Low-power clock generator system for wireless sensors networks

Sara Sofia Barbosa dos Santos

Thesis to obtain the Master of Science Degree in Electronics Engineering

Supervisor: Prof. João Manuel Torres Caldinhas Simões Vaz

Examination Committee

Chairperson: Prof. Paulo Ferreira Godinho Flores
Supervisor: Prof. João Manuel Torres Caldinhas Simões Vaz
Member of the Committee: Prof. Marcelino Bicho dos Santos

September 2020
Declaration

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

Declaro que o presente documento é um trabalho original da minha autoria e que cumpre todos os requisitos do Código de Conduta e Boas Práticas da Universidade de Lisboa.
Acknowledgments

First I want to thank my supervisor, Professor João Caldinhas Vaz, for the opportunity to work on a subject so interesting and demanding, and for all his guidance and support, as well as for his suggestions and critical observations throughout this dissertation.

I would also like to acknowledge the Wireless Circuits - Lx research group from the Lisbon pole of the Instituto de Telecomunicações and the FCT project with the reference PTDC/EEI-EEE/30539/2017 for the opportunity to work on this project.

A special thanks to my family for all the support and encouragement and to my friends for all the help, good moments and good company, making this a fun and inspiring experience. To Ricardo Raseth, I am grateful for all your patience, motivation and for all the time you dedicated to helping me with this project. Thank you for always being there for me.
Abstract

Extending the battery life of devices is an objective transversal to any area of electronics. Especially when working with Wireless Sensors Network (WSN), usually installed in remote locations, the sensors battery life gains greater importance.

To achieve the goal of prolonging the time a battery is powering the circuit, WSN systems often rely on duty-cycle schemes of operation: the system is periodically and briefly powered-up to receive/transmit data, being in sleep mode most of the time. Despite that technique, the block responsible for keeping track of when the communication time slots must happen waking-up other blocks from the sleep mode - the wake-up timer - must be always operating. Thus, it is of extreme importance, and also a big challenge, to implement a wake-up timer with minimum power consumption. With the reduction of power consumption comes a lower frequency stability. The solution is to design a reference frequency oscillator to operate as the system clock which is only active during communication time slots and that is used to calibrate the wake-up oscillator.

In this work a crystal oscillator is proposed as the system clock for its high frequency stability. The wake-up timer is an Ultra-Low Power (ULP) oscillator, implemented as a RC oscillator with digital calibration of the frequency through corrections in the bias current. A Frequency-Locked Loop (FLL) is implemented to synchronize the oscillators.

Keywords

Clock generator system, Crystal oscillator, Ultra-low power oscillator, Calibration, Frequency-locked loop
Resumo

Prolongar a vida útil da bateria dos dispositivos é um objetivo transversal a qualquer área da eletrônica. Especialmente ao trabalhar com redes de sensores sem fios, geralmente instaladas em locais remotos, a vida útil da bateria dos sensores ganha uma maior importância.

Para atingir o objetivo de aumentar o tempo de vida da bateria, os sistemas de redes de sensores sem fios são apenas ativados periodicamente para receber/transmitir dados, estando no modo inativo o resto do tempo. No entanto, o oscilador de wake-up, bloco responsável por detectar quando os intervalos de tempo de comunicação devem ocorrer retirando os outros blocos do modo inativo, deve estar sempre ativo. Assim, é de extrema importância, para além de ser um grande desafio, implementar um oscilador de wake-up com consumo de energia mínimo. Da redução do consumo de energia advém uma estabilidade de frequência mais baixa. A solução passa por projetar um oscilador de frequência de referência para operar como relógio do sistema que só está ativo durante os intervalos de tempo de comunicação e que é utilizado para calibrar o oscilador de wake-up.

Neste trabalho é proposto um oscilador de cristal como relógio do sistema devido à sua alta estabilidade de frequência. O oscilador de wake-up é um oscilador de consumo ultra-baixo, implementado como um oscilador RC com calibração de frequência digital, através de correções na corrente de polarização. Os osciladores são sincronizados através de uma malha de frequência síncrona.

Palavras Chave

Sistema de geração de relógio, Oscilator de cristal, Oscilator de ultra-baixo consumo, Calibração, Malha de frequência síncrona
Contents

1 Introduction ........................................... 1
   1.1 Objectives ........................................... 3
   1.2 Motivation ........................................... 4
   1.3 Methodology ........................................... 4
   1.4 Document organization ................................. 5

2 State of the Art .................................. 7
   2.1 Reference frequency oscillator ......................... 9
      2.1.1 Crystal oscillator ................................ 9
      2.1.2 Quick start-up crystal oscillator .................. 10
      2.1.3 Temperature compensated crystal oscillator ........ 11
      2.1.4 Other resonators .................................. 12
      2.1.5 No resonator ...................................... 13
   2.2 Ultra-low power oscillator ............................. 13
      2.2.1 With resonator .................................... 14
      2.2.2 No resonator ...................................... 14
   2.3 Synchronization ...................................... 15

3 Oscillators theory ................................ 17
   3.1 Harmonic oscillator .................................. 19
      3.1.1 Resonator ......................................... 20
      3.1.2 Oscillator topologies ............................... 22
      3.1.3 Oscillator analysis ................................ 24
      3.1.4 Quick start-up techniques ......................... 28
   3.2 Relaxation oscillator ................................ 29
      3.2.1 Ring oscillator ................................... 29
      3.2.2 Charge-pump oscillator ............................ 31
   3.3 Phase noise and jitter ................................ 33
   3.4 Period counter ....................................... 36
## List of Figures

3.1 Representation of Butterworth Van Dyke model of a crystal resonator for the fundamental mode. .................................................. 20  
3.2 Representation of the crystal resonator reactance for a 32 MHz crystal. .... 21  
3.3 Representation of a Colpitts oscillator Alternate Current (AC) schematic. ... 22  
3.4 Representation of two three-point crystal oscillator schematics. ................. 23  
3.5 Representation of a Hartley oscillator AC schematic. .......................... 23  
3.6 Representation of a crystal oscillator output voltage during start-up and steady-state... 24  
3.7 Graphical representation of three-point crystal oscillator according to the feedback method. ... 24  
3.8 Representation of the linear and non-linear blocks of a split crystal oscillator. ... 25  
3.9 Representation of the three-point crystal oscillator circuit for small signal analysis. ... 26  
3.10 Graphical representation of three-point crystal oscillator in the complex plane. ... 27  
3.11 Representation of three-point crystal oscillator circuit as motional and circuit impedances. ... 28  
3.12 Representation of a ring oscillator. ........................................... 30  
3.13 Simplified schematic of comparator-based relaxation oscillator. ................ 30  
3.14 Representation of capacitor voltage and oscillator output voltage. .............. 32  
3.15 Representation of an oscillator output power spectrum. ....................... 33  
3.16 Representation of Leeson's model oscillator phase noise. ....................... 34  
3.17 Representation of jitter noise in a square wave. ................................ 35  
3.18 Representation of phase jitter noise. ......................................... 36  
3.19 Representation of period counter and respective signals. ........................ 37  
3.20 Representation of a flip-flop using Not-AND (NAND) gates. .................... 38  
3.21 Representation of two versions of master-slave Flip-Flop (FF). ................. 39  
3.22 Dynamic transmission gate-based FF. ....................................... 40  
3.23 Dynamic Clocked Complementary Metal Oxide Semiconductor (C2MOS) FF. .... 40  
3.24 Representation of two versions of True Single-Phase Clock (TSPC) FF. ........ 41  
3.25 Dynamic TSPC positive edge-triggered FF. .................................. 42
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>Clock generator system block diagram.</td>
<td>45</td>
</tr>
<tr>
<td>4.2</td>
<td>Crystal oscillator schematic.</td>
<td>47</td>
</tr>
<tr>
<td>4.3</td>
<td>Crystal oscillator output voltage variation with $R_{fb}=1\ k\Omega$, 100 $k\Omega$ and 20 $M\Omega$.</td>
<td>48</td>
</tr>
<tr>
<td>4.4</td>
<td>Crystal oscillator output voltage and frequency transient.</td>
<td>49</td>
</tr>
<tr>
<td>4.5</td>
<td>Single-Metal Oxide Semiconductor (MOS) crystal oscillator schematic with power-down.</td>
<td>50</td>
</tr>
<tr>
<td>4.6</td>
<td>Single-MOS crystal oscillator output voltage.</td>
<td>50</td>
</tr>
<tr>
<td>4.7</td>
<td>Crystal oscillator phase noise.</td>
<td>51</td>
</tr>
<tr>
<td>4.8</td>
<td>Voltage buffer schematic.</td>
<td>52</td>
</tr>
<tr>
<td>4.9</td>
<td>Voltage buffer output voltage.</td>
<td>53</td>
</tr>
<tr>
<td>4.10</td>
<td>Crystal Oscillator (XO) frequency for temperature and voltage supply variation.</td>
<td>53</td>
</tr>
<tr>
<td>4.11</td>
<td>Conventional relaxation oscillator schematic.</td>
<td>55</td>
</tr>
<tr>
<td>4.12</td>
<td>Operational Amplifier (OpAmp) comparator schematic.</td>
<td>56</td>
</tr>
<tr>
<td>4.13</td>
<td>OpAmp comparator output voltage.</td>
<td>57</td>
</tr>
<tr>
<td>4.14</td>
<td>Latched comparator schematic.</td>
<td>57</td>
</tr>
<tr>
<td>4.15</td>
<td>Latched comparator schematic with preamplifier.</td>
<td>58</td>
</tr>
<tr>
<td>4.16</td>
<td>Latched comparator schematic with double cross connections.</td>
<td>59</td>
</tr>
<tr>
<td>4.17</td>
<td>Output voltage of latched comparator with double cross connections.</td>
<td>59</td>
</tr>
<tr>
<td>4.18</td>
<td>Representation of the Set-Reset (SR) latch and respective Not-OR (NOR) gate.</td>
<td>60</td>
</tr>
<tr>
<td>4.19</td>
<td>Relaxation oscillator voltage waveforms in various points: $V_C$, R, S and Q.</td>
<td>61</td>
</tr>
<tr>
<td>4.20</td>
<td>Relaxation oscillator supply current.</td>
<td>61</td>
</tr>
<tr>
<td>4.21</td>
<td>Improved RC oscillator schematic.</td>
<td>62</td>
</tr>
<tr>
<td>4.22</td>
<td>Improved RC oscillator voltages and currents.</td>
<td>63</td>
</tr>
<tr>
<td>4.23</td>
<td>Implemented current reference generator with start-up circuit.</td>
<td>64</td>
</tr>
<tr>
<td>4.24</td>
<td>RC Oscillator (RCO) frequency for temperature and voltage supply variation.</td>
<td>65</td>
</tr>
<tr>
<td>4.25</td>
<td>Detailed clock generator system block diagram.</td>
<td>66</td>
</tr>
<tr>
<td>4.26</td>
<td>Implemented calibration loop period counter.</td>
<td>67</td>
</tr>
<tr>
<td>4.27</td>
<td>Period counter output signals.</td>
<td>68</td>
</tr>
<tr>
<td>4.28</td>
<td>Counter output example with transition of all bits.</td>
<td>69</td>
</tr>
<tr>
<td>4.29</td>
<td>Flowchart of calibration algorithm.</td>
<td>71</td>
</tr>
<tr>
<td>4.30</td>
<td>System calibration simulation.</td>
<td>72</td>
</tr>
<tr>
<td>A.1</td>
<td>Representation of the complete circuit of controlled RC oscillator.</td>
<td>89</td>
</tr>
</tbody>
</table>
List of Tables

1.1 Project specifications. ................................................................. 4

2.1 Performance summary of XO presented in literature. ...................... 11

2.2 Performance summary of RC Oscillator (RCO) presented in literature. 15

4.1 Start-up time and energy trade-off. ............................................... 48

4.2 Single-MOS crystal oscillator performance summary. ..................... 51

4.3 Comparison of crystal oscillator with state of the art works. .......... 54

4.4 SR latch truth table. ................................................................. 60

4.5 Comparison of RC oscillator with state of the art works. ............... 66

4.6 Output bits of a binary counter, Gray counter and Johnson counter, from 0 to 7. 69

4.7 Comparison of RC oscillator before and after calibration with state of the art works. 73
## Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>AC</td>
<td>Alternate Current</td>
</tr>
<tr>
<td>AFC</td>
<td>Automatic Frequency Calibration</td>
</tr>
<tr>
<td>ATCXO</td>
<td>Analog Temperature Compensated Crystal Oscillator</td>
</tr>
<tr>
<td>BAW</td>
<td>Bulk Acoustic Wave</td>
</tr>
<tr>
<td>BVD</td>
<td>Butterworth Van Dyke</td>
</tr>
<tr>
<td>C\textsuperscript{2}MOS</td>
<td>Clocked Complementary Metal Oxide Semiconductor</td>
</tr>
<tr>
<td>CCO</td>
<td>Current-Controlled Oscillator</td>
</tr>
<tr>
<td>CFI</td>
<td>Constant Frequency Injection</td>
</tr>
<tr>
<td>CMOS</td>
<td>Complementary Metal Oxide Semiconductor</td>
</tr>
<tr>
<td>DCXO</td>
<td>Digitally-Controlled Crystal Oscillator</td>
</tr>
<tr>
<td>DTCXO</td>
<td>Digital Temperature Compensated Crystal Oscillator</td>
</tr>
<tr>
<td>FBAR</td>
<td>Film Bulk Acoustic Resonator</td>
</tr>
<tr>
<td>FF</td>
<td>Flip-Flop</td>
</tr>
<tr>
<td>FLL</td>
<td>Frequency-Locked Loop</td>
</tr>
<tr>
<td>IoT</td>
<td>Internet of Things</td>
</tr>
<tr>
<td>LSB</td>
<td>Least Significant Bit</td>
</tr>
<tr>
<td>LUT</td>
<td>Look-Up Table</td>
</tr>
<tr>
<td>MEMS</td>
<td>Micro Electromechanical System</td>
</tr>
<tr>
<td>MOS</td>
<td>Metal Oxide Semiconductor</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Definition</td>
</tr>
<tr>
<td>--------------</td>
<td>------------</td>
</tr>
<tr>
<td>MSB</td>
<td>Most Significant Bit</td>
</tr>
<tr>
<td>NAND</td>
<td>Not-AND</td>
</tr>
<tr>
<td>NMOS</td>
<td>N-type Metal Oxide Semiconductor</td>
</tr>
<tr>
<td>NOR</td>
<td>Not-OR</td>
</tr>
<tr>
<td>OpAmp</td>
<td>Operational Amplifier</td>
</tr>
<tr>
<td>PFD</td>
<td>Phase-Frequency Detector</td>
</tr>
<tr>
<td>PLL</td>
<td>Phase-Locked Loop</td>
</tr>
<tr>
<td>PMOS</td>
<td>P-type Metal Oxide Semiconductor</td>
</tr>
<tr>
<td>ppm</td>
<td>parts per million</td>
</tr>
<tr>
<td>PTAT</td>
<td>Proportional to Absolute Temperature</td>
</tr>
<tr>
<td>PVT</td>
<td>Process, Supply Voltage and Temperature</td>
</tr>
<tr>
<td>RMS</td>
<td>Root Mean Square</td>
</tr>
<tr>
<td>RCO</td>
<td>RC Oscillator</td>
</tr>
<tr>
<td>SAR</td>
<td>Successive Approximations Register</td>
</tr>
<tr>
<td>SAW</td>
<td>Surface Acoustic Wave</td>
</tr>
<tr>
<td>SR</td>
<td>Set-Reset</td>
</tr>
<tr>
<td>TCXO</td>
<td>Temperature Compensated Crystal Oscillator</td>
</tr>
<tr>
<td>TSPC</td>
<td>True Single-Phase Clock</td>
</tr>
<tr>
<td>TVC</td>
<td>Time-to-Voltage Converter</td>
</tr>
<tr>
<td>ULP</td>
<td>Ultra-Low Power</td>
</tr>
<tr>
<td>UMC</td>
<td>United Microelectronics Corporation</td>
</tr>
<tr>
<td>VCO</td>
<td>Voltage-Controlled Oscillator</td>
</tr>
<tr>
<td>VTC</td>
<td>Voltage Transfer Characteristic</td>
</tr>
<tr>
<td>WSN</td>
<td>Wireless Sensors Network</td>
</tr>
<tr>
<td>XO</td>
<td>Crystal Oscillator</td>
</tr>
</tbody>
</table>
1 Introduction

Contents

1.1 Objectives ................................................................. 3
1.2 Motivation ................................................................. 4
1.3 Methodology .............................................................. 4
1.4 Document organization ............................................... 5
Introduction

WSNs are used in multiple short-range low data-rate applications. A desirable property of the battery powered sensor nodes is to have a long energy autonomy, which is only attainable if ultra-low power circuits and systems are wisely used. The part of the clock generator system that is always operating must reduce its power to the minimum to allow the battery life to be extended.

