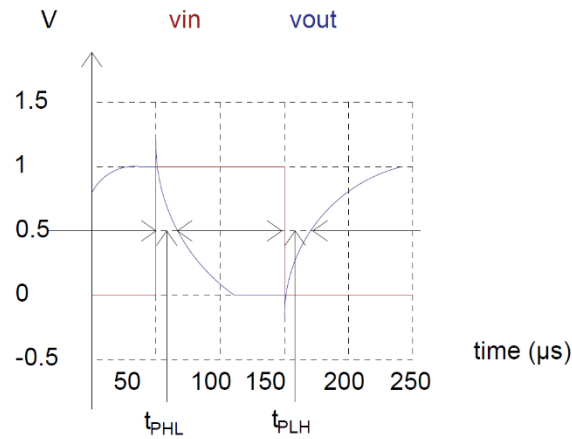


Figure 89: Propagation delays of the CMOS Inverter [1]

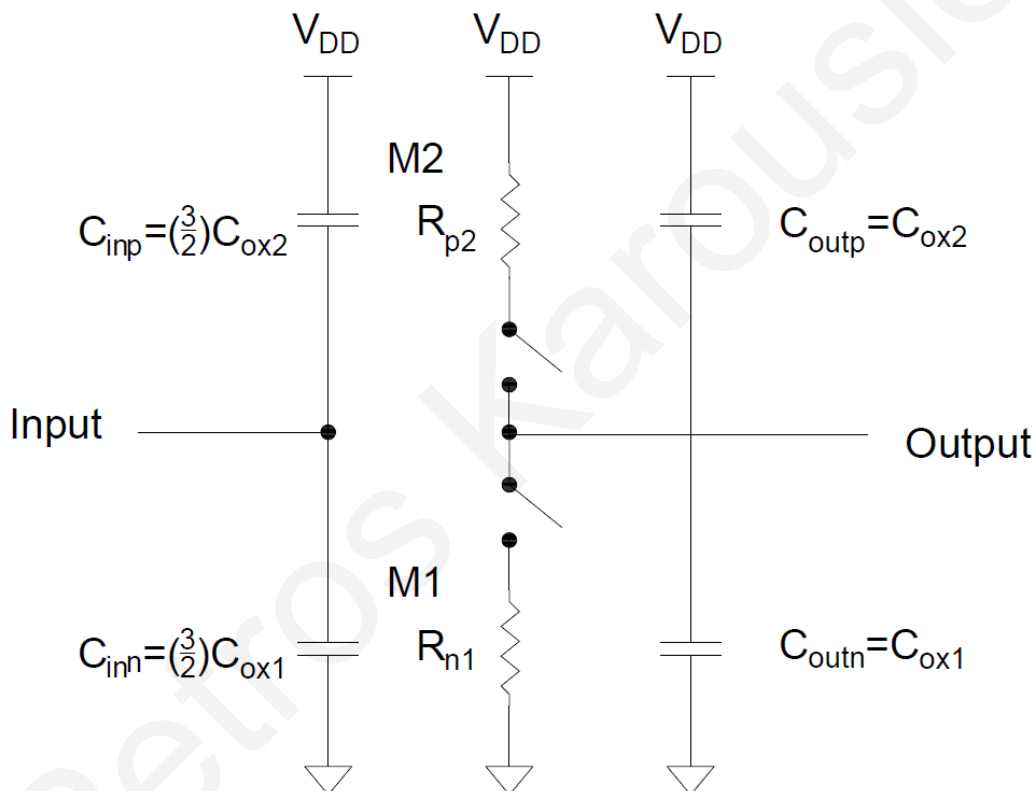


Figure 90: CMOS Inverter digital model showing switching characteristics [1]

Layout of the Inverter

The layout of the inverter must be created with precautions due to latch-up according to [1]. This condition, when activated, the output of the inverter will not change regardless the input signal and the logic output of the inverter is stuck at a specific logic state and the only solution is to remove the power to restart the operation of the inverter. In the following figure one can observe the two styles for designing the inverters. In both situations, Input and Output are designed using metal2 whilst VDD and Ground signals are designed using metal1.

The figure 92 illustrates the inverter's cross section, with both PMOS and NMOS devices, showing the parasitics. As can be seen by the figure above that shows the propagation delay of the inverter, when there is a transition in the input signal, there is a glitch at the output, meaning that the

input signal feeds through the output of the inverter, thus triggering the output to follow the same direction as the input [1].

Additionally, as can be seen by the figure 92, the cross section of the CMOS Inverter shows the parasitic bipolar transistor that are created in the layout of the device. The two bipolar transistors are created by the regions P+/N well/P substrate and N+/P substrate/N well, therefore there are a PNP, denoted as Q1 in the following figure, and an NPN transistor, denoted as Q2. The resistance of the N well is denoted as RW1 and RW2. The resistance of the substrate is represented as RS1 and RS2. Capacitor C1 is the capacitance between the drain of each transistor and the N well whilst C2 is the capacitance between the drain of each transistor and the substrate [1].

The two cases where latch-up condition may occur, is when the input signal changes state, from ground to VDD or from VDD to ground. These cases are causing a positive feedback to the parasitic circuit. When the switching from low to high happens rapidly, the glitch from the pulse is fed through C2 resulting in forward biasing the base-emitter junction of the Q2. With this condition, bipolar transistor Q1 is turned on due to increased current that runs through RW1 and RW2. Eventually, Q1 transistor being on results in an increase in current through resistors RS1 and RS2, forcing Q2 to stay on. As mentioned above, this positive feedback forces Q1 and Q2 to operate in full and never to shut down, unless the system is turned off [1].

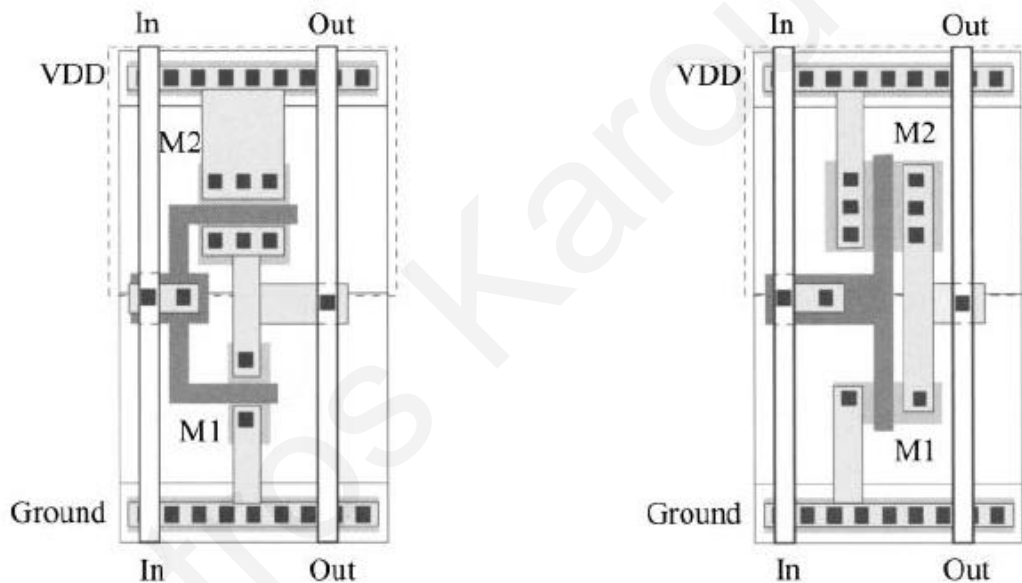


Figure 91: CMOS Inverter layout design methods [1]

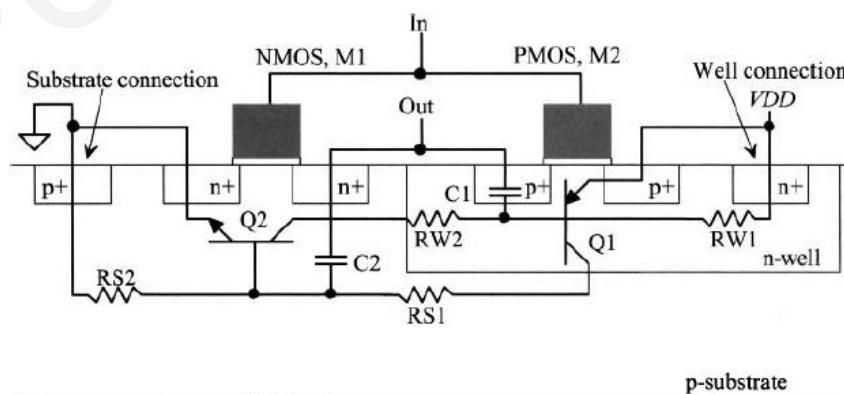
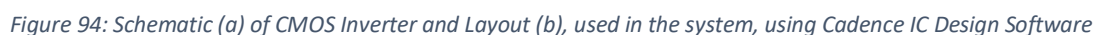


Figure 92: Cross-section showing a CMOS Inverter. The parasitic bipolar transistors and resistors can be seen, responsible for latch up condition. [1]



Also, guard rings placed around the devices, prevent signals to transfer from one device to the other. Their purpose is to reduce substrate noise that is produced by nearby circuits by providing a low impedance path to ground for substrate noise. Guard rings can be implemented by using either n+ or p+ areas around the circuits. By placing a p+ area with contacts around a device connected to ground, the substrate noise is reduced. Finally, poly is not used to connect the gates of the transistors, because long poly routing has a high resistance in comparison to metal. Poly over non active areas goes over field oxide, which is thicker and doesn't cause parasitic channels, thus, metal2 is used to connect the gates, as shown in the figure above [1].



Device	Width (μm)	Length (μm)
PMOS	1.2	0.18
NMOS	0.4	0.18

Table 6: CMOS Inverter device sizes

Characteristics	Result
V_{SP} (calculated)	1.14 V
V_{SP} (measured)	0.883 V
R_n (calculated)	7096 Ω
R_p (calculated)	5903 Ω

Table 7: CMOS Inverter characteristics

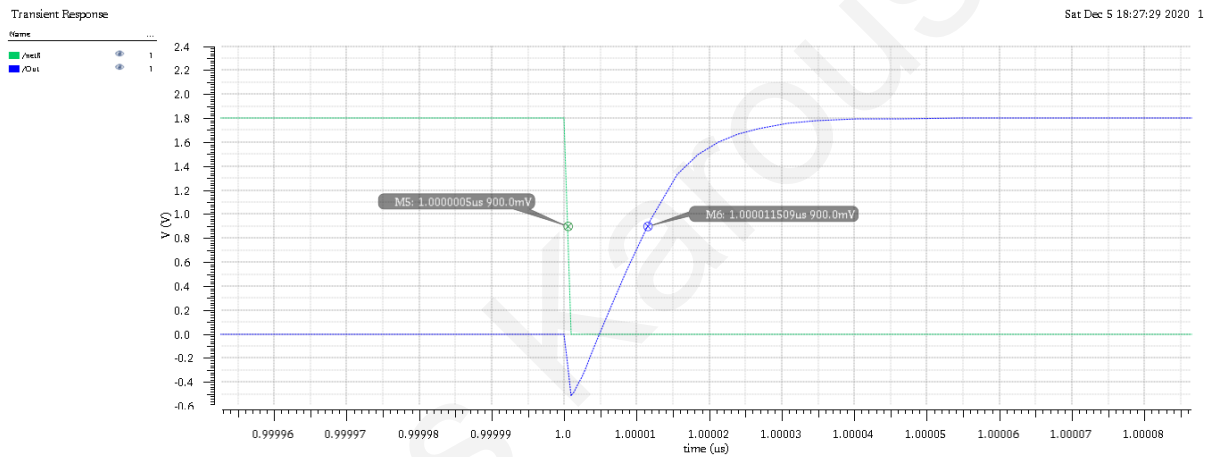


Figure 95: Simulated propagation delay of the CMOS Inverter from low to high

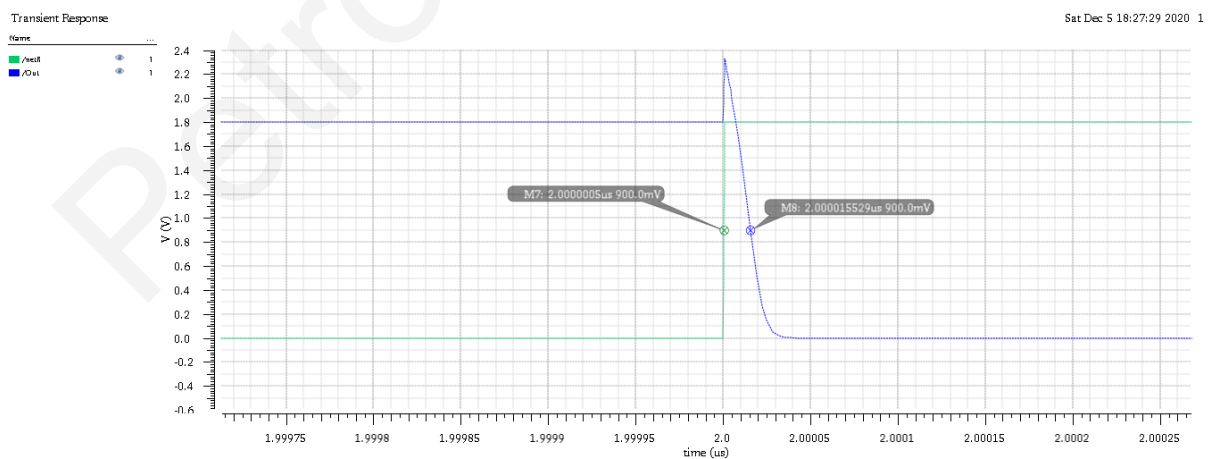


Figure 96: Simulated propagation delay of the CMOS Inverter from high to low

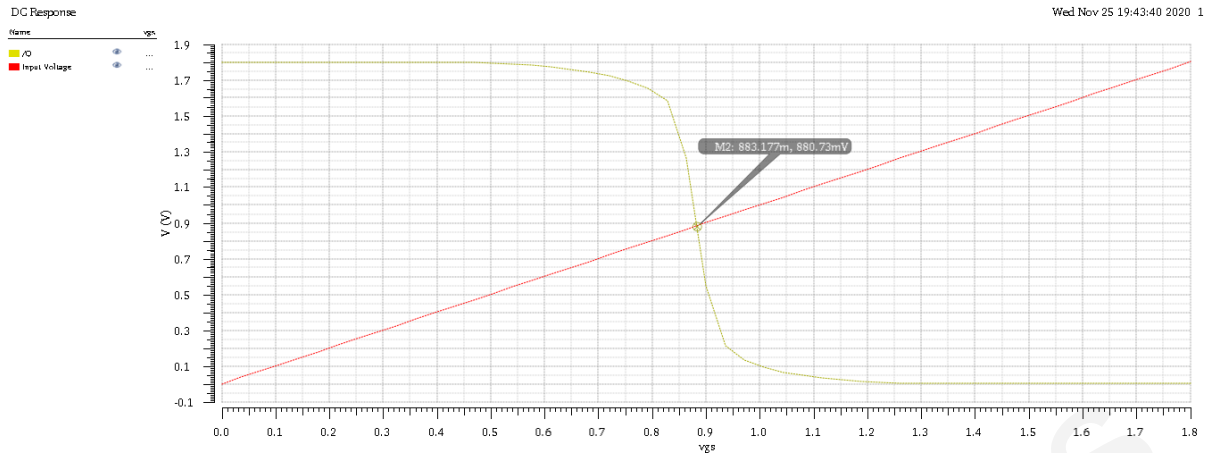


Figure 97: Simulated switching point of the CMOS Inverter

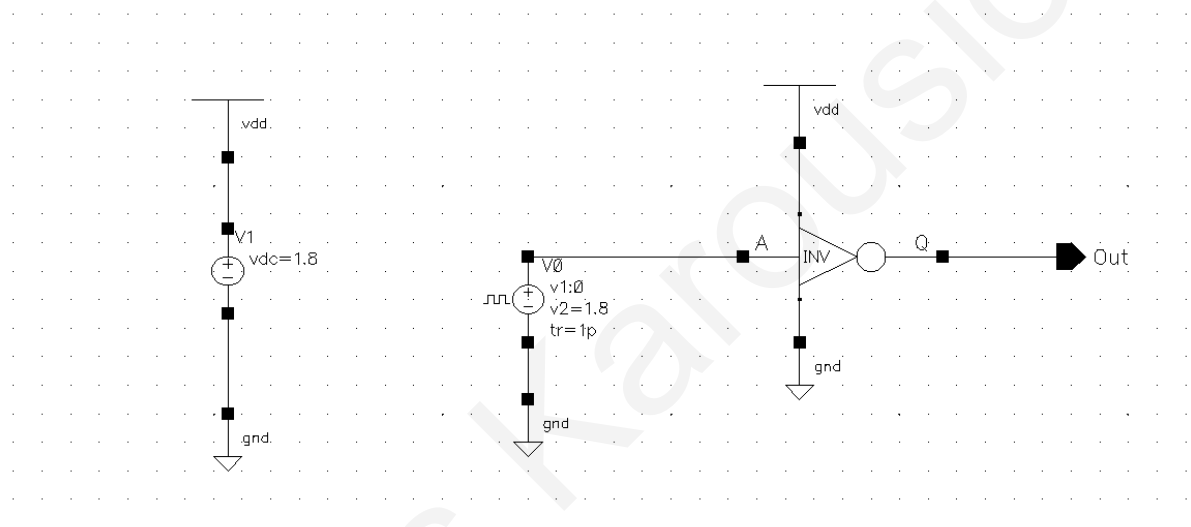


Figure 98: Test bench used in simulations

NAND Gate

Similarly to the CMOS Inverter, the input is connected to both NMOS and PMOS devices. The NAND Gate can be divided into two parts, as can be seen in the figure below. The first one is comprised by two PMOS devices connected in parallel and to VDD. The second part is implemented by using two NMOS devices connected in series and to ground [47]. The output is denoting as \overline{AB} .

