A High-Speed Delay-Locked Loop for RF applications

By

Olasoji Makinwa

in partial fulfilment of the requirements for the degree of

Master of Science
in Electrical Engineering

at the Delft University of Technology,
to be defended publicly on Monday 30th of September 2024 at 3:00 PM.

Supervisor:	dr. S.M. Alavi
Thesis committee:	dr. M. Babaie
	dr. F. Sebastiano

.

An electronic version of this thesis is available at http://repository.tudelft.nl/.

Acknowledgements

First and foremost, I would like to thank Morteza Alavi for giving me this opportunity to do my master thesis with him. He has been of great support from the beginning to the end. He gave me a lot of freedom during this thesis which allowed me to explore many different topics. I've learned so much during this one year of my thesis, which I will be forever grateful for.

I would like to thank Anil Kumaran, my daily supervisor, for all the help during this thesis. Helping me set up EM-simulations, helping me debugging problems in cadence and especially doing the layout. His help has helped tremendously in me finishing my thesis.

I would also like to thank my parents for the motivational words, the support and assistants when times were difficult during the thesis. I would like to thank my friends with whom I've had wonderful chats, walks and dinners during the very stressful times of my thesis.

I would also like to thank my girlfriend who helped me also focus on other things besides the thesis. Especially the wonderful vacation together, which helped me recuperate for the final stretch of the thesis.

Lastly, I would like to thank John Gajadharsing for the support from Ampleon during this master thesis This opportunity I'm very grateful for.

Olasoji Makinwa

Abstract

This thesis details the design of a high-speed RF delay locked loop(DLL). A new proposed phase detector architecture, coined the mixing phase detector, allows for phase detection at high frequencies, resulting in multiphase generation at these higher frequencies with a DLL. The mixing phase detector modulates the high frequency signal to DC, allowing for elementary circuits to be used that do not need high frequency capabilities. The mixing phase detector has been designed, entirely laid out, and will be taped out shortly, with which simulations have been performed and have been compared with the state-of-the-art. The simulated results, show the possibility of generating eight phases accurately at 11.1 GHz, which would be higher than the state-of-the-art.

Contents

Abstract 3
1 Introduction 6
1.1 Thesis outline 7
2 Digital transmitters 8
2.1 Amplifiers 8
3 Clock generation 12
3.1 IQ efficiency 12
3.1.1 IQ Mapping 13
3.1.2 Multiphase 14
3.2 Clock generators 14
3.2.1 Polyphase filter 15
3.2.2 Injection-locked ring oscillator 15
3.2.3 Delayed lock loop 15
3.3 System design 16
4 The Delay-locked loop 17
4.1 Conventional DLL 18
4.2 Phase accuracy 19
4.3 Phase detector 20
4.3.1 Mixing phase detector 21
4.3.2 Phase shifted Mixer 22
5 The Delay line 24
5.1 Delay elements 24
5.1.1 Cross-coupled current starved inverter 26
5.1.2 Ratioed cross-coupled current starved inverter 27
5.1.3 CML delay element 28
5.1.4 Design of the CML delay element 28
5.1.5 Layout vs Schematic 29
5.1.6 Phase accuracy 30
5.2 Balun 31
5.2.1 Balun matching 32
5.2.2 CML to CMOS 33
5.2.3 8:1 multiplexer 33
6 Mixing Phase detector 35
6.1 Analysis 35
6.2 Mixer 37
6.3 Calibration and acquisition circuit 38

6.3.1 Comparator....39
6.3.2 Charge pump....40
6.4 Design Equations....40
6.5 Phase noise....41
7 DLL Results....42
7.1 DLL....42
7.2 Mismatch....44
7.3 Chip....45
7.4 Comparison....46
8 Future work and Conclusion....47
8.1 Conclusion....47
8.2 Future work....47
Bibliography....48

1 Introduction

With the ever-growing usage and need for cellular communications, the demand for higher bandwidth, more channel capacity, and overall smart devices has increased. These demands require transmitters (TXs) that satisfy the following: A high data modulation bandwidth to be able to send data at high rates; the capacity to support complex modulation schemes, to increase the channel capacity; the capability to work at higher operating frequency, such that more devices can be connected simultaneously; and lastly the power efficiency of the transmitter ought to be high, as the global power consumption of transmitters is only increasing. All these factors put together make the design of a TX a daunting task, especially now that the TXs are starting to make ground for the next generation of mobile communications, 5G and 6G.

With 5G there is also the need for next-generation transmitters which support the 3GP standards. Consequently, various paradigm shifts and methods have been proposed for transmitter architectures to approach this challenge. The digital-intensive transmitter (DTX) has garnered interest due to its digitally intensive nature, lending itself nicely to smaller technology nodes. These architectures allow for more circuit functionalities to be implemented per unit area. These DTXs can also use switch-mode power amplification structures, which boast higher efficiencies. Lastly, it could potentially lead to full integration of a transmitter in one package (DAC, Mixer, Power amplifier, and matching). However, this comes with other limitations compared to the analog-intensive TXs. The primary challenges are higher quantization noise and extensive sampling spectral replicas in the frequency spectrum due to the direct digital-to-RF conversion without utilizing an explicit filtering operation in the DTX. Moreover, operating at higher frequencies demands meeting even more stringent requirements.

In the analogue transmitter, the maximum operating frequency is determined by the transit frequency, f_T, which can lie far into the 100GHzs. Thus, an analog TX that operates at 10GHz is realizable. For digital TXs however, this is more challenging due to the limitation of the frequency at which the transistor can switch between fully on and off states. This feature is one of the critical limitations of DTXs in supporting next-generation mobile communication. Consequently, in this thesis, methods and techniques that enable a DTX to operate at higher frequencies, i.e., >6GHz, are pursued.

This thesis aims to innovate methods that facilitate a DTX to operate at the 7-to-8 GHz band, a newly allocated frequency band for upcoming 6G and Wi-Fi 7 standards. Moreover, the DTX will aim to produce more than 20 dBm of power and manage a 256-QAM OFDM signal with more than 100MHz modulation bandwidth.

One of the essential building blocks is the accurate clock generator that simultaneously generates up to eight clock phases, each with a phase shift of 45 degrees to its adjacent clocks. Consequently, this thesis proposes a high-speed digital locked loop operating at more than 8GHz band that facilitates the realization of the DTX for a newly allocated frequency band of upcoming 6G and Wi-Fi 7 standards.

1.1 Thesis outline

In this thesis the following topics are covered (in order):

- **Digital transmitters**: A brief overview of the digital transmitter (DTX) will be given to understand what they are and how the delay-locked loop will relate to the design of a future DTX.
- **Clock generation**: In this chapter, various methods of clock generation for DTXs are presented and compared with each other. An important aspect of the clock generation is how it relates to the efficiency of a DTX. It will be shown, that for efficient IQ generation, an eight-phase clock is necessary. Afterwards, it will be explained why the delay-locked loop is chosen for this task.
- **The Delay-locked loop (DLL)**: In this chapter, the working of the DLL will first be explained. It starts with how the conventional DLL works and what the limitations are of the conventional architecture at high frequencies. It will be shown that the DLL is limited by the phase detector, because the phase-detector's ability to do phase detection at these high frequencies becomes very difficult. To combat this issue, a new proposed phase-detector is described, and it will be explained why it should fare better at these higher frequencies.
- **Delay line**: In this chapter, the design of the delay line will be covered. This will cover how an eight-phase delay line is designed that works at 10GHz. Also, it will be detailed how the delay line is interfaced with the off-chip clock and how the eight phases are going to be read from the chip.
- **Mixing phase-detector**: In this chapter, the new architecture for the phase detector is covered. This phase-detector, coined Mixing phase-detector (MPD), will be able to accurately lock two phases at frequencies higher than 10 GHz. First a qualitative analysis is done, in which an understanding of the MPD will be detailed. Then, the circuit implementation is covered.
- **Results**: In this chapter, the DLL, together with the implemented MPD, will be simulated based on the extracted schematic from the designed chip. The results will be used as a baseline for the comparison with the state of the art. The maximum obtained frequency will be shown to be 11.1 GHz. With these tentative results, it will be shown that it surpasses the state of the art.

2 Digital transmitters[1]

In this section, the operation of digital intensive transmitters (DTXs) is briefly reviewed. This review will cover the operation of the Polar and Cartesian transmitters which form the basis for understanding the digital-intensive counterparts. Afterwards, techniques for Polar and Cartesian are reviewed, which have been used in the current state of the art DTXs.

2.1 Amplifiers

The DTXs to be discussed are realized as either Polar or Cartesian architectures depicted in Figure 2.1. In the polar DTX scheme, the digital baseband data is first converted into the amplitude (AM and phase (PM) components. The phase component modulates the digital controlled oscillator (DCO) and subsequently drives a bank of switches acting as a digital power amplifier (DPA). This bank of digital switches will then modulate the amplitude of the signal, which together yields the AM and PM modulation. The synchronicity between the AM and PM paths is crucial, as this directly affects the signal integrity.

In the cartesian architecture, also called quadrature configuration, the orthogonal *I* and *Q* vectors modulate the amplitude and phase of the resultant RF signal using rectangular coordinates instead of polar coordinates. The cartesian architecture comprises of two orthogonal vectors, phase shifted by 90 degrees. When summed together, the amplitude and phase are determined by the amplitude of the in-phase, I, and quadrature phase, Q, such that the AM and PM components are determined by the magnitude of $I + jQ$ and the angle, $arctan(Q/I)$.

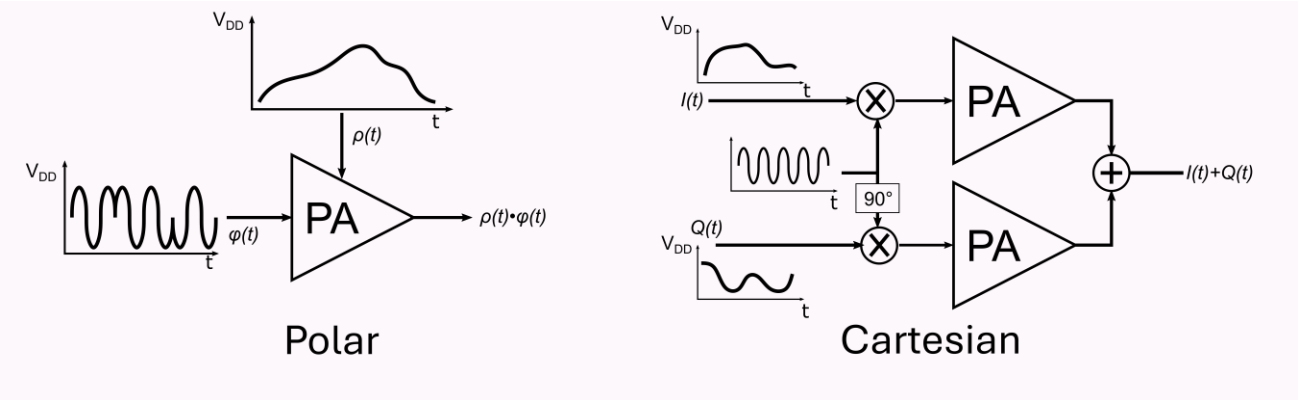


Figure 2.1: Canonical Polar and Cartesian structures.

In the polar architecture, the AM signal modulates the PA supply while the PM signal drives the PA's gate. The primary benefit of using polar is that only the phase needs to be modulated for a given amplitude, compared to the cartesian TX, where both PA's must vary their amplitude to achieve the designated angle.

One variant of the analog polar amplifier uses the Envelope Elimination and Restoration (EER) technique. The input signal is separated into amplitude and phase components. To get the phase information, a limiting circuit is used which clips the signal to only retain the phase information. To get the amplitude, a top detector circuit is used which tracks the changes in amplitude of the signal and removes the phase modulated information. In Figure 2.1, it can be seen how the phase signal has no amplitude modulation, and the amplitude signal has no phase information. In [1], such a polar amplifier is highlighted which utilizes a very efficient class-D power amplifier.

[1] Part of this Chapter is written as a part of ET4399, a Research Project at the University

Nowadays, instead of EER techniques, the generation of phase and amplitude is done digitally with a CORDIC (coordinate rotation digital computer). The CORDIC transforms the baseband signal to the phase and amplitude information. This transformation is given by equations (2.1) and (2.2) respectively.

$$\rho = \sqrt{I^2 + Q^2} \tag{2.1}$$

$$\phi = \text{atan2}\left(\frac{Q}{I}\right) \tag{2.2}$$

It can be seen from the equations above that the relationship between input and output is very nonlinear due to the *atan2* function, which leads to bandwidth expansion of the modulated signal [2]. This phenomenon is undesired, and in polar systems, the effective bandwidth of the modulated signal becomes larger, and more sophisticated hardware is required. It is also important that the AM and PM signals are synchronized. Otherwise, the misalignment will lead to significant spectral regrowth. Despite these challenges, digital-intensive polar architectures when used at full power achieve 100% efficiency irrespective of the accompanying I and Q, thus producing a constant efficiency curve. The constant efficiency curve will be explained in the next section about I/Q generating techniques. Digital Polar architectures have been devised as shown in [3], [4] and[5] .

In [6], a Cartesian DTX is designed and the DPA will be used as an example. The structure consists of an array of switches controlled by an amplitude control word (ACW) where the bank's current is proportional to the number of ON switches. In Figure 2.3, the switch banks are shown, where depending on the bit resolution, a certain number of switches are placed in parallel. When one of the codes from the ACW is set to HIGH the input clock will be passed through the AND gate and the NMOS will be turned on and off at the clock frequency. This will then modulate the current to the clock's frequency.

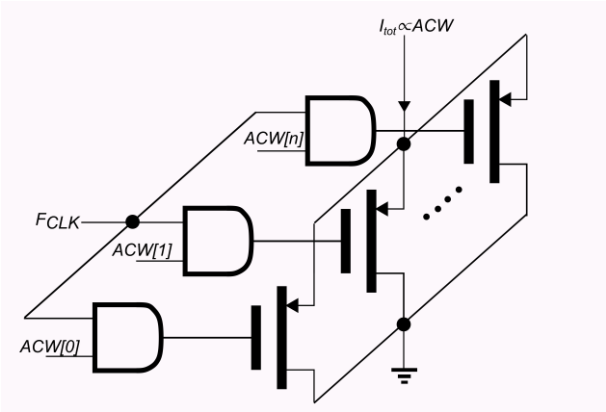


Figure 2.3: Current mode DTX.

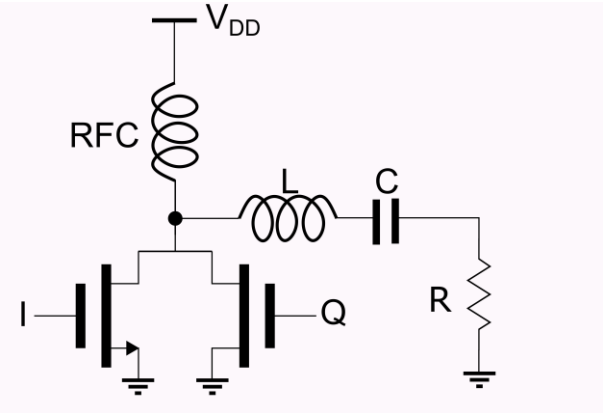


Figure 2.2: Class E amplifier.

The bank of switches is then connected to the amplifier which can be any class of amplifier but in this example a class-E amplifier is used. The advantage of a class-E amplifier is its high peak efficiency. In Figure 2.2, a simplified schematic of the class E amplifier is shown. The switch bank is replaced by two MOSFET's denoted by the I and Q path. The switch-bank in a class-E amplifier modulates the on-resistance of the switch-bank. When all the switches are turned on, the total on-resistance will be small, and all the power will be dissipated by the load (assuming infinite *Q* for the passives). If a fewer number of switches are turned on the on-resistance will be increased, and more power is dissipated in the which leads to a smaller amplitude at the load. Which allows for AM modulation in a class-E amplifier. However, this modulation scheme is quite non-linear because

the output voltage is determined by the voltage division between the on-resistance and the load. Which has the following relationship.

$$V_{out} = V_{DD} \frac{R_L}{r_{on} + R_L} \quad (2.3)$$

The non-linearity can be addressed by either using DPD (digital predistortion) to correct for the non-linearity. Or, as presented in the work of [6], by sizing the switch-bank non-linearly, the non-linearity of the transfer function can be counteracted.

Another structure that has been proposed is the switched capacitor power amplifier (SCPA) , as shown in [7] and [8]. The operational principle of these amplifiers can be seen in Figure 2.4. The amplifier comprises of N capacitors which are either switched to the supply (VDD) or ground. The output voltage will be linearly related to the number of capacitors connected to the supply over the capacitors. The ideal relationship is given by

$$V_{out} = V_{DD} \frac{n}{N} \quad (2.4)$$

where n is the total amount of active capacitors and N is the total amount of switches. Such structures have a linear dependence on the input code.

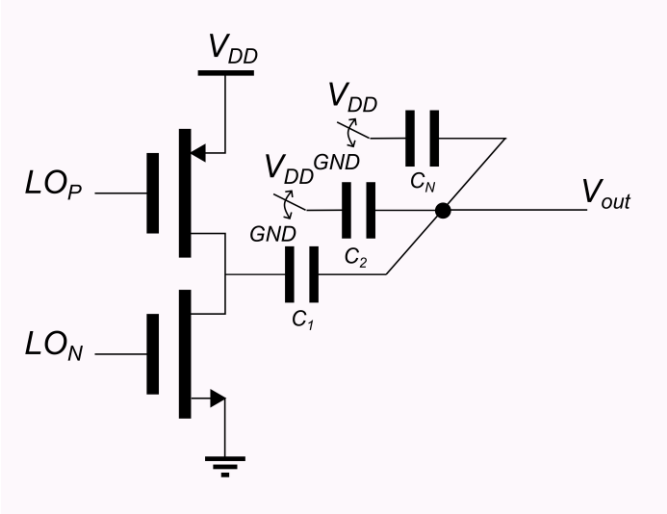


Figure 2.4: Switched capacitor power amplifier.

To qualify the performance of the various DTXs some key metrics are detailed to quantify the performance of these structures. These are:

- AM-AM: How linear is the relationship between the input and output amplitude
- AM-PM: How much does the input amplitude perturb the output phase
- Efficiency: How much power is burned by the active and passive components.

AM to AM distortion occurs due to there being a nonlinear variation in the number of switches on and switches off. As explained in [7], depending on the ACW a certain number of switches will be on, and off which have a finite on resistance and off resistance. This means that the equivalent resistance created by the on-resistance is also dependent on the off resistance. Ideally, the off resistance is infinite but in practice this is finite. Also, the circuit parasitic's contribute to this non-ideality.