The main architecture presented in literature consists in having an ULP oscillator to keep track of time and a precise oscillator, typically the system crystal oscillator master clock, working during sensor data communication time periods. The ULP oscillator accuracy and frequency stability would not be sufficiently high to track the time correctly for long periods of time, thus it is synchronized with a more stable oscillator, the crystal oscillator. The synchronization must be performed during the sensor data communication time periods, when the crystal oscillator is operating. WSNs for Internet of Things (IoT) applications obey communication standards that take into account the need for short-time communications slots and long periods in idle state to reduce power consumption.

1.1 Objectives

The objective of this project is to develop a clock generator system for wireless sensors networks, which must be able to enter an idle state when no sensor data communication is performing, in order to extend the system battery life. The system is comprised by a reference frequency oscillator, a wake-up oscillator and a calibration loop to synchronize both oscillator frequencies. An on-chip 32 MHz crystal oscillator - a standard frequency used by several manufacturers - provides the reference frequency and a 31.25 kHz ULP relaxation oscillator - selected to be a power of two multiple of the reference frequency - is used as a wake-up oscillator, to keep track of time.

For the WSN system to enter the idle state, all blocks except the wake-up oscillator must be powered-down, therefore, all circuits must be dimensioned to have the lower power consumption possible but the wake-up timer requires special attention. Since it is always powered-up, the wake-up timer is implemented by an ULP oscillator. This oscillator must send a signal for the system to exit the idle state, in order to allow for a data communication to happen, and, simultaneously to the data transfer, it must be
synchronized with the reference oscillator to calibrate its frequency. The synchronization can be accomplished through a FLL, where a digital word is generated and compared with its ideal value to calibrate the wake-up oscillator.

The system must be designed in United Microelectronics Corporation (UMC) 130 nm technology. Table 1.1 summarizes this project main specifications for easy reading. Besides these specifications, all circuits are dimensioned and designed for minimum area and the ULP oscillator frequency error after calibration must originate a deviation of less than $\pm 5 \text{ min}$ after 12 h of operation, which corresponds to a frequency error of $\pm 230 \text{ Hz}$.

<table>
<thead>
<tr>
<th>Table 1.1: Project specifications.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Technology</td>
</tr>
<tr>
<td>Voltage supply</td>
</tr>
<tr>
<td>Reference oscillator</td>
</tr>
<tr>
<td>Frequency</td>
</tr>
<tr>
<td>Power consumption</td>
</tr>
<tr>
<td>ULP oscillator</td>
</tr>
<tr>
<td>Frequency</td>
</tr>
<tr>
<td>Power consumption</td>
</tr>
<tr>
<td>Frequency error (after calibration)</td>
</tr>
</tbody>
</table>

1.2 Motivation

Microelectronics is an area of extreme importance in modern society. Working to decrease more and more the technologies size and cost, while increasing its performance, microelectronics have been presenting increasingly better results. This leads to a growth in the communication systems performance and development of concepts as WSN and IoT, connecting everyday devices and networks. Sensors represent an area which is growing exponentially as the wireless sensors networks allow to monitor, collect and transmit information collected by the sensors, particularly in inhospitable places.

Designing low-power circuits while maintaining a good performance is a challenge for every designer. Power consumption is a recurrent concern in circuits design, therefore, in a clock generator system which must be always operating, this subject is of extreme importance. The desire for minimum power consumption is always accompanied by a trade-off with performance.

1.3 Methodology

In this work to achieve the proposed objectives a theoretical study of crystal oscillators and relaxation oscillators is made first. Existing topologies and their advantages and disadvantages are analyzed. After
chosen the circuit topology to be used for each system block, it is implemented and simulated using Cadence software. The circuits are first partially implemented with ideal components which are then replaced by technology components, to understand their influence in the circuits. Improvements are added to the circuits in order to fulfill the system specifications.

A basic topology of the crystal oscillator is implemented to study its operation. Improvements are added to decrease its start-up time and energy consumption. Similarly, the ULP oscillator study starts with a conventional, simpler circuit and an improved topology is then implemented. The digital word which controls the ULP oscillator frequency is generated through a digital calibration logic block. This block implements an algorithm created using the hardware description language Verilog. A digital counter is designed and simulated to detect the frequency error between both oscillators. With all the blocks implemented the complete system is tested and its performance is simulated.

1.4 Document organization

This document is structured in five chapters. The first chapter presents the objectives and the motivation for developing this work, as well as the methodology to follow. The second chapter corresponds to the state of the art where most relevant works in literature are discussed. Works related to reference oscillators and ULP oscillators, as well as possible architectures for the system synchronization, are analyzed. In the third chapter a theoretical study of crystal and relaxation oscillators and the digital counter is done. The phase noise and jitter, important parameters in the oscillators, are also studied in the same chapter. The fourth chapter presents this work results. Each circuit is designed, simulated and redesigned until the specifications are fulfilled. Afterwards, the completed system is simulated in a mixed-signal design. Finally, the fifth chapter presents some conclusions about this work and the work to be developed in the future in order to improve the system performance.
2

State of the Art

Contents

2.1 Reference frequency oscillator ........................................... 9
2.2 Ultra-low power oscillator .............................................. 13
2.3 Synchronization ......................................................... 15
State of the Art

To fulfill the tight power specifications and increase battery time of IoT and WSNs applications the literature frequently implements architectures using a duty-cycle scheme, where a wake-up timer powers up the system periodically. After powered-up an ULP oscillator is synchronized with a high frequency stability reference oscillator, correcting its oscillation frequency.

Solutions presented in literature for the reference frequency oscillator, the ULP oscillator and the synchronization system are described in this chapter. For the reference frequency oscillator works with and without external resonator are presented. Possible improvements of the start-up time and frequency stability with temperature variation are also presented. In section 2.2 implementations of ULP oscillators with and without a resonator are described and, finally, section 2.3 presents different options for frequency synchronization of both oscillators.

2.1 Reference frequency oscillator

A reference clock is necessary whenever digital or mixed-signal circuits are used. This clock is generated by a precise oscillator which must fulfill the requirements of the WSN and the standards used, such as low power consumption, high frequency stability and low phase noise. There are different types of oscillators capable of generating a reference frequency that fulfills the necessary requirements, some of which are presented in this section.

2.1.1 Crystal oscillator

The classical solution for the reference clock is a Crystal Oscillator (XO). Different methodologies are based on a three-point oscillator, an oscillator with a single active device and an impedance at each point (gate, drain and source). Two capacitive impedances and one inductive impedance are used to obtain an oscillator of the Colpitts family or vice-versa to obtain a Hartley oscillator. The first references at literature use these simple topologies and more recent works introduce improvements to the basic XO. A crystal oscillator where the transistor drain is grounded and its source is biased by a current source
is presented in [1]. The crystal resonator is connected between the gate and the drain, originating a one-pin oscillator.

Another possible implementation for a crystal oscillator is the Pierce topology [2–5], with the transistor source grounded and with an inductive resonator. In [2] a Digitally-Controlled Crystal Oscillator (DCXO) based on a Pierce oscillator is reported. To improve the frequency stability of the oscillator the core circuit is a cross-coupled differential pair. In this implementation two identical bank capacitor arrays are used to calibrate the load capacitance of the oscillator, tuning the oscillation frequency.

With the objective of reducing power consumption a crystal oscillator operating below 0.5 V is described in [3]. For the oscillator to work with a low voltage supply the transistors length is reduced and the transistors optimum width must be calculated. For the same purpose, [4] reduces the oscillation amplitude, to decrease the current consumption. A voltage reference circuit bias the oscillator with a voltage value, lower than the supply voltage. In the end a level shifter circuit is used to increase the oscillation amplitude back to voltage supply levels.

2.1.2 Quick start-up crystal oscillator

In applications where low power consumption is mandatory, the long start-up time of crystal oscillators can be a problem. To address this problem the literature reports different methodologies to reduce the start-up time of a crystal oscillator, reducing its energy consumption during this period.

In [6] a quick start-up oscillator based on Constant Frequency Injection (CFI) is presented. A single frequency signal is injected at the crystal frequency to accelerate the start-up time of the oscillator. The signal is generated by an ULP ring oscillator which frequency is digitally calibrated. However, to assure good results the injection signal must match the oscillation frequency.

To guarantee that energy is given to the crystal oscillator at its resonance frequency in [7, 8] a frequency-variant signal is injected. A chirp-modulated signal [7], generated by a Voltage-Controlled Oscillator (VCO) and a sweep circuit, sweeps the frequency from a high to a low value, passing through the resonator frequency. In [7], additionally to the injection of a frequency-variant signal, the negative resistance of the crystal oscillator is increased in the initial state through a multistage inverter, digitally controlling the gate width of the inverters. When the steady-state is reached the negative resistance is decreased again. In [8] a signal that dithers between two frequencies, lower and higher than the crystal frequency, is generated. The dithered signal frequency is generated by a digitally controlled oscillator.

Another approach to the start-up time problem is a precisely-timed energy injection [9]. The crystal resonator is disconnected from the oscillation circuit and it is pre-energized, being injected only one impulse signal into the crystal resonator. The signal is generated by a ring oscillator with a specific duration, calculated to minimize the start-up time. The impulse signal in the time domain translates into a sinc-shaped injection signal in the frequency domain. The sinc-shaped signal used is centered at the
injection frequency. The signal wide bandwidth assures that the resonator is energized at the correct frequency, even with a large deviation from the resonator frequency, eliminating the problem of obtaining a precise injection frequency. Work [9] claims to present a quick start-up technique much less sensitive to injection frequency errors than prior works.

Instead of injecting a signal in the oscillator, in [10] the value of the load capacitor of the crystal oscillator is decreased, increasing the negative value of the circuit impedance real component, allowing the oscillator to reach the steady-state faster. An envelope detector, along with a comparator, detects if the clock amplitude is sufficient, if so, the load capacitor value is increased.

A few examples of crystal oscillators presented in the literature are compared. The more important parameters of each oscillator are summarized in Table 2.1.

Table 2.1: Performance summary of XO presented in literature.

<table>
<thead>
<tr>
<th>Reference</th>
<th>[2]</th>
<th>[7]</th>
<th>[9]</th>
<th>[10]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Complementary Metal Oxide Semiconductor (CMOS) process</td>
<td>65 nm</td>
<td>180 nm</td>
<td>65 nm</td>
<td>90 nm</td>
</tr>
<tr>
<td>Frequency</td>
<td>26 MHz</td>
<td>39.25 MHz</td>
<td>50 MHz</td>
<td>24 MHz</td>
</tr>
<tr>
<td>Supply voltage</td>
<td>1.8 V</td>
<td>1.5 V</td>
<td>1.0 V</td>
<td>1.0 V</td>
</tr>
<tr>
<td>Start-up time</td>
<td>3.2 ms</td>
<td>158 µs</td>
<td>2.2 µs</td>
<td>200 µs</td>
</tr>
<tr>
<td>Steady-state power consumption</td>
<td>2.16 mW</td>
<td>181 µW</td>
<td>195 µW</td>
<td>95 µW</td>
</tr>
<tr>
<td>Phase noise (@ 1 kHz)</td>
<td>-136.1 dBc/Hz</td>
<td>-147 dBc/Hz</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>Fast start-up technique</td>
<td>None</td>
<td>Chirp Injection and negative resistance boost</td>
<td>Precisely-timed injection</td>
<td>Dynamically-adjusted load</td>
</tr>
</tbody>
</table>

A crystal oscillator without any technique to accelerate the start-up is compared with three different options of oscillators with fast start-up techniques: chirp injection and negative resistance boost; dynamically-adjusted load and precisely-timed injection.

### 2.1.3 Temperature compensated crystal oscillator

Different types of Temperature Compensated Crystal Oscillator (TCXO) can be implemented to improve the frequency stability. The main solutions are Analog Temperature Compensated Crystal Oscillator (ATCXO), Digital Temperature Compensated Crystal Oscillator (DT CXO) and a combination of the two.

Examples of ATCXOs are presented in [11–14]. In [11] a temperature dependent voltage, which can be modeled as a cubic function by variable gain amplifiers and a voltage adder, is produced. The voltage
produced drives a varactor which changes the crystal load capacitance, adjusting the crystal frequency. Another TCXO that produces a temperature dependent voltage to adjust the frequency is presented in [12]. The temperature compensation circuit uses logarithmic, exponential and gain amplifiers. The output voltage of the temperature sensor is converted into a cubic function voltage by the temperature compensation circuit. The method presented in [13] uses two temperature independent multipliers and an adder as a cubic function generator. The compensation voltage is generated introducing the output voltage of the temperature sensor in the cubic function generator.

In [14] a TCXO where the compensation voltage is created by variable gain amplifiers is also presented. Although the compensation voltage generator is analog, a non-volatile memory is needed to store the cubic function coefficients. Work [15] presents a DTCXO. The compensation network incorporates a temperature sensor and a microcontroller. The microcontroller reads the temperature sensor output and controls the crystal oscillator voltage, altering the oscillator frequency.

To obtain a better frequency stability solutions which combine analog and digital compensation are studied in [16, 17]. In [16], first the analog compensation is implemented, generating a cubic function voltage, and then the result is calibrated with the digital compensation. The analog compensation network consists of a temperature sensor and a cubic voltage generator. The temperature sensor and the memory used to store the coefficients of the compensation function constitute the digital compensation network which can be accomplished by a microprocessor [17]. The coefficients of the compensation function voltage are stored in the memory. Both compensation networks outputs are used to adjust the crystal frequency. The microprocessor generates a square-wave which is used in the compensation circuit. The output voltage varies linearly with the square-wave period and duty cycle. The output voltage, applied to the voltage controlled crystal oscillator, adjusts the crystal frequency.

### 2.1.4 Other resonators

Aiming to eliminate the need for an external resonator, such as the quartz crystal, other types of resonators have been developed to implement a precise reference frequency oscillator.

The oscillators defined in [18, 19] consist in Micro Electromechanical System (MEMS) resonators and amplifiers. A fractional-N synthesizer is used to divide the oscillator frequency to obtain the desired one [19].

Oscillators based on a Bulk Acoustic Wave (BAW) resonator are described in [20–22]. In [21] a digitally-tuned BAW-based oscillator is presented. The active circuit is implemented as a cross-coupled differential pair and two tuning capacitor banks are used to correct the oscillator frequency. A temperature compensated BAW-based oscillator is presented in [22]. The compensation parameters, calculated as a function of the temperature sensor output, are stored in a memory and used to calibrate the oscillator frequency.
In the oscillator based on a Film Bulk Acoustic Resonator (FBAR) described in [23] the core circuit is based on the Colpitts topology. Digital tuning is implemented using switched capacitors to adjust the resonator frequency.

A differential Colpitts oscillator is described in [24]. This oscillator uses a Surface Acoustic Wave (SAW) resonator and it consists of two single-ended Colpitts oscillators connected through the SAW resonator. The oscillator frequency is tuned by switched capacitors and varactors.

2.1.5 No resonator

In order to reduce oscillators form-factor, oscillators without a resonator have been reported in several works [25–28]. By integrating the resonator, the pads connecting it to the chip are not longer necessary, therefore its power consumption decrease. Oscillators without resonator can achieve a good frequency stability, however it is not enough. With these oscillators it would be difficult to achieve sufficiently low values of phase noise.

An alternative to a XO is an integrated LC oscillator [25,26]. In [25] a differential LC Colpitts oscillator with temperature compensation is implemented. A Proportional to Absolute Temperature (PTAT) voltage is used to address a Look-Up Table (LUT) where the temperature correction values are stored, which are used to adjust the oscillator frequency. The output frequency is divided using a prescaler circuit to achieve the desired frequency. A LC oscillator with a temperature compensation technique based on temperature null concept is studied in [26]. The objective is to make the oscillator temperature insensitive. A phase shift is introduced leading the oscillator to operate at the phase where the frequency temperature sensitivity is minimum.

In [27,28] relaxation oscillators are presented as another alternative to crystal oscillators. The relaxation oscillator has a RC network which defines the oscillation frequency with the time of charging and discharging the capacitor. The oscillator frequency stability is improved using composite resistors to compensate the temperature coefficient of the oscillator [27]. In [28] a temperature dependent calibration current is added to the charging current. The calibration current is controlled by a digital signal.

2.2 Ultra-low power oscillator

To increase the life-time of the battery, an IoT sensor is kept in sleep mode and it is only powered up periodically, by a wake-up oscillator. To keep track of time and generate the wake-up signal at the right moment the wake-up oscillator is always on, so its power consumption must be minimized. ULP oscillators, either with or without a resonator, are used as wake-up oscillators. There are different options for an ULP oscillator, depending on the application and the required precision, some of which are described in this section.
2.2.1 With resonator

An ULP oscillator to wake-up the system can, for instance, be based on a MEMS [29, 30] or crystal resonator [31, 32]. In [29, 30] the MEMS resonator is an interdigitated comb-drive resonator. In [29] is presented a single transistor Pierce oscillator and [30] reports a differential oscillator. A charge-pump and a ring oscillator are used as a voltage boosting circuit to reduce the device transconductance required for the oscillator to work. In [30] the voltage boosting circuit is also used to bias the temperature compensation circuit.