The operation is as follows: If any of the inputs A or B has logic signal 0, at least one of M1 and M2 NMOS devices will not operate thus disconnecting the output from the ground node. However, one PMOS device will be active therefore connecting the VDD node to the output. The only case that both inputs have logic 1, both NMOS devices are active, connecting the output node to ground. On the other hand, both PMOS devices are switched off therefore disconnecting the VDD node from the output. A NAND gate can be implemented having k inputs. This can be done by simply placing k NMOS devices and k PMOS devices. Finally, if both inputs are tied together, the gate operates as an inverter [47].

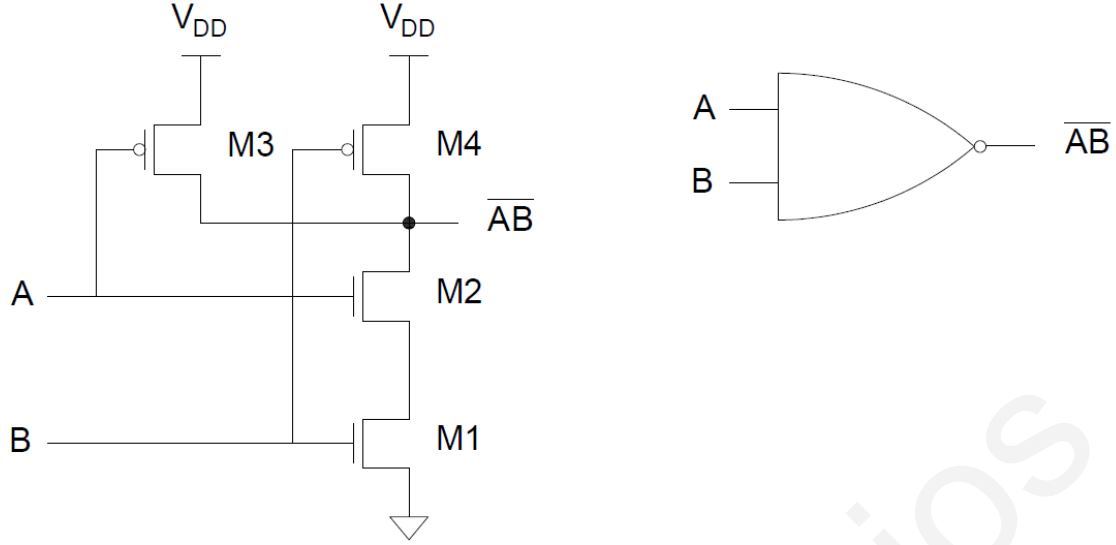


Figure 99: Schematic of the NAND Gate and its symbol [1]

A	B	Pull-Down Network	Pull-Up Network	\overline{AB}
0	0	OFF	ON	1
0	1	OFF	ON	1
1	0	OFF	ON	1
1	1	ON	OFF	0

Table 8: NAND Gate truth table [47]

As mentioned in [1], Considering that not only NMOS devices have the same width W_n and length L_n but also PMOS devices have the same width W_p and length L_p , we can determine the gate's switching voltage, V_{SP} . The two PMOS connected in parallel operate as a single transistor with width twice as big, resulting in $2W_p$. The NMOS connected in series operate as a single transistor with twice as big channel length, resulting in $2L_n$. As a result, the transconductance of the NAND Gate concludes to $\frac{\beta_n}{2 \cdot 2\beta_p}$

The voltage switching point, in order to be found, requires that the inputs are connected together, therefore, the NAND Gate functions as an Inverter. For a NAND Gate with k inputs the voltage switching point becomes:

$$V_{SP} = \frac{\sqrt{\frac{\beta_n}{k^2 \beta_p}} \cdot V_{THN} + (V_{DD} - V_{THP})}{1 + \sqrt{\frac{\beta_n}{k^2 \beta_p}}}$$

Layout of the NAND Gate

The layout of the NAND Gates, Inverters and TGs, must be of the same height. This can assist in stacking them one next to the other. Concerning the NAND Gate layout, in our case 3 input and 4 input devices, the NMOS MOSFETs that are connected in series, they share their drain and their source. The propagation delay of the parallel connected PMOS devices can be found by:

$$t_{PLH} = 0.7 \cdot \frac{R_p}{N} \cdot (N \cdot C_{oxp} + C_{load})$$

The propagation delay of the series connected NMOS devices can be found by:

$$t_{PHL} = 0.35 \cdot R_n \cdot C_{oxn} \cdot N^2 + 0.7 \cdot N \cdot R_n \cdot C_{load}$$

The propagation delay of the NAND Gate when the output signal rises from Low to High can be found by:

$$t_{PLH} = 0.7 \cdot \frac{R_p}{N} \cdot (N \cdot C_{oxp} + \frac{C_{oxn}}{N} + C_{load})$$

The propagation delay of the NAND Gate when the output signal falls from High to Low can be found by [1]:

$$t_{PHL} = 0.7 \cdot N \cdot R_n \left(N \cdot C_{oxp} + \frac{C_{oxn}}{N} + C_{load} \right) + 0.35 \cdot R_n C_{oxn} \cdot N^2$$

NAND with 3 inputs

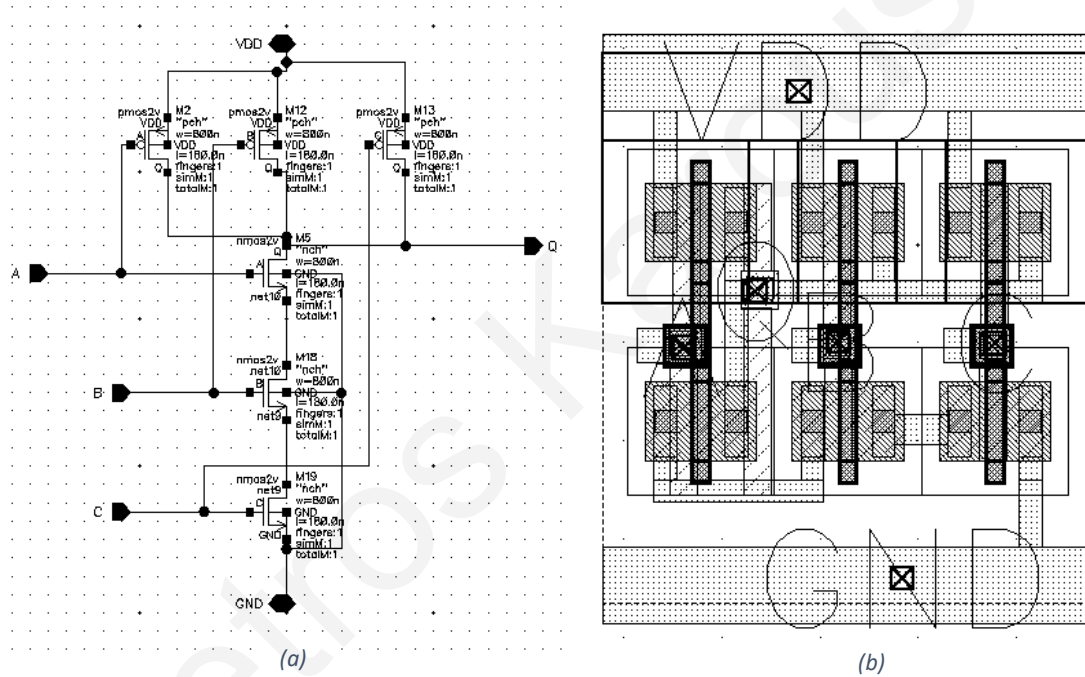


Figure 100: (a) Schematic and (b) layout of the NAND gate

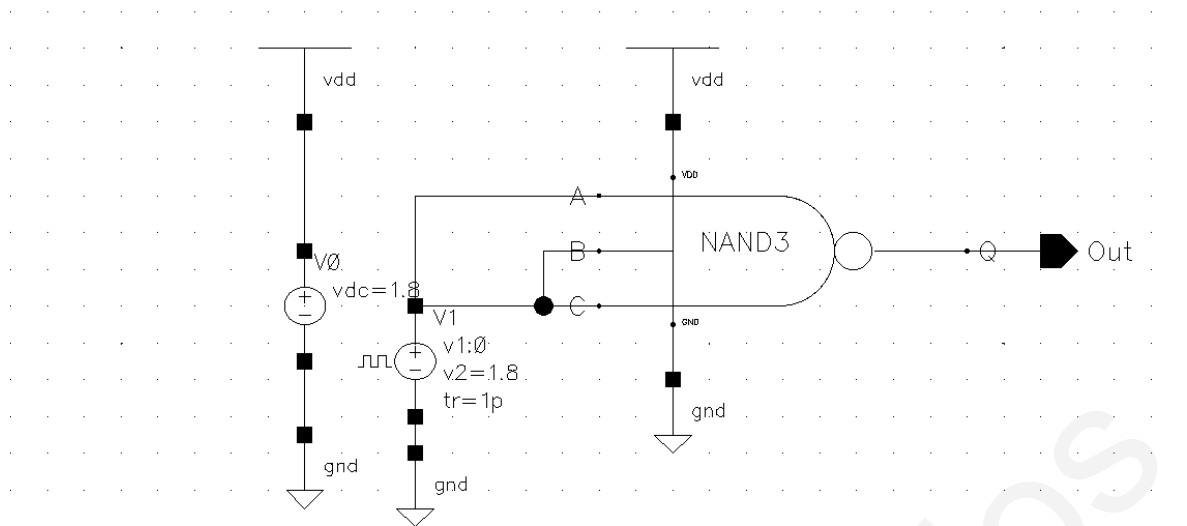


Figure 101: Test bench for the NAND gate with 3 inputs to find the switching point.

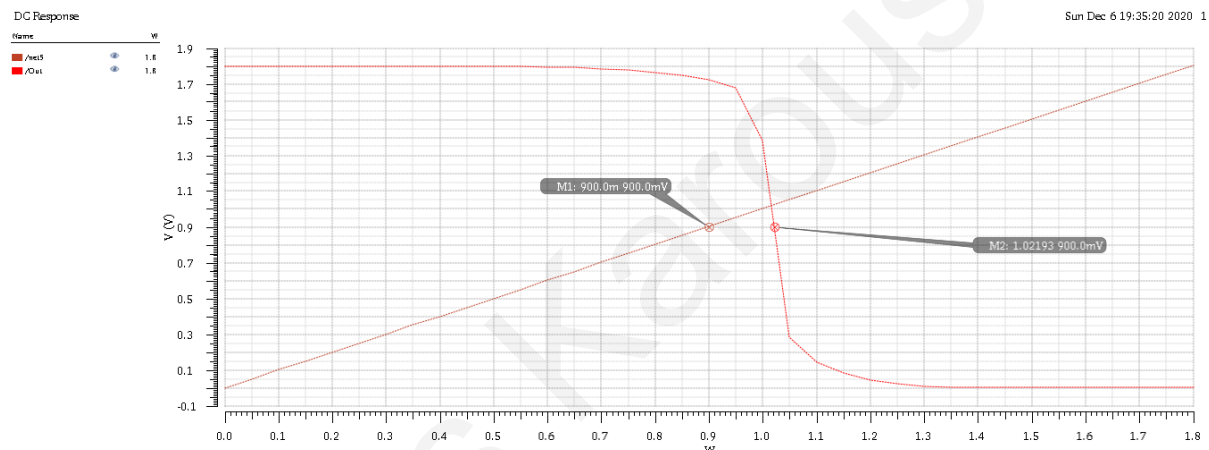


Figure 102: Simulated switching point of the 3-input NAND Gate

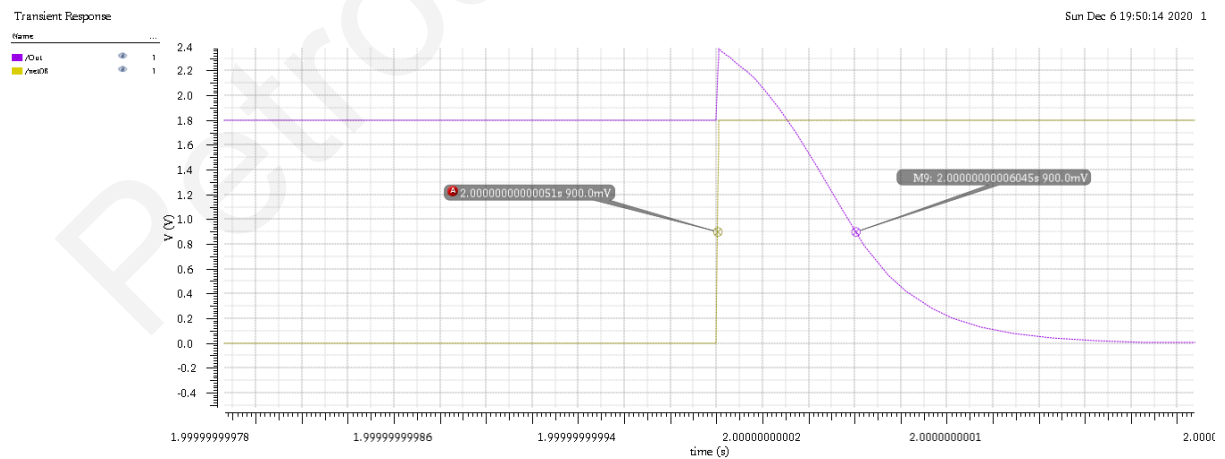


Figure 103: Simulated propagation delay of the 3-input NAND Gate from high to low

