Design and Modelling of Clock and Data Recovery Integrated Circuit in 130 nm CMOS Technology for 10 Gb/s Serial Data Communications

A THESIS SUBMITTED TO

THE DEPARTMENT OF ELECTRONICS AND ELECTRICAL

ENGINEERING

FACULTY OF ENGINEERING

UNIVERSITY OF GLASGOW

IN FULFILMENT OF THE REQUIREMENTS

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

By

Maher Assaad

January 2009

In Memory of my father Mohammad

Who passed away in January 2004

Abstract

This thesis describes the design and implementation of a fully monolithic 10 Gb/s phase and frequency-locked loop based clock and data recovery (PFLL-CDR) integrated circuit, as well as the Verilog-A modelling of an asynchronous serial link based chip to chip communication system incorporating the proposed concept. The proposed design was implemented and fabricated using the 130 nm CMOS technology offered by UMC (United Microelectronics Corporation). Different PLL-based CDR circuits topologies were investigated in terms of architecture and speed. Based on the investigation, we proposed a new concept of quarter-rate (i.e. the clocking speed in the circuit is 2.5 GHz for 10 Gb/s data rate) and dual-loop topology which consists of phase-locked and frequency-locked loop. The frequency-locked loop (FLL) operates independently from the phase-locked loop (PLL), and has a highly-desired feature that once the proper frequency has been acquired, the FLL is automatically disabled and the PLL will take over to adjust the clock edges approximately in the middle of the incoming data bits for proper sampling. Another important feature of the proposed quarter-rate concept is the inherent 1-to-4 demultiplexing of the input serial data stream. A new quarter-rate phase detector based on the non-linear early-late phase detector concept has been used to achieve the multi-Giga bit/s speed and to eliminate the need of the front-end data pre-processing (edge detecting) units usually associated with the conventional CDR circuits. An eight-stage differential ring oscillator running at 2.5 GHz frequency centre was used for the voltage-controlled oscillator (VCO) to generate low-jitter multi-phase clock signals. The transistor level simulation results demonstrated excellent performances in term of locking speed and power consumption. In order to verify the accuracy of the proposed quarter-rate concept, a clockless asynchronous serial link incorporating the proposed concept and communicating two chips at 10 Gb/s has been modelled at gate level using the Verilog-A language and time-domain simulated.

Publications

Conference Contributions

1. M.ASSAAD and D. R. S. Cumming, "CMOS IC Design and Verilog-A Modeling of 10-Gb/s PLL-Based Deserializer for Inter-Chip Communication in SOC.", international symposium on system on chip 2007, Nov. 2007.

2. M. Assaad and D. R. S. Cumming, "20 Gb/s Referenceless Quarter-Rate PLL-Based Clock Data Recovery Circuit in 130 nm CMOS Technology", 15th International Conference on Mixed Design of Integrated Circuits and Systems. MIXDES 2008. pp. 147–150, 2008.

Acknowledgments

I am grateful to many people who made this work possible. First of all, I would like to deeply express my great gratitude for Professor David R. S. Cumming, my PhD supervisor, for his support throughout this work. I am very grateful to him especially for the ideal opportunity that he gave me in joining the Microsystem Technology group, offering me a 3-years fully funded studentship and the freedom of choosing my own research subject, I am also grateful to him for his constant encouragement to complete my PhD work.

I would like to thank Dr. Mark Milgrew for his CAD tools help, Billy Allan for his computer support, Douglas Iron, Karen Phillips, Alexander Ross and Stuart Fairbairn.

I would like to deeply thank my ex-wife Lucie St-Laurent for her endless listening and encouragement even when she is ill and still suffering from her cancer. I would like to thank my son Shady for the wonderful time I spent with him in Glasgow and his patience and understanding for leaving him at home for long hours while I am working in the office and his mother Lucie in Montreal to continue fighting against her cancer with the painful radiotherapy and chemotherapy. I would like to deeply thank my mother Fatima Harfoush for her continuous moral support and encouragement in my private life and to complete my PhD work.

Finally, I would like to thank my little princess and future wife Dima Elkhadem for her early support and encouragements.

I am frankly considering myself so lucky having all above great people around me during my PhD study at the University of Glasgow.

January 5th 2009

Contents

List of Figures

1 Introduction

1.1 Background and Motivation

Due to continuing progress in integrated circuit technology, system-on-chip (SOC) is becoming larger requiring many long on-chip wires to connect modules. However it is becoming increasingly hard to communicate synchronous data between high speed modules. To take advantage of the increased processing speed available and to improve the overall system performance requires high-speed inter-chip communication networks. Higher I/O bandwidth requirement has led to the use of point-to-point serial links. As well as increasing the I/O bandwidth these links can lower resource costs such as power and area, and reduce the impact of problems associated with inter-chip communication such as skew and crosstalk. The multi-bit parallel bus and the source synchronous point-to-point parallel link have been widely used in short-distance applications such as multiprocessor interconnections. However, in a high performance SOC, a long parallel link suffers from several problems. An asynchronous serial link is one solution that can overcome such problems since it occupies less area owing to having fewer communication wires. A dedicated point-to-point asynchronous serial link is shown in Figure 1-1(b).
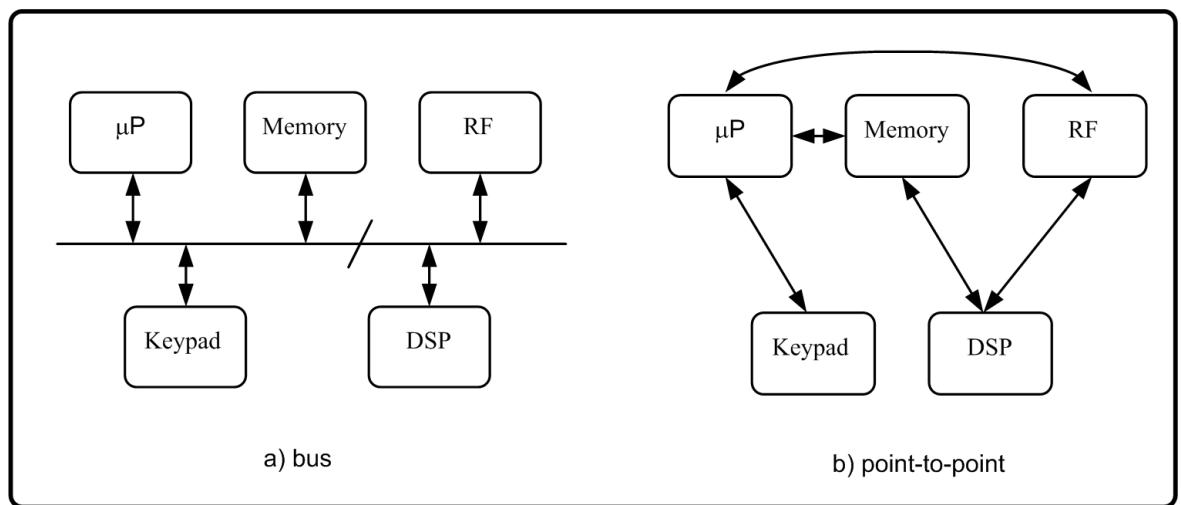
Figure 1-1: Example of communication in system on chip, (a) traditional bus-based communication and, (b) dedicated point-to-point links.

Serial links have been widely used for long-haul fibre optic and cable based communication medium (e.g. WAN, MAN and LAN) and in some computer networks, where the cable cost and synchronization difficulties make parallel communication impractical. Serial links have recently found a greater number of applications in consumer electronics, such as USB (Universal Serial Bus) that connects peripheral electronic systems to computer, and SATA (Serial Advanced Technology Attachment) which communicates the computer motherboard with mass storage devices (e.g. hard disk) and PCI-Express (Peripheral Component Interconnect) normally connect cards (sound, video or other) to the motherboard. Therefore serial communication has become the solution to higher and more efficient data transmission in order to meet the demands and trends of the higher capacity of communication technology. A relatively recent analytical study has been conducted by R. Dobkin [81] in which comparing in term of power and area serial to parallel links that have been implemented in various feature size of CMOS technologies. The result of that study is illustrated in Figure 1-2 and provides the following important remarks:

1. For any particular feature size of the CMOS technology, there is a limiting value of the link length above which, it is better to implement the link as serial rather than parallel because it is more advantageous in term of power and area.

2. The limiting value discussed in 1 which defines the frontiers between the two types of the link implementations is scaling down as the relative scaling down of the CMOS technology feature size.
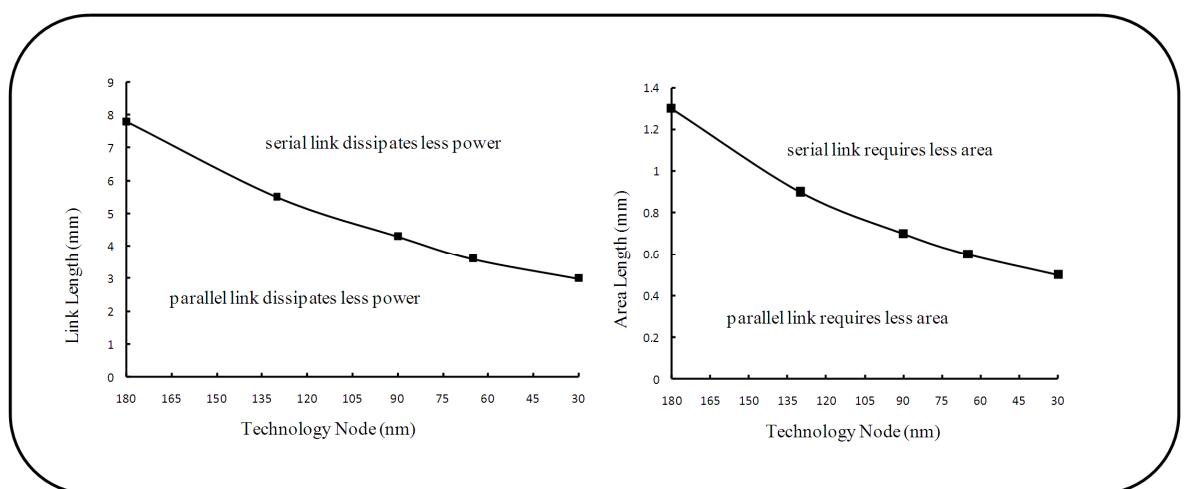
Figure 1-2: Area and power for serial and parallel links versus technology node [81].

Therefore, for a particular CMOS technology feature size and link length, a serial link may have the following advantages over the parallel one:

1. A serial link generally occupies less area; hence the communication and area cost is reduced due to decreased number of pins and occupied area. The saved area can be used to isolate the link better from its surrounding components and to integrate more units.

2. The presence of multiple conductors in parallel and close proximity as in bus and point-to-point parallel links implies cross-talk and especially at higher frequency. In a serial link the undesired cross-talk is minimized.

3. The skew between the clock and data signals normally occurs in bus and point-to-point parallel links is irrelevant in a serial link, because the transferring of data is carried out without a clock signal.

4. A serial link can provides reliable intra/inter chip data communication at multi Gb/s rate.

1.2 Research Objectives and Summary of Contributions

The processing speed of chips in a PCB (Printed Circuit board), or modules within an SOC is normally higher than the speed at which those units normally communicate. In this thesis we attempt to make the communication speed (e.g. 10 Gb/s) few order of magnitude higher than the processing speed of units (e.g. 1.25 Gb/s) themselves by using a SERDES based serial link. The contributions of this thesis can be summarized as follows.

- A referenceless quarter-rate PLL-based clock and data recovery has been proposed in which the deserializer does not need a clock reference, the deserializer is clocked at quarter-rate (2.5 GHz) of the incoming data rate (10 Gb/s) and the input data stream is 1-to-4 automatically demultiplexed for further processing.

- In order to verify the accuracy of the proposed concept, a 10 Gb/s serial link based chip-to-chip communication medium incorporating the proposed concept has been implemented using the Verilog-A language and simulated in Cadence.

1.3 Organization of the Thesis

The reminder of the thesis is divided into six chapters.

1.3.1 Chapter 2

In this chapter we first present the limitations and problems associated with the use of the traditional multi-bit parallel bus and point-to-point parallel link as communication mediums, and second we present a review of the literature relevant to the design of different architectures of clock and data recovery circuits.

1.3.2 Chapter 3

The PLL theory will be presented in this chapter and analytical expressions will be developed. The resulting equations will relate the PLL parameters such as stability and bandwidth to the low pass filter components values.

1.3.3 Chapter 4

This chapter will focus on the current-mode logic transistor level design and optimization at 10 Gb/s of the different parts of the proposed concept Those parts are the voltage controlled oscillator, the proposed quarter-rate phase detector and proposed quarter-rate frequency detector.

1.3.4 Chapter 5

Once all the circuits are designed and optimized at transistor level, their parameters (i.e. delay, rise and fall times) will be extracted and implemented in their correspondent Verilog-A description. This chapter will be dedicated to implement a complete 10 Gb/s serial link in Verilog-A language using the proposed concept.

1.3.5 Chapter 6

This chapter will concentrate on the layout implementation, post-layout transistor level simulations and characterization of the proposed concept of quarter-rate clock and data recovery circuit as well as its comprising blocks.

1.3.6 Chapter 7

This chapter draws conclusions and offers some suggestions for future works.

2 Introduction

This chapter contains a review of literature describing the problems associated with the use of traditional multi line parallel busses as a communication medium in today system-on-chip (SOC). One solution that has been proposed is the point-to-point source synchronous parallel link that is briefly described here. An alternative approach that is proposed in this thesis is clockless serial link. It has the potential to be a high-speed, low cost, and skew insensitive solution to the problems of communication in SOC based upon a shared bus.

2.1 Conventional Bus Limitations

Interconnects in a SOC have followed the bus paradigm. In a bus-based system, as illustrated in Figure 2-1, the intellectual properties (IP)[1] are interconnected through a set of parallel wires. A separate wire is distributed to all IP's carrying a global clock signal used for synchronous transmission and reception of data. As in a digital system, improving SOC performance requires enhancing the IP's processing speed and increasing the bandwidth of the interconnects.
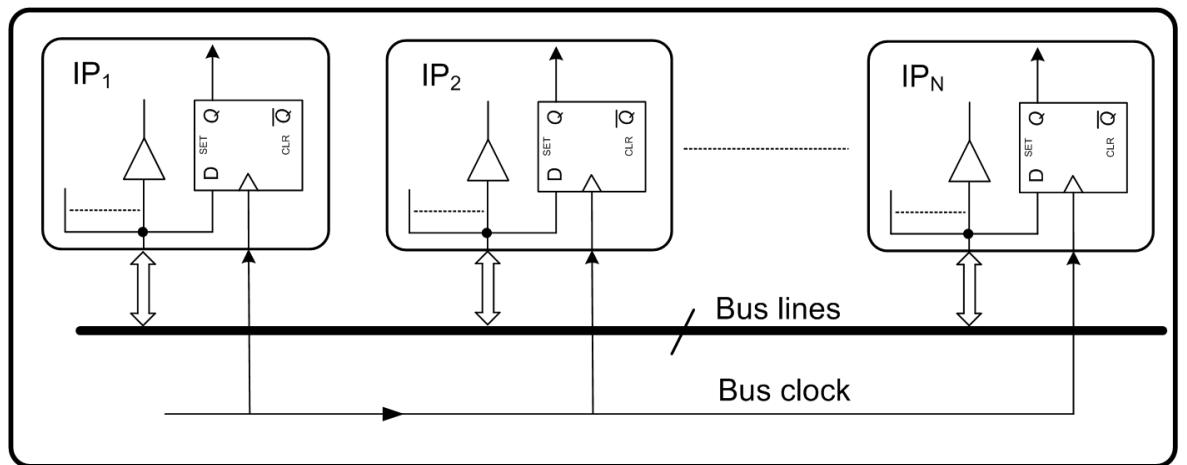
Figure 2-1: SOC based upon a shared bus.