An ULP crystal oscillator is proposed in [31]. With the objective of reducing power consumption a duty-cycling technique is applied to the amplifier. The duty-cycle is chosen in order to maintain the oscillation amplitude high enough for the clock buffer to detect it. To reduce the power consumption, in [31] the amplifier operates at subthreshold region, therefore amplifier’s negative resistance and power consumption are more sensitive to process variations. A Successive Approximations Register (SAR) is used to calibrate the amplifier, reducing its negative resistance sensitivity to process variations. The calibration is made adjusting the size of the amplifier transistors, connecting several of them in series.

A self-charged crystal oscillator is presented in [32]. The self-charged XO uses clock slicers, pulse boosters and charging transistors to compensate the losses of the crystal. Charging transistors replace the load capacitors in the XO, delivering the required energy to the resonator. The clock slicers capture the positive clock edges which amplitude is increased by the pulse boosters. For the start-up, [32] implements a conventional XO instead of the self-charged XO. An amplifier and load capacitors are used with the crystal. When the steady-state is reached the crystal resonator is disconnected from the conventional oscillation circuit and connected to the self-charging circuit.

2.2.2 No resonator

Oscillators with no resonator have usually lower frequency stability than oscillators that use one, therefore these oscillators are periodically synchronized with a more frequency stable oscillator to calibrate its frequency. However resonator-less oscillators have a smaller area and cost when compared with other types.

Work [33] presents a charge-pump based oscillator which produces a square output. A capacitor is charged and discharged depending on the direction of the charge-pump current. Two comparators are feed the capacitor voltage and compare it with two threshold voltages generating a square-wave signal which controls the current direction. This work uses a Set-Reset (SR) flip-flop to obtain the oscillator output from both comparators outputs and two voltage references generators are needed at the comparators input.

A simplified topology of a charge-pump-based oscillator is present in [34]. Instead of two compara-
tors, a hysteresis comparator is used to control the direction of the charge-pump current. Besides the charge and discharge stages, the oscillator presented in [35] adds a precharge and a hold stage. By adding these stages, [35] turns the oscillation frequency independent to the comparator transition delay.

To improve oscillator frequency stability several works introduce calibration [34, 36, 37]. In [34] the frequency stability is improved recurring to digital calibration. Both the capacitor array and the current source are digitally controlled. Work [36] introduces a relaxation oscillator using auto-calibration. In this work a digitally controlled resistor and capacitor arrays are designed to tune the oscillation frequency. The calibration is fulfilled in two steps: first the coarse calibration which consists in varying the resistor trim bits; following the fine calibration by varying the capacitor trim bits, which produces smaller variations in the frequency. In [37] the oscillation frequency is adjusted by digitally trimming only the reference current resistor.

To decrease even further the power consumption [37] implements a current comparator with latch to substitute the voltage comparator used in previous works. The circuit consists in a reference current generator, a voltage-to-current converter and current comparator and latch which controls the capacitor charge and discharge phase, similarly to the other works presented. The current generator resistor consists of two resistors in series with opposite temperature coefficient, reducing the circuit dependence on temperature. The implementation of equal sizes transistors in the current generator and voltage-to-current converter minimizing the process variation effect.

Table 2.2 presents relaxation oscillators important parameters for comparison. Root Mean Square (RMS) jitter and the frequency coefficient of deviation, $\sigma_f/\mu_f$, calculated by dividing the frequency standard deviation by its mean, are important parameters to understand the oscillation frequency deviation.

<table>
<thead>
<tr>
<th>Reference</th>
<th>[33]</th>
<th>[34]</th>
<th>[35]</th>
<th>[36]</th>
<th>[37]</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMOS process</td>
<td>350 nm</td>
<td>130 nm</td>
<td>180 nm</td>
<td>180 nm</td>
<td>180 nm</td>
</tr>
<tr>
<td>Frequency</td>
<td>80 kHz</td>
<td>32 kHz</td>
<td>32.7 kHz</td>
<td>31.25 kHz</td>
<td>32.7 kHz</td>
</tr>
<tr>
<td>Supply voltage</td>
<td>1.0 V</td>
<td>1.2</td>
<td>0.6 V</td>
<td>1.8 V</td>
<td>0.85 V</td>
</tr>
<tr>
<td>Power consumption</td>
<td>1.14 $\mu$W</td>
<td>80 nW</td>
<td>51 nW</td>
<td>360 nW</td>
<td>54.2 nW</td>
</tr>
<tr>
<td>$\sigma_f/\mu_f$</td>
<td>3.95%</td>
<td>NA</td>
<td>NA</td>
<td>0.51%</td>
<td>0.83%</td>
</tr>
<tr>
<td>RMS jitter</td>
<td>NA</td>
<td>1.03 $\mu$s</td>
<td>NA</td>
<td>NA</td>
<td>168.5 ns</td>
</tr>
</tbody>
</table>

### 2.3 Synchronization

High frequency stability oscillators tend to have a high power consumption so ULP solutions, which are less stable, are implemented as wake-up oscillators. However, to have a reliable wake-up timer it
must have a stable and precise frequency. Calibration is the solution to this problem. The ULP oscillator is periodically synchronized with a reference oscillator reducing any frequency error in the time keeping.

Work [38] proposes a basic Phase-Locked Loop (PLL) synchronization circuit. An important part of this circuit is a Phase-Frequency Detector (PFD) which produces an error signal proportional to the phase difference between reference and ULP oscillators. The PFD output controls a charge-pump which converts the phase error into a voltage used to control the VCO, tuning its frequency. The VCO output signal is fed back to the PFD, forming a closed loop.

The wide frequency range necessary to compensate Process, Supply Voltage and Temperature (PVT) variations implies a high VCO gain, leading to poor phase noise. To cover a wide frequency range with smaller VCO gain discrete and continuous tuning mechanisms are used. Instead of a single tuning curve, it is divided into multiple sub-bands covering the desired frequency range, each with a smaller VCO gain. Several works [39–41] implement a continuous tuning performed by a PLL, but different frequency calibration techniques can be used in the discrete tuning, which can be classified as period-based or counter-based methods.

Work [39] introduces a open-loop period-based calibration method as the discrete tuning mechanism. To compared reference and feedback oscillator frequencies, their periods are measured and compared. Time-to-Voltage Converter (TVC) transforms the signals periods into voltages for comparison. These voltages are then given to a comparator, determining the oscillator frequency error. In the PLL a dual-edge phase detector detects the difference between two signals at the rising and falling edges, generating two signals, respectively, up and down. These signals are fed to a charge pump, and the circuit determines which oscillator frequency is higher, tuning the VCO control voltage. The continuous tuning mechanism operates in a closed loop.

A two-steps digital calibration system is presented in [40]. The VCO is replaced by a Current-Controlled Oscillator (CCO) preceded by voltage-to-current converters. The discrete calibration mechanism is similar to previous works: two counters and a digital comparator select the center current of the CCO, determining the control word applied to it. The second calibration step, corresponding to the continuous tuning, uses a PFD and a charge-pump to achieve the target current, a value around the one selected in the first calibration step.

In [41] a fast Automatic Frequency Calibration (AFC) is presented for the discrete tuning of the frequency. This work implements a open-loop counter-based method, the VCO is disconnected from the PLL during the discrete calibration. Two reference clock signals, with different phases, and a feedback clock signal are given to three counters, and each reference counter output is compared to the feedback oscillator counter output. According to [41] using two reference clock signals eliminates the dependency on the initial phase error. When the difference between two signals reaches two periods the frequency error is detected and the VCO control voltage updated, originating a faster calibration.
Oscillators theory

Contents

3.1 Harmonic oscillator ......................................................... 19
3.2 Relaxation oscillator ......................................................... 29
3.3 Phase noise and jitter ....................................................... 33
3.4 Period counter ................................................................. 36
Oscillators theory

An oscillator produces a periodic signal with constant frequency, without the need for any input signal. Depending on whether or not the oscillator output is sinusoidal, the oscillators are characterized into two groups: linear or harmonic oscillators and non-linear or relaxation oscillators, respectively. The first group can output a square wave resorting to an output buffer. In this chapter a brief introduction to both types of oscillators is presented, where the fundamental concepts necessary to understand their operation are explained.

Harmonic oscillators studied in this chapter use a resonator to provide the oscillation frequency, so the start-up time of these oscillators is an important aspect to study since it's largely responsible for its high power consumption. Along with the start-up time and energy, phase noise and jitter are also studied as oscillators relevant characteristics. Two methods are used to analyze the oscillators: Barkhausen criterion and two-port impedance method, being the two-port impedance method studied more profoundly.

With fast start-up time and low power consumption, at expenses of frequency stability, relaxation oscillators are good solutions to use as wake-up timers. To answer to the low frequency stability problem of the relaxation oscillator synchronization options are studied, namely a period counter to determine the number of cycles of one oscillator in a period of the other, allowing to compare and synchronize both oscillators frequency. Possible FFs and counter topologies are analyzed in this chapter, as well as the logic used to implement the circuits.

3.1 Harmonic oscillator

An harmonic oscillator can be seen as a resonator working with an active circuit which compensates the resonator losses. This resonator is used to provide a stable oscillation frequency. The quartz crystal oscillator is used as an example to explain the behavior of the resonator oscillator. The popularity of quartz crystals as resonators come from their high quality factor, $Q$. Using these resonators, the oscillators can achieve low sensitivity to PVT variations and high frequency stability, allowing better results than other resonators, such as MEMS, FBAR, SAW, BAW, or even integrated LC circuit.
3.1.1 Resonator

A resonator, namely the quartz crystal resonator exemplified in this chapter, is often an off-chip component, which can not be integrated in the chip. Therefore, one or two pads are needed to connect it to the circuit, which is one disadvantage of resonator oscillators.

A crystal resonator is a thin plate of quartz with two electrical connections, made by metalizing its opposite faces, which uses the piezoelectric property of quartz. When a voltage is applied to the crystal a physical deformation occurs, redistributing the electrical charge of the crystal. The asymmetrical displacement of charge, negative and positive charges in opposite faces of the crystal, originates a voltage between its terminals. When the applied signal reaches the crystal’s resonant frequency the oscillation amplitude is maximum [42–44].

A quartz crystal resonator can have different cut types, which results in different temperature curves, fulfilling different frequency stability specifications [45] and can operate either at the fundamental or overtone mode. Overtones are multiples of the crystal oscillation frequency, for AT-cut crystals (most common cut type of crystal which consists in cutting it at an angle of 35° from the Z-axis, producing crystals with low temperature coefficient) overtones frequencies are approximately the odd harmonics. To operate at oscillation frequencies above 50 MHz the crystal is used at overtone mode since to achieve higher fundamental frequencies the crystal becomes too thin and fragile [43].

The crystal resonator at fundamental frequency can be approximated by the Butterworth Van Dyke (BVD) model [46, 47]. For overtone mode the equivalent circuit is more complex since it’s necessary to add circuitry for each overtone. Figure 3.1 represents the equivalent circuit of a crystal resonator based on the BVD model.

![Figure 3.1: Representation of Butterworth Van Dyke model of a crystal resonator for the fundamental mode.](image)

The series of $L_m$, $C_m$ and $R_m$, called motional arm, represent the mechanical behavior of the resonator. The inductance $L_m$ is related to the mass of the resonator, capacitance $C_m$ is a function of its stiffness and resistance $R_m$ represents the losses of the crystal and mounting. The shunt capacitor $C_0$ is the capacitance between the metalized electrodes of the crystal and includes the package and stray capacitances due to mounting structure [42–44, 48].

According to the model represented in Figure 3.1 the crystal has two resonant frequencies: the series and parallel resonant frequencies, $f_s$ and $f_p$, respectively. At the crystal series resonant frequency, $L_m$
and $C_m$ are in resonance and cancel each other out, meaning the crystal circuit is equivalent to the parallel of $R_m$ and $C_0$. Series resonant frequency, $f_s$, is very close to the frequency at which the crystal is purely resistive, thus it can be considered coincident [44] and is given by

$$f_s = \frac{1}{2\pi\sqrt{L_mC_m}}. \quad (3.1)$$

The BVD model has also a parallel resonant circuit. Above $f_s$, the series of $L_m$ and $C_m$ is inductive, resonating with the parallel capacitor $C_0$ at the parallel resonant frequency, $f_p$. Similarly to the crystal behavior at $f_s$, at $f_p$ the crystals reactance is almost zero and the crystal can be considered purely resistive. The parallel resonant frequency is given by

$$f_p = \frac{1}{2\pi\sqrt{\frac{L_mC_m}{C_0+C_m}}} \quad (3.2)$$

and the relation between the operating frequency and series resonant frequency is the pulling factor,

$$p = \frac{f - f_s}{f_s} \quad (3.3)$$

Replacing $f$ by $f_p$ in (3.3) the interval between the two resonance frequencies, $\Delta f$, is determined. Since $C_0$ is much higher than $C_m$, typically 300 times higher,

$$\frac{\Delta f}{f_s} = \sqrt{1 + \frac{C_m}{C_0} - 1} \approx \sqrt{1 + \frac{1}{300} - 1} \approx 0.17\%. \quad (3.4)$$

Figure 3.2 illustrates the reactance of crystal, modeled by the BVD model, in a tight frequency range around the oscillation frequency. Both resonant frequencies are represented in Figure 3.2.

Figure 3.2: Representation of the crystal resonator reactance for a 32 MHz crystal.
Is visible in Figure 3.2 that the reactance, \( X \), is negative below \( f_s \) and above \( f_p \), meaning the crystal has a capacitive behavior in that regions. Between these two frequencies the crystal is inductive and, as referred before, the reactance and susceptance are approximately zero at \( f_s \) and \( f_p \), respectively. Is also perceptible that in the inductive region the crystal reactance evolves faster when compared to a simple inductor, which implies the crystal has a higher quality factor. The value of \( Q \) influences the oscillation grow, a higher \( Q \), necessary for a better frequency stability, means a slower oscillation start-up. Quartz crystals are characterized by a very high quality factor, in the order of \( 10^5 \) [43].

Crystals resonators can be used as series or parallel resonant crystals. A parallel resonant crystal operates between \( f_s \) and \( f_p \), in the so-called area of parallel resonance. In this type of crystals \( f_s \) is set typically 100 parts per million (ppm) below the desired oscillation frequency by the manufacturer, therefore the crystal needs a capacitor, \( C_L \), across its terminals to pull the oscillation frequency to the desired frequency. Series resonant crystals operate at the series resonant frequency, not needing any additional capacitor to pull the frequency [48].

### 3.1.2 Oscillator topologies

An oscillator can be studied as a gain element with a feedback network. Different configurations can be considered in the design of a crystal oscillator based in one of the simplest circuits, the Colpitts oscillator [49]. It can be represented as a three-point oscillator, as shown in Figure 3.3. The capacitors \( C_1 \) and \( C_2 \) are the input and output circuits, respectively, and the inductor is the feedback element.

![Figure 3.3: Representation of a Colpitts oscillator AC schematic.](image)

The Colpitts oscillator can be adapted to use a crystal resonator. The crystal can either substitute the inductor, where it must operate above the series resonance frequency and below the parallel resonance frequency (region where it appears inductive), or it can be used in the feedback network, in series with the inductor, at series resonance frequency (region where it has resistive behaviour). Numerous configurations can be obtained. In the case where the crystal replaces the inductor, grounding different terminals of the transistor originates different circuits of the Colpitts oscillator family. The grounded-source crystal oscillator, developed by G. W. Pierce [50], and the grounded-drain topology, described by
Grounded-drain crystal oscillator needs only one external pin to connect the crystal, since one of its terminals is grounded, therefore can be called a one-pin crystal oscillator. This topology suffer from a bigger influence of parasitics than the Pierce oscillator, since the ground pad is typically shared by all the circuits in a board. The disadvantage of the two external pins to connect the crystal, as in the Pierce oscillator, is that two pads and two bond wires are required. The bond wires are characterized by an inductor, which are in series with the crystal and could pull the oscillation frequency. However, crystal inductance is in the order of mH while bond wires inductance is around nH, therefore these can be despised and there is no significant frequency modification.

It would be possible to operate with the crystal in the capacitive region, however the circuit must present an inductive impedance. One possible configuration is the Hartley oscillator [51], shown in Figure 3.5.

This topology is seldom used since it requires two inductors which must be integrated, making this a disadvantageous circuit compared to the ones presented previously.
3.1.3 Oscillator analysis

In the three-point oscillator presented in subsection 3.1.2, a small perturbation in the circuit is enough to start the oscillation assuming oscillators are dimensioned to be unstable for small signals. The perturbation can be, for example, caused by noise or the circuit power-up transient. Figure 3.6 shows the evolution of the output voltage of an oscillator through time, after a perturbation has generated the oscillations. The start-up and steady-state regimes of the oscillator are indicated in the figure.

![Figure 3.6: Representation of a crystal oscillator output voltage during start-up and steady-state.](image)

Start-up is defined as the region during which the oscillations amplitude is increasing from zero towards the maximum value. After that the oscillator enters the steady-state region, where the amplitude remains constant and the signal is periodic.

One possible way to examine the circuit and to determine the necessary condition for the start-up is recurring to the feedback method. The oscillator is represented by a gain block and a feedback network, as illustrated by Figure 3.7.