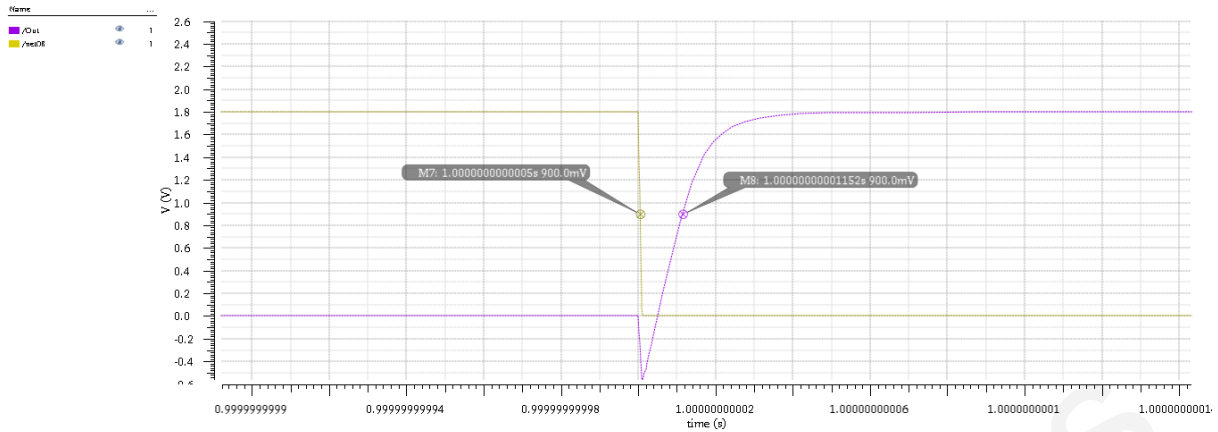


Figure 104: Propagation delay of the 3-input NAND Gate low to high

NAND with 4 inputs

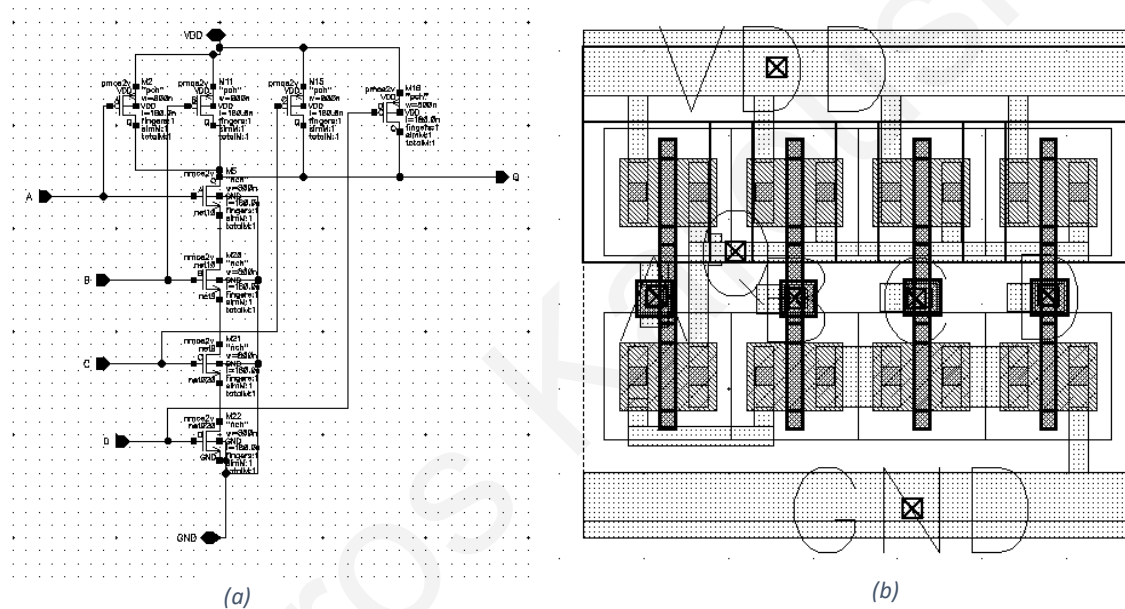


Figure 105: (a) Schematic and (b) Layout of the NAND gate with 4 inputs

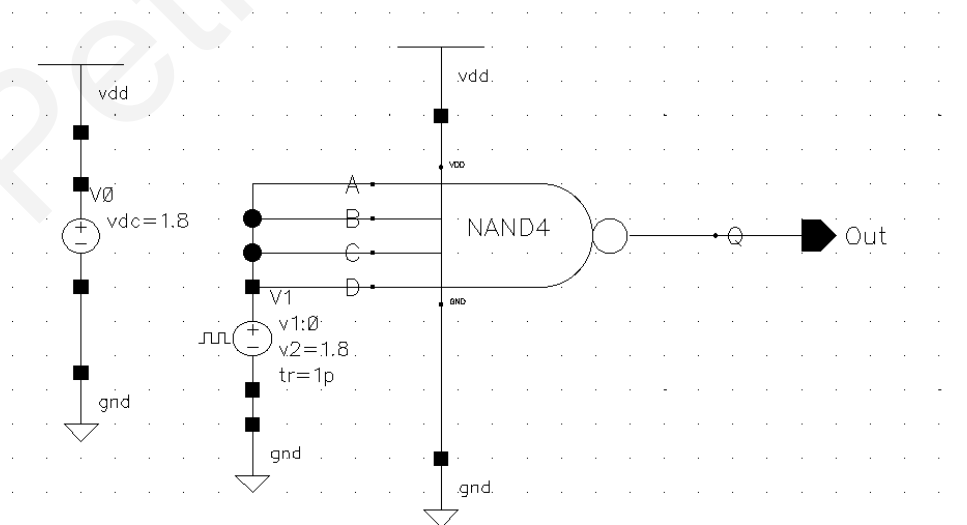
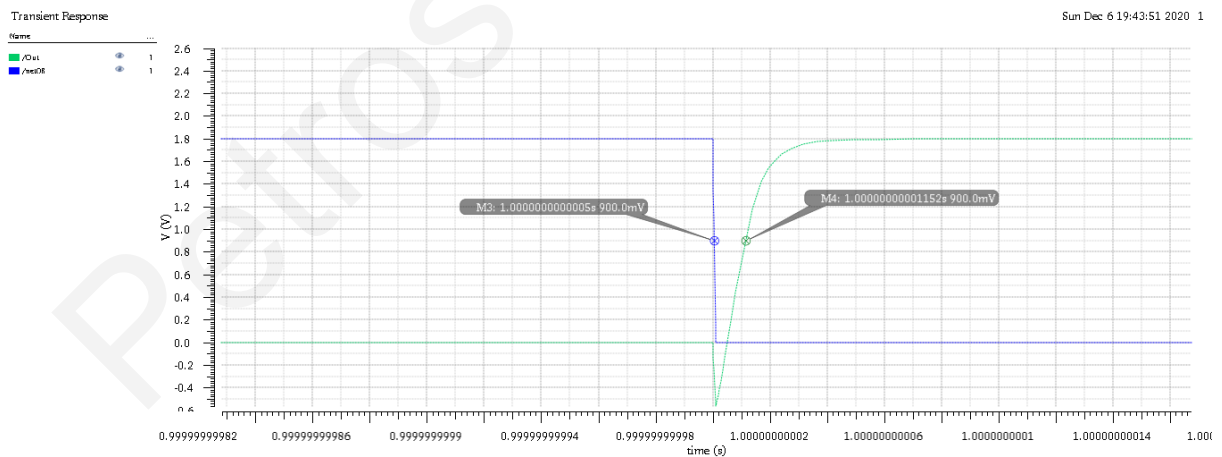
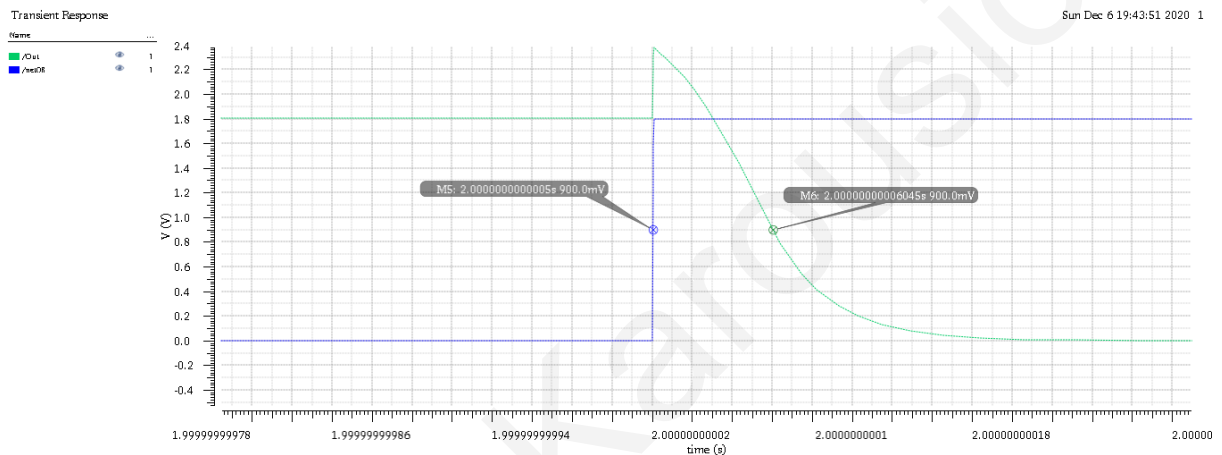
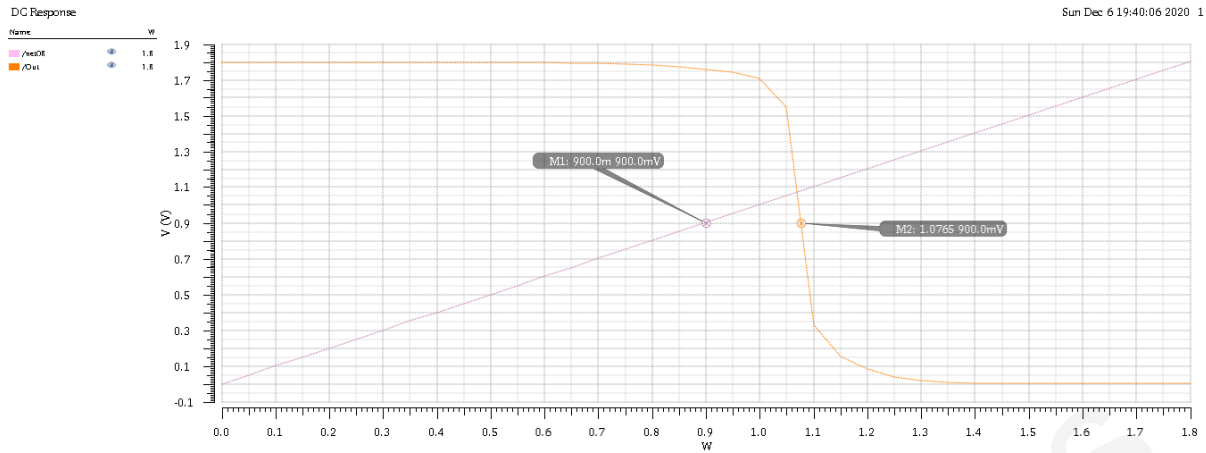


Figure 106: Test bench for simulating 4-input NAND Gate



Device	Width (μm)	Length (μm)
PMOS	0.8	0.18
NMOS	0.8	0.18

Table 9: NAND 3-input and 4-input device sizes

Characteristics	Result
V_{SP} (calculated) 3-input	1.25 V
V_{SP} (measured) 3-input	1.01 V
V_{SP} (calculated) 3-input	1.2667 V
V_{SP} (measured) 3-input	1.075 V
R_n (calculated)	3829 Ω
R_p (calculated)	8800 Ω

Table 10: NAND characteristics

Simulation Results of the Converter

The total current draw from the analogue part of the DAC can be calculated by:

$$\frac{\Delta V}{16 \times R_C} = 489.225 \mu A$$

However, when the system is simulated, the current drawn from the resistors at steady operation is 443.413 μA , which results in $R_C = 253.712 \Omega$ due to mismatches. The total power dissipated by the system is 0.85 mW.

In order to consider a Data Converter accurate, INL and DNL errors must be less than ± 1 LSB, as mentioned before [1].

Integral nonlinearity: Is the difference between the data converter output values and a reference straight line drawn through the first and last output values. It describes the linearity of the overall transfer curve [1].

$$INL_n = (\text{Output value for input code } n) - (\text{Output value of the reference line at that point after Gain and Offset are removed})$$

The following figure shows the calculation of INL after running 200 Monte Carlo simulations. The calculations were programmed using MATLAB. One can observe that the worst case of INL error is less than 0.5 LSB, which is desired.

Differential nonlinearity: Is the difference between the ideal and nonideal values. How well a DAC can generate uniform analogue LSB multiplies at its output [1].

$$DNL_n = (\text{Actual increment height of transition } n) - (\text{Ideal increment height})$$

In the following figure, we can see the results of 200 Monte Carlo simulations, which result in DNL being less than half LSB which translates to monotonicity.

In [1] the maximum INL and DNL can be calculated with an approximation for the Resistor String, which has worse INL and DNL than the Dual-Resistor String. Firstly, assuming that the worst INL occurs at the middle of the resistor string, the maximum INL can be found by:

$$|INL|_{max} = 0.5 \times LSB \times 2^N \times (\% \text{matching})$$

$$|DNL|_{max} = (\% \text{matching}) \times LSB$$

Taking into consideration that poly resistors have mismatch less than 0.7%, $|INL|_{max} = 0.010935$ and $|DNL|_{max} = 0.00048235$, which are less than what we found through Monte Carlo

simulations. This is expected because in Monte Carlo simulations take into consideration process variations, mismatches and temperature variations.

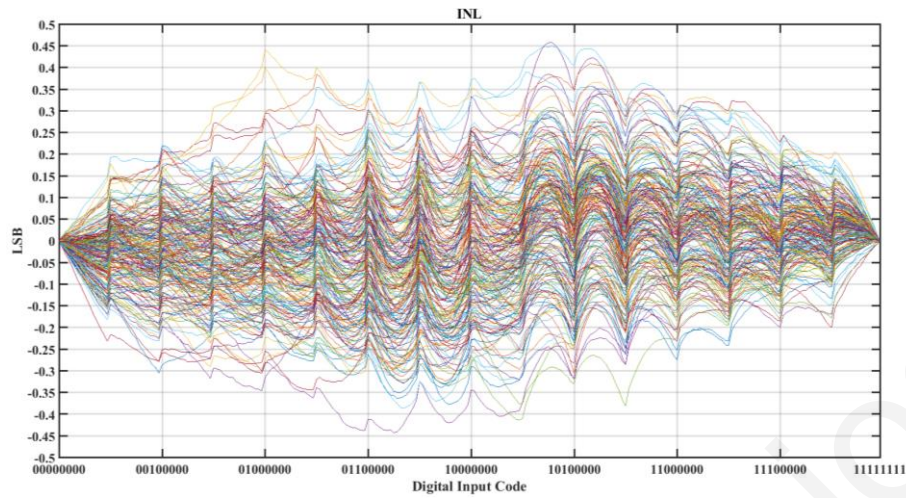


Figure 110: Graphic representation of 200 Monte Carlo simulations showing worst case INL error

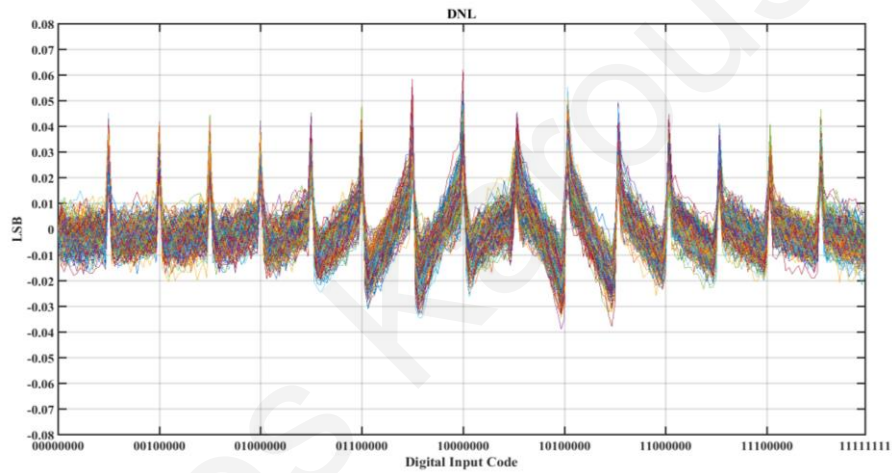


Figure 111: Graphic representation of 200 Monte Carlo simulations showing worst case of DNL error

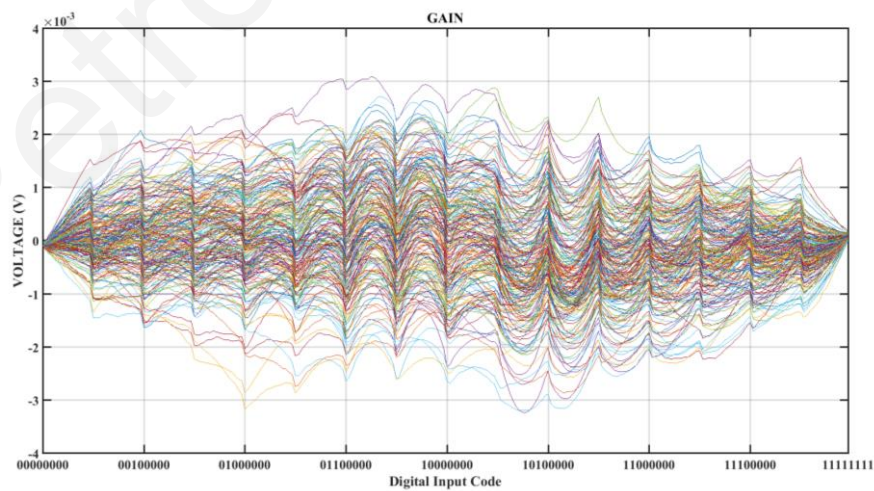


Figure 112: Graphic representation of 200 Monte Carlo simulations showing the worst case of Gain error

The gain offset was calculated at input bit=0. The output at that current state should be zero. After running 200 Monte Carlo simulations, DAC's output is illustrated in the figure 113.

