

POLITECNICO DI TORINO

Master's Degree in Electronic Engineering



**A novel miniaturised CMOS architecture
for on-chip data and clock recovery in
Wireless RF ASK-demodulators**

Supervisors:

Prof. Danilo DEMARCHI

Prof. Sandro CARRARA

Eng. Gian Luca BARBRUNI

Candidate

Matilde CERBAI

Co-Supervisors:

Prof. Paolo MOTTO ROS

October 2022

Abstract

Wireless power and data transmission is an efficient method to develop miniaturised and highly distributed neural interfaces. Amplitude Shift Keying (ASK) modulation is commonly used in the case of ultra-miniaturised implants.

Over the years, several ASK-based Clock Data Recovery (CDR) architectures have been proposed with different objectives, such as: increasing the data rate, decreasing the silicon area and reducing the power consumption. The state-of-the-art includes: synchronous architectures mainly relied on complex Phase Locked Loop (PLL) based circuits or asynchronous systems with the need of on-chip oscillators. This master's thesis is a part of an ambitious project focused on reverting blindness, carried out in the Bio/CMOS Interfaces Group at Integrated Circuits Laboratory (EPFL, Neuchatel). The idea is to develop an innovative implantable cortical visual prosthesis based on ultra-miniaturised, wireless, individually addressable and free-floating CMOS implants for precise intracortical neurostimulation.

The present work aims to create a miniaturised and low-power architecture that guarantees both data and clock recovery starting from an ASK-modulated signal. The newly proposed CDR architecture consists mainly of an RF ASK demodulator and a clock recovery system, both implemented in Cadence using TSMC-180 *nm* CMOS technology.

The demodulator receives the wirelessly transmitted ASK-modulated signal at 433.92 *MHz* and generates the relative digital waveform. The latter is reconstructed for a data rate as high as 6 *Mbps* and a modulation index in the range of 9-30%.

Two CDR versions have been implemented. In the first one, a frequency multiplier receives a precise train of pulses from the demodulator output to generate the clock. In the second case, the clock is directly extracted from the demodulated digital signal, which is set with a higher frequency than the data. For both implementations, a control block is introduced to pull up the clock and save it in a memorisation structure. After clock memorisation, data are ready to be sampled synchronously. Both the demodulated clock and data are serialised; thus, a transmission protocol is defined to differentiate between them.

Simulations validate the functionality of the two entire architectures. The second

solution outperforms the first one both in terms of area and power consumption. The whole CDR architecture occupies $1500 \mu m^2$ and consumes $15 \mu W$ while operating with a clock data rate of $6 Mbps$. The newly proposed solution is, therefore, a valid alternative to the state-of-the-art, especially in RF applications.

In the future, the demodulator can be optimised to work at a higher data rate while consuming even less power. Area reduction is possible considering smaller technology nodes, such as $90 nm$.

Keywords: ASK Modulation, CMOS, Wireless RF Transmission, Clock Data Recovery, Low-Power, Low-Area, High Data Rate.

Sommario

La trasmissione di dati e di potenza wireless è un metodo efficiente per sviluppare interfacce neurali miniaturizzate e facilmente diffusibili. La modulazione Amplitude Shift Keying (ASK) è comunemente usata nel caso di impianti ultraminiaturizzati. Negli anni sono state proposte diverse architetture Clock Data Recovery (CDR) basate sulla modulazione ASK con diversi obiettivi, quali: l'incremento della velocità di trasmissione di dati, la diminuzione dell'area di silicio e la riduzione del consumo. Lo stato dell'arte attualmente comprende: complesse architetture sincrone basate principalmente su Phase Locked Loop (PLL) o sistemi asincroni che necessitano di oscillatori on-chip.

Questa tesi di Master è parte di un ambizioso progetto incentrato sul ripristino della cecità, svolto nel Bio/CMOS Interfaces Group presso l'Integrated Circuits Laboratory (EPFL, Neuchatel). L'idea è di sviluppare un'innovativa protesi visiva basata su impianti CMOS ultraminiaturizzati, wireless, per una neurostimolazione intracorticale precisa.

Il presente lavoro mira allo sviluppo di un'architettura miniaturizzata e a basso consumo che garantisca sia il recupero dei dati che del clock a partire da un segnale modulato ASK. La nuova architettura CDR proposta consiste principalmente in un demodulatore RF ASK e un sistema di recupero del clock, entrambi implementati in Cadence utilizzando la tecnologia CMOS TSMC-180 *nm*.

Il demodulatore riceve il segnale modulato ASK trasmesso in modalità wireless a 433,92 *MHz* e genera la relativa forma d'onda digitale. Quest'ultimo viene ricostruita per una velocità dati fino a 6 *Mbps* e un indice di modulazione nell'intervallo 9-30%.

Sono state implementate due versioni CDR. Nella prima, un moltiplicatore di frequenza riceve un preciso treno di impulsi dall'uscita del demodulatore per generare il clock. Nel secondo caso, il clock viene estratto direttamente dal segnale digitale demodulato, che viene impostato con una frequenza superiore ai dati. Per entrambe le implementazioni viene introdotto un blocco di controllo per ricavarsi il clock e salvarlo in una struttura di memorizzazione. Dopo la memorizzazione del clock, i dati sono pronti per essere campionati in modo sincrone. Sia il clock demodulato che i dati sono serializzati; per questo, è stato definito un protocollo di trasmissione

per differenziarli.

Le simulazioni convalidano la funzionalità di entrambe le architetture. La seconda soluzione risulta migliore della prima sia in termini di area che di consumo energetico. L'intera architettura CDR occupa $1500 \mu m^2$ e consuma $15 \mu W$ mentre opera con una velocità dati di clock di $6 Mbps$. La nuova soluzione proposta è, quindi, una valida alternativa allo stato dell'arte, soprattutto nelle applicazioni RF. In futuro, il demodulatore può essere ottimizzato per funzionare a una velocità dati più alta ed con un consumo più basso. La riduzione dell'area è possibile considerando nodi tecnologici più piccoli, come $90 nm$.

Table of Contents

List of Tables	IX
List of Figures	X
Acronyms	XIV
1 Introduction	1
1.1 Project aim	1
1.2 Design specifications	3
1.3 Outline of the project	4
2 State-of-the-art	5
2.1 ASK Demodulators	5
2.2 Clock recovery system	14
2.3 Frequency multiplier	18
2.4 Oscillator	21
3 ASK Demodulator	26
3.1 Design implementation	26
3.1.1 Rectifier	28
3.1.2 Digital shaper	29
3.1.3 Buffers	30
3.2 Performance analysis	34
3.3 Power&area analysis	38
3.4 Noise analysis	41
4 Clock recovery system	42
4.1 Design implementation	42
4.2 Switch	43
4.3 Frequency multiplier	44
4.4 Data/Clock Control	46

4.4.1	Transmission protocol	46
4.5	Clock memorization and data sampling	49
4.5.1	Oscillator	52
4.6	Alternative solution	53
4.6.1	Transmission protocol	53
5	Complete proposed architecture	61
5.1	Pre-layout simulations	61
5.2	Comparison between two proposed architecture	62
5.2.1	Power and Area analysis	62
5.3	Monte-Carlo analysis	65
5.4	Phase variation robustness	67
5.5	Bit Error Rate(BER) analysis	70
5.6	Layout	73
5.7	Post-layout simulations	75
5.8	Comparison with previous works	79
6	Conclusion and Future developments	80
6.1	Future Developments	81
	Bibliography	82
A	Appendix	88
A.1	ASK modulated signal generation	88
A.2	Adapted duty cycle ratio variation with respect to DR and MI	89
A.3	Phase robustness test	91
A.4	BER test	93

List of Tables

2.1	Comparison solution literature	13
2.2	Clock regimes [24]	14
2.3	Comparison between solutions of literature	25
3.1	Amplitude levels variation with MI	34
3.2	Comparison with the paperwork [17]	38
4.1	Truth table SR register	55
4.2	Comparison between the implemented oscillator	60
5.1	Power (a) without and (b) with frequency multiplier in pre-layout	63
5.2	Area (a) without and (b) with frequency multiplier in pre-layout	64
5.3	Monte Carlo results of proposed oscillators	67
5.5	Post-layout (a) power and (b) area analysis without frequency multiplier	77
5.6	Comparison with other works	79

List of Figures

1.1	CMOS Current Driver. Reprinted from [5]	2
1.2	ASK modulated and demodulated signal	3
2.1	Voltage-mode demodulator. Reprinted from [7]	6
2.2	Current-mode demodulator. Reprinted from [8]	6
2.3	Adaptive bias approach oscillator. Reprinted from [13]	7
2.4	Demodulator circuit. Reprinted from [16]	7
2.5	Simplified demodulator circuit. Reprinted from [16]	8
2.6	Block scheme. Reprinted from [17]	8
2.7	Demodulator circuit. Reprinted from [17]	9
2.8	Event-based demodulator and signals. Reprinted from [19]	10
2.9	Demodulator circuit. Reprinted from [21]	11
2.10	Signals. Reprinted from [21]	11
2.11	Self-sampling demodulator. Reprinted from [23]	12
2.12	Phase-Locked Loop. Reprinted from [25]	15
2.13	Delay-Locked Loop. Reprinted from [25]	15
2.14	Mesochronous solution. Reprinted from [25]	15
2.15	Dual-rail transmitter. Reprinted from [25]	16
2.16	Dual-rail receiver. Reprinted from [25]	17
2.17	C-Muller circuit. Reprinted from [25]	17
2.18	PLL as frequency multiplier block diagram. Reprinted from web	18
2.19	DLL as frequency multiplier block diagram. Reprinted from [28]	19
2.20	Eight-phase delay line. Reprinted from [28]	19
2.21	Edge combiner. Reprinted from [28]	20
2.22	(a) Mixer architecture and (b) Frequency multiplier mixer-based. Reprinted from [29]	20
2.23	Ring oscillator. Reprinted from [30]	21
2.24	Current starved oscillator. Reprinted from [30]	22
2.25	(a) 3 and (b) 5 stages CS RO. Reprinted from [30]	22
2.26	CS oscillator with R_s poly. Reprinted from [35]	23
2.27	Stack sleep delay cell. Reprinted from [36]	23

2.28	CS Sleep VCO. Reprinted from[36]	24
2.29	CS RO. Reprinted from[37]	24
3.1	ASK Demodulator. Reprinted from [17]	26
3.2	Adapted version of Implementation 1	27
3.3	ASK modulated signal with $f_c= 433.92 MHz$ and MI=15%	28
3.4	Rectifier schematic	29
3.5	Rectifier output	29
3.6	Digital shaper output	30
3.7	Buffer circuit	32
3.8	Output of the first buffer	33
3.9	Output of the second buffer	33
3.10	Demodulator layout	34
3.11	Variation of $\frac{V_{H_{avg}}}{V_{L_{avg}}}$ according to MI	35
3.12	Variation of $\frac{V_{H_{avg}}}{V_{L_{avg}}}$ according to MI	36
3.13	Random signal modulated@433.92 MHz	37
3.14	Random signal demodulated@433.92 MHz	37
4.1	First implementation	42
4.2	Frequency multiplier@12 MHz	44
4.3	Modular block frequency multiplier	45
4.4	XOR (a) NAND- and (b) NOR-based. Reprinted from web	45
4.5	Frequency multiplier waves	46
4.6	Transmission protocol	46
4.7	Switch control signal	47
4.8	First control block implementation	47
4.9	Second control block implementation	48
4.10	Third control block implementation	48
4.11	Control signals	49
4.12	Clock memorization block	49
4.13	Classical implementation of 2x1 multiplexer. Reprinted from web	50
4.14	Low area MUX2x1 solutions. Reprinted from [38]	50
4.15	Transmission gate D-FF. Reprinted from web	51
4.16	Memorisation block layout	51
4.17	On-chip oscillator circuit	52
4.18	Alternative transmission protocol	53
4.19	Clock Data Recovery architecture	54
4.20	Control block CDR	54
4.21	SET-RESET register (a) structure (b) symbol. Reprinted from web	55
4.22	Power on reset circuit. Reprinted from [39]	56
4.23	Control block layout	57

4.24	POR layout	57
4.25	Ring oscillator with capacitors	58
4.26	3 stages CS VCO	58
4.27	5 stages CS VCO	59
4.28	5 stages Starved Sleep VCO	59
4.29	Oscillator circuit power on	60
4.30	On-chip oscillator layout	60
5.1	Frequency multiplier solution	61
5.2	w/o Frequency multiplier solution	62
5.3	Pre-layout power (a) without and (b) with frequency multiplier . . .	64
5.4	Pre-layout area (a) without and (b) with frequency multiplier . . .	64
5.5	Monte Carlo first oscillator	65
5.6	Monte Carlo small dimension oscillator	66
5.7	Monte Carlo big dimension oscillator	66
5.8	Clock memorization block signals	68
5.9	Parameters settled by the user	69
5.10	Phase robustness analysis with $\Delta = 16\%$	69
5.11	Simulation with $\Delta = 16\%$	70
5.12	BER logic scheme	71
5.13	Majority voter. Reprinted from web	71
5.14	Boolean function and truth table of the Majority Voter	72
5.15	Zoom of a BER test with 1000 bits	72
5.16	Standard cell layout. Reprinted from web	74
5.17	Layout of the CDR architecture	74
5.18	Demodulator pre- vs post-layout simulations	75
5.19	CDR architecture post-layout simulations	75
5.20	Oscillator pre- vs post-layout simulations	76
5.21	Correct CDR architecture post-layout simulation	77
5.22	Post-layout (a) power and (b) area analysis without frequency multiplier	78

Acronyms

ASK

Amplitude Shift Keying

CDR

Clock Data Recovery

PLL

Phase Locked Loop

EPU

External Processing Unit

SMND

Smart Micro Neural Dust

LDO

Low Drop Out

SAR

Specific Absorption Rate

DR

Data Rate

MI

Modulation Index

PPM

Pulse Width Modulation

PPM

Pulse Position Modulation

C2MOSDFF

CMOS D-Flip Flop

FOM

Figure Of Merit

DLL

Delay Locked Loop

LPF

Low Pass Filter

VCO

Voltage Controlled Oscillator

RO

Ring Oscillator

DVS

Dynamic Voltage Scaling

DFS

Dynamic Frequency Scaling

CS

Current Starved

SNR

Signal to Noise Ratio

VCDL

Voltage Controlled Delay Line

DFF

D-Flip Flop

SR

Set-Reset

TG

Transmission Gate

POR

Power-On Reset

BER

Bit Error Rate

MV

Majority Voter

DRC

Design Rule Check

LVS

Layout Versus Schematic

Chapter 1

Introduction

1.1 Project aim

Nowadays, visual prostheses can stimulate the patient's visual system, completely bypassing the components that do not work. Various approaches are exploited and studied: the retinal, subretinal, and suprachoroidal prostheses act straight on the different eye parts. This type of implantation is effective for patients affected by genetic pathologies such as *Retinis pigmentosa*, due to which the eye receptors start to degenerate, or *Macular degeneration*, mainly an age-related disease that leads to sight loss of the field of view central area. However, it does not take into account people born blind. A possible solution can be the *Optical Nerve Stimulation* or the **Visual Cortex Stimulation**. The latter allows for treating patients that do not develop a disease related to old age or tissue degeneration, especially the ones born blind. This solution generally includes a complex system based on two parts:

- **External module:** a video camera is mounted on glasses worn by the patient to catch information and images from outside. The image processing block does the post-processing and sends the needed information wirelessly to the implanted block through a transmitting coil;
- **Internal module:** the receiving coil transmits power and data directly to the biocompatible embedded processor that made the stimulation back end block active to generate the impulses to be sent to the μ -electrode array. Each μ -electrode provides a direct and localized stimulation: the higher the electrode number, the higher the resolution.

The main idea of this project is linked to the Cortical approach, trying to fix all its related weaknesses. The real challenge of this work is to create a wireless array of freestanding electrodes through which direct stimulation of the brain's occipital lobe could be possible, without any need for wires, to reduce infection risks. The

most crucial objective is related to the internal part. It is strictly linked to the final ambitious goal to go closer to the concept of Body and Neural Dust [1], [2]: the implanted array and each small chip should be Ultra-Low-Power and Ultra-Low-Size.

Firstly, the External Processing Unit EPU [3][4] detects information through a real-time image processing, and an RF Amplitude Shift Keying (ASK) modulated signal is generated. The choice has fallen on this modulation type to transmit wireless power and data through an inductively coupling transmitting coil. The receiving coil catches the ASK signal and gives it to each Smart Micro Neural Dust (SMND) that composes the array. Each of those receivers embedded in the array is composed of different blocks: the power supply is generated by a Low Drop Out (LDO) regulator that feeds the overall circuit. The ASK modulated signal is rectified and put as input to an ASK demodulator, through which the digital waveform is obtained. Logic and memory are introduced to manage the demodulated bit-stream: both are provided to a CMOS current driver made up of a voltage-to-current converter and an H-bridge circuit that jointly provide biphasic current stimulation [5].

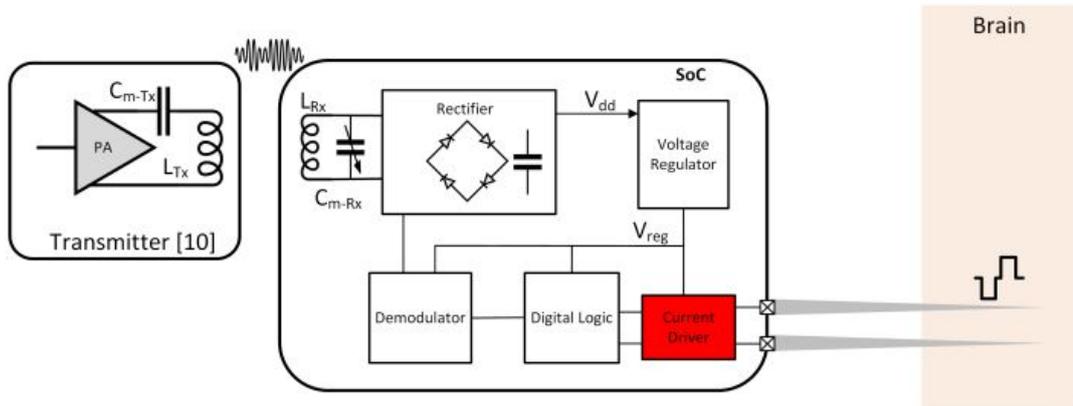


Figure 1.1: CMOS Current Driver. Reprinted from [5]

One of the most critical parts is the introduction of the implantable device inside the brain. In fact, a biocompatible and biotoxicity CMOS process should be followed to integrate the microelectrodes with the chip and guarantee robustness and mechanical properties to allow the insertion to be long-lasting and effective.

1.2 Design specifications

The thesis goal is focused on the receiver: the realization of a Demodulation system able to recover data and the clock for sampling. The system should be implemented in Cadence using TSMC-180 nm CMOS technology.

The power consumption and the area size are the critical aspects of this work: the chip should produce the power necessary for the neurostimulation without exceeding any Specific Absorption Rate (SAR) and heat dissipation limits to not generate injuries related to brain burning.

The demodulator needs the power to do the conversion from analog to a digital signal, especially considering the high value of the carrier frequency, equal to 433.92 MHz . In Figure 1.2, a generic ASK modulated signal has been reported (in blue) with superimposed the correspondent digital demodulated signal. As already mentioned, the ASK allows for transferring both power and data.

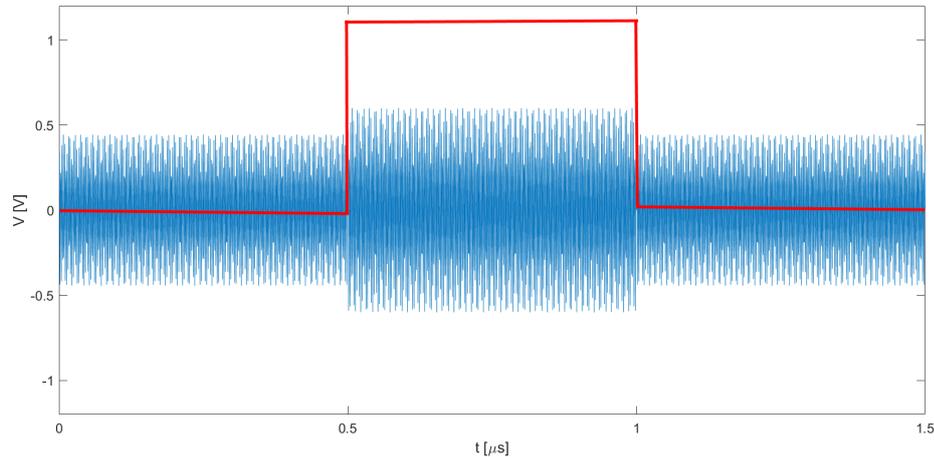


Figure 1.2: ASK modulated and demodulated signal

In summary, the project's specifications are the following:

- **High Data Rate** (DR): the ASK modulated signal received from the transmitter has a high data rate. Therefore, the demodulator should be able to work with the input frequencies, recalling this relation

$$DR = 2 \cdot f \quad (1.1)$$

where f is the signal frequency;

- **Adaptable Modulation Index** (MI): if the amplitudes of the ASK levels change, the demodulator should still work correctly. The high and low levels of the ASK modulated signal are linked to the MI according to the following equation;

$$MI_{\%} = \frac{V_{pp}^{max} - V_{pp}^{min}}{V_{pp}^{max} + V_{pp}^{min}} \times 100 \quad (1.2)$$

- **Low-Power and Low-Area** design: the final chip is a block of a 200 x 200 x 30 μm^3 CMOS freestanding μ -electrode array that should be implanted in the patient brain; thus its design should be miniaturised and low-power consuming;
- **Clock recovery**: a system able to correctly sample the demodulated data should be jointly designed with the demodulator.

1.3 Outline of the project

The main objective of this master's thesis is the design realization of a Clock and Data Recovery (CDR) architecture for ASK modulated signal. For this reason, the project has been organized as follows:

- *Chapter 2*: the state-of-the-art of all structure designed to compose the final architecture is deeply described;
- *Chapter 3*: the ASK demodulator design is presented;
- *Chapter 4*: the two proposed alternatives for the CDR design are introduced;
- *Chapter 5*: the final structure performances are analyzed. Tests are provided to verify the results and correctness of the simulations. The layout is done, and post-layout simulations are reported;
- *Chapter 6* presents the conclusions and future developments.

Chapter 2

State-of-the-art

2.1 ASK Demodulators

The demodulator is a circuit able to get its digital correspondent waveform from an analog modulated signal. In the case under analysis, the modulation type, as already pointed out, is the ASK. The classical demodulator comprises a rectifier, an envelope detector, and a comparator.

In [6] the demodulators have been classified into three different macro groups:

- **Voltage-mode:** the information is reported by the nodal voltages of the demodulator. The voltage-mode *envelope detector* extracts the low-frequency envelope of its RF input. The voltage-mode *average detector* generates the average of the signal obtained by the envelope detector. In the end, the *voltage comparator* uses the average voltage as a reference with respect to the envelope;
- **Current-mode:** the information is processed by the demodulator's branch currents. These demodulators work as the voltage-mode ones, with the only difference being that each block extracts and evaluates current instead of voltage. They are mainly used when a low-supply voltage is provided.
- **Mixed-mode:** the information relies on both the circuit's currents and voltages, taking advantage of the two modes. Generally, the first stage is a *transductor* through which the input voltage is converted into a current. The following blocks are the ones typical of a current-mode demodulator.

In literature, most demodulators are voltage-mode, as the one referred to [7] and reported in Figure 2.1. The voltage multiplier acts as a rectifier, while the envelope detector converts the signal envelope to a stream-bit. Then, the filter selects the wanted RF signal frequencies, while the average detector slows down the V_p . In the end, the two voltage values are compared through the comparator.

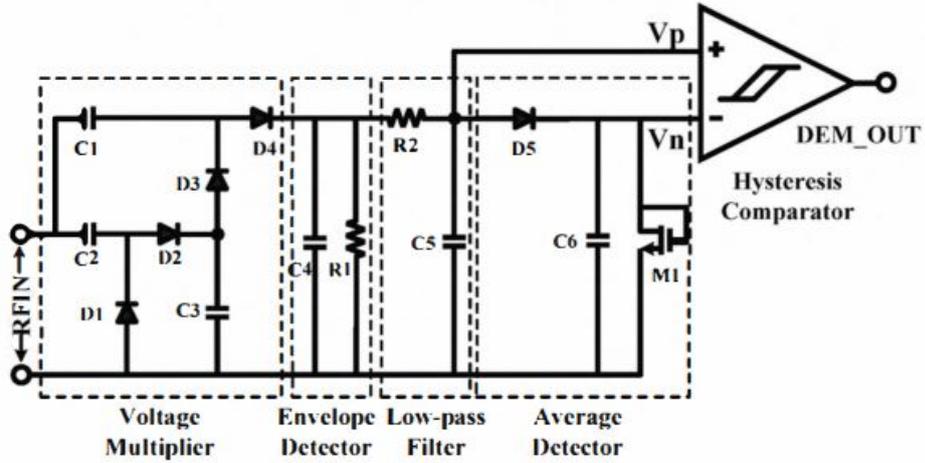


Figure 2.1: Voltage-mode demodulator. Reprinted from [7]

On the other side, the current mode is widely explored too [8], a possible structure is reported in Figure 2.2. The rectifier provides the non-negative signal, and the envelope detector works as a low pass filter: the signal and its phase-inverter copy are provided as inputs to the current-mode Schmitt trigger to bring back the input to the baseband. The absence of an averaging circuit reduces the occupied area, even if excellent results can also be obtained when this block is present, as reported in [9]. In this paper, a diode-capacitor network is used as a fast averaging circuit, which output sets a switching-mode current amplifier working in the current domain to recover, in the end, the digital signal through a decision circuit.

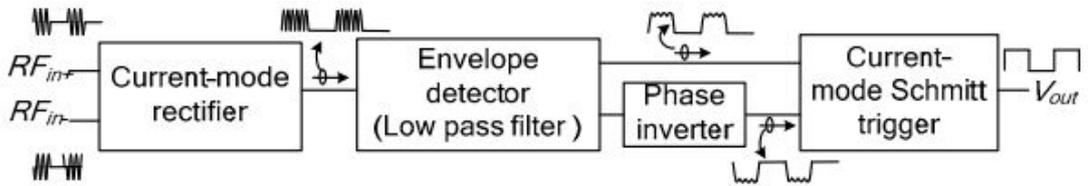


Figure 2.2: Current-mode demodulator. Reprinted from [8]

Moreover, by changing the mode and considering a structure with both differential input and output, as in [10] and in [11], a possible solution with a low index, as the two previous ones, can also be found in [12]. The structure is complex, even if it still has a relatively low-power consumption.

In [13], the low modulation index is obtained with the addition of an adaptive bias approach through the following structure:

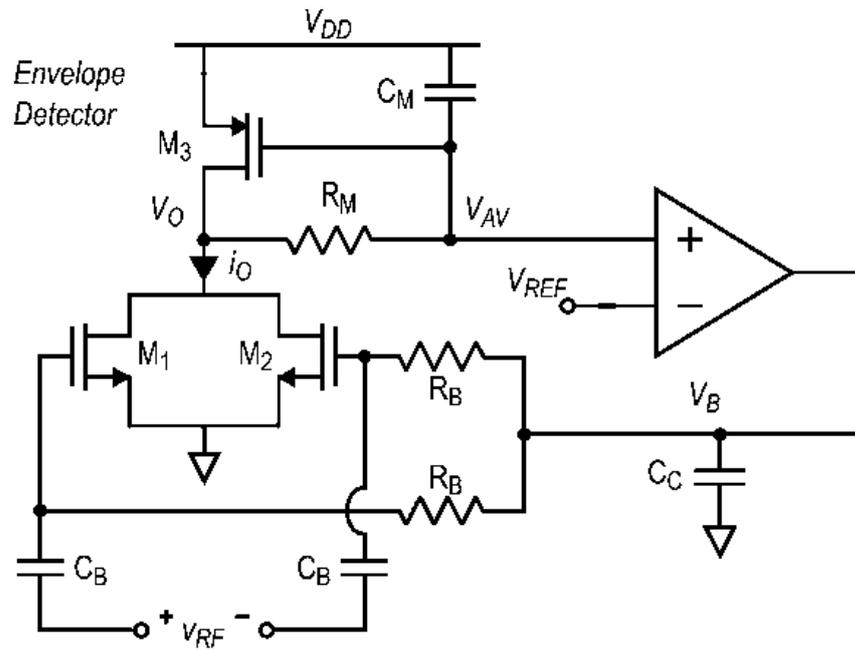


Figure 2.3: Adaptive bias approach oscillator. Reprinted from [13]

The feedback loop guarantees a small variation of current on the positive input of the amplifier, keeping low the power consumption, and at the same time, it increases the demodulator sensitivity.