Advances in Integrated Circuit (IC) fabrication technology have led to an exponential growth of IC speed and integration level [1]. However, in a multi-IP based SOC, the bus becomes a communication bottleneck. As more processing units are added to it, the energy

---

[1] IP is a creation of the mind with a commercial value, the holder of the IP has exclusive right to it.

dissipation per binary transition grows and the overall system speed is reduced due to the increased number of attached units leading to higher capacitive load. As shown in Figure 2-2, the multi-bit bus also has other problems such as skew[2], crosstalk[3] and large area [2]. Since the data signal carried by the bus must be synchronized with the global clock signal, skew has become a primary limit on increasing the operational frequency. Moreover, the crosstalk between adjacent bus lines causes data signal delay and noise and hence makes the on-chip communication unreliable. The cost of using a bus is also a serious issue since they occupy a large area of silicon. Therefore the use of multi-bit buses for on-chip communication, with a global clock, will limit further improvement of future SOC.
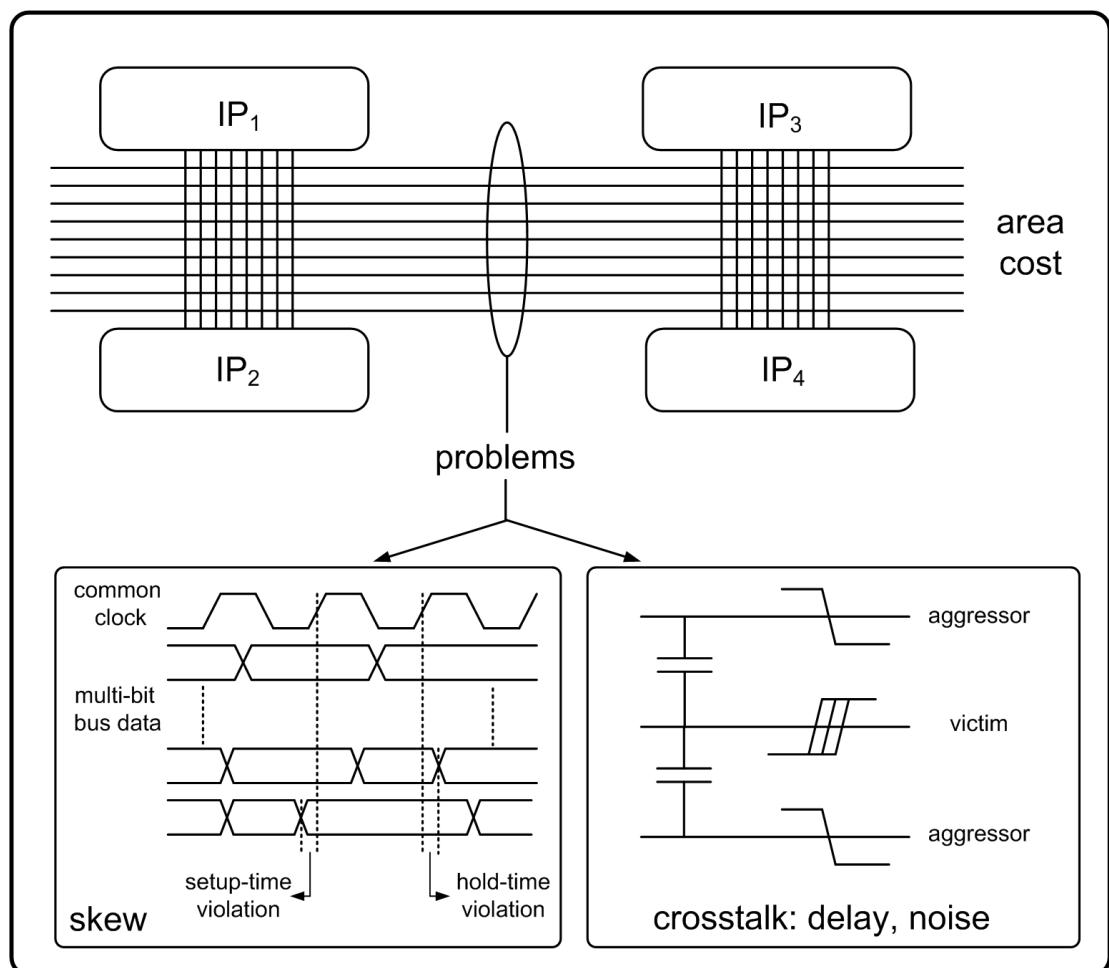
Figure 2-2: Problems associated with multi-bit shared bus in SOC.

---

[2] Skew is defined as the difference in arrival time of bits transmitted at the same time.

[3] Crosstalk refers to the undesired effect created by the transmission of a signal on one channel in another channel.

2.2 Point-to-Point Links

The physical and electrical constraints of busses make them viable for only small scale systems that incorporate few IP's, such as memory or peripheral busses. For larger scale systems such as multi-processors or communication switches an alternative and attractive solution is to replace the bus by a point to point link as a medium of communication. This approach has advantages from both circuit and architectural points of view. From a circuit design perspective, a point-to-point link has a higher communication bandwidth than a bus, due to its reduced signal integrity problems. Moreover, a point-to-point transmission line offers greater flexibility in the physical construction of the system. From an architectural perspective, the bandwidth demands of high-speed systems make the shared bus medium the main performance bottleneck. For this reason, the hierarchical bus has been gradually replacing single busses as a medium of communication in high performance multi-IP SOC [3], while the architecture of most high performance communication switches is based on point-to-point interconnection [4, 5].

2.3 The Key Elements of a Link

There are three key components in a link: the transmitter, the channel and the receiver. The transmitter converts the digital data stream into an analog signal; the channel is the transmission medium in which the signal is travelling; and the receiver converts the analog received signal back to a digital data sequence. Figure 2-3 illustrates the block diagram of a typical link and its primary components.

The transmitter comprises an encoder and a modulator, while the receiver contains a demodulator and a decoder. Generally, the bit sequence is first encoded, by inserting some redundant bits to guarantee signal transition and ease the timing recovery operation. But, in this work, the data is not coded and sent directly on the channel using a simple non-return-to-zero (NRZ) format, and the signal levels (high and low) are represented by two different electrical voltages.
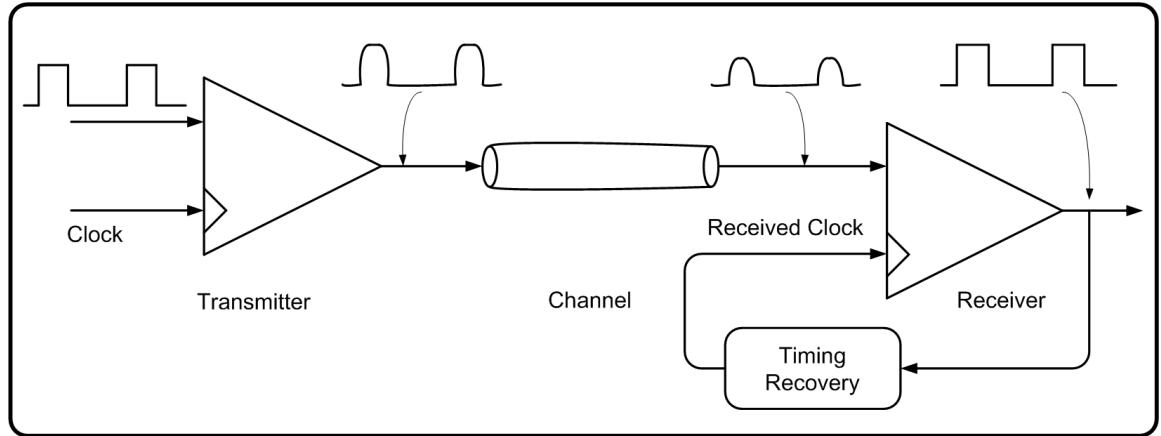
Figure 2-3: A basic link with its three components: transmitter, channel, and receiver.

The conversion of a discrete-time data sequence into a continuous-time analog signal is called modulation. The transmitted signal is binary, and is synchronized to the transmitted clock. The smallest duration between any two successive edges of the signal is called the bit time. Moreover, in order to reduce the power consumption associated with the signaling, low voltage logic swing, such as that used in current-mode logic (CML) is used for the transmitted signal. The channel is a cable or fiber optic based link and is the physical medium that carries the signal from the transmitter output to the receiver input. The channel generally filters the transmitted signal and causes frequency-dependant channel attenuation and signal distortion, leading to reduced received signal amplitude and inter-symbol[4] interference (ISI), i.e. a symbol is distorted by noise introduced by earlier symbols or by the reflections of earlier symbols due to termination mismatch or impedance discontinuities in the channel. Channel attenuation and ISI are present in all links, but their magnitudes depend on the characteristics of the channel and the signal frequencies relative to the channel bandwidth. The receiver recovers the data stream from the received analog signal. The conversion operation from the continuous-time analog signal back to the original discrete-time digital signal is called demodulation. Another important task of the receiver is to amplify and sample the received signal using a timing recovery or clock recovery circuit. This circuit automatically adjusts the edges of extracted clock in the middle of the bits to properly sample it.

---

[4] A symbol in digital communication is the smallest number of data bits transmitted at one time, it could be one bit (i.e. 0, or 1), or few bits transmitted simultaneously resulting in symbol rate.

2.4 Point-to-Point Parallel versus Serial Link

Point-to-point link architecture can be divided into two classes, namely serial links and parallel links. In a serial link, the clock is embedded in the data stream and has to be extracted in the receiver from the stream itself using a clock recovery circuit, while in a parallel link an explicit clock signal is transmitted separately from the data signal over a single interconnect. Figure 2-4 shows a conventional source-synchronous point-to-point parallel link. Transmission of all data signals and the reference clock signal is triggered synchronously by the transmitted clock. Point-to-point parallel link have been widely used in short-distance applications such as multi-microprocessor interconnection [6-10] and consumer products with extensive multimedia applications [11, 12]. Improving the bandwidth of point-to-point parallel links is achieved by increasing the bit rate per pin and integrating a large number of pins into the system. The link architecture shown in Figure 2-3 is a serial link. Parallel on-chip data streams are serialized into one data sequence. As described earlier the receiver uses the signal transitions to recover the embedded clock and eventually align its local clock edges accordingly for optimal data detection.
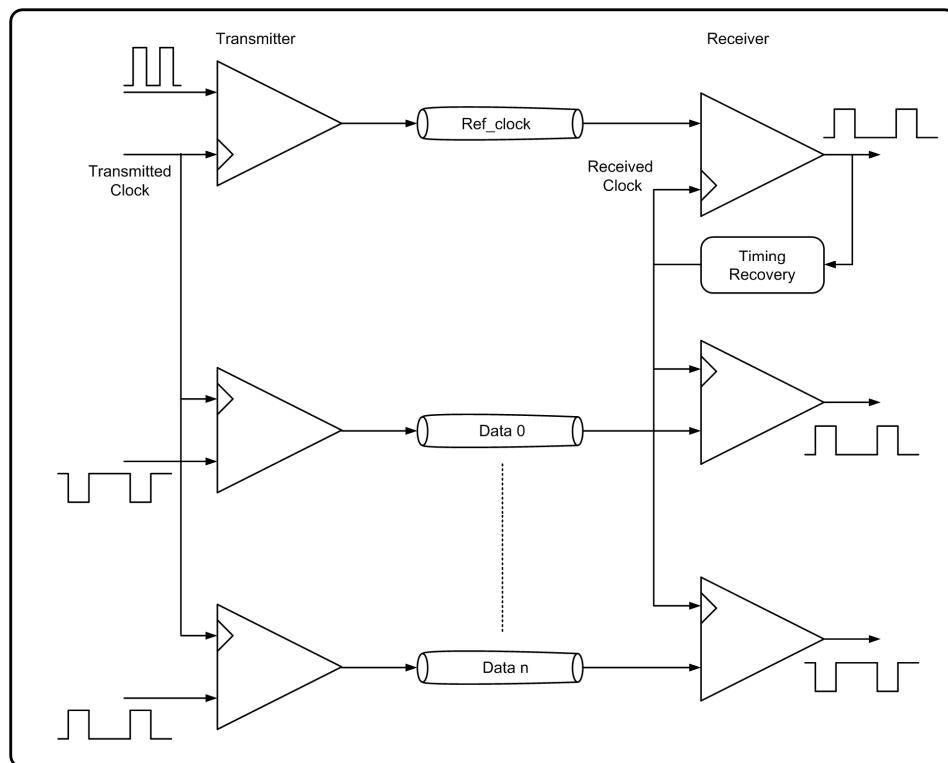
Figure 2-4: Source-synchronous parallel link, the clock is sent along for timing recovery.

Serial links are the design of choice in any application where the cost of communication channels is high and duplicating the links in large number is uneconomical. Its application spans every sector, including short and long distance communication and the networking markets [13-16]. The principal design goal of serial links is to maximize the data rate across the link and to extend the transmission range. Although, serial links requires serializer and deserializer circuits, but they are more advantageous over parallel links because they occupy less area and they are inherently insensitive to delay and skew.

2.5 Point-to-Point Serial Link Block Diagram

Exchanging high speed serial data involves three primary components as previously described: transmitter, channel and receiver. A transmitter gathers low rate parallel data and serializes it into high speed serial data. The signal is then transported through the channel to the receiver. The receiver must then demodulate the signal, extract the clock and demultiplex the data. The received information is fed out of the receiver as low speed parallel data for further processing as illustrated in Figure 2-5.
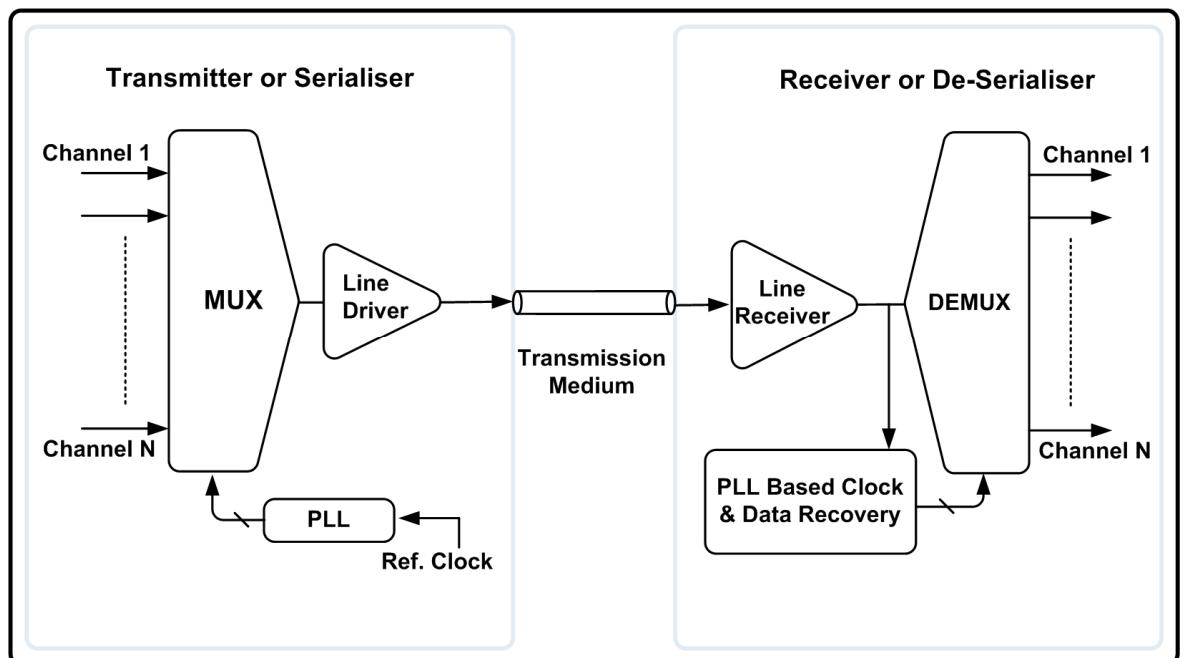
Figure 2-5: Simplified top level block diagram of a serial link.

2.5.1 Serializer or Transmitter

The transmitter's role is to accept several parallel data streams with a specified rate and then serialize and drive the data into the channel. As an example, a 10 Gb/s serializer would require eight parallel streams of 1.25 Gb/s each. Serializing involves multiplexing the data into an ordered bit stream using a NRZ format.