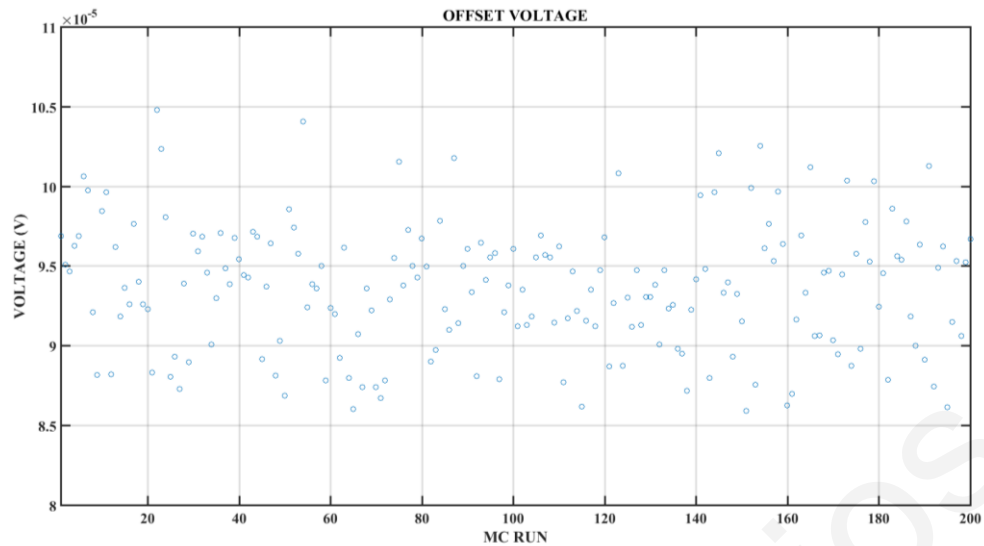


Figure 113: Graphic representation of 200 Monte Carlo simulations showing the worst case of Offset Voltage error
Gain Error: When the best-fit line of the transfer curve is different from the ideal transfer curve.

$$\text{Gain Error} = \text{Ideal Slope} - \text{Actual Slope}$$

Settling time: The total time it takes the output to respond to a step change of input. The input and output are at 0.1% tolerance [6].

Actual Voltage (mV)	Voltage with Tolerance (mV)	Word	Case
892.96875	892.07578	01111111	1
900	899	10000000	
1792.96875	1791.96875	11111111	2
0	1	00000000	

Table 11: Simulated Settling Time



Figure 114: Simulated Settling Time from high to low



Figure 115: Simulated Settling Time from bit=01111111 to bit=10000000

The two figures below illustrate the results of 300 Monte Carlo simulations for the worst case of the DAC, which is the transition from input word 01111111 to 10000000. On the 1st figure, the standard deviation is 0.782 mV. In 3x standard deviation, the result is 2.346 mV which is less than 0.5 $LSB = 3.5156 \text{ mV}$. The same thing applies to the 2nd figure, with standard deviation 0.784 mV.

The Monte Carlo simulations statistical data from the foundry to evaluate expected process variation and mismatches upon fabrication. As mentioned in [48], process variations can be separated into interdie and intradie variations. The first one takes into consideration the variation that can happen between the devices on the same chip. The latter takes into consideration the variations that might occur between chips on the same wafer or in different wafers. Additionally, as stated in [49], mismatches are the differences between two identical elements on the chip.

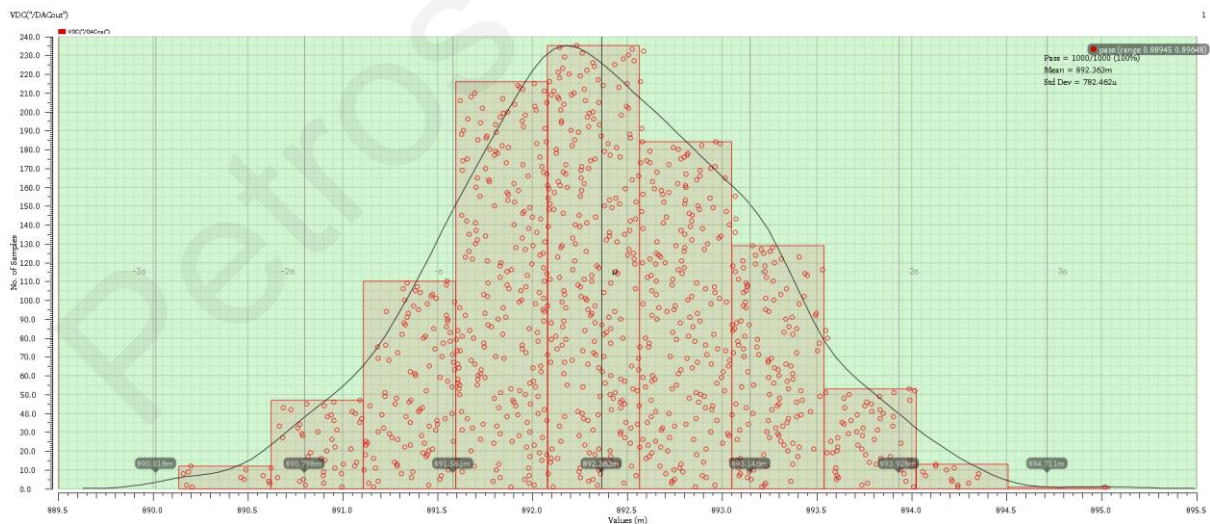


Figure 116: Monte Carlo simulations showing the analogue output at bit=01111111

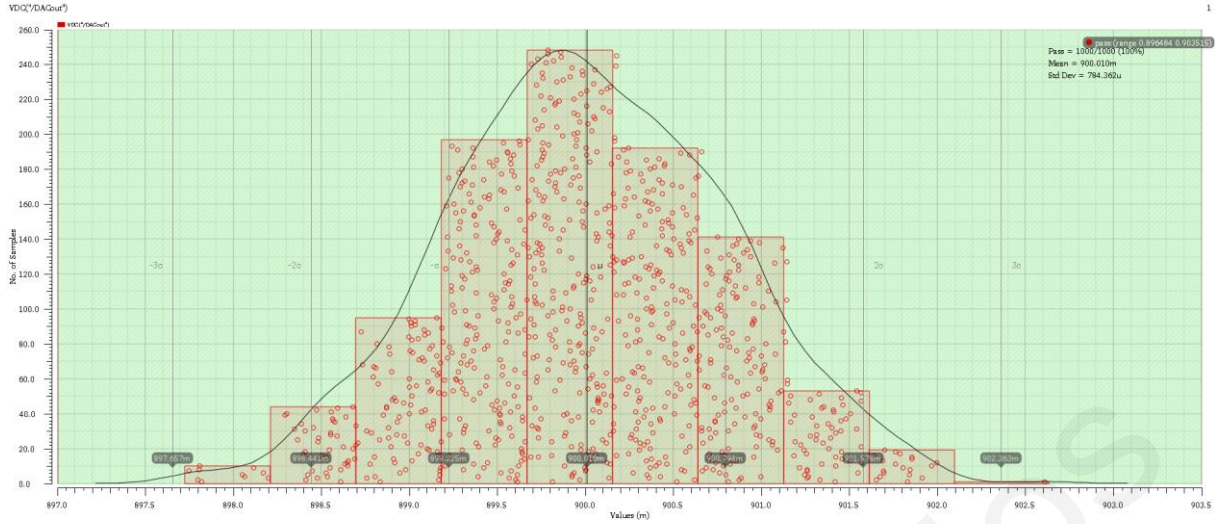


Figure 117: Monte Carlo simulations showing the analogue output at bit=1000000

The Digital to Analogue converter was altered, in order to reduce power consumption resulting in not satisfying the above criterion for resistors. The coarse resistor ladder now has resistors of 5.0017 K Ω thus the total current drawn from the system is now:

$$\frac{\Delta V}{16 \times R_c} = 22.5 \mu A$$

The reason that the above equation is not satisfied is since it will require resistors of larger area. However, due to area restrictions, the resistors of the fine ladder were chosen to be 20.2376 K Ω . This means that the total resistance from the fine resistor string that will be connected to the coarse resistor string will be:

$$R_c || (2^4 * R_f) = 4924.94 \text{ K}\Omega$$

One can understand that this difference of 5.0017 K Ω – 4924.94 K Ω = 76.76 Ω will affect the INL and DNL of the converter. INL and DNL have maximum error approx.. 0.2 LSB. Comparing that result with the first version of the DAC, the INL is significantly reduced, because the overall system uses less current. However, the DNL in the second case is higher due to higher error when the fine resistor string is connected in parallel with the fine resistor string.