AM-PM distortion can occur due to deviations in the time constant between NMOS and PMOS. Consider the SCPA shown in Figure 2.4. If the switches have an unequal on-resistance, when turning the capacitor from ground to the supply the duration will be different leading to phase deviation dependent on the amount of switches are turned on. This same phenomenon can happen with the current mode structure; however, it is less prominent because all the switches are NMOS.

Lastly, the efficiency is strongly determined by the preceding circuitry of the switches and the power dissipated in the switches. To drive these switches, buffers are needed to be able to switch the gates of the DPA on and off. The power lost due to driving the switches is given by (2.5)

$$P_{SW} = fCV^2 \qquad (2.5)$$

where f is the switching frequency and C is the gate capacitance. Thus, by using a structure which reduces the amount of switches the losses incurred due to driving the gates are diminished. It is for this reason that the structure shown in amplifier. Comparing the voltage mode DPA with the current mode, the switching losses are twice as much since it uses two switches per bit.

Many tradeoffs are needed to be made for the design of the digital transmitter. The focus however of this thesis will be regarding the phase generation for the digital transmitter. In the next section, various clock generators are analyzed.

3 Clock generation[2]

The Cartesian architecture is quite favorable over polar architectures due to its simplicity. However, as stated in the previous chapter, the IQ-efficiency of the Cartesian architecture is lower than that of the polar. To mitigate the problems of this low efficiency, techniques have been employed to increase the efficiency of Cartesian architectures.

In this section, the different techniques for IQ generation in DTXs is explained. Then afterwards, the various circuit implementations for clock generators are shown.

3.1 IQ efficiency

To understand the IQ efficiency, consider the 4 QAM situation as shown in Figure 3.1. To create this constellation the sequence of inputs needs to be *{I+, Q+}, {I-, Q+}, {I+, Q-}* and *{I-, Q-}*. These inputs are generated in a cartesian system by adding two quadrature currents together. The magnitude of the two currents results in the output RF current shown in equation (3.1).

$$I_{RF} = \sqrt{I_I^2 + I_Q^2} \qquad (3.1)$$

I_I and I_Q are the two quadrature currents which are related to the drain currents of the power amplifiers. These currents respectively drive the I and Q paths. In the 4 QAM situation, to generate these four constellation points the quadrature currents need to be maximum; this means that the ACW turns on all the switches in the switch bank[3]. The power that is delivered to all four corners is given by equation (3.2).

$$P_{dis} = \left(\sqrt{I_I^2 + I_Q^2}\right)^2 R_L = 2I_{I=Q}^2 R_L \qquad (3.2)$$

From the equation we see that the power delivered to the load is twice the maximum available current from each switch bank. Ideally, all the drain current from the switch-bank would go to the load, resulting in power which is proportional to four times the available maximum current. In the cartesian structure, however, this is not possible due to the inherent nature of the additive properties of the quadrature clocks. This results in an efficiency penalty when compared to the polar amplifier.

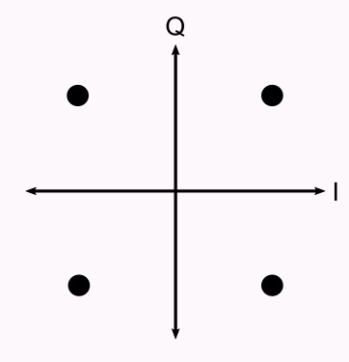


Figure 3.1: 4 QAM Modulation

In the polar amplifier, to get the same constellation points, only one switch-bank is needed. This switch-bank is modulated by the DCO with a 45°,135°, 225° or 315° clock. All the current that is allocated to the power amplifier is available for the load to be dissipated. This results in a 3dB power penalty for using a cartesian architecture over a polar amplifier when transmitting power in the diagonal corners. The I/Q efficiency is given by equation (2.7), where I_U *is* the unit current source of the switch bank.

$$\eta = \frac{I_{U,I}^2 + I_{U,Q}^2}{I_U^2} \qquad (3.3)$$

[2] Part of this Chapter is written as a part of ET4399, a Research Project at the University

[3] In this case a balanced structure is implicitly assumed, where one switch bank will source current, and the other bank can sink current. This allows for the positive and negative quadrature currents to be generated.

Plotting this function shows concentric circles with constant efficiencies for all possible I and Q combinations for polar amplifiers. For the cartesian system, a clover pattern with varying efficiencies dependent on the I and Q combination. In Figure 3.2, these patterns are shown and highlighted for all IQ combinations.

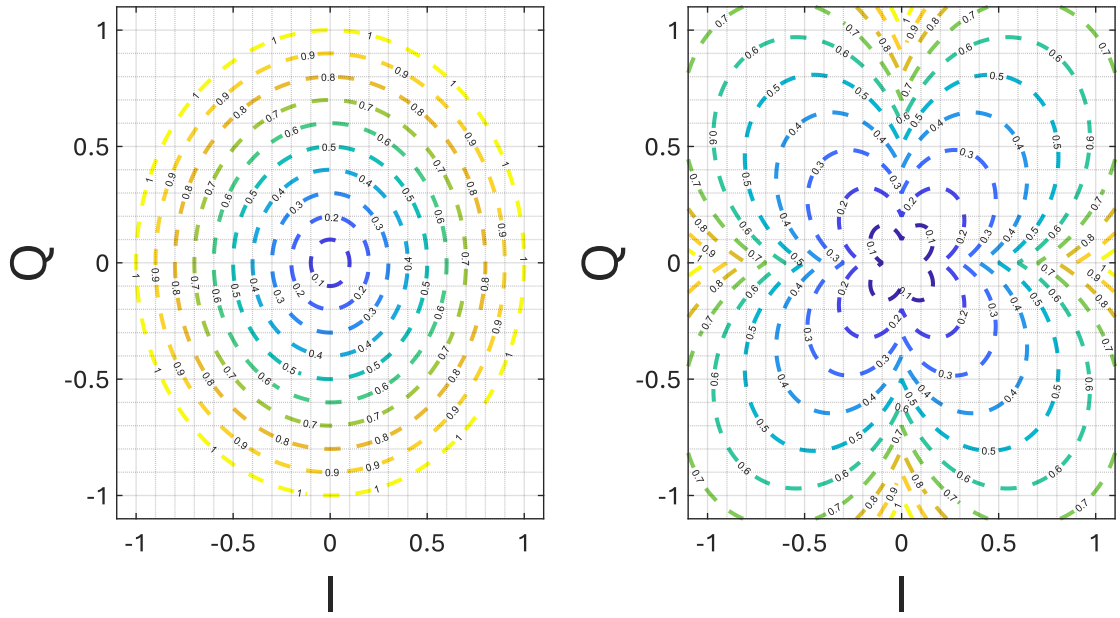


Figure 3.2: IQ efficiency of polar transmitters compared with cartesian transmitters.

To address this limitation of cartesian transmitters, two methods are shown that aim to optimize the I/Q efficiency.

3.1.1 IQ Mapping

In [7], they showed that mapping the IQ plane onto a diamond-shaped constellation rather than a square yields the doubling of the I/Q efficiency in the cartesian DTX. This can be easily understood with the help of Figure 3.3. The I/Q plane has been rotated such that the constellation points only lie on the I or Q axis. Now, when transmitting these cartesian coordinates, only one switch-bank needs to be switched on, which increases the IQ efficiency by 3dB. This technique for the case of 4 QAM would lead to equal performance as the polar amplifier. However, for constellations with more points such as 8 QAM or higher, the constellation will have

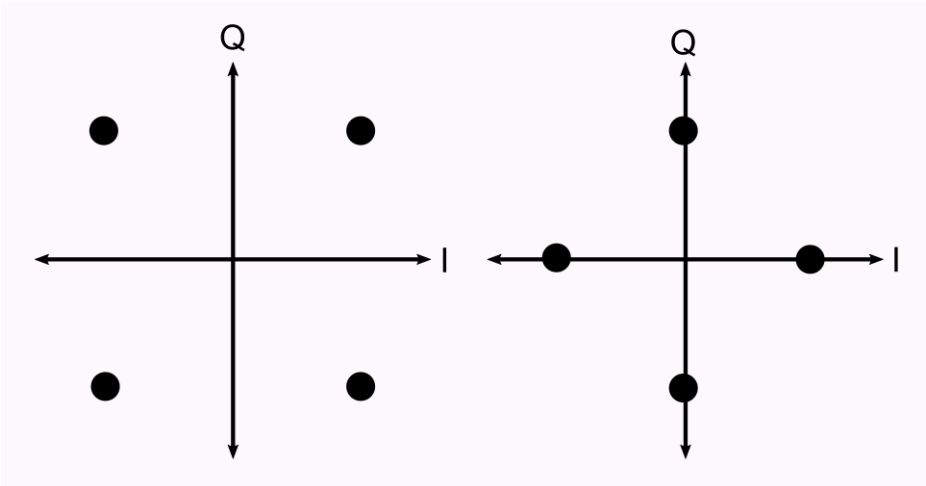


Figure 3.3: Rotating the cartesian plane by 45 degrees results in a diamond shaped constellation.

points along the diagonal axes. For all the points along these diagonals two power amplifiers are needed and would still fare worse to the polar architecture.

3.1.2 Multiphase

To further reduce the effects of lower efficiencies at any constellation point, a multiphase clocks can be used. The multiphase architecture uses more clocks to be used as the basis to construct IQ constellations. By increasing the number of clocks, for certain IQ constellation points a single power amplifier can be used instead of the summation of two. This will improve IQ efficiency at the IQ constellation points with a dedicated clock. Theoretically, by adding an infinite number of dedicated clocks, polar operation could be achieved from a cartesian architecture. However, the circuitry required to generate the dedicated clocks becomes difficult. In [8], eight phases were utilized to create this multiphase system. The restrictions on the number of phases were directly a consequence of capability of the clock generating circuit.

Multiphase cartesian DTXs have bridged the gap between cartesian architectures to polar amplifiers. The benefits of the simplicity of the cartesian architecture and the efficiency of the multiphase prove to be a good combination to create good DTXs. For this reason, an eight-phase clock generation circuit is going to be designed to create a multiphase DTX. In the next section, various clock generation circuits are discussed.

3.2 Clock generators

The first architecture is the CMOS divide-by-2 circuit. This circuit is relatively simple because it only requires master-slave latches to create the quadrature clocks. Also, due to the clocked-based nature of the latches, the phase relation between the clock and the output is well defined. Perturbations on the signal stem from the clock jitter, the circuits' mismatch, and intrinsic random and deterministic jitter.

To create the quadrature clocks, a differential input clock is needed. The latches sample the previous input and hold this for one period and are shifted by half a period. This creates the quadrature clocks. In Figure 3.4, the circuit and the waveform are shown.

The limitation of this circuit is that the input frequency is halved to obtain the quadrature clock. For low frequencies, this is not a problem but as the frequency increases, the maximum frequency is quickly reached. To get quadrature outputs at 8 GHz, a 16 GHz input clock is required, which the digital circuit cannot support in 40nm technology. To then create eight phases, another divide-by-2 circuit is needed, requiring a 32 GHz clock.

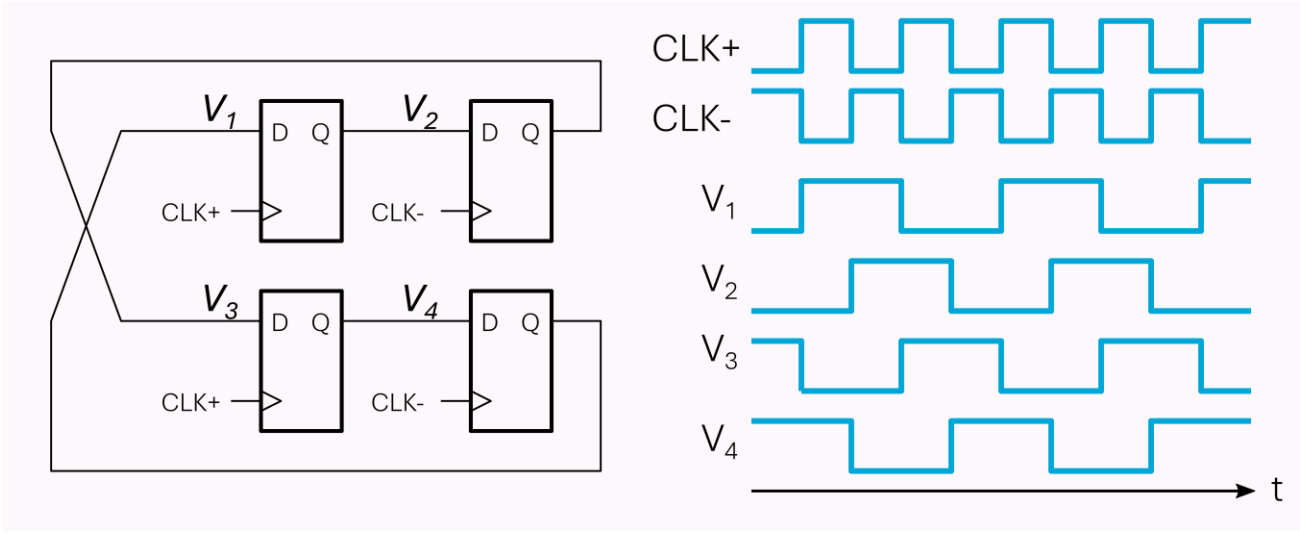


Figure 3.4: CMOS divide-by-2 circuit for quadrature clock generation.

To mitigate frequency division other solutions can be considered. These are.

- The Polyphase filter.
- Injection-locked ring oscillator.
- Delay locked loops.

3.2.1 Polyphase filter

Polyphase filters can construct quadrature clocks using complementary (differential) input clocks. These signals are supplied to an RC filter which creates a 45-degree phase shift at the RC filter's corner frequency. Depending on the input port it will create a positive 45-degrees advance/delay by combining two clock vectors with a shift of 45 degree plus that of 135. Vectorially, they will add up to a 90-degree rotation. In Figure 3.5, the circuit is shown.

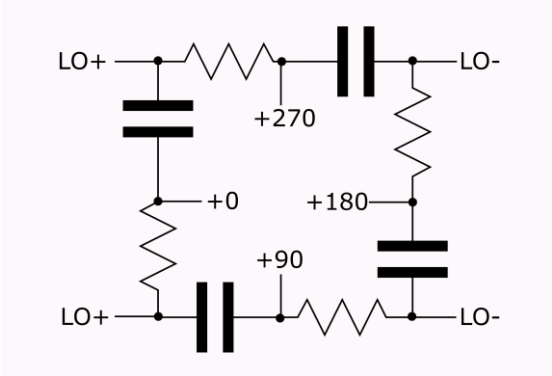


Figure 3.5: Polyphase filter.

The positive aspect of this circuit is that the frequency of operation is not reduced because the circuit is completely passive. Also, due to the circuit being passive, it does not require active elements, indicating the power dissipation of this circuit is low. In [9], they used such a circuit to create the phases. The downside of the polyphase filter circuit is that it is a tuned circuit, which operates best at this tuned frequency. Creating more than four phases is also difficult for the polyphase filter.

3.2.2 Injection-locked ring oscillator

In [10] they show an injection-locked ring oscillator (ILRO) to create multiphase clock signals. This solution uses a polyphase filter to create the four phases and feeds the ring oscillator, creating 8 phases. It was shown that the phase error of the polyphase filter is reduced due to the ring oscillator. However, the polyphase filter is narrowband and only works for a small operation frequency range. The results shown in the paper were good, so it shows that this is a viable method.

3.2.3 Delayed lock loop

The delay locked loop (DLL) can be used to create multiple phases. It is a that aligns a skewed clock with the reference clock. To do so, a phase detector compares the reference and the skewed clock. If one leads the other the voltage is adjusted, and the voltage-controlled delay (VCD) is tuned until they match.

The advantage of DLLS is that they operate at the same frequency as their input reference clock, like the ILRO. Still, unlike the ILRO, it does not need a polyphase filter to generate the quadrature phases, as the DLL creates the necessary phases. Also, the phase noise is better in a DLL since it does not accumulate. The disadvantage is that while the DLL may be locked, the individual phases might be unequal, creating offsets between the phases which poses a challenging design.

3.3 System design

The delay locked loop is chosen to be used for the clock generation of a DTX because it allows for a wide band operation for the DTX at high frequencies. The CMOS divide-by-2 circuit provides a wide band usage but is limited by frequency demands. It loses high frequency capabilities because the frequency is divided twice to create eight phases. The ring oscillator does provide high operating frequency, but its tunability is limited because the injected phases need to be derived from a polyphase filter. The polyphaser filter has a narrow band because it needs to operate at the center frequency determined by the RC pole. The DLL is the only structure which is wideband and truly derives all the phases inherently from a single clock.

The DLL will create an eight-phase clock output, which will be given to a clock distribution circuit which will be directed to the *I (in phase)* and *Q (quadrature phase)* switching units. These switching units consist of a mixing switch which upconverts the baseband signal and baseband switches, which encode the baseband data. In Figure 3.6, the DLL creates phases from 0 degrees to 315 degrees in 45-degree increments which connects to the clock distribution network. From this two of the selected clocks will be directed to the I and Q switching units. The top switch is the mixing switch, and the bottom switches encode the baseband.

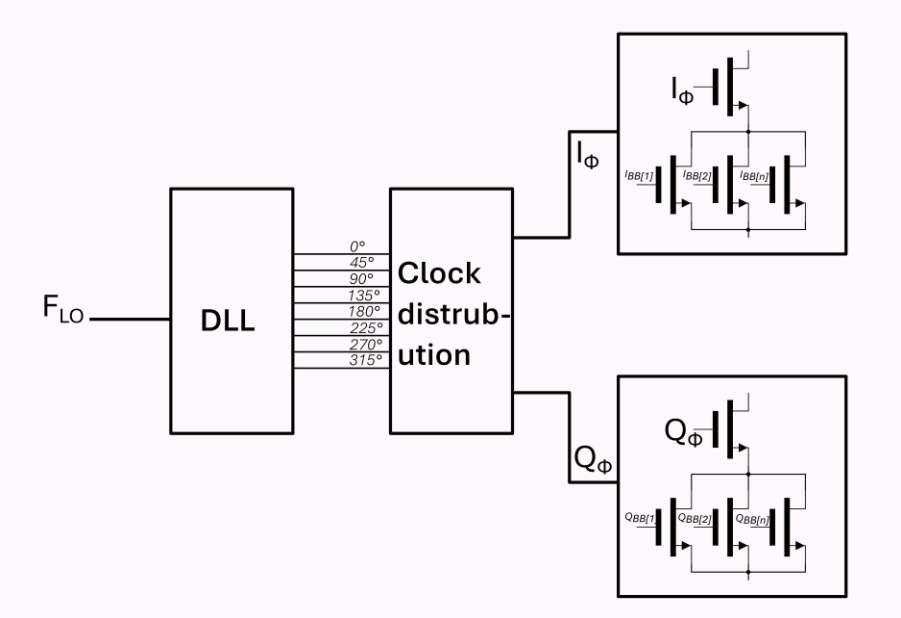


Figure 3.6: DLL in a DTX

The switches shown will be part of the power amplifier as explained in the previous chapter. The configuration of the power amplifier will not be explained in this thesis as this is future work.