Driving the channel requires adding a 50 Ω output load amplifier, or in certain cases may require adding a sophisticated circuit that is capable of driving an optical driver. In most communication systems, the data is first encoded. The encoding process may include compression, encryption, error checking and framing [17]. Another important role of the encoder is to introduce additional transitions to the data stream to help a phase-locked loop (PLL) in the receiver acquire the correct clock frequency of the transmitter. The 8B/10B encoding scheme is the most popular and it guarantees at least one transition every 5 bits [18]. A PLL in the transmitter clocks the multiplexer and the multiplexer then performs the serialization function. Multiple clock frequencies are needed in order to properly perform the multiplexing operation. The PLL in the transmitter is responsible for generating the multiple clock frequencies, often known as the frequency synthesizer or the clock multiplier unit. The frequency synthesizer is required to have low phase noise and jitter to generate a similarly low phase noise data stream. The PLL locks the phase of an internal high speed clock to an externally supplied low speed reference. For example, a 10 Gb/s system may have a 156.25 MHz reference clock, and a 10 GHz internal clock. The PLL must then compare and match the two frequencies after dividing the internal clock by 64. The multiplexer is generally unable to drive the transmission medium directly, so a line driver is needed [19, 20]. The line driver matches the internal circuit impedance to the transmission line impedance and amplifies the signal to a suitable voltage swing. An important figure of merit of the transmitter is the output data jitter. The internal voltage-controlled oscillator (VCO), the multiplexer and all other circuits create and add jitter to signal. The VCO jitter is normally partially filtered out by the PLL.

2.5.2 Transport Channel

The channel carries the data signal from the transmitter to the receiver and could be electrical, optical or a combination of both. For long-haul communications the channel is a dominant source of phase noise and jitter. However for short-distance communications, the channel is considered as a negligible source of noise and jitter.

2.5.3 Deserializer or Receiver

The receiver must extract a clock from a noisy and jittered high frequency signal, and the extracted clock is then used to sample the received data stream. This process is called clock and data recovery (CDR) and it is difficult because the extraction process is based on the data signal transitions, the presence of which is not guaranteed. A line amplifier with a 50 Ω input impedance amplifies the signal to a suitable level for internal circuits while minimizing the distortion. Noise injection from this amplifier must be minimized because the received data signal is already saturated with jitter coming from the transport channel.

If the data is of the NRZ type, then the PD must also be able to handle random data that has random transition locations. Moreover, the key parameters of the PLL must be tuned to a signal with high noise content as compared to the PLL in the transmitter which has a low noise reference at its input. Additional circuits are needed to sample the data using the recovered clock unless the PD does so automatically. In some cases, a low frequency reference clock may be used to bring the frequency of the receiver's VCO close to the data rate before clock extraction occurs.

The architecture with a reference clock enhances the operation range of the receiver's PLL. Its drawback is that two separate PD's are needed and a circuit that can switch between them is necessary. This introduces two loops sharing common components which must be able to operate independently. A common component in a dual loop PLL is a lock detector circuit that determines if phase lock is lost in the data loop. If lock is lost the loop switches back to the external reference loop.

The dual loop architecture is useful in a high noise environment where the data jitter can cause the PLL to become unstable. Once the clock is extracted from the serial signal, the data can then be demultiplexed through a series of multiplexers at decreasing clock rates. For example, in a 10 Gb/s system the first re-sampled data would pass through a 1-to-2 demultiplexer driven by a 5 GHz clock. The second stage would consist of two 1-to-2 demultiplexers driven by a 2.5 GHz clock, and so on. If a multiphase clock is used, then multiple samples can be taken with separate samplers. This allows the use of a clock at a fraction of the data bit rate, hence reducing the power consumption associated with clock switching.

2.6 CDR Based Serial Link Applications

Much of this work focuses on the design of circuits and architecture development that will eventually leads to the implementation of a 10 Gb/s intra-chip and inter-chip high-speed interconnections in system-on-chip (SOC). The architectures and circuits presented here have a wider applicability to any high-speed communication system; such applications include the following [21]:

- LANs (local area networks), for broadband data communication links between computers over optical fibers such as Fiber-Distributed Data Interface (FDDI).

- WANs (Wide Area Networks) for multimedia applications.

- High-speed read/write channels for magnetic data-storage devices.

- High-speed serial data communication on metallic transmission media, such as coaxial cables and twisted pairs.

- Fiber optic receivers for long-haul optical communication networks.

2.7 CDR Principle and Architectures

A figure of merit in data signal detection process in the presence of noise is called the signal-to-noise ratio (SNR); the SNR is dependent on the location of the sampling instance. If the sampling point or instant is synchronized such that the peak value of the bit pulse is sensed, then the value of the SNR factor is maximal as illustrated in Figure 2-6.
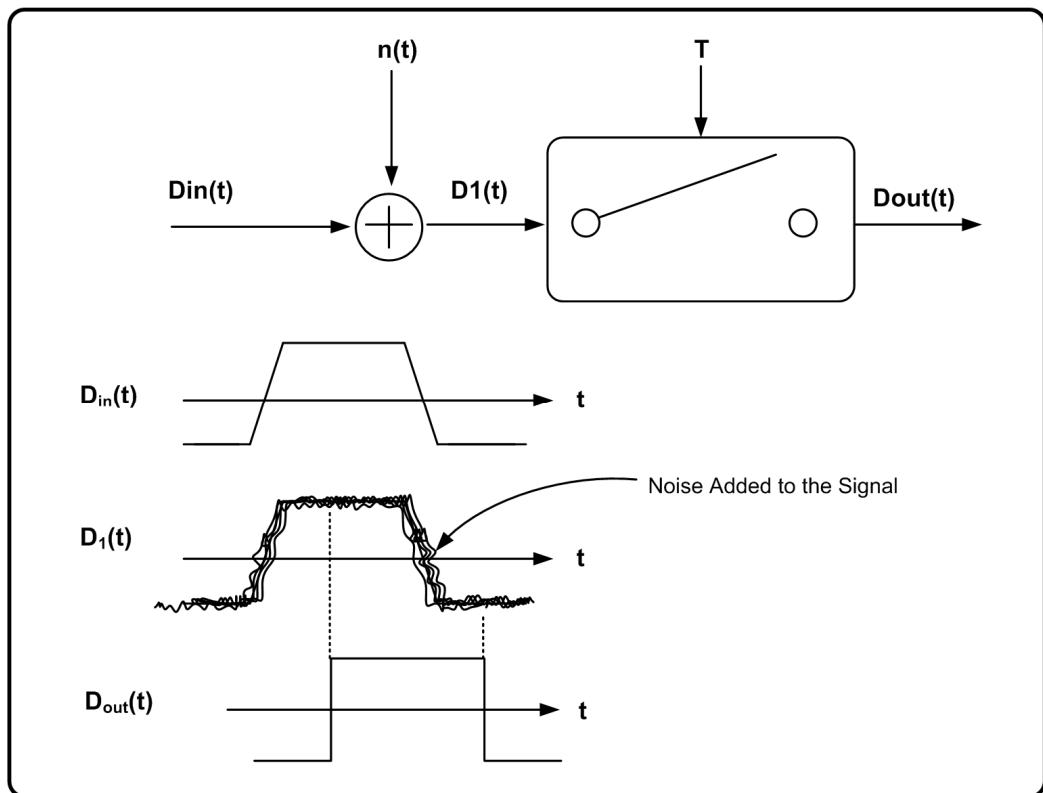
Figure 2-6: Detector with peak value sampling.

Synchronous sampling requires two conditions to be simultaneously satisfied. First, the frequency of the generated sampling clock signal has to be equal to the data rate. Second, the clock signal is sampling the data at its peak point. Satisfaction of these two conditions is commonly referred to as the process of clock and data recovery.

CDR architectures are generally categorized into two major groups: open-loop CDRs and phase-locking CDRs. The former one will be briefly described in Section 2.9, but the focus will be on the latter example as it is robust, reliable and can be monolithically integrated with other circuits.

2.8 Properties of NRZ Data Signal

When the incoming data has a spectral energy at the clock frequency, a synchronous clock can be obtained by passing the data stream through a band-pass filter, often realized as an LC tank or surface acoustic wave (SAW) device, tuned to the nominal clock frequency. In most signaling formats such as NRZ, the data signal has no spectral energy at the clock frequency making it necessary to use the clock recovery process. The power spectral density of an NRZ data signal is given by the following relationship.

$$P(\omega) = T_b \left[\frac{\sin(\omega.T_b / 2)}{\omega.T_b / 2} \right]^2 \qquad (2.1)$$

The spectral density vanishes at ($f = m/T_b$) as shown in Figure 2-7.
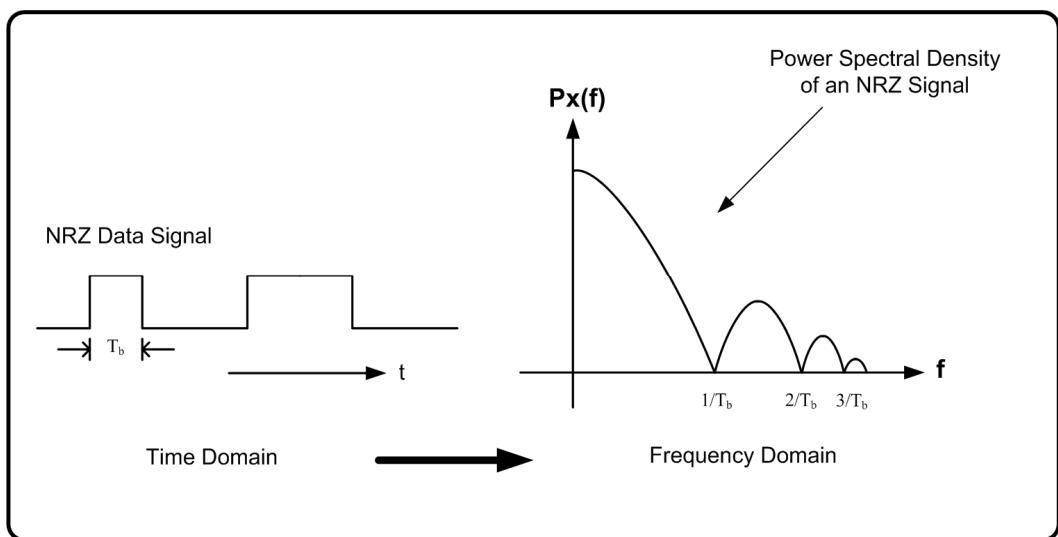
Figure 2-7: Spectrum of an NRZ data signal.

Due to the lack of a spectral component at the bit rate of NRZ format, a clock recovery circuit may lock to spurious signals or simply not lock at all. Thus, NRZ data usually undergoes a non-linear operation at the front end of the circuit so as to create a frequency component at the bit rate. A common approach is to detect each transition and generate a corresponding pulse, this technique known as the edge detection.

2.9 Open Loops CDR Architectures

An edge detection system is illustrated in Figure 2-8(a). First, the NRZ signal is differentiated with respect to time, thus creating positive and negative pulses at each edge of the data waveform. Second, the differentiated data signal is rectified, hence generating only positive pulses at the location of the data signal transition. The resulting spectrum will contains power at the frequency equal to data rate. The precise frequency component can be extracted using a narrow-band filter, thus generating a periodic signal with a frequency equal to the data rate. Using the edge detection method, a CDR circuit can be realized according to the block diagram of Figure 2-8(b). The phase shifter in the recovered clock path is used to guarantee an optimum phase setting of the clock with respect to the incoming data. Thus the bit error rate (BER) during data recovery is minimal.
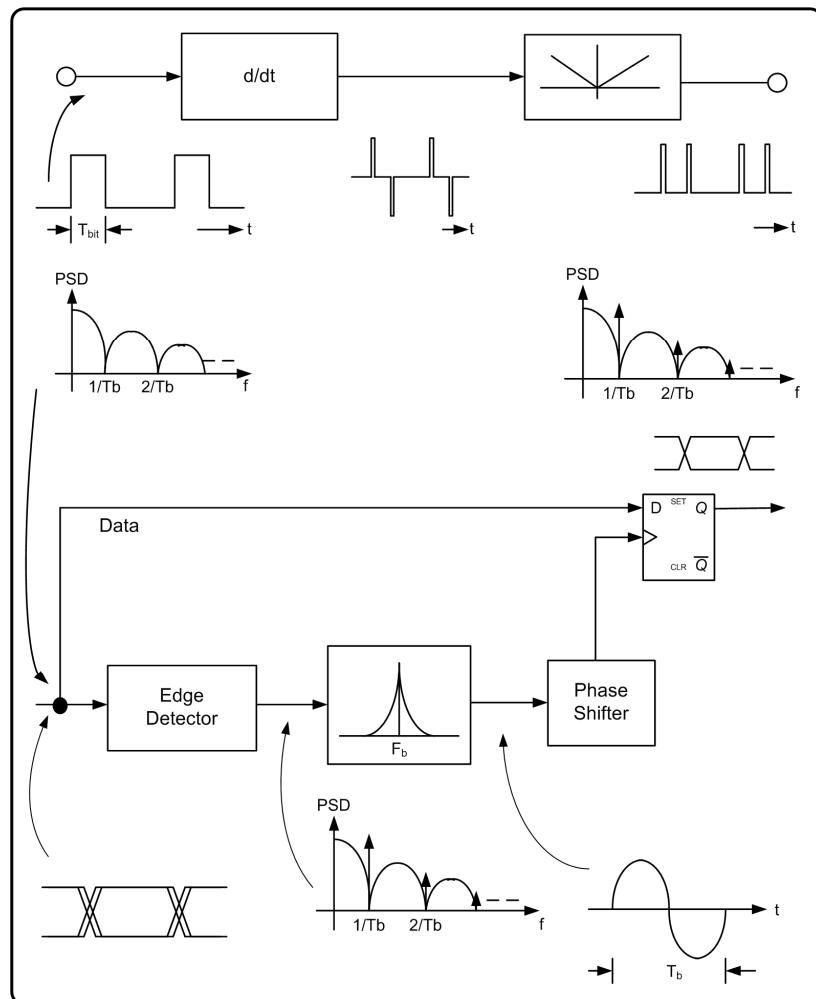
Figure 2-8: Open loop CDR architecture using edge detection technique.

2.10 Phase-Locking CDR Architectures

In this approach, the clock and data recovery is done by synchronizing the random data to a clock signal generated by a voltage controlled oscillator (VCO) in a phase locked loop. During each data transition, the location of that transition with respect to the clock edge is detected. If the data leads the clock, the clock speed is increased. If the data lags the clock, the clock is slowed down. If the zero crossings of the data and the clock coincide, the clock frequency is kept constant to ensure phase lock. Figure 2-9 shows a generic CDR circuit. The VCO generates a clock signal. The phase and frequency of this signal is compared to that of the incoming data in the phase detector, generating an error signal that is passed through the charge pump and the low pass filter to set the voltage required by the VCO to oscillate at the frequency of interest. Phase-locking of the clock to the data means that their phases are different by a small but constant offset. The generated clock signal is also used to retime the data in the decision circuit. As the incoming data is regenerated in this block, its additive noise is suppressed while the amplitude is significantly magnified.
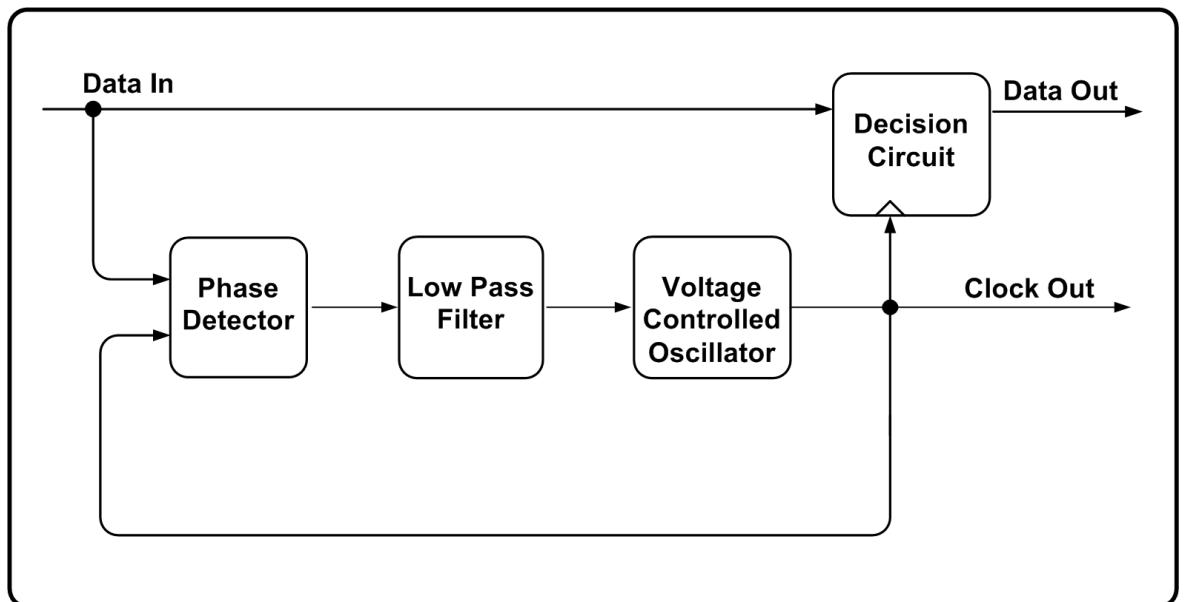
Figure 2-9: Generic phase-locking CDR circuit.