Moreover, in literature, many area-saving solutions can be found, as C-less architectures [14], [15] and [16]; the latter structure is reported below

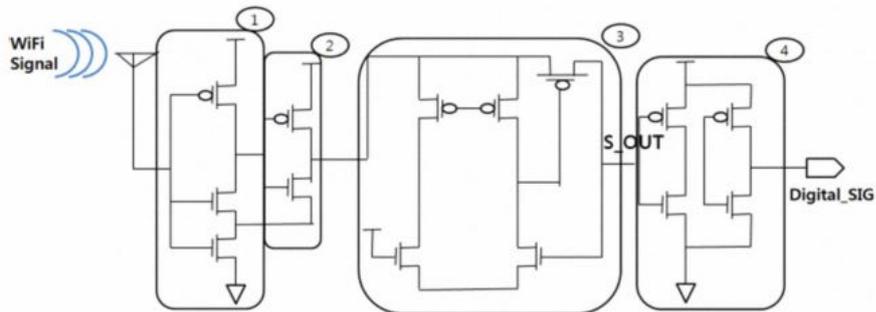


Figure 2.4: Demodulator circuit. Reprinted from [16]

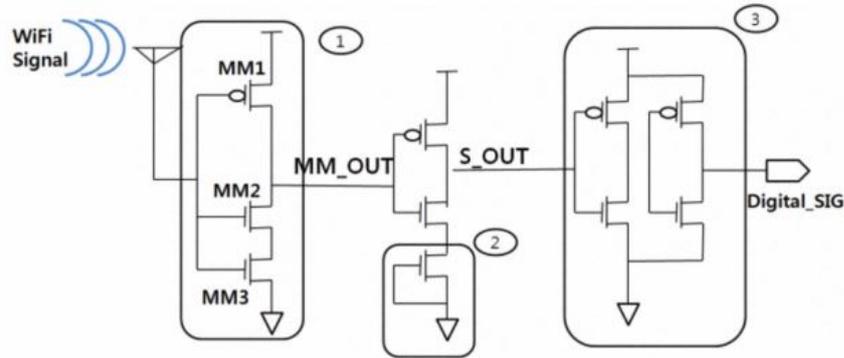


Figure 2.5: Simplified demodulator circuit. Reprinted from [16]

The circuit in Figure 2.4 shows the first implementation: the first and the second blocks generate a low pass filter, the third one is an envelope detector, and the last one is a buffer stage. In the simplified circuit version in Figure 2.5, the first three circuits are merged, followed by a configuration of three transistors to increase signal stability. In terms of area-saving, a voltage-mode special implementation is proposed in [17], and it is pretty similar to [18].

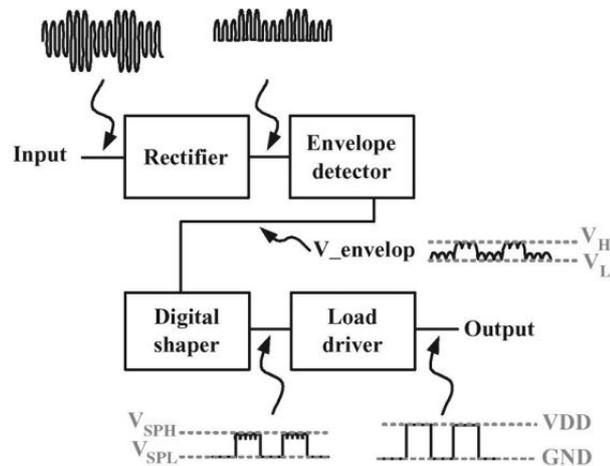


Figure 2.6: Block scheme. Reprinted from [17]

The block diagram above highlights the structure reported in [17]: the ASK input is rectified, and after getting the envelope detection, the digital signal is extracted by the digital shaper, and then it is made sharper through the load driver block. As can be seen in Figure 2.7, everything is obtained, with only fifteen transistors: in fact, no R-C elements have been implied. Moreover, there is no need for reference

voltage because of the self-sampled loop mechanism. This solution looks simple and can reach a high data rate, with the downside of not having a clock recovery system.

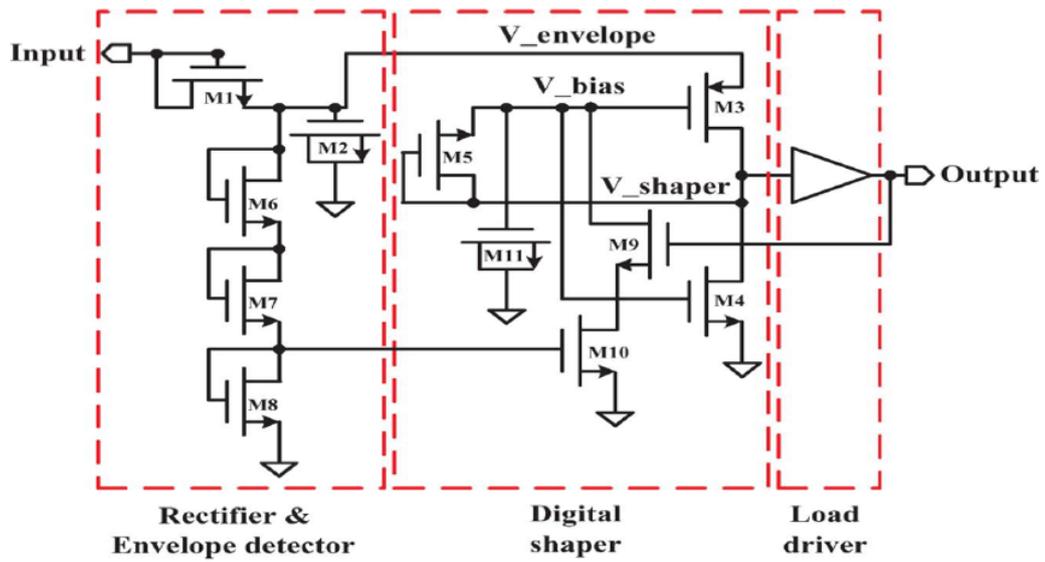


Figure 2.7: Demodulator circuit. Reprinted from [17]

Taking into account the final project aim, some solutions with an embedded clock recovery system have been considered too. A very unconventional digital architecture is proposed in [19], also quite similar to [20], applied to a cortical stimulator.

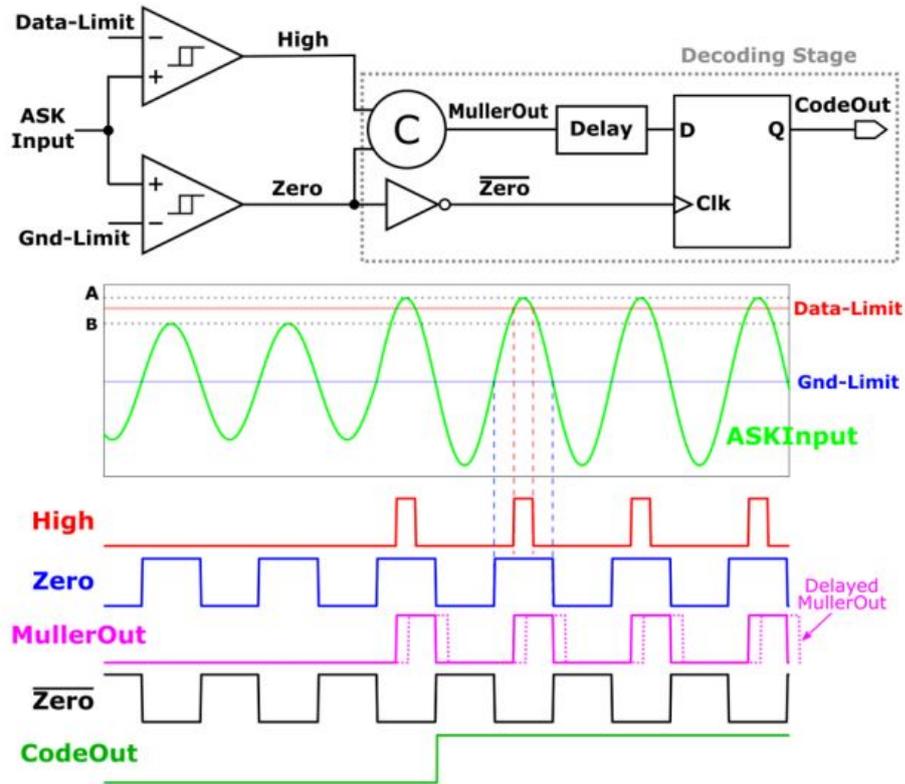


Figure 2.8: Event-based demodulator and signals. Reprinted from [19]

Two levels have been used: the *Data-Limit* between the two ASK modulated amplitudes and the *Gnd-Limit*, the system ground reference. According to these two levels, two hysteresis comparators have been used: they detect *High* when the ASK input overcomes the *Data-Limit*, while the *Zero* is a low-value every time the ASK signal goes below the *Gnd-Limit*. Afterward, synchronization between the two comparators' output is performed through a *Muller gate*, which output is high only when both *Zero* and *High* are valid; it maintains this level until the falling edge of the *Zero* signal occurs. The register stores the signal thanks to a clock generated from $\overline{\text{Zero}}$, giving the correct value as output. This solution is quite interesting even if the presence of a delay and two hysteresis comparators is quite heavy from the area point of view.

In [21], a circuit based on ASK and Pulse Position Modulation (PPM) is exploited: from an ASK-PPM signal, a digital ASK signal is extracted, employing an ASK detector, as a stream of pulses with a different duty cycle. This signal is used to toggle and control a system based on a charge pump, clock generator, and

a latched comparator to recover the data. The main trouble with this solution is that it does not provide a 50% duty cycle.

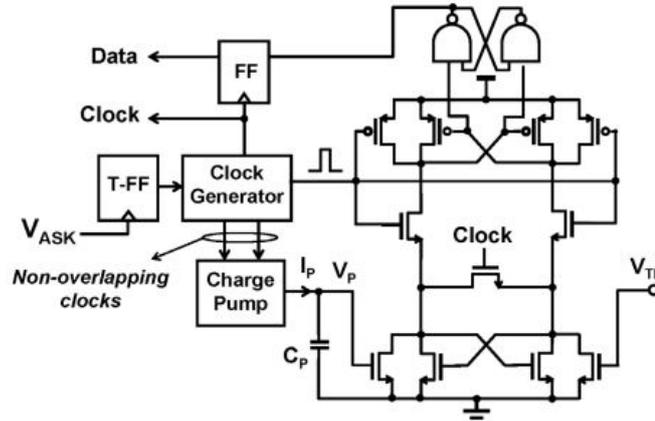


Figure 2.9: Demodulator circuit. Reprinted from [21]

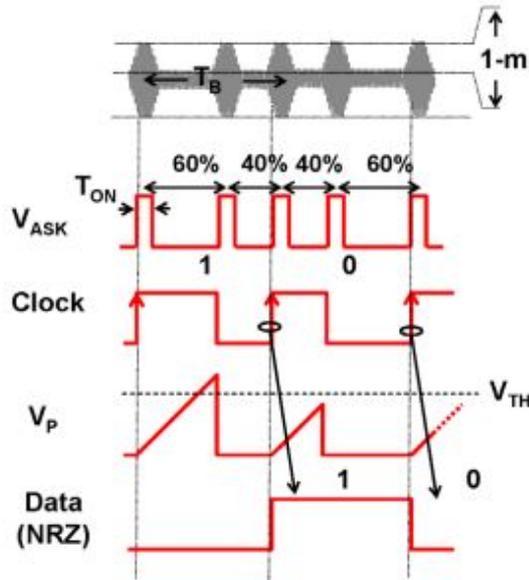


Figure 2.10: Signals. Reprinted from [21]

Others solutions with self-biased clock extractor can be found in [22] and [23]. The last one is fascinating due to its self-sampling scheme:

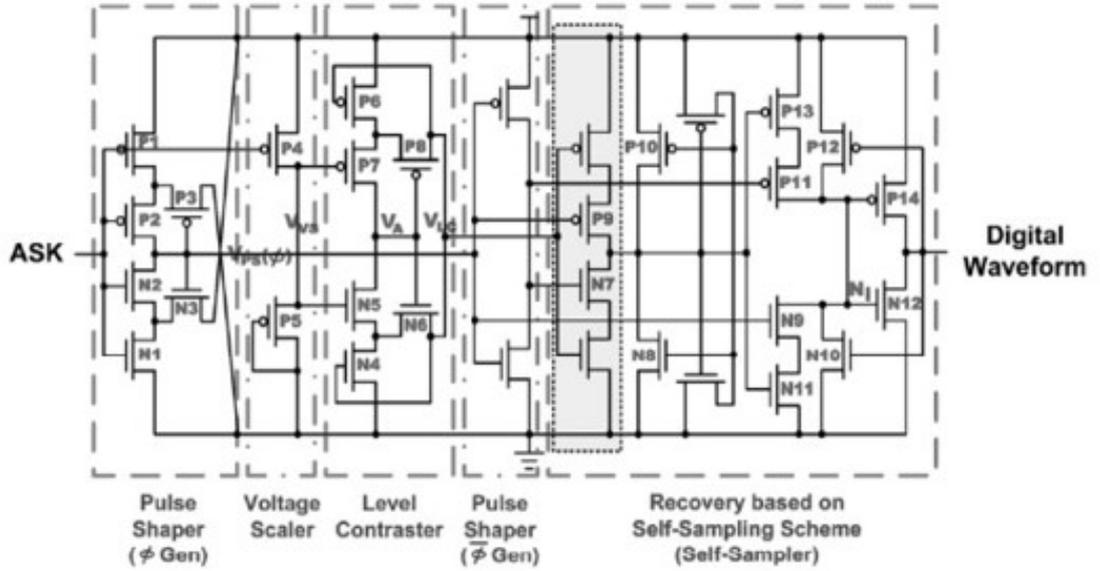


Figure 2.11: Self-sampling demodulator. Reprinted from [23]

The *pulse shaper* generates the clock and its inverted version directly from the ASK signal, a quite high-consuming operation. Then, the *voltage scaler* shifts the input to a minimum level and performs the peak difference, with respect to the ground, between the two amplitudes of the modulated signal. The *level contraster* acts as a comparator to get the required signal, sampled by the final stage, the *self-sampler*, built from CMOS D-flip-flop (C2MOSDFF).

This architecture is R-C less, the reason why the area-saving is consistent. It also shows optimal performances with very low MI and can reach a high data rate, embedding a data recovery system. The main disadvantages are the circuit complexity, the high dependence on the input characteristics, and the power consumption.

To better view all described implementations. The main features have been highlighted better to choose the solution most compliant with the project specifications. Moreover, a new Figure of Merit (FOM) has been introduced to underline which characteristics among the ones reported have been considered more noteworthy.

$$FOM = \frac{DataRate}{Power \cdot Area} \left[\frac{kbit/s}{mW \mu m^2} \right] \quad (2.1)$$

This FOM [17] takes into account mainly the power and the area considering that the architecture should be both low-power and low-area.

Year,Ref	f_c [MHz]	Technology node[μm]	Area [μm^2]	DR Mbps	P_{diss} [μW]	MI [%]	Clock recovery	FOM [$\frac{kbit/s}{mW\mu m^2}$]
2004 [14]	2	0.35	91 x 140	0.01	10234	-	-	7.85×10^{-6}
2006 [21]	1	0.6	-	-	-	100	yes	-
2008 [23]	2	0.18	32.3 x 14.5	1	336	5.5-50	yes	6.51
2008 [15]	2	0.35	3025	0.250	1010	27	-	0.084
2010 [16]	900	-	-	-	7.065	-	-	-
2010 [17]	13.56	0.35	101 x 32	1.2	306	11.11	-	1.24
2011 [7]	915	0.18	150 x 150	0.04	-	100	-	-
2012 [8]	100	0.13	-	2	55	-	-	-
2012 [22]	13.56	0.35	3554.2	1.356	274	70	yes	2.1
2014 [11]	13.56	ASK	4700	-	18	< 4	yes	-
2015 [13]	915	0.18	6000	2	< 4.13	> 5	-	8.26
2015 [9]	13.56	0.18	3000	2	35	> 7	-	19.5
2016[18]	5	0.18	920	0.5	17	5	-	31.96
2017 [12]	13.56	0.18	5780	1	30	2.33-100	-	5.91
2019 [19]	13.56	-	-	-	-	3-30	yes	-
2020 [10]	90	0.065	-	0.05	0.075(@0.2V)	1	-	-

Table 2.1: Comparison solution literature

The table compares the different solutions found in the literature, taking into account the carrier frequency with which the ASK signal is modulated.

Moreover, the area, and the power are the design's crucial points, considering that the wanted data rate is relatively high too.

The implementation should also work correctly when a slight variation of the two modulated amplitude levels occurs. It should be able to correctly recover the data through a clock recovery to sample the demodulated data.

The technology node of the design has to be considered because it directly influences the area dimension and power consumption.

2.2 Clock recovery system

Once the signal has been demodulated, a clock is needed to sample the transmitted data correctly. This task should be managed in parallel or before the data demodulation or embedded in the demodulation block. There are different ways to proceed:

- **Encoding techniques:** the signal necessary to sample the data can be directly "hidden" inside the transmitted data and recovered on the receiving part through some decoding blocks. An example is the *Manchester encoding*: through a serial transmission, the clock is drowned in the bit; thus, by detecting the central point of the bit, the clock is recovered. The drawback is related to the frame rate halving.
Moreover, the clock recovery can be performed in different ways: the modulation techniques, for example, can lead to catching up in parallel while the signal is demodulating. This way of proceeding has already been mentioned in Section 2.1;
- **Synchronization between on-chip clock and circuit clock:** in this case, the two clocks can have different characteristics and can be grouped in different synchronization regimes[24] :

Regime	Frequency (f)	Phase ϕ
Synchronous	Same	$\neq 0$
Mesochronous	Same	$\neq 0$
Plesiochronous	Different (same nominal f)	$\neq 0$
Asynchronous	Different	$\neq 0$

Table 2.2: Clock regimes [24]

In the **synchronous** case, the information transmission is handled by sending a control signal to get when the data is valid.

A **pseudo-synchronous** scenario can be taken into account, too: the two frequencies are different but are related by a multiple. The docking between the two clocks can be done through a Phase Locked Loop (PLL)(Figure 2.12) or a Delay Locked Loop(DLL)(Figure 2.13). The first is used when the coupling is done between an off-chip and on-chip clock, while the second is used when everything happens on the integrated circuit.

Phase-Locked Loop (VCO-Based)

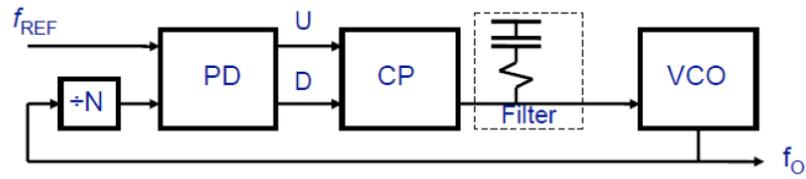


Figure 2.12: Phase-Locked Loop. Reprinted from [25]

Delay-Locked Loop (Delay Line Based)

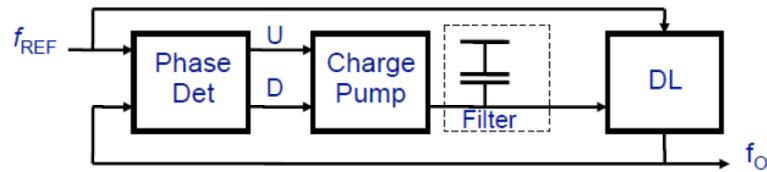


Figure 2.13: Delay-Locked Loop. Reprinted from [25]

In the **mesochronous** situation, a preamble is sent before the data transmission, and then the data at a frequency f can be oversampled at $2^f \cdot f$. The preamble can be implied, as below, to adjust an analogic delay line to re-phase the clock and avoid metastability problems. The efficiency loss is related to the preamble that should be as long as needed to allow the system to find the correct delay, with the advantage of having both the transmitter and receiver have the same working frequency.

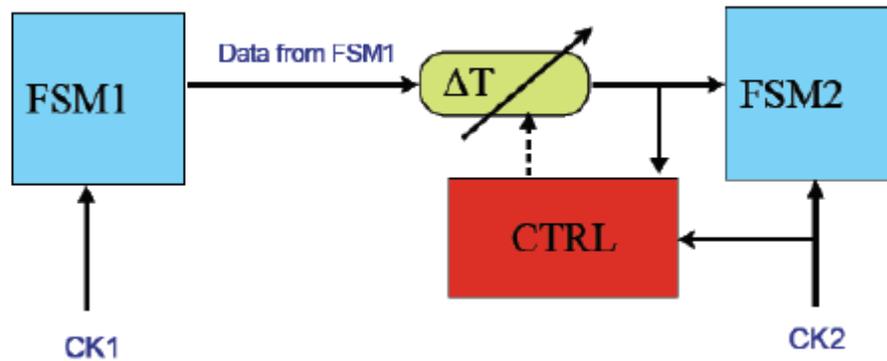


Figure 2.14: Mesiochronous solution. Reprinted from [25]

The **plesiochronous** case can be managed almost in the same way as the previous

one with the only difference that is having a small change in the frequency, the transmission should be shorter than the de-phasing due to which the synchronization could be lost: more synchronization cycles will be needed.

When the two clocks are completely freed, the **asynchronous** case is exploited: a synchronization between them should be provided. The *Handshake Synchronizer Machine* can be a good solution. It is needed when the transmitter and receiver exchange data and control signals to sample them.

This transmission protocol is defined *Dual-rail* [26] because to send n data, $2n$ bits are needed: the data are coded on two bits. The transmitter reported below should be able to discriminate using the introduced protocol between the different cases: **00** the data is not sent, **11** is not used. In contrast, **01** and **10** are codes for which the data is VALID and correspond to 0 and 1 respectively.

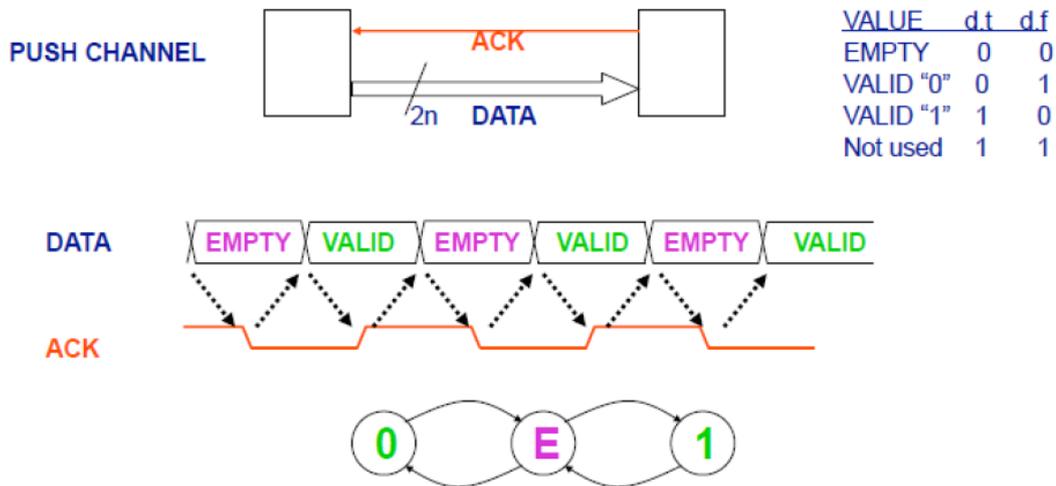


Figure 2.15: Dual-rail transmitter. Reprinted from[25]

According to this, the ACK control signal knows that it can receive another data when it sees 00; otherwise, the data transmission is already ongoing. In this solution, the request signal arrives with the data; thus, it is unnecessary to waste time recognizing it.

The receiver in Figure 2.16 detects when new data is sent. The NAND gates determine when data is VALID, checking that one of the two lines is 1; then the OR operation is performed to confirm further that the data will be valid only if all the lines are one. At least, the C-MULLER block defines that only when all the lines' couple are all VALID is the data VALID.

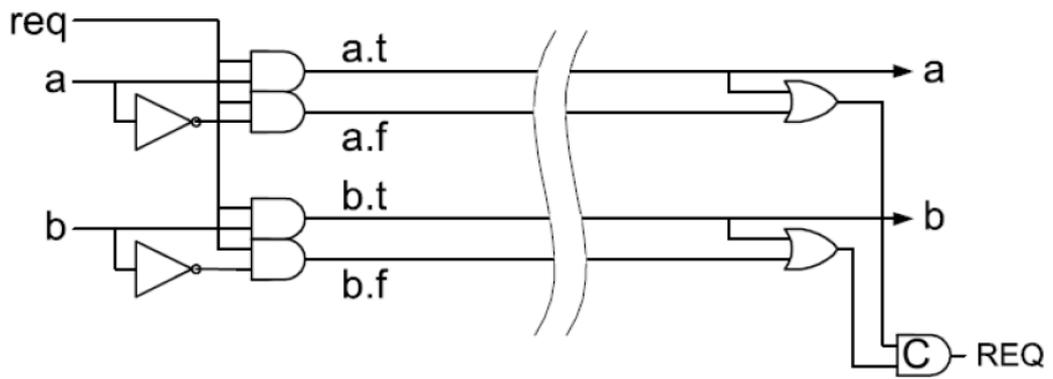
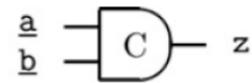
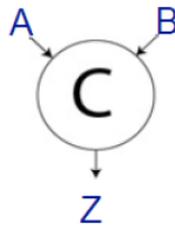


Figure 2.16: Dual-rail receiver. Reprinted from[25]

Production Rules

$$\begin{cases} a * b \Rightarrow z \uparrow \\ \bar{a} * \bar{b} \Rightarrow z \downarrow \end{cases}$$



Specification

$$z = a * b + z * (a + b)$$

a	b	z
0	0	0
0	1	no change
1	0	no change
1	1	1

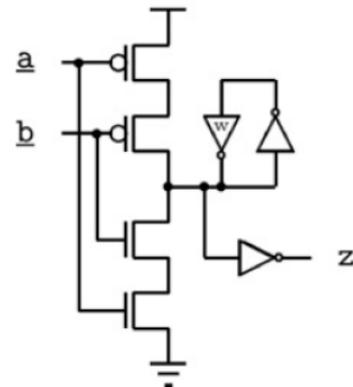


Figure 2.17: C-Muller circuit. Reprinted from[25]

2.3 Frequency multiplier

As it is widely known, there are different ways to implement a frequency multiplier:

- **Phase Locked Loop (PLL)**[27]: to generate a stable RF frequency signal, this solution is the most adopted in the literature. The PLL is a control system that generates a signal whose frequency is related to the one of the input.

The output signal frequency is the input one multiplied by a constant. By simply changing this constant, the oscillating frequency can be modified. The classical structure of a Frequency multiplier PLL-based is reported below:

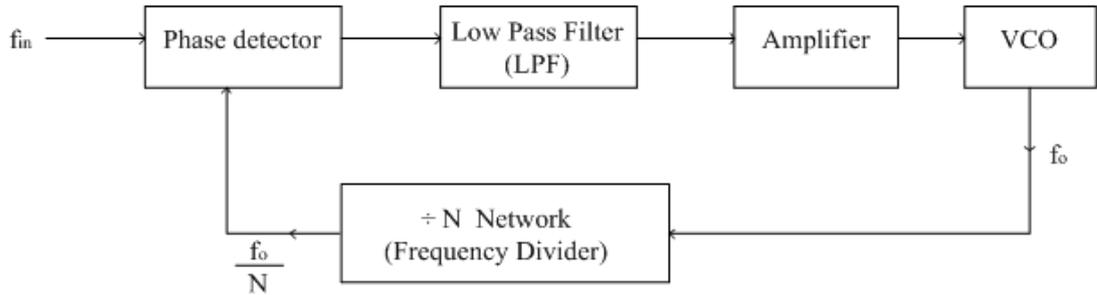


Figure 2.18: PLL as frequency multiplier block diagram. Reprinted from web

The fusion between a PLL and a Frequency Divider leads to a Frequency multiplier. The *phase detector* compares the phases of its two inputs: the original input and the feedback signal. The output of the phase detector will be the voltage error. Its high-frequency noise signal is removed by the *Low Pass Filter* (LPF). The *amplifier* amplifies only the low-frequency components of the signal. Then, the *Voltage Controlled Oscillator* (VCO) provides an output frequency proportional to f_s . The *Frequency divider* is to select the proper constant divider N (always an integer). The PLL tries to maintain $f_0 = \frac{f_0}{N}$.

- **Delay-Locked-Loop (DLL)**: this type of circuit is not commonly used, but, in general, it is better than PLL in terms of lower sensitivity to supply noise and its lower phase noise. The DLL [28] can create N equally spaced clock edges that can lead to a higher-frequency output. It can generate several phases according to the number of stages and minimum spacing related to each stage delay.

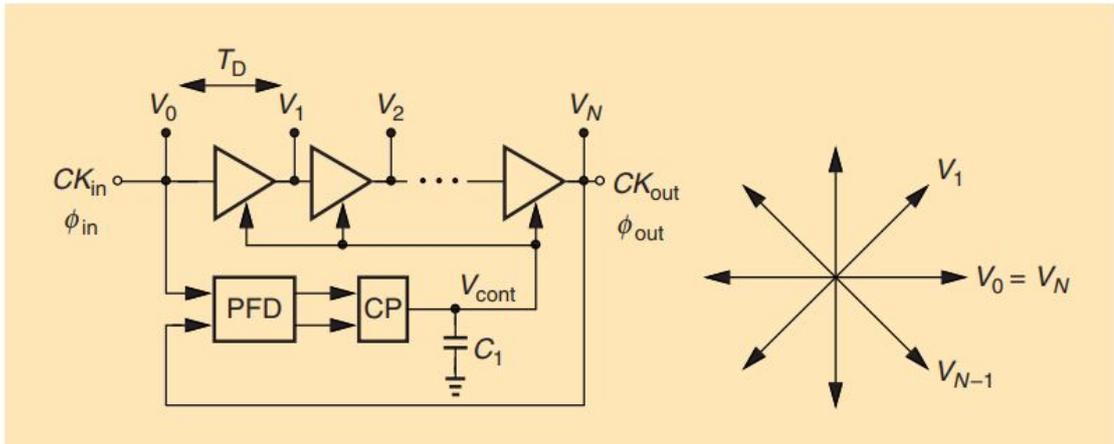


Figure 2.19: DLL as frequency multiplier block diagram. Reprinted from [28]

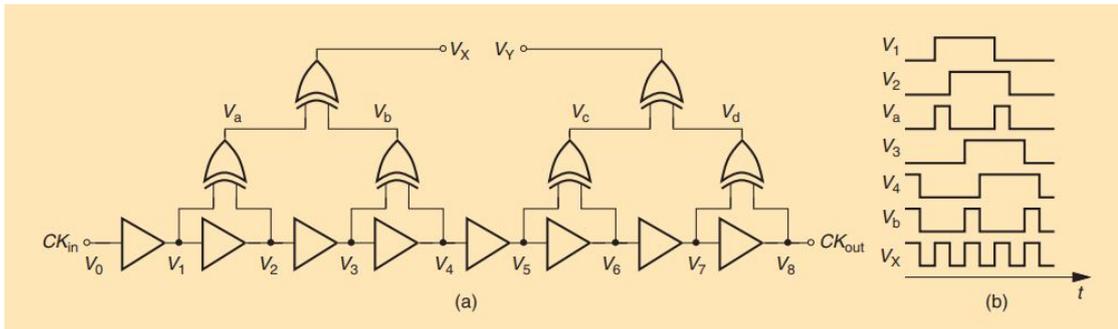


Figure 2.20: Eight-phase delay line. Reprinted from [28]

Two possible implementations of an edge combiner are reported: in Figure 2.20, it is possible firstly to double the frequency and then to make it four times simply through inverters and XOR gates, in 2.21 the same operation is obtained with AND and NOR gates.