The DLL contributes to three aspects of the performance of the DTX:

- Phase: the accuracy of the output clocks will determine the accuracy of the IQ constellation
- Amplitude: The duty cycle of the clocks will influence the amplitude of the output which influences the IQ constellation
- System efficiency: The power consumption of the DLL reduces the system efficiency.

In the next section the working of the conventional DLL is first explained, then a new approach will be covered for creating DLLs at high frequencies.

4 The Delay-locked loop

The Delay-Locked loop (DLL) is a feedback system that locks the system's output to its input by controlling the delay of the output signal. The output signal is derived from the input signal and is the delayed version of the input signal. The delay between the input and output signal can be due to unwanted patristics slowing down the signal or purposely due to delay elements such as inverters. In any case, irrespective of the nature of the delay the DLL will aim to force the input and output to be equal.

Using the DLL's locking property, multiple phases can be generated from a single reference input by introducing controlled delay elements. These elements will be tuned such that the output phase matches the input. When this condition is reached, a number of equally spaced phases will be generated of depending on the number of delay elements.

A detection mechanism is needed to lock the loop and detect the phase difference between the input and output. It is crucial that the phase detection is done accurately to ensure equally spaced phases. This requirement, however, becomes a challenge at higher frequencies since it requires sophisticated circuitry to provide this locking condition. From literature many approaches revolve around the commonly used phase detector used for Phase locked loops (PLL), this structure however struggles to work from around 9 GHz onwards.

A new topology for the phase detection is proposed to work at higher frequencies. This structure utilizes a mixer to detect the phase difference and lock the loop. With this architecture, phase generation up to 10 GHz, using a 40nm bulk CMOS process, is aimed to be achieved.

This chapter first describes the operation of the DLL. Then, the conventional approach to phase detection is explained, and finally, the operating principle of the proposed mixing phase detector is detailed.

4.1 Conventional DLL

The conventional DLL used for phase generation consists of a tunable delay line consisting of *N* voltage-controlled delay elements, a phase detector (PD) and a charge-pump (CP). In Figure 4.1, the conventional DLL structure is shown in which, as an example, three delay elements are used. Each delay element is tuned by a signal generated from the PD together with the CP. The PD compares the delayed signal and the clock and produces an output signal proportional to the difference. The CP then charges or discharges the capacitor until the phase becomes zero. This voltage is the voltage that locks the loop.

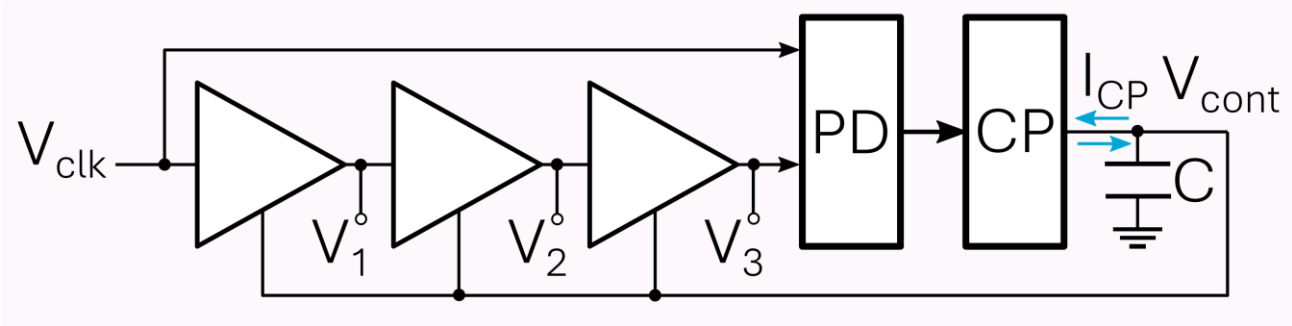


Figure 4.1: The conventional DLL consists of several delay elements, a phase detector and a charge-pump.

Initially the voltages V_1, V_2 and V_3 are unlocked. They are spaced equally but not locked to the reference clock. In Figure 4.2 it can be seen that all the delays are equal to t_0, which is the minimum delay. Then, when the feedback loop takes control, all the delays will be increased together such that V_3 is equal to the reference. At this point, the three voltages will be equally spaced. This principle can be applied to an arbitrary amount of delay elements to create multiple phases.

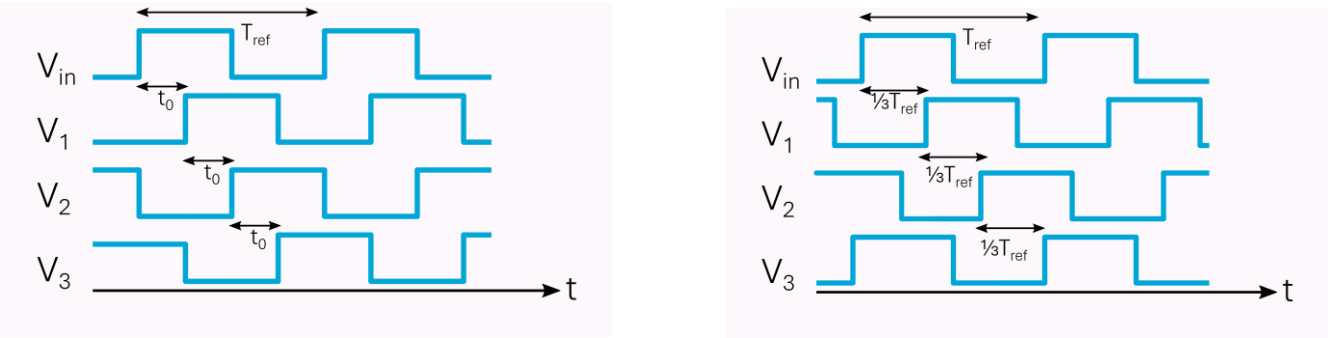


Figure 4.2: From locked to unlocked, when in the unlocked state each delay is given by the minimum. When locked each phase will be equally spaced.

The speed at which the DLL locks depends on the current strength of the charge pump, the size of the capacitor and the size of the capacitor. Since the feedback loop has a single pole (due to the integrator), it is a first order system, commonly referred to as a Type I DLL.

A feature of the DLL is that it does not accumulate jitter [10], as compared to the PLL. This is due to the inherent nature of how the DLL works. At each cycle, the reference clock is passed through the delay line and traverses the delay line. After the signal has traversed each delay element it does not recirculate back to the input. In a PLL, however, the signal will traverse through the delay elements and will be fed back to the input. This gives the DLL, compared with the PLL, an inherent better phase noise performance.

The transfer function between the input clock and the delayed clock, ϕ_{DL}, is an all-pass function. Intuitively, this can be understood by the fact that the delayed signal first must arrive at the PD, then compared

with the reference and then the PD produces an output which will change the delay of the next clock signal. So, the phase of the output clock is only determined by the input clock. Mathematically this operation can be found as followed; the phase detector together with the charge pump produce an output signal that is given by

$$V_{cont} = (\phi_{DL} - \phi_{CLK})\frac{I_{CP}}{sC} \quad (4.1)$$

The sensitivity of the delay elements, K_{DL}, given in rad/V, determines how fast the circuit delays as function of voltage. This yields the following relationship:

$$\phi_{DL} = V_{cont}K_{DL} + \phi_{CLK} \quad (4.2)$$

Combining equation (4.1) and (4.2) yields

$$(\phi_{DL} - \phi_{CLK})\frac{I_{CP}}{sC} = (\phi_{DL} - \phi_{CLK})\frac{1}{K_{DL}} \quad (4.3)$$

This equation shows that ϕ_{CLK} has to equal ϕ_{DL}. Thus, the relation between the delayed clocked and input clock is constant and independent of the loop (an all-pass filter). Therefore, the phase of the signal is only perturbed by the input clock this means that also the phase noise is only generated from the reference. However, when multiple delay elements are used, the phase at the end of the line is different from the input due to the delays of the delay elements. This means that even though the loop cannot change the instantaneous phase, the internal noise of each delay element will contribute to the output phase noise.

4.2 Phase accuracy

The accuracy of the DLL is an important measure of its performance. The accuracy depends on the locking accuracy of the phase detector and the mismatch between the delay elements. Each delay element is driven by the same control signal, as shown in Figure 4.1, but the response of each delay element can vary, and this cause phase error. To quantify this error, a short analysis is performed of the DLL.

The delay line of the DLL can be analysed as a Brownian bridge, where each random step taken is one delay of a delay element. If the DLL were to perfectly lock, the end and start point will be fixed. This means that for any sequence of random steps, the sum of delays will remain the same. Each delay is denoted with T_1, T_2 *up to* T_8 , which are random variables with expected value t_0. It follows that when the DLL is locked, the following equation is held.

$$E[\,T_1 + T_2 + \cdots + T_8] = t_p \quad (4.4)$$

Where t_p *is* the period of the reference frequency used for the DLL. For each step taken away from the beginning of the delay line, the uncertainty grows, and the variance increases. At the same time the closer to the endpoint the uncertainty decreases. It is to be expected that at the middle of the delay line the uncertainty is the greatest. Mathematically, when each delay element has a normally distributed variance and equal mean, the variance for each delay of the delay line is given by

$$Var(T_n) = \frac{n(N-n)}{n}\sigma_t = \frac{n(8-n)}{n}\sigma_t \quad (4.5)$$

where *n,* is the *n'*th delay element, N the number of delays and σ_t the variance of a single delay element. From this equation it can be shown that when 8 phases are used, the standard deviation of the worst delay is $2\sqrt{\sigma_t}$. This value is essential for deciding how to determine the scaling for each delay element. Because the sizing of the transistors directly influences the severity of the mismatch between the transistors.

The phase detector can also introduce mismatch due to imperfect locking and leading to an offset. For optimal performance this offset, σ_{pd}, should be smaller than the variance of the delay cell. If so, it is absorbed in the offset introduced by each delay cell. If it is much larger, the last delay of the delay line by significantly larger.

4.3 Phase detector

The DLL can be considered a phase locked loop (PLL) with a frequency division of one. The phase detector therefor must update and track the signal at the rate of the reference clock. This makes the phase detector a possible bottleneck due the high operating frequency. Moreover, in a PLL, when the phases are locked it means that the phases move together, with or without a constant phase offset. The phases don't have to be equal because when the phases move together the frequencies are exactly equal. In the DLL when the phase detector locks it means that the phases must be exactly equal to each other. The phase-detector's offset can lead to problems in accurately locking the DLL.

The commonly used phase detector in PLLs is the phase frequency detector (PFD). This PD is used extensively in PLLs because it locks when the frequencies are far apart and when they are close together the phase is locked. In Figure 4.3, the PFD is shown. The operation of the PFD is as follows: when either of the signals goes high, one of the latches stores the output as high. When the other signal goes high both latches are set to high and the AND gate resets both latches. The duration of which one latch is on and the other off produces a signal which is proportional to the phase between the two signals.

With the two latch signals, denoted V_{up} and V_{down}, a charge-pump is driven. This charge-pump positively charges the charge-pumps capacitor when the Up signal is on longer than the Down signal and negatively charges when the down signal is on longer. When the two signals arrive at the same time, the net current entering the capacitor will be equal and the value of the capacitor will remain at a fixed value.

The PD's accuracy is determined by its speed and the charge pumps current accuracy. Due to the loading or too high operation frequency the up and down signal can arrive at different times, even if the input signals arrive at the same time. Another source of error is due to mismatch of the current sources, this will result in a different number of currents when the signals arrive at the same time. This will also incur a phase mismatch.

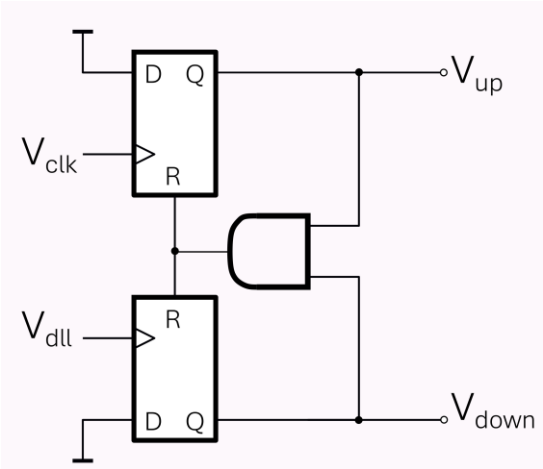


Figure 4.3: Phase frequency detector. Based on the arrival time of the CLK and DLL signal the up and down signal determine which signal is leading of lagging.

To reconcile these issues, the design of the phase detector and the charge pump have been extensively analysed and optimized to reach high speed operation.

In [11] they used a distributed approach to create 5 phases at 8-10 GHz by using a distributed delay line. By using multiple phase detectors, they enforce that all delays are equal. Then when all the phases are equal, the total loop is locked by the fact that the last delay should be equal to the first delay. With this approach high speed operation is achieved but it is narrowband.

In [12] they designed a very accurate charge pump which aims to eliminate the mismatch introduced by the phase detector. In these two papers, two examples of phase detectors where showcased. In this thesis a new architecture is proposed which uses a mixer to perform phase detection.

4.3.1 Mixing phase detector

Instead of using a PFD, a mixer can be used to measure the phase differences of two signals which have the same frequency. This is because when two signals have the same frequency the mixer will down convert the signal to DC. Unlike the PLL, where the two frequencies are different and would result in the mixer modulating the signal to an intermediate frequency which is not at DC. The amplitude of the DC value will depend on the phase difference of these signals. A lowpass filter can extract the DC signal by filtering the high frequency content. Using this property, the output frequency of the phase detector has been shifted away from the operating frequency. Consequently, the speed of the circuitry that follows after the mixer is lower. This would ideally result in a circuit that is frequency independent. Comparing this to the PFD, due to the high frequency the circuitry limits the maximum operation frequency.

The mixer has two issues which makes it difficult to use. Firstly, the acquisition range of the mixer is 180 degrees and secondly, the mixer has a dead zone at 0 and 360 degrees. The acquisition range refers to the fact that the polarity of gain of the mixer flips polarity after 180 degrees. The dead zone refers to the fact that the mixer produces little to no gain at these two points. This point coincides at exactly the point where the two signals are in phase, making it impossible to detect the difference of the signals lying close to each other or being exactly equal.

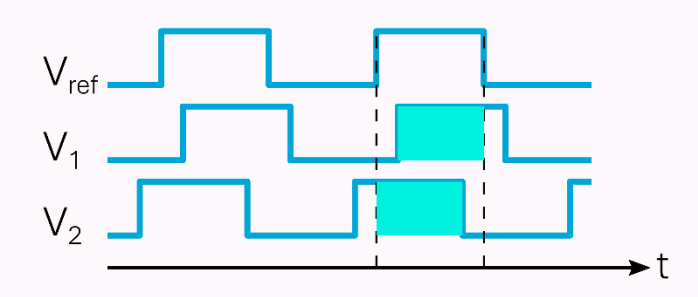


Figure 4.5: The shaded area represents the output of the mixer.

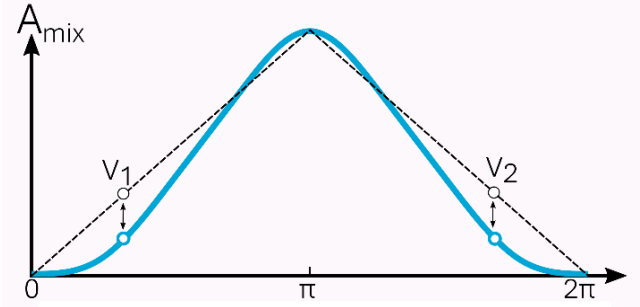


Figure 4.4: Near 0 and 2π the output of the mixer is less sensitive due to non-idealities of real switches.

In Figure 4.5 three signals are shown, with the first signal being the reference, V_{ref}, and the other two being the phase shifted versions of V_{ref} which are V_1 and V_2. The waveforms V_1 and V_2 are mixed with V_{ref}, which is indicated by the overlap of the two signals. As shown, for a leading or lagging signal the output of the mixer is the same, which gives ambiguity in resolving whether the signal is leading or lagging. This becomes a problem when trying to lock to the reference signal.

From Figure 4.4 it can be seen that to lock to the reference signal the output of the mixer should either be at 0 or 2π. In the case of V_1 it should be delayed less and V_2 should be delayed more. A feedback loop can be designed that takes the output of the mixer and uses it to bring the output to zero which would lock the loop. However, the sensitivity of the mixer changes polarity depending on if the phase difference lies between 0 and π or π and 2π. This requires additional knowledge of the relationship between the phase of the reference and the delayed signal.

The other problem is that at 0 and 2π there is a dead zone, due to the mixers nonidealities. Making it difficult to the detect the signal near the locking points of the mixer. Which will to a phase mismatch between the reference and the delayed signal.

Lastly the locking points are difficult to lock to, since when the signal passes zero, the polarity of the mixer changes. This means if the signal were to overshoot, it won't be able to get back to the origin. To alleviate these issues, a phase shift is added to the mixer.

4.3.2 Phase shifted Mixer

To fix the issue of the dead zone present in the mixer, a phase shift is introduced to move the locking point away from this dead zone. Now, when the reference is mixed with the delayed clock the output will be outside the dead zone. In Figure 4.6 the effect of the phase shift is shown showing that at 0 and 2π the mixer is out of the dead zone.

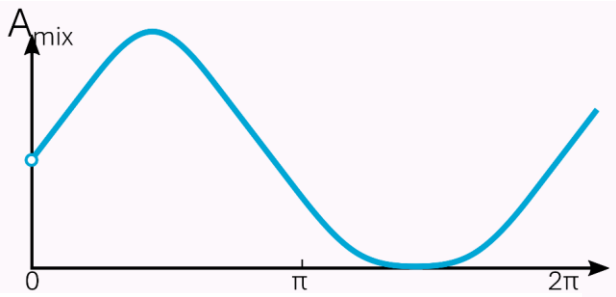


Figure 4.6: By introducing a phase shift to the mixer, the output of the mixer at 0 and 2π

A second property of this phase shift is that the mixer's output is monotonic around the locking point. This means that if the mixer were to overshoot above or under the locking point the slope remains the same, allowing the feedback loop to settle accurately around this point. In Figure 4.4, at its locking point the sign of the slope of the mixers output changes. This results in unstable behaviour at the locking points which makes it difficult to accurately lock at these points whereas comparing it with Figure 4.4.