2.11 Full-Rate and Half-Rate CDR Architectures

Phase-locking CDR architectures can be divided into two major groups; full-rate and half-rate. In a full-rate circuit the location of the data transition is compared to the falling or rising edge of the clock which has a frequency equal to the data rate as illustrated in Figure 2-10(a). Therefore, data retiming can be performed using flip-flops that operate either on rising or falling edge of the clock signal. In a half-rate circuit, the location of data transition is compared to that of both the rising and falling edges of clock as shown in figure 2-10(b). For this architecture the clock frequency is equal to one half of the data rate, and the retiming of the data signal is performed using flip-flops triggered on both the falling and rising edges of the clock signals. The main advantage of using half-rate CDR circuit is the reduction of the clocking frequency by a factor of two. Hence, reducing the dynamic power consumption associated with the switching activity of the clock. The DC power dissipation is also reduced because the biasing current is less for circuits working at lower operating frequencies.
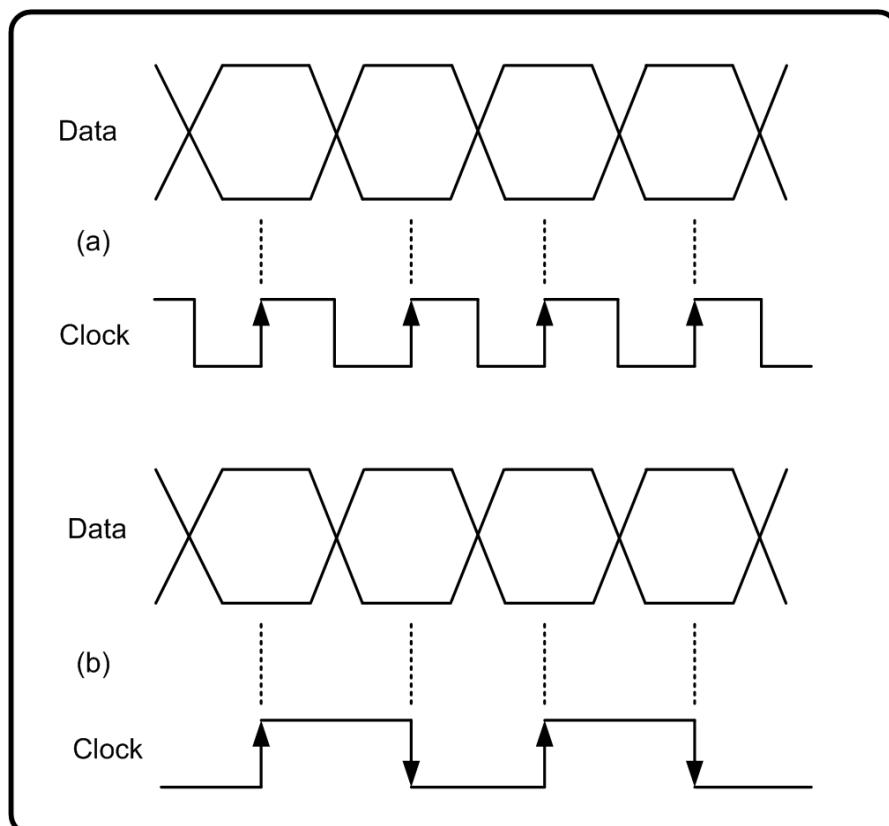
Figure 2-10: (a) Full-rate and (b) half-rate data recovery.

2.12 Periodic Data Signal Phase Detector

In a CDR circuit, the phase information between the data signal and the VCO clock signal is provided by a key component called the phase detector. The phase detector provides information about the spacing between the zero crossing of the data and the clock in term of width modulated pulses. This information is used to set the VCO's control voltage to a value required by the VCO to oscillate at the frequency of interest. When the phase locked state is achieved, the control voltage remains unchanged and the phase detector output does not alter that. A commonly used type of phase detector used with periodic data is an exclusive OR (XOR) gate. As shown in Figure 2-11(b), if $(\Delta\phi)$ is the phase difference between its two inputs, the output of the XOR gate will carry pulses as wide as $(\Delta\phi)$. As illustrated in Figure 2-11(a), the average value of the XOR output signal is linearly proportional to the phase difference of its two input signal where (K_{PD}) is the gain of the phase detector.
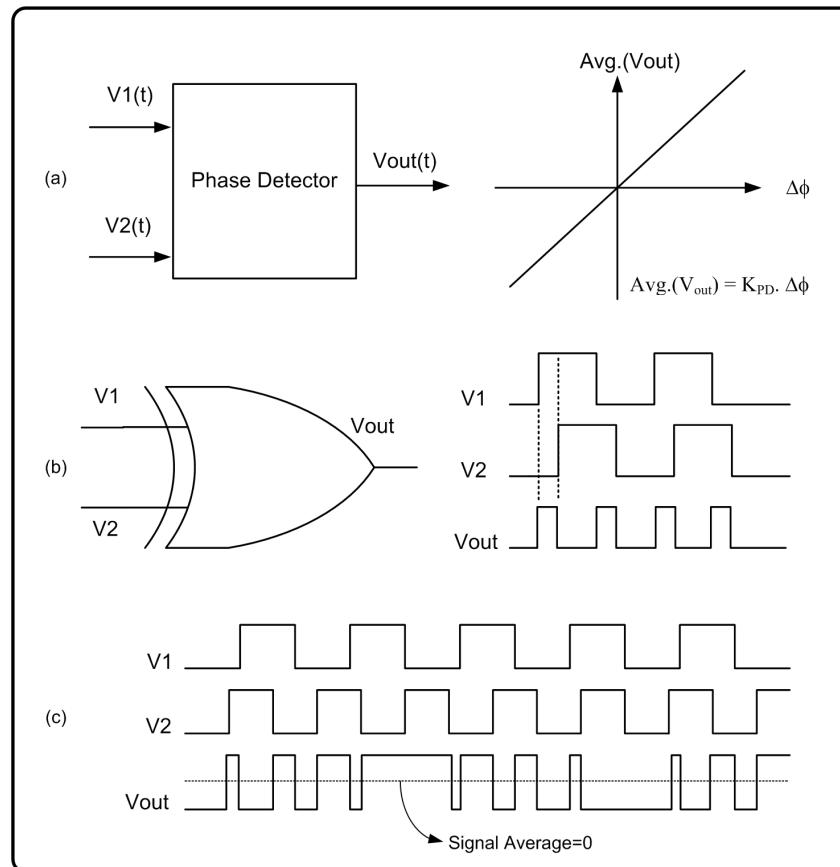
Figure 2-11: XOR gate operating with periodic data signal.

Although this simple approach proves to be useful for applications where the two inputs have identical frequencies and different phases, it falls short in providing frequency error information as the two inputs frequencies start to grow apart from each other. The reason is that if the two frequencies are not equal, the detector generates a beat frequency with an average value of zero (Figure 2-11(c)). The beat signal can still provide efficient information about the phase and frequency difference if the two frequencies are slightly different. To improve the capture range of the phase detector, phase locked loop circuits use additional means of frequency acquisition.

A circuit that can detect both phase and frequency difference is extremely useful because it significantly increases the acquisition range and lock speed of PLL's. The sequential phase and frequency detector (PFD) proves to provide a large range for periodic waveforms [22]. Figure 12-2 shows the implementation of this circuit and the corresponding waveforms when the two inputs have different frequencies and phases. As shown in Figure 2-12(b), if the frequency of input A is greater than of input B, then the PFD produces positive pulses at Q_A, while Q_B remains zero. Conversely, if $f_A < f_B$, positive pulses appear at Q_B while Q_A = 0. If $f_A = f_B$, then the circuit generates pulses at either Q_A or Q_B with a width equal to the phase difference between the two inputs as illustrated in Figure 2-12(c). Thus the average value of difference (Q_A-Q_B) is an indication of the frequency or the phase difference between A and B. The sequential PFD is a major block used for phase detection in frequency synthesizers and clock generators. Its compact and power-efficient structure makes it attractive for low power applications. However, this circuit cannot be used to provide phase error information for random data because in contrast to periodic data a zero crossing at the end of each bit is not guaranteed. Consecutive ones and zeros are very likely to appear in a random sequence hence producing erroneous pulses at Q_A and Q_B.

If for instance, the PLL is in locked state the clock frequency and the data rate will be the same, and the clock edges will be in the middle of the data bits, hence no error pulses will be required to adjust the phase and frequency of the VCO clock signal. However, the sequential PFD produces pulses at Q_A and Q_B driving the VCO clock signal away from its locked state. Therefore this type of PFD is not suitable for random data sequences.
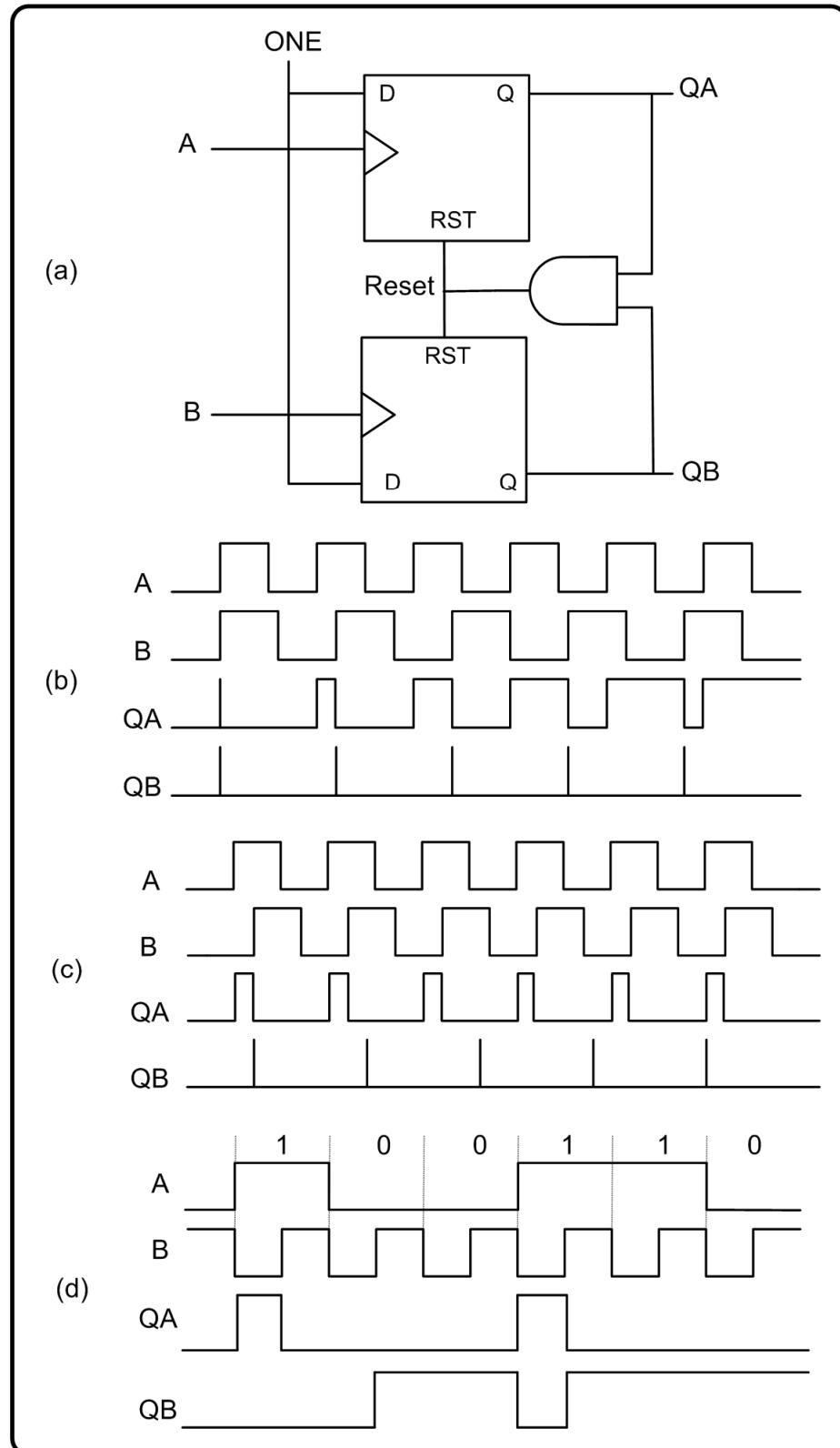
Figure 2-12: (a) Sequential PFD detector. Its response for (b) $f_A > f_B$,
(c) A leading B, and (d) for random data signal.

2.13 Random Data Signal Phase Detectors

Binary data is commonly transmitted in the NRZ format. In this format each bit has duration T_b (bit period), is equally likely to be zero or one, and is statistically independent of other bits. A NRZ data signal has two properties that make the clock recovery task difficult. First, data may exhibit long sequences of consecutive ones or zeros, demanding the clock recovery circuit to "remember" the bit rate during such an interval. This means that, in the absence of data transitions, the clock recovery circuit should not only continue to produce clock, but also cause only a negligible drift in the clock frequency. Second, the spectrum of NRZ data has nulls at frequencies that are integer multiples of the bit rate. Due to the absence of a spectral component at the bit rate in the NRZ format, a CDR circuit may lock to spurious signals or simply may not lock at all. Phase detectors operating with random data sequences are generally categorized in two groups, linear and binary. In a linear phase detector, the phase error signal is linearly proportional to the phase difference, falling to zero in the locked condition. In a binary phase detector, an early or late (binary) signal is generated in response to a phase difference between the clock and data.

2.13.1   Full-Rate Linear Phase Detector for Random Data

In a linear PD, such as the one proposed by Hogge [23], the phase error information is generated at each data transition and produced by taking the difference of two pulses. One of them is width modulated the width is linearly proportional to the phase difference between the clock and data, whereas the other pulse has a fixed width. Gate-level implementation of Hogge's phase detector is shown in Figure 2-13. The NRZ input data signal is sent through two D-type flip-flops. The first flip-flop samples the data signal on the rising edge of the clock, whereas the second flip-flops samples the output of the first one on the falling edge of the clock. If the three signals, D_{in}, A, and D_{out} are applied to two XOR gates, the resulting output signals will have the properties of a linear phase detector. The Error output signals will appear at each data transition with a width proportional to the phase difference between the clock and the data. The reference output will always have pulses as wide as half the clock period. An important feature of the Hogge PD is the automatic retiming of the data sequence.