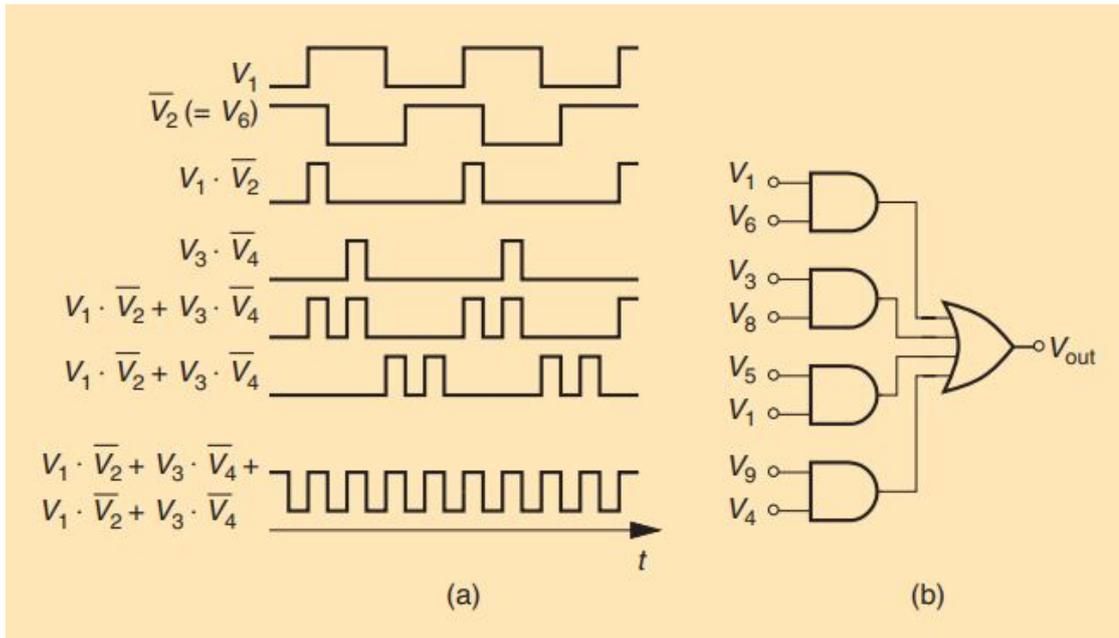


Figure 2.21: Edge combiner. Reprinted from [28]

Thus, even if this solution is easier to be realized than an ordinary PLL, the multiplication factor cannot be easily changed, and the presence of delay mismatches can give birth to jitter and spurs.

- **Mixers:** in the analog case, the mixers can also be used as a frequency multiplier.

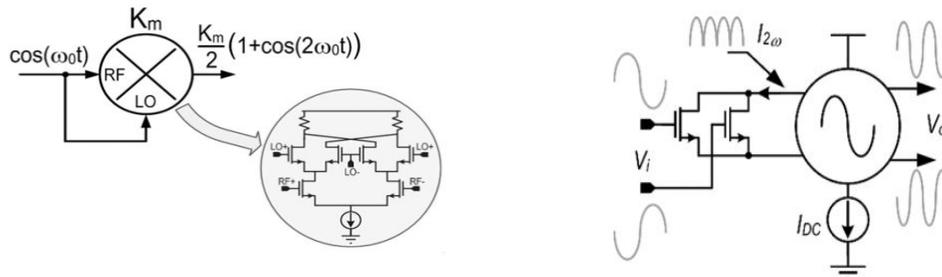


Figure 2.22: (a) Mixer architecture and (b) Frequency multiplier mixer-based. Reprinted from [29]

The solution in [29] reported in Figure 2.22 combines the Gilbert-cell mixer with a push-pull configuration, with the advantages of smaller input capacitance

and higher conversion gain to guarantee a differential output with lower power dissipation.

In this Section, only the most common ways to realize a frequency multiplier with high stability and low noise have been reported. Other solutions can be adopted, for example, simply exploiting the device's non-linearities or using an analog multiplier. The focus is put on solutions that can be implemented in CMOS technology.

2.4 Oscillator

Generally, a clock is needed in architecture where memorization or temporization should be performed. Moreover, this is always generated by an oscillator, which output is buffered. In this project, the presence of an on-chip oscillator is crucial to allow clock memorization.

A thorough study of the solutions in literature has been done.

The classical circuit is the single-ended *Ring Oscillator* (RO) [30], as the one reported in Figure 2.23

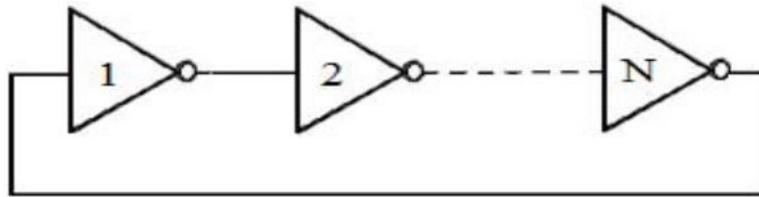


Figure 2.23: Ring oscillator. Reprinted from [30]

where the oscillator frequency is affected by the number of stages (N) and the delay of each inverter (t_d), as follows:

$$f_{osc} = \frac{1}{2Nt_d} \quad (2.2)$$

Another version of this solution can provide for a differential stage, reaching a better phase noise value [31]. The second classical type, widely exploited in literature, is the *Current Starved Oscillator* (CS), reported below:

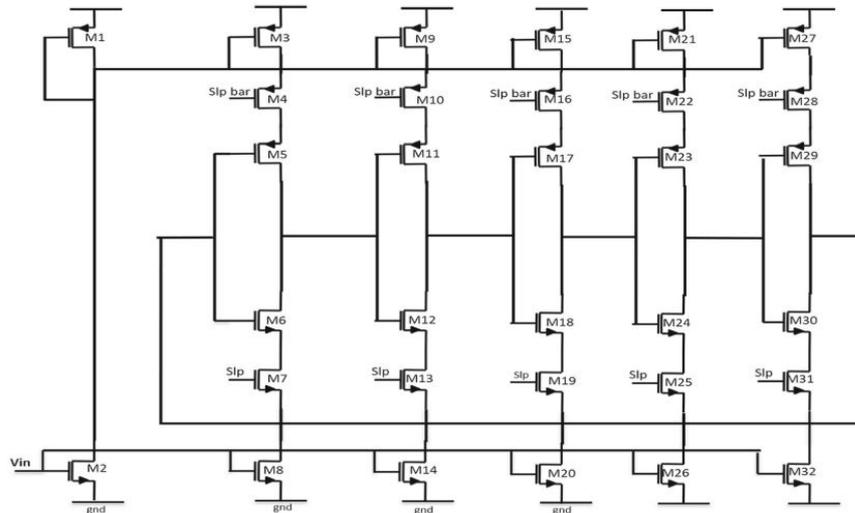


Figure 2.28: CS Sleep VCO. Reprinted from[36]

In normal mode, the sleep transistor is switched off, and the stacked pMOS and nMOS lead to a reduction of the subthreshold leakage. On the other side, in active mode, the sleep transistor minimizes the resistive path, reducing the delay. This CS Sleep VCO results to be a better solution in terms of power, phase noise, and frequency with respect to the classical CS Oscillator, with the only payload of the higher occupied area having a more considerable number of transistors implied. Another attractive solution is found in [37], where the architecture is an adapted version of a RO:

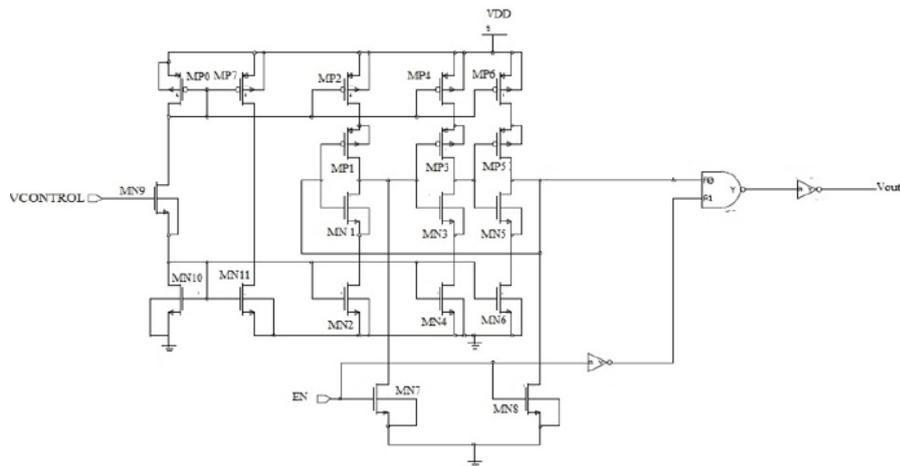


Figure 2.29: CS RO. Reprinted from[37]

This low-area and low-power application is quite attractive, obtained through a reduced frequency range.

The table presented here recaps all the solutions proposed in the literature, trying to underline the most important features that a well-designed oscillator must have.

Table 2.3: Comparison between solutions of literature

Year	f_{OSC} [GHz]	Oscillator type	Technology [nm]	Power [mW]	Phase noise [dBc/Hz]
2015 [1]	2.26-3.5	3-5 stages RO	130	0.031-0.055	-
2015 [2]	4	3 stage CS-VCO	180	7.49	-105.31
2006 [3]	320×10^{-6}	RO	135	$> 1.2 \times 10^{-3}$	-
2011 [4]	4×10^{-3}	Current Controlled O	35	73.7×10^{-3}	-
2012 [5]	$(18.7-32.29) \times 10^{-3}$	3 stages RO	35	6.22	-
2020 [6]	1	Starved Sleep VCO	90	0.00812	-115
2014 [7]	2.42	3 stages RO	180	2.47	-126.4
2019 [8]	$(7-11) \times 10^{-3}$	Current Starved RO	180	4.9×10^{-3}	-119.38

Considering the project specifications, the most feasible architecture has been identified in yellow: the phase noise should always be controlled to have a robust design in terms of oscillator frequency holding, but also, the power has to be reduced as much as possible.

Chapter 3

ASK Demodulator

3.1 Design implementation

The demodulator structure is obtained by adapting the one reported and analysed in [17]. For the sake of simplicity, the schematic is reported below:

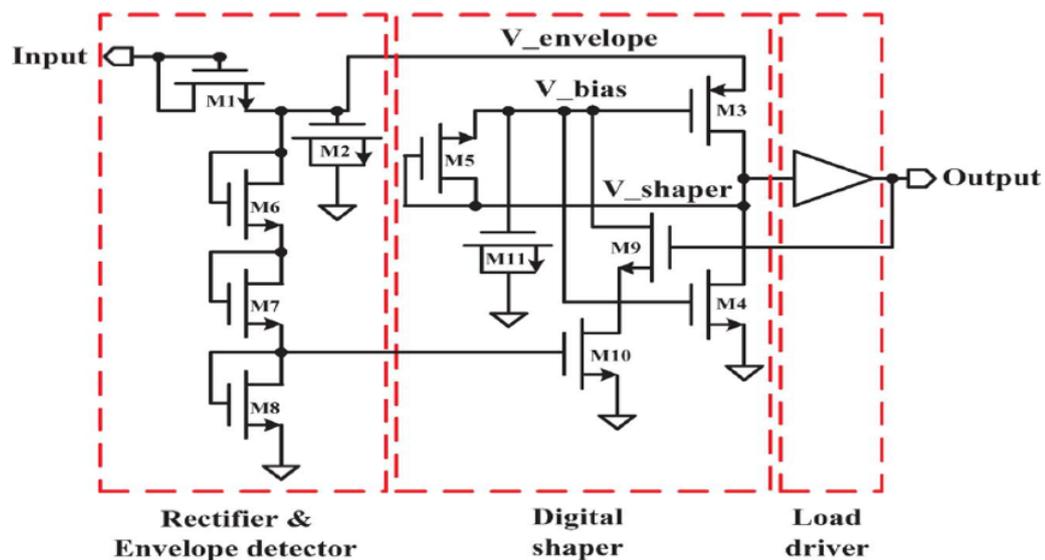


Figure 3.1: ASK Demodulator. Reprinted from [17]

As it can be easily noticed from the Figure, the circuit is composed of three main blocks:

- **Rectifier and Envelope detector:** the ASK modulated signal in input is clipped above the zero to get only positive values and to evaluate the envelope

of the resulted signal;

- **Digital shaper:** the rectified and well-shaped signal is stretched in order to make the analog signal a digital one;
- **Buffer stage:** The real and perfect digital shape is obtained through this block.

This simple ASK demodulator does not provide any clock able to sample the demodulated signal, while it is fundamental.

The paperwork has been re-proposed with some modifications to adapt the solution to the wanted application. For this purpose, the schematic realized on Cadence is reported below:

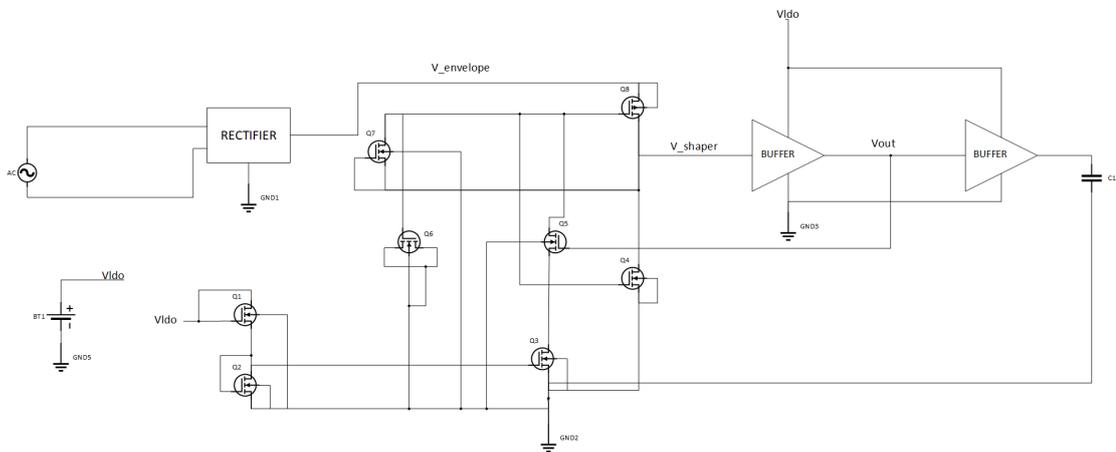


Figure 3.2: Adapted version of Implementation 1

As seen from the image, some changes are introduced with respect to the original solution.

Firstly, the ASK modulated signal is implemented in Matlab with a carrier frequency of 433.92 MHz with a defined modulation index (MI). An example is reported in the Figure below:

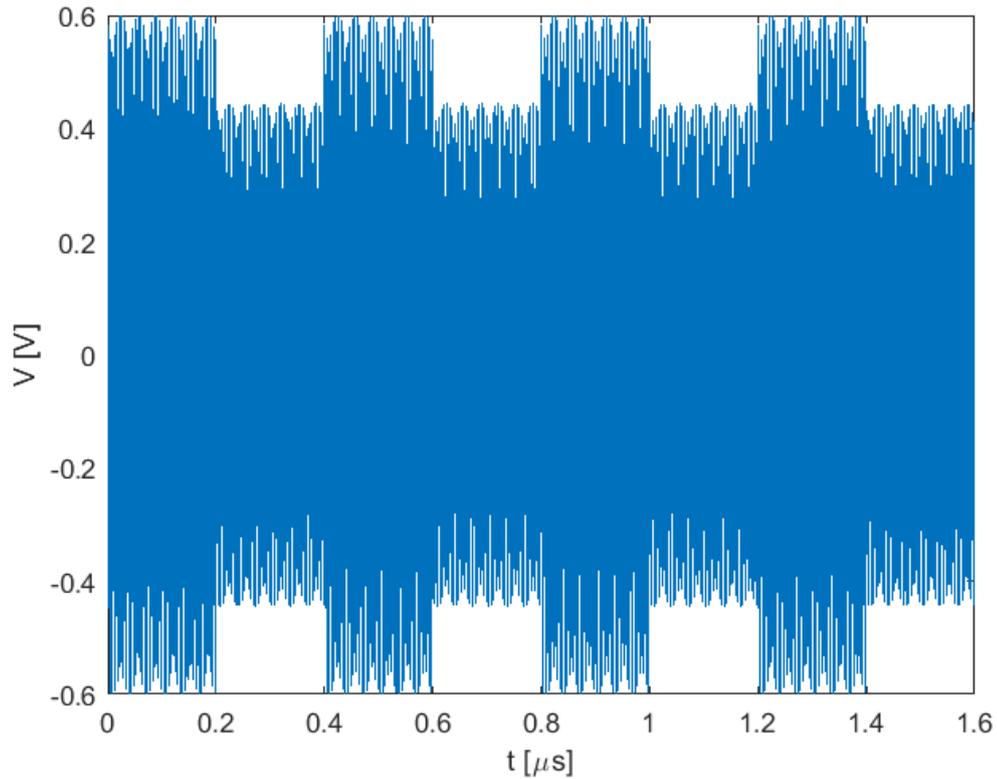


Figure 3.3: ASK modulated signal with $f_c = 433.92\text{ MHz}$ and $MI = 15\%$

3.1.1 Rectifier

The ASK signal passes through an additional block, the **Rectifier**, according to which the signal is rectified and the envelope is done. The rectifier has a modular structure, as can be observed in the Figure reported below:

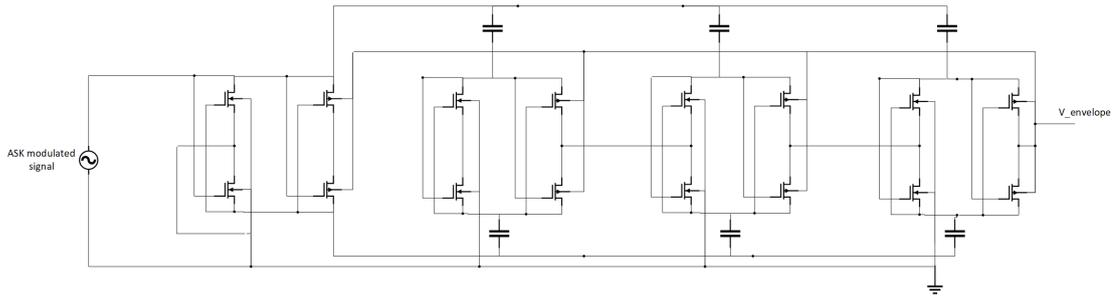


Figure 3.4: Rectifier schematic

It receives the PLUS and MINUS of the ASK modulated signal in the input. At the same time, the rectifier’s output is the $V_{envelope}$ used to get the digital signal with the *Digital shaper*.

In the Figure below, the rectifier output is reported: the values are between almost 1.5 V and 1.2 V.

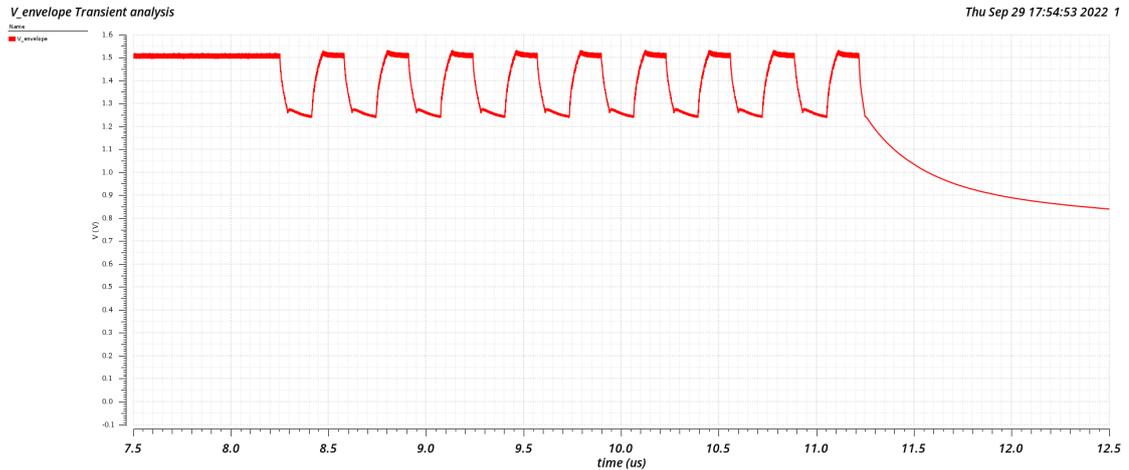


Figure 3.5: Rectifier output

The rectifier block has not been considered a real block of the CDR structure because its optimization in terms of area and power could not be possible, considering that it is also implied for other operations in the overall chip.

3.1.2 Digital shaper

The *Digital shaper* block is made up of only six transistors, plus the two connected to the gate of the transistor $M7$, to act as bias voltage and to mimic the same voltage that in the previous implementation is obtained with the direct connection with the envelope and rectifier block. The implied transistors are all *pmos2v* and

$nmos2v$, characterized by a $V_{th} = 0.6 V$. These two types will be used for all the circuit designs. The only exception is related to the MOSFET needed to realize the bias voltage for which $nmos2vnvt$ have been implied, considering their lower V_{th} . The Digital shaper output is attenuated with respect to the rectifier one, and the signal is stretched closer to the final one, as shown in the following image.

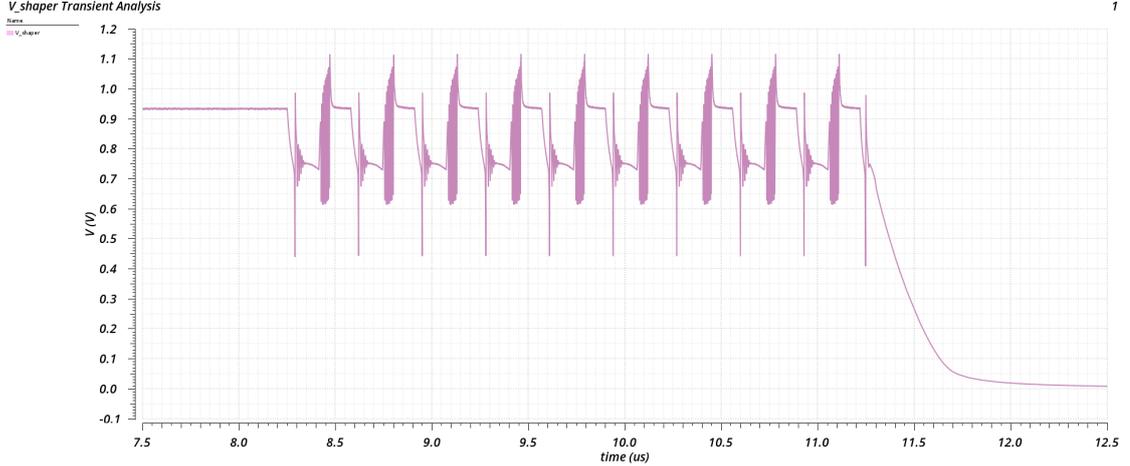


Figure 3.6: Digital shaper output

To get the correct shape of the traces, many parametric analyses have been done, varying the width (**w**) and the length (**l**) of the used transistors:

- **w** between 220 nm and 4 μm ;
- **l** between 180 nm and 1.8 μm .

The iter has been repeated firstly, keeping the minimum length (180 nm) fixed while varying the width to get both the best separation between the low and the high value and a trace lower than the overall V_{DD} of the system, equal to 1.1 V . Otherwise, the signal will not be correctly clamped through the *Buffer*.

3.1.3 Buffers

The main target is to have the input lower than 1.1 V , the V_{DD} . It is important to reduce the number of implied transistors to have an area-efficient demodulator. There are two options:

- use of a voltage shifter to get lower voltages to obtain a value feasible with the input of the Buffer stage;
- reduces the input values to have it lower than the power supply of the Buffer.

The second option is taken into account, trying to size the transistors efficiently to obtain the wanted values.

As widely known, the buffer is realized by employing a chain of two inverters. To evaluate how to size the transistors' dimensions, the ratio $\frac{\beta_p}{\beta_n}$ has been taken into account. To start, the V_{inv} equation, for long channel transistor and low V_{DD} case, is taken into account for the first inverter:

$$V_{inv} = \frac{V_{DD} + V_{tp} + V_{th} \sqrt{\frac{1}{r}}}{1 + \sqrt{\frac{1}{r}}} \quad (3.1)$$

where $r = \frac{\beta_p}{\beta_n}$

This has been reversed to get the r expression:

$$\frac{\beta_p}{\beta_n} = \frac{1}{\left(\frac{V_{DD} - V_{tp} - V_{inv}}{V_{inv} - V_{tn}} \right)^2} \quad (3.2)$$

By considering that $V_{tn} = |V_{tp}| = 0.6 V$ and $V_{inv} = \frac{V_H + V_L}{2}$, the result leads to $\frac{\beta_p}{\beta_n} > 1$, which directly means that the first inverter of the buffer should be *HIGH SKEWED*: the width dimension of the pMOS has to be larger than the one of the nMOS.

Moreover, an additional problem has been experienced mainly related to the input shape: while in the original paper the rectified signal was between $0 V$ and a specific V_{value} , in this case, it oscillates between $0.7 V$ and $0.9-1.0 V$, approximately. The low level could activate both the pMOS and the nMOS of the first inverter; thus, the first inverter has been adapted to solve this problem, putting an nMOS in series in the pull-down network. Another advantage is obtained: if these are switched off, there is a much lower leakage current than one would have with just one pull-down transistor, the so-called ***Stack effect***.

The schematic of the circuit has been reported below:

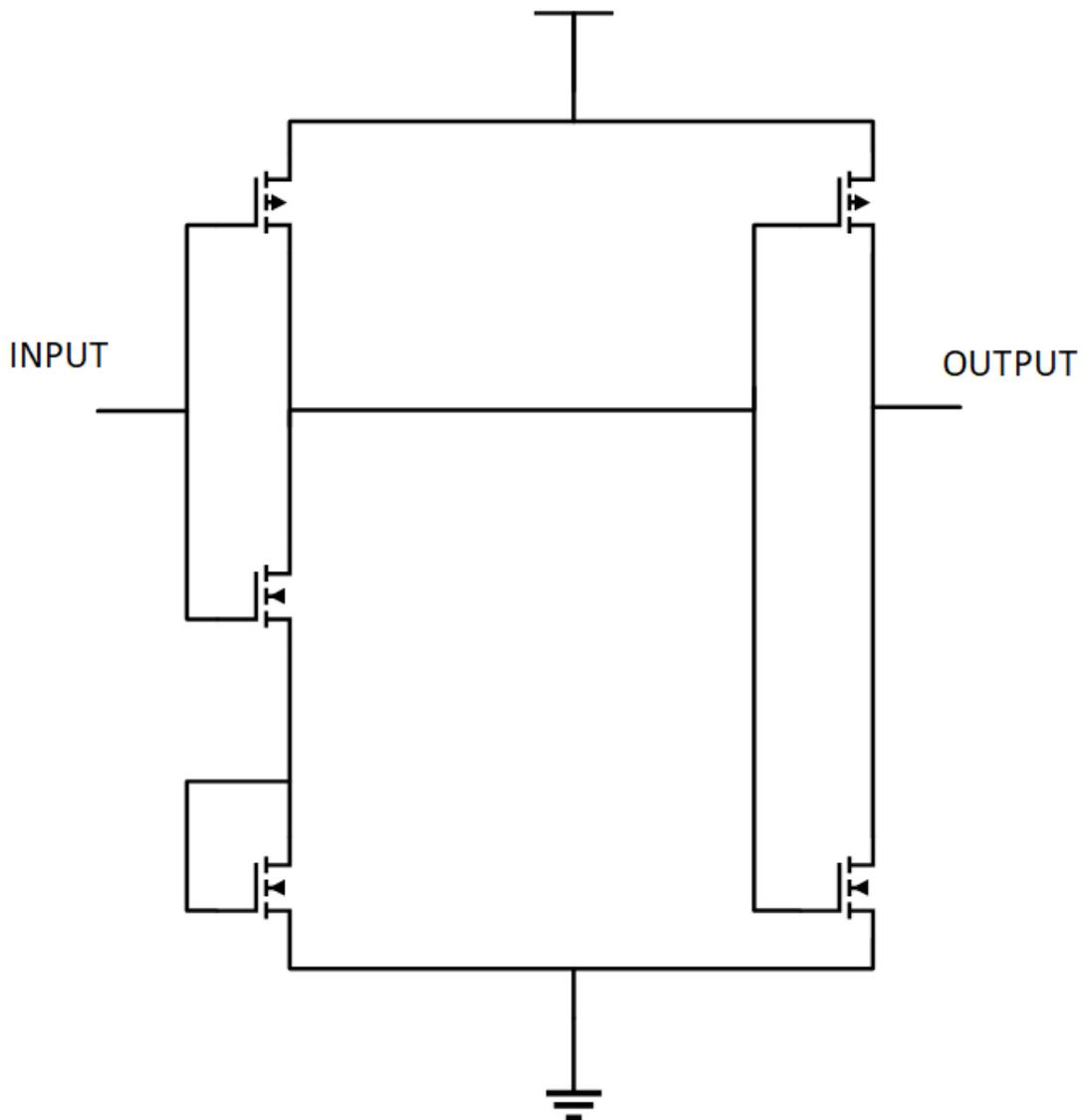


Figure 3.7: Buffer circuit

The second inverter is characterized by the minimum dimension. Moreover, an additional buffer with minimum size has been introduced at the output of the circuit, out of the feedback loop. This introduction has been handled considering that, after having realised stable negative feedback, the small error introduced at the output is always presented again at the input: the additional buffer is considered essential to get a perfect digital shape. For explanation, these behaviours have been reported in Figures below. This image shows how the output reconstructs the input analog signal correctly.