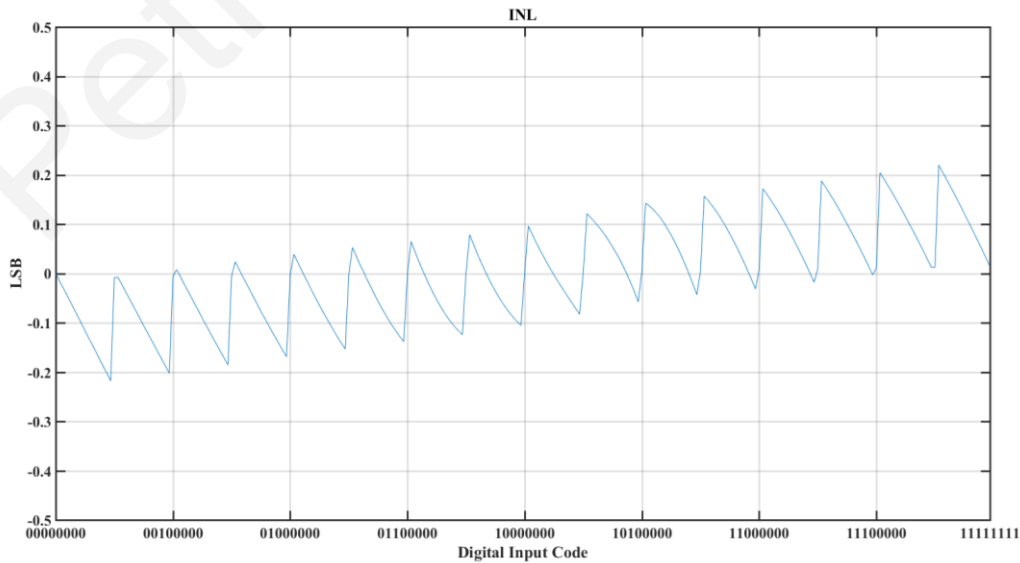


Figure 118: Simulated INL error

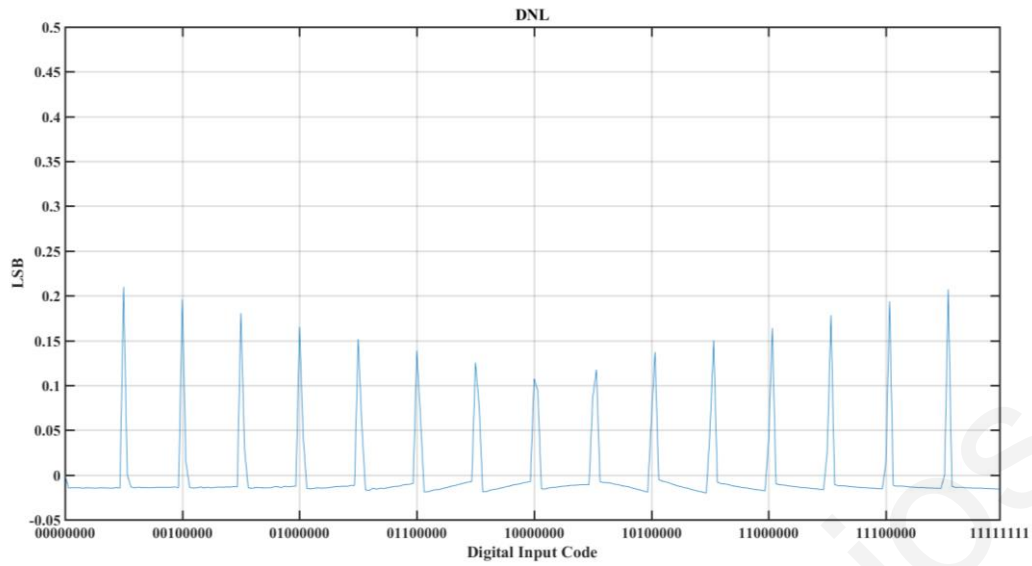


Figure 119: Simulated DNL error

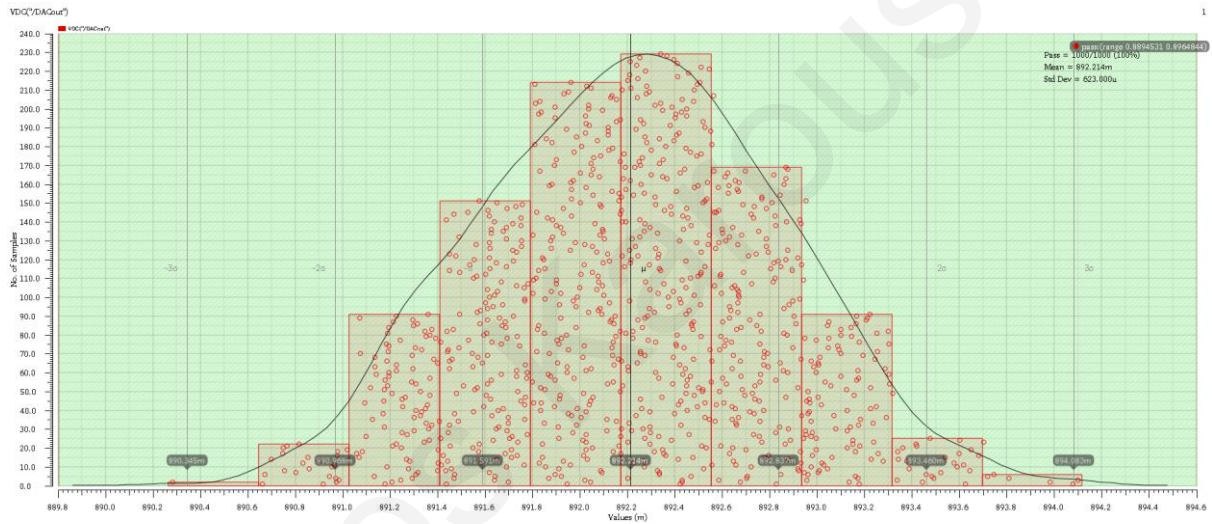


Figure 120: Monte Carlo simulations showing the analogue output at bit=01111111

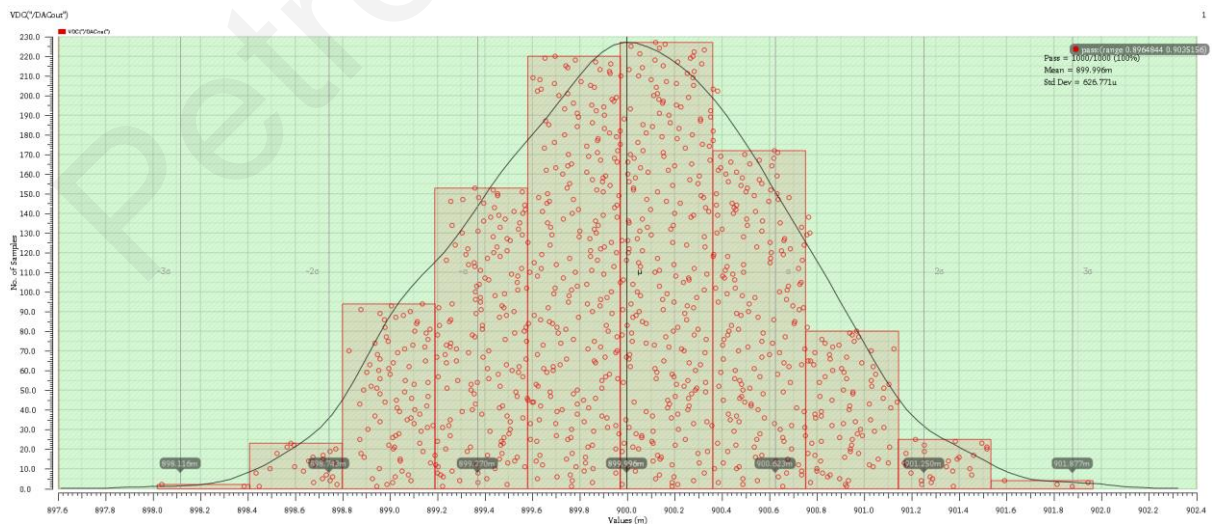


Figure 121: Monte Carlo simulations showing the analogue output at bit=10000000

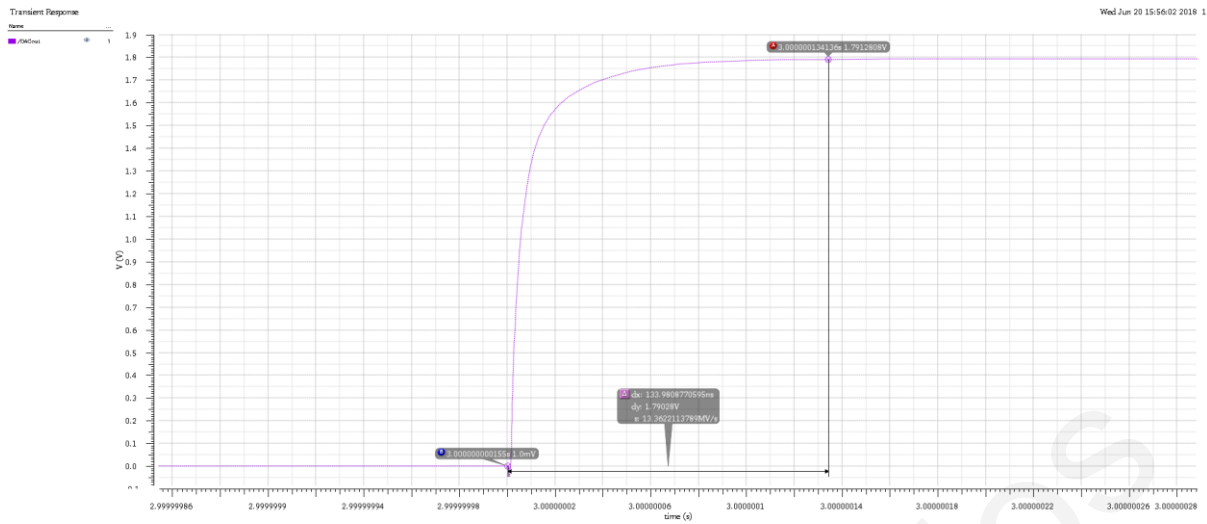


Figure 122: Simulated rise time

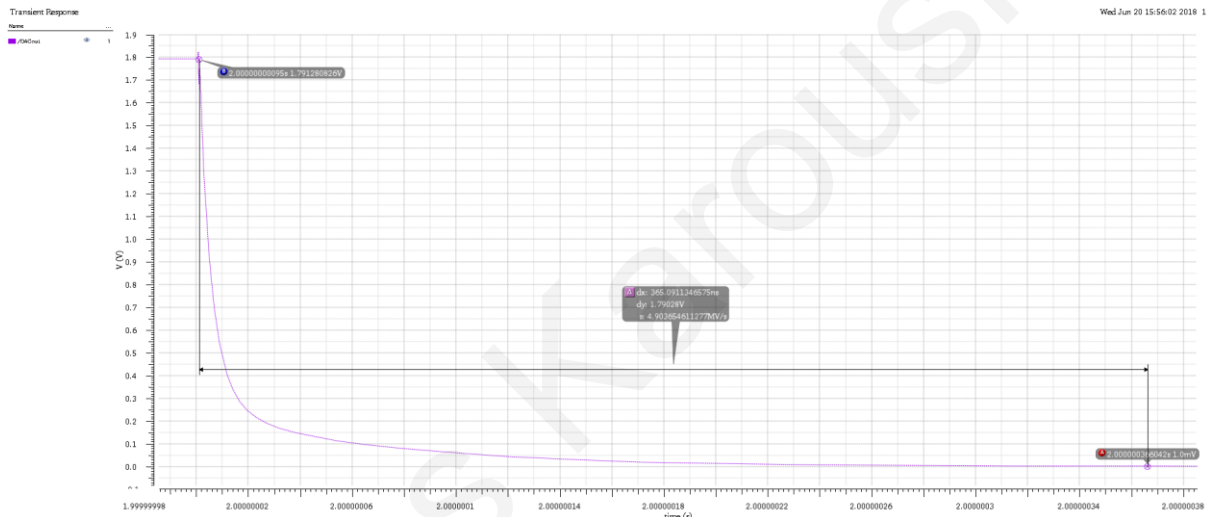


Figure 123: Simulated fall time

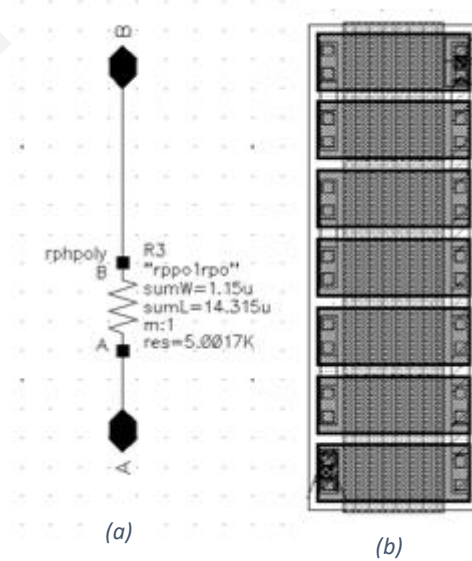


Figure 124: (a) MSB resistor schematic and (b) MSB resistor's layout

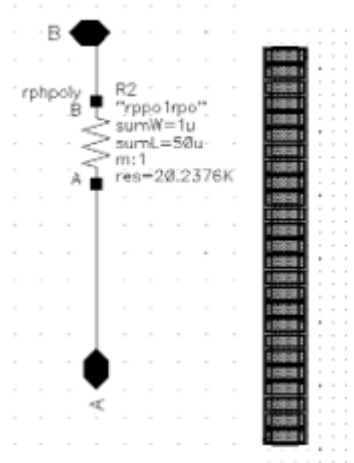


Figure 125: (a) LSB resistor schematic (b) and (c) layout

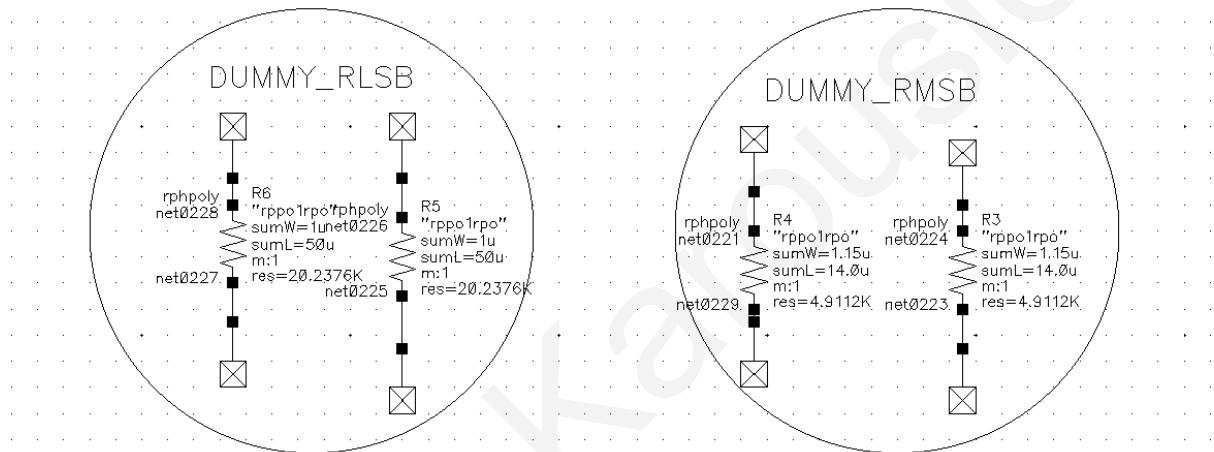


Figure 126: Dummy resistors for (a) LSB and (b) MSB

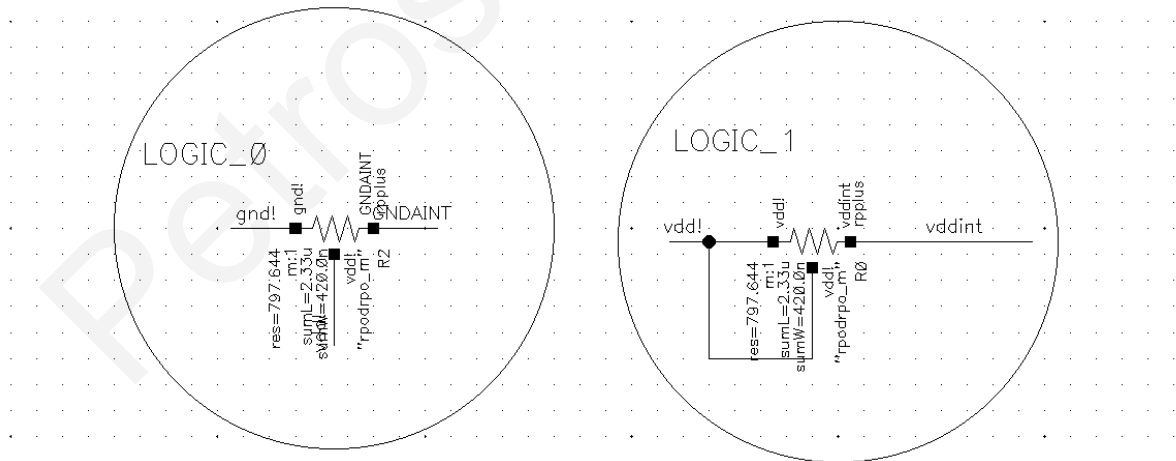


Figure 127: Pull down and pull up resistors used for biasing always on TG switches

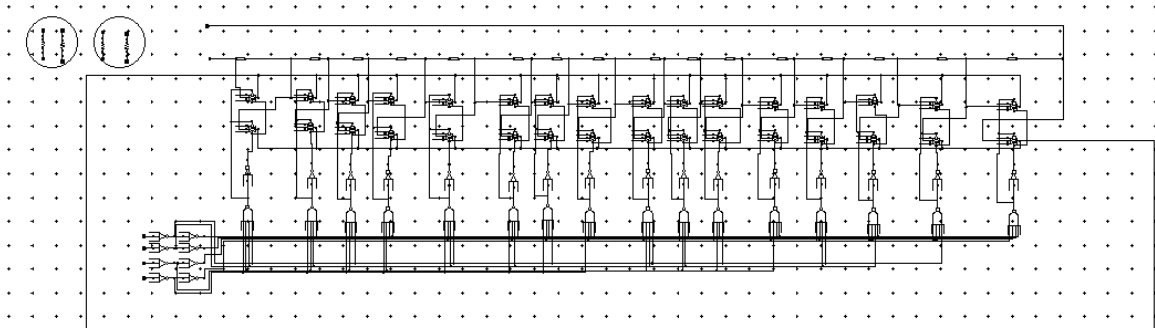


Figure 128: MSB part of the converter

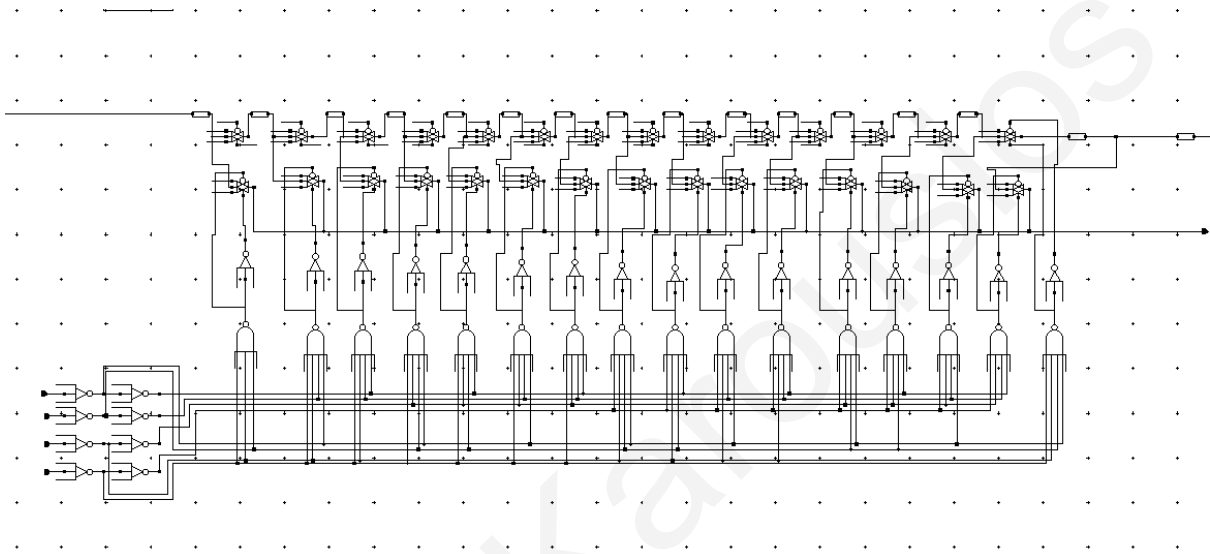
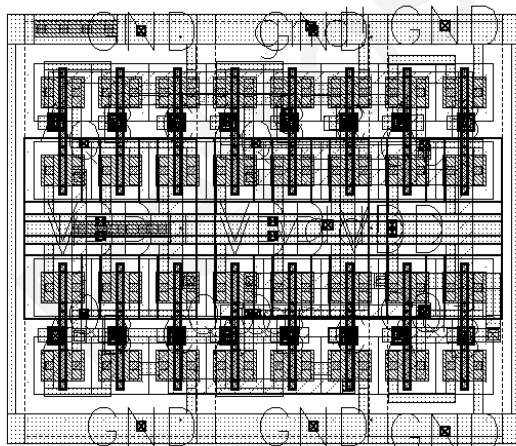
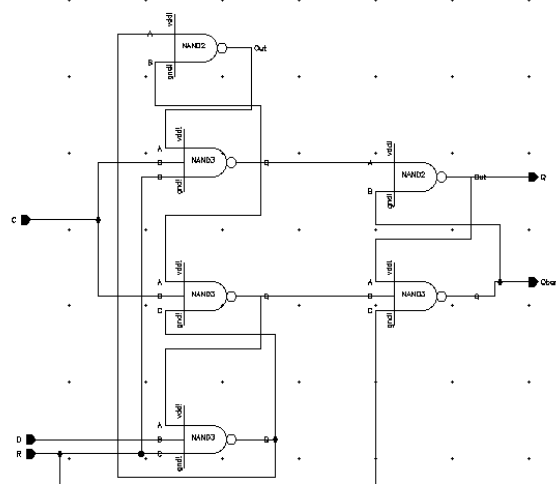


Figure 129: LSB part of the Converter



(a)



(b)

Figure 130:(a) Ripple Counter Layout and (b) D-latch schematic



Figure 131: Converter's Layout

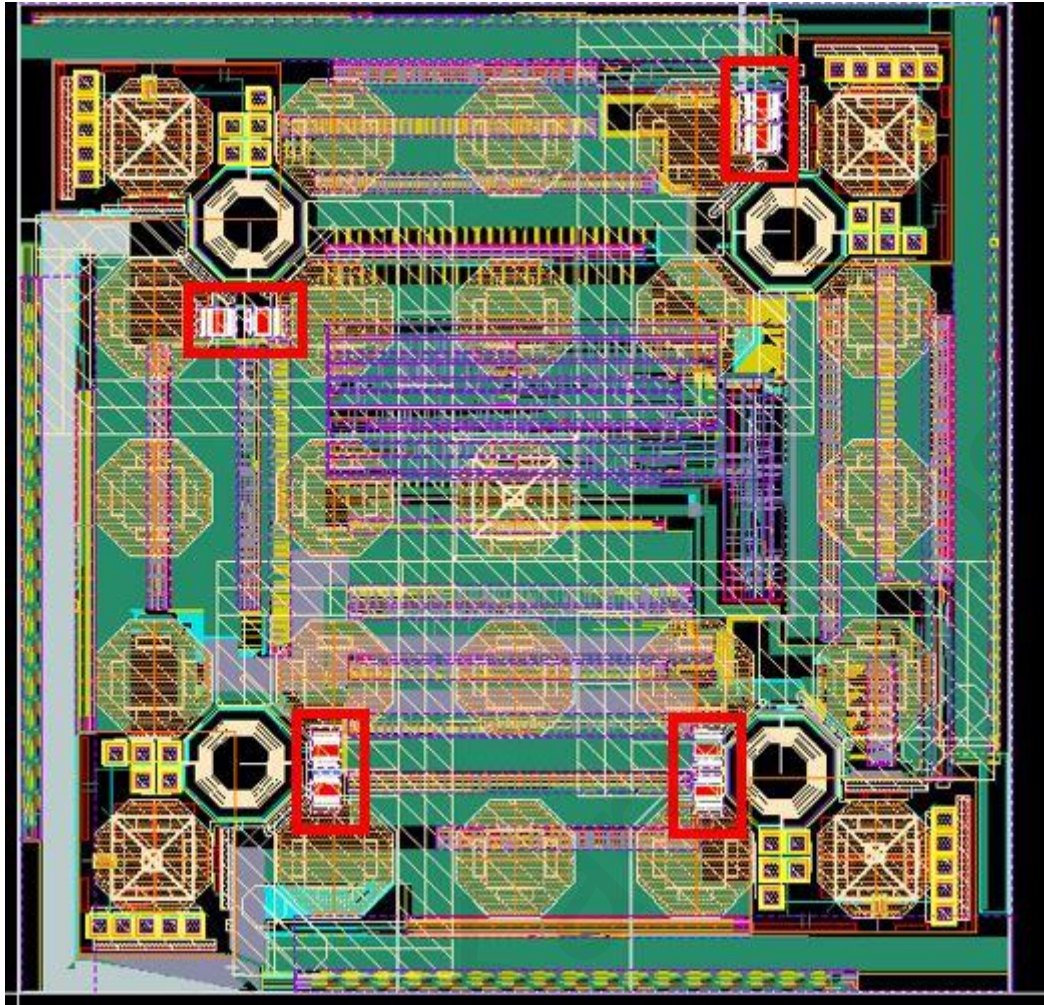


Figure 132: Position of the 8x Converters on the chip.