![Figure 3.7: Graphical representation of three-point crystal oscillator according to the feedback method.](image)

The Barkhausen criterion states that for a circuit to be an oscillator it must present for the loop a unitary gain and a phase shift equal to 0° or an integer multiple of 360°. This is the steady-state oscillation condition. To guarantee that the oscillations start, even if there are slightly variations in the environment which change the oscillation frequency, the loop gain, \( A\beta \), must be higher than one. This corresponds to having one pair of conjugated poles at the right-half complex plane. With the loop gain higher than unity the oscillations amplitude grow, therefore a non-linear circuit is needed to limit
that amplitude. When the amplitude reaches the desired value, the loop gain equals the unity due to
the non-linear circuit, forcing the poles to the imaginary axis. At this moment the oscillator enters the
steady-state regime.

To do a more profound analysis of the crystal oscillator the one-port impedance method is used.
The crystal oscillator can be divided into two blocks: linear and non-linear behavior circuits. The shunt
capacitor $C_0$, of the equivalent crystal schematic, is considered to be in the non-linear block, along with
the transistor and capacitors $C_1$ and $C_2$. The motional arm of the crystal forms the linear block. The split
of the oscillator is represented in Figure 3.8.

![Figure 3.8: Representation of the linear and non-linear blocks of a split crystal oscillator.](image)

Considering $C_m$ and $L_m$, capacitor and inductor of the resonator motional arm, as ideal elements
the resonator impedance, $Z_m$, real part is equal $R_m$ and its imaginary part corresponds to

$$Im\{Z_m\} = X_m = j\omega L_m - \frac{j}{\omega C_m}.$$  \hspace{1cm} (3.5)

The circuit impedance, $Z_c$, can be also be divided into real and imaginary parts, $R_c$ and $X_c$. The
oscillator start-up conditions are

$$\begin{cases} 
R_c + R_m < 0 \\
X_c + X_m \approx 0.
\end{cases} \hspace{1cm} (3.6)$$

The oscillator start-up analysis is made for the critical case where the start-up condition is calculated for
small signals, using the equality

$$Z_c + Z_m = 0. \hspace{1cm} (3.7)$$

This condition must be analyzed to find the expression that gives the optimum point for start-up. From
(3.7)

$$Re\{Z_c\} = -Re\{Z_m\} = -R_m \hspace{1cm} (3.8)$$

and

$$Im\{Z_c\} = -Im\{Z_m\} = \frac{j}{\omega C_m} - j\omega L_m. \hspace{1cm} (3.9)$$
showing that if \( Z_m \) is inductive the circuit must appear capacitive.

Expanding impedance \( Z \), the circuit represented in Figure 3.9 is obtained. The circuit represents the small signal analysis of a three-point oscillator.

\[
Z_c = \frac{Z_1 Z_0 + Z_2 Z_0 + Z_1 Z_2 Z_0 g_m}{Z_1 + Z_2 + Z_0 + Z_1 Z_2 g_m},
\]

(3.10)

where \( Z_1 \) and \( Z_2 \) are the equivalent impedances of \( C_1 \) and \( C_2 \), respectively, and \( Z_0 \) is the impedance of the crystal shunt capacitance, \( C_0 \). The parasitics capacitances of the transistor are incorporated in \( Z_1 \), \( Z_2 \) and \( Z_0 \). Finally, the transconductance of the transistor \( M_1 \) is represented by \( g_m \).

Once again, \( Z_c \) can be divided into real and imaginary components, being

\[
\text{Re}\{Z_c\} = -\frac{g_m C_1 C_2}{\omega^2 (C_0 C_1 + C_1 C_2 + C_0 C_2)^2 + g_m^2 C_0^2},
\]

(3.11)

and

\[
\text{Im}\{Z_c\} = -\frac{\omega^2 (C_0 C_1 + C_1 C_2 + C_0 C_2) + g_m^2 C_0}{\omega [\omega^2 (C_0 C_1 + C_1 C_2 + C_0 C_1)^2 + g_m^2 C_0^2]}.
\]

(3.12)

To simplify the analysis of Figure 3.9 circuit, a graphical analysis is performed. The impedance \( Z_c \) is plotted in function of the parameter \( g_m \) and represented in the complex plane \([52]\). This plot can be observed in Figure 3.10\(^1\), where important points are marked to facilitate the analysis.

The function \( Z_c(g_m) \) is represented by a half circle in the third quadrant since \( Z_c \) real and imaginary parts are both negative. This function is purely imaginary at two points, corresponding to \( g_m \) equal to

\(^1\) The graph is not represented at scale, otherwise the points A and B wouldn’t be visible.
zero and infinity, meaning \( Z_c(g_m) \) is not defined in any other quadrant, as represented in Figure 3.10. These points coordinates can be calculated through (3.10) as

\[
Im\{Z_c\}|_{g_m=0} = -\frac{C_1 + C_2}{\omega(C_0C_1 + C_1C_2 + C_0C_1)} \quad (3.13)
\]

for \( g_m \) equal to zero and for when \( g_m \) tends to infinity the function can be approximated by

\[
Im\{Z_c\}|_{g_m=\infty} \approx -\frac{1}{\omega C_0} \quad (3.14)
\]

Since \( Z_m \)'s real part is constant it corresponds to a vertical line in the complex plane. Tracing a vertical line at \( -R \) the points resulting from equaling the linear and non-linear blocks impedance are obtained. The points obtained, \( A \) and \( B \) in Figure 3.10, correspond to the critical and maximum values of \( g_m \), respectively, for which the oscillation is viable. Oscillation can occur for any point of the arc between points \( A \) and \( B \).

Figure 3.11 represents the circuit of the crystal oscillator with the motional and circuit impedances mentioned previously, where the circuit impedance, \( Z_c \), is divided into the real and imaginary parts, \( R_c \) and \( C_c \).

Considering the start-up condition is verified, and using the inequality in (3.6), the current amplitude, \( I \), through crystal resonator can be calculate from Figure 3.11 and is given by

\[
I = I_0 e^{\frac{R_c - R_m}{2L_m} t}, \quad (3.15)
\]

where \( I_0 \) is the initial current amplitude, showing that during start-up the current grows exponentially [53].
From (3.15) the start-up time can be calculated as

$$t = \tau \ln \left( \frac{I}{I_0} \right),$$

being \(\tau\), the start-up time constant,

$$\tau = \frac{2L_m}{R_c - R_m}. \quad (3.17)$$

For the start-up to be as fast as possible \(\tau\) must be minimized. It should be noted that \(-R_c\) is the real part of \(Z_c\), which means the start-up time constant is minimum for \(Re\{Z_c\}_{min}\) since

$$-R_{c_{max}} = Re\{Z_c\}_{min} = \frac{C_1C_2}{2\omega C_0(C_0C_1 + C_1C_2 + C_0C_2)}. \quad (3.18)$$

Observing Figure 3.10 it is clear that this corresponds to point \(C\). Point \(C\) corresponds to the optimum value of \(g_m\), since it translates to minimum start-up time, and it can be expressed as

$$g_{m_{opt}} = \omega(C_0C_1 + C_1C_2 + C_0C_2)/C_0. \quad (3.19)$$

Substituting (3.18) into (3.17), \(\tau_{min}\) can be written as

$$\tau_{min} \approx 4\omega L_m C_0 \left( \frac{C_1 + C_2}{C_1C_2}C_0 + 1 \right), \quad (3.20)$$

assuming \(R_m\) is negligible face to \(Re\{Z_c\}_{min}\).

### 3.1.4 Quick start-up techniques

The start-up time of a crystal oscillator is in the order of magnitude of millisecond. Comparatively with the steady-state regime, during start-up the crystal oscillator power consumption is significantly higher. For these reasons, in low power applications, it's very important to reduce the start-up time of
the oscillator. There are several techniques presented in the literature, being that the more relevant ones have been presented in chapter 2.

For the start-up time to be reduced, $\tau$ must be also decreased. From (3.17) either it’s chosen a crystal with a lower $L_m$ or a higher $R_m$, or the start-up time reduction will depend on increasing $R_c$. To increase $R_c$ there are three ways: increase $g_m$; reduce $C_0$; or increase $C_1$ and $C_2$. Transconductance $g_m$ can be increased to a maximum of $g_{m,\text{opt}}$, beyond this point $R_c$ starts to decrease, as visible in Figure 3.10. Recalling (3.13) and (3.14), reducing $C_0$ increases the half-circle diameter and from Figure 3.10, a higher diameter implies a higher value of $R_c$. Finally, by increasing $C_1$ and $C_2$ the value of $R_c$ increases, as shown in (3.11).

The transistor $g_m$ depends on $I_{DS}$, the drain-to-source current. Increasing $g_m$ by providing a higher current, while decreasing the start-up time, could increase the oscillator power consumption. The parameters $L_m$, $R_m$ and $C_0$ are specified by the chosen crystal, so these are limited to the manufacturers’ products. $C_1$ and $C_2$ are also specified by the crystal manufacturers, so there are limited options.

In addition to dimensioning the circuit to have the minimum start-up time another technique may be used. Injecting a signal into the crystal, at the oscillation frequency, increases the internal signal noise, reducing the start-up time. The injected signal frequency must not differ from the desired frequency by more than 0.5% for it to work properly [8].

3.2 Relaxation oscillator

Also called non-linear, the relaxation oscillators are characterized by non-sinusoidal outputs, contrary to linear, or harmonic, oscillators which output is sinusoidal. Thus, for clock applications, where the output is a square wave, relaxation oscillators can be used. When compared to harmonic oscillators, presented in section 3.1, relaxation oscillators have a higher jitter noise, however these oscillators have a faster start-up and lower power consumption, making them viable for low-power applications.

A relaxation oscillator is characterized by the charging and discharging of one energy-storing element, which can either be a capacitor or an inductor but, for integrated applications, the capacitor is a better option. The charging and discharging time of the component translates into the circuit time constant which defines the oscillation frequency. Different topologies of relaxation oscillators can be implemented, two of which are presented in this section.

3.2.1 Ring oscillator

One of the possible implementations of a non-linear oscillator is a ring oscillator, also denominated phase-shift oscillator, characterized by the use of Complementary Metal Oxide Semiconductor (CMOS)
inverters and capacitors as the energy-storing element. The number of inverter stages in a ring oscillator is always an odd number being three the minimum to ensure Barkhausen’s criterion, specified in subsection 3.1.3. Figure 3.12 illustrates a ring oscillator composed by three inverters connected in a closed loop and respective load capacitors.

![Figure 3.12: Representation of a ring oscillator.](image)

Capacitors $C_{1,2,3}$ correspond to the output capacitance of the respective inverter plus the input capacitance of the next one. Each stage inverts its input signal adding a propagation delay for each transition. Being $t_{dhl}$ and $t_{dlh}$ the high-to-low and low-to-high propagation delay, respectively, and $n$ the number of inverter stages the oscillation frequency is given by

$$f_{osc} = \frac{1}{2n(t_{dhl} + t_{dlh})}. \quad (3.21)$$

The propagation delay is the time delay between input and output inverter signals, measure at $V_{DD}/2$, the middle value between the low and high logic levels. It can also be defined as the time required for the output voltage to reach $V_{DD}/2$ and it can expressed as [54]

$$t_{dhl} = -C \int_{V_{DD}/2}^{V_{DD}} \frac{dv_{out}}{i_{dn}}$$

and

$$t_{dlh} = C \int_{0}^{V_{DD}/2} \frac{dv_{out}}{i_{dp}}. \quad (3.23)$$

where $i_{dn}$ and $i_{dp}$ are the N-type Metal Oxide Semiconductor (NMOS) and P-type Metal Oxide Semiconductor (PMOS) drain currents which discharge and charge the capacitor, respectively. During the high-to-low transition, when $v_{out}$ is $V_{DD} - V_{thN}$, the NMOS changes from saturation to linear region of operation. Solving $i_{dn}$ and $i_{dp}$ for each region (3.22) and (3.23) become

$$t_{dhl} = -C \int_{V_{DD}}^{V_{DD} - V_{thN}} \frac{dv_{out}}{i_{dn}} - C \int_{V_{DD} - V_{thN}}^{V_{DD}/2} \frac{dv_{out}}{i_{dn}}$$

and

$$t_{dlh} = C \int_{0}^{-V_{thP}} \frac{dv_{out}}{i_{dp}} + C \int_{-V_{thP}}^{V_{DD}/2} \frac{dv_{out}}{i_{dp}}. \quad (3.25)$$
respectively, with $V_{th_{N,P}}$ as the NMOS and PMOS threshold voltages. Saturation and linear region drain currents are defined as

$$i_{dn_s} = \frac{g_N(V_{DD} - V_{thN})^2}{2}$$  \hspace{1cm} (3.26)

and

$$i_{dn_l} = g_N v_{out} \left( V_{DD} - V_{thP} - \frac{v_{out}}{2} \right)$$  \hspace{1cm} (3.27)

for the NMOS transistor. The PMOS drain current can be expressed as

$$i_{dp_s} = \frac{g_P(V_{DD} - V_{thP})^2}{2}$$  \hspace{1cm} (3.28)

for the saturation region and as

$$i_{dp_l} = g_P(V_{DD} - v_{out}) \left( V_{DD} + V_{thP} - \frac{V_{DD} - v_{out}}{2} \right)$$  \hspace{1cm} (3.29)

for the linear region. Parameters $g_N$ and $g_P$ are the NMOS and PMOS transconductance.

Computing (3.24) and (3.25) the propagation delays dependence on the inverter parameters is obtained as

$$t_{dhl} = \frac{C}{g_N(V_{DD} - V_{thN})} \left( \frac{2V_{thN}}{V_{DD} - V_{thN}} + \ln \left( \frac{3V_{DD} - 4V_{thN}}{V_{DD}} \right) \right)$$  \hspace{1cm} (3.30)

and

$$t_{dlh} = -\frac{C}{g_P(V_{DD} + V_{thP})} \left( \frac{2V_{thP}}{V_{DD} + V_{thP}} + \ln \left( \frac{3V_{DD} + 4V_{thP}}{V_{DD}} \right) \right).$$  \hspace{1cm} (3.31)

### 3.2.2 Charge-pump oscillator

In a charge-pump oscillator, comparators are used to verify if the capacitor voltage is between determined thresholds, adjusting the oscillator output accordingly. The capacitor charge and discharge stage is controlled by the output signal which is feedback to the capacitor switch.

The possible comparator topologies to be used in this oscillator are either open-loop comparators or regenerative comparators. The first type does not have any feedback whereas the latter one has positive feedback. An example of an open-loop comparator is a two-stage OpAmp. Due to the high gain of the OpAmp the voltage transfer curve of the comparator is almost ideal, switching almost instantly between the two output voltage values. A regenerative comparator can be implemented as a latched comparator which implementation consists in two inverters connected in loop, back-to-back. The simplified schematic of a charge-pump oscillator is presented in Figure 3.13.

The circuit includes two comparators to decide if the capacitor voltage is between the defined reference voltages. The capacitor is charge or discharge by one of the current sources, depending on the
SR latch output. Comparator 1 output is high if the capacitor voltage goes below the reference level $V_{\text{min}}$, resetting the SR latch. When the capacitor voltage becomes larger than $V_{\text{max}}$ comparator 2 sets the circuit output to high. The SR latch output, $v_{\text{out}}$, controls the capacitor switch so the capacitor is charged when the output is low and discharged when it is high. A representation of the voltage across the capacitor and at the oscillator output is presented in Figure 3.14.

The capacitor voltage varies between $V_{\text{min}}$ and $V_{\text{max}}$. At every transition of the oscillator output voltage the switch goes to the other position, changing the current source biasing the capacitor. The
oscillator output is a square wave generated by the SR latch which frequency,

\[ f = \frac{I_{\text{ref}}}{2C\Delta V_c} \]  

(3.32)
depends on the reference current, \( I_{\text{ref}} \), the capacitor value, \( C \), and the capacitor voltage variation, \( \Delta V_c \).

### 3.3 Phase noise and jitter

Noise is a random perturbation of a signal which manifests in the signal amplitude or phase. There are several noise sources in an oscillator, such as thermal noise, shot noise and flicker noise, generating these perturbations. Since the amplitude of the output signal is controlled and stabilized by the non-linear behavior of the oscillator the phase noise is more relevant than the amplitude noise. The random perturbation on a signals phase also creates a random perturbation on its period. The first effect is usually studied on the frequency domain and it is denominated phase-noise, the second one on the time domain and is called jitter. Phase noise is important for analog signals analysis, and jitter is the relevant parameter for digital signals.

In the frequency domain, an ideal oscillator output power spectrum is concentrated at the oscillation frequency. When the signal has phase noise the energy is spread for the adjacent frequencies and the frequency spectrum corresponds to the one represented in Figure 3.15.

![Figure 3.15](image)

**Figure 3.15:** Representation of an oscillator output power spectrum.
The phase noise, $L$, is defined as the noise power in a bandwidth of value $B$, typically 1 Hz, centered at a $\Delta f$ offset from the carrier, $f_0$, and can be expressed as

$$L = \frac{P_n/B}{P_s},$$

(3.33)

where $P_s$ is the oscillator output power and $P_n$ is the noise power at $\Delta f$ from the carrier. The units are dBc/Hz, being dBc the dB relative to the carrier. Figure 3.16 represents an oscillator single-sideband phase noise.