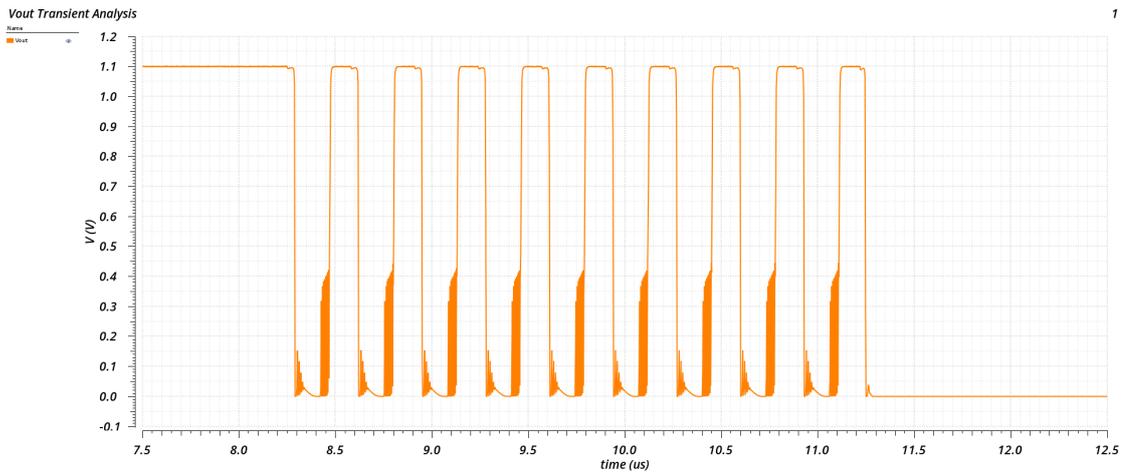


Figure 3.8: Output of the first buffer

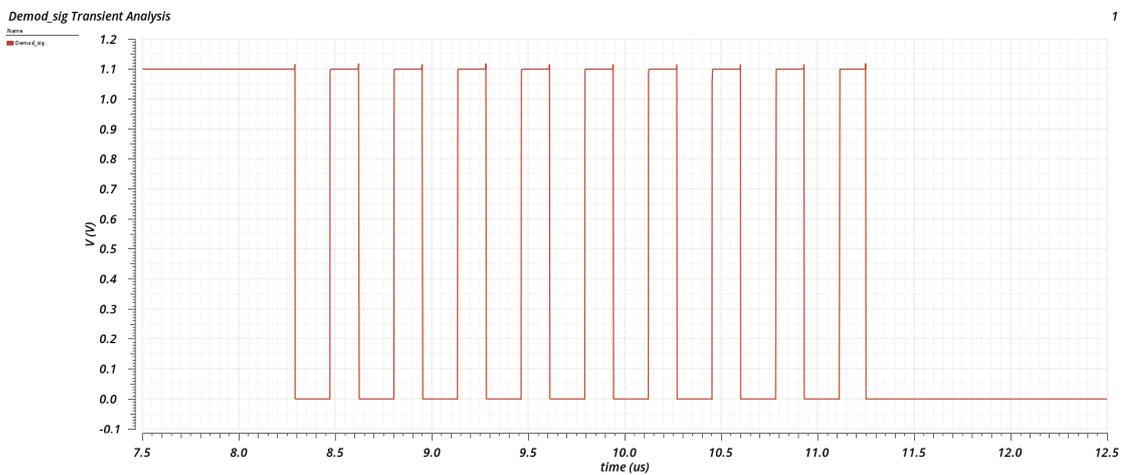


Figure 3.9: Output of the second buffer

The layout has been designed. The demodulator is composed of a reduced transistor number, each with a different length(L) and width (W). In this case, the detached and the integrated body are needed due to the absence of the power supply for half of the circuit.

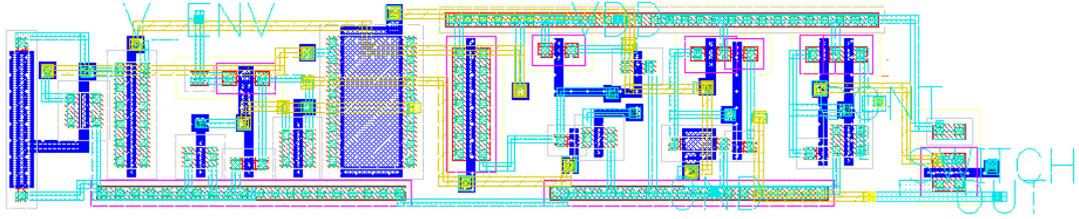


Figure 3.10: Demodulator layout

As can be displayed, a part of the circuit is related to V_{DD} , while the other half to $V_{envelope}$.

3.2 Performance analysis

The simulations have been performed considering different parameters to check the strength of this implementation properly and to figure out weak points simultaneously. Three parameters, strictly linked to each other have been taken into account:

- **Data rate:** as it has been previously defined, the Data rate is strictly related to the frequency of the input signal and the bit rate. The data rate has been swept between 1 Mbps and 6 Mbps;
- **Modulation index (%):** it is widely known that this parameter is mainly related to the high voltage level (V_H) and to the low level (V_L) of the ASK modulated signal. The analysis is done by considering two different cases:
 1. **constant** $V_H = 600 \text{ mV}$ changing the low level voltage;
 2. **constant** $V_L = 500 \text{ mV}$ changing the high level voltage;

The MI is presented for different cases, reported in the following table, by considering the two cases introduced above.

MI (%)	$V_H = 600 \text{ mV}$	$V_L = 500 \text{ mV}$
9 %	500 mV	600 mV
15 %	444 mV	676 mV
20 %	400 mV	750 mV
30 %	324 mV	930 mV

Table 3.1: Amplitude levels variation with MI

The values above are reported to consider how the input voltage amplitude changes.

- **Adapted duty cycle ratio:** a new figure of merit has been introduced to determine if going from the *Analog* world to the *Digital* one the duration of one and the one of zero is correctly handled. It has been defined as follows:

$$\frac{VH_{avg}}{VL_{avg}} = \frac{duration(VH(t))}{duration(VL(t))} \quad (3.3)$$

The introduced ration takes into account the averaged duration in time of one and zeros in the case in which the ASK modulated signal is composed of nine zeros and nine ones.

Ideally, it should be equal to 1, meaning that the duration of ones is equal to the one of the zeros.

The three parameters taken into account have been considered to create some graphs to see the limits of the proposed ASK demodulator.

The analysis has been performed considering the amplitude of the two levels of the ASK modulated signal.

The study has been applied first to a constant VL while changing the VH in agreement with the tuning of the MI, as can be seen from Figure 3.11.

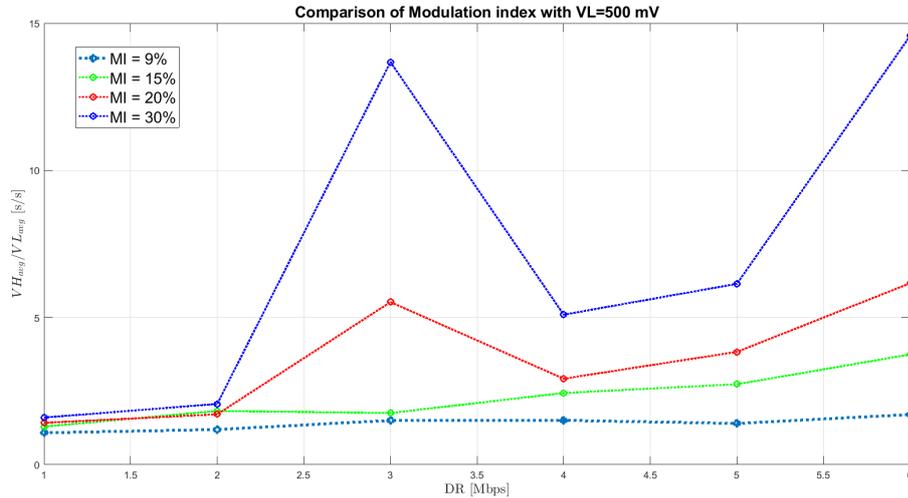


Figure 3.11: Variation of $\frac{VH_{avg}}{VL_{avg}}$ according to MI

The image takes into account on x -axis the Data Rate variation, while on y -axis the $\frac{VH_{avg}}{VL_{avg}}$ ratio. The curves represent different types of MI. As it can be immediately observed from the graph, by increasing the MI, the ideality is lost, and only the results related to MI equal to 9% and 15% can be considered acceptable up to 4 Mbps. The worst case can be observed for a 30% MI for which the duration of ones is 14 times higher than the one of the zeros.

The analysis has also been done considering a constant VH while modifying the VL with respect to the MI change, as reported in this image.

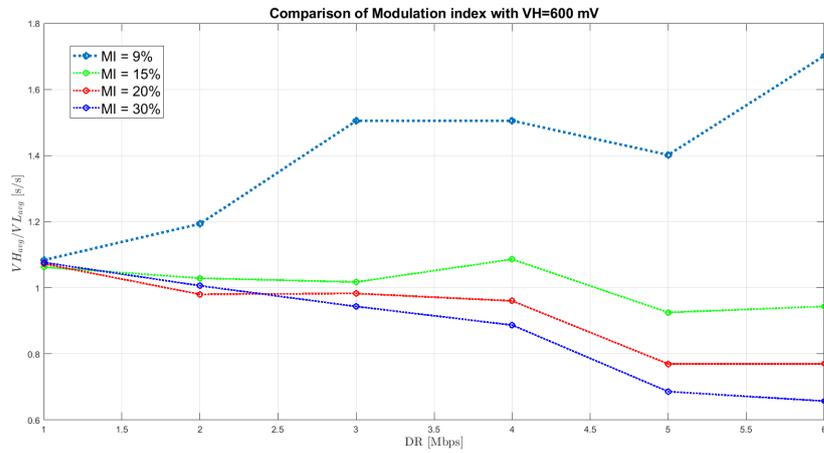


Figure 3.12: Variation of $\frac{VH_{avg}}{VL_{avg}}$ according to MI

The Figure shows the MI trends. As it can be noticed, the demodulator works well up to a DR= 6 Mbps with a modulation index of 15 %. The variability of the MI is quite broad, between 9% and 30% according to the data rate taken into account. The demodulator looks compliant with the wanted specifications.

After evaluating the worst case, an alternation between ones and zeros, a random bit sequence has been considered by generating the modulated signal with Matlab.

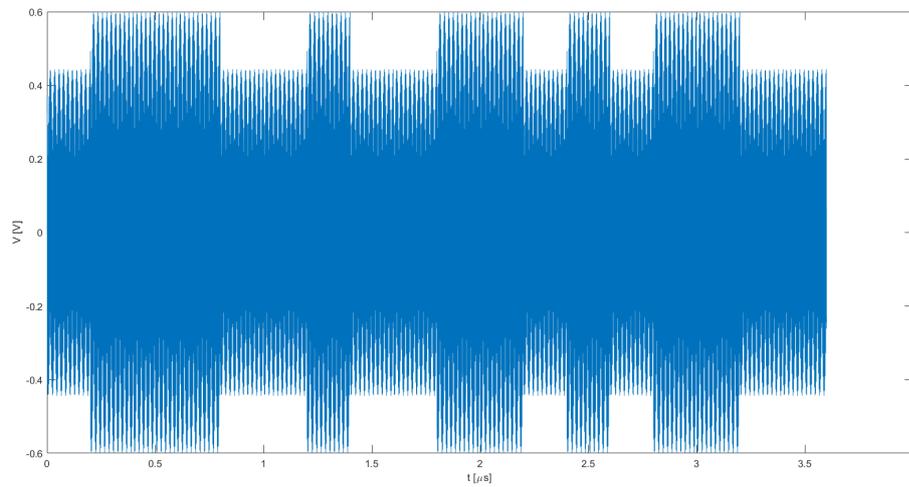


Figure 3.13: Random signal modulated@433.92 MHz

The demodulator has been simulated with this kind of input, and then the obtained trace has been reported on Matlab, getting the following result:

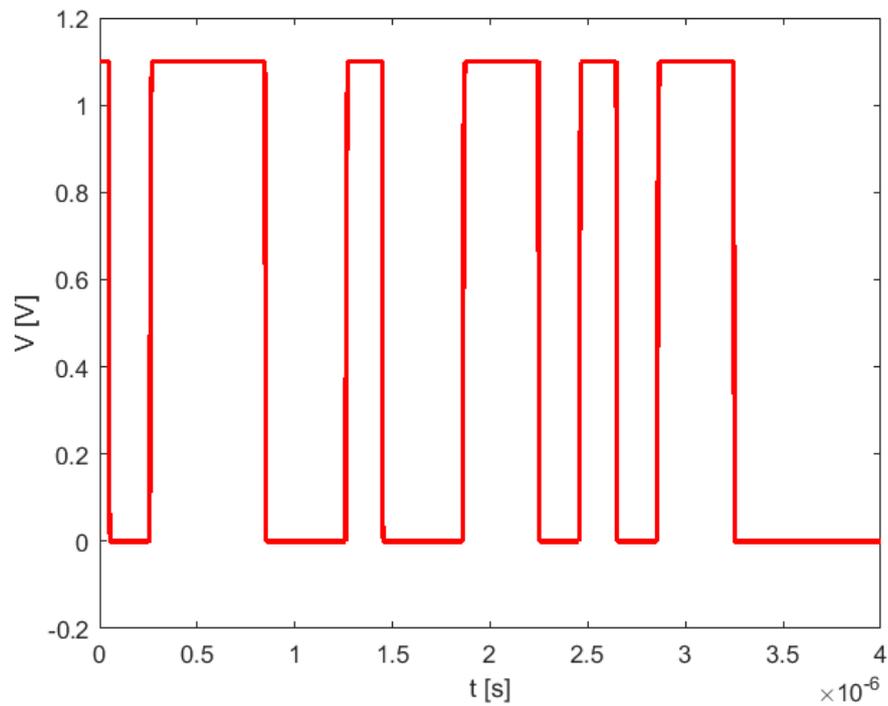


Figure 3.14: Random signal demodulated@433.92 MHz

3.3 Power&area analysis

To correctly evaluate all the demodulator features and to have a general overview of the implementation, the area has been approximately calculated as equal to $(8.93 \times 9.32) \mu m^2$, value precisely computed and reported in Chapter 5. In addition, power consumption has been taken into account.

The procedure applied is done taking into consideration the **Averaging power** on a period, T . The analysis has been performed with $T = 333 \text{ ns}$, $MI = 15\%$.

$$P = \frac{1}{T} \int_{t_0}^{t_0+T} p dt = \frac{1}{T} \int_{t_0}^{t_0+T} V \cdot I dt \quad (3.4)$$

This computation was repeated twice because, as already pointed out, the bias voltage was built using two *nmos2vnvt*, which led to the power consumption of almost $31 \mu W$: by simply replacing them and changing the ratio between W and L , a reduction is obtained. Thus, it is reported in the second case where the power takes into account two contributions:

1. the one related to the VDD coming from the rectifier:

$$P = \frac{1}{333 \text{ ns}} \int_0^T V_{envelope} \cdot I dt = 8.26 \mu W \quad (3.5)$$

2. the one related to the VDD coming from the LDO:

$$P = \frac{1}{333 \text{ ns}} \int_0^T V_{ldo} \cdot I dt = 1.15 \mu W \quad (3.6)$$

By considering this overall view of the implemented circuit, it can be possible to make a comparison with the paperwork that was the starting point.

The table below tries to highlight the most important differences.

Implementation	Data rate (DR)	Area	Power consumption
Paperwork [17]	1.2	101 x 32	306
My implementation	1-6	8.93 x 9.32	9.41 (averaged for 6 Mbps)

Table 3.2: Comparison with the paperwork [17]

The values reported above consider an outstanding improvement with respect to the starting point: the implementation realized gets a consistent reduction in terms of area, almost one order of magnitude. The same can be reported for power consumption. Moreover, the designed demodulator is robust for a wider range of input data rates: it works properly between 1 and 6 *Mbps*. In addition, this work

operates at a higher frequency because the input ASK modulated signal has a carrier frequency of 433.92 MHz (RF signal), while the one in the paper works at a lower frequency (13.56 MHz). The demodulator is adaptable to the modulation index of the input, too; it has a good behavior between 9 and 30%, while the one presented in the paper works as expected only for a fixed value (11.11%).

Moreover, to reduce the power consumption, attention should be put on the sum of the **Static Power** and the **Dynamic Power**:

$$P \sim E_{SW} (C_L + C_{SC}) V_{DD} V_{swing} f + (I_{DC} + I_{leak}) V_{DD} = P_{dynamic} + P_{static} \quad (3.7)$$

The Dynamic Power is influenced by the load (C_L) and the short circuit capacitances (C_{SC}) and by the frequency, too; on the other hand, the Static Power depends on the two contributions of short circuit and leakage current.

As it has been previously mentioned, the Dynamic Power has a direct dependency on operative frequency; therefore, the analysis of the following relation provides some information on how much energy is dissipated when a commutation occurs:

$$P_{dyn} = C_L \cdot V_{DD}^2 \cdot f \cdot P_{0 \rightarrow 1} = C_{switched} \cdot V_{DD}^2 \cdot f \quad (3.8)$$

By knowing $C_{switched} = E_{SW} C_L$; it can be taken into account that P_{dyn} depends on the load capacitance too.

There are several techniques able to reduce the power consumption:

- **Static Power reduction:**

1. ratioed circuits;
2. low threshold nMOS and pMOS devices;
3. stacking effect;
4. body bias.

- **Dynamic Power reduction:**

1. reduction of the switching activity (α) by clock gating and sleep mode technique;
2. capacitances reduction implying small transistors and short wires;
3. lower power supply by Dynamic Voltage Scaling (DVS);
4. lower operating frequency by Dynamic Frequency Scaling (DFS).

In the case under analysis, power supply reduction techniques could not be applied by considering that half of the circuit uses as V_{DD} the rectifier output, while the other one takes into account the power supply provided by the LDO; thus, not having a uniform power supply, it is not mean using a powerful technique such as the DVS.

Moreover, as already pointed out, the power consumption improvement has already been made by identifying the "weak" components in terms of power, the two transistors used to create the $V_{bias} = 600 \text{ mV}$. Indeed, the circuit power consumption has been reduced by the 69 % by simply substituting the two Low-Threshold transistors with two with Standard-Threshold. It is widely known that the lower the V_{th} , the higher the leakage current according to the following relation:

$$I_{leakage} = I_0 e^{\frac{V_{GS} - V_{th}}{nV_T}} \left(1 - e^{-\frac{V_{DS}}{V_T}} \right) \quad (3.9)$$

$I_{leakage}$ is mainly related to the static current, and thus this leads directly to an increment in the Static power consumption.

3.4 Noise analysis

A noise analysis has been performed to test the demodulator robustness. The frequency range is between 100 *MHz* and 1 *GHz*, with a step of 10 *MHz*.

The noise evaluation has been done according to the concept of Signal to Noise Ratio (SNR):

$$SNR = 20\log\left(\frac{A_{signal}}{A_{noise}}\right) \quad (3.10)$$

The extracted noise report refers to two contributions:

1. **Total Input Referred Noise:** it takes into consideration how the input noise affects the demodulator behaviour at the input:

$$SNR = 20\log\left(\frac{1.1V}{54.47mV}\right) = 26.1dB \quad (3.11)$$

This result is meaningful considering that the rectifier output is noisy because it has been obtained from a signal with a very high carrier frequency.

2. **Total Output Voltage:** it reports the value from which we can obtain the real SNR, considering that it takes into account the noise at the output of the circuit:

$$SNR = 20\log\left(\frac{1.1V}{160\mu W}\right) = 76.74dB \quad (3.12)$$

The obtained value considers to have a good SNR by evaluating as signal amplitude the full sweep of the demodulated signal, from 0 to V_{DD} (1.1 V).

Chapter 4

Clock recovery system

4.1 Design implementation

After the signal demodulation, it is necessary to get the clock to sample the data correctly. The clock has been extracted directly in two different ways: in the first case, it has been considered to recover it from the demodulated signal and, through a frequency multiplier, increase its frequency to sample the demodulated data correctly; in the second case, sending the clock at higher frequency with respect to the data to be compliant with the Nyquist criterion.

The clock is synchronously generated from the demodulated signal. In the first case, a complex control system has been designed and reported in the figure below

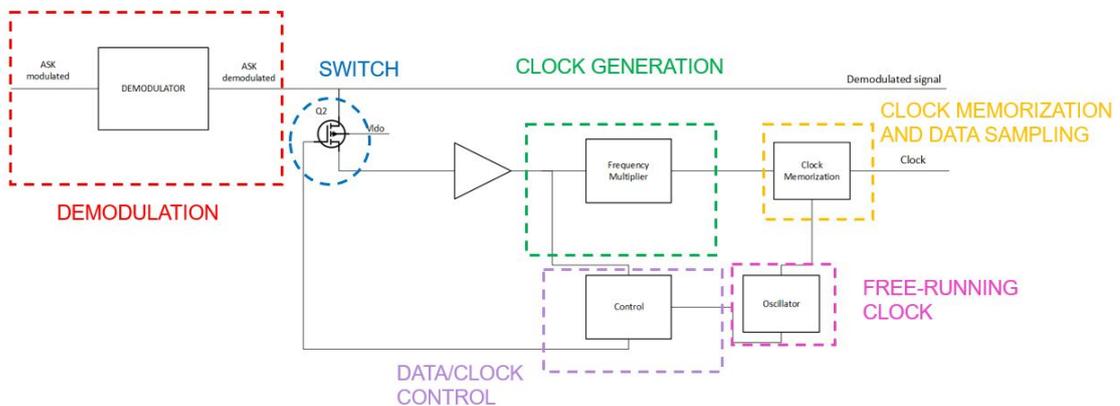


Figure 4.1: First implementation

The system is organized as follows:

- **Demodulation:** it is responsible for the data demodulation, as explained in

the previous Chapter;

- **Switch:** it activates the clock control system;
- **Frequency multiplier:** from the demodulated signal, the clock is extracted, and it is led at a higher frequency using this block;
- **Data/Clock Control:** it differentiates when data or clock is sent;
- **Clock memorization and data sampling:** the clock memorization is performed, and with the saved clock, the data is sampled.

4.2 Switch

The entire system is controlled by a switch that should let the clock information pass. Moreover, the first data transmission is sent to memorize only the clock. This switch should switch on only the first time; after, it should remain switched off for the entire data transmission.

The MOSFET involved is a *pMOS*: at the beginning, all the circuit's nodes are zeros; for that reason, with an nMOS, the switch would never be open. The drawback is related to the power consumption:

- The body terminal should be connected to V_{DD} in the pMOS case;
- After the first transmission to extract the clock information, to switch off the pMOS forever, the gate should be maintained continuously to V_{DD} .

Moreover, after the clock passage through the switch, a buffer has been introduced to completely reconstruct the demodulated signal shape ruined by the switch's non-linearities.

4.3 Frequency multiplier

The clock is extracted through a frequency multiplier. The circuit should be as much as low-area and low-power as possible. To correctly sample the data, the clock should be at a higher frequency with respect to the data to be sampled. The Nyquist theorem has to be taken into account:

$$f_s \geq 2f \quad (4.1)$$

Where f is the frequency of the signal to be sampled and f_s is the sampling frequency.

To be sure of doing a correct sampling operation, f_s is chosen as four times the signal one. By letting the designed system work at the higher data rate tolerated by the demodulator, the clock is demodulated at the higher data rate (6 Mbps). The multiplier can reach a frequency four times higher than the starting one. Thus, having the clock at 12 MHz, the data transmission can be done at 3 MHz by exploiting the demodulator upper bound.

In the state-of-the-art of frequency multiplier, complex solutions are proposed, as described in Chapter 2. Moreover, all the ideas do not fit with the specifications of low-power and low-area. Thus, the most straightforward implementation has been adopted to reach the wanted result.

The clock is generated by a chain of two modular blocks, at which output the signal frequency is increased by a factor of two.

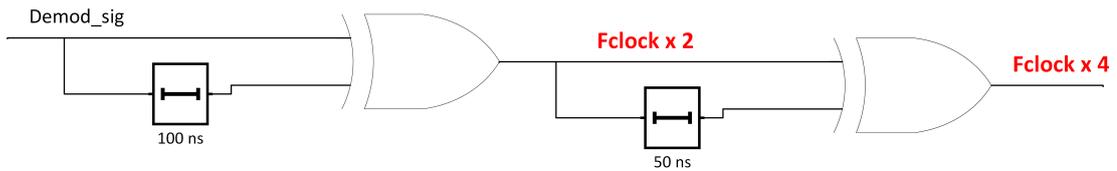


Figure 4.2: Frequency multiplier@12 MHz

The two blocks have been designed accordingly: the XOR gate that detects a phase difference receives as input the demodulated signal, and its copy delayed by one-fourth of its clock period. The output of the XOR is two times the original frequency.

To get the wanted frequency, this block has been applied two times.

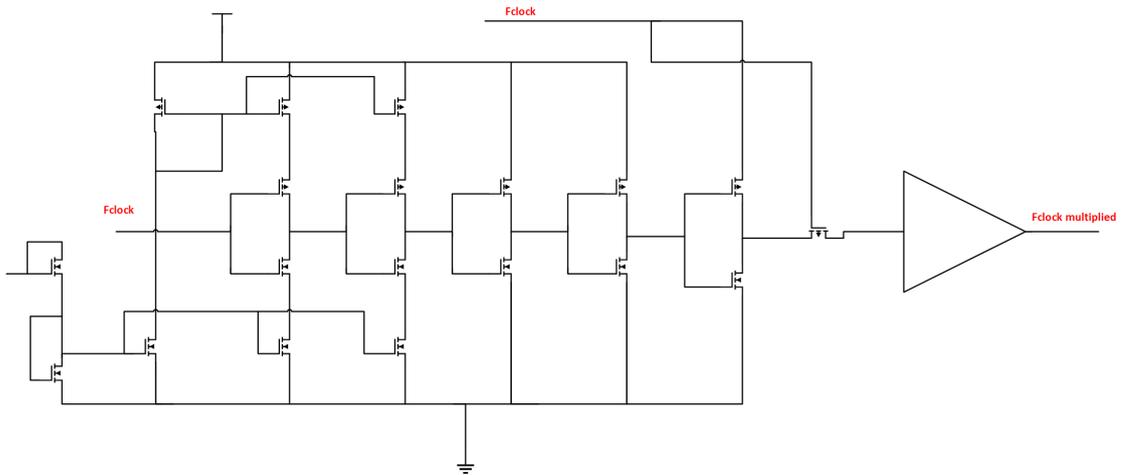


Figure 4.3: Modular block frequency multiplier

In Figure 4.3, the module of the frequency multiplier is reported. It is organized into three sub-blocks:

1. **Delay line:** the delay line can be implemented in different ways. The delay block has been designed to reduce as much as possible the transistors' number. The one taken into account is based on the concept of the *Voltage Controlled Delay Line (VCDL)*: it is realized using the "current starved" inverter. The delay is made adaptable by changing the voltage value connected to the gate transistors of the lower part of the circuit. Due to the MOSFET non-linearities, a buffer is introduced before the XOR gate too;
2. **XOR gate:** generally, the XOR gate is realized in the two ways reported below:

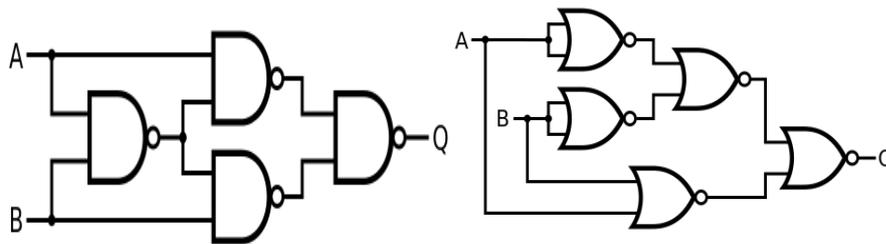


Figure 4.4: XOR (a) NAND- and (b) NOR-based. Reprinted from web

In order to realize a low power and low size phase difference detector, as the XOR gate, simply three transistors have been implied with respect to the classical sixteen/twenty MOSFETs.

3. **Buffer**: this block is a high-skewed buffer inserted to restore the correct shape of the signal. The XOR gate introduces a change in amplitude, mainly related to the phase difference.

For clarity, the waves represent the correct behavior and the outputs of the frequency multiplier.

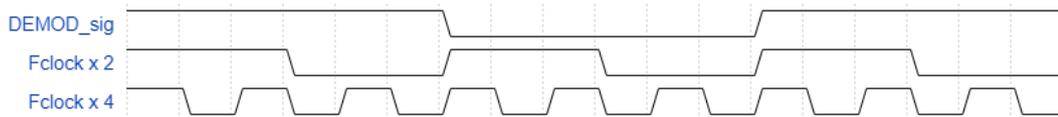


Figure 4.5: Frequency multiplier waves

4.4 Data/Clock Control

To correctly generate and save the clock, a control system should be introduced. As already said, the clock is directly obtained from the demodulated signal.