By introducing this phase shift the locking point has been moved. To determine what the new locking is, the phase shift that is added needs to be very accurately determined to determine what output corresponds with the locking point.

The phase shifted mixer has two inputs, the reference and the delay line output, denoted f_{ref} and f_{DL}, respectively. The reference frequency is imposed with a phase shift given by delaying the reference clock denoted with T_D. The output of the mixer is proportional to the phase difference of the reference and the delay line output plus the additional phase shift interposed by the time delay. This yields the following equation:

$$A_{mix} = \cos\left(2\pi f_{ref}(t + T_D)\right) \cdot \cos(2\pi f_{DL}(t + T_{DL}) \quad (4.6)$$

Where T_{DLL} is the delay of the delay line and A_{mix} is the output of the mixer. Taking the average of this and setting f_{ref} *equal to* f_{dl} we obtain the following expression:

$$A_{mix,DC} = \frac{1}{2}\cos\left(2\pi f_{ref}(T_d - T_{DL})\right) \quad (4.7)$$

This quantity by itself is difficult to work with because it contains both the unknown delay, T_{DL}, and the fixed delay T_D, which is also unknown. To be able to use A_{mix} in the feedback loop the fixed delay has to be known. Because this is the point where the mixer has to lock to. To solve this, the reference signal is first mixed with the delayed version of the reference signal. The mixing product in this case gives the phase difference interposed by the delay. After determining this time delay, the delay line output is mixed with the delayed reference and the output will be compared to this value. In principle, if the reference is mixed with the delayed reference is equal to the delayed reference mixed with the delayed line signal, the phases must be equal and thus locked. To the comparison the two voltages will also need to be stored such that they can be compared. In Figure 4.7 this process is shown.

It is important that there is no mismatch in the acquisition of the both the reference and delayed phase. As this will immediately lead to mismatch. To mitigate this the output of the mixer is obtained through the same path

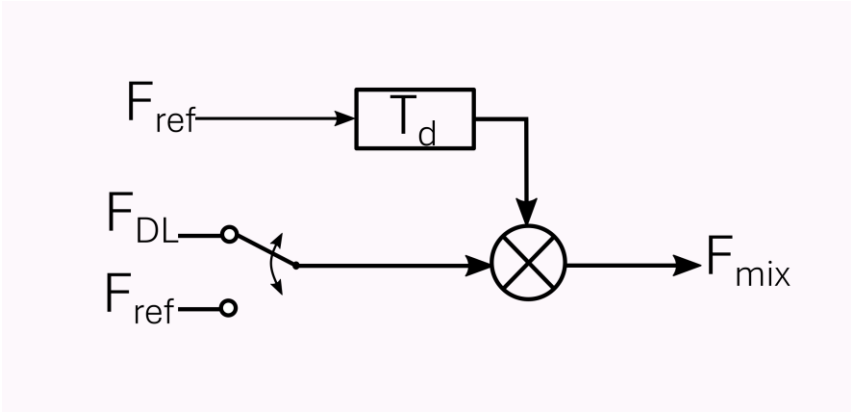


Figure 4.7: Acquisition of the delay of reference and delayed signal.

meaning that there won't be any mismatch due to the mixer at the output .

To conclude, the conventional phase detector used for DLLs proves challenging at high speeds, so it is important to mitigate the effects of the high-speed clock on the phase detector. With the mixing phase detector, the output will be down converted to DC, which makes the job of the phase detector and feedback loop easier as the circuit does not need to operate at the operating frequency. The next section will discuss the circuit implementation of the delay line first and then the chapter after, the circuit implementation and analysis of the mixing phase detector.

5 The Delay line

At the heart of the DLL lies the delay line. The delay line needs to operate at the clock's intended operational frequency, at least 10 GHz. For a 10 GHz clock frequency, the minimum delay between delay elements must be 12.5ps or lower to create eight phases with a 45-degree phase difference. However, creating such small delays becomes challenging because they approach the limit of how small they can be generated with sufficient swing in 40nm technology.

In this chapter, the design of the delay element will be covered first, then how the delay line is designed and finally how the delay line will be interfaced with from outside the chip and the phase-detector.

5.1 Delay elements

For the choice of delay elements, three delay elements were considered. Firstly, the current starved inverter, secondly the ratioed starved inverter, and lastly the current mode logic (CML) delay element. Which in the end was chosen for the CML delay element, as will be shown has the smallest delay.

To obtain minimum delays, ideally, single inverters are chosen instead of buffers. Because to create a buffer two inverters are needed, which will have a delay of two inverters. This arrangement doubles the minimum delay and does not allow maximum operating speeds. However, to using single inverters as delay elements is also not favorable. Inverters as delay elements have the problem that when ehy are chained, each output will be each 180-degree phase shifted between each consecutive delay element. A commonly used structure is applied in which a pseudo or true differential delay line is used in which the two branches of the differential cascade of inverters are fed with 180-degree shifted clocks. Then, by tapping the phases alternating from one branch to the other, the 180-degree phase shift operation is performed thus getting consecutive delays with minimum delay. In Figure 5.1, it can be seen how the delay elements are tapped alternatingly.

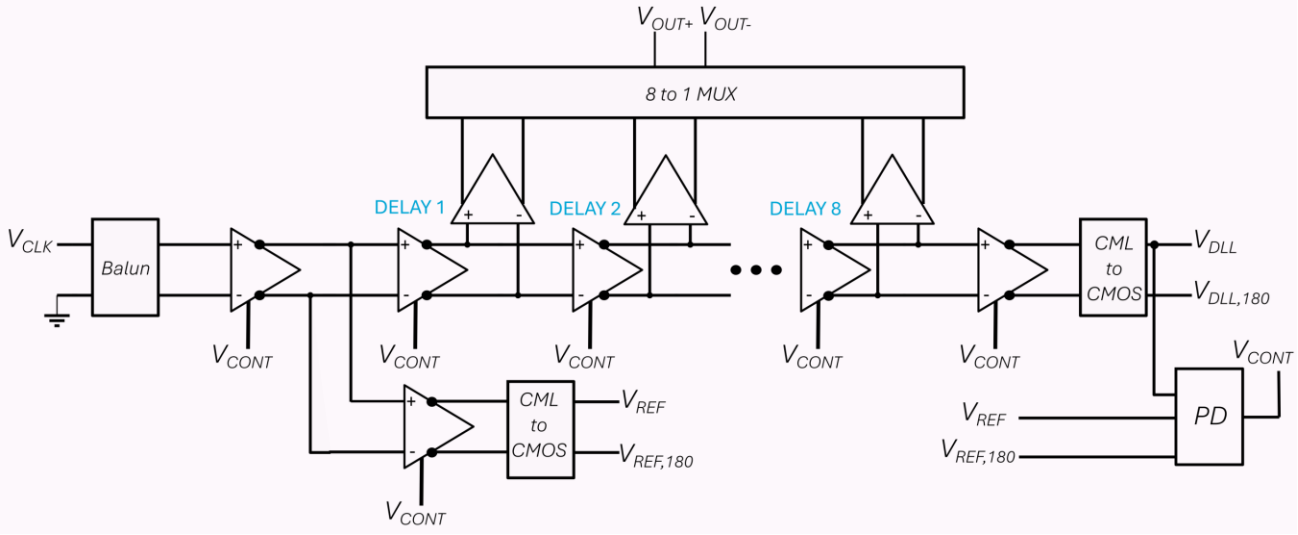


Figure 5.1: Block diagram of the delay line.

When tapping the delay of the inverters alternatingly, the delay of one branch is given by the rise time, but in the other branch by the fall time. In Figure 5.2, the differential signals are shown for the first three delays and how signal is constructed by alternating the delays.

The rise and fall time must be equal to obtain equal spaced delays in the reconstructed differential signal. However, the rise and fall time in a voltage-controlled delay line will not be equal because either the rise or fall time is adjusted to change the delay. A phase aligner is used to align the rise align the rise and fall time of the delay elements, ensuring that both branches are always 180 degrees phase shifted respect to each other.

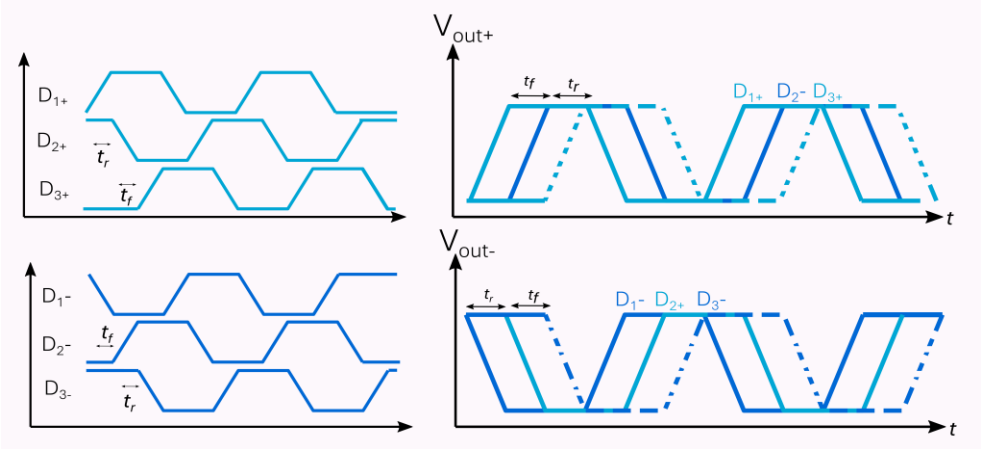


Figure 5.2: By tapping the differential signal alternatingly, minimum delays can be obtained in a delay line.

Each delay element is loaded with one delay-element and a buffer. To ensure that the last delay element also has the same loading, a dummy delay element is placed at the end. To compensate for the additional delay, the reference is also delayed by the same delay element. Figure 5.1, it can be seen how the last delay element is loaded with the dummy delay element and the reference is also passed through the delay element.

The input delay element is also a controlled delay element. This is done for two reasons. Firstly, it buffers the signal from the balun, which might have not enough voltage swing. Secondly, the rise and fall time presented to the first delay element will be the same as the rise and fall times of the delay elements. If the input of the first delay element has a much faster rise time than the previous delay element, the rise time of the first delay element will be different than the rest.

In the next section, the different delay elements will be discussed.

5.1.1 Cross-coupled current starved inverter

The first structure is the cross-coupled current starved inverter. In Figure 5.3, the cross coupled starved inverter is shown. Transistors M_3, M_6, M_7 and M_8 are two inverters which form the input of the delay element. The transistors M_1, M_2, M_9 and M_{10} are the current-starving transistors which limit the amount of current the cascode of PMOS transistors and NMOS transistors can source and sink, respectively. By controlling the gate voltage, the pull-up and pull-down speeds are varied.

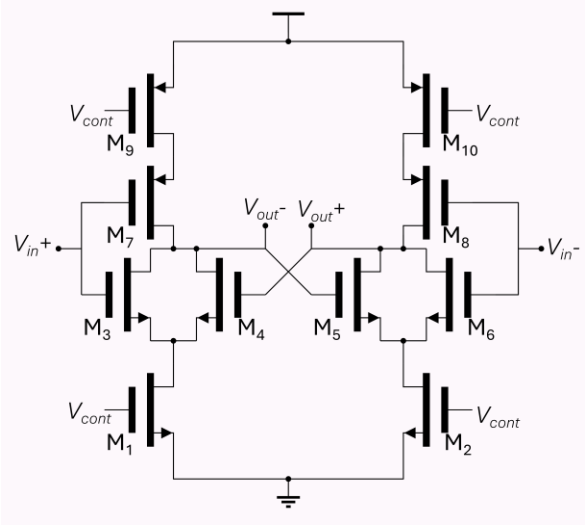


Figure 5.3: Schematic of cross-coupled current starved inverter.

The transistors M_4 and M_5 form the phase aligners. These two transistors are cross coupled by connecting the drain and source to one branch and the gate to the other branch. This creates a latching process that tries to enforce complementary outputs. Which ensures that when V_{out}- is high V_{out}+ is low and that during the transitioning from high to low and low to high the speed is equal.

In Figure 5.4, the delay versus the control voltage is plotted. As can be seen, the maximum speed of the circuit is 40ps which is too slow but the tunability range is sufficient. The circuit was designed using minimum sized transistors and the loading of each cell is equal to two inverters. The first inverter is the input of the next delay cell, and the second inverter is that of the output buffer.

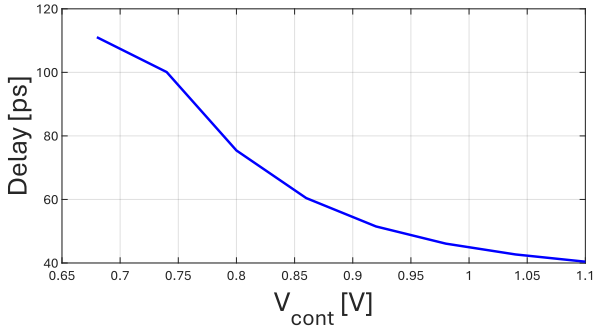


Figure 5.4: Delay versus control voltage of the current starved cross-coupled inverter.

To increase the speed, the capacitance or current must increase. Increasing the transistor size increases the current but the capacitance increases proportionally equally, which does not improve the performance. To improve the speed a ratioed circuit topology can be used. This reduces the input capacitance while the current sinking and drawing remains similar. However, this comes at the cost of a lower voltage swing.

5.1.2 Ratioed cross-coupled current starved inverter

In Figure 5.5 the circuit is shown using ratioed logic. The top PMOS transistors are removed compared with the non-ratioed circuit shown in Figure 5.3. This reduces the input capacitance caused by the PMOS gate transitor, which is twice the size of M_1. The top PMOS transistors are controlled by V_{cont}, which alters the pullup strength and thus changing the speed. The bottom pair of transitors are now also removed since the design is ratioed.

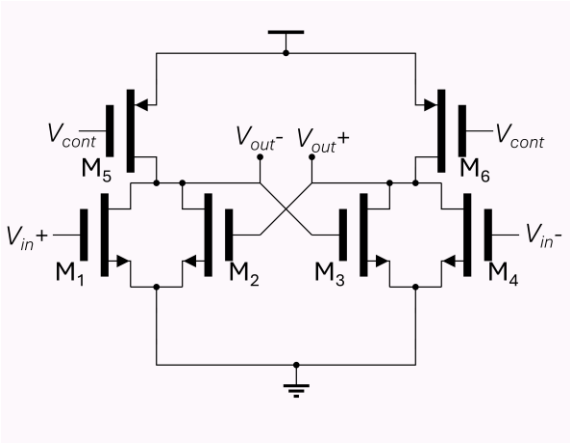


Figure 5.5: Ratioed current starved inverter.

In Figure 5.6, The delay versus control voltage is shown. Showing an improvement in the speed due to the lower gate capacitance. At its lowest it reaches a 12 ps delay and has a sufficient tuning range. Also, due to the ratioed design the speed of the circuit is maximum when the PMOS is completely turned on which happens when the control voltage is at zero volt.

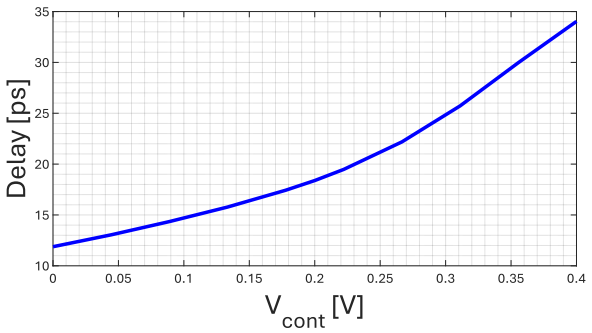


Figure 5.6: Delay versus control voltage of the ratioed delay element.

These two circuits work well Cadence's schematic editor, however, in post layout both these circuits perform worse. The typical performance penalty was a 2-3x speed reduction at the very high frequencies. The parastic capacitances of the interconnects are too large, which make these circuits not feasable.

To get the required speed, the circuit has to be oversized to produce a delay of 5-6 ps such that after parastic extraction (PEX), it will produce a delay of around 12.5 ps. The problem, however, with these architectures was there are no more turning knobs to improve speed since sizing all transitors larger does not improve the the speed. So, for this reason the last architecture, the CML delay line, was chosen to be used as the delay element.

5.1.3 CML delay element

The CML delay element is a differential pair with active or resistive load. The circuit bears similarity with the circuit shown in Figure 5.5 .The difference is that the tail current is determined by a current source, not the input pair. This means that the input capacitance does not need to increase for more current to be sourced. The tail current can be increased with some freedom with respect to the input pair. This increase cannot happen arbitrarily because the current source must stay in saturation, which is determined by the voltage drop of the input transistors. But it still allows for some freedom in the design.

Two conditions must hold for the CML delay element to operate at high speeds. Firstly, the current source must be in saturation, and secondly, the differential pair needs to fully current steer. If the current source is pushed out of saturation the speed of the circuit is reduced. Moreover, if the circuit does not fully current steer, the voltage swing of the next stage will be reduced. Which in turn reduces the voltage swing of the next stage, creating a shrinking voltage swing as the signal passes through the delay line.

In Figure 5.8, the CML delay element is shown. The circuit consists of the main differential pair, transistors (M_3, M_4, M_7 and M_8) and the phase aligner, transistors (M_2, M_5 and M_6). The active load controls the speed of the pull up. With the addition of the current sources M_1 and M_2 some freedom is given to improve the speed performance.

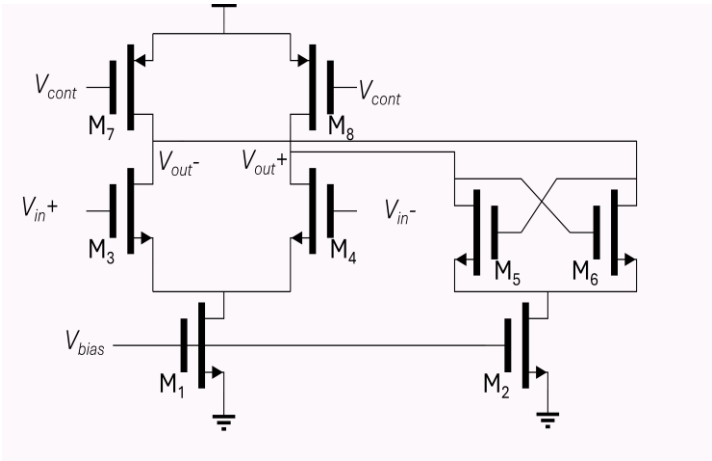


Figure 5.8: CML delay element.