In the lock condition, the clock signal zero crossings will appears in the middle of the bits, meaning that the bits are sampled at their optimum points.
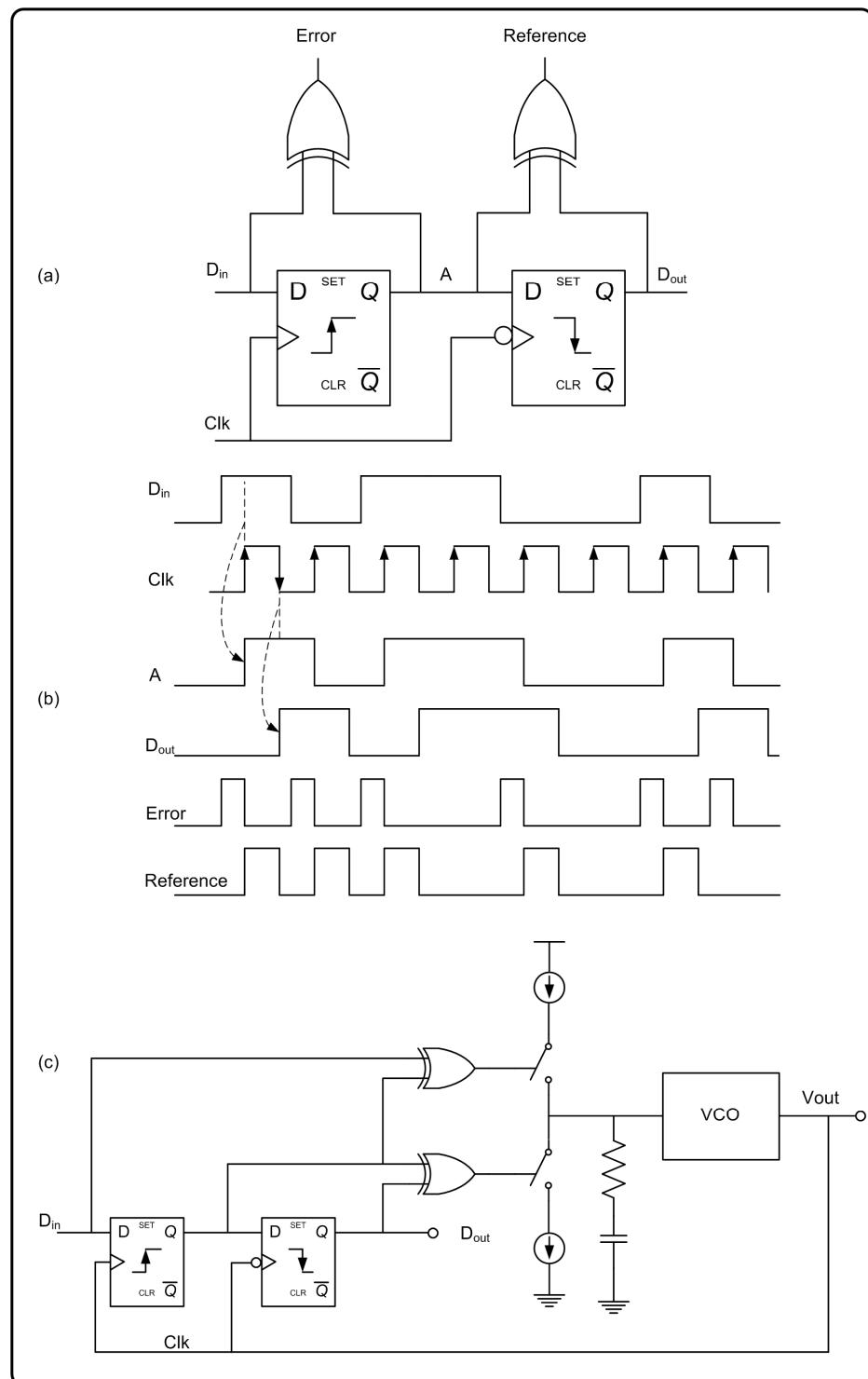
Figure 2-13: (a) Hogge PD implementation, (b) operation and (c) its CDR circuit.

2.13.2   Full-Rate Binary Phase Detector for Random Data

In a binary phase detector, a binary error signal is generated in response to an arbitrary phase difference between the clock and the data. This binary error signal determines whether the clock phase is "early" or "late" with respect to the data phase. A commonly used binary phase detector is the one proposed by Alexander [24], in which the zero crossings of the data are measured as early or late events when compared with the transitions of the clock signal. The structure of the Alexander phase detector allows for automatic retiming of the data. During any particular clock interval, this binary phase detector provides three binary samples of the data signal: the previous bit (*A*), a sample of the current bit at the zero crossing (*B*); and the current bit (*C*) (Figure 2-14(b)). Figure 2-14 (a) depicts the value of these samples for the late and early clocks. The retimed data is taken from A. The location of the clock edge with respect to the data edge can be determined based on the following rules:

- If A = B ≠ C, clock is early.

- If A ≠ B = C, clock is late.

- If A = B = C, no data transition has occurred

Using the above observations, the three samples can be used to produce a phase error in a CDR circuit. The early signal can be formed as B ⊕ C and the late signal is generated as A ⊕ B. The desired phase error can be obtained by subtracting the early signal from the late signal. Figure 2-14(d) shows a CDR circuit employing an Alexander phase detector. The XOR gate outputs drive voltage-to-current converters so that the two signals can be summed in the current domain, and the result is applied to the loop filter. The high gain of the Alexander PD yields a small phase offset in the locked condition. CDR circuits using similar PD are described in [25-27].
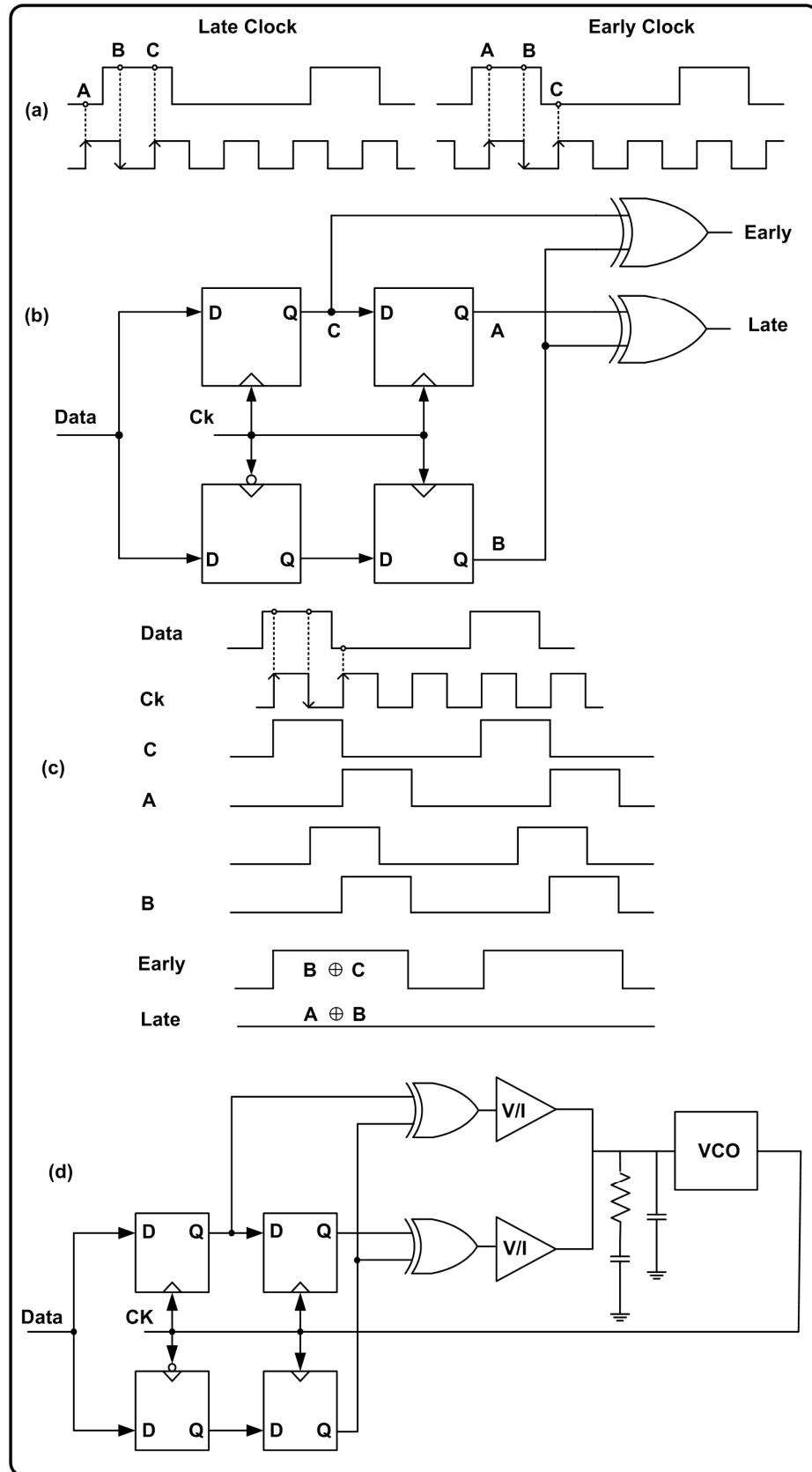
Figure 2-14: (b) Alexander PD, (c) waveforms operation and, (d) its CDR circuit.

2.13.3   Half-Rate Binary Phase Detector for Random Data

Let us now consider the Early-Late method for half-rate operation. Since the Alexander PD already requires sampling on both clock edges for full-rate detection, it is then necessary to use additional phases of the clock if it is to operate in the half-rate mode. Shown in Figure 2-15, the solution involves sampling the data in both the in-phase and quadrature phases of the clock, CK_I and CK_Q respectively. Now A, B and C play the same role as the consecutive samples in a full-rate counterpart. As depicted in Figure 2-15, the implementation incorporates three flip-flops sampling the data using CK_I and CK_Q, and two XOR gates that produce $A \oplus B$ and $B \oplus C$. In the locked condition, the rising edge of CK_Q occurs in the vicinity of the data zero crossings.
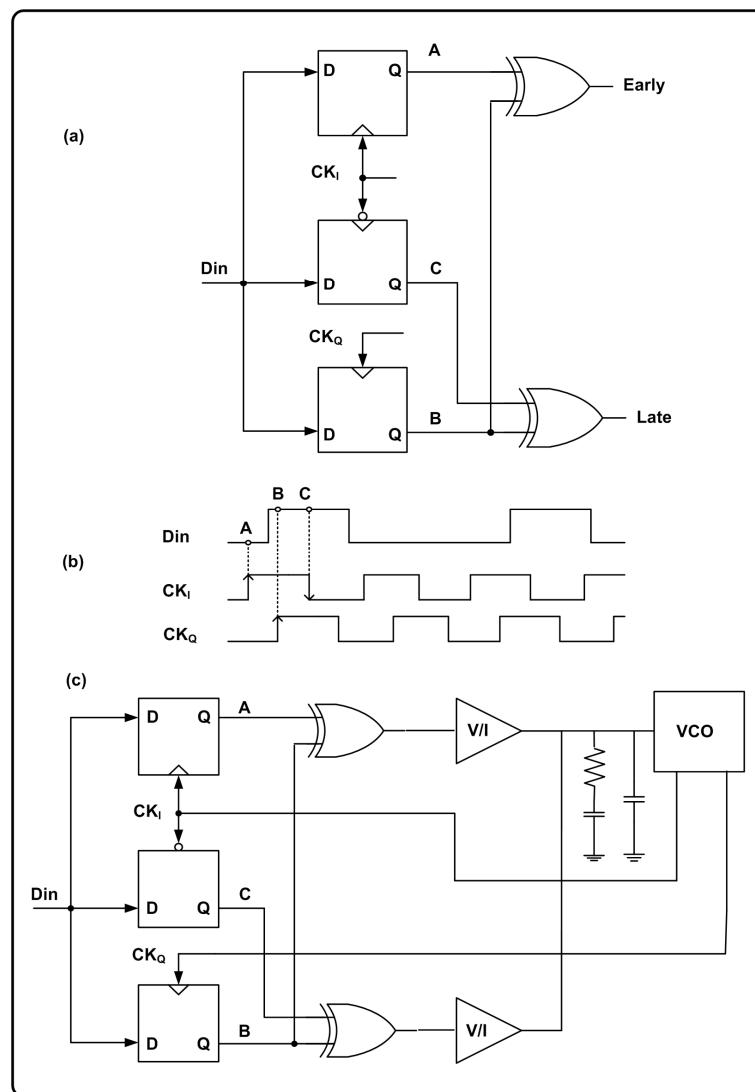
Figure 2-15: (a) Half-rate binary PD implementation, (b) use of quadrature clocks for half-rate phase detection, and (c) its CDR circuit.

2.14 Frequency Detectors

Data communication standards require operation at a precise data rate. Therefore the frequency of the VCO should be equal to the data rate. However, the VCOs in the CDR circuits are generally designed with a large tuning range to accommodate for the process and temperature variations. On the other hand, the phase-locking CDR circuits have narrow capture range. This range is primarily determined by two factors: the PLLs bandwidth and the phase detector topology. The loop bandwidth is a communication standard dependent and does not exceed normally a few MHz. The capture range of the linear PD is a fraction of one percent of the incoming data rate, and it is typically a few percent for binary a PD. Therefore the CDRs capture range is much smaller than the VCO's tuning range. For this reason, it is unlikely that CDR circuits will acquire lock to the data when the circuit turns on and the VCO starts oscillating at a frequency that is very different from the data rate. This limitation calls for an aided acquisition mechanism. Various frequency detection techniques have been used that operate with or without a reference signal. The idea is that as the circuit is turned on, the frequency detector (FD) pushes the VCO frequency close to the data rate. When the frequency difference between the VCO and the data rate is small enough to fall into the capture range of PD, the FD is then disabled and the PD takes over. A frequency detector must generate an output the average of which represents the polarity and magnitude of the frequency difference at its inputs. Considering the block diagram of the circuit shown in Figure 2-16, and assuming for instance that all input signals are periodic, example:

$$
\begin{aligned}
x_1(t) &= A_1 \cos \omega_1 t, \\
x_{2I}(t) &= A_2 \cos \omega_2 t, \\
x_{2Q}(t) &= A_3 \sin \omega_2 t,
\end{aligned}
\tag{2.2}
$$

Then the output at each of the two multipliers is:

$$
\begin{aligned}
x_1(t) \cdot x_{2I}(t) &= (\frac{A_1 A_2}{2})[\cos(\omega_1 + \omega_2)t + \cos(\omega_1 - \omega_2)t] \\
x_1(t) \cdot x_{2Q}(t) &= (\frac{A_1 A_2}{2})[\sin(\omega_1 + \omega_2)t + \sin(\omega_1 - \omega_2)t]
\end{aligned}
\tag{2.3}
$$

The component at $(\omega_1 + \omega_2)$ can be removed by low-pass filtering, thus the above equations simplify to:

$$x_A(t) = (\frac{A_1 A_2}{2}) [\cos(\omega_1 - \omega_2)t]$$
$$x_B(t) = (\frac{A_1 A_2}{2}) [\sin(\omega_1 - \omega_2)t]$$

(2.4)

Hence the signal $x_C(t)$ at the point C is given by:

$$x_C(t) \propto (\frac{A_1 A_2}{2})^2 \cdot (\omega_1 - \omega_2) = \alpha \cdot \Delta\omega$$

(2.5)