4.4.1 Transmission protocol

Firstly, a transmission protocol should be defined to distinguish between the clock and the data transmission correctly. The one adopted is reported below:

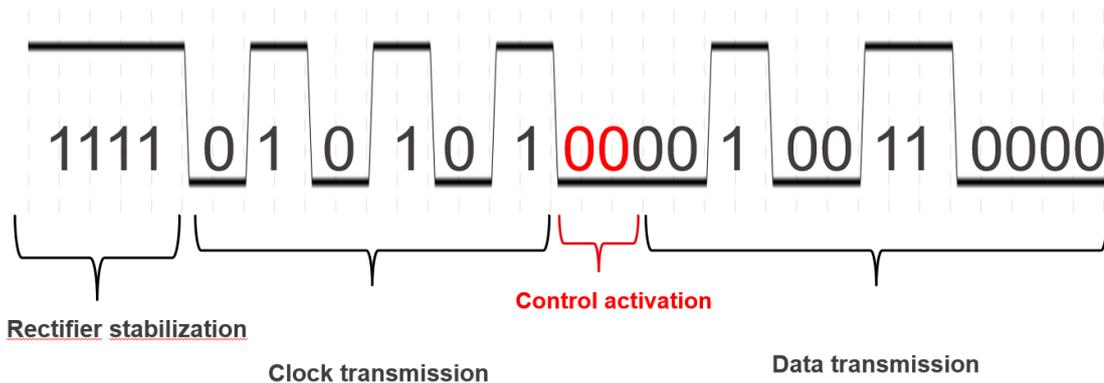


Figure 4.6: Transmission protocol

The first part is a train of ones needed to stabilize the rectifier. Then, a train of pulses from which the clock will be extracted, made by the alternation of ones and zeros, is sent.

The two red zeros activates the control to let the clock reaches the frequency multiplier. After these two zeros, every kind of data can be present, and it will be

sampled with the clock generated by the frequency multiplier.

Once the transmission protocol has been defined, the control has been designed in order to be able to make permanently inactive the switch pMOS by applying a high voltage to its gate. The wanted signal is reported below:

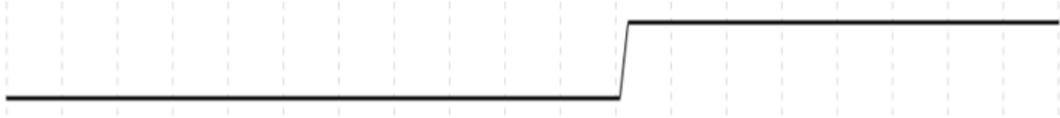


Figure 4.7: Switch control signal

Different control block systems are implemented, trying to reduce the transistors' number every time.

The first implementation takes into account a duty cycle not of the 50% by considering the possibility that the demodulator cannot reconstruct a perfect duty cycle. Thus, the duration of one will be longer than the one of the zero: this is why the control block, in the first cases, will be more complex than the others.

The first implementation is reported below.

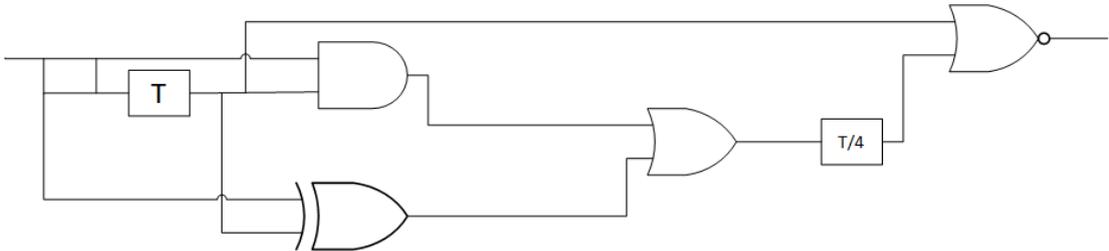


Figure 4.8: First control block implementation

By playing with the demodulated signal, some logic and delay block the switch control signal is obtained.

Moreover, the *DeMorgan theorem* is applied to reduce the logic.

A recall of the theorem is necessary:

$$\overline{A \cdot B} = \overline{A} + \overline{B} \quad (4.2)$$

$$\overline{\overline{A} + \overline{B}} = A \cdot B \quad (4.3)$$

By checking the logic equation of the circuit above, the expression can be reported as:

$$A \cdot B + [(A + B) \cdot \overline{A \cdot B}] \quad (4.4)$$

By remembering the elementary equations:

$$A \cdot \overline{A} = B \cdot \overline{B} = 0 \quad (4.5)$$

$$A + \overline{A} = B + \overline{B} = 1 \quad (4.6)$$

The expression is simplified as:

$$\begin{aligned} A \cdot B + \overline{A} \cdot A + \overline{B} \cdot A + \overline{A} \cdot B + \overline{B} \cdot B &= \\ B \cdot A + B \cdot (\overline{A} + \overline{B}) + A \cdot (\overline{A} + \overline{B}) &= \\ B \cdot (A + \overline{A}) + A \cdot \overline{B} &= B + A \cdot \overline{B} \end{aligned} \quad (4.7)$$

The circuit is reduced as the following:

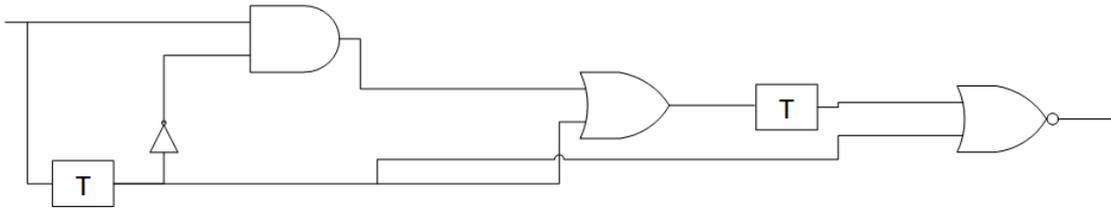


Figure 4.9: Second control block implementation

The new reduced circuit implies a lower number of logic gates. Even if the two one-period delays are quite expensive both in terms of area and power. Moreover, by considering to have an almost perfect duty cycle, the circuit can still be simplified:

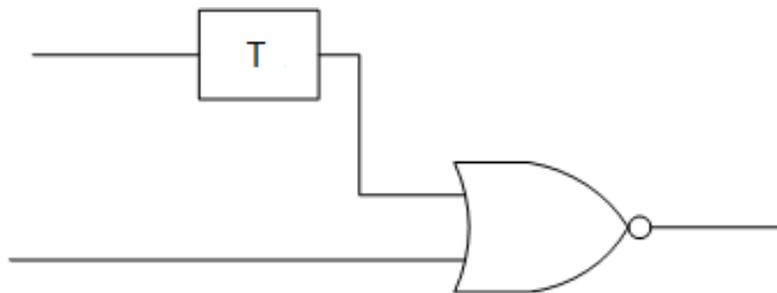


Figure 4.10: Third control block implementation

The delay is considered equal to a period T. The resultant waves are reported below:

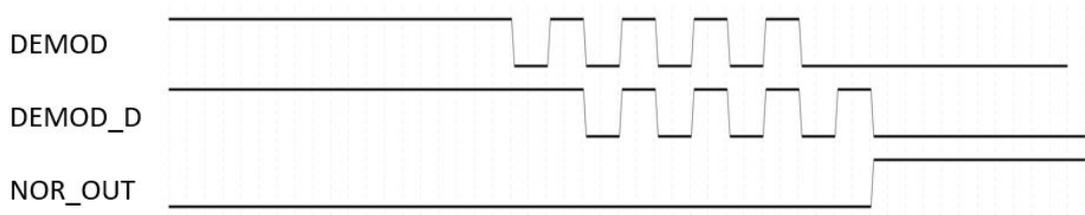


Figure 4.11: Control signals

4.5 Clock memorization and data sampling

Once the clock is extracted and generated by the frequency multiplier, it should be memorized continuously to make it available for data sampling. The structure for the clock memorization is reported below:

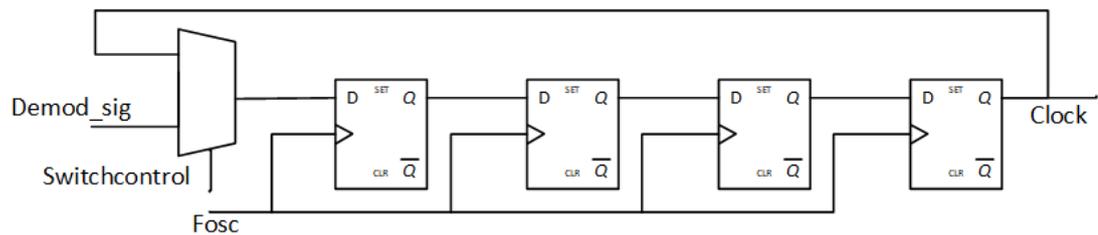


Figure 4.12: Clock memorization block

The design is composed of two main blocks:

- **Multiplexer:** it has two inputs; one is the output of the frequency multiplier while the other one is the output of the D-FF chain. The enable is the control block system's output, the SR register's SET. The multiplexer 2x1 is generally realized in the following way:

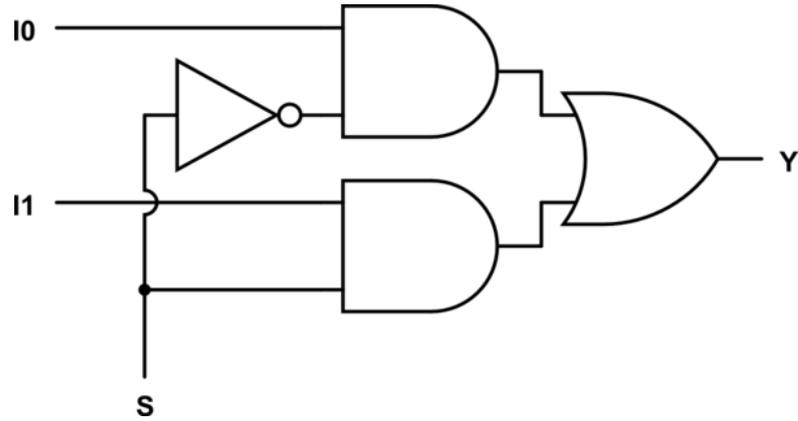


Figure 4.13: Classical implementation of 2x1 multiplexer. Reprinted from web

According to the enable value, one of the two inputs is selected. Considering the low area and low power specifications, the idea exploited in [38] has been applied. The proposed solutions take into account four different implementations. The ones that imply only two transistors have been discarded due to the non-full swing produced output and the consequent generation of a defective signal.

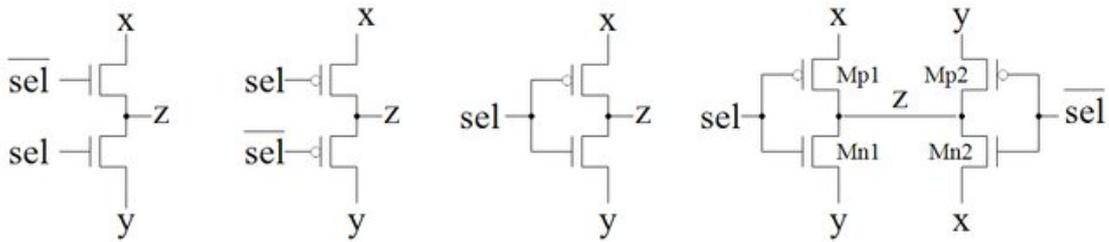


Figure 4.14: Low area MUX2x1 solutions. Reprinted from [38]

The chosen circuit is the fourth design due to its stronger robustness.

- **D Flip-Flop:** The D Flip-Flop (DFF) is the elementary cell of the chain. The signal is delayed by half a clock cycle according to how its clock is generated. Also, in the case of the D-FF, there are different types of implementation: the ones realized with logic gates and the ones with transmission gates (TG). The second solution is considered because the transmission gate leads to faster signal propagation. Moreover, the presence of the clock and the negative clock allows having a more robust circuit with a small power consumption payload. The circuit used is reported below.

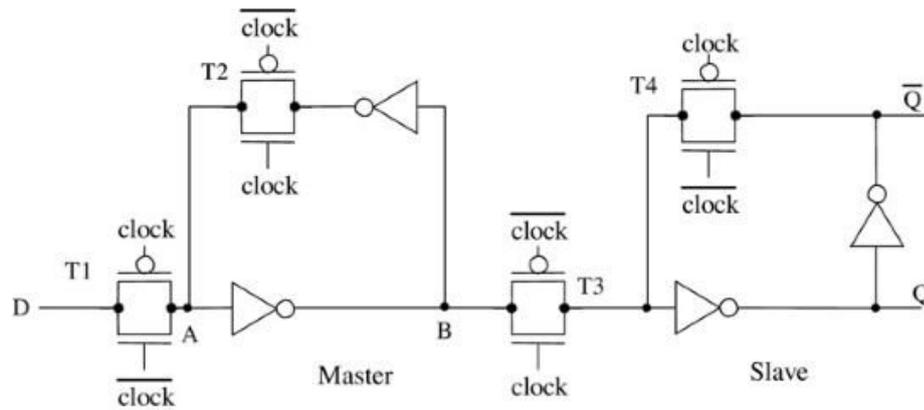


Figure 4.15: Transmission gate D-FF. Reprinted from web

As a recap, the signal arrives from the frequency multiplier, and, after, only zeros are present, giving the possibility to memorize the clock. The signal passes for the first cycle through the D-FF chain, and the first multiplexer input is active. When the signal becomes only zeros, the multiplexer's selector, the SR register's output, is switched according to the control output, and the second input passes. The memorized clock is continuously reported as input in the loop to be ready every time a new bit should be sampled.

The memorisation block layout has been organized in order to put the clock(CLK) and its inverted version (\overline{CLK}) in the center for all the four involved D-FFs: in this way, the propagation delay is reduced. It is mainly equal for all the sequential blocks. Moreover, the power supply has been shared between two D-FFs by two, while the ground is in common between the four blocks by flipping the two D-FFs at the bottom. The first small block is the multiplexer.

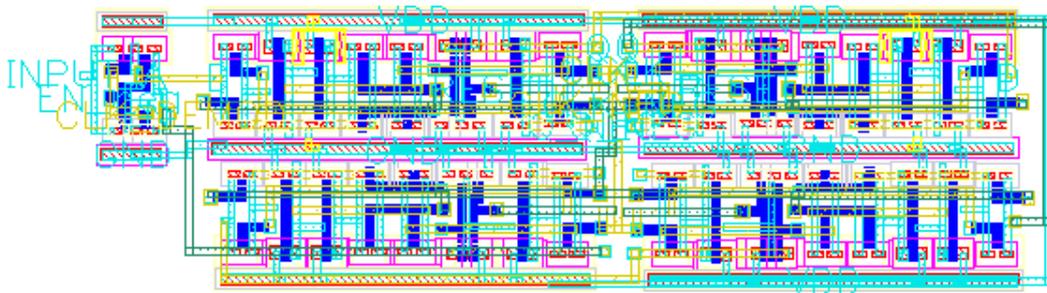


Figure 4.16: Memorisation block layout

4.5.1 Oscillator

Sequential logic is needed to memorize the clock; this is why the D-FF chain is implied. Moreover, as it is well known, any Flip-Flop needs a clock. An on-chip oscillator has been designed in order to be able to save the memorized clock. The oscillator, in Figure 4.17, has been directly derived from the modular block of the frequency multiplier by simply replacing the XOR gate with an inverter and connecting the output to the input of the oscillator to create the loop to make the circuit oscillates. The oscillator output frequency should be around 36 MHz because a frequency at least three times higher is needed to sample the frequency multiplier output (12 MHz). However, the final frequency value generated by the oscillator is chosen around four times the signal to be sampled: between $48\text{-}50\text{ MHz}$. This value, still compliant with the Nyquist criterion, is obtained considering that the shape of the frequency multiplier output is not perfect: an oversampling lets the clock be as expected. This frequency increase will affect the power, making this solution more consuming.

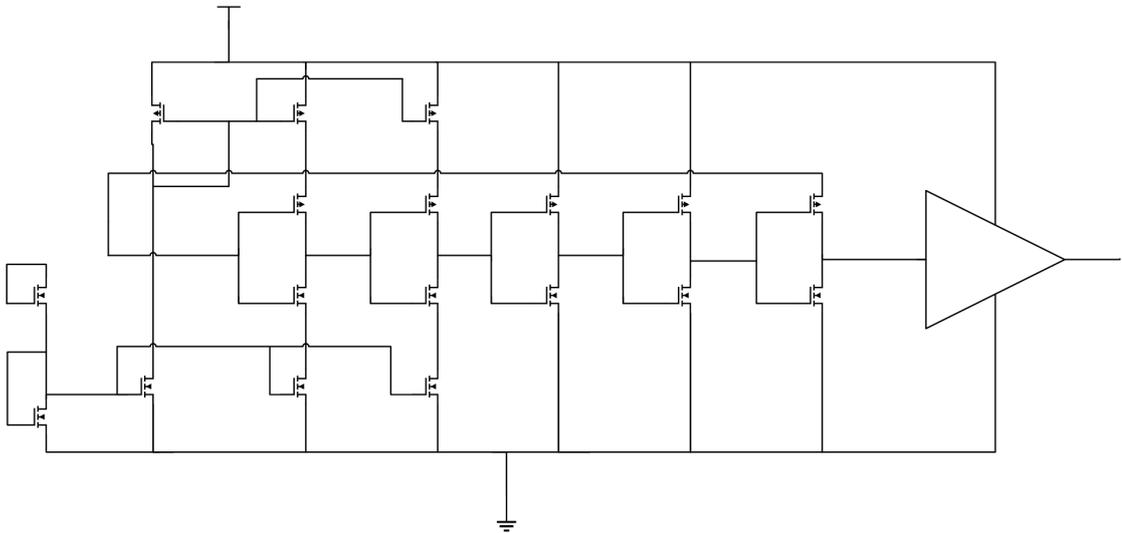


Figure 4.17: On-chip oscillator circuit

The implementation allows to get the wanted frequency implying an R-C less circuit, which is very convenient for area occupation.

4.6 Alternative solution

An alternative solution has been implemented by reducing the power consumption and the occupied area.

4.6.1 Transmission protocol

The idea is to propose a different way to transmit the signal by replacing the previous transmission protocol:

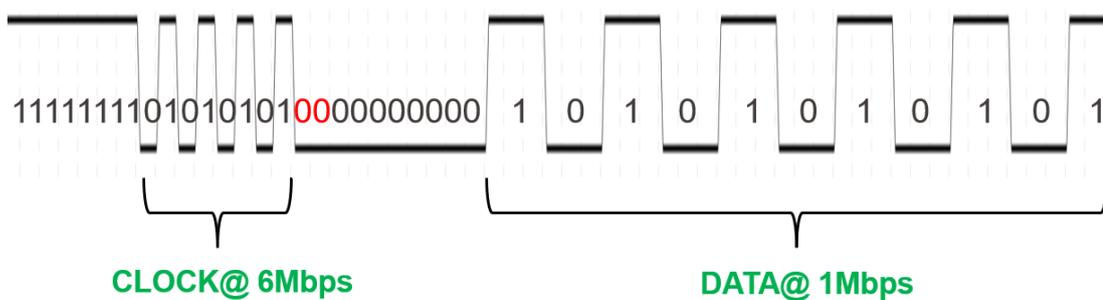


Figure 4.18: Alternative transmission protocol

As before, the first ones are necessary to let the rectifier stabilize, then a higher-frequency train of pulses is presented (6 Mbps). As already explained, the two red zeros underline when the control starts working. In this case, the data is sent, from the beginning, at a lower frequency (1 Mbps) with respect to the clock: there is no need anymore of the frequency multiplier. This type of proceeding can reduce the overall architecture's power and area. The main drawback is related to the type of demodulator taken into account. The upper bound of the designed ASK demodulator is 6 Mbps; thus, this is the higher frequency at which the clock can be transmitted: if the demodulator can be replaced with one with higher performance, the solution can be adaptive.

The architecture of Figure 4.1 is modified as follows:

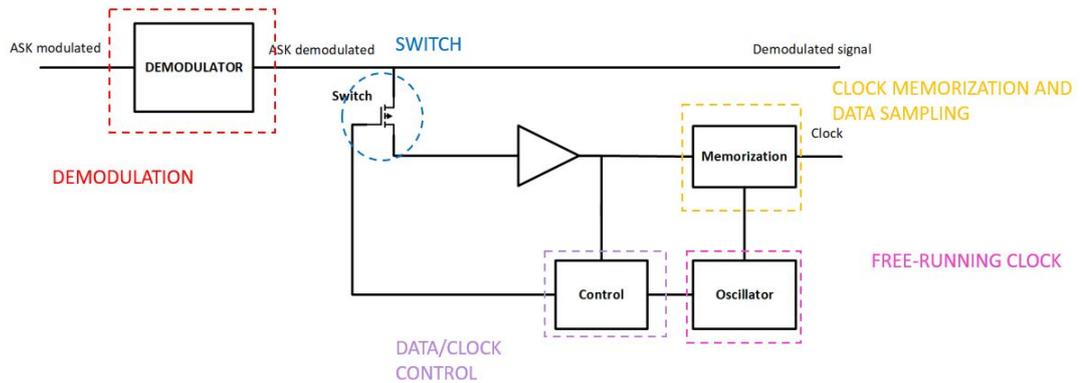


Figure 4.19: Clock Data Recovery architecture

To make this solution as low power as possible, some modification has been performed.

Control block

The D-FF has been used smartly to delete the presence of delay lines in the control block, which are expensive in terms of area and power.

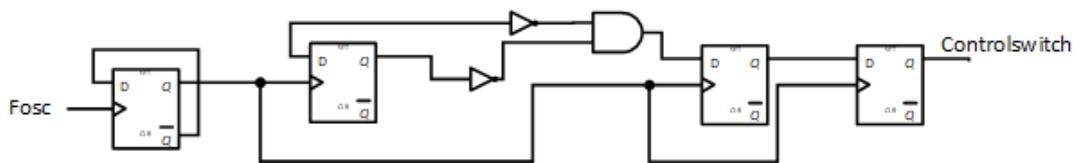


Figure 4.20: Control block CDR

The D-FF takes the role of the delay line. Now that the control signal is created, to be sure that it will remain switched off permanently, it should remain constant to one in order to maintain the switch closed.

The first idea is to introduce a SR register, realised in this way:

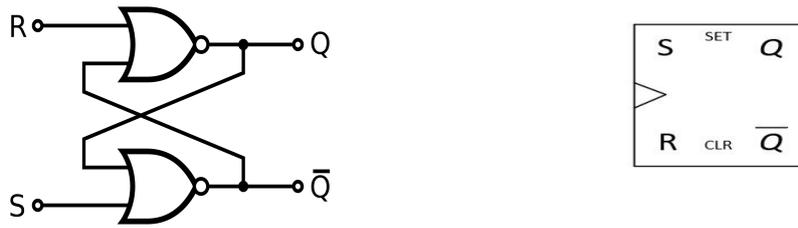


Figure 4.21: SET-RESET register (a) structure (b) symbol. Reprinted from web

The RESET signal is made active only for a short period: a narrow impulse is present before the alternation of zeros and ones. In fact, by following the truth table

SET	RESET	OUT
0	0	No change
0	1	Reset
1	0	Set
1	1	Not allowed

Table 4.1: Truth table SR register

after the reset phase, when the register is resettled, the SR register goes to the memorization mode. The input is reported as output until a modification in the RESET state is not performed again.

A Power On Reset (POR) circuit has been designed to generate this small impulse to resettle the SR register before working in the memorization state. The POR circuit has been reported below: it is a complete stand-alone circuit [39], with the following structure.

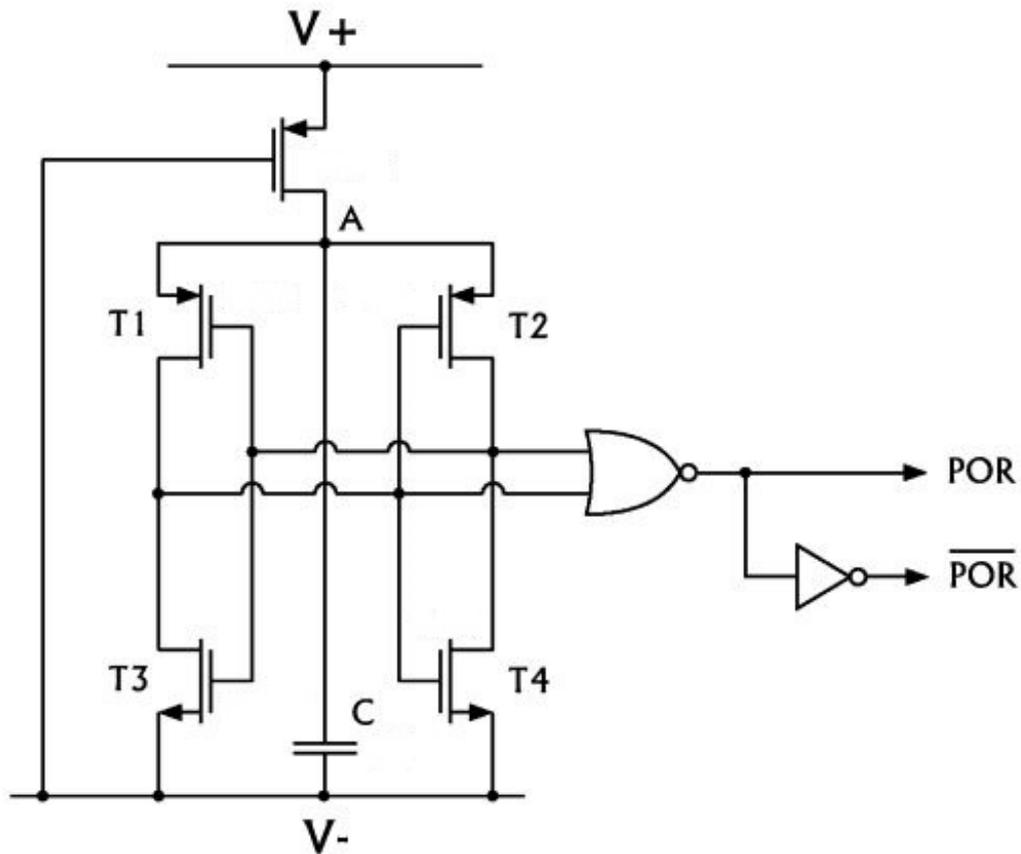


Figure 4.22: Power on reset circuit. Reprinted from [39]

In that way, the control signal is obtained. Then, as shown in Figure 4.20, the SR register has been replaced with a D-FF to have a more modular structure; the POR_NEG is used again to reset the D-FF.

In addition, the D-FF implies a clock generated from the oscillator. The D-FF is used in toggle configuration by connecting the \overline{Q} to D and using the oscillator frequency as a clock. Therefore, this D-FF behaves as a frequency divider whose halved output is used as the clock for all the D-FFs of the control. Ultimately, this block is entirely digital; thus, the power saving will be consistent.

The control block layout is relatively compact, thanks to the reduced number of logic gates. The DFF connected in toggle configuration and the inverter needed to generate the halved clock and its inverted version are not reported to smartly connect them in the final layout.

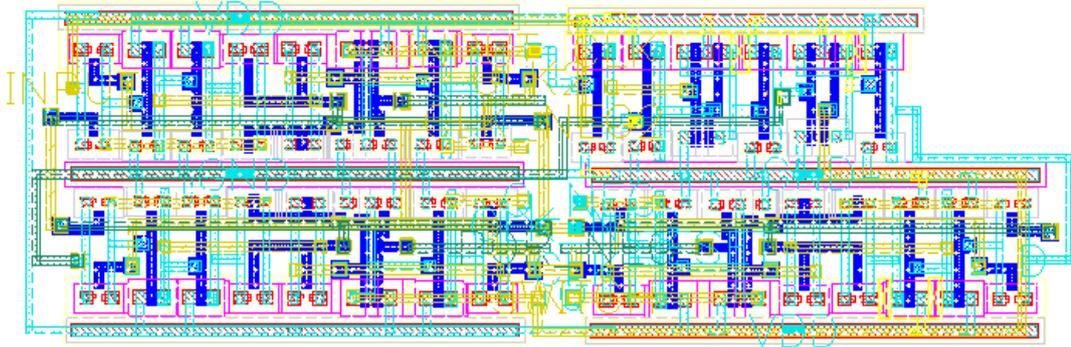


Figure 4.23: Control block layout

In the Figure below the POR layout is reported too.

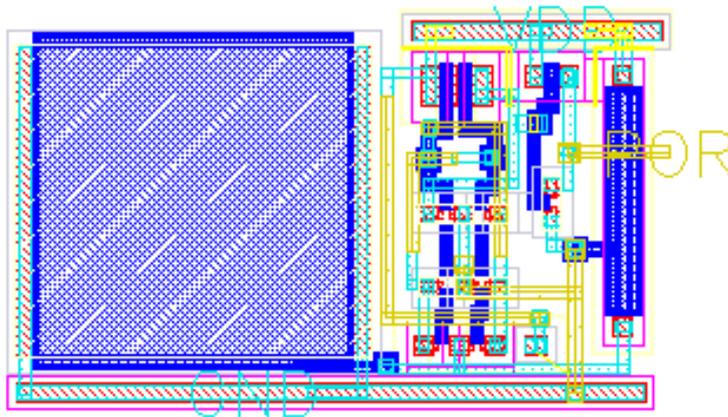


Figure 4.24: POR layout

Oscillator

The state-of-the-art has been widely exploited by referring to Chapter 2. In the first solution, the idea was to use the same delay line structure proposed in the frequency multiplier to optimize the process design. In this case, the low power concept becomes crucial. Several types of oscillators have been implemented in Cadence to analyze all the critical metrics. The oscillating frequency now should be 12 MHz .