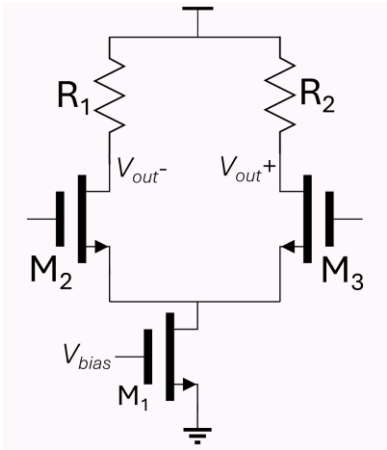


Figure 5.7: CML Buffer

Each delay cell is also loaded with a buffer to read the output phase. This buffer is also a CML delay element with but with a resistive load shown in Figure 5.7. Thus, the total loading of each delay cell equal to one delay element and one buffer.

5.1.4 Design of the CML delay element

The CML delay element is designed to work at the maximum possible frequency. This is done by imposing the smallest input capacitance while drawing the current. Once the sizing of the transistors has been determined, the circuit is scaled to meet mismatch requirements.

Firstly, the current sources M_1 and M_2 *must* be in saturation. This means that the transistors should be able to swing without pushing the current source out of saturation. This depends on the voltage drop of transistors M_3 and M_4 and the low voltage of the CML delay element. In Figure 5.9 the low voltage is shown.

The minimum voltage the CML can attain is determined by the voltage needed to fully current steer the next CML stage. As explained prior, if the current does not fully steer each consecutive stage of the delay line will have a smaller voltage and swing and the signal will diminish throughout the delay line. The minimum differential voltage for full differential steering is $\sqrt{2}(V_{GS}-V_{TH})$ where V_{GS} is the gate source voltage of the input pair and V_{TH} the threshold voltage.

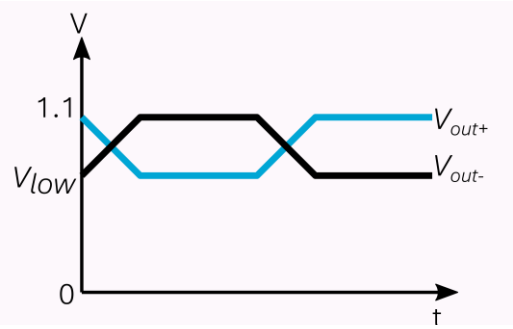


Figure 5.9: Ideal voltage swing of CML delay element

The VGS of the input pair is determined by the current determined by the current source. Making the input transistor much larger than the current source reduces the swing needed for current steering but leads to a large capacitance. Aa much smaller input transistor than the current source requires to a large voltage swing but a small input capacitance is obtained. It follows that the relationship between the input pair and the current source is determined by the voltage swing each delay cell should attain.

5.1.5 Layout vs Schematic

Because of the high frequency, a back-and-forth iterative approach had to be undertaken. The parasitic capacitance at V_{out}+ *and* V_{out+} increasingly loads the CML. These additional capacitances stem from:

1. Input pair of the differential pair and phase aligner
2. Active load
3. Interconnects to the buffer

These capacitances are unavoidable and scale together. With proper layout coupling from the supply, ground and other signals can be avoided but there is a baseline of parasitic capacitance. To improve the speed the sizing of the components can still be adjusted to improve the speed in the layout.

The tail current is increased so that the speed increases and to make sure that the input pair does not push the tail in to saturation. By increasing the input pair, the input capacitance also increases but the total capacitance has not equally increased. Namely, in the schematic the dominant capacitance is the gate capacitance, but in layout it is not. It is the summation of the parasitics, which become a big part of the total capacitance at the very high frequencies. Consequently, the voltage swing does increase because more current is sourced.

The final values determined for the CML circuit are tabulated in the table below.

Table 5.1: CML transistor sizing

	Transistors	*Width*
Current source, main	M_1	20u
Current source, auxiliary	M_2	10u
Input pair, main	M_3 *and* M_4	3u
Input pair, auxiliary	M_5 *and* M_6	1.5u
Active load	M_7 *and* M_8	6u

The phase aligner, referred to as the auxiliary pair, is sized to be half that of the main differential pair. This was found to provide good phase alignment, but at the expense of some speed. The current sources are biased at 0.6V to provide 1mA to the main differential pair and buffer, and 500uA for the auxiliary pair. Which totals 2.5mA per delay element or 2.75mW.

In Figure 5.11 and Figure 5.10 the ideal wave from schematic and the post layout waveform is plotted. From the post layout simulation, it can be seen that the delays have been reduced. The delays from schematic are around 5 ps. The delays from post layout have been increased to 10.7 ps. Showing that at 10 GHZ the DLL is able to lock at 10 GHz.

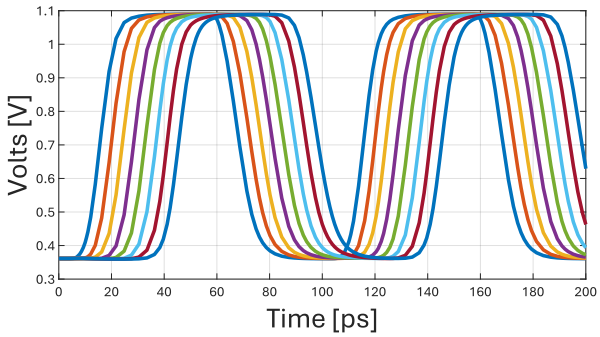


Figure 5.11: Simulated waveform of the CML delay elements.

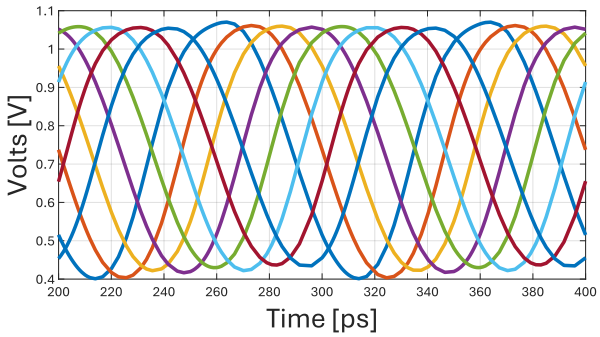


Figure 5.10: Post layout simulation waveform of the CML delay.

In Figure 5.12, the delay versus control voltage is plotted from the extracted layout is plotted. Showing a minimum delay of 10.7 ps to 30 ps.

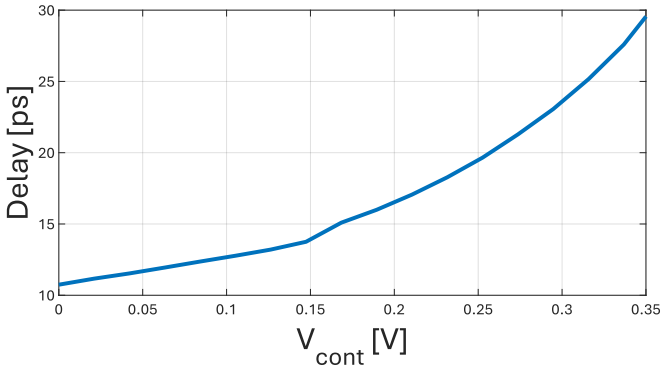


Figure 5.12: Delay vs control voltage of the delay cell.

5.1.6 Phase accuracy

The mismatch stems primarily from the mismatch between the delay cells, which is proportional to the size of the transistors. By increasing the size, the mismatch goes up with the square of the area, the power however, goes up linearly.

In Figure 5.13, the mismatch of the delay element is shown, which gives a standard deviation of 500fs with an average current draw of 580 uA. This means that the standard deviation for the middle delay is 1 ps. Relating this to phase accuracy for a 10 GHz clock this would result in a phase deviation of 3.6 degrees. To further reduce the error lets ay be a half the sizing of the transistor would have to increase by a factor four. This would lead to 2 mA per delay element. Without calibration the cost of getting high accuracy is costly in power.

Efforts were made by making the current source relatively large and decreasing the gate voltage to allow the littlest current for the most amount of size. The input pair was sized to be as small as possible with regard

to the speed performance and the active load was scaled in accordance with the rest of the transistors. To ensure that post layout the mismatch and process variation was kept to a minimum the current sources were laid out in a common centroid fashion. The input pairs could not be interleaved due to the constraint parasitic capacitances introduced by the coupling.

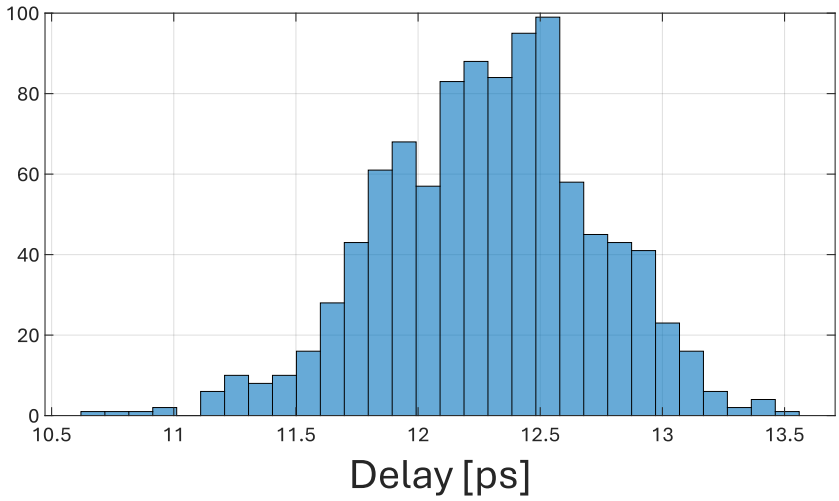


Figure 5.13: Montecarlo simulation of 1000 trails with a mean of 12.28 ps and standard deviation of 0.43 fs

To further reduce the mismatch calibration techniques must be used, shown in [11]. With such structures the potential error can be minimized, and the sizing of the devices can be reduced to reach phase requirements. In this design, none of such techniques have been employed since the focus of this DLL is the phase detector.

5.2 Balun

The balun creates a differential signal from the signal-ended input. It should ensure that the amplitudes and phases are preserved properly from the generator to the chip. A phase aligner in the input pair can slightly correct amplitude and phase imbalance, but it leads to a duty cycle variation and is unfavorable.

The choice of balun was between active and passive. The advantages of using active baluns are that their area is comparatively lower and that they are easier to use over a wide range due to their wideband capability. Secondly, the use of active components can increase the gain of the signal, alleviating any attenuation from the source. The drawbacks are the power consumption and complexity of the circuit.

The alternative is a passive balun constructed from a transformer. The advantage of this circuit is the simplicity of the design and that it does not consume power. However, the balun is comparatively large and requires more area than the active circuit. Lastly, the circuit has to be matched at an operating frequency which will make it more narrow band. Despite these concerns, due to the ease of the design and robustness the passive balun was chosen.

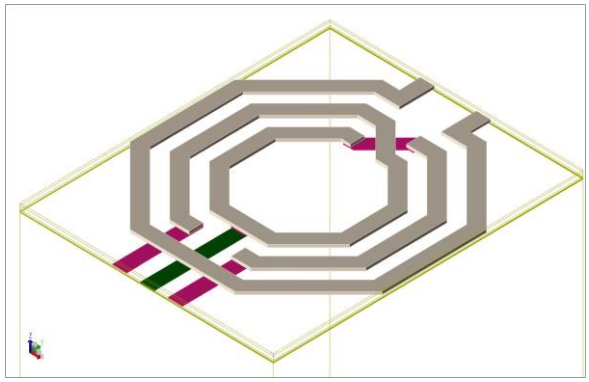


Figure 5.14: Balun

For the balun, a planer transformer was used. It consists of one spiral for the primary and two spirals for the secondary with a center tap as can be seen in Figure 5.14. The design flow for creating the balun was primarily an iterative approach. The inductances needed to be large enough and the capacitive coupling minimal. The requirements for the balun are not that stringent since the only goal is to obtain a differential signal. This differential signal will also be amplified before it is fed to the delay line. Imperfect matching will lead to a smaller voltage swing at the input, but this can be solved with a larger input voltage.

In ADS EM simulations were done to characterize the balun. The following parameters were obtained.

- L_p=730 pH
- L_s=1250 pH
- M = 400 pH
- k =0.4

In Figure 5.15 the inductance is plotted where the inductance stays relatively flat up until 10 GHz

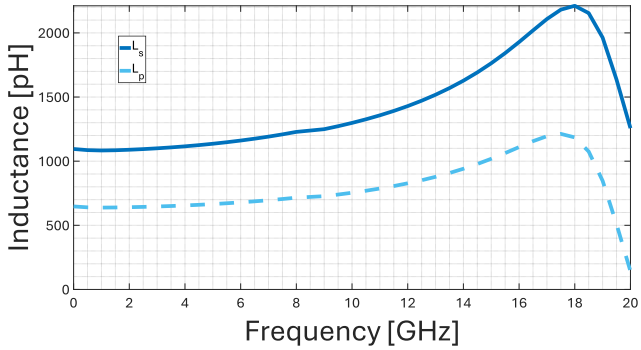


Figure 5.15: Inductance of the transformer vs frequency.

5.2.1 Balun matching

The transformer can be modeled with his T-equivalent circuit where we have the following parameters.

$$L_a = L_p - M, L_b = L_s - M, L_c = M \quad (5.1)$$

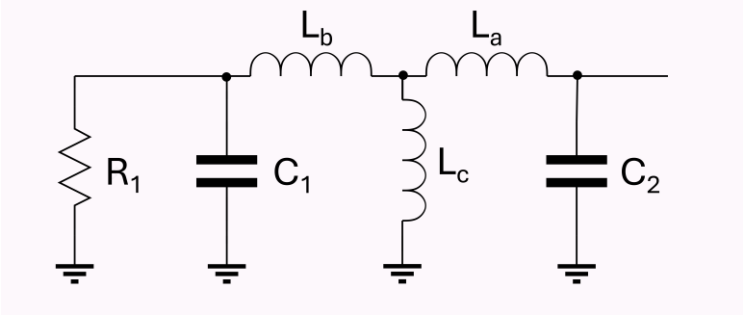


Figure 5.16: Simplified model of the balun

To match the balun, the capacitor, C_1 , is added to match the impedance to 50 ohms. Capacitor C_2 , is the capacitance due to the pad of the connection which is around 300 fF. The analytical solution of this circuit is difficult and not completely accurate due to coupling capacitances not included in this model. So, to match the circuit ADS was used and C_1 is tuned.

For R_1 a 400 Ohm resistor was used to ensure that the real resistance is a lot larger than the reactance of the inductors. Which will yield the maximum swing. In Figure 5.17, S_{11} is plotted

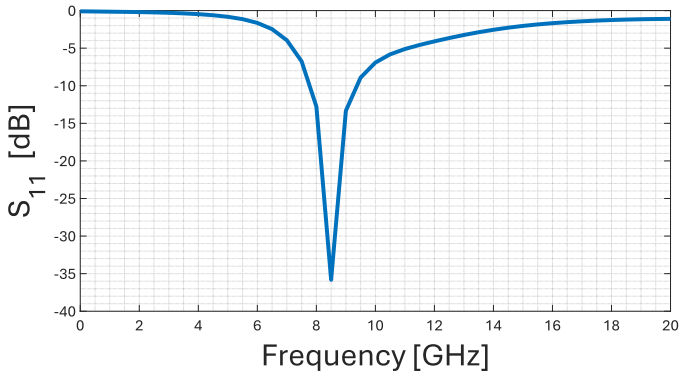


Figure 5.17: Matching of the Balun

5.2.2 CML to CMOS

The CML to CMOS converter is an OTA that converts the differential signal into a single ended full swing signal. The OTA is an NMOS device and is buffered with an additional inverter to ensure full swing.

In Figure 5.18, the OTA is shown. An additional inverter is added to create the complementary full swing signal. The complementary signal is used for the phase detector as the reference phase. In the previous chapter

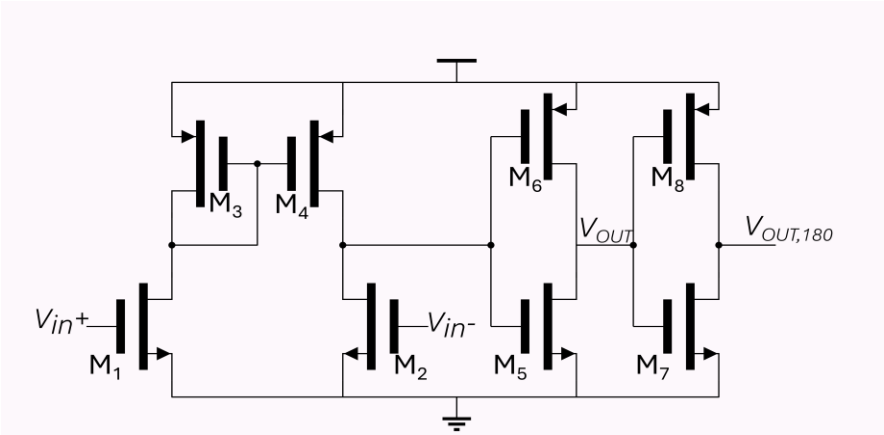


Figure 5.18: CML to CMOS converter.

it was explained that for the phase detector the reference needs to be mixed with itself (see Figure 4.7). If this signal is tapped twice from the same output the loading will be different and will introduce a phase offset. To ensure equal loading, both the reference and delayed signal are buffer with another inverter, but only the reference inverted signal is used twice. In Figure 5.18, the circuit is shown and in Table 5.2 the values.

Table 5.2: CML to CMOS component values

	Transistors	*Width*
CML to CMOS input	M_1,M_2	3u
Load	M_3, M_4	7u
Inverters	M_5 *and* M_7	1u
Inverters	M_6 *and* M_8	2u

5.2.3 8:1 multiplexer

The 8 to 1 multiplexer is used to read out the phases of the delay line from the chip. In the real application, the DLL is embedded in the chip and this multiplexer would be replaced with a buffer so that the signals can be used on chip. The power consumption of the 8:1 multiplexer is therefore not considered in the performance of the DLL, but it does affect the phase accuracy of the individual signals. To ensure that the MUX does not corrupt the nature of the phases the multiplexer was designed as oversized as possible.