Eq. 2.5 shows that the signal $x_C(t)$ issued from the FD is directly proportional to the frequency difference at its inputs ($\Delta\omega$) and changes sign with that difference. The topology of Figure 2-16(a) is called a "quadricorrelator" [28]. This technique requires that the signal $x_1(t)$ contains a spectral line or component, thus circuit must then be proceeded by an edge detector for operation with an NRZ random data signal (Figure 2-16(b)).
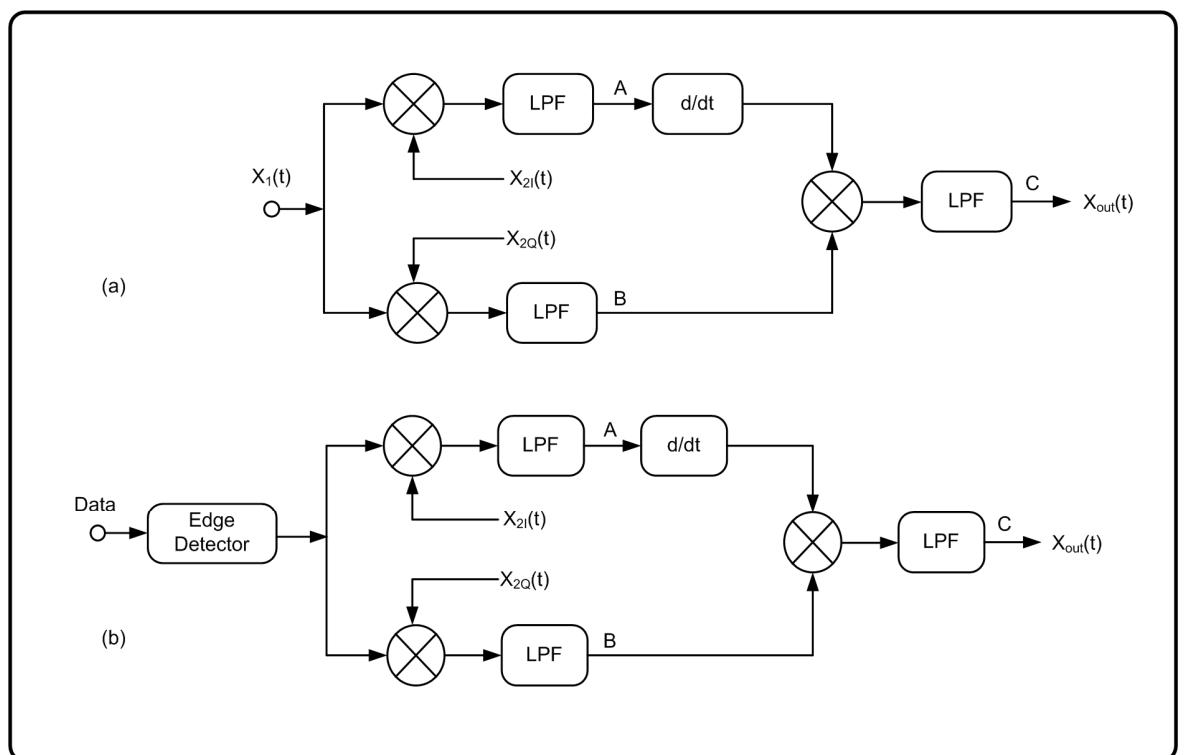
Figure 2-16: Analog quadricorrelator FD for (a) periodic signal and, (b) random data signal.
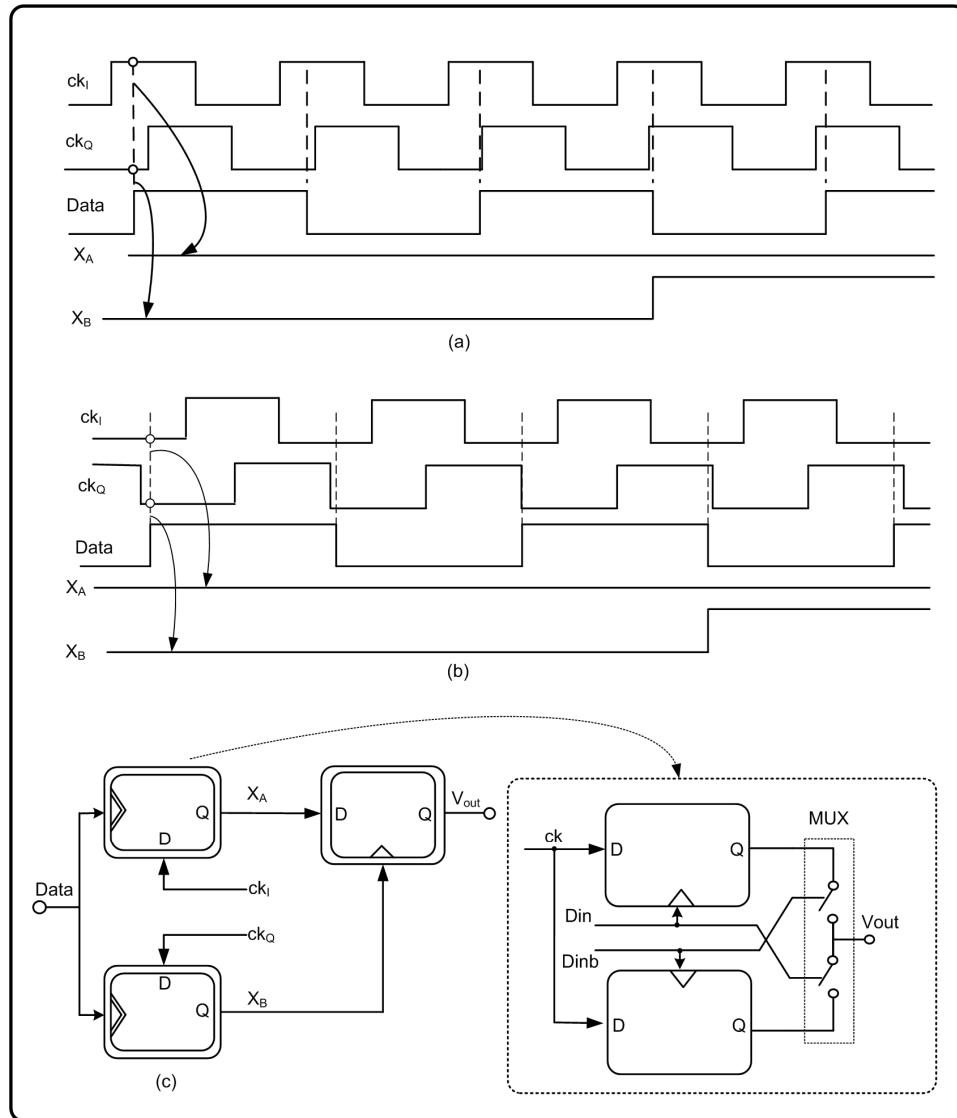
Figure 2-17: Digital quadricorrelator FD, (a) waveform for fast, (b) for slow, (c) Implementation.

It is possible to construct a digital version of the analog quadricorrelator, thus eliminating the need for an analog edge detector and achieving robust operation. As illustrated in Figure 2-17(c), using two double-edge triggered flip-flops that sample the quadrature phases (ck_I and ck_Q) of the clock by the data edges generates two beat waveforms with a 90° phase shift [29]. As shown in Figure 2-17(a), the rising edges of the signal $x_B(t)$ are used to sample the level on $x_A(t)$. The result of this sampling will be high for a VCO frequency that is higher than the data rate, and it will be low for a VCO frequency that is lower than the data rate. Other examples of FDs are described in [30, 31]. A half-rate FD is presented in [32].

2.15 CDR Architectures

After studying the design and analysis of PD and FD for periodic and random signal complete CDR architectures can now be developed. A robust architecture must perform the following operations: phase and frequency acquisition to ensure lock despite process and temperature variations of the VCO frequency and; data retiming inside the phase detector to avoid systematic skew [28].

2.15.1   Full-Rate Referenceless CDR Architecture

Using random data based FD eliminates the need for external reference frequencies. Figure 2-18 depicts a referenceless architecture containing two loops: a frequency loop employing a digital quadricorrelator FD from Figure 2-16, and a phase loop incorporating one of the phase detectors studied in Sections 2.13.1-2.13.3. Upon start-up or the loss of phase lock, the FD produces a DC voltage that drives the VCO frequency toward the input data rate. When the frequency error is small and falls within the capture range of the phase loop, the PD then takes over, phase-locking the clock to the data.
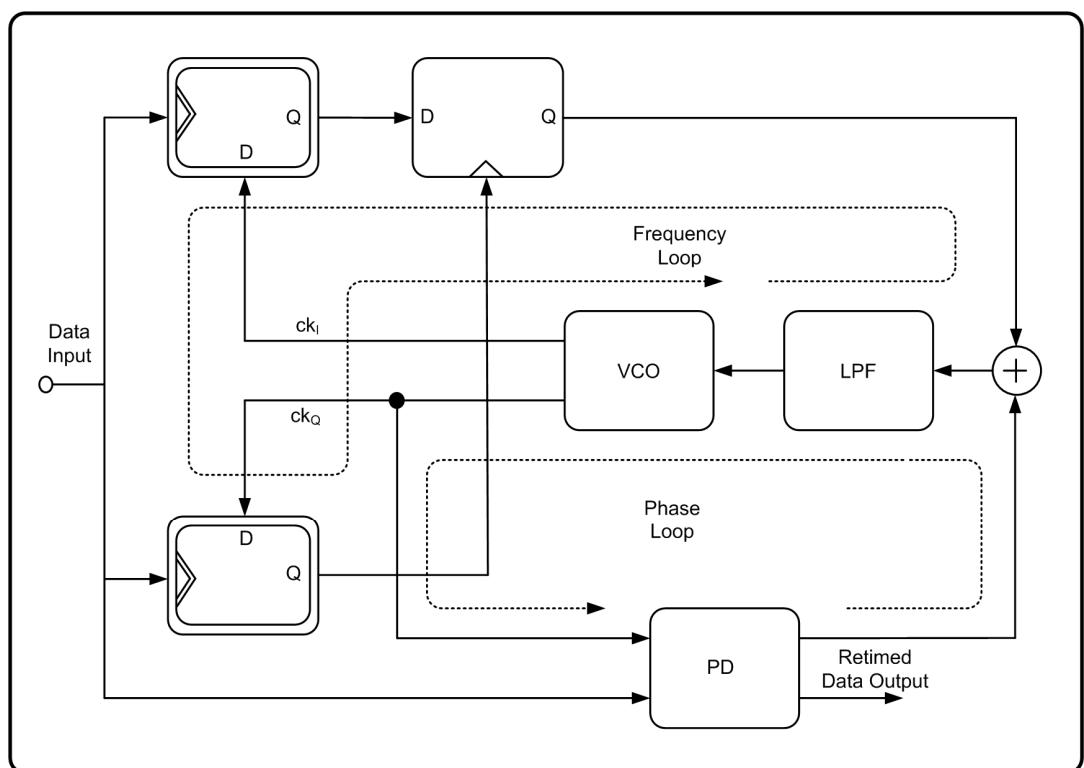
Figure 2-18: Referenceless CDR architecture incorporating PD and FD.

2.15.2   *Dual-Loop CDR Architecture with External Reference*

As illustrated in Figure 2-19, the PD and the FD are connected to the loop filter through a multiplexer (MUX). When the circuit's power is turned on, the multiplexer activates the frequency loop first and the circuit locks to the reference clock bringing the VCOs frequency close to the data rate. Once the required frequency is reached, the frequency detector is then disabled reducing the power consumption, and the MUX switches to the phase loop. The operating mode of the multiplexer is determined by a lock detector that measures the frequency difference between the reference clock and the VCOs frequency [33].
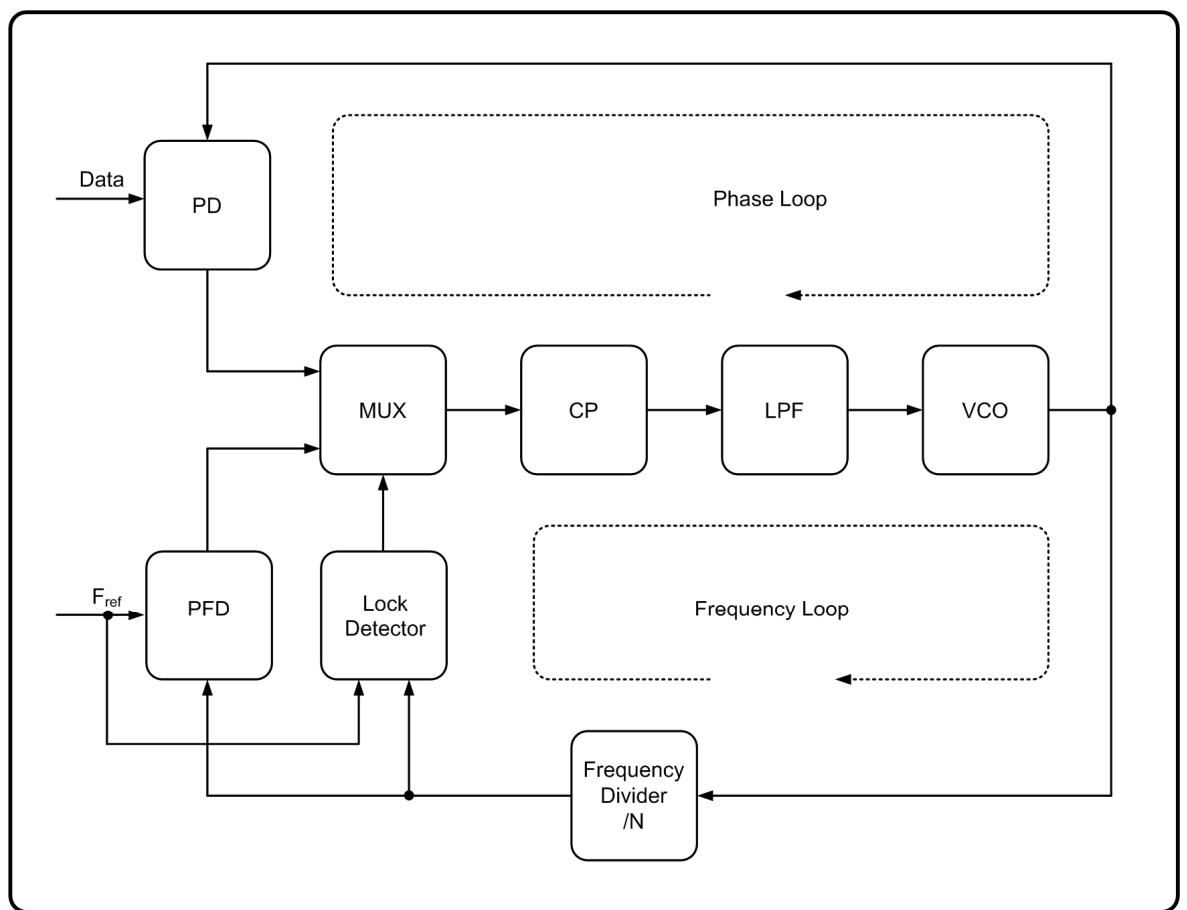
Figure 2-19: Dual loop CDR architecture with an external reference clock.

2.16 Summary of Prior Art

State of the art works on CDR circuits are summarized in Table 2.2. The indicated data rate corresponds to the data speed at the CDR input. The clock frequency is the frequency of the clock signal that is used to transmit the data and has to be extracted by the CDR circuit.

Ref.	Technology	Supply Voltage (V)	Data rate Gb/s	Architecture Loop Number	Clock rate GHz	Architecture Type
Our [34]	0.13-µm CMOS	1.2	10	2-loop PLL	2.5	Quarter-Rate
[25]	0.18-µm CMOS	1.8	3.125	2-loop PLL	1.56251	Half-Rate
[26]	0.18-µm CMOS	1.8	10	2-loop PLL	5	Half-Rate
[27]	0.18-µm CMOS	1.8	10	1-loop PLL	5	Full-Rate
[35]	0.18-µm CMOS	2.5	10	1-loop PLL	5	Half-Rate
[36]	0.35-µm CMOS	3.3	0.622	2-loop PLL	0.622	Full-Rate
[37]	Si-Bipolar	5	2.488	2-loop PLL	2.488	Full-Rate
[38]	0.35-µm CMOS	3.3	1.25	1-loop PLL	0.625	Half-Rate
[39]	Si-Bipolar	4.5	1.5	1-loop PLL	1.5	Full-Rate
[33]	Si-Bipolar	5.25	2.488	2-loop PLL	2.488	Full-Rate
[40]	0.4-µm CMOS	3.3	2.5	1-loop PLL	2.5	Full-Rate
[41]	0.18-µm CMOS	1.8	5	1-loop PLL	2.5	Half-Rate
[42]	0.18-µm CMOS	1.8	9-16	2-loop PLL	4.5-8	Half-Rate
[43]	0.35-µm BiCMOS	3.3	10	2-loop PLL	10	Full-Rate
[44]	1-µm BiCMOS	3	2.5	2-loop PLL	2.5	Full-Rate

Table 2-2: Summary of the prior art, including the work done in this thesis.