Figure 3.16: Representation of Leeson’s model oscillator phase noise.

The phase noise decreases as the offset frequency increases, reaching a noise floor value, for frequencies further away from the oscillation frequency. The spectrum is divided into three bands: to frequencies closer to the oscillation frequency the phase noise has the characteristic of a $f^{-3}$ function; the next frequency band represented in Figure 3.16 follows a $f^{-2}$ behavior; and finally, for frequencies further away from the oscillation frequency, the phase noise is represented as a constant value, corresponding to the flat band.

Different mathematical models can be used to describe and analyze the oscillator sideband phase noise, being the most commonly used the Leeson’s model [55]. This model considers the oscillator can be modeled by an amplifier with positive feedback through a second order resonator.

According to Leeson’s model the oscillator phase noise, $L$, can be mathematically characterized by

$$L(\Delta f) = 10 \log \left[ \frac{2FkT}{P_s} \left( 1 + \frac{f_0}{2Q_L \Delta f} \right)^2 \left( 1 + \frac{\Delta f^{-3}}{|\Delta f|} \right) \right],$$

(3.34)
which can be separated into three terms. For frequencies further away from oscillation frequency, the first term prevails, given origin to the flat region of the function which is proportional to the amplifier noise figure, $F$, the Boltzmann’s constant, $k$, and the temperature, $T$ and inversely proportional to $P_s$, the oscillator output power. For frequencies in the middle band, Leeson’s function varies according to $f_0/2Q_L \Delta f$, where $Q_L$ is the loaded quality factor of the resonator and $f_0/2Q_L$ represents half the resonator bandwidth. Finally, for frequencies closer to the oscillation frequency, the dominant term is $\Delta f_{f-3}/|\Delta f|$, where $\Delta f_{f-3}$ corresponds to the corner frequency that separates $f^{-3}$ and $f^{-2}$ regions.

Since jitter is represented in time domain it is more appropriated than phase noise to analyze square waves. Jitter takes into account the wave middle value crossings of the rise and fall edge of the output signal. Faster transitions imply less jitter, since there is less time for the perturbations to affect the signal. A representation of jitter in a square wave is shown in Figure 3.17.

![Figure 3.17: Representation of jitter noise in a square wave.](image)

Jitter characterization can be divided into different types: cycle-to-cycle jitter, period jitter, long-term jitter and phase jitter. To measure cycle-to-cycle jitter several pairs of adjacent cycles are compared and the maximum deviation value of two consecutive cycles is the cycle-to-cycle jitter. Period jitter is defined as the change along randomly selected cycles when compared with the ideal cycle. Long-term jitter, also known as accumulated jitter, is identical to period jitter but the analysis is performed over consecutive cycles. Finally, phase jitter measures the deviation from the ideal signal period, in time domain.

Phase jitter corresponds to the integration of phase noise over a certain frequency band, usually between 12 kHz and 20 MHz. It’s typically given by its RMS value and it is presented in seconds. It can be calculated as the area bellow phase noise function between the selected frequencies, as shown in Figure 3.18.

The RMS value of phase jitter is calculated as

$$PJ_{RMS} = \frac{1}{2\pi f_0} \sqrt{\frac{2}{f_2} \int_{f_1}^{f_2} \left| \frac{\epsilon(\Delta f)}{10} \right|}$$

(3.35)

where $f_0$ is the central frequency, at which the oscillator operates, and the integration limits are the frequencies represented in Figure 3.18 as the area boundaries.
3.4 Period counter

This section studies period counters, or frequency dividers, as important circuits in the synchronization of two signals. To obtain a period counter edge-triggered FFs are used. This FF arrangement presents, at its output, a signal which frequency is a sub-division of the input signal frequency. Depending on how the clock signal is fed to the flip-flops, counters can be classified as synchronous or asynchronous: in a synchronous counter the same clock signal is applied to all the FFs; while in an asynchronous counter (also denominated ripple counter) each FF clock is driven by the previous FF output signal. Because each FF is supplied with a different clock signal the output suffers from delays. Figure 3.19 presents an asynchronous period counter with two FFs and respective relevant signals.

The configuration presented in Figure 3.19a represents an asynchronous counter, ou ripple counter, where each FF output triggers the following one. In each FF, a feedback loop connects the inverted output, $\overline{Q}$, to the input terminal $D$, generating a signal at $Q$ with half the frequency of the input clock signal. Since each FF divides the input signal frequency by 2, a divider with $n$ FFs outputs a frequency of $f_{in}/2^n$, being $f_{in}$ the input clock signal frequency. The counter output is a $n$-bits digital word, where $n$ is the number of FFs, being the last FF output the Most Significant Bit (MSB). The represented counter is a up counter, all the output bits start at zero and the count increases by one at each positive edge of the clock.

Memory gates, such as FFs, can be implemented by static or dynamic logic [56]. Static logic preserves the output value until the power is turned off, and it uses positive feedback. Dynamic logic is
based on charge storage, and it holds the output value for a limited time due to current leakage. The clock signal must refresh the circuit nodes with a frequency high enough to prevent the node voltages to decay to low values. Circuits built with dynamic logic are, usually, simpler, faster and with lower power dissipation. Both logic options are explored in subsections 3.4.1 and 3.4.2.

### 3.4.1 Static logic

Static CMOS gates have each node connected to either the supply voltage or ground, except during switching transient. There are no floating nodes in the circuit. When switching both NMOS and PMOS transistors are on, resulting in a low resistance path between the supply voltage and ground, which generates a current peak. When in a stable state, the circuit power consumption is given by the static power, which corresponds to the circuit current leakage times the supply voltage.

Static logic memory gates are based on positive feedback, for example, a SR-latch can be obtained cross-coupling NAND gates. Introducing a clock signal, the circuits become more complex, and different topologies can be used to implement a flip-flop. The first topology presented is implemented using
NAND gates with two and three ports. This circuit topology, presented in Figure 3.20 has a total of 28 transistors.

![Figure 3.20: Representation of a flip-flop using NAND gates.](image)

Figure 3.20 exemplifies two versions of a master-slave FF topology. Both versions consist in two latches connected in chain. To achieve a positive edge-triggered FF, a negative latch as master stage and a positive latch as slave stage are cascading. The first version illustrated uses two latches each implemented with two cross-coupled inverters and transmission gates; the second version is built with two cross-coupled inverters and additional transistors for clock control.

The two master-slave circuits represented in Figure 3.21 have 16 transistors each and require two non-overlapping clock signals, $Clk$ and $\overline{Clk}$. Additional circuitry is necessary to generate the non-overlapping clock signal, making the circuits more complex.
3.4.2 Dynamic logic

Unlike the static logic, dynamic logic is based on charge storing in a capacitor. Leakage current reduce the capacitor charge, decreasing the node voltage. Therefore, the clock signal frequency should be sufficiently high to periodically update the circuit node voltages before it drops to ambiguous values. Dynamic flip-flops are simpler circuits, with less transistors than the static logic gates and less power dissipation, allowing faster circuits.

Over time several examples of dynamic logic FFs have been implemented, aiming to obtain faster circuits with lower power consumption. Figure 3.22 presents a master-slave circuit based on transmission gates.
When compared to the static logic transmission gate-based FF, this circuit is much simpler, having only 8 transistors. The capacitors illustrated in Figure 3.22 represent the equivalent capacitance of the respective transmission gate and inverter.

The circuit operation can be divided into two modes: hold and sampling mode. When the signal Clk is low, the input value is stored in the X node and the Y node is at high impedance state, the master stage is on sampling mode and the slave stage is on hold mode; the output holds its value. In the rising edge of the clock signal, the X node value propagates to the output and the master-slave transmission gate turns off; the master stage is on hold mode and the slave stage is on sampling mode. The clock signals must be non-overlapping; in an overlap situation the circuit would be reduced to two inverters, meaning every change in the input would be translated to the output.

A similar method to implement a dynamic FF but without the need for non-overlapping clock signals is the Clocked Complementary Metal Oxide Semiconductor (C²MOS) logic. C²MOS circuitry consists in an inverter and two MOS switches connected in series [58]. A C²MOS-based edge-triggered FF is illustrated in Figure 3.23.
Similarly to the transmission gate-base FF, this topology has two operating modes: hold mode and evaluate mode. When the $Clk$ signal is low, master stage is at evaluate mode, operating as an inverter; while the output node is at high impedance, meaning the slave stage holds the previous output value. For a high signal $Clk$, X node is at high impedance mode and the master stage holds the X node value (input changes are not propagated); slave stage acts as an inverter, updating the output value.

Although this circuit needs two complementary clock signals, they do not need to be non-overlapping, as long as the clock signal rise and fall times are sufficiently low. In case this condition is not verified, there is a time period where both MOS switches are on, creating a path between the input and output nodes, propagating an unwanted value to the output. In the case of overlapping situations, depending on the input value, either the master or the slave stage are at high impedance, meaning the circuit holds the previous output value.

A new dynamic logic topology is depicted in [59], presenting a simpler option, with a single clock signal. True Single-Phase Clock (TSPC) logic eliminates the need for complementary clock signals, simplifying the generation and distribution of this signal. Two examples of a TSPC latch are presented in Figure 3.24.

![Figure 3.24: Representation of two versions of TSPC FF.](image)

With a single clock signal, when the $Clk$ is high both stages operate as inverters, propagating the input to the output; in the time period the $Clk$ is low the output node is at high impedance, holding its value, since both stages are off. The two stages of this TSPC latch are denominated by N-C$^2$MOS stages - the clocked transistor is a NMOS.

In the case of the simplified latch circuit, Figure 3.24b, there exists only one clocked transistor. The operation is similar to the first version: for a high signal $Clk$ the two stages act as inverters and the output reproduces the input value; when the signal $Clk$ turns low, the first stage output node is at high impedance, maintaining the output value unchanged.

A single-phase edge-triggered FF can be implemented using TSPC logic, cascading a positive and a negative latch, obtaining a 12-transistors circuit. Figure 3.25 illustrates a more effective FF topology, still
based on TSPC logic, but with only 9 transistors \cite{60}. This three-stage flip-flop consists in a P-C\textsuperscript{2}MOS stage, a N-precharge stage and a N-C\textsuperscript{2}MOS stage, represented in Figure 3.25 in this order, from left to right.

![Dynamic TSPC positive edge-triggered FF.](image)

The second stage is either in precharge or evaluate mode. When $Clk$ is low the first stage samples the input, inverting it; the second stage charges $Y$ node to the supply voltage; and, finally, the last inverter is off, putting the output in a high impedance state, and holding the output value. On the rising edge of the clock, the P-C\textsuperscript{2}MOS stage is off. The second stage evaluates the $X$ node and, depending on its value, the $Y$ node is discharged or kept in a high impedance state. The third stages passes the $Y$ node value to the output, $X$ node value must be kept constant while the $Y$ node value is being propagated. On the high phase of the clock the $X$ node is in a high impedance state, meaning any change in the input is disregarded. The circuit output corresponds to the FF complementary output, $\overline{Q}$. If the $Q$ output is needed an inverter must be added to the circuit.
4

Clock generator system design

Contents

4.1 Reference oscillator .................................................. 46
4.2 Ultra-low power oscillator ........................................... 54
4.3 Synchronization ......................................................... 66
Clock generator system design

A clock signal must be precise and stable, and must be generated consuming minimum possible power. Reference oscillators, widely used as clock generators since they produce a very frequency stable signal, typically have high power consumption. In this application, where the clock signal must always be active, this oscillator type alone is not the solution.

To achieve a solution with the lowest possible power consumption, a system is designed where the reference oscillator is turned on only during data communication time slots, short periods of time when the frequency stability is of extreme importance. Outside these time slots, the reference oscillator is powered off and an ULP relaxation oscillator is used to keep track of time. This oscillator has a significantly lower frequency stability when compared to the reference oscillator, however, it is enough to know when it’s time for the next data communication and turn on the reference oscillator. To improve the ULP oscillator frequency stability, a synchronization with the reference oscillator is made every time the latter is powered on.

A clock generator system, composed by a reference frequency oscillator, a RCO and the calibration circuitry, is implemented in UMC130 technology, from United Microelectronics Corporation, and simulated using Cadence software. The individual circuits are then connected together as illustrated in Figure 4.1 and the clock generator system is simulated.

![Figure 4.1: Clock generator system block diagram.](image)

A XO is chosen to generate the frequency reference due to its high frequency stability and, due to its high quality factor, it can maintain the oscillation spending less energy than other references. The XO has the disadvantage of using an external resonator, which implies the use of two additional pads, however, when compared with other resonators, quartz crystals have a compact size and low cost. The
XO frequency is selected as 32 MHz, a standard frequency for this type of oscillators and also imposed by the sensor node receiver demodulator.

To achieve synchronization between the oscillators both their frequencies must be multiple of each other. Therefore, the RCO frequency is selected as 31.25 kHz, 1024 times slower than the reference oscillator. The wake-up timer must allow digital calibration and output a square wave signal with minimum power consumption, therefore, the topology selected is a digitally-controlled RCO, which frequency can be tuned adjusting the current charging/discharging the capacitor through a digital word from the calibration logic block.

A period counter, an essential circuit of the calibration block, counts the number of periods of the crystal oscillator during one cycle of the RCO. Any value different from 1024 implies an error in the RCO frequency. This error is given to the calibration logic block where an algorithm decides the control word to calibrate the RCO with the minimum iterations possible. After the calibration, all blocks, except the RCO, must be powered off.

Some care is taken in the dimensioning of the circuits to minimize power consumption and area and, in the case of the crystal oscillator - the reference frequency oscillator - to minimize the startup energy. The objective is to analyze the behavior of each system block, individually and altogether, simulating and redesigning the circuits until a version which fulfills the specifications is achieved.

4.1 Reference oscillator

Two solutions for the crystal oscillator are studied in this chapter. A first version using an inverter as the active element is implemented and analyzed to understand the oscillator behaviour and the circuit elements influence at the XO output signal and performance. A single-MOS XO topology is implemented as the second and final version of the reference oscillator. This topology allows to add the power-down control to the XO. After the single-MOS XO is dimensioned, a voltage buffer is introduced at its output, originating a square signal.

The first version of the implemented XO follows a Pierce topology, using a CMOS inverter as active device to compensate the crystal losses. The oscillator schematic can be seen in Figure 4.2, with the crystal represented by its equivalent circuit.

The chosen crystal unit is FA128 from Epson Seiko [61] and its model equivalent schematic and components values are visible in the schematic. An important factor in the choice of the crystal is the load capacitor value, which should be minimum to decrease the power consumption, and circuit area since the capacitors represent a significant part of the chip. A 6 pF load capacitor is divided in two 12 pF capacitors in series, $C_1$ and $C_2$, assuring the circuit symmetry.

The feedback resistor, $R_{fb}$, inserted in parallel with the crystal generates the biasing voltage for the
PMOS. This resistor value is proportional to the loop gain, with an insufficient loop gain the crystal oscillator would have a long start-up time or would even fail to start the oscillations.

The dimensioning of the crystal oscillator starts with the inverter. The inverter transistors are dimensioned to center the Voltage Transfer Characteristic (VTC) at half the value of $V_{DD}$. This characteristic of the inverter is adjusted according to

$$V_M = \frac{rV_{DD}}{1 + r},$$

(4.1)

where $V_M$ corresponds to the point where the input and output voltage are equal, and

$$r = \frac{k_pV_{Dsat_p}}{k_nV_{Dsat_n}}.$$  

(4.2)

The parameter $k_{p,n}$ represents the ratio between width, $W$, and length, $L$, of PMOS and NMOS, respectively [56].

To achieve the fastest start-up possible, the inverter must provide optimum transconductance value, $g_{m_{opt}}$, calculated using (3.19). The inverter total transconductance is the sum of both transistors transconductance. Knowing the start-up energy

$$E_{startup} = \int_0^{t_{startup}} v_{dd}(t)i_{dd}(t)dt,$$

(4.3)

and that $i_{dd}$ is related to the transistors drain current $i_{ds}$ (and therefore with the transistor $g_m$) the minimum start-up energy occurs for the minimum transconductance which guarantees the oscillation, $g_{m_{min}}$. A transconductance value below $g_{m_{opt}}$ is achieved with transistors with smaller width and higher gate
length, giving rise to a static characteristic $v_o(v_i)$ which is more vertical in the transition zone, and, consequently, a inverter with faster transitions. Table 4.1 presents a comparison for three transistors sizing, where it is verified that there is a trade-off between start-up time and start-up energy.

Table 4.1: Start-up time and energy trade-off.