The structure of the multiplexer is created by using a chain of 2:1 CML multiplexers as shown in Figure 5.19. Each stage is scaled larger than the previous stage. The scaling factor was empirically determined based on the performance of the post layout simulation. It was found that scaling each MUX by a factor of two works.

The active load is replaced by resistors which have better matching performance compared to the MOSFETS.

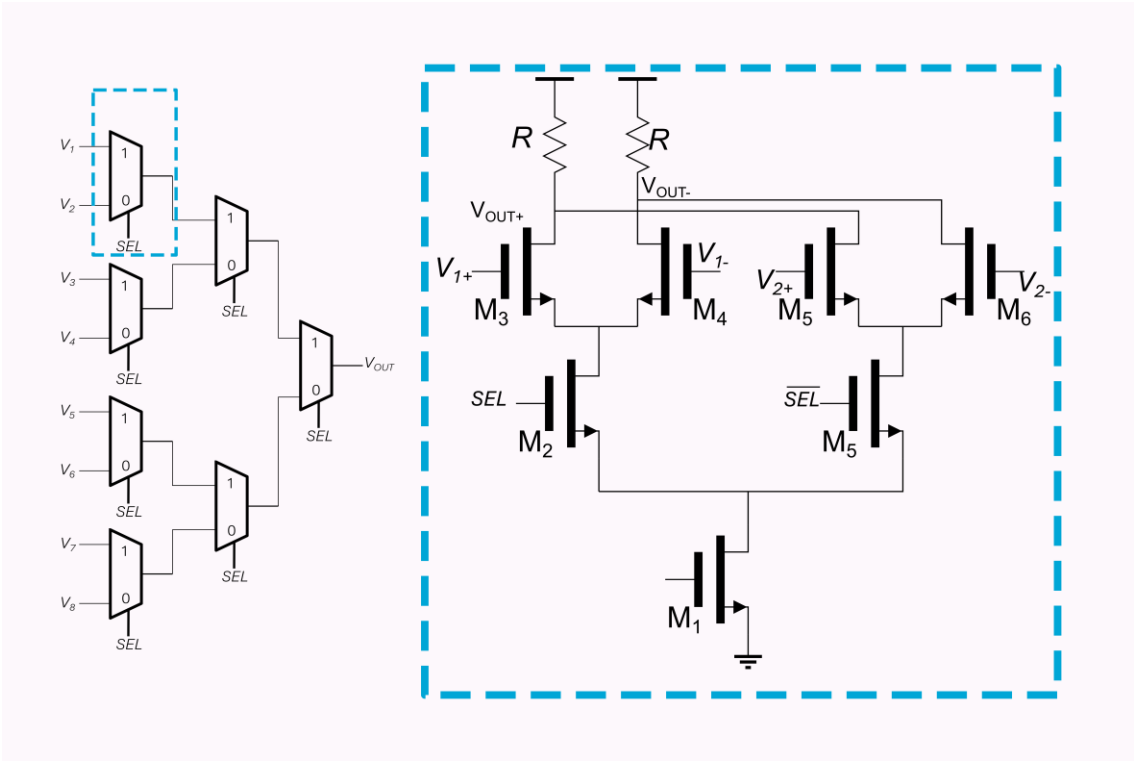


Figure 5.19: CML 8:1 multiplexer constructed by a tree of 2:1 multiplexers.

Lastly a final CML is added to drive the 50 Ohm termination that will match the output impedance to the measurement device. The sizing of the transistors and resistor sizes are tabulated below.

Table 5.3: CML MUX transistor sizing

	Transistors	*Mux [1]*	*Mux [2]*	*Mux [3]*	*Output driver*
Current source	M_2	20u	40u	80u	160u
Input pair	M_3 *and* M_4	3u	6u	12u	24u
Resistor	-	600 Ohm	300 Ohm	150 Ohm	50 Ohm
Current (biased at 0.6v)		1m	2m	4m	8m

6 Mixing Phase detector

In this section, the design of the proposed phase detector is detailed. As stated in Chapter 4; by introducing a phase shift, the non-idealities of the mixer can be fixed. Using this technique, the mixer's deadband is removed and the slope of the mixer's transfer curve is made monotonic at the locking point. This allows for a feedback loop to be used that can accurately lock the two phases with a mixer as a phase detector.

First, the mixing phase detector will be qualitatively described, and then the design of each component will be detailed.

6.1 *Analysis*

In Figure 6.1, the block diagram of the proposed phase detector architecture is shown. The reference signal is applied to the delay line's input and propagates through the delay line. The multiplexer either selects the reference clock or the delay line's delayed clock. When the multiplexer's select signal, *SEL*, is *LOW*, it is referred to as the acquisition period and, when the select signal is *HIGH, the* calibration period takes place. During the calibration period, the reference signal is mixed with the time shifted version of itself. Where, T_D represents this time shift. By mixing the reference with itself, the unknown time delay of the delay block is determined. This is done to exactly determine the value of T_D, which is very PVT (process voltage temperature) sensitive value.
The mixer's output is then connected to a low-pass filter to obtain the DC average, stored on the capacitor, C_1.

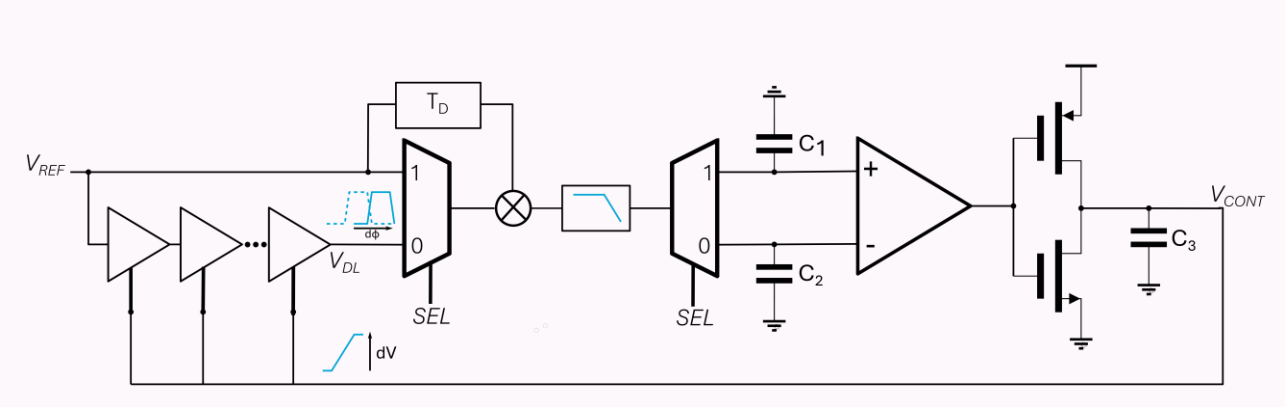


Figure 6.1: Block diagram of the phase detector.

After the calibration period, the mixer will be biased in the phase domain at a point with monotonic gain. In Figure 6.2, this biasing process is shown in both the phase and time domain. In the phase domain, the locking point is shifted away from the origin. In the time domain, a first order response is seen where the voltage settles to the DC value which stores the biasing point of the mixer.

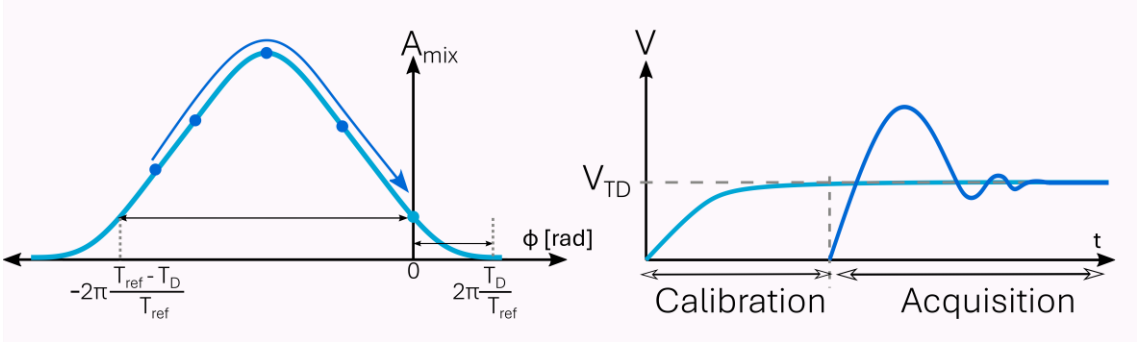


Figure 6.2: Phase and time domain representation of going from an unlocked state to a locked state.

After the calibration phase, the acquisition phase starts. The delay line's output is mixed with the reference and stored on the other capacitor, C_2. Initially, the delay line is not locked and trails behind the reference signal. If the delay lines phase lies between 0 and $-(T_{ref} - T_D)$, the mixers voltage will increase, because the comparator sees that the mixers output voltage is larger than the stored calibration voltage. Subsequently, the comparator drives the charge-pump, which then charges the capacitor, C_3. An increase in voltage leads, in this case, to an increase in the delay of each delay element. This will bring the delay line's signal closer to the reference. Once the voltages are equal, the delay line's output will be locked to the reference. In Figure 6.2, first, the mixer's output voltage experiences transient behavior, the voltage overshoots, rings around the target point and then locks. This is only possible due to the monotonic gain at the biasing point. Without it, it would become unstable.

The signal's phase must lie between 0 and $-(T_{ref} - T_D)$. Otherwise, the signal will lie below the calibration voltage. In Figure 6.2, if the voltage were to be lower than the calibration voltage, the comparator would discharge the capacitor. It would then try to lock the delay line to -2π. This would lead to a false locking situation because it tries to lock to the wrong point. Luckily, it will be shown that this situation cannot occur.

The delay line has a minimum delay, which will be denoted with T_0. This delay will always offset the delay line with respect to the reference, irrespective of the operating frequency. In the phase domain, it follows that the delay line starts at $-2\pi(1-(T_0-T_D)/T_{ref})$. When the frequency decreases, the period increases which shifts the delayed signal farther from the locking point until it approaches $-2\pi(T_{ref} - T_D)/T_{ref}$. If we then take the limit of the reference period to go to infinity, the following inequality is obtained:

$$\lim_{T_{\text{ref}}\to\infty} -2\pi\left(1-\frac{T_0-T_D}{T_{ref}}\right) \geq \lim_{T_{\text{ref}}\to\infty} -2\pi\frac{T_{\text{ref}}-T_D}{T_{\text{ref}}} \quad T_0 > T_D \qquad \textbf{(6.1)}$$

$$-2\pi \geq -2\pi \qquad \textbf{(6.2)}$$

From (6.1) and (6.2), it follows that if T_{ref} becomes arbitrarily large, it will never become smaller than the calibration phase and will always be able to lock. The only condition that must be held is that the minimum delay of the delay line is larger than the phase shift ,i.e., $T_0 > T_D$ .This proves shows that the acquisition range of the mixing phase detector is 360 degrees. This aspect of the proposed phase detector solves the issue of the limited acquisition range of the mixer as well.

With this qualitative analysis done, the details of each block will be explained.

6.2 *Mixer*

A simple NMOS *AND*-gate-based-mixer is chosen as the mixer in the proposed phase detector (see Figure 6.3). Transistors M_6 and M_7 are the inputs from the reference clock and the delay line, respectively. The transistors M_4 and M_5 form the multiplexer, which allow the circuit to switch between the reference and the delay line's output. The bottom switch is the common switch for both M_6 and M_7 The delayed version of the reference clock.

To create the delay, an inverter was used, which is driven by the inverted reference signal. This inversion was done to ensure that the reference and delayed line are equally loaded. Both the reference and delay line produce the 180 degrees shift, but only the reference is used.

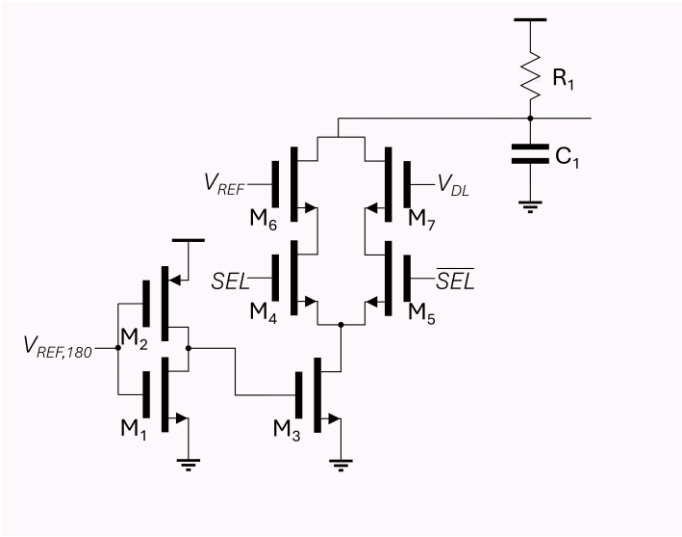


Figure 6.3: NMOS mixer for phase detection.

By having few components, the mismatch between the two paths is reduced. In this case most of the mismatch stems from the difference between the four transistors (M_6,M_4,M_5,M_7). All the other transistors, resistors and capacitors are common. Compare this to the conventional phase detector where the mismatch stems from the difference in current from the current sources and the difference in UP and DOWN signal from the PFD. The final circuit design parameters of the mixer are tabulated in Table 6.1.

Table 6.1: Component values of the mixer

Component	*Values*
M_3, M_4, M_6 and M_7	3u
M_1	1u
M_2	1u
R_1	8.5 kOhm
C_1	200fF

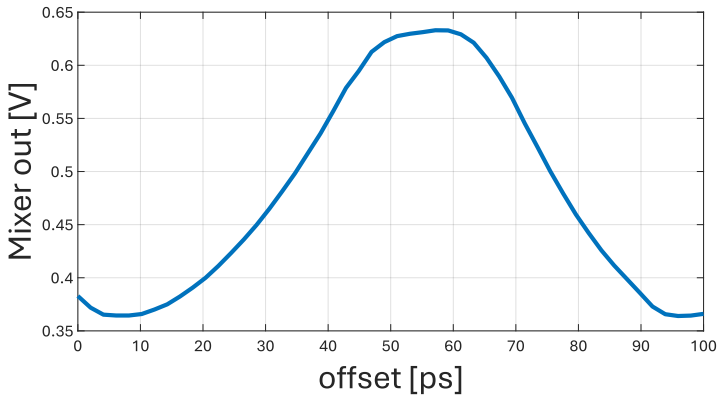


Figure 6.4: Transfer function of the mixer.

In Figure 6.4, the transfer of the mixer is shown at 11 GHz. The transfer flattens at the maximum and minimum due to the dead zone. By imposing the phase shift the curve is shifted such that at zero picoseconds the output voltage is monotonic.

6.3 Calibration and acquisition circuit

As stated in the analysis section, in the calibration and acquisition phase, the phase detector samples the clock reference and the delayed signal from of delay line. Using the two values sampled and stored onto the C_1 and C_2 capacitors, the comparator compares the signals to create the signal required to drive the charge pump.

The comparator produces a voltage of $½V_{DD}$ when both capacitor values are equal. At this voltage, the charge-pump should stop charging the capacitor. The transistors can be sized properly such that when the comparator reaches $½V_{DD}$, the NMOS and PMOS are turned on equally, and the net charge flowing into the C_3 is zero, as seen in Figure 6.5. However, as soon as there is a difference between these two thresholds, the charge-pump will continue charging the capacitor.

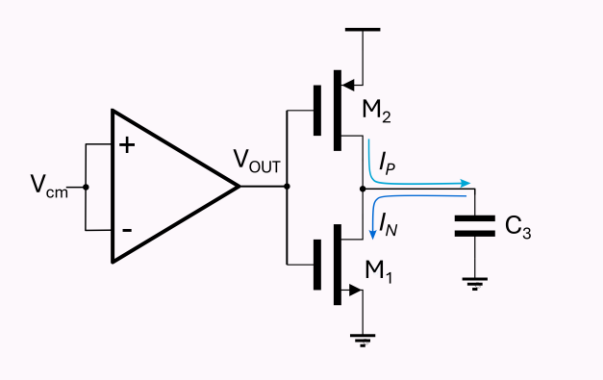


Figure 6.5: When the input of the comparator is equal, the net current flowing into C_3 should be zero.

Due to channel length modulation, as the voltage increases, the currents, I_P and I_N will change too. So, when the capacitor keeps on charging, eventually, the two currents will be equal, and the capacitor will stop charging. This point at which the capacitor stops charging or discharging is easily susceptible to change, either due to the voltage at the drains of the transistors or the sizing. Another error source stems from the comparator. The comparator's offset will cause the output voltage to be larger or smaller than $½V_{DD}$.

To eliminate these sources of mismatch, a servo loop is used, which removes all the offsets, by utilizing an auto-zero controller. In Figure 6.6, the proposed servo loop is shown. During the calibration phase, the *SEL* signal is set low, and the reference signal from the capacitor is read. The output from the charge-pump is now connected to the autozeroing capacitor. The charge-pump will steadily charge (or discharge) the capacitor until the voltage settles. This settling point is where the voltage across C_{AZ} stops charging the capacitor. In Figure 6.6, it can be seen how the settling occurs. First, the auto-zero capacitor is discharged, and the comparator sees that the reference voltage is larger than the auto-zero voltage. The comparator then goes high and charges the capacitor. When the charge-pump stops charging the offset V_{os} is found and compensated for with the auto-zero capacitor.

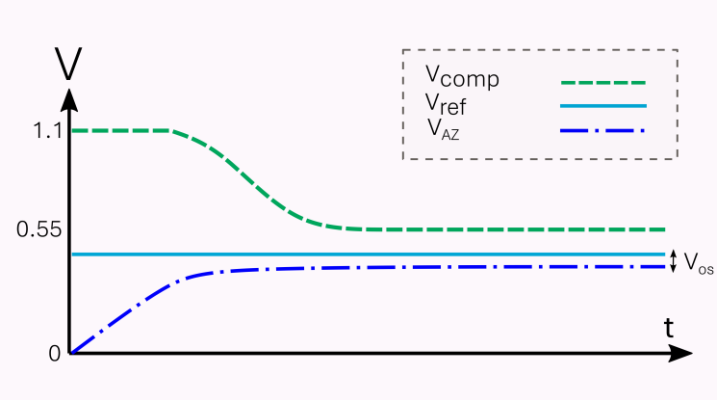


Figure 6.6: Settling of the auto-zeroing process.