We presented in this chapter the problems and limitations associated with the use of busses as a medium of synchronous communication in today's complex SOC. To alleviate previous problems an asynchronous link based on PLL CDR circuits has been proposed as a high performance alternative solution. Furthermore, we reviewed the current state of the art of PLL-based CDR; from the literature it is apparent that there is considerable scope of improvement in their designs for asynchronous link based communication in SOC. This thesis therefore presents a detailed study of quarter-rate PLL-based CDR circuit.

3 Introduction

In this chapter, a mathematical development of the PLL will be carried out covering the following subjects:

1. Simplified time-domain analysis of the PLL in the locked state. In other words, studying the tracking property of the PLL, in which any change in the frequency input will be tracked by the output through the phase error signal [Eq. 3.10]

2. Frequency-domain stability analysis of the PLL with a simple RC filter and without a charge pump. We will find analytical expressions relating the value of the filter components R and C to the stability parameters such as, the phase margin (ϕ_m), damping factor (ξ), and bandwidth (ω_{-3dB}) [Eq. 3.19-3.22].

3. Same as in (2), but for a charge pump PLL (CP-PLL) [Eq. 3.33-3.35].

4. Stability parameters comparison of the PLL and the CP-PLL [Table 5.1].

5. CDR jitter specifications and its relation to the PLL parameters [Eq. 3.49, & 3.58].

A phase-locked loop (PLL) is a circuit that synchronizes the phase and frequency of a signal generated by a local oscillator with that of a reference signal, by means of the phase difference between the two signals. PLLs are primarily used in communication systems. For example, they recover clock signals from digital data signals, recover the carrier from satellite transmission signals, perform frequency and phase modulation/demodulation, and synthesize exact frequencies for receiver tuning [47].

3.1   Simplified PLL Block Diagram

As shown in figure 3-1, the PLL circuit consists basically of three blocks:
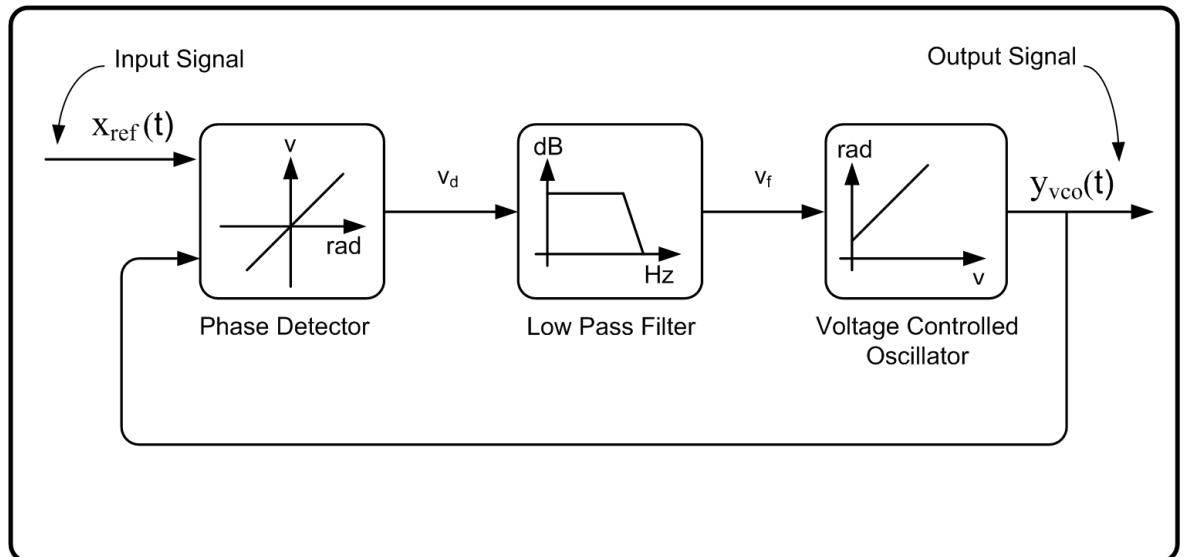
Figure 3- 1: Simplified PLL block diagram.

1. Phase detector (PD), a simple one can be realized using an analog multiplier. Since the PD is performing a multiplication, hence the output signal v_d will have the following form:

$$v_d = k_{pd} \cdot f(\Phi_{ref} - \Phi_{vco})$$ (3.1)

Where f is a function of the phase difference between the reference and the oscillator signals and k_{pd} is the conversion gain of the *PD* measured in units of volt per radian (V/rad).

2. Low pass filter LPF. Its output voltage is denoted by v_f.

3. Voltage controlled oscillator *VCO*, the *VCO*'s angular frequency ω_{vco} is controlled by the output filter voltage v_f according to the following expression:

$$\omega_{vco} = \omega_0 + k_{vco} \cdot v_f$$ (3.2)

Where ω_0 is the free running angular frequency, corresponding to $v_f = 0$, and k_{vco} is the

VCO conversion gain, expressed in units of radians per volt per second (rad/V.sec).

3.2 PLL time-domain operation in the locked state

In this section, the time-domain operation of the PLL will be studied. When the PLL is

operating in the synchronized state, the angular frequency of both the input reference

signal (ω_{ref}) and the VCO's output signal (ω_{vco}) will be equal. Let the following

expressions represent, respectively, the input reference and the VCO output signal:

$$x_{ref}(t) = x_0.\sin(\omega_{ref}t + \varphi_0)$$
$$y_{vco}(t) = y_0.\cos(\omega_{vco}t + \psi_0)$$

(3.3)

As the PD is performing a multiplication, the signal at its output is giving by:

$$v_d = \beta\, x_{ref}(t).y_{vco}(t) = \beta\, x_0.y_0.\sin(\omega_{vco}t + \psi_0).\cos(\omega_{ref}t + \varphi_0)$$
$$v_d = k_{pd}\left\{\sin\left[(\omega_{vco} + \omega_{ref})t + (\psi_0 + \varphi_0)\right] + \sin\left[(\omega_{vco} - \omega_{ref})t + (\psi_0 - \varphi_0)\right]\right\}$$

(3.4)

Where β is a constant. After filtering we obtain the following signal

$$v_f = k_{pd}\left\{\sin\left[(\omega_{vco} - \omega_{ref})t + (\psi_0 - \varphi_0)\right]\right\} = k_{pd}\sin\theta_e(t)$$
$$where \quad k_{pd} = \beta\frac{x_0 y_0}{2} \quad and \quad \theta_e(t) = (\omega_{vco} - \omega_{ref})t + (\psi_0 - \varphi_0)$$

(3.5)

At the start, there is no voltage (v_f) applied to the input of the VCO, thus ($\omega_{vco} = \omega_{ref}$). As

shown in equation 3.5, the signal issued from the filter v_f carries information about the

frequency error (ω_{vco}-ω_{ref}) and the phase error (ψ_0-ϕ_0), between the input (reference) and

the output (VCO) signals.

Since; $-1 \le \sin\theta_e \le 1 \quad \forall\theta_e$ then $-k_{pd} \le v_f \le k_{pd}$ and, based on the Eq. 3.2, the angular

frequency (ω_{vco}) of the VCO will be limited by the range [ω_{min}, ω_{max}] such that:

$$\omega_{min} \leq \omega_{vco} \leq \omega_{max}$$

$$\text{Where } \omega_{min} = \omega_0 - k_{pd}.k_{vco} \text{ and}$$
(3.6)

$$\omega_{max} = \omega_0 + k_{pd}.k_{vco}$$

Since the VCO's angular frequency is limited by the range $[\omega_{min}, \omega_{max}]$, then in the locked state, there exist a value in that range which is equal to the reference angular frequency (ω_{ref}). Therefore, based on the equation 3.1, the following double equations are fulfilled in the locked state of the PLL:

$$v_f = \frac{\omega_{ref} - \omega_0}{k_{vco}} = \frac{\omega_{vco} - \omega_0}{k_{vco}}$$
(3.7)

The above equation shows that in the locked state any change that may occur on (ω_{ref}) or (ω_{vco}), will be tracked by the PLL, and the filter voltage (v_f) will change accordingly. As an example, if for instance a random signal decreases the VCO frequency by an amount of ($\Delta\omega_{vco}$), then the filter voltage (v_f) will be increased by an amount of (Δv_f) and the VCO will be controlled by the total voltage ($v_f + \Delta v_f$) and hence the VCO angular frequency will be increased to compensate for the action of the disturbance. The result will then be:

$$\Delta\omega_{vco} = k_{vco}.\Delta v_f$$
(3.8)

Therefore, as soon as the VCO angular frequency is driven away from the reference one by a random signal, or a temperature variation, a phase error signal is generated and hence a voltage will also be generated, forcing the VCO to be synchronized with the reference angular frequency. As shown in Eq. 3.5, the signal issued from the filter is giving by:

$$v_f = k_{pd}\left\{\sin\left[(\omega_{vco} - \omega_{ref})t + (\psi_0 - \varphi_0)\right]\right\} = k_{pd}\sin\theta_e(t)$$
(3.9)

For a small phase error, the last equation can be simplified to:

$$v_f = k_{pd}.\theta_e(t) \quad then$$

$$\Delta v_f = k_{pd}.\Delta\theta_e(t)$$

$$\Delta v_f = \frac{\Delta\omega_{vco}}{k_{vco}} \quad and \tag{3.10}$$

$$\Delta\theta_e(t) = \frac{\Delta\omega_{vco}}{k_{pd}.k_{vco}}$$

The last expression of the phase error signal shows that the VCO is forced to shift its angular frequency to become identical to the reference one through the phase error signal $\Delta\theta_e(t)$.

3.3 Frequency-domain PLL stability analysis

In the previous section, an elementary time-domain analysis of the PLL in the locked state was performed, and an approximation expression was developed relating the phase error signal to the required change of the VCO's angular frequency in order to maintain synchronization. Since the PLL is a feedback loop system, a stability analysis of that system is necessary in order to guarantee its stability; otherwise the PLL may oscillate and never reach the required steady state. In this section a frequency domain analysis will be carried out to determine the stability limits and conditions of the PLL circuit, as well as a calculation of the low pass filter components (i.e. *R* and *C*) based on the previous conditions results. In order to transform the time domain PLL block diagram of figure 3-1 to the frequency domain, a simple case will be considered, and its results will be generalized.

3.3.1 PLL with a simple RC filter and without a charge pump

Let us consider the filter illustrated in figure 3-2. According to Ohm's law, one can write:
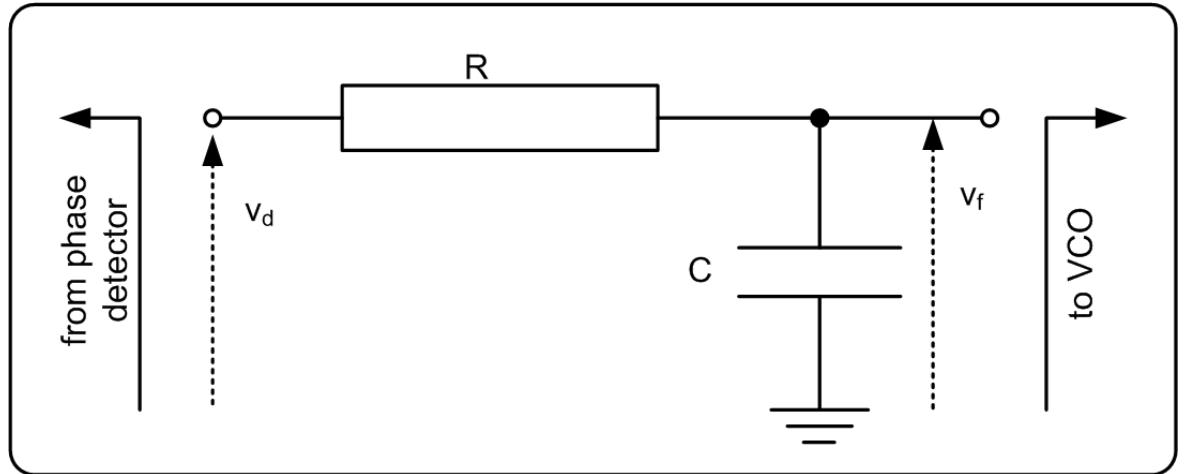
Figure 3-2: RC filter.

$$v_d(t) = RC\frac{dv_f}{dt} + v_f(t) \tag{3.11}$$

Taking the Laplace transform of equation 3.11 give us:

$$V_d(s) = RCsV_f(s) + V_f(s) = [RCs+1]V_f(s), \text{ then}$$

$$TF_{filter}(s) = \frac{V_f(s)}{V_d(s)} = \frac{1}{RCs+1} = \frac{1}{s\tau+1} \tag{3.12}$$

Where TF(s) is the transfer function of the *RC* filter of figure 3-2 and $\tau\,(=RC)$ is the time constant of the filter. By integrating with respect to time Eq. 3.2, one can obtain

$$\int_0^t \omega_{vco}\,dt = \int_0^t \omega_0\,dt + k_{vco}\int_0^t v_f(t)\,dt \text{ Then, } \omega_{vco}(t)t = \omega_0 t + k_{vco}\int_0^t v_f(t)\,dt \text{ and}$$

$$[\omega_{vco}(t) - \omega_0)]\,t = k_{vco}\int_0^t v_f(t)\,dt \text{ , hence}$$

$$\theta_{vco}(t) = k_{vco} \int_0^t v_f(t)\, dt \qquad\qquad (3.13)$$

Taking the Laplace transform of equation 3.13 give us the following:

$$\Theta_{vco}(s) = \frac{k_{vco}}{s} V_f(s)$$

Rearranging the last equation, we obtain: $TF_{vco}(s) = \dfrac{\Theta_{vco}(s)}{V_f(s)} = \dfrac{k_{vco}}{s}$

(3.14)

$TF_{vco}(s)$ is the transfer function of the VCO. Taking the Laplace transform of the first equation of 3.10, one can obtain:

$$V_f(s) = k_{pd}\,\Theta_e(s)\text{, then } TF_{pd}(s) = \frac{V_f(s)}{\Theta_e(s)} = k_{pd} \qquad\qquad (3.15)$$

$TF_{pd}(s)$ is the transfer function of the phase detector. We now have the *s* or frequency domain transfer function of all the PLL blocks; therefore we can redraw the PLL block diagram in frequency domain.
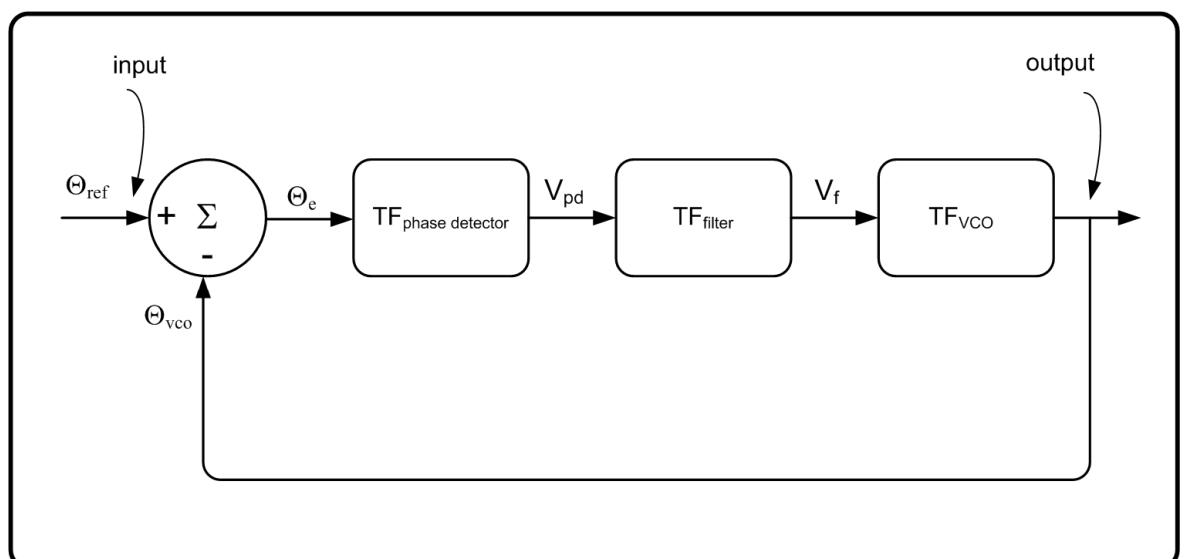
Figure 3-3: Frequency-domain PLL block diagram.