<table>
<thead>
<tr>
<th>$W_p$ [µm]</th>
<th>$W_n$ [µm]</th>
<th>$L_{p,n}$ [nm]</th>
<th>$g_m$ [mS]</th>
<th>$t_{\text{startup}}$ [µs]</th>
<th>$E_{\text{startup}}$ [nJ]</th>
<th>$P_{\text{startup}}$ [mW]</th>
</tr>
</thead>
<tbody>
<tr>
<td>40</td>
<td>8</td>
<td>400</td>
<td>4.69</td>
<td>750</td>
<td>378</td>
<td>0.50</td>
</tr>
<tr>
<td>68</td>
<td>16</td>
<td>250</td>
<td>12.98</td>
<td>420</td>
<td>459</td>
<td>1.09</td>
</tr>
<tr>
<td>70</td>
<td>18</td>
<td>200</td>
<td>16.87</td>
<td>445</td>
<td>565</td>
<td>1.27</td>
</tr>
</tbody>
</table>

The optimum $g_m$ value, which allows oscillation to build with lowest start-up time, calculated through (3.19), is 12.8 mS. The simulated $g_m$ of 12.98 mS translates to the minimum start-up time solution. Although the start-up time is higher for the solution with the lower $g_m$, the current, and therefore power consumed, is lower than for the other solutions. This translates into less energy spent in the start-up of the oscillator and, since the objective is to obtain a low-power clock generator system and the energy spent is a major design concern, it is the best solution for this low-power application.

After dimensioning the inverter, the crystal oscillator in Figure 4.2 is simulated to analyze its performance. A transient analysis shows the output voltage of the crystal oscillator, from start-up to the steady-state sinusoidal wave with rail-to-rail excursion. The simulation results for different $R_{fb}$ values are shown in Figure 4.3.

Figure 4.3: Crystal oscillator output voltage variation with $R_{fb}=1$ kΩ, 100 kΩ and 20 MΩ.

Figure 4.3 demonstrates the feedback resistor impact on the start-up time. For too low values of $R_{fb}$, 1 kΩ, the loop gain is insufficient, limiting the output voltage, which keeps the oscillator from starting. To $R_{fb}$ equal to 10 kΩ the oscillator starts but the output voltage doesn’t reach $V_{DD}$. $R_{fb}$ should be high
enough to guarantee sufficient loop gain to start the oscillations, biasing the inverter in active region, only for \( R_{fb} \) higher than 20 M\( \Omega \) this condition is verified. The output voltage is, in fact, higher than \( V_{DD} \) and lower than ground due to the presence of capacitors in the circuit.

Defining the start-up time as the time it takes for the output signal amplitude to reach 90\% of its final value, in the solution presented in Figure 4.3 for 20 M\( \Omega \) it is taken as 750 \( \mu s \) (as expressed in Table 4.1). To assure the output signal is ready to be used from this moment it is important to verify if the oscillation frequency is stabilized. Figure 4.4 represents the output signal transient and its frequency evolution with time.

**Figure 4.4:** Crystal oscillator output voltage and frequency transient.

The oscillator is used to demodulate a signal which has frequency modulation, therefore it is important to guarantee the oscillation frequency is close enough to its final value. For Figure 4.4 at 750 \( \mu s \) the oscillation frequency deviation is approximately 230 Hz, which translates in a frequency deviation of 7.2 ppm face to its final value.

Completed the XO studied using the inverter-based topology the second version is implemented. A single NMOS transistor, biased by a current source, is used as active element in this version of the XO implementation represented in Figure 4.5. Power-down is added to allow the use of a duty-cycle scheme in order to save energy.

Although this topology requires biasing circuitry, unlike the XO inverter-based, it gives the possibility to improve the oscillator performance at the expense of using a more complex circuit. This topology allows more easily for quick start-up techniques to be added. The large value feedback resistor is implemented with transistor NM2 operating in weak-inversion, biased by transistors NM3 and NM4. To power-down the oscillator, PM4 imposes zero voltage at \( V_{SG} \) of current mirror transistors and the bias.
resistor is disconnected. When the power-down signal, \( pwr \), turns low the circuit enters into power-down mode, reducing the total current to 716 pA.

Several solutions are simulated and the one which presents a lower start-up energy is selected as the preferred solution. Figure 4.6 illustrates the crystal oscillator output voltage and frequency for this solution.

Besides the start-up time, an important characteristic of the crystal oscillator is its phase noise, and consequently, its phase jitter. The analysis presented in Figure 4.7 indicates the output phase noise variation in terms of the frequency offset and allows the calculation of phase jitter.
To calculate the oscillator phase jitter some points are retrieved from the phase noise graph and used in Abracon’s phase jitter calculator [62], with an integration band between 12 kHz and 20 MHz. Crystal oscillator RMS jitter for this integration band is 51.7 fs. This and other relevant parameters of crystal oscillator performance are summarized in Table 4.2.

Table 4.2: Single-MOS crystal oscillator performance summary.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$t_{\text{start-up}}$ [µs]</td>
<td>660</td>
</tr>
<tr>
<td>$E_{\text{start-up}}$ [nJ]</td>
<td>640</td>
</tr>
<tr>
<td>$P_{\text{steady-state}}$ [µW]</td>
<td>899</td>
</tr>
<tr>
<td>$I_{\text{power down}}$ [pA]</td>
<td>716</td>
</tr>
<tr>
<td>Phase noise (@1 kHz) [dBc/Hz]</td>
<td>-116.26</td>
</tr>
<tr>
<td>RMS jitter [fs]</td>
<td>51.7</td>
</tr>
</tbody>
</table>

The values presented in Table 4.2 correspond to the final version of the implemented crystal oscillator. The start-up time and energy values are measure when the output voltage amplitude is 90 % of its final value.

The crystal oscillator of Figure 4.5 is simulated for PVT variations to understand how it performs under non-ideal conditions. The oscillator is simulated for a 10% variation of the supply voltage and temperature between -40 °C and 125 °C. The process variation covers all the on-chip components: transistors, capacitors and resistor. The output voltage response to these variations is simulated.

The crystal oscillator operates correctly to all the dispersion corners. The XO start-up time varies between 414 µs and 2.1 ms with start-up energy of 629 nJ and 829 nJ, respectively. For the corners with higher $V_{DD}$ the result is an output wave with faster start-up time. The oscillation frequency has
a variation of ± 125 ppm face to the nominal value. The biggest deviation in the frequency is caused by variations in the capacitors value, since the capacitors are put in the circuit to pull the frequency to the desired value. Regarding the phase noise, the PVT variation is between -128.83 dBC/Hz and -50.92 dBC/Hz.

A voltage buffer, represented in Figure 4.8, is introduced at the XO output to obtain a square signal, so it can be used as a clock signal. The buffer is based on a conventional CMOS Schmitt trigger with feedback to improve the output voltage transitions [63].

The feedback loop is implemented with two back-to-back inverters, represented by transistors NM3, NM4, PM3 and PM4. The buffer circuit is dimensioned for a 5 pF load capacitor. With the XO at the buffer input, its output is simulated and shown in Figure 4.9, along with the input signal. The sine wave corresponds to the crystal oscillator output - voltage buffer input - and the square wave depicts the buffer output signal.

To guarantee 50 % of duty-cycle, the difference from both threshold voltages to $V_{DD}/2$ must be equal. Another option would be to a buffer implemented with a latched comparator. When using this type of buffer there would be only one threshold voltage, however, if this threshold deviates from $V_{DD}/2$ the duty-cycle would deviate from 50 %. In this application, the Schmitt trigger implementation is selected due to a lower power consumption, still achieving fast transitions. With the load capacitor of 5 pF the buffer power consumption is 464 µW.
Comparing with the results obtained for the crystal oscillator, in the nominal corner, the buffer increases the RMS jitter to 938 fs, with a phase noise at 1 kHz of -115.4 dBc/Hz. The buffer outputs a square wave signal with a duty-cycle of 49.9% and rise and fall times of 0.84 ns and 0.5 ns, respectively.

The complete XO with the voltage buffer at its output is simulated for PVT variations. The frequency deviation for temperature deviation between -40 °C and 125 °C and for 10% variation of the supply voltage is represented in Figure 4.10.

From the temperature variation the circuit registers a frequency deviation of -1.3 ppm/°C. For the
voltage supply variation the frequency has a deviation of 7.85 kHz which translates to a -0.1 \%/V deviation.

The output signal duty-cycle varies between 6.4 \% and 62.5 \%. This high variation represents a problem when using this signal as a clock signal and, therefore, it needs to be solved. One possible solution would be to use a duty-cycle corrector, a circuit which takes an input signal with a duty-cycle different from 50\% and corrects it to that value.

Table 4.3 summarize important parameters obtained for the implemented crystal oscillator with buffer and its comparison with several works presented in literature and a XO from manufacturer Epson Seiko.

### Table 4.3: Comparison of crystal oscillator with state of the art works.

<table>
<thead>
<tr>
<th>Reference</th>
<th>This work</th>
<th>[2]</th>
<th>[7]</th>
<th>[9]</th>
<th>[10]</th>
<th>[64]</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMOS process [nm]</td>
<td>130</td>
<td>65</td>
<td>180</td>
<td>65</td>
<td>90</td>
<td>NA</td>
</tr>
<tr>
<td>Frequency [MHz]</td>
<td>31.97</td>
<td>26</td>
<td>39.25</td>
<td>50</td>
<td>24</td>
<td>32</td>
</tr>
<tr>
<td>Supply voltage [V]</td>
<td>1.2</td>
<td>1.8</td>
<td>1.5</td>
<td>1.0</td>
<td>1.0</td>
<td>1.8</td>
</tr>
<tr>
<td>Start-up time [µs]</td>
<td>660</td>
<td>580</td>
<td>555</td>
<td>3200</td>
<td>158</td>
<td>2.2</td>
</tr>
<tr>
<td>Start-up energy [nJ]</td>
<td>640</td>
<td>1650</td>
<td>621</td>
<td>6910</td>
<td>350</td>
<td>10</td>
</tr>
<tr>
<td>Steady-state power [µW]</td>
<td>899</td>
<td>2910</td>
<td>1164</td>
<td>2520</td>
<td>180</td>
<td>200</td>
</tr>
<tr>
<td>Phase noise (@1 kHz) [dBc/Hz]</td>
<td>-116.3</td>
<td>-115.4</td>
<td>-117.3</td>
<td>-136.1</td>
<td>-147</td>
<td>NA</td>
</tr>
<tr>
<td>RMS phase jitter (12 kHz-20 MHz) [fs]</td>
<td>51.7</td>
<td>938</td>
<td>914</td>
<td>420</td>
<td>(10 kHz-5 MHz)</td>
<td>NA</td>
</tr>
<tr>
<td>Quick start-up</td>
<td>no</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
</tr>
</tbody>
</table>

\(^a\) including buffer and load capacitor of 5 pF
\(^b\) including buffer and load capacitor of 500 fF
\(^c\) calculated

The XO with buffer is simulated with two different values of load capacitor, the capacitor $C_0$ presented in Figure 4.8. During the design the voltage buffer is implemented using a 5 pF load capacitor. Such a high value is chosen to guarantee the correct operation of the buffer when introduced in the complete system. The disadvantage of dimensioning this capacitor for excess is that the simulated power consumption is significantly higher. With this new load capacitor value the buffer power consumption is decreased from 464 µW to 138 µW. To be able to compare the implemented XO and buffer to works presented in literature the results obtained for a load capacitor of 500 fF are added to Table 4.3.

### 4.2 Ultra-low power oscillator

The charge-pump conventional topology is used as first version of the ULP oscillator to analyze its power consumption and to understand of how much it can be reduced. However, this topology is likely to have some problems, namely that the comparators suffer with PVT variations, causing errors in the
thresholds and the oscillation frequency. A second version using current comparators, and which should result in better performance, is designed. A lower power consumption and lower sensibility to PVT variations are expected.

The conceptual idea in which the ULP oscillator is based is represented in Figure 4.11. This charge-pump oscillator - or relaxation oscillator - which provides a square wave at its output without the need for additional circuitry, used as reference a oscillator presented in literature [33]. The relaxation oscillator main design concern is power consumption since it is always on, keeping track of time, to control the power-down signal sent to the other blocks.

The charge-pump oscillator has its frequency defined by the charge and discharge time of the capacitor $C_0$. The capacitor voltage is compared with both thresholds, $V_{ref^-}$ and $V_{ref^+}$, and $C_0$ is charged or discharge according to the comparison result. For a capacitor voltage value higher than $V_{ref^+}$ the bottom comparator output goes high and sets the SR latch. Otherwise, for a capacitor voltage value below $V_{ref^-}$ the top comparator output resets the SR latch. Two current sources, each of amplitude $I_{ref}$, charge and discharge the capacitor. The current sources are represented in Figure 4.11 by the PMOS and NMOS named PM1 and NM1, respectively.

This oscillator is chosen to operate at a frequency power of two multiple of 32 MHz so it can be

![Figure 4.11: Conventional relaxation oscillator schematic.](image-url)
synchronized with the reference oscillator using a binary counter for the comparison, simplifying the counter implementation. A frequency $1/1024$ of $32 \text{ MHz}$ is chosen, which means it should operate at $31.25 \text{ kHz}$. To choose the capacitor value it’s necessary to define the reference current, $I_{ref}$, and the reference voltage variation, $\Delta V$. According to (3.32), for a frequency of $31.25 \text{ kHz}$, and defining $I_{ref}$ as $10 \text{ nA}$ and $\Delta V$ as $200 \text{ mV}$, the capacitor value must be $800 \text{ fF}$. To have $200 \text{ mV}$ of voltage variation, the minimum voltage reference, $V_{ref}^-$, is set to $500 \text{ mV}$ and the maximum voltage reference, $V_{ref}^+$, is set to $700 \text{ mV}$.

Four different comparators topologies are implemented and simulated. Every circuit starts with ideal components that are then replaced by technology components. The first topology to be tested is the two-stage OpAmp which schematic is represented in Figure 4.12.

![OpAmp comparator schematic.](image)

The first stage is a differential pair and it’s chosen a PMOS differential pair since the input voltage reference is a high value. The second stage of the OpAmp is a high gain stage. The transistors are dimensioned to achieve a rail-to-rail output, as visible in Figure 4.13.

The comparator output voltage is simulated with the reference voltage in the negative input, represented in the schematic by $v_n$, and varying the positive input voltage, $v_p$, from zero to $V_{DD}$. For values of $v_p$ lower than the reference voltage the output is low and, for values higher than the reference voltage the output is high, in this case reaching $V_{DD}$.

The second comparator implemented is represented in Figure 4.14. It consists of two inverters with feedback and with two diode-connected transistors in parallel with the NMOS transistors, forming a
latched comparator. These diode-connected transistors increase the power consumption since once active, they are always in the saturation zone.

This comparator uses a differential topology. The output is taken at the non-inverting comparator.
output, represented in the schematic by $v_{out}$, identical to the previous comparator. This comparator output is not rail-to-rail, unless a large current consumption is used to give a high output. To try to solve this problem and improve the comparator gain, increasing its output voltage, a preamplifier is used along with the latched comparator. The schematic of the comparator with a preamplifier is represented in Figure 4.15.

![Figure 4.15: Latched comparator schematic with preamplifier.](image)

With the simple latch comparator the output is about 240 mV while with the preamplifier the comparator output is increased to approximately 630 mV, with a characteristic similar to the one of the OpAmp comparator. This circuit has a higher output voltage excursion because it only has two stacked transistors. However, it spends more current than previous topologies.

One last comparator is implemented as represented in Figure 4.16. This latched comparator implementation eliminates the two diode-connected transistors and adds the double cross connections, which increases the gain of the comparator, increasing its output voltage swing.

Similarly to the other latched comparators presented, this comparator has differential outputs. The positive output it represented in Figure 4.17 for a variation of the input voltage between zero and $V_{DD}$, with the reference voltage connected to the negative input.

It's visible in Figure 4.17 that the comparator has hysteresis: for increasing values of the input voltage the transition happens for 700 mV, and for decreasing values of the input voltage the output switches at 500 mV. Both trip points, points at which the comparator output voltage switches, must correspond to the reference voltages. Using this comparator the oscillator power consumption would be higher than with previous versions. However, it would allow to implement a relaxation oscillator using only one
The last block of the oscillator to be implemented is the SR latch. It’s chosen a SR latch implemented...
with two NOR gates, as represented in Figure 4.18. The SR latch generates the oscillator output, a square wave of amplitude $V_{DD}$, according to the comparators output.

![SR latch schematic.](image)

![NOR schematic.](image)

**Figure 4.18:** Representation of the SR latch and respective NOR gate.

The transistors dimensions of each gate are minimized to lower its power consumption. The latch inputs are the outputs of both comparators and its output is the signal that controls if the capacitor charging and discharging switch. The SR latch operation is characterized in Table 4.4.

**Table 4.4:** SR latch truth table.

<table>
<thead>
<tr>
<th>S</th>
<th>R</th>
<th>Q</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low</td>
<td>Low</td>
<td>Q</td>
</tr>
<tr>
<td>Low</td>
<td>High</td>
<td>Low</td>
</tr>
<tr>
<td>High</td>
<td>Low</td>
<td>High</td>
</tr>
<tr>
<td>High</td>
<td>High</td>
<td>X</td>
</tr>
</tbody>
</table>

With both inputs at low the SR latch output holds the previous states while both input at high results in an undefined and unwanted state. This state does not have a established output value, the result would depend on which gate is faster to switch, and therefore, the SR latch output is unpredictable.

With the different topologies of comparators designed and simulated the OpAmp option is chosen to implement the comparators in the oscillator. All circuits are designed together and the oscillator is simulated. The schematic of the relaxation oscillator is the one presented in Figure 4.11.