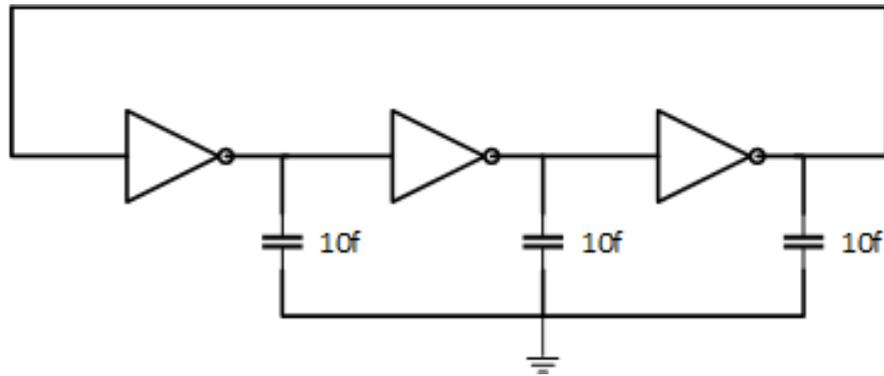


Figure 4.25: Ring oscillator with capacitors

The first implementation is mainly related to the classical Ring Oscillator. A chain of three inverters is sufficient to reach the wanted frequency, introducing in between three capacitors @ 10 fF . Even if this solution looks straightforward, it is not area-efficient due to the presence of the capacitors.

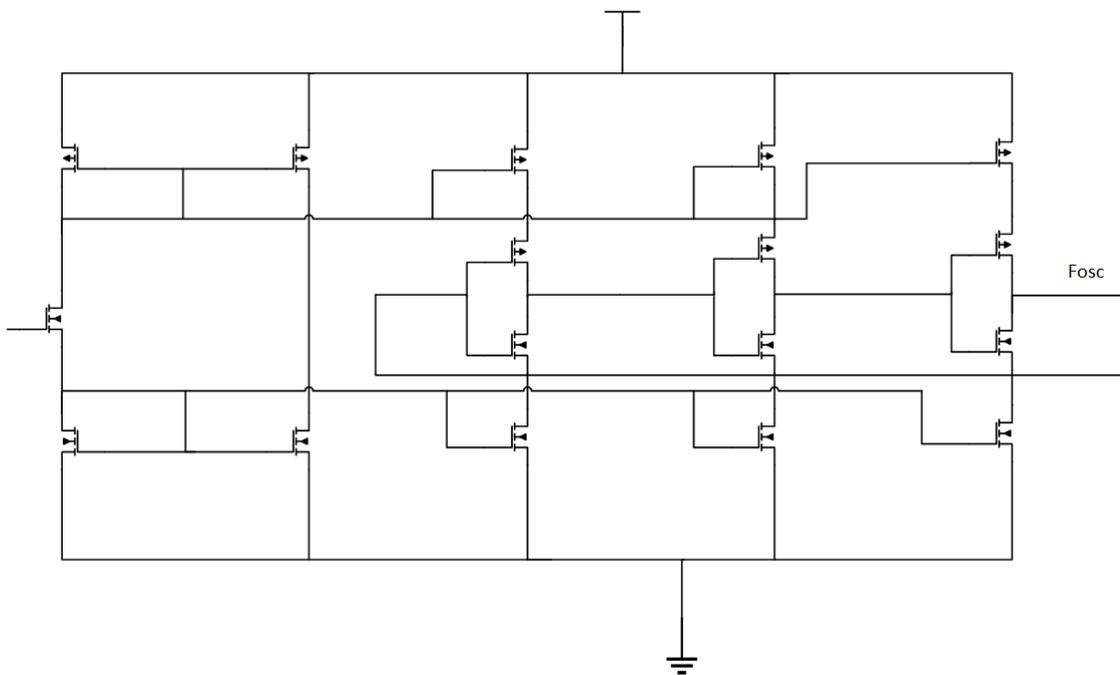


Figure 4.26: 3 stages CS VCO

The second solution shows the same approach as the first one by not requiring capacitors (Figure 4.26). The number of transistors involved is higher, but the area occupied is lower. Moreover, the current starved solution increases the equivalent

resistance of each inverter by adding in series to the inverter transistors, leading to an overall increase in the inverter delay.

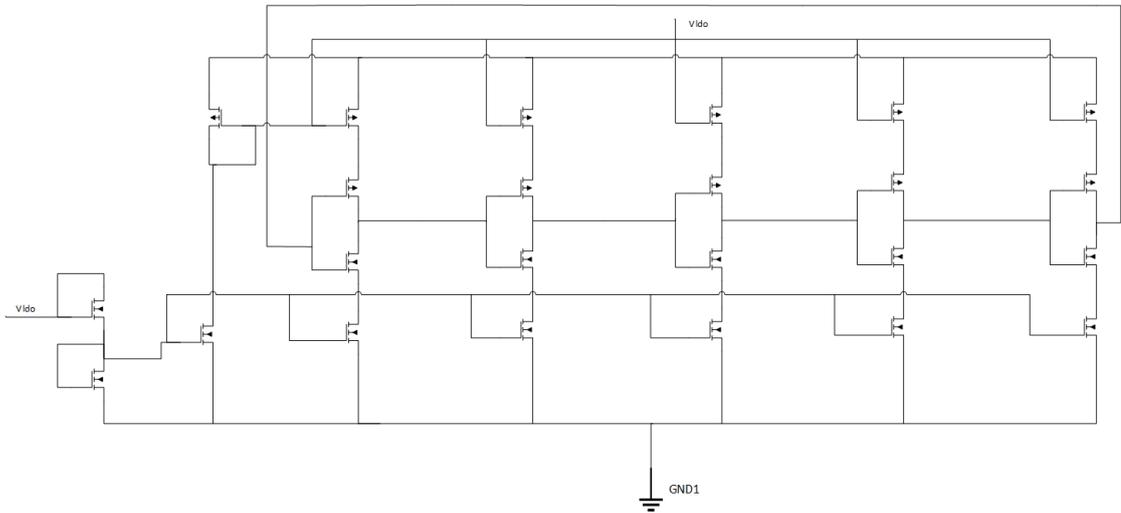


Figure 4.27: 5 stages CS VCO

The circuit above works in the same way as the previous one. The main difference is related to the stages implied. Having more inverters, the delay of each one should be smaller: the dimensions of the transistors are proportionally smaller.

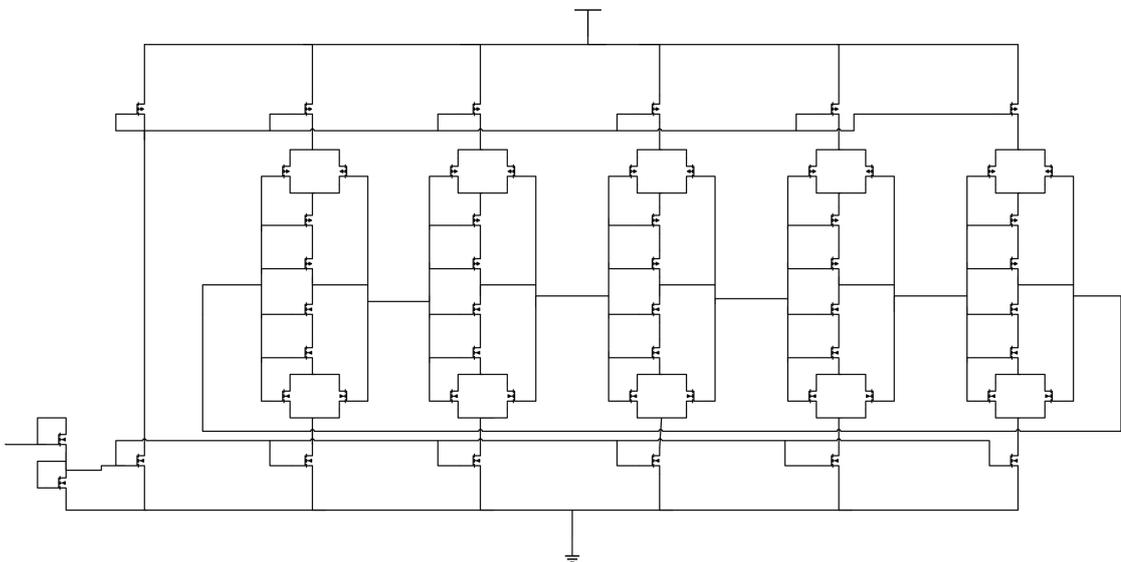


Figure 4.28: 5 stages Starved Sleep VCO

The proposal shown above has been implemented to test if a design based on the

exploitation jointly of the stack effect and sleep mode, with a higher number of transistors, would bring real advantages in terms of power. Finally, the table below compares the characteristics of all the presented designs.

Table 4.2: Comparison between the implemented oscillator

Oscillator type	#transistors and C elements	Power [mW]	Phase noise [dBc/Hz]
3 stages RO	6 (3 Cap)	21.8	-117.3
3 stages CS VCO	17	0.00143	-101.9
5 stages CS VCO	24	0.00148	-99.64
5 stages Starved Sleep VCO	54	0.00173	-105.9

All the solutions are comparable in terms of phase noise. The first one is wholly discarded due to its power consumption. The CS VCO's three stages-based implementation is chosen for its reduced number of transistors and low power consumption.

Moreover, a circuit based on standard cells has been considered to let the oscillation start. The inverter chain combined with the AND gate generates the impulse that feeds only once the gate and the drain of the nMOS, making it active only once to let the oscillation starts.

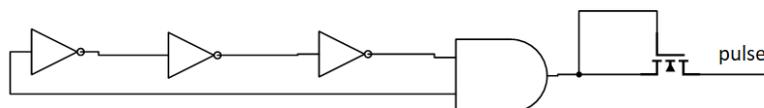


Figure 4.29: Oscillator circuit power on

The on-chip oscillator layout has been designed as compact as possible, even if the activation logic occupies a small area. The clock traces have been connected to have a more robust signal.

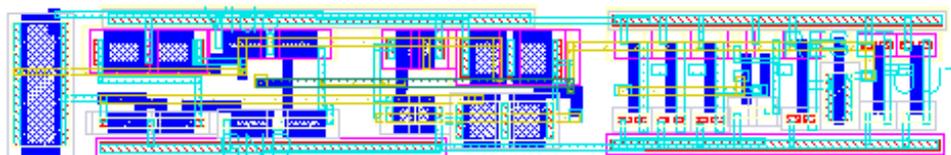


Figure 4.30: On-chip oscillator layout

The block that is needed to let the oscillation start can be easily identified as the compact right-most block.

Chapter 5

Complete proposed architecture

5.1 Pre-layout simulations

The pre-layout simulations of both the implementations have been done to make a direct comparison also for what concern the waveform looks.

The first simulation shows the demodulated signal (upper trace) as the frequency multiplier input, where both the clock and the data have the same data rate. Then, its output (second trace) results in the demodulated signal at a 4-times higher frequency. The last signal takes into account the memorized clock.

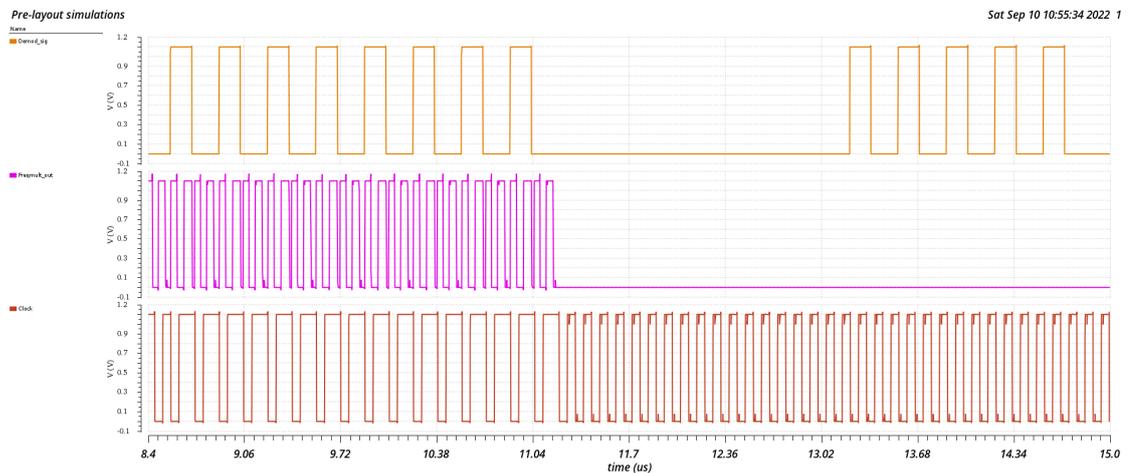


Figure 5.1: Frequency multiplier solution

The image below shows the simulation in the solution without a frequency multiplier.

The demodulated signal is characterized by the train of pulses at a higher frequency with respect to the data one. The second trace highlights the output of the switch, while the third one is the switch control signal: once the two zeros that activate the clock are detected, the switch is permanently closed. The last trace reports the memorized clock at a lower frequency with respect to the previous case.

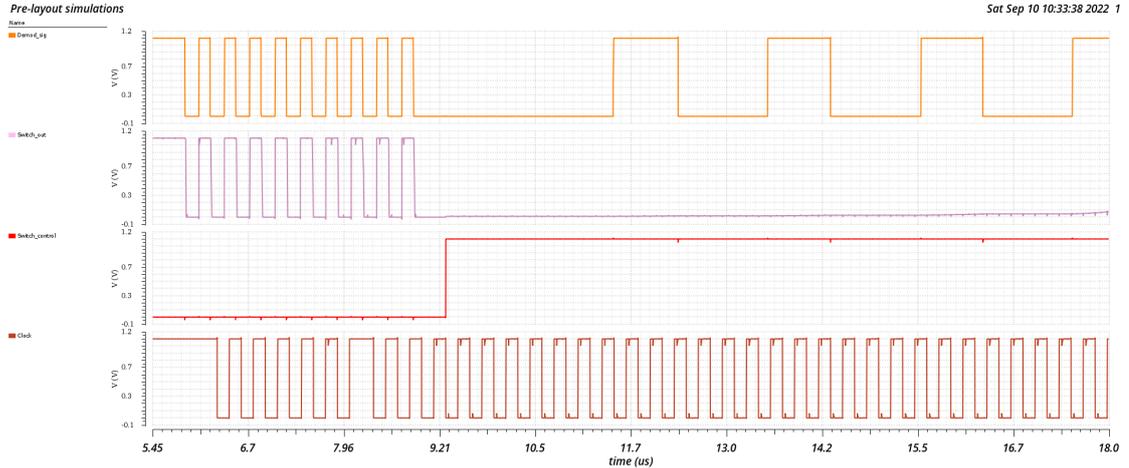


Figure 5.2: w/o Frequency multiplier solution

No further simulations, mainly related to the demodulation operation, have been considered necessary considering what has already been shown in Chapter 3.

5.2 Comparison between two proposed architecture

The two architectures show consistent differences: the main one is the frequency multiplier block. This element affects the whole system's speed, power, and area. The first solution has one main advantage related to the opportunity of transmitting data at the maximum data rate allowed by the demodulator design. In the second case, this lead is ultimately sacrificed with low-power, and lower-area added advantage due to the frequency multiplier absence.

5.2.1 Power and Area analysis

The principal analysis should be focused on the two most important requirements: power and area. The reduction of both these two parameters is critical. Moreover, the average power of all the blocks has been evaluated to underline the differences between the two solutions.

- **Demodulator:** this part of the implementation is the most consuming since high power is needed to transform an analog signal, as the ASK modulated input, to a digital one. Moreover, the envelope voltage has been defined as a starting value that should not be modified and is higher than the power supply, affecting the total power consumption.
- **On-chip oscillator:** this block is the highest in terms of power consumption after the demodulator. It generates the clock that should be continuously active, doing a transition for every period. Moreover, in the frequency multiplier case, it consumes proportionally to frequency; this is why the value increases a lot in the frequency multiplier case.
- **Memorization and Control:** these circuits are entirely digital; thus, the power is affected by the frequency of the clock: as it is widely known, the higher the frequency, the higher the dynamic power consumption. For this reason, a higher power value is observed in the frequency multiplier solution, where a higher clock frequency is needed. In the frequency multiplier case, the control block consumption is almost entirely related to the delay line.
- **Frequency multiplier:** this block is out of range with respect to the other for power and area. Its characteristics, even if it allows exploiting the demodulator upper bound at its maximum, are pretty expensive in terms of low-power and area meanings.

By comparing pie charts and the corresponding numbers, it is evident that the second solution is more compliant with the project specifications. A table has been filled to compare the provided solution to the literature systems.

One thing that should be highlighted in the frequency multiplier case is that the area of the delay, the on-chip oscillator, and the frequency multiplier block have been evaluated approximately; no layout design was performed with respect to the case without frequency multiplier, chosen as the final CDR architecture.

Block	Power [μW]
Demodulator	9.406
Memorization	0.285
Control	0.307
On-chip oscillator	1.776
Total	11.774

Block	Power [μW]
Demodulator	9.92
Memorization	1.168
Control	9.3
On-chip oscillator	7.908
Frequency multiplier	25.56
Total	53.856

Table 5.1: Power (a) without and (b) with frequency multiplier in pre-layout

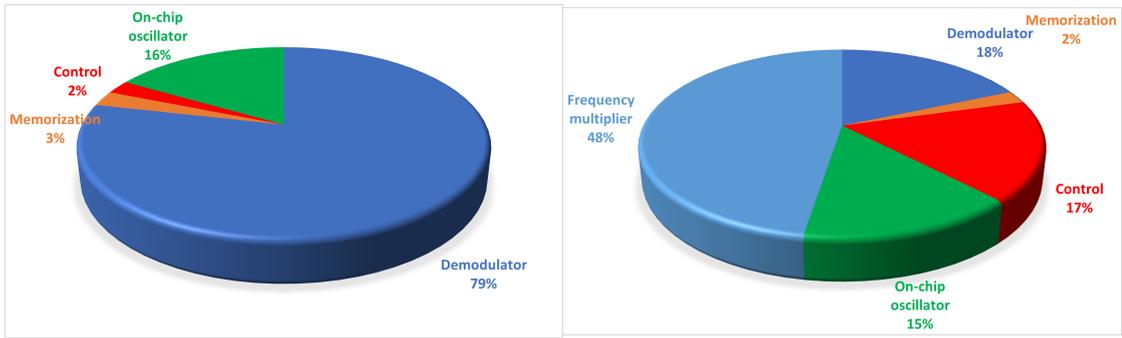


Figure 5.3: Pre-layout power (a) without and (b) with frequency multiplier

Block	Area [μm^2]
Demodulator	210
Memorization	386.73
Control	349.23
On-chip oscillator	223.14
Total	1169.1

Block	Area [μm^2]
Demodulator	210
Memorization	386.73
Control	1200
On-chip oscillator	180.505
Frequency multiplier	780
Total	2757.235

Table 5.2: Area (a) without and (b) with frequency multiplier in pre-layout

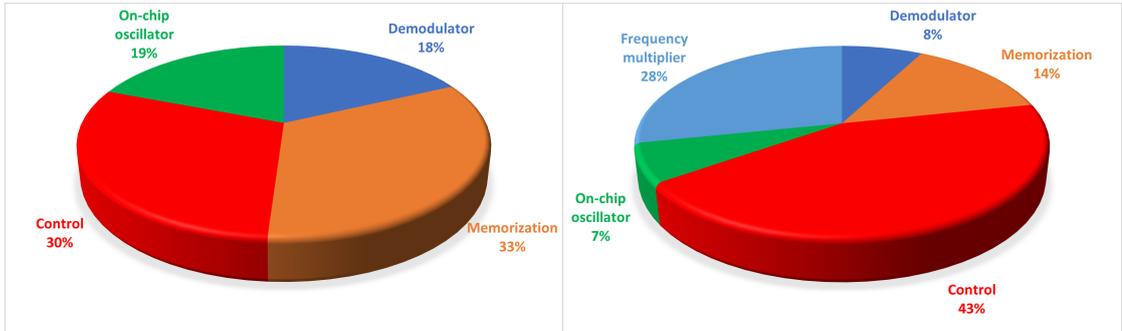


Figure 5.4: Pre-layout area (a) without and (b) with frequency multiplier

The power consumption in the frequency multiplier case looks five times higher than the other one. All the blocks show a significant rise in consumption, linked to the higher employed frequency and the higher number of transistors. The area is

more than doubled. The demodulator and the memorization block are unchanged. However, the control shows a substantial increase mainly linked to the delay line, comprised of five modular blocks implied for the frequency multiplier design. The oscillator is slightly smaller.

5.3 Monte-Carlo analysis

The Monte Carlo analysis is performed on the frequency value of the oscillator that is generating the clock to memorize the one used for the sampling.

Three different versions of the same oscillator design, reported in Figure 4.26, have been done:

- the first one by connecting the pMOS body to the source and the nMOS one to drain;
- the second one connecting the pMOS and nMOS body to Vdd and GND respectively with *minimum transistors dimension* to get the desired frequency value;
- the third one connecting the pMOS and nMOS body to Vdd and GND respectively with *bigger transistors dimension*;

Moreover, the Monte Carlo analysis has been performed with $N=2000$ samples and a standard deviation $\sigma=3$. The three simulation results have been reported below.

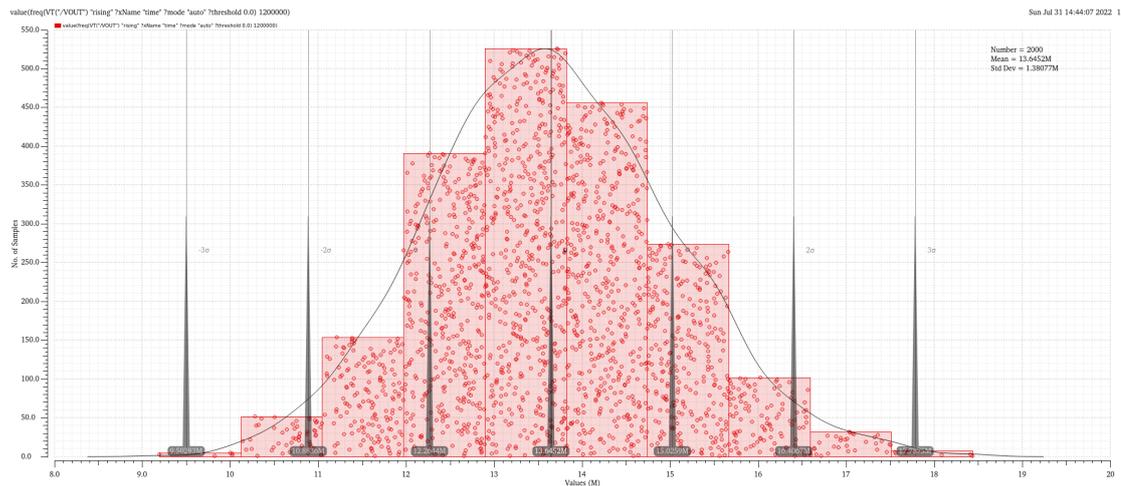


Figure 5.5: Monte Carlo first oscillator

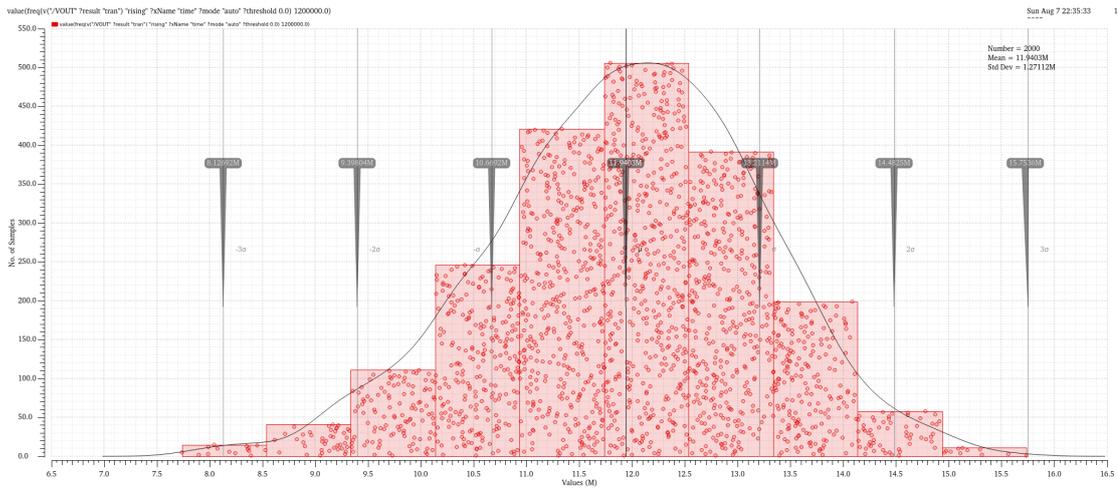


Figure 5.6: Monte Carlo small dimension oscillator

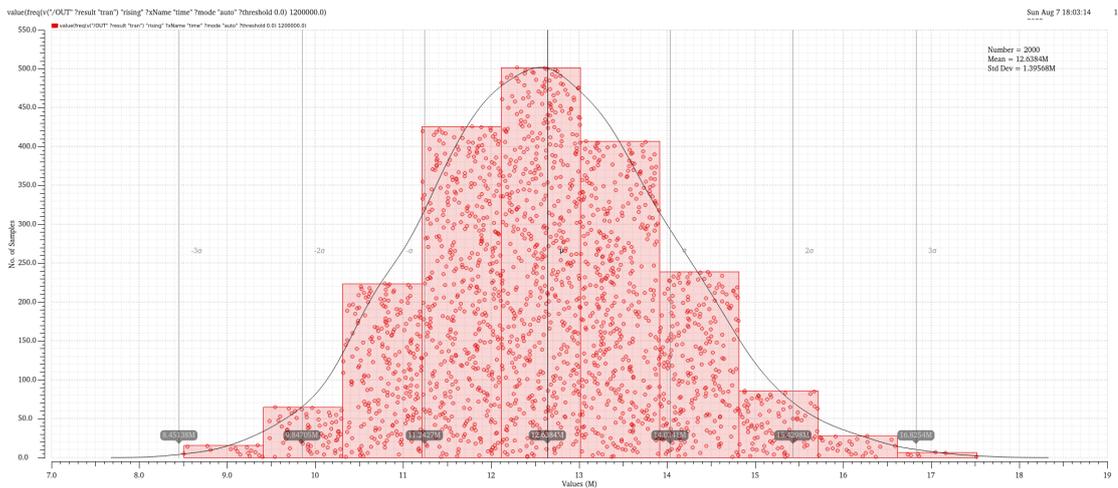


Figure 5.7: Monte Carlo big dimension oscillator

The results show a Gaussian shape, as expected. The table below has been reported in order to make a further comparison. Considering that the target frequency is around 12 MHz , the mean and standard deviation values have been taken into account. As it can be observed from the results, the discard between μ and σ is small: taking into consideration the three cases, the percentages are 10.10%, 10.64%, and 11%, respectively. These no optimal values show how is difficult to find a constant oscillation frequency.

By the way, even if the standard deviation is worse than one of the other solutions, the chosen oscillator is the third one: and although the transistors dimension is

more significant, the effective area of the second and the third solution is almost comparable from a layout point of view. In contrast, the second solution shows a higher power consumption. For what concerns the first implementation, connecting the body, not to the lowest/highest voltage of the circuit, shows a higher risk of the latch-up occurrence and the introduction of a detached/integrated body in the layout phase, occupying more silicon area.

Oscillator type	Mean (μ) [MHz]	Standard deviation (σ) [MHz]
Body source-drain connected	13.65	1.38
Smaller transistors dimension	11.94	1.27
Bigger transistors dimension	12.64	1.40

Table 5.3: Monte Carlo results of proposed oscillators

Once the third solution has been chosen, two tests have been performed with an oscillator frequency equal to the two extreme values of the Gaussian tails: -3σ (8.45 MHz) and 3σ (16.83 MHz). Obviously, for the lower bound, the system does not work correctly anymore because the duty cycle of the memorized clock should not be wholly recovered to 50%. Conversely, everything still works as it should for the upper bound.

5.4 Phase variation robustness

Some tests have been performed to validate the memorization block's robustness. The D-FF chain design is crucial: simply thinking that a slight change in the input phase can create a severe problem in clock memorization. Before the validation, consideration related to the parameters at stake has been considered.

The following relation strictly connects the frequency of memorized clock and the oscillator one:

$$f_{osc} = 2^{N-2} \cdot f_{clock} \quad (5.1)$$

where N considers the D-FF numbers' composing the chain. Thus, two parameters should be fixed, and the third derived through the expression above. In the final solution, the one without a frequency multiplier, the clock frequency is 3 MHz, and the oscillator one is around 12 MHz. At the same time, the number of stages is equal to 4.

Moreover, the oscillator frequency can be maintained by constantly changing the number of D-FFs according to the clock frequency. Figure 5.8 shows how the memorization block works: the demodulated signal is shifted by half a clock cycle

by each D-FF; thus, in this case, at the chain output, the signal will be delayed by two clock cycles, reported as input, in order to not have a conflict of values. Thus, at the input, there is the bit whose value is one, the one reported from the output will be one for the same instant of time.