As already mentioned from figure 89, the propagation delay can be calculated as t_{PLH} and t_{PHL} . The average propagation delay can be calculated by $t_p = \frac{t_{PHL} + t_{PLH}}{2}$. The worst case for propagation delay for varistor and varactor is 3.945 ns and 0.353 μ s respectively.

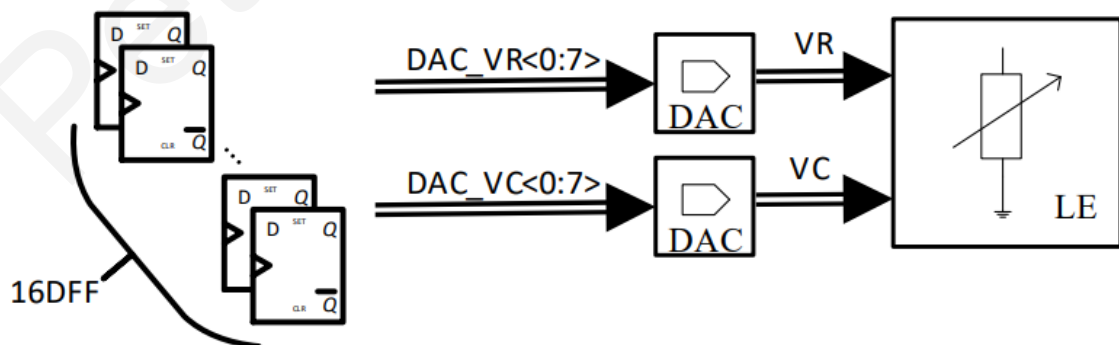


Figure 133: Block diagram for testing propagation delay for the varistor and the varactor [L. Petrou, personal communication, January 18, 2021]

Characterization of Digital to Analogue Converter

Characterizing a device when it is fabricated is crucial not only for the customer but for the designer also, since, in the first case, the customer will have in his/her hands the specifications of the device, and in the second case, the designer testing the device will have proof that the circuit functions properly. This characterization can be performed by a system called Automatic Test Equipment ATE, which can perform multiple tests on multiple chips in seconds, such as the SNR characterization of the device. This system is beneficial since the cost is reduced due to automation whilst the test procedures are duplicated in each test achieving repetition. However, cost is not reduced if the chips are not characterized in bulk. Additionally, programming is a disadvantage alongside with setting the system up [50].

When testing a chip, also known as Device Under Test DUT, a designer can use a specially designed PCB, which will aid in the characterization. The voltage drop on the lines must be mitigated, digital and analogue grounding paths should be not only separated but with long distance between them as well as power-supply decoupling. In this case, no PCB was designed [50].

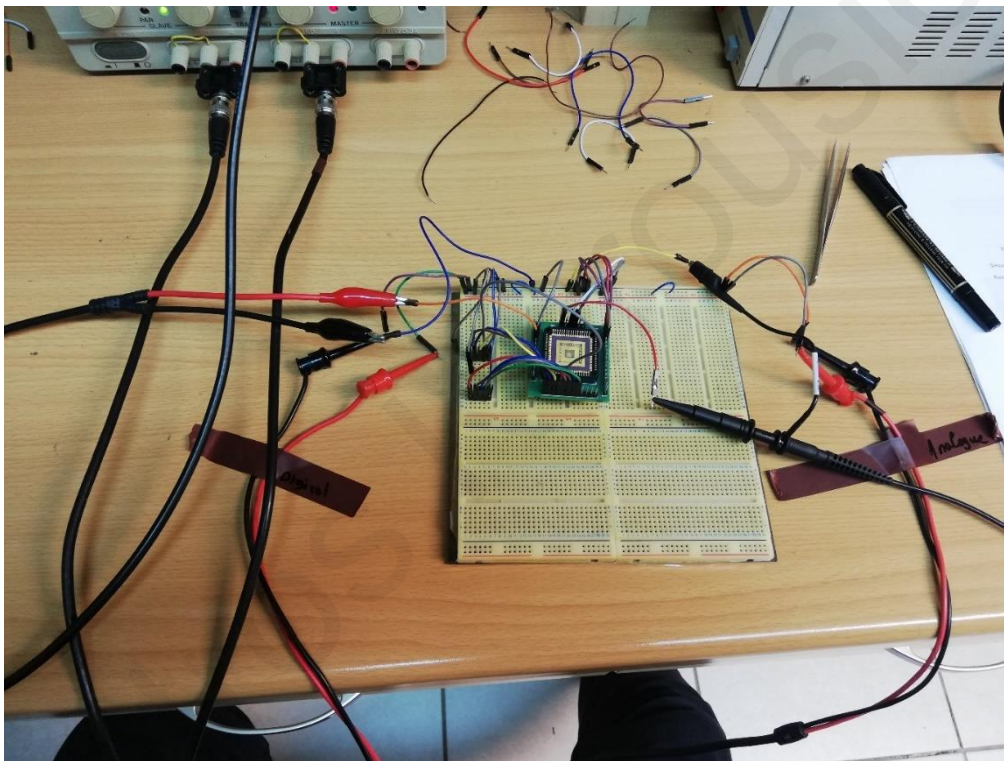


Figure 134: Testing the converter. Using different Power Sources for Digital and Analogue circuits

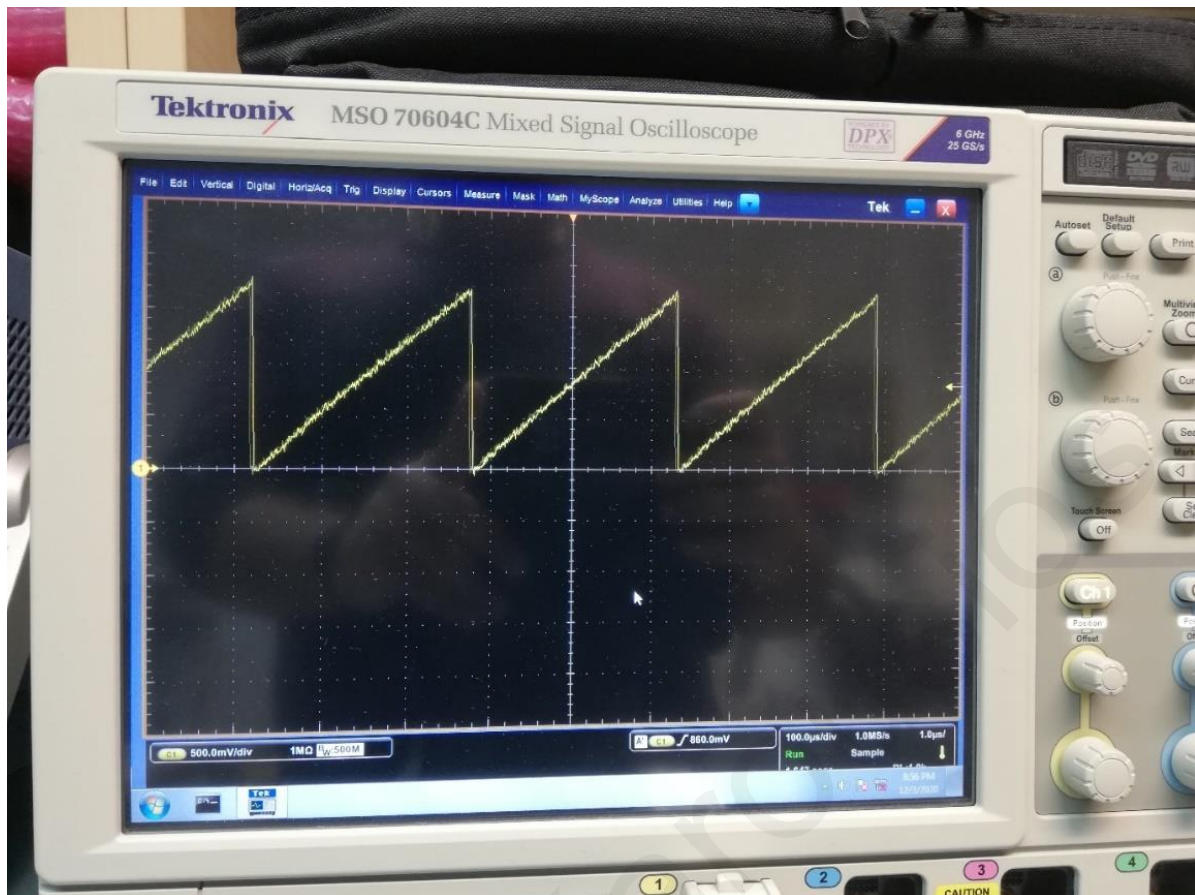


Figure 135: Output of the converter using Ripple Counter at 1MHz frequency

The above figure shows the output of the converter when there is clock input at the Ripple Counter of 1 MHz frequency. We can observe that the converter's output is monotonic under that frequency.



Figure 136: Probe's impedance that was used to test the converter

The above picture shows the probe that was used to measure the output of the converter. It has resistance of great value, which is ideal for our case. However, it also adds delay to the system due to 8 pF capacitance. The settling time becomes:

	From	To	Time (μ S)
Rise Time	00000000	11111111	7.35
	01111111	10000000	15.9
Fall Time	11111111	00000000	7.4
	10000000	01111111	17.7

Table 12: Converters measured settling times

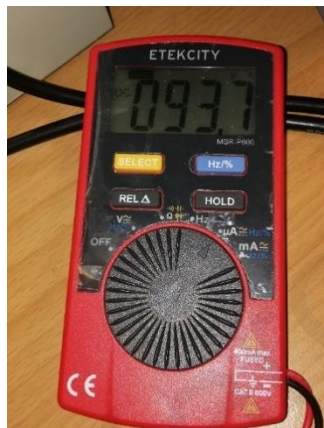


Figure 137: Total current dissipated by the chip

The total current on shown in the above figure, is when 4x DACs are operating. This means that for each DAC the current drawn is 23.425 μ A instead of the calculated current that was 22.5 μ A.

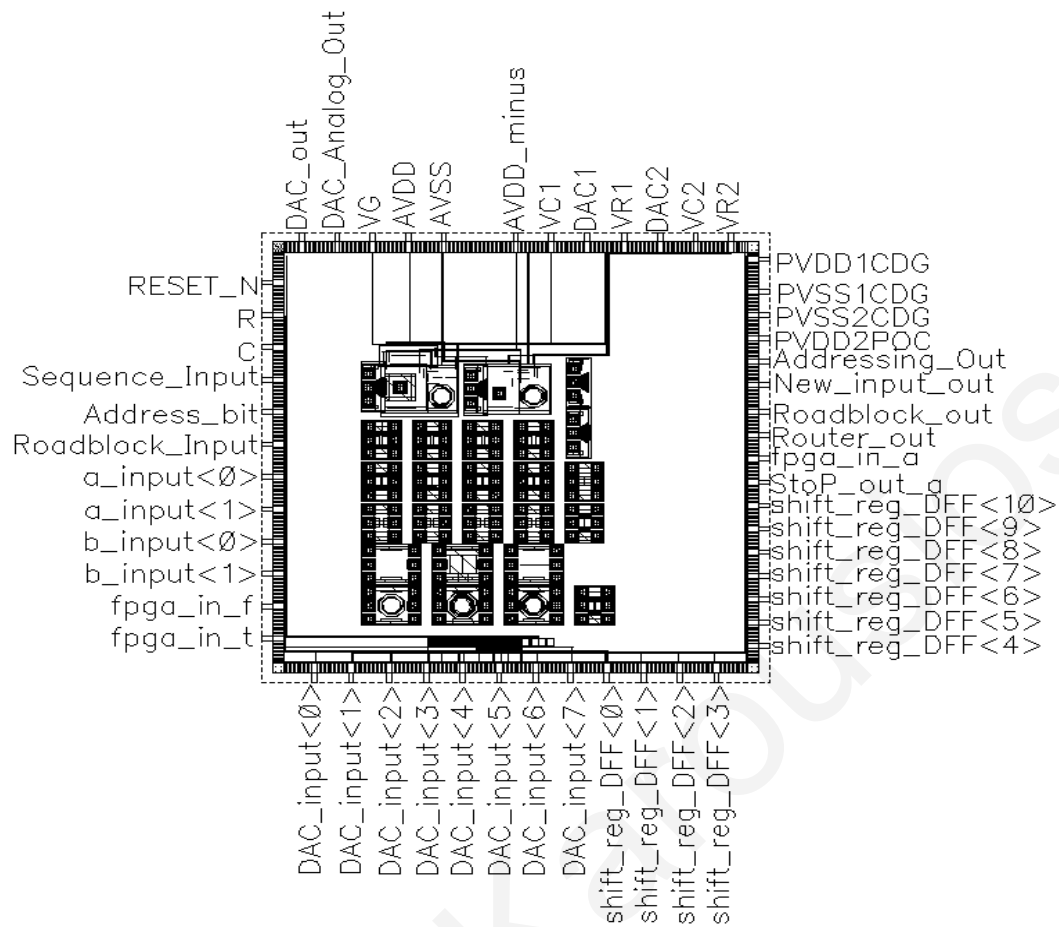


Figure 138: Pin-out of the chip varactor [L. Petrou, personal communication, December 2020]

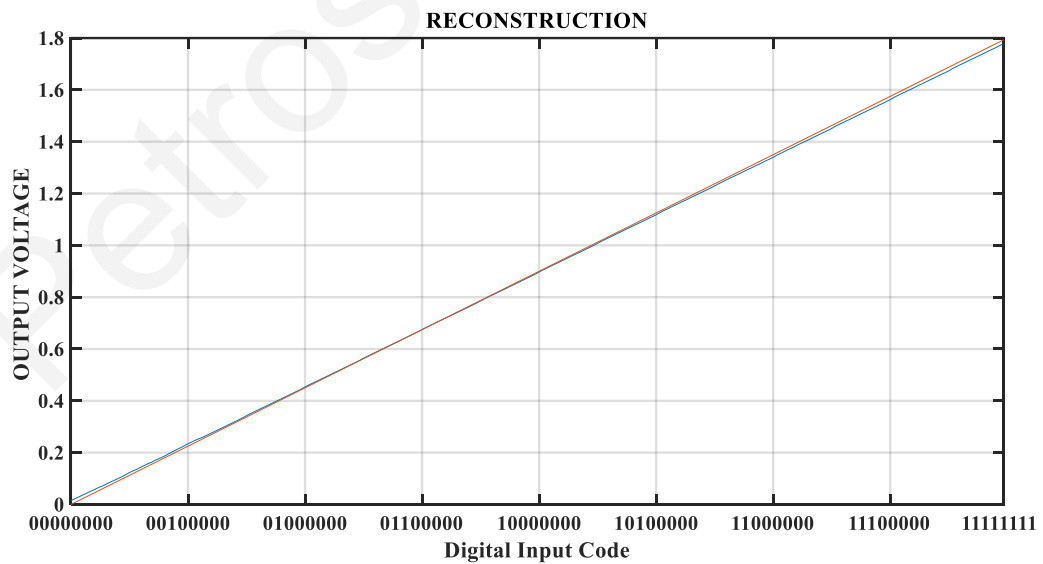


Figure 139: Ideal (purple) vs measured (blue) output of the converter.