After the calibration period, the value has been stored and will be used as the voltage comparator will compare with C_{DL}. When the *SEL* signal is set to *HIGH*, the voltage stored on C_{DL} is compared with the voltage stored on C_{AZ}. If there is a difference between the reference and delay line capacitor the charge pump capacitor, will be charged until the difference between the capacitors is zero. Because of the autozeroing, it can be assured that the capacitor voltages are equal.

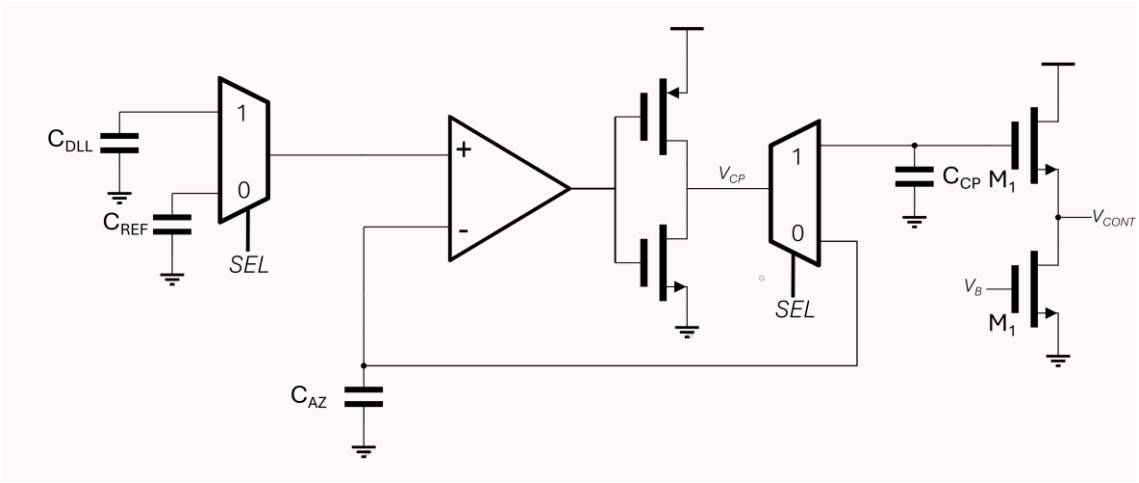


Figure 6.7: Auto zero loop to remove offsets from the OTA and charge-pump.

The voltage at the output of the charge-pump during the acquisition and calibration periods will be different. During the calibration phase, the output of the charge-pump will be the value of the auto-zeroing capacitor. During the acquisition period, the charge pump drives the delay line's control voltage. The output impedance of the charge-pump must be large such that the effects of channel length modulation will not influence the source and sink strength of the PMOS and NMOS too much. Otherwise, an offset will be incurred going from the calibration to the acquisition period.

To further ensure that the charge-pump's performance remains constant a level shift is added to the output. In Figure 5.12 of Chapter 4, it can be seen that the voltage ranges from 0 to 0.4 volts. For the charge pump to produce such a low voltage, the NMOS will be pushed into the triode region. By adding a level shifter, it can be assured that when a zero-volt output is needed, the output at the charge-pump is raised. A source follower creates this level, which shifts the level by one threshold voltage, which is about 400 mV (low V_T). This puts the charge-pump at a more comfortable operating region from 400 to 800 mV.

6.3.1 Comparator

The comparator is a simple 5 transistor OTA. The gain is not important, because the charge-pump (integrator) provides infinite loop gain. The bandwidth of the OTA is also not too much of a concern since the signal is down converted to DC. It only has to track the change of this DC value, which depends on how quickly the mixers output changes. This in turn, is determined by the rate at which the integrator charges the capacitor, which changes the control voltage of the delay line. If the capacitor is relatively large, the bandwidth is determined by the charge pump capacitor.

For the OTA, a PMOS input pair was chosen since the output swing of the mixer at the locking point is low (see Figure 6.4.) For the sizes, all transistors where sized with transistor lengths of 1um. In Figure 6.8, the implementation of the 5-transistor OTA is shown.

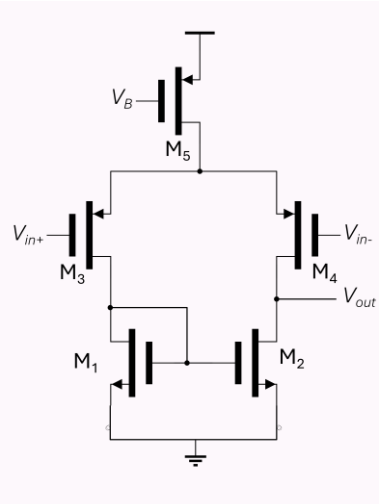


Figure 6.8: 5-OTA

6.3.2 Charge pump

For the charge pump, a gain-boosted, complementary circuit is used. The gain boosting was used because the self-gain of a transistor in 40nm is relatively little (6-10x), and due to a low voltage supply, the amount of cascoding permitted is limited. Similar to the OTA, the charge pump operates at a low frequency. Thus, the bandwidth of the charge-pump does not need to be large.

In Figure 6.9, the circuit is shown. The transistors M_3 and M_7 are the input pair of the charge pump. Transistors M_4 and M_5 are the cascode transistors. The gain of the cascode is $g_{m3}r_{o3}(1+g_{m4}r_{o4})$. By adding the common source amplifier, the source of M_4 is tried to be fixed in place. When the source goes down the gate of M_4 goes up via the common source amplifier. This boosts the output impedance by another $g_{m2}r_{o2}$. From the simulation a gain of 32 dB was obtained.

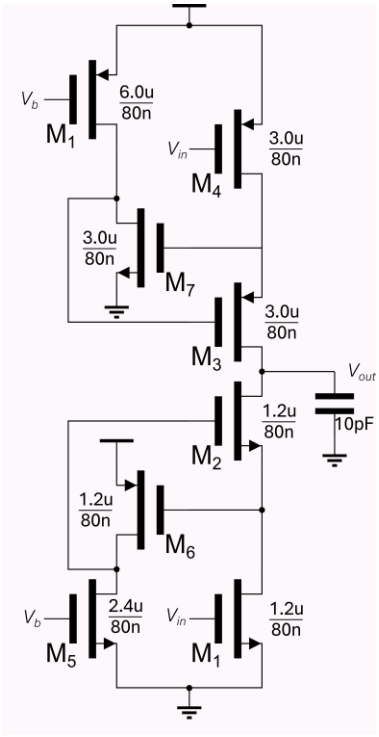


Figure 6.9: Gain boosted charge-pump

6.4 Design Equations

With the components designed, the whole circuit is put together and additional some equations are derived, which mathematically explain the working foundation of the MPD. With these equations some design choices can be made.

This analysis looks at the DLL going from an unlocked state to a locked state. The mixer produces an output voltage that is proportional to the difference of the reference and the delayed signal given by (5.4),

$$V_{DL} = I_{MIX} \cdot \frac{R_{MIX}}{1 + sR_{MIX}C_{MIX}} (\phi_{DL} - \phi_{REF}) \quad \textbf{(6.3)}$$

The reference value is also obtained from the mixer, but it has already settled which yields (5.4)

$$V_{REF} = I_{MIX}R_{MIX}\phi_{REF} \quad \textbf{(6.4)}$$

The OTA senses the differences and amplifies the difference, combining equation (5.4) with (5.5). The following relationship is obtained

$$V_{OTA} = A_0(V_{DLL} - V_{REF}) = A_o I_{MIX} R_{MIX} \left(\frac{\phi_{DLL} - \phi_{REF}}{1 + sR_{MIX}C_{MIX}}\right) \quad \textbf{(6.5)}$$

The OTA drives the charge-pump which generates the control signal for the delay line. The control voltage is given by (5.7)

$$V_{cont} = A_0(V_{DLL} - V_{REF})\frac{I_{CP}}{sC_{CP}} = A_o I_{MIX} R_{MIX} \left(\frac{\phi_{DLL} - \phi_{REF}}{1 + sR_{MIX}C_{MIX}}\right)\frac{I_{CP}}{sC_{CP}} \quad \textbf{(6.6)}$$

The sensitivity of the delay line is given by (6.8)

$$\phi_{DLL} = V_{cont} \cdot K_{DLL} \cdot \frac{2\pi}{T_0} \quad \textbf{(6.7)}$$

Equating equation (5.7) with (5.8) yields

$$A_o I_{MIX} R_{MIX} \left(\frac{\phi_{DLL} - \phi_{REF}}{1 + sR_{MIX}C_{MIX}}\right)\frac{I_{CP}}{sC_{CP}} = \frac{\phi_{DLL}}{2\pi K_{DL}} T_0 \quad \textbf{(6.9)}$$

Solving the equation results in:

$$\frac{\phi_{DLL}}{\phi_{REF}} = \frac{A_0 I_{MIX} R_{MIX} g_m}{\frac{T_0}{2\pi K_{DL}} R_{MIX} C_{MIX} s^2 + \frac{T_0}{2\pi K_{DL}} s + \frac{I_{CP}}{C_{cp}} A_o I_{MIX} R_{MIX}} \quad (6.10)$$

Which yields a second0order transfer function from which the damping and natural frequency are the following expressions

$$\omega_n = \sqrt{\frac{2\pi K_{DL}}{T_0} \cdot \frac{A_0 I_{mix} I_{CP}}{C_{mix} C_{CP}}}, \zeta = \frac{1}{2R_{MIX} C_{MIX}} \sqrt{\frac{T_0}{2\pi K_{DL}} \frac{C_{MIX} C_{CP}}{A_0 I_{MIX} I_{CP}}} \quad (6.11)$$

From the equations in (5.10) some useful conclusions can be drawn. The gain of the comparator does not need to be large because the charge pump can be sized to determine the speed of the DLL. This is good, because making the comparator's gain excessively large can quickly lead to the system to oscillate due to the comparator clipping.

Another conclusion is that the settling is frequency-dependant. At high frequencies, the system's bandwidth increases, and the damping becomes less. As the frequency reduces the system becomes more stable because the damping starts to increase, and it will oscillate less.
So, to design the DLL the system should be stable at the highest frequency because then it will be stable for a lower frequency. In the next chapter the simulations of the DLL are shown.

6.5 Phase noise

One of the strengths off the DLL is its resilience against phase noise. This is because to the propagating signal does not recirculate back to the input of the delay line, which eliminates the accumulation of phase noise. The phase noise that does appear is a result of the instantaneous phase variations injected by each delay element, which worsens the farther the through the line it travels. Another source that controls the instantaneous phase, is the control voltage supplied by the phase detector. It will be shown however, that this contribution from the phase detector is negligible and that it is limited by the phase noise contributions of the delay element in the delay line.

From equation (6.7), it was derived that the control voltage changes the phase determined by the sensitivity, K_{DL}. The noise power of the control voltage is given by

$$\phi_{DLL}^2 = |V_n^2| \cdot \left(K_{DLL} \cdot \frac{2\pi}{T_0}\right)^2 \quad (6.12)$$

The noise contributing sources are

- The input pair of the differential pair
- The active load
- The phase detector (charge pump, mixer, etc)

The noise injected by the differential pair is directly seen at the output, the noise from the phase detector however is shaped by the integrator of the charge pump. So, if we compare the two noise powers we get

$$|V_{n,diff}^2| = V_{PD}^2 | \frac{g_{m,cp}}{\omega^2 C_{cp}^2} | \quad (6.13)$$

It follows that to the phase noise contribution can be many orders larger than the differential pair. Especially, when a large capacitor is added. So, the DLL is truly limited by the phase noise introduced by the delay elements.

7 DLL Results

In this chapter, post extracted simulations are performed of the chip. First, the locking process of the DLL is shown. Then, Montecarlo simulations are run, which will demonstrate the robustness of the architecture Lastly, the results will be compared with the state of the art.

7.1 DLL

The first step is the calibration phase where the mixer determines the locking voltage. The reference voltage is selected and is mixed with the time-shifted version of itself. This results in a capacitor being charged, storing the locking voltage. As shown in Figure 7.1, the reference signal rises and settles to a certain voltage. At the same, the autozero capacitor is being charged, which stores the offset introduced by the OTA and the comparator. When the OTA reaches 0.55 V, the auto zero voltage is slightly lower than the reference voltage. After this phase, the DLL has achieved the settling condition for locking.

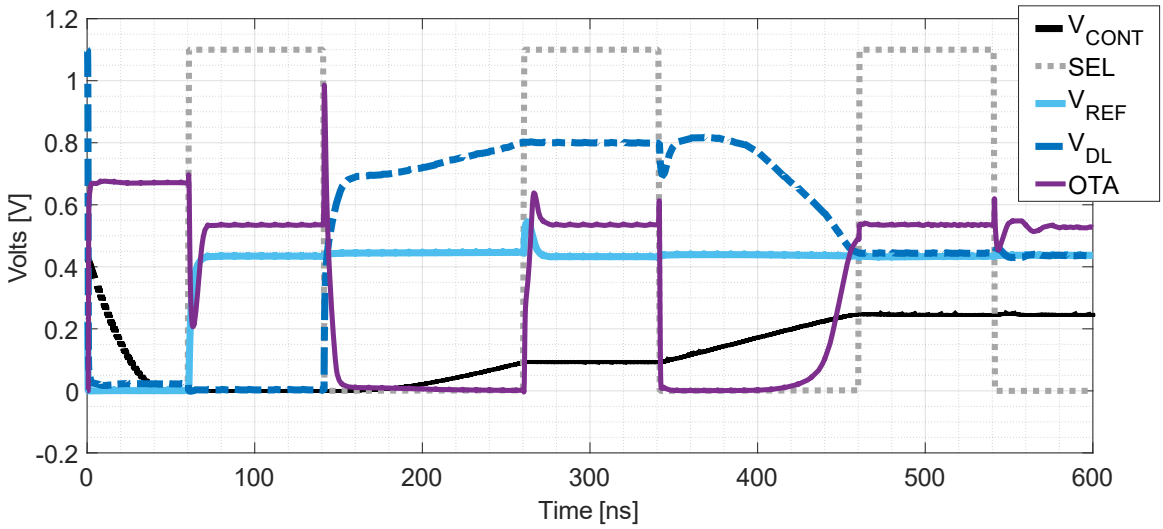


Figure 7.1: Settling process of the DLL

When the select signal goes low, the output of the delay line is mixed with the reference. Due to the initial offset being high, the voltage is a lot higher than the reference. The comparator goes low because the delay line's mixed voltage is higher, and this results in the charge pump charging the capacitor. Once the two voltages approach each other the comparator slowly goes upwards to 0.55 volts. The voltage slightly overshoots and settles, and both the DLL voltage and reference voltage are equal. Hence the two phases are equal.

The selection signal switches on and off multiple times. This is done because the CML delay line's amplitude changes as function of the control voltage, this amplitude variation changes the output swing of the mixer and introduces an offset. So first, a coarse calibration is done to get close to the target and then when the amplitude is at around the right point, it updates the calibration point.

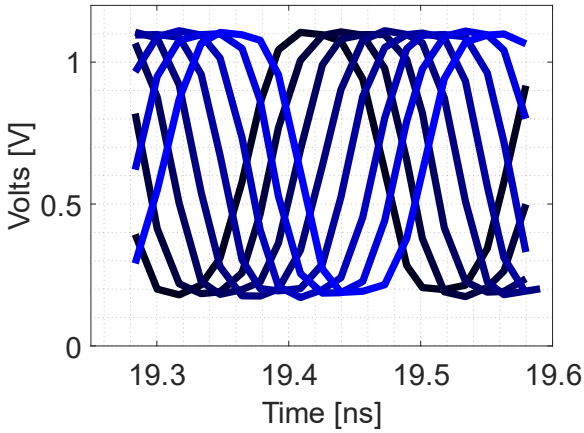


Figure 7.2: The DLL running at 5 GHz in an unlocked state

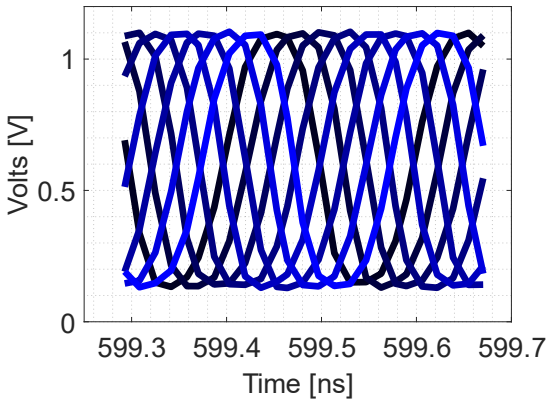


Figure 7.3: The DLL the locked condition at 5 GHz

In Figure 7.2 and Figure 7.3, the single ended output phases are shown going from the unlocked to locked state, respectively. In Figure 7.4, the reference and delay line signal are shown, where it is shown that the two signals are superimposed.

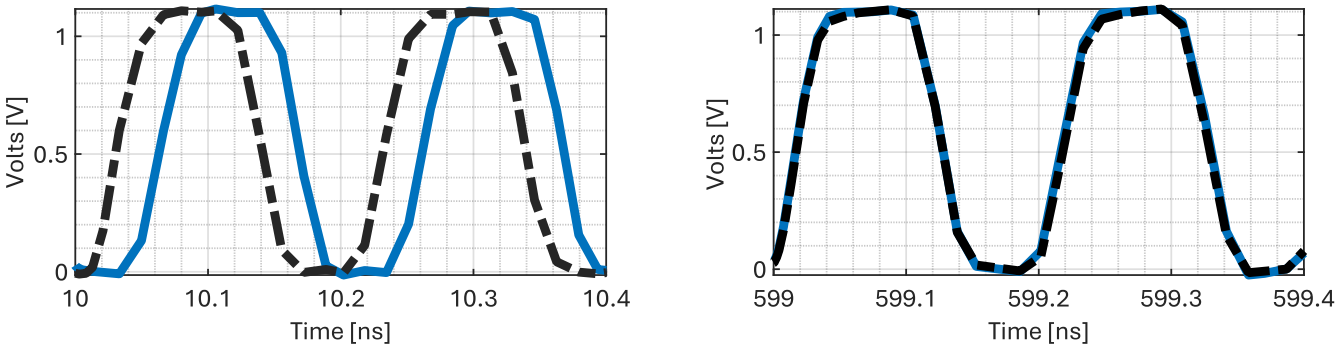


Figure 7.4: Phase detector going from unlocked to locked. The solid curve is the reference clock, the dotted line the delay line's output. After settling these two waveforms are locked

In Figure 7.2, the voltage swing does not fully swing between the rails because of the CML output not being full swing. The voltages that the phase detector see are full swing, due to the CML-2-CMOS converter, as shown in Figure 7.4. This aims to remove any AM mismatch between the reference signal and delayed signal, since the AM variation manifests into PM variation.