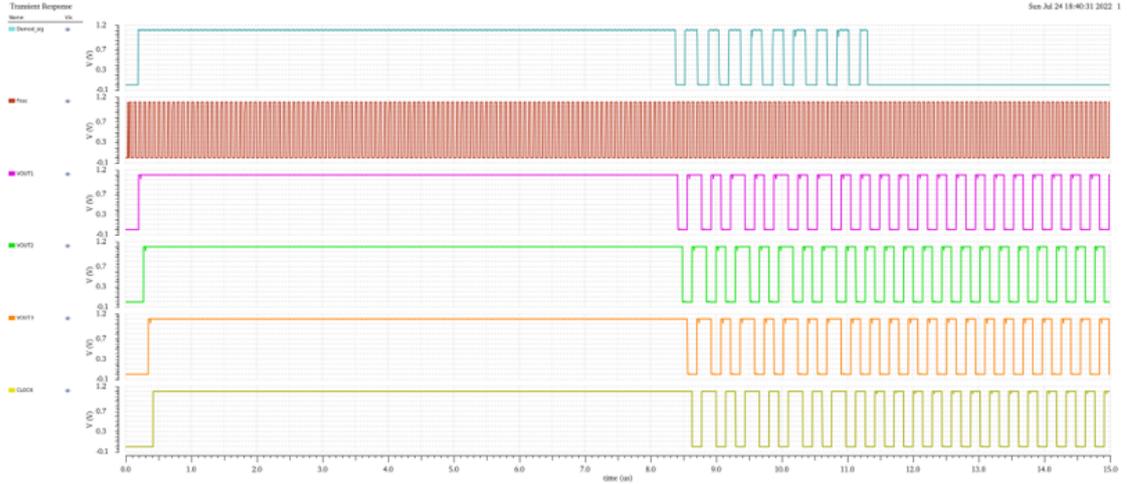


Figure 5.8: Clock memorization block signals

After the functionality clarification, an analysis to check the phase robustness of this structure has been done. The ideal case is when the demodulated signal at the input of the memorization block presents a duty cycle perfectly equal to 50%. If this is not the case, this structure should be able to fix the problem and reconstruct a perfect clock signal. To test it, a signal with a specific phase difference has been generated using a Matlab script, following this expression:

$$DC \in [DC^{nom} - \alpha \cdot \Delta, DC^{nom} + \alpha \cdot \Delta](\%) \quad (5.2)$$

where $DC^{nom} = 50\%$ and $\alpha \in [0,1]$.

The generated signal is created by concatenation of period composed by 1 and 0; for each period the duty cycle (DC) is changed accordingly to the formula above: the value of α is randomly chosen for each iteration, while the user settles Δ , when the window in Figure 5.9 appears.

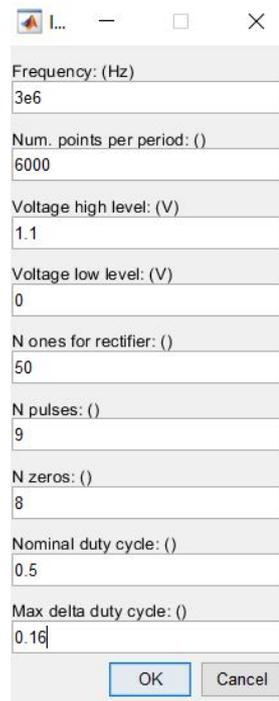


Figure 5.9: Parameters settled by the user

For clarity, an example is proposed: if Δ is 16%, the DC range will be between 36 and 44 %. The Figure 5.10 shows a DC of 43, 58, and 58 % again in the first, second, and third periods, respectively.

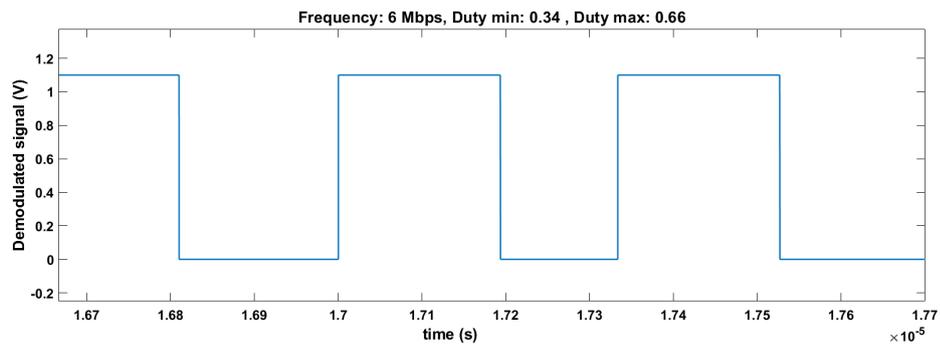


Figure 5.10: Phase robustness analysis with $\Delta = 16\%$

Moreover, the screenshot below shows that, after a period of stabilization, the structure can memorize continuously the clock with the correct duty cycle.

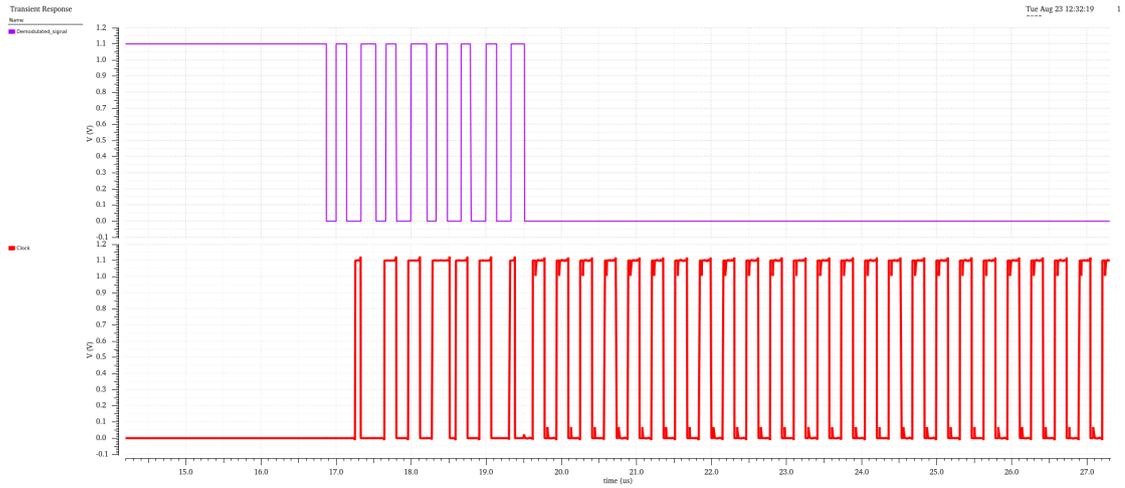


Figure 5.11: Simulation with $\Delta = 16\%$

The phase robustness is also mainly related to the oscillator output, which samples the demodulated signal constantly to memorize it. This test is performed with the oscillator with a more significant transistors dimension and the one with a smaller dimension: for both of them, the two working upper bounds have been defined. The first can reconstruct the signal with a maximum $\Delta = 16\%$, while the second one up to 25%. Even if the oscillator with smaller transistors dimension works appropriately until the higher value of phase variation, being its upper bound unmeaningful, the first solution has been chosen for the reasons already discussed above.

5.5 Bit Error Rate(BER) analysis

The Bit Error Rate (BER) analysis is performed to determine how accurate the transmission of the demodulated data is. By definition, the BER is expressed as the incorrect number of bits in a serial stream communication, and it is expressed as follows:

$$BER_{\%} = \frac{N_{err}}{N_{bits}} \quad (5.3)$$

To study how many bits have been corrupted through the transmission, logic has been postponed to the final CDR architecture, as reported in Figure 5.12.

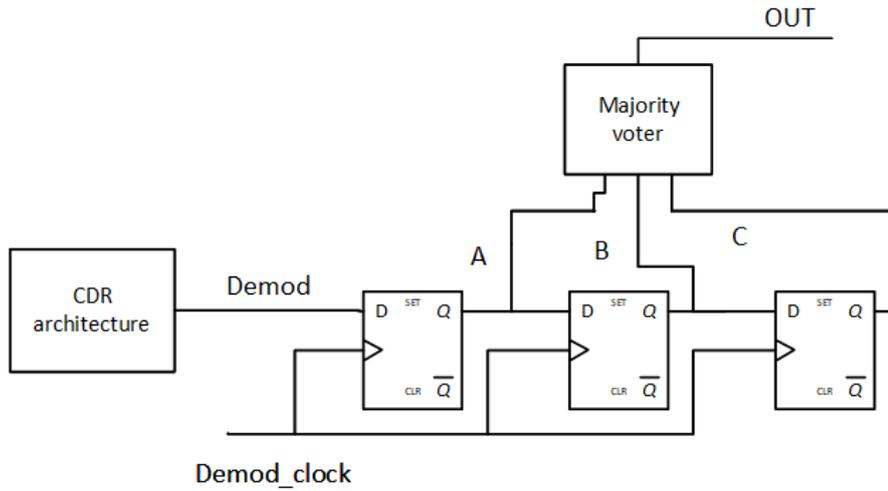


Figure 5.12: BER logic scheme

The demodulated data is shifted by a D-FFs chain, which clock is the one generated by the on-chip oscillator.

The output of each D-FF is the input of the logic in Figure 5.13, that acts as a *Majority voter* (MV)

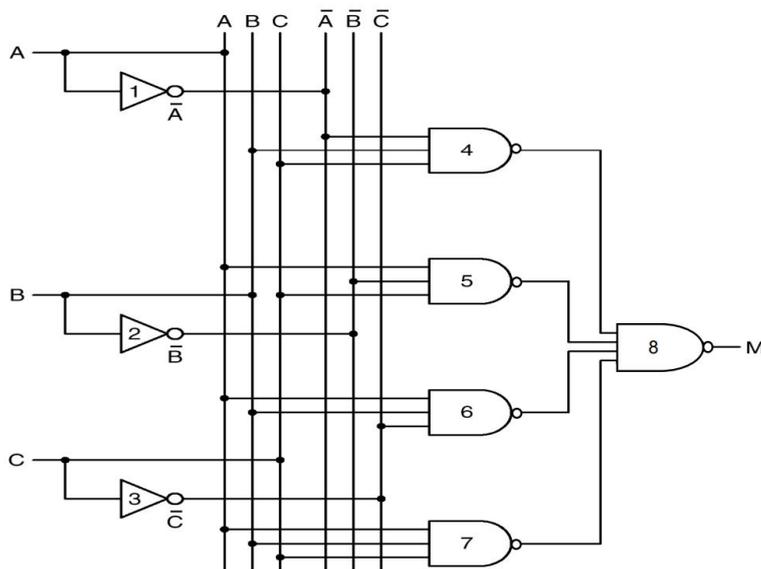


Figure 5.13: Majority voter. Reprinted from web

with the following boolean function and truth table.

$$M = \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC$$

A	B	C	M
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Figure 5.14: Boolean function and truth table of the Majority Voter

The truth table shows that only when two ones occur the output will be one; otherwise, the result will be zero. The logic output should be compared with the digital version of the original ASK signal. Figure 5.15 shows the traces of the different signals: in red, the ASK modulated signal is reported, followed by its demodulated version; then, the A, B, and C waveforms represent the shifted version of one-third of the bit of the demodulated signal. These three signals are the input of the majority voter. The signal OUT shows the result of the logic, as already figured out in Figure 5.12.

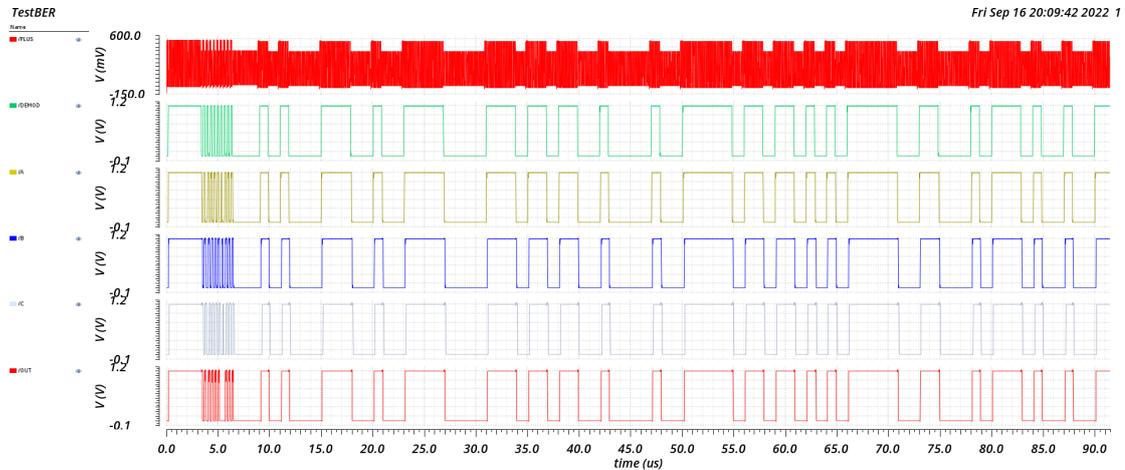


Figure 5.15: Zoom of a BER test with 1000 bits

The mathematical computation of the BER is performed through an algorithm implemented in Matlab. The two signals have been divided bit by bit. The difference between the original and the output of the logic is performed bit by bit, to find out when the two signals would be different. In this way, the number of corrupted

bits will be extracted.

The result shows that on 1000 transmitted bits, only three resulted corrupted, thus:

$$BER_{\%} = \frac{N_{err}}{N_{bits}} = 3 \cdot 10^{-3} \quad (5.4)$$

5.6 Layout

The layout of each block composing the CDR architecture has been done separately, and then, after the post-layout simulations of every single block, the entire layout has been provided. For each block, the following steps have been chronically applied:

1. **Pen&Paper draft:** the layout has been firstly roughly sketched using the *stick diagram* in order to better position the transistors to occupy the minimum area and put in common the maximum number of terminals between near transistors;
2. **Layout of the single components:** an approach based on the "from micro to the macro" policy has been followed, meaning that the normal cells and components layout has been done before to create the bigger blocks.
3. **Design Rule Check (DRC):** the layout design is verified according to the fabrication rules; its part should be compliant with some specific dimensions, and some distances have to be respected;
4. **Layout Versus Schematic (LVS):** when the layout is DRC-clean, the check on the correspondence between nets names and labels of the device should be verified too to get the LVS-clean.

The layout of the entire CDR architecture is realized to get the final area value. To reduce the occupied area, some V_{DD} and GND stripes have been shared, following one of the classical layout approach related to the standard cell, as reported in Figure 5.16.

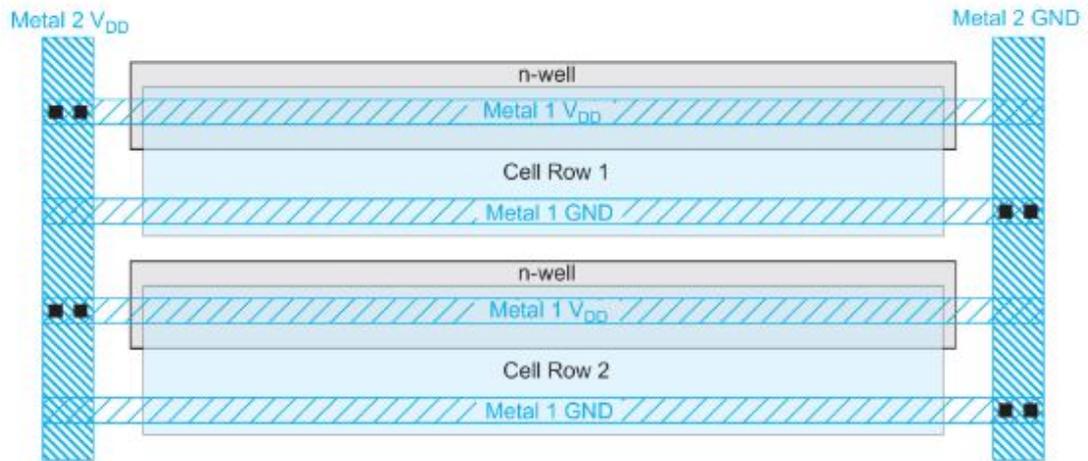


Figure 5.16: Standard cell layout. Reprinted from web

The red rectangles have been introduced to highlight the different parts. Only METAL 1, 2 and 3 have been used to made the connections.

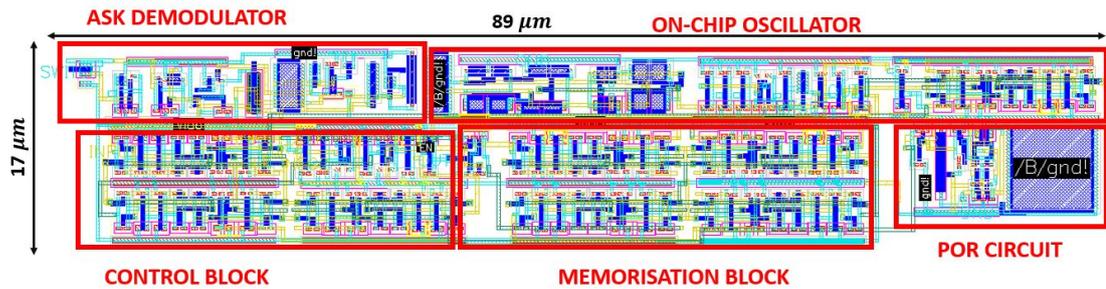


Figure 5.17: Layout of the CDR architecture

As it can be displayed from the Figure above, the final layout looks really compact and it finally occupies an area of $17 \times 89 \mu m^2$.

5.7 Post-layout simulations

After the layout design, the post-layout simulations have been done to test the architecture's correctness. Firstly, some problems related to the demodulator have been met, and as shown in Figure 5.18, the duty cycle of 50% was not recovered. To highlight this, the figure below shows the difference between the pre-and post-layout waveforms.

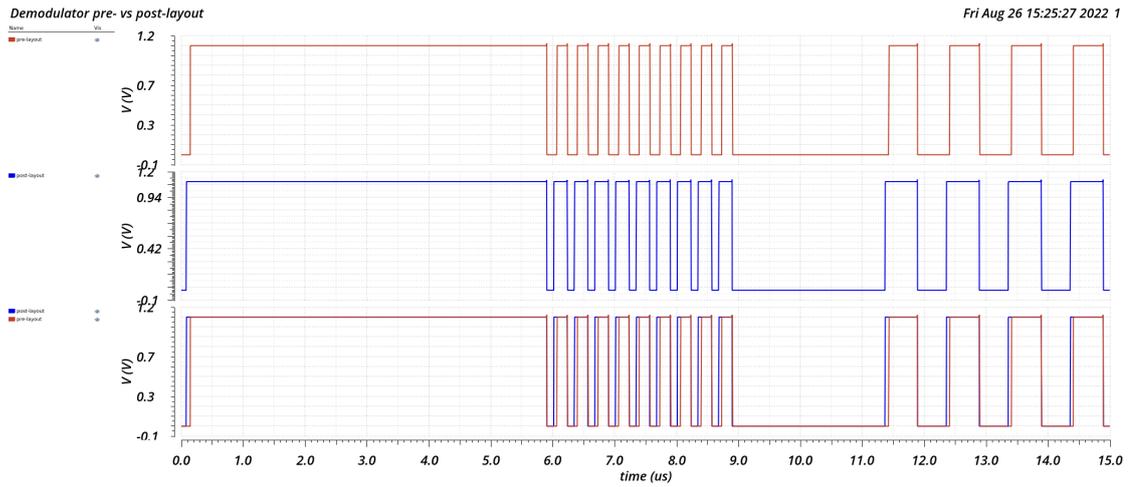


Figure 5.18: Demodulator pre- vs post-layout simulations

In Figure 5.19, it has been pointed out how problems also involve the switch; thus, the pMOS dimension should be adjusted to get the desired signal.

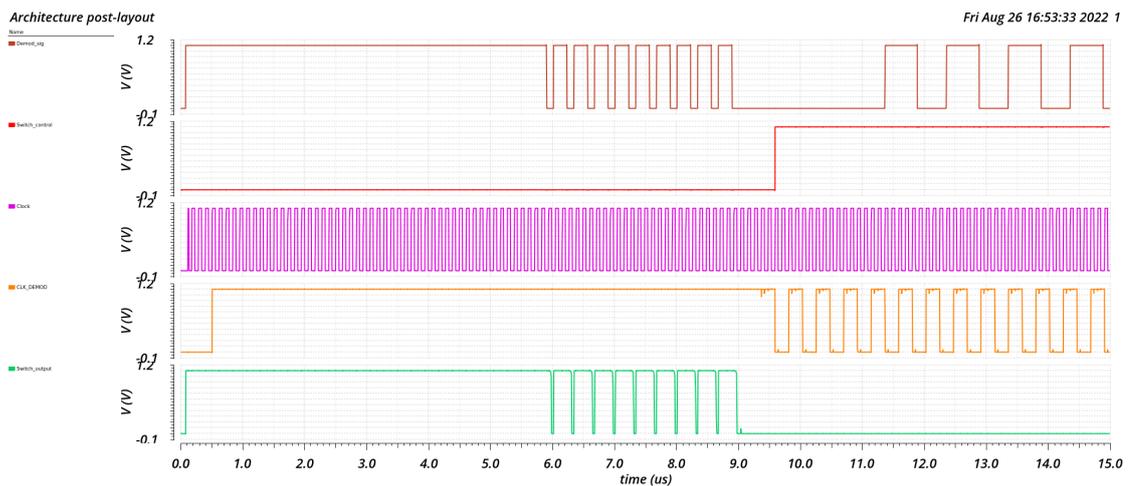


Figure 5.19: CDR architecture post-layout simulations

Moreover, a simulation, reported in Figure 5.20, related to the oscillator frequency is performed, too, because the recovered and memorized clock did not have the wanted frequency. In the post-layout simulations, the oscillator reported a reduced output frequency from 12 MHz to 8-9 MHz , the lower bound found in the Monte Carlo analysis for which the duty cycle and the frequency of the demodulated clock do not work correctly anymore.

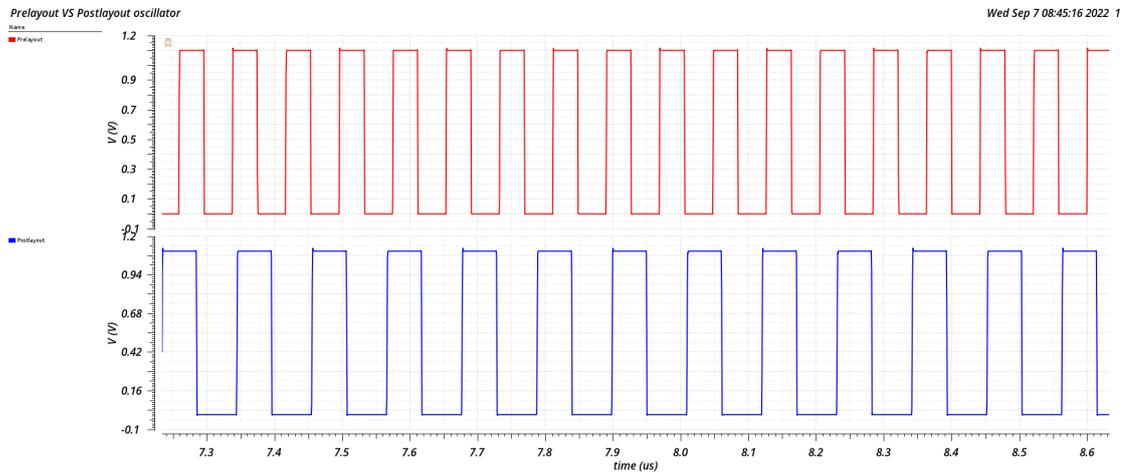


Figure 5.20: Oscillator pre- vs post-layout simulations

Once the problems have been detected, the perfect signal shape of the demodulated signal has been restored, changing the transistor dimensions of the demodulator and the switch.

For what concerns the oscillator, an oscillator working @16-17 MHz is implemented in the pre-layout phase to reach the wanted value of precisely 12 MHz in the post-layout simulations phase. Thus, the final simulation with all the correct signal shapes is reported below.

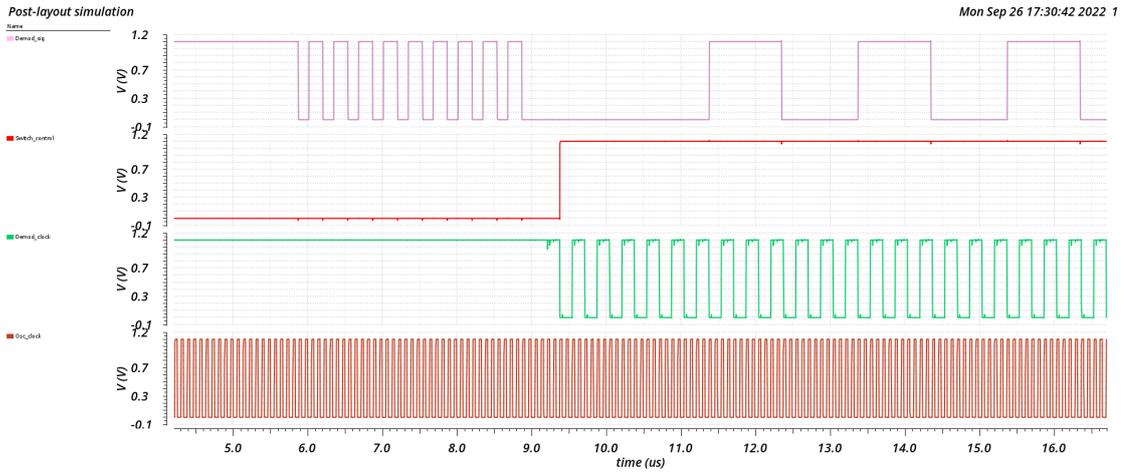


Figure 5.21: Correct CDR architecture post-layout simulation

Naturally, the change in the dimensions of the transistors affects the power consumption and the area values of some blocks. The computation has been done again to underline the difference between the pre-and post-layout simulations.

Block	Power [μW]	Block	Area [μm^2]
Demodulator	11.57	Demodulator	210
Memorization	0.0079	Memorization	386.73
Control	0.627	Control	349.23
On-chip oscillator	2.81	On-chip oscillator	200
Total	15.0149	Total	1146

Table 5.5: Post-layout (a) power and (b) area analysis without frequency multiplier

In the table, it has been highlighted how the demodulator and the oscillator power consumption are increased with respect to the ones found in the pre-layout case. Since the demodulator dimensions have been changed, this modification is also reflected in power. On the other side, the oscillator consumption is higher because the oscillation frequency has been increased and thus the power consequently. In contrast, the memorization block shows a reduction. Moreover, the control block presents a slight increase probably linked to the fact that, in this case, the power-on reset consumption has been taken into account. In contrast, in the pre-layout analysis, this has not been considered. For what concerns the area, results are almost unchanged: the oscillator dimension is reduced due to the variation of the transistors dimension to obtain a higher frequency. The total area value is evaluated by simply summing up the ones of all the blocks; thus, the whole layout

area will consider how the different blocks will be connected, which will be different from the value provided in the table.

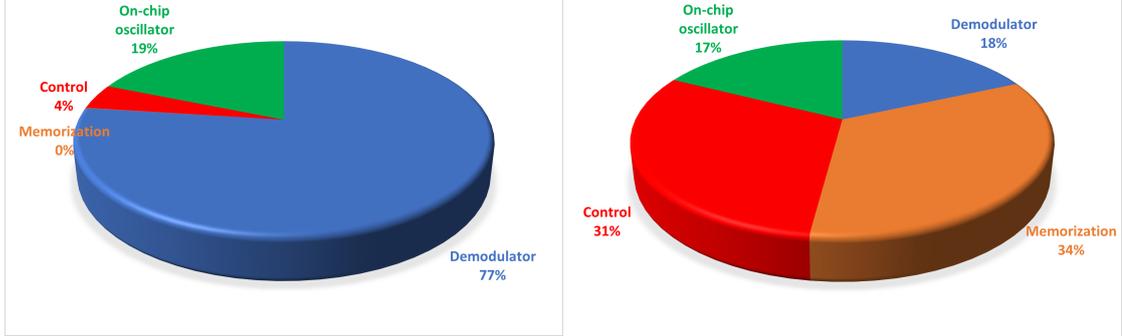


Figure 5.22: Post-layout (a) power and (b) area analysis without frequency multiplier

Something interesting can be related to the fact that the blocks with higher power consumption show a smaller area and vice-versa: this is meaningful by considering that area and power are two parameters for which a trade-off has to be always found.

Moreover, the noise analysis has been re-done, reporting the following changes:

1. **Total Input Referred Noise:**

$$SNR = 20 \log \left(\frac{1.1V}{16.195mV} \right) = 36.64dB \quad (5.5)$$

2. **Total Output Voltage:**

$$SNR = 20 \log \left(\frac{1.1V}{242.04\mu W} \right) = 73.15dB \quad (5.6)$$

The obtained values do not show a quite great change with respect to the previous case by considering as signal amplitude the full sweep of the demodulated signal, from 0 to 1.1 V.

5.8 Comparison with previous works

Table below reports some information about the CDR architecture proposed by the literature. The main focus is related to the ASK system, even if some FSK architectures have been considered. The most critical aspects are area and power. To compare all the solutions, two FOM have been taken into account directly got from [17] and [40]: the first one considers the DR and the carrier frequency, while the second one takes into account power and area with respect to DR. As displayed from the results, the proposed solution is very competitive in low-power and low-area applications, getting better performances than the other implementations.

Parameter	Ref.[41]	Ref.[42]	Ref.[43]	Ref.[44]	Ref.[45]	Ref.[46]	Our work
f_c (MHz)	1.5	10	1	13.56	-	5/10	433.92
Mod. tech.	ASK-PPM	ASK-PWM	ASK-PPM	ASK	FSK	FSK	ASK
Method	Charge pump	Monostable circuit	Charge pump	PLL	Digital Logic	Digital logic	Adaptable transm.
Tech. (μm)	0.25	0.18	0.25	0.18	0.13	0.13	0.18
DR (Mbps)	0.0455	0.5	0.01976	0.106	0.1	4	6
Area (μm^2)	-	-	3.19×10^7	273600	1.7×10^7	2.9×10^7	1513
Power (μW)	31	29.52	100	900	0.05	380	15.0149
FOM	-	-	6.33×10^{-6}	4.41×10^{-4}	0.118	3.63×10^{-4}	255.75
FOM	0.111	0.292	0.073	0.094	-	1.17	0.436

^aFOM taken from [17] = $\frac{Data\ rate}{Power \cdot Area} \left(\frac{kbit/s}{mW \mu m^2} \right)$

^bFOM taken from [40] = $\left(\frac{Data\ rate^2}{f_c} \right)^{\frac{1}{3}} \left(\frac{Mbps^2}{MHz} \right)^{\frac{1}{3}}$

Table 5.6: Comparison with other works

Chapter 6

Conclusion and Future developments

Some years ago, the necessity to have chips with a smaller silicon area grew and the demand for low-power devices increased hand in hand.