The gain of the system can be calculated by using the least-squares technique which is described in [51]

$$G = \frac{\sum_{k=0}^{2^N-1} k \times \sum_{k=0}^{2^N-1} U[k] - 2^N \times \sum_{k=0}^{2^N-1} k \times U[k]}{Q((\sum_{k=0}^{2^N-1} k)^2 - 2^N \times \sum_{k=0}^{2^N-1} k^2)} = 0.9854$$

Where $U[k]$ is the output of the converter at that digital input k , U_0 is the first output of the converter at $k=0$, G is the gain, Q is the LSB of the tested converter, which in our case is 0.00689 V, V_{os} is the offset voltage and $\varepsilon[k]$ is the residual error of the converter at the specific input [51].

$$V_{os} = \frac{(Q \times \sum_{k=0}^{2^N-1} k + 2^N \times U_0) \times \sum_{k=0}^{2^N-1} k \times U[k] - (Q \times \sum_{k=0}^{2^N-1} k^2 + U_0 \times \sum_{k=0}^{2^N-1} k) \times \sum_{k=0}^{2^N-1} U[k]}{Q((\sum_{k=0}^{2^N-1} k)^2 - 2^N \times \sum_{k=0}^{2^N-1} k^2)}$$

With the offset value to be $V_{os} = 0.01039$

The INL, as mentioned above, can be calculated by drawing a line between the first and last values of the DUT and then subtract the output at a certain digital input from the position of the reference line, after the gain and offset parameters have been compensated. This is difficult to happen since gain and offset are not linear thus leaving the only option to calculate INL by using the above results of gain and offset as described in [51].

$$\frac{U[k] - V_{os} - \varepsilon[k]}{G} = Q \times k + U_0$$

$$INL[k] = \frac{\varepsilon[k]}{2^N \times Q \times G}$$

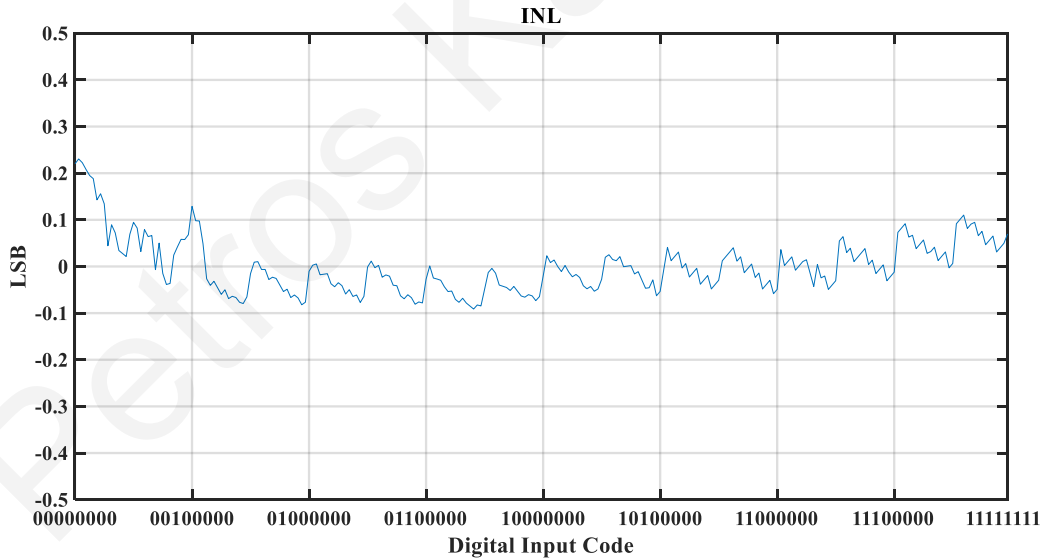


Figure 140: Measured INL error after Gain and Offset errors are removed

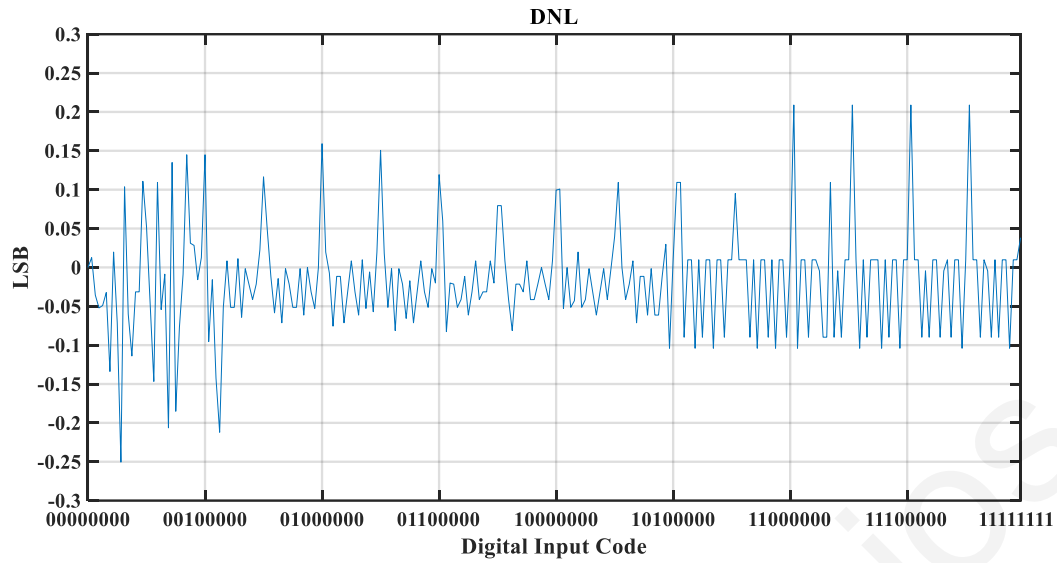


Figure 141: Measured DNL error

Parameter	Value	Explanation
INL (LSB)	< 0.25	<ul style="list-style-type: none"> The coarse resistor connected in parallel with the fine resistor string. Now all the resistors in coarse string do not have the same value. Mismatches make coarse resistors not identical
DNL (LSB)	< 0.25	<ul style="list-style-type: none"> This error is introduced by the current that flows through the selected analog switches that connect the two resistor strings and their “ON” resistance. The dummy switches between the fine resistors must be identical with those of the coarse resistor. Mismatches might force them not to be identical.
Offset (V)	0.01039	<ul style="list-style-type: none"> Mismatch of the components during fabrication [52] <ul style="list-style-type: none"> Stress-induced mismatch [52]
Gain	0.9854	
Measured LSB (mV)	6.8908	<ul style="list-style-type: none"> Mismatches between resistors might create uneven voltage division
Power Dissipation (μ W)	42.165	-

Table 13: Summary of measured results

V. Conclusions

Comparing same architectures

The Dual-Resistor String Designed in this thesis, is obviously not unique and can be compared with same or similar architectures, either state of the art or devices that someone can buy at everyday basis. Nevertheless, it was designed with a particular application in mind, where low power was the top priority and the settling time was the lowest. This DAC fully satisfies the requirements for controlling programmable Metasurfaces.

In [43], the authors designed 10-bit, low power inverted ladder DAC using 0.35 μm technology with 3.3 V power supply voltage and consumes 22 μA current. This design that uses passive compensation like the one in this thesis, has DNL < 0.35 LSB and INL < 0.7 LSB. Another design by Hatamzadeh et al. [53] they designed a Resistor String DAC that uses passive compensation. Additionally, in this design they removed two resistors from the fine resistor string because in addition with the switches, that string divided the voltage to 66 parts instead of 64 (6 bits string). The proposed architecture has a 12-bit resolution, less than 10 ns settling time and was simulated using 0.35 μm technology. [54] Another implementation of a resistor string DAC has resolution of 3 bits and was designed by using 130 nm technology and its purpose was for phone applications. With that being said, INL and DNL errors are quite low, less than 0.0005 LSB with medium power consumption of 361.574 μW . Additionally, the area occupying is the DAC is 32.80 μm x 46.90 μm . In this paper [55], the authors aimed to design an area efficient converter by implementing a 12-bit dual-resistor string DAC, implemented in 180 nm with 1.8 V supply voltage. The INL and DNL errors are less than 0.25 LSB, worst settling time is 82 ns, power consumption is 138 μW . Lastly, Kommangunta et al [56] proposed a dual-resistor string with 8-bits resolution using 180 nm technology with 3.3 V supply voltage and 65.23 μW power consumption. The total area that the converter occupies is 0.0297 mm^2 and maximum DNL < 0.004 LSB and maximum INL < 0.024 LSB.

Parameter	[43]	[53]	[54]	[55]	[56]	This Work
Technology (μm)	0.35	0.35	0.13	0.18	0.18	0.18
Resolution (bits)	10	12	3	12	8	8
Settling Time (ns)	< 15	< 10	< 20	< 82	-	< 365
DNL (LSB)	< 0.35	< 0.2	< 0.0005	< 0.25	< 0.04	< 0.25
INL (LSB)	< 0.7	< 0.3	< 0.0005	< 0.25	< 0.024	< 0.25
Power Consumption (μW)	72.6	-	< 362	138	65.23	42.165

Table 14: Comparison between converters implemented by using similar architectures

Improvements

First of all, the work that was done in this thesis can be improved in many ways. Firstly, as mentioned in [39], by scaling down the fabrication technology, the area used, and the capacitances are reduced. Thus, delays are now reduced and the settling time of the DAC drops. Additionally, due to reduction in supply voltage, the power dissipation is decreased. For example, the drain current results into:

$$I_{D,scaled} = \frac{1}{2} \times \mu_n \times C_{ox} \times \frac{W}{L} \times (V_{GS} - V_{TH})^2 \times \frac{1}{\alpha}$$

Where α is the factor that vertical and lateral dimensions are reduced and $\alpha > 1$. This shows a reduction in the drain current which also applies for the triode region.

Lastly, an improvement that can result in a reduction of INL and DNL errors due to mismatches, is the use of common centroid technique for the resistors described in [46]. The reason that this

technique is needed is due to the effects of stress on the resistors, which is lowest at the center of the die and highest at the corners of the die. The mismatch that occurs due to stress can be found by:

$$\delta_s = \pi_{cc} \times d_{cc} \times \nabla S_{cc}$$

Where δ_s is the stress-induced mismatch, π_{cc} is the piezoresistivity that is formed on a line that connects the two middles (centroids) of the two matched devices, d_{cc} is the distance of that line and ∇S_{cc} is the stress gradient.

Piezoresistivity can be improved by either placing the resistors facing the less stress direction of the die or either by creating the resistances with a material that is not susceptible to stress gradient. The stress gradient can be decreased by either placing the resistors on a low stress location on the die or by choosing appropriate packaging.

The distance of the line that connects the centroids can be decreased by the common centroid technique. Firstly, the devices must be split into smaller equal devices. Therefore, when they are placed, they can share the same axis of symmetry thus reducing distance d_{cc} .

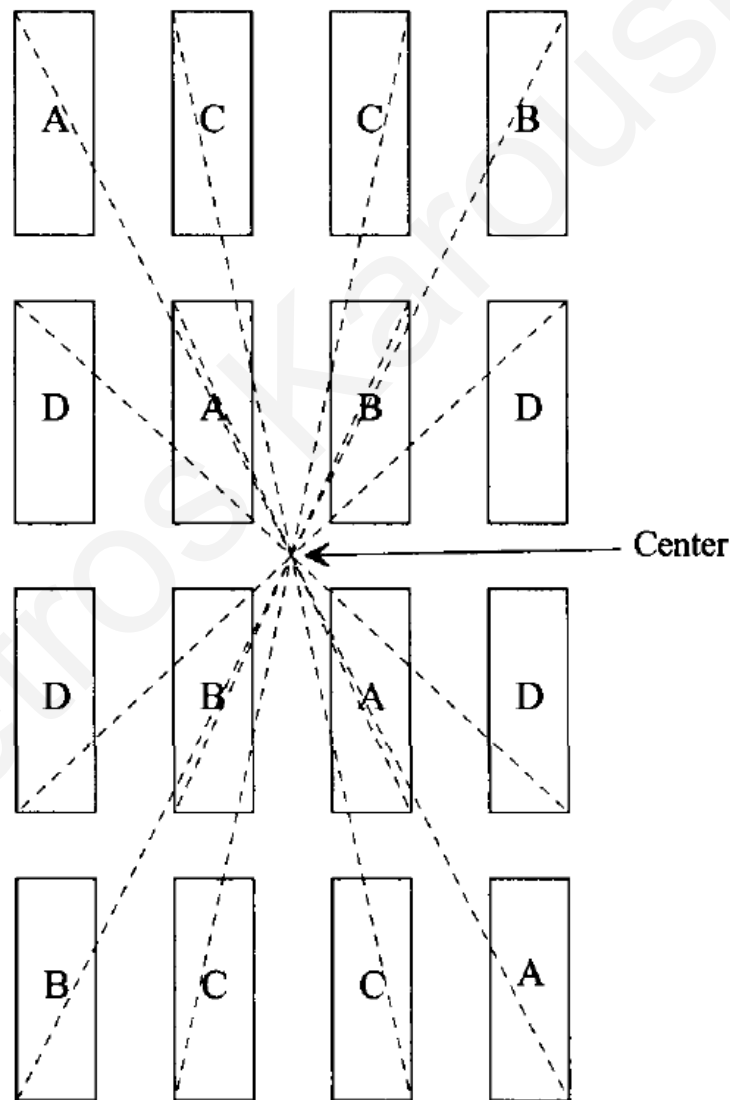


Figure 142: Common Centroid technique for used for matching four resistors [1]

Future work

Despite from scaling down the fabricating technology, there are not much to research for future implementations. However, a very promising use of a resistor string DAC is proposed in [57]. The authors developed a resistor string DAC by using memristive devices with 4 bits resolution. Memristors is a two-port passive element and can be described by flux ϕ and charge q . The converter has static characteristics $INL = 0.123 \text{ LSB}$ and $DNL = 0.075 \text{ LSB}$.

VI. References

- [1] R. J. Baker, *CMOS Circuit Design, Layout, and Simulation*. 2010.
- [2] “8-CHANNEL 0-24V ANALOG TO DIGITAL CONVERTER ADC.”
<https://shop.controleverything.com/products/8-channel-0-24v-analog-to-digital-converter-adc>.
- [3] Visorsurf Project, “VISORSURF,” 2017. <http://www.visorsurf.eu/> (accessed Apr. 21, 2020).
- [4] K. M. Kossifos *et al.*, “Toward the Realization of a Programmable Metasurface Absorber Enabled by Custom Integrated Circuit Technology,” *IEEE Access*, vol. 8, 2020.
- [5] L. Petrou, P. Karousios, and J. Georgiou, “Asynchronous Circuits as an Enabler of Scalable and Programmable Metasurfaces,” *IEEE Int. Symp. Circuits Syst.*, 2018.
- [6] W. M. C. Sansen, *Analog Design Essentials*. Springer, 2006.
- [7] C. Shi, J. Wilson, and M. Ismail, “Design techniques for improving intrinsic accuracy of resistor string DACs,” *ISCAS 2001 - 2001 IEEE Int. Symp. Circuits Syst. Conf. Proc.*, vol. 1, pp. 400–403, 2001, doi: 10.1109/ISCAS.2001.921877.
- [8] B. Jr Latinus, K. Brown, and R. Geiger, “Enhancing Performance in Interpolating Resistor String DACs,” pp. 541–544.
- [9] S. Saponara, T. Baldetti, L. Fanucci, E. Volpi, and F. D’Ascoli, “High-level modeling of resistor string based digital-to-analog converters,” *Analog Integr. Circuits Signal Process.*, vol. 66, no. 3, pp. 407–416, 2011, doi: 10.1007/s10470-010-9544-y.
- [10] B. Razavi, “The Current-Steering DAC [A Circuit for all Seasons],” *IEEE Solid-State Circuits Magazine*, pp. 11–15, 2018.
- [11] E. Balestrieri, P. Daponte, and S. Rapuano, “Digital to analog converters: A metrological overview,” *9th IMEKO Work. ADC Model. Testing, IWADC 2004, Held Together with 13th IMEKO TC4 Symp. Meas. Res. Ind. Appl.*, no. January, pp. 52–57, 2004.
- [12] D. J. Liu, C. H. Lin, S. C. Yi, and J. J. Chen, “A resistor string DAC for video processing,” *Proc. - 3rd Int. Conf. Intell. Inf. Hiding Multimed. Signal Process. IIHMSP 2007.*, vol. 2, pp. 503–506, 2007, doi: 10.1109/IIH-MSP.2007.56.
- [13] M. Kosakowski, R. Wittmann, and W. Schardein, “Statistical averaging based linearity optimization for resistor string DAC architectures in nanoscale processes,” *2008 IEEE Int. SOC Conf. SOCC*, pp. 261–265, 2008, doi: 10.1109/SOCC.2008.4641524.
- [14] T. Shi *et al.*, “A Low Power and High Precision DAC in 0.13 μ m CMOS for DVS System,” no. i, pp. 67–70, 2010.
- [15] T. D. Godfrey, A. A. Eilsen, and A. J. Fleming, “Digital-to-analog converter considerations for achieving a dynamic range of 1 ppm in precision mechatronics systems,” *2015 IEEE Conf. Control Appl. CCA 2015 - Proc.*, pp. 786–791, 2015, doi: 10.1109/CCA.2015.7320713.
- [16] S. Wang, M. Ahmad, and B. Bhattacharya, “A Novel Cyclic D / A Converter,” pp. 1224–1227, 1993.
- [17] P. Chen and T. Liu, “Switching Schemes for Reducing Capacitor Mismatch Sensitivity of Quasi-Passive Cyclic DAC,” *IEEE Trans. Circuits Syst. II Express Briefs*, vol. 56, no. 1, pp. 26–30, 2009.
- [18] K. S. Lee and Y. M. Lee, “Switched-capacitor cyclic DAC with mismatch charge compensation,” *Electron. Lett.*, vol. 46, no. 13, pp. 902–903, 2010, doi: 10.1049/el.2010.1182.
- [19] D. Marche, Y. Savaria, and Y. Gagnon, “A New Switch Compensation Technique for Inverted R-2R Ladder DACs,” vol. 53, no. 9, pp. 1689–1699, 2005, doi:

10.1017/CBO9781107415324.004.