The capacitor charges until its voltage reaches 700 mV, the maximum reference voltage, when the
output of the comparator responsible for setting the SR latch turns high. At this instant, \( Q \) is high and therefore the capacitor starts discharging. When the capacitor voltage reaches 500 mV it is the other comparator output that turns high, resetting the SR latch and starting to charge the capacitor again. The time the capacitor takes to charge from one reference voltage to the other defines the oscillator frequency. To achieve a frequency of 31.25 kHz a capacitor of 704.92 fF is used.

Finally, after verifying the correct operation of the relaxation oscillator its power consumption is simulated. Figure 4.20 shows the oscillator current over time.
The oscillator current is approximately 28 nA with peaks at the output transitions that reach 17.7 µA. Integrating the current over one period gives 87.94 nA which corresponds to a power consumption of 105.5 nW. Since only one comparator is operating at a time, to reduce the oscillator power consumption, an enable signal can be added to the comparators, turning them off when not operating. Finally, RCO phase noise is simulated. Using this simulation the oscillator RMS phase jitter is calculated as 3.77 µs, when used an integration band between 12 Hz and 20 kHz.

The charge-pump topology presented, although using the comparators with the lowest power simulated still consume more than what is pretended for this oscillator. Simulating the circuit for PVT variations, in some situations the oscillator fails to output a square wave. To improve the circuit performance a different implementation of an RCO [37] is designed. The simplified schematic is represented in Figure 4.21 and the complete circuit is presented in Appendix A.

![Figure 4.21: Improved RC oscillator schematic.](image)

Capacitor voltage, $v_{C1}$ ($v_{C2}$), is converted to current, $i_{M4}$ ($i_{M9}$), by transistor NM4 (NM9). This current is then compared with the bias current, $i_{R1}$ of resistor R1, in the current comparator, composed by NM4 (NM9) and NM5 (NM8). When $i_{M4}$ reaches $i_{R1}$ value, the latch, corresponding to transistors NM6, PM6, NM7 and PM8, toggle its outputs, $v_{clk}$ and $v_{clkB}$. When one side of the latch rises the other automatically drops. Transistors NM5 and NM8 also operate as clamp transistors, limiting $v_{clk}$ and $v_{clkB}$ amplitude. By keeping these signals at low levels the impact of comparator delay in the circuit frequency diminish. A level converter circuit - not represented in Figure 4.21 - receives $v_{clk}$ and $v_{clkB}$ signals and outputs the complementary square signals $Q$ and $Q_B$, which are applied to the gates of the transistors that act as switches (NM3 and NM10), controlling the charge and discharge of the capacitors. Signals $Q$ and $Q_B$ are the oscillator outputs.

Still based on an RC network to define the current and a comparator to assure transitions at the correct instant, this new implementation achieves much more power-efficient results. The major difference
face to the previous version is the use of a current comparator. Figure 4.22 illustrates relevant voltages and current of the RCO.

![Figure 4.22: Improved RC oscillator voltages and currents.](image)

In Figure 4.22, $i_{M4}$ exceeds $i_{R1}$ before the capacitor is discharged, pulling $i_{M4}$ to 0 A. This happens due to delays in the comparison, adding an error to the frequency and it is corrected during design by adjusting the capacitors value, pulling the nominal frequency to the desired value, 31.25 kHz.

In this RCO a self-biased current source, based on a threshold voltage reference current source, is implemented [65]. Instead of depending on the voltage supply, the PM1 current is directly dependent on the current source output current, PM2 drain current. By applying this technique to a threshold voltage reference the voltage supply sensitivity reduces significantly, however, due to the high negative temperature coefficient of the threshold voltage, this current source is temperature dependent.

Self-biased current sources have two stable points, being one the zero current point. A start-up circuit [66] is required to force the current source to operate in the stable non-zero point with $I_{R1}$ current amplitude. Figure 4.23 shows the self-biased threshold voltage reference current source and respective start-up circuit.

At the initial instant capacitor $C_3$ is discharged and the voltage at its terminals is zero, putting the transistor NM12 gate at $V_{DD}$. Transistor NM12 starts conducting forcing its drain current to transistor PM2, which mirrors it to PM1. At this point PM1 and PM2 are in saturation region and $v_{sd1}$ of PM1, which corresponds to NM11 gate voltage, starts rising, turning on NM11 which operates as a switch. When NM11 turns on forces a voltage close to zero to transistor NM12 gate. With no drain current, MN12 is disconnected from the circuit.

Regarding the current source, the voltage at NM2 terminal varies with the voltage supply variations, reducing the voltage variation at $R_1$ terminals, therefore decreasing the oscillation frequency depen-
dence on $V_{DD}$. A triple-well NMOS is used to eliminate the body effect.

To fulfill the ULP specification, the current source is dimensioned to have 6.5 nA in each branch, corresponding to a power consumption of 78.5 nW. Current decrease is limited by the size of the transistors. In order to reduce the current, keeping the transistors in the desired operating zones, is necessary to increase even further the length of the transistors.

From the RCO circuit, the bias current is given as a function of NM1 gate voltage and resistor $R_1$,

$$i_{R_1} = \frac{v_{gs1}}{R_1}$$  \hspace{1cm} (4.4)

and the capacitor voltage, $v_{C_1}$, can be expressed as

$$v_{C_1} = v_{gs4} = \frac{i_{R_1}}{2C \times f_{osc}}.$$  \hspace{1cm} (4.5)

From (4.4) and (4.5) it follows that the oscillation frequency depends on $R_1$ and $C_1$ as

$$f_{osc} = \frac{i_{R_1}}{2C \times v_{gs4}} = \frac{v_{gs1}}{2R_1C_1 \times v_{gs4}},$$  \hspace{1cm} (4.6)

showing that if NM1 and NM4 transistors have equal dimensions, gate voltages $v_{gs1}$ and $v_{gs4}$ cancel out and the oscillation frequency is only dependent on the resistor and the capacitor values. The process deviation consequences are diminished, since both transistors vary equally and it’s important to have low PVT variations to keep the oscillation frequency constant.

The RCO oscillation frequency deviation for temperature and voltage supply variations is represented in Figure 4.24. The oscillator is simulated for two situations: for temperatures between -40 °C and 80 °C.
at nominal supply voltage and with a 10% variation of $V_{DD}$ at nominal temperature, 27 °C.

For the temperature variation the RCO presents a 0.46%/°C deviation. Regarding the voltage supply, its variation translates to 1.85%/V. Synchronizing both oscillators reduce significantly the RCO frequency deviations.

To allow for the frequency to be adjusted according to the reference frequency, calibration is added to the RCO through a digital control word, turning it into a digitally-controlled RCO. Calibration is achieved by adjusting the capacitor bias current and is done in two phases: coarse and fine calibration. For the coarse calibration, current branches are added in parallel with the current mirror, charging the capacitor. After the coarse calibration, the fine calibration takes place. The bias resistor is divided into a fixed part plus a digitally controlled part, composing the bias current fine calibration.

The coarse calibration current branches have binary weights and, due to the size of the PMOS transistors, only four current branches are used. Simulating variations of the control word for the nominal corner the calibration performance is tested. After the coarse calibration, the frequency still has a variation of approximately ±2 kHz. The fine calibration should be dimensioned to cover the variation from coarse calibration, assuring the error after 12 h of operation do not exceed ±5 min. This represents a frequency deviation of ±230 Hz from the desired frequency, 31.25 kHz. To achieve this results seven resistors with binary weights are used, being the least significant resistor 3.85 kHz.

Using binary-weighted resistors, instead of all the resistors with the same value (adequate for thermometer coding), the circuit area and the number of control bits needed is reduced, however in situations with a large number of bits transitions the result is more unstable. In the total, 11 bits are used in the
RCO frequency calibration control.

The implemented RCO performance is summarized in Table 4.5 and it is compared with state of the art works presented in literature. With a nominal frequency of 31.24 kHz, 0.032 % deviation from the desired value, the RCO has a low power consumption, better than several works in the literature and the RMS phase jitter is matched to the values reported.

Table 4.5: Comparison of RC oscillator with state of the art works.

<table>
<thead>
<tr>
<th>Reference</th>
<th>This work</th>
<th>[33]</th>
<th>[34]</th>
<th>[35]</th>
<th>[36]</th>
<th>[37]</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMOS process [nm]</td>
<td>130</td>
<td>350</td>
<td>130</td>
<td>180</td>
<td>180</td>
<td>180</td>
</tr>
<tr>
<td>Frequency [kHz]</td>
<td>31.25</td>
<td>80</td>
<td>32</td>
<td>32.7</td>
<td>31.25</td>
<td>32.7</td>
</tr>
<tr>
<td>Supply voltage [V]</td>
<td>1.2</td>
<td>1.0</td>
<td>1.2</td>
<td>0.6</td>
<td>1.8</td>
<td>0.85</td>
</tr>
<tr>
<td>Power consumption [nW]</td>
<td>78.5</td>
<td>1140</td>
<td>80</td>
<td>51</td>
<td>360</td>
<td>54.2</td>
</tr>
<tr>
<td>Phase noise (@1 kHz) [dBc/Hz]</td>
<td>-57.78</td>
<td>NA</td>
<td>-61.33</td>
<td>NA</td>
<td>NA</td>
<td>NA</td>
</tr>
<tr>
<td>RMS phase jitter (120 Hz-20 kHz) [µs]</td>
<td>1.05</td>
<td>NA</td>
<td>1.03</td>
<td>NA</td>
<td>NA</td>
<td>0.17</td>
</tr>
</tbody>
</table>

4.3 Synchronization

Aside from the reference oscillator implemented as a crystal oscillator and a RC oscillator used as the digitally-controlled oscillator, the system is also composed by the calibration circuitry. The complete synchronization system used to calibrate the digitally-controlled oscillator with the crystal oscillator is illustrated in Figure 4.25, where \( x_1 \) and \( x_2 \) represent the crystal resonator connections.

![Figure 4.25: Detailed clock generator system block diagram.](image)

With the power off only the RCO is operating. XO and buffer are power down, and without variations in the input the period counter and calibration logic outputs are stable, reducing significantly the power consumption. RCO is operating normally, keeping track to time to power up the system at the correct moment. When the power is on, and after the start-up time of the XO has passed, the buffer gets a
sinusoidal wave from the oscillator and turns it into square wave with amplitude of $V_{DD}$. Since the buffer outputs a square wave before the XO has reached the steady-state the signal is not yet stable, therefore, the first cycles of this wave are ignored.

The calibration circuitry is divided into the period counter and the calibration logic. These blocks are responsible for detecting the error of the RCO frequency face to the required frequency and generate the RCO control word which adjusts its frequency. To perform this, the period counter counts the number of XO cycles during one period of the RCO. The period counter reset signal, responsible for bringing the counter output to its initial value, preparing it for the next iteration, is generated by the calibration logic block.

Knowing the XO inputs a 32 MHz signal to the counter, if the RCO is at the desired frequency of 31.25 kHz the counter reaches 1024 cycles during one period of the RCO. A 11-bit counter allows to go up to 2047 cycles per RCO period, corresponding to a frequency of two times the nominal. The counter detects frequency variations between 15.625 kHz (2048 XO cycles for each RCO cycle) and 32 MHz (one XO cycle for each RCO cycle) with a minimal step of 30.5 Hz at the nominal frequency, a range enough to cover the RCO corners frequency variation. Figure 4.26 shows the 11-bit period counter schematic.

The circuit implemented is an asynchronous up counter, each FF clock input is connected to the previous FF output - except for the first one which clock is the XO output signal - and the Least Significant Bit (LSB) corresponds to the first FF complementary output, $Q_B$, and the MSB to the last FF $Q_B$ signal. To obtain a better performance and lower power consumption dynamic and static flip-flops are used to design the counter. The dynamic FFs perform better at high frequencies than the static ones, however this topology fails at lower frequencies, therefore static logic is used for the last five FFs. The dynamic FFs implemented follow the TSPC topology, introduced by Yuan et al. [60], and uses low-leakage transistors to reduce the counter power consumption. The static FFs are based on 3-port NAND
circuits.

Figure 4.27 shows the output of the four FFs with higher weight for one complete cycle of the counter - one period of the most significant output bit. The input clock signal has a frequency of 32 MHz and each FF output frequency is half the previous one.

With the decrease of clock frequency each FF consumes less than the previous one. From the first dynamic FF, with a clock frequency of 32 MHz, through the last static FF, with a clock frequency of 31.25 kHz, the counter has a total power consumption of 997 nW for one counter cycle (from 0 to 2047). Being a digital circuit, the period counter has a low current during the majority of time, except for the output transitions where current peaks occur.

Situations where a high number of transitions is involve are not ideal for a binary counter. In a binary counter the bits change sequentially, therefore it goes through scenarios where the majority of bits has to change. With the FFs with different operating speeds, the transitions occur in different instants, generating unwanted intermediate states. A possible solution for this problem is to use a counter where in each cycle only one bit is changed, for instance, Gray counter or Johnson counter. The output bits of a counter from 0 to 7, implemented using each of these topologies are demonstrated in Table 4.6.

The dark background indicates the transitions face to the previous output. It is visible than the number of bits switching in the binary counter is much higher than in the other two solutions, however these two circuits have other disadvantages.

Gray code for a counter of 11 bits implies a high number of logic gates. A n-bit Gray counter logic is usually n-1 gates deep [67], with the number of gates in each level increasing face to the previous level. About $2^n+n$ logic gates are needed to implemented a n-bit Gray counter. With all this logic, in addition to a high circuit complexity, the power consumption increases significantly. An alternative could be to use
Table 4.6: Output bits of a binary counter, Gray counter and Johnson counter, from 0 to 7.

(a) Binary counter

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

(b) Gray counter

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

(c) Johnson counter

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

A binary counter and convert its output to Gray code. However, since the conversion is based in logic gates, and although its number is reduced, any change in the binary counter output is propagated to the Gray counter output, thus, intermediate undesired states are visible even after the conversion. This is not an adequate solution for this problem.

Another solution may be to use a ring counter such as a Johnson counter, a topology which also presents variations in only one bit in each cycle. Yet, this is a synchronous counter, originating a metastability problem. The need for more bits when compared with other counter topologies is also a disadvantage of the Johnson counter. While the binary counter need only n bits to originate \(2^n\) states in a Johnson counter of \(2^n\) only \(2n\) state are used. For this type of counter to reach 2048 cycles it would need 1024 bits, meaning 1024 FFs.

Although the alternatives presented are good solutions to binary counters in some situations, derived to the number of bits of the wanted counter, none of these is satisfactory for this counter. The chosen solution is to use the binary counter and delay the reading of its output in the calibration logic block. Figure 4.28 shows the simulation of the worst case where all counter bits change.

![Figure 4.28: Counter output example with transition of all bits.](image)

The counter output reading is synchronized with the negative edge of the clock signal (crystal oscillator signal) to allow the output to stabilized before the reading and guarantee the read value is the correct one. The counter must be dimensioned to guarantee the worst case takes less than half a clk period to change all the counter bits.

A digital calibration block is added to the previously designed analog circuits and the system perfor-
mance is analyzed with a mixed-signal simulation. The calibration logic block is responsible for generating the correct 11-bit calibration word which will bring the RCO frequency closest possible to 31.25 kHz. This calibration word is divided into two parts: one corresponding to the capacitor charging current branches; and one to the bias resistors.

As seen in section 2.2 both current branches and bias resistors are binary weighted, with 4 and 7 control bits, respectively, and being the current branches responsible for a higher variation in the current. The calibration logic block is implemented using hardware description language Verilog which is depicted in Appendix B.

Calibration word is changed after the RCO output signal positive edge. Since the calibration word is applied to the current branches and bias resistors the current is directly affected. The idea is to avoid current peaks (resulting from calibration word transition) to happen at the RCO output signal edges, which would translate to a bias current different from the expected at the beginning of the RCO cycle every time the calibration word is actualized. If these peaks coincide with the edge of the RCO output signal, its frequency would eventually settle at the correct value, however, it would take longer. By delaying the calibration word update the calibration is faster.

Even so the RCO frequency takes a few cycles to stabilize, thus after the calibration word update the reset signal waits for two RCO cycles before allowing the count to begin, given time for the frequency to settle. At the end of the next RCO cycle the counter final value is used to select the necessary correction. A flowchart indicating the calibration algorithm is presented in Figure 4.29.

If it is the first time the system is calibrated it starts with the default calibration word, obtained by simulating the RCO for the nominal corner. After that, the initial calibration word is the one selected in the previous calibration. This would allow for process variations to be corrected in the first calibration and in the next ones only errors related with temperature and supply voltage are corrected, reducing the calibration time.

When one RCO period has passed, the counter output is read and the error is calculated as the difference to 1024 cycles. To simplify the implementation of the comparison between the error value and the defined limits, the error absolute value is used. If the error is negative, that is, the number of cycles counted is greater than 1024, its absolute value is calculated as the complement for two of its value. Coarse calibration controls the 4 bits which correspond to the current branches charging the capacitors. If the error is less than 80 cycles it automatically goes to fine calibration, otherwise depending on the error the coarse correction is 1xLSB or 2xLSB. The coarse calibration at the nominal corner covers a frequency range of 3.91 kHz and 53.2 kHz.