The power consumption of the phase-detector when locked remains at around 560 uW, so its contributions are quite negligible to the total power consumption.

7.2 Mismatch

By performing Montecarlo simulations the effect of the mismatch is analysed. This pertains to the phase detector's accuracy and the phases of the DLL. The simulations were first run at 10 GHz and then at 11GHz. In Figure 7.5, a bar plot shows the phase offset of the phase detector at 10 GHz. From which can be seen that a mean of 110 fs is obtained with a standard deviation of 302fs. The mismatch stems from the difference of the input pair from the transistors (see Figure 6.3). It was found that doubling the size of the input transistor results in $\sqrt{2}$ smaller phase mismatch of the phase detector which implies that all the mismatch is due to the input pair of the mixer. Increasing the size of the input pair too much will result into too much loading and hinders the DLL. This either slows the system down or requires larger current driving capabilities of the whole delay line.

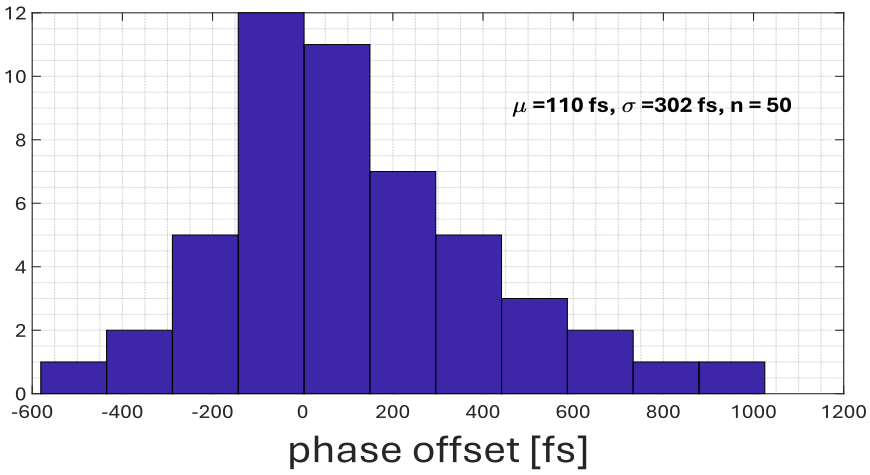


Figure 7.5: Phase difference between the reference and delayed signal in a locked condition.

In Figure 7.7, the standard deviation per phase is plotted. At 10 GHz, the standard deviation is 750 fs at worst, found at the last phase. This goes contrary to the expectation that a Brownian bridge process gives the phase accuracy of the phases, and that on average, the worst phase error should be seen in the middle. The source of this error seems to be due to the swing of the CML not being full swing. This results in amplitude variation manifesting into phase deviation, which results in an additional error. This results in the last delay element having a different statistic than the first element. Thus, that the assumption that delays have the same statistic is false and thus, the process is not described by a Brownian bridge model.

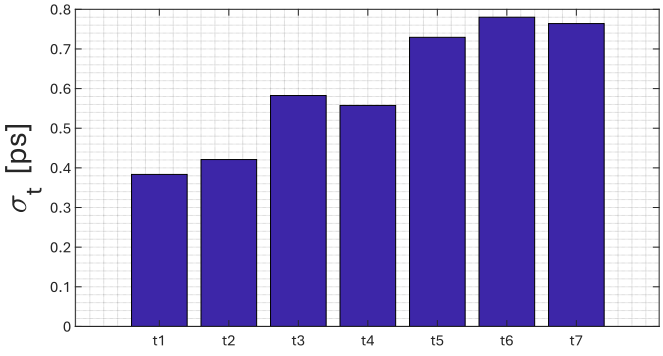


Figure 7.7: Phase offsets at 10 GHz

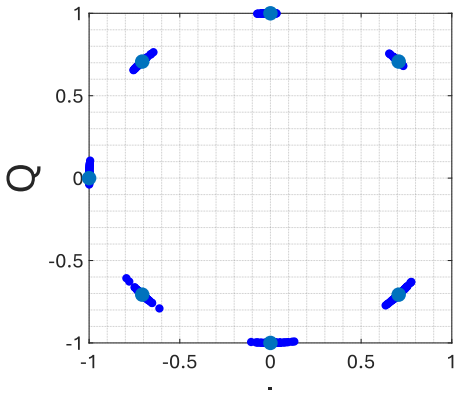


Figure 7.6:IQ plot if the DLL at 10 GHz from various Montecarlo runs

To further quantify the performance of the DLL's phase accuracy the EVM is calculated. The average EVM is 3.7%, with a standard deviation of 1.7%.

The circuit can operate up to 11.1 GHz. However, the yield of the circuit becomes worse. In Figure 7.9, the phase detector has roughly the same mean but has huge outliers. This stems from the DLL not being able to always lock due to some of the Montecarlo iterations failing. If we only consider the iterations within +-1ps, the yield is 68%. The standard deviation per phase remains about the same for the iterations withing this range. However, since the operational frequency increases, the phase mismatch (in degrees) has increased. In the Appendix, a couple of these trails are shown.

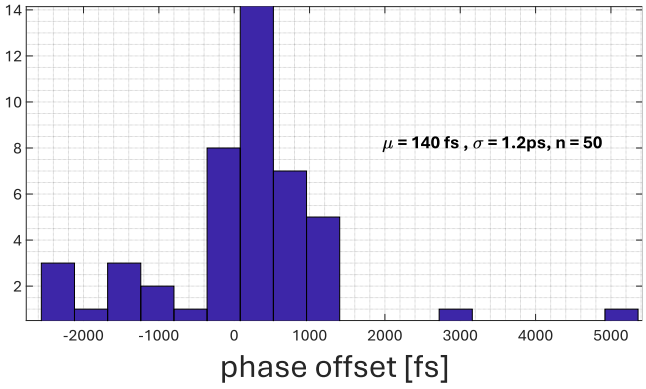


Figure 7.9: Phase offset at 11.1 GHz.

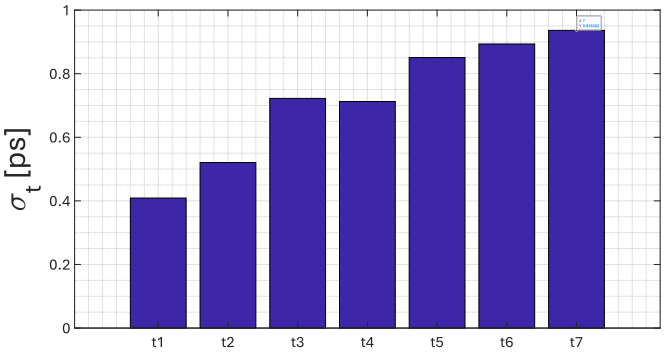


Figure 7.9: Phase offsets at 11.1 GHz.

7.3 Chip

The simulations were run using the extracted core seen in Figure 7.10. The chip's size is 950u by 950u and the DLL itself is 180u x 80u.

On the left side of the chip, the control signals are placed. A0, A1, A2 are used to control the multiplexer to select one of the eight phases. The CAL (calibration) pin is used to select between the calibration and acquisition phase of the phase detector. It was chosen to put this externally to see how the frequency of the calibration signal influences the performance of the DLL.

Two bias pins are used to bias the delay line and the output stage of the multiplexer. The inputs are open drain and will be biased externally by a current. This is done to be able to overdrive the current of the chip to see how fast the chip can be maximally operated ay.

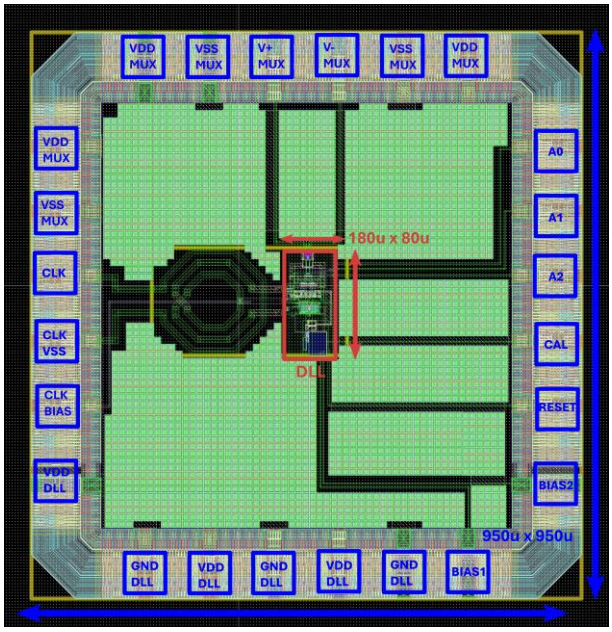


Figure 7.10: Chip layout

7.4 Comparison

	[13]	*[14]*	*[15]*	*[16]*	*[17]*	*This work*
Process	90nm	40nm	90nm	65nm	7nm	40nm
Supply	1.2 V	1.1 V	1.0 V	1.2	1.2	1.1 V
Architecture	Distributed DLL	conventional phase detector	Distributed DLL	Delta QDLL	QLL with ring-oscillator	Mixing phase detector
Frequency range[GHz]	2 – 8	2.0 – 7.7	8 – 10	3.5 – 11	4-16	5 – 11.1
Output phases	3	16	5	4	4	8
Power	28mW@8GHz	18.3@GHz 7.4	15mW @10GHz	7mW @7GHz	48mW @16GHz	26mW@11.1 GHz 19mW@8Ghz
Phase detector offset	-	292.9fs +- 17.9 fs	-	-		100fs +- 302fs
Maximum phase error[4]	1.53 ps@8Ghz 1.93ps@2GHz	0.58ps@ 7.4 GHz	1.3 ps	0.2ps	<1 degree	-
Max Standard deviation						0.8ps @10Ghz
Active Area	0.06444 mm^2	0.0168 mm^2	0.03 mm^2	0.012mm^2	0.105 mm^2	0.0144 mm^2

The chip's performance is summarised in the table above and compared with that of the prior art. It can be seen that the proposed mixing phase detector architecture allows for the most phases at the highest frequency. Special attention must go to the excellent work of [16]They also use a mixer for quadrature phase generation. This structure creates quadrature clocks with two parallel delay lines fed by the same clock. They mix the two paths until the output goes to zero. With this structure, they get very good quadrature clocks because, similarly to the proposed DLL, a low frequency, high loop gain feedback system can be created. In the structure of their work, they needed to have a coarse correction circuit to compensate for the narrow acquisition of the mixer, which was fixed in this thesis's work with the phase shift. The coarse correction circuit initially shifts the circuit close to where it must lock, and then the fine-tuning circuit locks it. Lastly, the circuit can only be used to create quadrature clocks, whereas the phase detector proposed in this thesis can create any number of phases over a wideband.

With the chip's simulated values, the results are very competitive with the state-of-the-art. It is possible to create 8 phases at higher frequencies than any *DLL* architecture so far.

[4] In the literature, they often use worst maximum error measured. However, this does not tell much about how the error is distributed along the line. Hence why the maximum standard deviation is given instead.

8 Future work and Conclusion

8.1 Conclusion

This thesis aimed to create an accurate clock generator that simultaneously generates eight clock phases. It aims to operate at more than 8 GHz such that a DTX can be realized in the newly allocated frequency band of upcoming 6G and Wi-Fi 7 standards. To create these phases, a delay locked loop has been selected and designed to be the clock generator for a DTX. This is because it generates the eight phases simultaneously from a single clock at the clock frequency, and it can create these phases over a wideband.

The delay lock loop has been designed which generates these eight phases at 11.1 GHz, this meets the requirements, but also, fares well with the current state of the art. This result was obtained due to the new proposed phase detector in the DLL, coined the *Mixing Phase Detector*.

Using a mixer as a phase detector, the fundamental limitation of the conventional phase detector is overcome, namely, operating at the clock's frequency. Using a mixer as phase detector, the frequency is down-converted to DC ,overcoming the hurdle of operating at the clock's frequency.

The phase detector has been designed and implemented on chip, from which the tentative results were evaluated, discussed, and used as the baseline to compare with state-of-the-art. At the time of publication, the chip has not been taped out; it will be taped out shortly. With the taped-out chip, the measurements will have to prove if this architecture holds up to the expectations.

8.2 Future work

The DLL's next step is to be implemented in a DTX, with the goal to create the multiphase DTX at high frequencies.

Another aspect of the design of the DLL can be improved. In this thesis, the operating frequency was decoupled from the phase detector, relaxing the constraint of the phase-detector circuitry because it now operates at a low frequency. Another decoupling would be the delay element's power and phase accuracy. Currently, the phase accuracy of the delay elements is open-loop and can only be fixed by sizing the transistors to reduce the mismatch. However, a high-power penalty is paid for higher phase accuracy at these frequencies. By employing an additional servo loop, it could be possible to lock the delay elements "delay" to each other. Then, when these phases are equal, the mixing phase detector will lock the loop. This would lower the power consumption and improve the phase accuracy.

Bibliography

[1] F. H. Raab, “Intermodulation Distortion in Kahn-Technique Transmitters,” 1996.

[2] J. Zhuang, K. Waheed, and R. B. Staszewski, “A technique to reduce phase/frequency modulation bandwidth in a polar RF transmitter,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 57, no. 8, pp. 2196–2207, 2010, doi: 10.1109/TCSI.2009.2037394.

[3] S. M. Yoo *et al.*, “A class-G switched-capacitor RF power amplifier,” *IEEE J Solid-State Circuits*, vol. 48, no. 5, pp. 1212–1224, May 2013, doi: 10.1109/JSSC.2013.2252754.

[4] M. Beikmirza, Y. Shen, L. C. N. De Vreede, and M. S. Alavi, “A Wideband Energy-Efficient Multi-Mode CMOS Digital Transmitter,” *IEEE J Solid-State Circuits*, vol. 58, no. 3, pp. 677–690, Mar. 2023, doi: 10.1109/JSSC.2022.3222028.

[5] Q. Zhu *et al.*, “A Digital Polar Transmitter with DC-DC Converter Supporting 256-QAM WLAN and 40-MHz LTE-A Carrier Aggregation,” *IEEE J Solid-State Circuits*, vol. 52, no. 5, pp. 1196–1209, May 2017, doi: 10.1109/JSSC.2017.2672979.

[6] M. Hashemi, Y. Shen, M. Mehrpoo, M. S. Alavi, and L. C. N. De Vreede, “An Intrinsically Linear Wideband Polar Digital Power Amplifier,” *IEEE J Solid-State Circuits*, vol. 52, no. 12, pp. 3312–3328, Dec. 2017, doi: 10.1109/JSSC.2017.2737647.

[7] Y. Shen, R. Bootsman, M. S. Alavi, and L. C. N. de Vreede, “A 1–3 GHz I/Q Interleaved Direct-Digital RF Modulator As A Driver for A Common-Gate PA in 40 nm CMOS,” in *2020 IEEE Radio Frequency Integrated Circuits Symposium (RFIC)*, IEEE, Aug. 2020, pp. 287–290. doi: 10.1109/RFIC49505.2020.9218324.

[8] M. Hashemi *et al.*, “17.5 An intrinsically linear wideband digital polar PA featuring AM-AM and AM-PM corrections through nonlinear sizing, overdrive-voltage control, and multiphase RF clocking,” in *2017 IEEE International Solid-State Circuits Conference (ISSCC)*, IEEE, Feb. 2017, pp. 300–301. doi: 10.1109/ISSCC.2017.7870380.

[9] X. Xiang, W. Gai, L. Shi, A. He, and K. Sheng, “An 8–12GHz 0.92° Phase Error Quadrature Clock Generator Based on Two-Stage Poly Phase Filter with Intermediate Point Compensation,” in *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*, IEEE, May 2019, pp. 1–5. doi: 10.1109/ISCAS.2019.8702803.

[10] B. Razavi, “The Delay-Locked Loop [A Circuit for All Seasons],” *IEEE Solid-State Circuits Magazine*, vol. 10, no. 3, pp. 9–15, Jun. 2018, doi: 10.1109/MSSC.2018.2844615.

[11] K. J. Hsiao and T. C. Lee, “An 8-GHz to 10-GHz distributed DLL for multiphase clock generation,” *IEEE J Solid-State Circuits*, vol. 44, no. 9, pp. 2478–2487, Sep. 2009, doi: 10.1109/JSSC.2009.2024804.

[12] J. Yang, Q. Pan, J. Yin, and P. I. Mak, “A 2.0-to-7.4-GHz 16-Phase Delay-Locked Loop With a Sub-0.6-ps Phase-Delay Error in 40-nm CMOS,” *IEEE Trans Microw Theory Tech*, vol. 71, no. 8, pp. 3596–3604, Aug. 2023, doi: 10.1109/TMTT.2023.3242333.

[13] M.-H. Hsieh, B.-F. Lin, Y.-S. Wang, H.-H. Chang, and C. C.-P. Chen, “A 2 – 8 GHz multi-phase distributed DLL using phase insertion in 90 nm,” in *2012 IEEE International Symposium on Circuits and Systems*, IEEE, May 2012, pp. 2015–2018. doi: 10.1109/ISCAS.2012.6271675.

[14] J. Yang, Q. Pan, J. Yin, and P. I. Mak, “A 2.0-to-7.4-GHz 16-Phase Delay-Locked Loop With a Sub-0.6-ps Phase-Delay Error in 40-nm CMOS,” *IEEE Trans Microw Theory Tech*, vol. 71, no. 8, pp. 3596–3604, Aug. 2023, doi: 10.1109/TMTT.2023.3242333.

[15] K. J. Hsiao and T. C. Lee, “An 8-GHz to 10-GHz distributed DLL for multiphase clock generation,” *IEEE J Solid-State Circuits*, vol. 44, no. 9, pp. 2478–2487, Sep. 2009, doi: 10.1109/JSSC.2009.2024804.

[16] Z. Wang and P. R. Kinget, “A Very High Linearity Twin Phase Interpolator With a Low-Noise and Wideband Delta Quadrature DLL for High-Speed Data Link Clocking,” *IEEE J Solid-State Circuits*, vol. 58, no. 4, pp. 1172–1184, Apr. 2023, doi: 10.1109/JSSC.2022.3197061.

[17] S. Chen *et al.*, “A 4-to-16GHz inverter-based injection-locked quadrature clock generator with phase interpolators for multi-standard I/Os in 7nm FinFET,” in *2018 IEEE International Solid - State Circuits Conference - (ISSCC)*, IEEE, Feb. 2018, pp. 390–392. doi: 10.1109/ISSCC.2018.8310348.

Appendix

Some of the Montecarlo runs at 11.1 GHz.