- [20] D. S. Karadimas, D. N. Mavridis, and K. A. Efstathiou, "A Digitally Calibrated R-2R Ladder Architecture for High Performance Digital-to -Analog Converters," *ISCAS*, pp. 4779–4782, 2006.
- [21] D. Karadimas, M. Papamichail, and K. Efstathiou, "A MOST-Only R-2R Ladder-Based Architecture for High Linearity DACs," *IEEE*, pp. 158–161, 2008.
- [22] S. Domingues, L. H. C. Braga, and A. C. Mesquita, "A CMOS-only R-2R Ladder D / A Converter for Image Sensor Applications," *Int. Conf. Signals, Circuits Syst.*, pp. 1–6, 2009.
- [23] T.-C. Lee and C.-H. Lin, "Nonlinear R-2R Transistor-Only DAC," *IEEE Trans. Circuits Syst. I Regul. Pap.*, vol. 57, no. 10, pp. 2644–2653, 2010.
- [24] A. A. Noorwali, S. M. Qasim, A. S. Doost, and A. Huynh, "A 16-bit 4 MSPS DAC for lock-in amplifier in 65nm CMOS," *Proc. 2016 IEEE 13th Int. Conf. Networking, Sens. Control*, pp. 4–8, 2016.
- [25] B. Razavi, "A Circuit for All Seasons: The R-2R and C-2C Ladders," *IEEE Solid-State Circuits Mag.*, no. c, pp. 10–15, 2019.
- [26] D. H. Lee, T. H. Kuo, and K. L. Wen, "Low-cost 14-bit current-steering DAC with a randomized thermometer-coding method," *IEEE Trans. Circuits Syst. II Express Briefs*, vol. 56, no. 2, pp. 137–141, 2009, doi: 10.1109/TCSII.2008.2011606.
- [27] P. Aliparast, Z. D. Koozehkanai, J. Sobhi, N. Nasirzadeh, and B. A. Soltany, "Design of a 10-bit low power current-steering digital-to-analog converter based on a 4-D thermometer decoding matrix," *Proc. 17th Int. Conf. - Mix. Des. Integr. Circuits Syst. Mix. 2010*, pp. 243–246, 2010.
- [28] Analog Devices Inc Engineering, *The Data Conversion Handbook*. 2004.
- [29] Maxim Integrated, "INL / DNL Measurements for High-Speed Analog-to-Digital Converters (ADCs)," *Maxim*, vol. 283, pp. 1–9, 2000, [Online]. Available: <http://www.maxim-ic.com/app-notes/index.mvp/id/283>.
- [30] Maxim, "Understanding SAR ADCs : Their Architecture and Comparison with Other ADCs," pp. 1–7, 2001, [Online]. Available: <https://www.maximintegrated.com/en/app-notes/index.mvp/id/1080>.
- [31] J. LeClare, "A Simple ADC Comparison Matrix," *Maxim - Appl. Note - Tutor.*, pp. 1–4, 2003, [Online]. Available: <http://pdfserv.maximintegrated.com/en/an/AN2094.pdf> <http://www.maximintegrated.com/app-notes/index.mvp/id/2094>.
- [32] a. Golda and a. Kos, "Analysis And Design Of Ptat Temperature Sensor In Digital CMOS VLSI Circuits," *Proc. Int. Conf. Mix. Des. Integr. Circuits Syst. 2006. Mix. 2006.*, no. 2, pp. 415–420, 2006, doi: 10.1109/MIXDES.2006.1706612.
- [33] S. Chaparro, J. J. Carrillo, and H. Alarcon, "A high-slope PTAT temperature sensor for frequency compensation of an RTC oscillator," *LASCAS 2016 - 7th IEEE Lat. Am. Symp. Circuits Syst. R9 IEEE CASS Flagsh. Conf.*, vol. 1, pp. 111–114, 2016, doi: 10.1109/LASCAS.2016.7451022.
- [34] A. Go and A. Kos, "A Set of Temperature Sensors and Maximum Temperature Selection Circuit," pp. 7–10, 2013.
- [35] B. Razavi, *Design of Analog CMOS Integrated Circuits*. 2001.
- [36] P. Identification, O. F. Embedded, T. Sensors, and F. O. R. Cmos, "Temperature Sensors for

- Cmos Circuits,” *Computer (Long. Beach. Calif.)*, pp. 5–8, 2007.
- [37] P. Harikumar, J. J. Wikner, and A. Alvandpour, “A 0.4-V Subnanowatt 8-Bit 1-kS/s SAR ADC in 65-nm CMOS for Wireless Sensor Applications,” *IEEE Trans. Circuits Syst. II Express Briefs*, vol. 63, no. 8, pp. 743–747, 2016, doi: 10.1109/TCSII.2016.2531099.
 - [38] R. Hedayati, “A Study of Successive Approximation Registers and Implementation of an Ultra-Low Power 10-bit SAR ADC in 65nm CMOS Technology,” 2017.
 - [39] B. Razavi, *Design of Analog CMOS Integrated Circuits*. 2001.
 - [40] B. Razavi, “The Bandgap Reference [A Circuit for All Seasons],” *IEEE Solid-State Circuits Mag.*, vol. 8, no. 3, pp. 9–12, 2016, doi: 10.1109/MSSC.2016.2577978.
 - [41] S. Koudounas, C. M. Andreou, and J. Georgiou, “A novel CMOS bandgap reference circuit with improved high-order temperature compensation,” *ISCAS 2010 - 2010 IEEE Int. Symp. Circuits Syst. Nano-Bio Circuit Fabr. Syst.*, no. 5, pp. 4073–4076, 2010, doi: 10.1109/ISCAS.2010.5537621.
 - [42] A. Martinez-Nieto, M. T. Sanz-Pascual, P. Rosales-Quintero, and S. Celma, “A bandgap voltage reference in 0.18 μ m CMOS technology,” *Midwest Symp. Circuits Syst.*, no. August, pp. 97–100, 2013, doi: 10.1109/MWSCAS.2013.6674594.
 - [43] Y. Perelman and R. Ginosar, “A Low-Power Inverted Ladder D/A Converter,” *IEEE Trans. Circuits Syst. II Express Briefs*, vol. 53, no. 6, pp. 497–501, 2006, doi: 10.1109/TCSII.2006.875313.
 - [44] R. Rivoir, F. Maloberti, and G. P. Torelli, “Digital to Analog Converter with Dual Resistor String,” 5,703,588, 2013.
 - [45] A. Grebennikov, *RF and Microwave Transistor Oscillator Design*. 2007.
 - [46] A. Hastings, “The Art of Analog Layout,” *Liquid Crystal Displays*. pp. 307–320, 2011, doi: 10.1002/9781118084359.ch21.
 - [47] N. E. H. Weste and D. M. Harris, *CMOS VLSI DESIGN: A CIRCUITS AND SYSTEMS PERSPECTIVE*. .
 - [48] M. Alioto, G. Palumbo, and M. Pennisi, “Understanding the effect of process variations on the delay of static and domino logic,” *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 18, no. 5, pp. 697–710, 2010, doi: 10.1109/TVLSI.2009.2015455.
 - [49] M. Terrovitis and C. J. Spanos, “Process Variability And Device Mismatch,” no. April 1998, 1996, [Online]. Available: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.40.2539>.
 - [50] F. Maloberti, *Data Converters*. 2007.
 - [51] IEEE Instrumentation and Measurement Society, *IEEE Standard for Terminology and Test Methods of Digital-to-Analog Converter Devices*, no. February. 2012.
 - [52] R. Palmer, “DC Parameters : Input Offset Voltage (V_{IO}),” *Texas Instruments*, no. March, pp. 1–24, 2001.
 - [53] N. Hatamzadeh, H. Esmailbeygi, and M. Lotfizad, “A New 12-bit 100-MS / Sec Resistor String DAC in 0.35- μ m CMOS Process,” *5th Conf. Knowledge-Based Eng. Innov. Iran Univ. Sci. Technol. Tehran, Iran*, pp. 316–321, 2019.
 - [54] F. Farah, M. El Alaoui, K. El Khadiri, H. Qjidaa, and A. Lakhassassi, “A design of a new resistor string DAC for phones applications in 130nm technology,” *2018 Int. Conf. Intell. Syst. Comput. Vision, ISCV 2018*, vol. 2018-May, no. 1, pp. 1–4, 2018, doi: 10.1109/ISACV.2018.8354063.

- [55] B. Yang, Y. Shin, K. Ryu, J. Min, and S.-W. Sung, "An Area-Efficient Coarse-Fine Resistor-String D/A Converter," vol. 5, no. 6, pp. 451–453, 2002.
- [56] V. Kommangunta, K. Shehzad, D. Verma, P. Kumar, and K. Y. Lee, "Low-Power Area-Efficient 8-bit Coarse-Fine Resistor-String DAC," *2020 IEEE Int. Conf. Consum. Electron. - Asia, ICCE-Asia 2020*, pp. 8–10, 2020, doi: 10.1109/ICCE-Asia49877.2020.9276800.
- [57] M. Jahromi, J. Shamsi, K. Mohammadi, and R. Sabbaghi-nadooshan, "Memristor based Digital-to-Analog Converter and its Programming," in *23rd Iranian Conference on Electrical Engineering (ICEE)*, 2015, pp. 1352–1356.

VII. Appendix I: MATLAB code for calculating DAC's parameters

```
format longg
%ideal DAC
VREF=1.8;
N=8;
Vstep=1.8/(2^N);
for i=0:255
    Vout(i+1,1)=Vstep*i;
end
Cd=xlsread('Z2.xlsx');
Vout(:,2)=Vout(:,1)/VREF; %the output is up to 1
figure
plot(Cd(:,1),Vout(:,2),'LineWidth',2); %normalized figure
Y=0;
YY=1;
X=0;
XX=255;
axis([X XX Y YY]);
xticks([0 20 40 60 80 100 120 140 160 180 200 220 240]);
title('IDEAL DAC','FontSize',15,'FontWeight','bold','FontName','Times New Roman');
set(gca,'YTick',[Y:0.05:YY]);
set(gca,'xticklabel',dec2bin(get(gca,'xtick'),8))
ylabel('Vout/Vref','FontSize',18,'FontWeight','bold','FontName','Times New Roman');
xlabel('Digital Input Code','FontSize',18,'FontWeight','bold','FontName','Times New Roman');
set(gca,'LineWidth',2,'FontSize',18,'FontWeight','Bold','FontName','Times New Roman');
grid on;
hold on;
stairs(Cd(:,1),Vout(:,2),'LineWidth',2);
%%%%%%%%%%%%%%
V=xlsread('TEST.xlsx'); %%%the exported vcsv from cadence
i=1;
while i<257
    Reconstruction(:,1)=V(:,1);
    %V2(i,1)=(V(i,(2))-V(1,(2)))/VREF;%Correction of the offset
    V2(i,1)=(V(i,(1)))/VREF;%no offset
    O(:,1)=V(1,1);%Voltage offset from initial matrix V (at starting point)
    i=i+1;
end
G(:,1)=Vout(:,1)-V(:,1);%Ideal slope - Actual slope (before the offset correction)
i=1;
LSB=(V2(256,1)-V2(1,1))/256;
i=1;
while i<257
    INL(i,1)=((V2(i,1)-V2(1,1))/Vstep)-(i-1);
    i=i+1;
end
i=1;
figure
plot(Cd,INL)
Y=-1;
YY=1;
X=0;
XX=255;
axis([X XX Y YY]);
xticks([0 32 64 96 128 160 192 224 255]);
```

```

title('INL','FontSize',15,'FontWeight','bold','FontName','Times New
Roman');
set(gca,'YTick',[Y:0.1:YY]);
set(gca,'xticklabel',dec2bin(get(gca,'xtick'),8))
ylabel('LSB','FontSize',18,'FontWeight','bold','FontName','Times New
Roman');
xlabel('Digital Input
Code','FontSize',18,'FontWeight','bold','FontName','Times New Roman');
set(gca,'LineWidth',2,'FontSize',18,'FontWeight','Bold','FontName','Times
New Roman');
grid on;
%new INL
%Vos=V(1,1) %from chapter 5 testing converters, offset voltage
%Q=(V(256,1)-offset)/Vout(256,1) %from chapter 5 testing converters,gain
i=1;
F1=0;
F2=0;
F3=0;
F4=0;
while i<257
    F1=F1+Cd(i,1);%k
    F2=F2+Cd(i,1)^2;%k^2
    F3=F3+V(i,1);%U[k]
    F4=F4+V(i,1)*Cd(i,1);%kU[k]
    i=i+1;
end
Q=((F1*F3-256*F4)/(Vstep*(F1^2-256*F2)))
Vos=((Vstep*F1+256*Vout(1))*F4-(Vstep*F2+Vout(1)*F1)*F3)/(Vstep*((F1^2)-
256*F2))
i=1;
while i<257
    e(i,1)=V(i,1)-Vos-(Vstep*Cd(i,1)+Vout(1))*Q;
    INLnew(i,1)=e(i,1)/(256*Vos*Q);
    i=i+1;
end
figure
plot(Cd,INLnew/Vstep);
Y=-0.5;
YY=0.5;
X=0;
XX=255;
axis([X XX Y YY]);
xticks([0 32 64 96 128 160 192 224 255]);
title('INL','FontSize',15,'FontWeight','bold','FontName','Times New
Roman');
set(gca,'YTick',[Y:0.1:YY]);
set(gca,'xticklabel',dec2bin(get(gca,'xtick'),8))
ylabel('LSB','FontSize',18,'FontWeight','bold','FontName','Times New
Roman');
xlabel('Digital Input
Code','FontSize',18,'FontWeight','bold','FontName','Times New Roman');
set(gca,'LineWidth',2,'FontSize',18,'FontWeight','Bold','FontName','Times
New Roman');
grid on;
i=1;
b=256;
while b>1
    DNL(b,i)=(V2(b,i)-V2(b-1,i))/Vstep-1;
    b=b-1;
end
%DNL(1,1)=V2(2,1)-V2(1,1);

```

```

i=1;
%DNLp(:,1)=DNL(:,1)*VREF/LSB(1,1);
figure
%plot(Cd,((DNL*VREF/Vstep)-1))
plot(Cd,DNL)
Y=-0.3;
YY=0.3;
X=0;
XX=255;
axis([X XX Y YY]);
xticks([0 32 64 96 128 160 192 224 255]);
title('DNL','FontSize',15,'FontWeight','bold','FontName','Times New Roman');
set(gca,'YTick',[Y:0.05:YY]);
set(gca,'xticklabel',dec2bin(get(gca,'xtick'),8))
ylabel('LSB','FontSize',18,'FontWeight','bold','FontName','Times New Roman');
xlabel('Digital Input Code','FontSize',18,'FontWeight','bold','FontName','Times New Roman');
set(gca,'LineWidth',2,'FontSize',18,'FontWeight','Bold','FontName','Times New Roman');
grid on;
figure
X=1:200;
%offset
plot(X,0,'o')
Y=-0.1;
YY=0.1;
X=1;
XX=200;
axis([X XX Y YY]);
xticks([0 20 40 60 80 100 120 140 160 180 200]);
title('OFFSET VOLTAGE','FontSize',15,'FontWeight','bold','FontName','Times New Roman');
set(gca,'YTick',[Y:0.05:YY]);
set(gca,'XTick');
ylabel('VOLTAGE (V)','FontSize',18,'FontWeight','bold','FontName','Times New Roman');
xlabel('MC RUN','FontSize',18,'FontWeight','bold','FontName','Times New Roman');
set(gca,'LineWidth',2,'FontSize',18,'FontWeight','Bold','FontName','Times New Roman');
grid on;
%GAIN
figure
X=0:255;
plot(X,G)
Y=-0.02;
YY=0.02;
X=0;
XX=255;
axis([X XX Y YY]);
xticks([0 32 64 96 128 160 192 224 255]);
title('GAIN','FontSize',15,'FontWeight','bold','FontName','Times New Roman');
set(gca,'YTick',[Y:0.01:YY]);
set(gca,'xticklabel',dec2bin(get(gca,'xtick'),8))
ylabel('VOLTAGE (V)','FontSize',18,'FontWeight','bold','FontName','Times New Roman');
xlabel('Digital Input Code','FontSize',18,'FontWeight','bold','FontName','Times New Roman');

```

```

set(gca,'LineWidth',2,'FontSize',18,'FontWeight','Bold','FontName','Times
New Roman');
grid on;
figure
plot(Cd,Reconstruction)
hold on;
plot(Cd,Vout(:,1));
Y=0;
YY=1.8;
X=0;
XX=255;
axis([X XX Y YY]);
xticks([0 32 64 96 128 160 192 224 255]);
title('RECONSTRUCTION','FontSize',15,'FontWeight','bold','FontName','Times
New Roman');
set(gca,'YTick',[Y:0.2:YY]);
set(gca,'xticklabel',dec2bin(get(gca,'xtick'),8))
ylabel('OUTPUT VOLTAGE','FontSize',18,'FontWeight','bold','FontName','Times
New Roman');
xlabel('Digital Input
Code','FontSize',18,'FontWeight','bold','FontName','Times New Roman');
set(gca,'LineWidth',2,'FontSize',18,'FontWeight','Bold','FontName','Times
New Roman');
grid on;

```