In the fine calibration other two corrections are available, 1xLSB for errors less than or equal to 32 cycles and 8xLSB for higher errors. During simulations of the complete system the value of the bias resistor responsible for the fine calibration defined during the RCO design is adjusted to cover the
desired frequency range. After this correction, at the nominal corner, this part of the calibration covers frequency variations of approximately 3.45 kHz, with a minimum step of about 35 Hz. At other corners that not the nominal the minimum step corresponds to - in the worst case - an absolute error variation of 10 XO cycles for each RCO cycle, therefore, the distance to the nominal value (1024) is at most 5 XO cycles. This value, 5 cycles, is selected as the limit which ends the calibration. If a lower limit is defined, with some PVT corners, the system would be in the calibration loop indefinitely. Figure 4.30 illustrates a calibration cycle example.

Figure 4.30 shows the evolution of the system calibration, from the time the XO is turned on until the calibration is complete. A signal, clk_rdy, indicates the clock start-up time has passed and it is ready for use. When this signal goes high the calibration starts; every three RCO cycles the frequency error is analyzed and the calibration word, cal_word, is actualized. The calibration absolute error, represented by the variable abs_error, decreases with each iteration. When it reaches less than five cycles of error (two cycles in the illustrated case) the variable cal_off is activated, signalizing the end of the calibration.

The calibration is limited at lower frequencies by the counter which can only detect a minimum RCO frequency of 15.625 kHz and at higher frequencies by the oscillator digital control which can only correct frequencies below 53.2 kHz. If the system is powered on every 12 hours to synchronize the oscillators
an error of ± 5 cycles translates into an error of approximately ± 5 minutes before or after the power up correct instant. To improve the system performance this error must be reduced to a minimum.

Variations of temperature and voltage supply are simulated for the complete system and the RCO frequency is compared before and after calibration. For temperatures between -40 °C and 80 °C the oscillation frequency presents a deviation of 93.8 ppm/°C after calibration. Regarding the 10% variation of the voltage supply the frequency deviation after calibration is 1.75 %/V. Table 4.7 compares this work RCO frequency deviation (before and after calibration) with state of the art works. All the frequency variation values corresponding to the state of the art works, with temperature and supply voltage deviation, are obtained with calibration.
Table 4.7: Comparison of RC oscillator before and after calibration with state of the art works.

<table>
<thead>
<tr>
<th>Reference</th>
<th>This work</th>
<th>[33]</th>
<th>[34]</th>
<th>[35]</th>
<th>[36]</th>
<th>[37]</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMOS process [nm]</td>
<td>130</td>
<td>350</td>
<td>130</td>
<td>180</td>
<td>180</td>
<td>180</td>
</tr>
<tr>
<td>Frequency [kHz]</td>
<td>31.25</td>
<td>80</td>
<td>32</td>
<td>32.7</td>
<td>31.25</td>
<td>32.7</td>
</tr>
<tr>
<td>Supply voltage [V]</td>
<td>1.2</td>
<td>1.0</td>
<td>1.2</td>
<td>0.6</td>
<td>1.8</td>
<td>0.85</td>
</tr>
<tr>
<td>Frequency deviation with temperature [ppm/°C]</td>
<td>93.8(^a)</td>
<td>4600(^b)</td>
<td>842</td>
<td>10</td>
<td>43.1</td>
<td>4000</td>
</tr>
<tr>
<td>Frequency deviation with supply voltage [%/V]</td>
<td>1.75(^a)</td>
<td>1.85(^b)</td>
<td>-2.5</td>
<td>NA</td>
<td>-1.2</td>
<td>5</td>
</tr>
</tbody>
</table>

\(^a\) after calibration  
\(^b\) before calibration
5 Conclusions

Contents

5.1 Future work ......................................................... 78
Conclusions

There are many challenges to overtake in the design of a low-power clock generator system. A high frequency stability of the clock signal must be guaranteed, as well as a duty-cycle of 50%. In the case of the oscillator used to keep track of time, power consumption is of extreme importance and achieving the desired performance without recurring to additional current presents itself as a big challenge. To assure the power up occurs at the correct instant, synchronization between both oscillators is introduced in the system. Synchronization should end as soon as possible without forgetting that the system must be low-power, that is, some care has to be taken to keep the power consumption to a minimum without compromising the oscillators tuning.

A 32 MHz crystal oscillator is chosen as the reference oscillator for its high Q, which translates into a high frequency stability. Keeping in mind the low-power specification of the system, the crystal oscillator transistors are thus dimensioned for the lowest start-up energy solution. The single-MOS topology is selected over the inverter-based XO due to the possibility to control its bias current, allowing to perform power down of the circuit and to improve the results implementing a quick start-up technique.

The buffer, used to transform the crystal oscillator output into a square wave, is dimensioned for a 5 pF load capacitor. With such a high value of load capacitor the circuit is guaranteed to operate correctly when introduced in the system, however its simulated power consumption suffers with this decision. To compare the implemented XO and buffer with literature works it is again simulated with a 500 fF load capacitor, a more reasonable value for the application in mind. The results obtained are not as good as state of the art works. Adding a quick start-up technique to the crystal oscillator would allow a faster start-up and, consequently, a lower value energy consumption.

A current-comparator-based RCO is chosen as the preferred ULP oscillator topology for this application due to its lower power consumption and correct operation under PVT variations. The ULP wake-up oscillator frequency is chosen as 31.25 kHz, a power of two relation with the XO frequency, therefore a binary counter can be used in the synchronization mechanism. This oscillator presents a high frequency deviation in the corners simulation, however the oscillation frequency error is decreased with calibration.

In this topology, the oscillator frequency is defined both by the capacitor size and bias current (or bias resistor). To avoid large area consuming capacitors banks, it is chosen to adjust the bias current in order
to correct the frequency. The implemented calibration algorithm assumes calibration is complete with errors equal to or less than 5 cycles of \( \text{XO} \) for each \( \text{RCO} \) cycle. This limit is defined by the worst case minimum step for PVT variations. At some situations the minimum step corresponds to an absolute error variation of 10 \( \text{XO} \) cycles for each \( \text{RCO} \) cycle, therefore, in the worst case, the distance to the nominal value (1024) is 5 \( \text{XO} \) cycles. If a lower limit is defined, in these cases the calibration would never end. After calibration, and considering the 5 \( \text{XO} \) cycles of error, the \( \text{RCO} \) still presents a maximum error of \( \pm 5 \text{ min} \) from every 12 h of operation. It is important to reduce this error to a minimum, possibly by adding more bits to the calibration word, decreasing its minimum step.

Comparing with others presented in the literature, this work presents a \( \text{XO} \) with higher start-up time, however no technique is implemented to accelerate this phase of the oscillator. The ULP oscillator offers comparable values of phase jitter and power consumption, surpassing some of the literature works. The temperature and supply voltage deviation after calibration is comparable to some of the literature works, however, after the calibration, the frequency still presents an error of 5 \( \text{min} \) for every 12 h of operation.

5.1 Future work

With the objective of enhancing the system performance additional improvements can be implemented. Respecting the crystal oscillator, adding a quick start-up technique, such as the ones illustrated in chapter 2, would reduce significantly the start-up time and energy, extending the battery life.

The \( \text{RCO} \) frequency takes a few cycles to stabilize, therefore, in the first cycles after the calibration word update, there is an error between the measured frequency and the real frequency value at which the oscillator will stabilize. Reducing the problem, by making the \( \text{RCO} \) frequency to converge to the real value faster, the calibration process would occur faster and more smoothly. A faster calibration process is important because it consumes a lot of power and therefore the longer it is, the more energy is spent; and because it should not take longer than the time required for the chip to communicate data with the base station. Apart from this solution, adding more bits to the fine calibration would allow a better control and smaller steps, reducing the calibration final error.

Still in respect with the system calibration, a faster algorithm can be implemented. By calculating the function which relates error and correction an approximate estimative of the correction needed for each situation is obtained. Two situations are simulated, with a high and a low calibration word and the respective errors are obtained. From these values the calibration function is obtained. By simulation is verified that the calibration function is non-linear, therefore, the more points are used to obtain the function, the better the fitting to the real correction values. The number of iterations of this algorithm is the number of points retrieved to calculate this function plus one iteration to calculate the correction needed for that error - the calibration duration is the same independently of the error. In the algorithm
implemented in this work, the number of iterations needed vary depending on whether the error is higher or lower. With this new solution, increasing the number of points means trading circuit simplicity and calibration speed for a better approximation to the correct calibration word. A study is needed to verify the improvement this algorithm could bring face to the algorithm implemented in this work.

A chip layout ready for fabrication is one of the future objectives for this work. Therefore the layout of analog blocks - XO, RCO and period counter - and digital block - calibration logic - should be designed. In the end all layouts must be integrated forming the mixed signal system. The implemented circuits have several components with equal values which should be paired. This is one of the main challenges of this layout work. Another challenge is the size of some of the components. The circuits use big dimensions capacitors and resistors due to the low oscillating frequency and power consumption.
Bibliography


1.55×0.85mm² 3ppm 1.0µA 32.768kHz MEMS-based oscillator,” in 2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC), pp. 226–227, Feb 2014.


RC oscillator complete schematic
The controlled RC oscillator complete schematic is shown in Figure A.1. This schematic includes the start-up circuit and the output level converter, as well as the controlled current branches and resistors for the calibration.

(a) Controlled RCO (without representation of current source resistor).

(b) Current source resistor, represented in the form of a controllable bank of resistors.

Figure A.1: Representation of the complete circuit of controlled RC oscillator.
Calibration logic Verilog code
module cal_circuitry (   
   input clk,   
   input rco,   
   input pwr,   
   input [10:0] count,   
   output reg reset_count,   
   output wire [10:0] cal_word) ;

parameter count_nom = 1024;   
parameter fine_corr1 = 7'b000_0001, fine_corr2 = 7'b000_1000,   
coarse_corr1 = 4'b0001, coarse_corr2 = 4'b0010;   
parameter limit1 = 5, limit2 = 32, limit3 = 80, limit4 = 256;   
parameter fine_cal_initial = 7'b100_0000;   
parameter coarse_cal_initial = 4'b1000;   
parameter clk_comp_ref = 200;

reg [11:0] abs_error;   
wire [11:0] error;   
reg [6:0] fine_corr, fine_cal_add, fine_cal;   
reg [3:0] coarse_corr, coarse_cal_add, coarse_cal;   
wire [3:0] corr_sel;   
reg clk_comp_out;   
reg retimed_rco, db_retimed_rco, delay_db_rco, db2_retimed_rco,   
delay_db2_rco;   
wire equal1, less1, greater1, equal2, less2, greater2, equal3, less3,   
greater3, equal4, less4, greater4;   
reg clk_rdy;   
reg [10:0] clk_comp_in, error_count;   
wire cal_off;   
reg first_cal=1;

// Multiplexer to select if clock comparator input must be counter output   
or the comparator reference   
// When clk_rdy turns high the counter must be used for the calibration –   
comparator is disconnected from counter to avoid errors in clk_rdy signal
always @ (clk_rdy or count)
begin
    case (clk_rdy)
    0: clk_comp_in <= count;
    1: clk_comp_in <= clk_comp_ref;
    default: clk_comp_in <= 0;
endcase
end

// Signal clock ready verification – comparator to verify if counter has reached 200 cycles
// Clock signal is defined as stable after 200 cycles
always @ (clk_comp_in or clk_comp_ref)
begin
    if (clk_comp_in >= clk_comp_ref)
        clk_comp_out <= 1;
    else
        clk_comp_out <= 0;
end

// Definition of clk_rdy signal
always @ (posedge clk)
begin
    clk_rdy <= clk_comp_out && pwr; // the count must have reach 200 cycles and power must be on
end

// Retimed RCO – synchronizes RCO signal with the CLK signal
always @ (negedge clk or negedge pwr)
begin
    if (!pwr)
        retimed_rco <= 0;
    else
        retimed_rco <= rco;
end

// Signal with double period than retimed_rco
always @ (posedge retimed_rco or negedge pwr)
begin
    if (!pwr)
        db_retimed_rco <= 0;
    else
        db_retimed_rco <= ~db_retimed_rco; // FF toggle with input always at '1'
end

// Delayed RCO – delays db_retimed_rco signal by half CLK cycle
always @ (posedge clk or negedge pwr)
begin
    if (!pwr)
        delay_db_rco <= 0;
    else
        delay_db_rco <= db_retimed_rco;
end

// Signal with double period than db_retimed_rco
always @ (posedge db_retimed_rco or negedge pwr)
begin
  if (!pwr)
    db2_retimed_rco <= 0;
  else
    db2_retimed_rco <= ~db2_retimed_rco; // FF toggle with input always at '1'.
end

// Delayed RCO – delays db_retimed_rco signal by half CLK cycle
always @ (posedge clk or negedge pwr)
begin
  if (!pwr)
    delay_db2_rco <= 0;
  else
    delay_db2_rco <= db2_retimed_rco;
end

// Counter reset
always @ (pwr or db_retimed_rco or db2_retimed_rco or clk_rdy)
begin
  if (!pwr)
    reset_count <= 1;
  else
    reset_count <= (db2_retimed_rco || db_retimed_rco) && clk_rdy;
end

// Multiplexer with flip-flop to update the error at the end of count
always @ (posedge reset_count)
begin
  if (cal_off)
    error_count <= count_nom;
  else
    error_count <= count;
end

// Subtractor responsible for calculate error
assign error = count_nom – error_count;

// Absolute error value multiplexer
always @ (error)
begin
  case (error[11])
    0: abs_error <= error; // if error is a positive number // absolute value is the same
    1: abs_error <= ~error + 1; // otherwise the absolute // value is the 2's complement of the error
    default abs_error <= error;
  endcase
end
Comparators to define in which interval is the error
comparator comp1 (  
dataA (abs_error),  
dataB (limit1),  
equal (equal1),  
less (less1),  
greater (greater1));

comparator comp2 (  
dataA (abs_error),  
dataB (limit2),  
equal (equal2),  
less (less2),  
greater (greater2));

comparator comp3 (  
dataA (abs_error),  
dataB (limit3),  
equal (equal3),  
less (less3),  
greater (greater3));

comparator comp4 (  
dataA (abs_error),  
dataB (limit4),  
equal (equal4),  
less (less4),  
greater (greater4));

// For errors equal or less than 20 cycles the calibration is turned off
assign cal_off = equal1 || less1;

// Logic responsible for the correction multiplexer select signal
assign corr_sel[0] = greater1 && (equal2 || less2);
assign corr_sel[1] = greater2 && (equal3 || less3);
assign corr_sel[2] = greater3 && (equal4 || less4);
assign corr_sel[3] = greater4;

// Error greater than limit3 -> coarse calibration
// Error less or equal to limit3 -> fine calibration

// Multiplexer to select the correct correction depending on the frequency error
always @(corr_sel[1:0])
begin
  case (corr_sel[1:0])
    2'b01: fine_corr <= fine_corr1; // 5 < error <= 32,  
    // correction = 000.0001
    2'b10: fine_corr <= fine_corr2; // error > 32,  
    // correction = 000.0100
    default: fine_corr <= 0;
  endcase
end

// Multiplexer to select the correct correction depending on the frequency error
always @ (corr_sel[3:2])
begin
  case (corr_sel[3:2])
    2'b01: coarse_corr <= coarse_corr1; // 80 < error <= 256,
           // correction = 0001
    2'b10: coarse_corr <= coarse_corr2; // error > 256,
           // correction = 0010
    default: coarse_corr <= 0;
  endcase
end

// Adder/subtractor
// adder de 4 bits
always @ (posedge delay_db2_rco)
begin
  if (error[1]) // if error[1] == 1 it adds both inputs
    coarse_cal_add <= coarse_cal + coarse_corr;
  else // otherwise it substracts both inputs
    coarse_cal_add <= coarse_cal - coarse_corr;
end

// adder de 7 bits
always @ (posedge delay_db2_rco)
begin
  if (error[1]) // if error[1] == 1 it adds both inputs
    fine_cal_add <= fine_cal + fine_corr;
  else // otherwise it substracts both inputs
    fine_cal_add <= fine_cal - fine_corr;
end

// Calibration word is the concatenation of coarse calibration word and fine calibration word
assign cal_word_add = {coarse_cal_add, fine_cal_add};

// Update both parts of calibration word
always @ (negedge delay_db_rco or posedge pwr)
begin
  if (!first_cal && reset_count) // && reset_count
    coarse_cal <= coarse_cal_add;
    fine_cal <= fine_cal_add;
  end
  else if (!first_cal && !reset_count)
    coarse_cal <= coarse_cal;
    fine_cal <= fine_cal;
  end
end
```verilog
coarse_cal <= coarse_cal_initial;
fine_cal <= fine_cal_initial;
end
end

// Calibration word is the concatenation of coarse calibration
// word and fine calibration word
assign cal_word = {coarse_cal, fine_cal};

// Multiplexer to update the first_cal variable
always @ (posedge db2_retimed_rco)
begin
  if (clk_rdy)
    first_cal <= 0;
  else
    first_cal <= 1;
end
endmodule
```