The world of implantable devices was and is not, nowadays, exempt from this demand. Indeed, this innovative path can be challenging, especially for this type of application.

This project is a new step on miniaturized and low-power devices' long and complicated path. This thesis leads to the realization of two different Clock Data Recovery (CDR) architectures that will be embedded in a CMOS microelectrode array implanted to restore vision in the brain of a blind patient. Both these systems use an ASK demodulator able to operate up to 6 *Mbps* with a modulation index of 15%. The input of the CDR architecture is an ASK signal modulated with a carrier frequency equal to 433.92 *MHz*. The digital signal will be converted into a biphasic waveform obtained from the demodulated input.

The two systems are based on an adaptive approach: the clock and data transmission are serialised to save only once the clock signal, sending data at a constant data rate compliant with the Nyquist criterion. The first CDR system is based on a frequency multiplier, giving the possibility to exploit the demodulator capabilities to their maximum, with the drawback of higher power consumption and occupied area. However, the second solution is compatible with a lower input data rate and shows low power and low-area characteristics. The two systems can provide a digitalized signal that can be correctly sampled thanks to the designed clock recovery system. The layout design has been completed, and some tests related to the architecture robustness, in terms of phase variation of the input signal and BER, have given excellent results.

6.1 Future Developments

New improvements can be obtained by taking into account some changes. Starting from the *Demodulator*, the $V_{envelope}$ of the rectifier, which in this project was a mandatory element, can be reduced through a voltage shifter or by directly replacing the *Rectifier* to minimize the overall system power dissipation. The demodulator itself can be replaced further to push the limits in data rate.

For what concerns the *Frequency multiplier*, a new approach to make it adaptable can be thought, being able to use it whatever the input frequency is.

Some adjustments can be applied to the *Memorization block* to provide a system able to activate the needed number of D-FF in the memorization chain according to the input frequency and to leave the on-chip oscillator design unchanged.

To get a narrower Gaussian bell, the *On-chip oscillator* can be made more robust from the Monte Carlo analysis point of view.

To reduce the system area making it more compact, the finger parameter can be wisely used. A smaller technology node can be implied, such as 90 nm, being also aware of how all the leakage phenomena can affect power consumption.

Bibliography

- [1] Sandro Carrara and Pantelis Georgiou. *Body Dust: Miniaturized Highly-integrated Low Power Sensing for Remotely Powered Drinkable CMOS Bio-electronics*. Apr. 2018 (cit. on p. 2).
- [2] Dongjin Seo, Jose Carmena, J.M. Rabaey, Elad Alon, and Michel Maharbiz. «Neural Dust: An Ultrasonic, Low Power Solution for Chronic Brain-Machine Interfaces». In: *Available at: ArXiv.org/abs/1307.2196*. (July 2013) (cit. on p. 2).
- [3] Francesca Rodino. «Design and development of an External Processing Unit for Wireless power and data transmission to miniaturized neural implants for reverting blindness». M.Sc. thesis. Turin, Italy: Politecnico di Torino, July 2022 (cit. on p. 2).
- [4] Gian Luca Barbruni, Fabio Asti, Paolo Motto Ros, Diego Ghezzi, Danilo Demarchi, and Sandro Carrara. «A 20 Mbps, 433 MHz RF ASK Transmitter to Inductively Power a Distributed Network of Miniaturised Neural Implants». In: *2021 IEEE International Symposium on Medical Measurements and Applications (MeMeA) (2021)*, pp. 1–6 (cit. on p. 2).
- [5] Gian Luca Barbruni, Paolo Motto Ros, Danilo Demarchi, Sandro Carrara, and Diego Ghezzi. «Ultra-Miniaturised CMOS Current Driver for Wireless Biphasic Intracortical Microstimulation». In: *2022 11th International Conference on Modern Circuits and Systems Technologies (MOCASST) (2022)*, pp. 1–4 (cit. on p. 2).
- [6] Fei Yuan. *CMOS Circuits for Passive Wireless Microsystems*. 1st. Springer Publishing Company, Incorporated, 2010. ISBN: 1441976795. DOI: 10.5555/1951760 (cit. on p. 5).
- [7] Hongqiang Zong, Jinpeng Shen, Shan Liu, Mei Jiang, Qingyuan Ban, Ling Tang, Fanyu Meng, and Xin'an Wang. «An ultra low power ASK demodulator for passive UHF RFID tag». In: *2011 9th IEEE International Conference on ASIC*. 2011, pp. 637–640. DOI: 10.1109/ASICON.2011.6157286 (cit. on pp. 5, 6, 13).

-
- [8] Hanphon Mitwong and Varakorn Kasemsuwan. «Low-voltage low-power current-mode amplitude shift keying (ASK) demodulator». In: *2012 IEEE International Conference on Electron Devices and Solid State Circuit (EDSSC)*. 2012, pp. 1–4. DOI: 10.1109/EDSSC.2012.6482777 (cit. on pp. 6, 13).
- [9] Narges Mousavi, Mohammad Sharifkhani, and Mohsen Jalali. «Ultra-low power current mode all- MOS ASK demodulator for radio frequency identification applications». In: *IET Circuits, Devices & Systems* 10.2 (2016), pp. 130–134. DOI: <https://doi.org/10.1049/iet-cds.2014.0252> (cit. on pp. 6, 13).
- [10] Yu-tso Lin, Tao Wang, Shey-shi Lu, and Guo-wei Huang. «A 0.5 V 3.1 mW Fully Monolithic OOK Receiver for Wireless Local Area Sensor Network». In: *2005 IEEE Asian Solid-State Circuits Conference*. 2005, pp. 373–376. DOI: 10.1109/ASSCC.2005.251743 (cit. on pp. 6, 13).
- [11] Mohammad Kafi Kangi, Mohammad Maymandi-Nejad, and Mahshid Nasseirian. «A Fully Digital ASK Demodulator With Digital Calibration for Bioimplantable Devices». In: *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 23.8 (2015), pp. 1557–1561. DOI: 10.1109/TVLSI.2014.2343946 (cit. on pp. 6, 13).
- [12] De-Ming Wang, Jian-Guo Hu, and Jing Wu. «An HF Passive RFID Tag IC With Low Modulation Index ASK Demodulator». In: *IEEE Transactions on Industrial Electronics* 66.3 (2019), pp. 2164–2173. DOI: 10.1109/TIE.2018.2840514 (cit. on pp. 6, 13).
- [13] Vincenzo Fiore, Egidio Ragonese, and Giuseppe Palmisano. «Low-Power ASK Detector for Low Modulation Indexes and Rail-to-Rail Input Range». In: *IEEE Transactions on Circuits and Systems II: Express Briefs* 63.5 (2016), pp. 458–462. DOI: 10.1109/TCSII.2015.2503651 (cit. on pp. 6, 7, 13).
- [14] Chua-Chin Wang, Ya-Hsin Hsueh, U Fat Chio, and Yu-Tzu Hsiao. «A C-less ASK demodulator for implantable neural interfacing chips». In: *2004 IEEE International Symposium on Circuits and Systems (ISCAS)*. Vol. 4. 2004, pp. IV–57. DOI: 10.1109/ISCAS.2004.1328939 (cit. on pp. 7, 13).
- [15] Tzung-Je Lee, Ching-Li Lee, Yan-Jhih Ciou, Chi-Chun Huang, and Chua-Chin Wang. «All-MOS ASK Demodulator for Low-Frequency Applications». In: *IEEE Transactions on Circuits and Systems II: Express Briefs* 55.5 (2008), pp. 474–478. DOI: 10.1109/TCSII.2007.912687 (cit. on pp. 7, 13).
- [16] Choi Myoeng-Jae and Jin Sung-Eon. «Design of low power ASK CMOS demodulator circuit for RFID tag: Design of All-MOSFET low power ASK demodulator». In: (Dec. 2010). DOI: 10.1109/EDSSC.2010.5713752 (cit. on pp. 7, 8, 13).

- [17] Chua-Chin Wang, Chih-Lin Chen, Ron-Chi Kuo, and Doron Shmilovitz. «Self-Sampled All-MOS ASK Demodulator for Lower ISM Band Applications». In: *IEEE Transactions on Circuits and Systems II: Express Briefs* 57.4 (2010), pp. 265–269. DOI: 10.1109/TCSII.2010.2043474 (cit. on pp. 8, 9, 12, 13, 26, 38, 79).
- [18] Mehdi Lotfi Navaii, Mohsen Jalali, and Hamed Sadjedi. «A 34-pJ/bit Area-Efficient ASK Demodulator Based on Switching-Mode Signal Shaping». In: *IEEE Transactions on Circuits and Systems II: Express Briefs* 64.6 (2017), pp. 640–644. DOI: 10.1109/TCSII.2016.2599262 (cit. on pp. 8, 13).
- [19] Rodrigo Iga, Sylvain Engels, and Laurent Fesquet. «An Event-Based Strategy for ASK demodulation». In: May 2019, pp. 1–5. DOI: 10.1109/EBCCSP.2019.8836831 (cit. on pp. 9, 10, 13).
- [20] J. Coulombe, J.-F. Gervais, and M. Sawan. «A cortical stimulator with monitoring capabilities using a novel 1 Mbps ASK data link». In: *Proceedings of the 2003 International Symposium on Circuits and Systems, 2003. ISCAS '03*. Vol. 5. 2003, pp. V–V. DOI: 10.1109/ISCAS.2003.1206181 (cit. on p. 9).
- [21] Hong Yu and Rizwan Bashirullah. «A Low Power ASK Clock and Data Recovery Circuit for Wireless Implantable Electronics». In: *IEEE Custom Integrated Circuits Conference 2006*. 2006, pp. 249–252. DOI: 10.1109/CICC.2006.321005 (cit. on pp. 10, 11, 13).
- [22] Jongsun Kim and Kenneth D. Pedrotti. «202pJ/bit area-efficient ASK demodulator for high-density visual prostheses». In: *Electronics Letters* 48 (2012), pp. 477–479 (cit. on pp. 11, 13).
- [23] Cihun-Siyong Alex Gong, Muh-Tian Shiue, Kai-Wen Yao, Tong-Yi Chen, Yin Chang, and Chun-Hsien Su. «A Truly Low-Cost High-Efficiency ASK Demodulator Based on Self-Sampling Scheme for Bioimplantable Applications». In: *IEEE Transactions on Circuits and Systems I: Regular Papers* 55.6 (2008), pp. 1464–1477. DOI: 10.1109/TCSI.2008.916422 (cit. on pp. 11–13).
- [24] Paul Teehan, Mark Greenstreet, and Guy Lemieux. «A survey and taxonomy of GALS design styles». In: *Design Test of Computers, IEEE* 24 (Oct. 2007), pp. 418–428. DOI: 10.1109/MDT.2007.151 (cit. on p. 14).
- [25] Maurizio Zamboni. «Sistemi Digitali Integrati». University Lecture. 2020 (cit. on pp. 15–17).
- [26] Tomaz FELICIJAN and Steve Furber. «An asynchronous ternary logic signaling system». In: *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on* 11 (Jan. 2004), pp. 1114–1119. DOI: 10.1109/TVLSI.2003.819571 (cit. on p. 16).

- [27] Harish G Shettar, Sujata Kotabagi, Nagaratna Shanbhag, Sachin Naik, Rahul Bagali, and Shreyas Nandavar. «Frequency Multiplier using Phase-Locked Loop». In: *2020 IEEE 17th India Council International Conference (INDICON)*. 2020, pp. 1–5. DOI: 10.1109/INDICON49873.2020.9342186 (cit. on p. 18).
- [28] *A Circuit for All Seasons: The Delay-Locked Loop* (summer 2018) (cit. on pp. 18–20).
- [29] Enrico Monaco, Massimo Pozzoni, Francesco Svelto, and Andrea Mazzanti. «Injection-Locked CMOS Frequency Doublers for -Wave and mm-Wave Applications». In: *Solid-State Circuits, IEEE Journal of* 45 (Sept. 2010), pp. 1565–1574. DOI: 10.1109/JSSC.2010.2049780 (cit. on p. 20).
- [30] Bhawika Kinger, Shruti Suman, K.G. Sharma, and P.K. Ghosh. «Design of Improved Performance Voltage Controlled Ring Oscillator». In: *2015 Fifth International Conference on Advanced Computing Communication Technologies*. 2015, pp. 441–445. DOI: 10.1109/ACCT.2015.127 (cit. on pp. 21, 22).
- [31] Jubayer Jalil, Mamun Bin Ibne Reaz, Mohammad Bhuiyan, Labonnah Rahman, and Tae-Gyu Chang. «Designing a ring-VCO for RFID transponders in 0.18 μm CMOS process». In: *TheScientificWorldJournal* 2014 (Jan. 2014), p. 580385. DOI: 10.1155/2014/580385 (cit. on p. 21).
- [32] Ashish Mishra and Gaurav Kumar Sharma. «Design of power optimal, low phase noise three stage Current Starved VCO». In: *2015 Annual IEEE India Conference (INDICON)*. 2015, pp. 1–4. DOI: 10.1109/INDICON.2015.7443417 (cit. on p. 22).
- [33] Kittipong Rongsawat and Apinunt Thanachayanont. «Ultra Low Power Analog Front-End for UHF RFID Transponder». In: *2006 International Symposium on Communications and Information Technologies*. 2006, pp. 1195–1198. DOI: 10.1109/ISCIT.2006.339969 (cit. on p. 22).
- [34] Shruti Suman, Monika Bhardwaj, and B.P. Singh. «An Improved Performance Ring Oscillator Design». In: *2012 Second International Conference on Advanced Computing Communication Technologies*. 2012, pp. 236–239. DOI: 10.1109/ACCT.2012.21 (cit. on p. 23).
- [35] Hyeonseok Hwang, Chan-Hui Jeong, Chankeun Kwon, Hoonki Kim, Youngmok Jeong, Bumsoo Lee, and Soo-Won Kim. «A 6MHz CMOS reference clock generator with temperature and supply voltage compensation». In: *2012 IEEE 11th International Conference on Solid-State and Integrated Circuit Technology*. 2012, pp. 1–3. DOI: 10.1109/ICSICT.2012.6467926 (cit. on p. 23).

- [36] Soumyaranjan Routray Prithiviraj Rajalingam Selvakumar Jayakumar. «Design and Analysis of Low Power and High Frequency Current Starved Sleep Voltage Controlled Oscillator for Phase Locked Loop Application». In: *Springer Nature B.V. 2020*. 2020. DOI: 10.1007/s12633-020-00619-7 (cit. on pp. 23, 24).
- [37] Mamun Bin Ibne Reaz Labonnah Farzana Rahman. «Design of Low Power and Low Phase Noise Current Starved Ring Oscillator for RFID Tag EEPROM». In: *Journal of Microelectronics, Electronic Components and Materials* 49 (2018), pp. 19–23. DOI: 10.33180/InfMIDEM2019.103 (cit. on p. 24).
- [38] Majid Amini Valashani, Mehdi Ayat, and Sattar Mirzakuchaki. «Design and analysis of a novel low-power and energy-efficient 18T hybrid full adder». In: *Microelectronics Journal* 74 (Apr. 2018), pp. 49–59. DOI: 10.1016/j.mejo.2018.01.018 (cit. on p. 50).
- [39] J.-P. Curty, N. Joehl, C. Dehollain, and M.J. Declercq. «Remotely powered addressable UHF RFID integrated system». In: *Solid-State Circuits, IEEE Journal of* 40 (Dec. 2005), pp. 2193–2202. DOI: 10.1109/JSSC.2005.857352 (cit. on pp. 55, 56).
- [40] Gian Luca Barbruni, Fabio Asti, Paolo Motto Ros, Diego Ghezzi, Danilo Demarchi, and Sandro Carrara. «A 20 Mbps, 433 MHz RF ASK Transmitter to Inductively Power a Distributed Network of Miniaturised Neural Implants». In: *2021 IEEE International Symposium on Medical Measurements and Applications (MeMeA)*. 2021, pp. 1–6. DOI: 10.1109/MeMeA52024.2021.9478678 (cit. on p. 79).
- [41] Lai Jiang Hang Yu Yan Li and Zhen Ji. «A 31 μ W ASK clock and data recovery circuit for wireless implantable systems». In: *International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS)* (Dec. 2010) (cit. on p. 79).
- [42] Hao Yan, Dian-cheng Wu, Yan Liu, Dong-hui Wang, and Chao-huan Hou. «A low-power CMOS ASK clock and data recovery circuit for cochlear implants». In: *2010 10th IEEE International Conference on Solid-State and Integrated Circuit Technology*. 2010, pp. 758–760. DOI: 10.1109/ICSICT.2010.5667407 (cit. on p. 79).
- [43] Hang Yul, Yan Lil, Lai Jiang, and Zhen Jii. «Ultra-low-power adaptable ASK clock and data recovery circuit for wireless implantable systems». In: *Electronics Letters* 49 (Dec. 2010). DOI: 10.1109/ISPACS.2010.5704714 (cit. on p. 79).

- [44] Sichen Yu, Zhonghan Shen, Xiaolu Liu, Huixiang Han, Xi Tan, Na Yan, and Hao Min. «A digital intensive clock recovery circuit for HF-Band active RFID tag». In: *IEICE Electronics Express* 11 (Mar. 2014), pp. 20140138–20140138. DOI: 10.1587/elex.11.20140138 (cit. on p. 79).
- [45] Aatmesh Shrivastava, Jagdish Pandey, Brian Otis, and Benton H. Calhoun. «A 50nW, 100kbps Clock/Data Recovery Circuit in an FSK RF Receiver on a Body Sensor Node». In: *2013 26th International Conference on VLSI Design and 2013 12th International Conference on Embedded Systems*. 2013, pp. 72–75. DOI: 10.1109/VLSID.2013.165 (cit. on p. 79).
- [46] Maysam Ghovanloo and Khalil Najafi. «A Wideband Frequency-Shift Keying Wireless Link for Inductively Powered Biomedical Implants». In: *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—I: REGULAR PAPERS*. Vol. 51. 12. 2004 (cit. on p. 79).

Appendix A

Appendix

A.1 ASK modulated signal generation

```
1 %%ASK generation with clock and random data
2 clear all
3 close all
4 clc
5 %Amplitude ASK modulated signal with MI=15\%
6 A1= 0.600;
7 A0=0.444;
8 f=433.92*10^6; %carrier frequency
9 %% DATA at lower data rate= 1 Mbps
10 n=100;
11 x= randi([0,1],[1,n]);% random number of bits
12 xdata= x;
13 bp=.0000005;
14 br=1/bp;
15 t1=bp/3000:bp/3000:bp;
16 ss=length(t1);
17 m=[];
18 for (i=1:length(xdata))
19     if (xdata(i)==1)
20         y=A1*sin(2*pi*f*t1)
21     else
22         y=A0*sin(2*pi*f*t1)
23     end
24     m=[m y];
25 end
26 tdata=bp/3000:bp/3000:bp*length(xdata)
27 %% CLOCK at higher data rate= 6 Mbps
28 x1= ones(1,35)
```



```

16 b_Vmin= find( file_Vmin(:,2) > 1.09);
17 b_Vmax= find( file_Vmax(:,2) > 1.09);
18 %Find how many points are
19 o_Vmin=length(b_Vmin);
20 o_Vmax=length(b_Vmax);
21 %Find the FOM
22 figure_Vmin=(o_Vmin/z_Vmin);
23 figure_Vmax=(o_Vmax/z_Vmax);
24 % Data rate exploited
25 Mbps= [1 2 3 4 5 6];
26 %Constant Vmin case: variation wrt the data rate and the MI
27 Vmin_9 = [ 1.0839 1.1934 1.5046 1.5054 1.4018 1.7020];
28 Vmin_15 = [ 1.2934 1.8243 1.7554 2.4355 2.7335 3.7445 ];
29 Vmin_20 = [ 1.4209 1.7118 5.5258 2.9209 3.8332 6.1642 ];
30 Vmin_30 = [1.6013 2.0624 13.6814 5.0974 6.1447 14.5684]
31 figure
32 plot(Mbps, Vmin_9, 'o:', 'LineWidth', 3 )
33 grid on
34 hold on
35 xlabel( 'DR [Mbps]', 'Interpreter','latex', 'FontSize', 14)
36 ylabel( '$VH_{avg}/VL_{avg}$ [s/s]', 'Interpreter','latex', 'FontSize',
37 , 14)
37 plot(Mbps, Vmin_15, 'go:', 'LineWidth', 2 )
38 plot(Mbps, Vmin_20, 'ro:', 'LineWidth', 2 )
39 plot(Mbps, Vmin_30, 'bo:', 'LineWidth', 2 )
40 legend( 'MI = 9%', 'MI = 15%', 'MI = 20%', 'MI = 30%', 'FontSize',16)
41 title( 'Comparison of Modulation index with VL=500 mV', 'FontSize',16)
42 hold off
43 figure
44 %Constant Vmax case: variation wrt the data rate and the MI
45 Vmax_9 = [ 1.0839 1.1934 1.5046 1.5054 1.4018 1.7020];
46 Vmax_15 = [1.0626 1.0291 1.0174 1.0863 0.9255 0.9437];
47 Vmax_20 = [1.0726 0.9802 0.9828 0.9604 0.7697 0.77];
48 Vmax_30 = [1.0765 1.0063 0.9435 0.8872 0.686 0.6574];
49 plot(Mbps, Vmax_9, 'o:', 'LineWidth', 3 )
50 grid on
51 hold on
52 xlabel( 'DR [Mbps]', 'Interpreter','latex', 'FontSize', 14)
53 ylabel( '$VH_{avg}/VL_{avg}$ [s/s]', 'Interpreter','latex', 'FontSize',
54 , 14)
54 plot(Mbps, Vmax_15, 'go:', 'LineWidth', 2 )
55 plot(Mbps, Vmax_20, 'ro:', 'LineWidth', 2 )
56 plot(Mbps, Vmax_30, 'bo:', 'LineWidth', 2 )
57 legend( 'MI = 9%', 'MI = 15%', 'MI = 20%', 'MI = 30%', 'FontSize',12)
58 title( 'Comparison of Modulation index with VH=600 mV', 'FontSize',16)
59 hold off

```

A.3 Phase robustness test

```

1 %%Phase robustness test for memorization block
2 close all
3 clear all
4 clc
5
6 %% ASK user Signal parameters
7 frequency_default = '3e6';
8 num_points_period_default = '6000';
9 high_level_default = '1.1';
10 low_level_default = '0';
11 N_ones_rectifier_default = '50';
12 N_periods_default = '9';
13 N_zeros_default = '8';
14 duty_cycle_nominal_default = '0.5';
15 duty_cycle_delta_max_default = '0.06';
16 error_button = 'No';
17 parameters = [];
18 frequency = str2double(answer{1});
19
20 % The following check is for the sake of data visualization only
21 if 1 <= log10(frequency) && log10(frequency) < 3
22     frequency_scaling_factor = 1;
23 elseif 3 <= log10(frequency) && log10(frequency) < 6
24     frequency_scaling_factor = 1e3;
25 elseif 6 <= log10(frequency) && log10(frequency) < 9
26     frequency_scaling_factor = 1e6;
27 elseif 9 <= log10(frequency)
28     frequency_scaling_factor = 1e9;
29 end
30 frequency_scaled = frequency/frequency_scaling_factor;
31
32 period = 1/frequency;
33
34 num_points_period = str2double(answer{2});
35 period_points_separation = period / num_points_period;
36
37 high_level = str2double(answer{3});
38 low_level = str2double(answer{4});
39
40 N_ones_rectifier = str2double(answer{5});
41
42 N_periods = str2double(answer{6});
43 N_zeros = str2double(answer{7});
44
45 duty_cycle_nominal = str2double(answer{8});

```

```

46 duty_cycle_delta_max = str2double(answer{9});
47 duty_cycle_min = duty_cycle_nominal - duty_cycle_delta_max;
48
49 time_one_period = [0:period_points_separation:period];
50
51 time = zeros(1,num_points_period*(N_periods+N_ones_rectifier+N_zeros)
52 );
53 output_signal = zeros(1,num_points_period*(N_periods+N_ones_rectifier
54 +N_zeros));
55 duty_cycle_array = zeros(1,(N_periods));
56
57 %% Rectifier ones
58 output_signal_rectifier_one_period = ones([1,num_points_period+1])*
59 high_level;
60 for i = 1:N_ones_rectifier
61     start_index_rectifier = 1 + num_points_period*(i-1);
62     stop_index_rectifier = 1 + num_points_period*i;
63     time(1,start_index_rectifier:stop_index_rectifier) =
64     time_one_period + (i-1)*period;
65     output_signal(1,start_index_rectifier:stop_index_rectifier) =
66     output_signal_rectifier_one_period;
67     fprintf(strcat("Step ",num2str(i)," of ",num2str(N_ones_rectifier
68 ),'\n'))
69 end
70 %% Signal
71
72 time_start_pulse = period*N_ones_rectifier + period_points_separation
73 ;
74 for i = 1:N_periods
75     duty_cycle = duty_cycle_min + round(rand*(2*duty_cycle_delta_max
76 ),2); % round to 2nd decimal digit to avoid errors on array size
77     duty_cycle_array(1,i) = duty_cycle; % store the values to check
78
79     t_on = cast((num_points_period + 1)*duty_cycle,"int32");
80     t_off = num_points_period + 1 - t_on;
81
82     output_on = ones([1,t_on])*high_level;
83     output_off = ones([1,t_off])*low_level;
84
85     output_signal_pulses_one_period = [output_on,output_off];
86

```

```

87     start_index_pulses = stop_index_rectifier + 1 + num_points_period
      *(i-1);
88     stop_index_pulses = stop_index_rectifier + 1 + num_points_period*
      i;
89     time(1,start_index_pulses:stop_index_pulses) = time_start_pulse +
      time_one_period + (i-1)*period;
90     output_signal(1,start_index_pulses:stop_index_pulses) =
      output_signal_pulses_one_period;
91
92     fprintf(strcat("Step ",num2str(i)," of ",num2str(N_periods),'\n')
      )
93 end
94 %% Zeros at the end
95
96 time_start_zeros = time_start_pulse + period*N_periods +
      period_points_separation;
97
98 end_of_pulses = zeros([1,num_points_period+1]);
99
100 for i = 1:N_zeros
101     start_index_zeros = stop_index_pulses + 1 + num_points_period*(i
      -1);
102     stop_index_zeros = stop_index_pulses + 1 + num_points_period*i;
103     time(1,start_index_zeros:stop_index_zeros) = time_start_zeros +
      time_one_period + (i-1)*period;
104     output_signal(1,start_index_zeros:stop_index_zeros) =
      end_of_pulses;
105
106     fprintf(strcat("Step ",num2str(i)," of ",num2str(N_zeros),'\n'))
107 end
108
109 file = [time' output_signal'];
110 writematrix(file,'C:\Users\cerba\Desktop\test\disag_test.txt');
111 fprintf("Output file saved\n");

```

A.4 BER test

```

1 %%BER evaluation
2 clear all
3 close all
4 clc
5 %Definition of bit period and number of bits
6 bit=1e-6;
7 nbit=1000;
8

```

```

9 %Importation of ASK modulated signal
10 in1=importfile('ask1_1000.csv',[1,inf]);
11 in2=importfile('ask2_1000.csv',[1,inf]);
12 in3=importfile('ask3_1000.csv',[1,inf]);
13
14 %Importation of Demodulated signal after BER logic
15 out1=importfile('out1_1000.csv',[1,inf]);
16 out2=importfile('out2_1000.csv',[1,inf]);
17 out3=importfile('out3_1000.csv',[1,inf]);
18
19 Signal_in=[in1;in2;in3];
20 Signal_in(:,1)=Signal_in(:,1)-Signal_in(1,1);
21 Signal_out=[out1;out2;out3];
22 Signal_out(:,1)=Signal_out(:,1)-Signal_out(1,1);
23
24 %Duration time instant
25 dt=Signal_out(2,1);
26
27 %Search for index higher than half amplitude
28 idx1=find(Signal_in(:,2)>0.44);
29 Signal_in(:,2)=0;
30 %Clamping to 1.1 V
31 Signal_in(idx1,2)=1.1;
32 Signal_out(Signal_out(:,2)>0.55,2)=1.1;
33 Signal_out(Signal_out(:,2)<=0.55,2)=0;
34
35
36
37 for i=1:nbit
38     range=find((Signal_in(:,1))>=(i-1)*bit+ritardo) & (Signal_in(:,1)<
39         i*bit+ritardo));
40     if (mean(Signal_in(range,2))>0.1)
41         Signal_in(range,2)=1.1;
42     else
43         Signal_in(range,2)=0;
44     end
45 end
46
47 sampling=(0:333.333333e-9:nbit*bit)';
48 indexes_in=dsearchn(Signal_in(:,1),sampling(:));
49 indexes_out=dsearchn(Signal_out(:,1),sampling(:));
50 in_sampled=Signal_in(indexes_in,2);
51 out_sampled=Signal_out(indexes_out,2);
52
53 for j=1:nbit
54     if (mean(out_sampled(3*(j-1)+1:3*j))>0.55)
55
56         out(j)=1;

```

```
57     else
58         out(j)=0;
59     end
60     if (mean(in_sampled(3*(j-1)+1:3*j)) > 0.55)
61         in(j)=1;
62     else
63         in(j)=0;
64     end
65 end
66
67 end
68 a=find(abs(in-out) > 0.001);
69 bit_error_rate=length(a)/nbit;
```