Based on the definition of the feedback system in control theory, the open loop transfer function *G(s)* of the PLL is giving as follow:

$$G(s) = TF_{pd}(s) \cdot TF_{filter}(s) \cdot TF_{vco}(s) = k_{pd} \cdot \frac{1}{s\tau+1} \cdot \frac{k_{vco}}{s} = \frac{k_{pd} \cdot k_{vco}}{s(s\tau+1)}$$

Setting $k_{pd} \cdot k_{vco} = k$ which is measured in sec^{-1}, the open loop transfer function *G(s)* becomes:

$$G(s) = \frac{k_{pd} \cdot k_{vco}}{s(s\tau+1)} = \frac{k}{s(s\tau+1)} \tag{3.16}$$

And the closed loop transfer function of the PLL, *H(s)* will be defined as follow:

$$H(s) = \frac{G(s)}{1+G(s)} = \frac{k\big/s(s\tau+1)}{1+k\big/s(s\tau+1)} = \frac{k/\tau}{s^2+s/\tau+k/\tau} \tag{3.17}$$

Since the denominator of the function *H(s)* is a polynomial of second order, the loop is second order and it has the following general form:

$$H(s) = \frac{\omega_n^2}{s^2 + 2 \cdot \xi \cdot \omega_n \cdot s + \omega_n^2} \tag{3.18}$$

Comparing Eq. 3.17 with 3.18, we obtain:

$$2 \cdot \xi \cdot \omega_n = \frac{1}{\tau} \text{ and } \omega_n^2 = \frac{k}{\tau}$$

$$\xi = \frac{1}{2\sqrt{k \cdot \tau}} \tag{3.19}$$

Where (ξ) is the damping factor of the loop and is unitless. (ω_n) is the natural angular frequency of the loop and is measured in radian per second (rad/s). From Eq. 3.19, we notice that increasing the factor ξ, and hence the loop stability, requires a decreasing of design parameters (*k*) and (τ).

3.3.2 Bode stability analysis of the PLL

For convenience, we rewrite the open loop transfer function of the PLL without a charge pump by substituting s with $(j\omega)$.

$$G(j\omega) = \frac{k}{j\omega(1 + j\omega\tau)}$$

The function $G(j\omega)$ is a complex function. Its magnitude and phase are giving as follow:

$$|G(j\omega)| = \frac{k}{\omega \cdot \sqrt{(1 + \omega^2\tau^2)}} \ and \ \tan\phi = -\frac{1}{\omega\tau} \tag{3.20}$$

The angular frequency for which $|G(j\omega)| = 1$ is called the cut-off frequency of the PLL and is denoted by (ω_{-3dB}).

$$|G(j\omega_{-3dB})| = \frac{k}{\omega_{-3dB} \cdot \sqrt{(1 + \omega_{-3dB}^2 \cdot \tau^2)}} = 1, \ then \ k^2 = \omega_{-3dB}^2 \cdot (1 + \omega_{-3dB}^2 \cdot \tau^2)$$

Rearranging last equation and using equation 3.19, one can obtain:

$$\omega_{-3dB}^4 + 4\xi^2\omega_n^2\omega_{-3dB}^2 - \omega_n^4 = 0$$

Solving the last equation with respect to (ω_{-3dB}), the cut-off frequency of the PLL can be determined in terms of the damping factor (ξ):

$$\omega_{-3dB} = \omega_n\sqrt{\sqrt{1 + 4\zeta^4} - 2\xi^2} \tag{3.21}$$

Substituting equation 3.21 in 3.20, one can obtain the phase of the open loop transfer function $G(j\omega)$ at the cut-off frequency ω_{-3dB} which correspond to the phase margin (ϕ_{margin}).

$$\phi = -\frac{\pi}{2} - \arctan(\frac{1}{\tau\omega_{-3dB}}) = -\frac{\pi}{2} - \arctan(\frac{2\xi\omega_n}{\omega_{-3dB}}) = -\frac{\pi}{2} - \arctan[\frac{2\xi}{\sqrt{\sqrt{1+4\zeta^4}-2\xi^2}}]$$

Then, $\phi = -\frac{\pi}{2} - \arctan[\frac{2\xi}{\sqrt{\sqrt{1+4\zeta^4}-2\xi^2}}]$

The phase margin of the function $G(j\omega)$ is defined as: $\phi_{margin} = \phi|_{|G|=1} +180^o$

Then, $\phi_{margin} = \frac{\pi}{2} - \arctan[\frac{2\xi}{\sqrt{\sqrt{1+4\zeta^4}-2\xi^2}}]$ (3.22)

 The PLL is normally stable when the phase margin is equal to 45° and higher. Thus to find the corresponding value of (ξ), the following equations should be solved with respect to (ξ):

$$\frac{\pi}{4} = \frac{\pi}{2} - \arctan[\frac{2\xi}{\sqrt{\sqrt{1+4\zeta^4}-2\xi^2}}]$$

$$1 = \frac{2\xi}{\sqrt{\sqrt{1+4\zeta^4}-2\xi^2}}$$

(3.23)

The solution resulting from solving 3.23 is ξ=0.42. Figure 3.4 illustrate the Bode diagram, and it corresponds to the amplitude and phase of the open loop transfer function $G(j\omega)$.
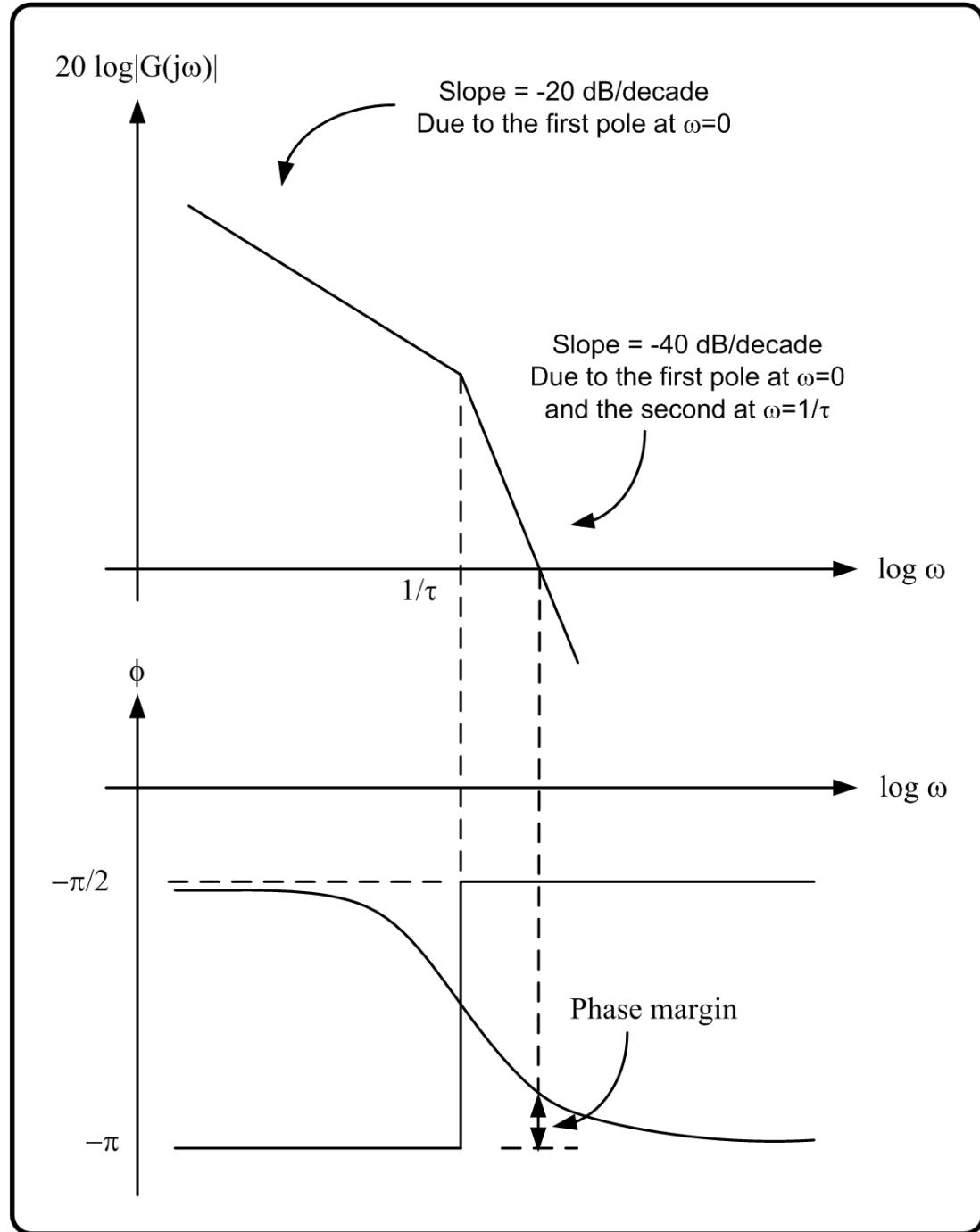
Figure 3-4: Bode diagram of a PLL with a simple RC filter.

Though this filter is simple, it does not allow independent optimization of the bandwidth and damping factor of the PLL. Reducing, for instance, the bandwidth (ω_{-3dB}) in order to reduce the noise of the output signal requires a reduction of the damping factor (ξ) and hence compromising the stability of the PLL.

3.3.3 Charge pump PLL (CP-PLL) with a simple RC filter

In a previous section, a PLL without a charge pump has been studied to determine its principal parameters such as the damping factor (ξ), and the natural angular frequency (ω_n) in terms of its design parameters (k) and (τ) as illustrated in Eq. 3.19. Bode stability analysis has been also performed on the same PLL and an analytical expression for its bandwidth (ω_{3dB}) and its phase margin (ϕ_{margin}) in terms of the damping factor (ξ) have been found. In this section, a charge pump PLL with a simple RC filter will be studied and compared to its counterpart without the charge pump. Let us consider the simple RC filter with a charge pump of current (I_p) as shown in figure 3-5.
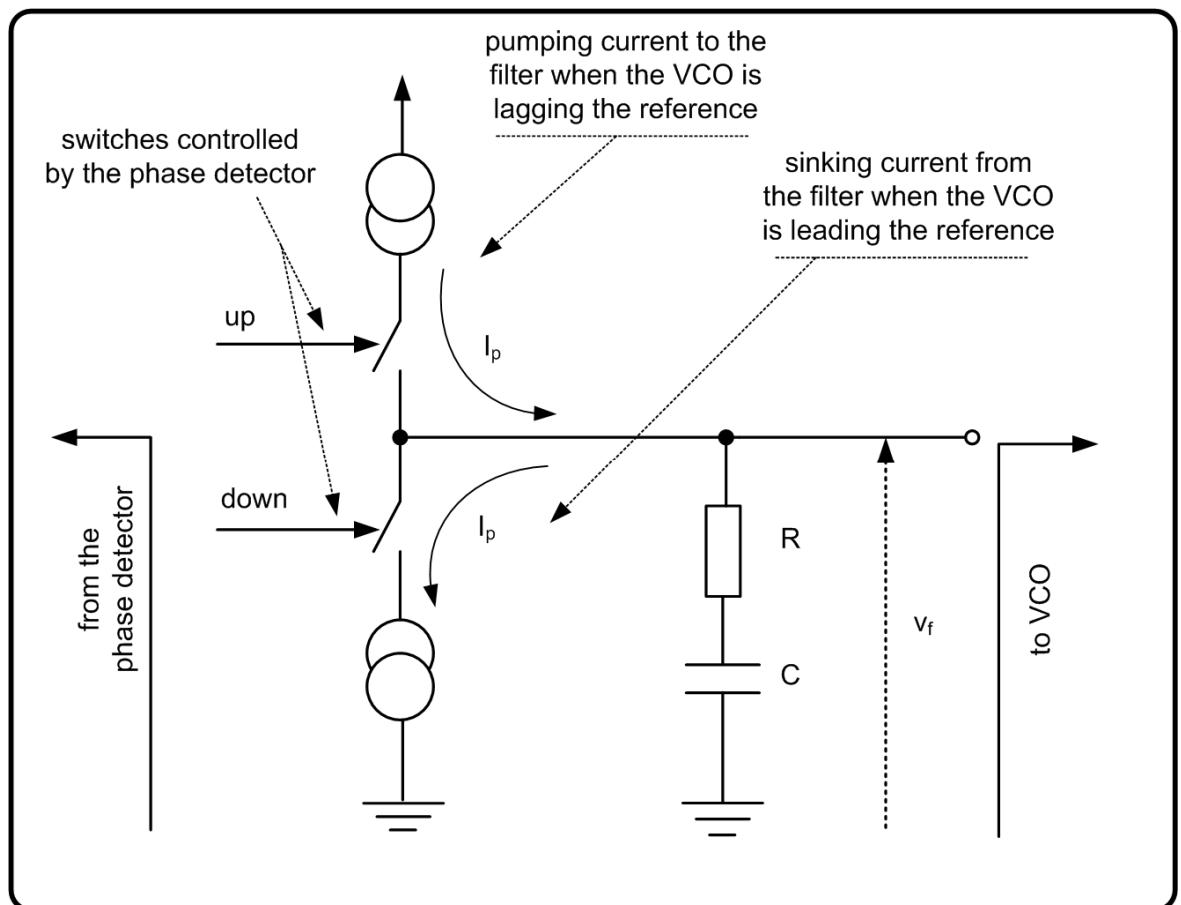
Figure 3-5: A simple RC filter with a charge pump.

With this type of circuit, the linear expression of the phase detector will be modified by incorporating current flowing into or from the filter. Based on the first expression of Eq. 3.10 and figure 3-5, one can write:

$$i_d(t) = \pm I_p \cdot \frac{\theta_e(t)}{2\pi} \tag{3.24}$$

Where ($i_d(t)$) is the current that has been delivered (pumping) or taken from (sinking) the filter in response to a phase error ($\theta_e(t)$). The sign in the last expression represent the polarity of the frequency difference-being positive or negative depending on the difference between the reference and the VCO signals. Considering the filter of the figure 3-5, the voltage at its output can be written as follow:

$$v_f(t) = i_d(t) \cdot Z_{filter} \tag{3.25}$$

Where, (Z_{filter}) is the total impedance of the filter. Taking the Laplace transform of the equation 3.25, we obtain the following equation:

$$V_f(s) = I_d(s) \cdot Z_{filter} = I_p \frac{\Theta_e(s)}{2\pi} \cdot (R + \frac{1}{sC}) = \frac{I_p}{2\pi} \cdot (\frac{RCs+1}{sC}) \cdot \Theta_e(s) \tag{3.26}$$

Thus, the transfer function of the combined phase detector and the charge pump filter blocks will be:

$$TF_{pd \,\&\, filter} = \frac{V_f(s)}{\Theta_e(s)} = \frac{I_p}{2\pi} \cdot (\frac{RCs+1}{sC}) = \frac{I_p}{2\pi} \cdot [\frac{\tau s+1}{s(\tau/R)}] = \frac{I_p R}{2\pi} \cdot \frac{\tau s+1}{\tau s} \tag{3.27}$$

The transfer function for the VCO is unchanged to that of the previous section, we rewrite it for convenience:

$$TF_{vco}(s) = \frac{\Theta_{vco}(s)}{V_f(s)} = \frac{k_{vco}}{s}$$

Based on the previous development, one can redraw the frequency domain block diagram of the CP-PLL
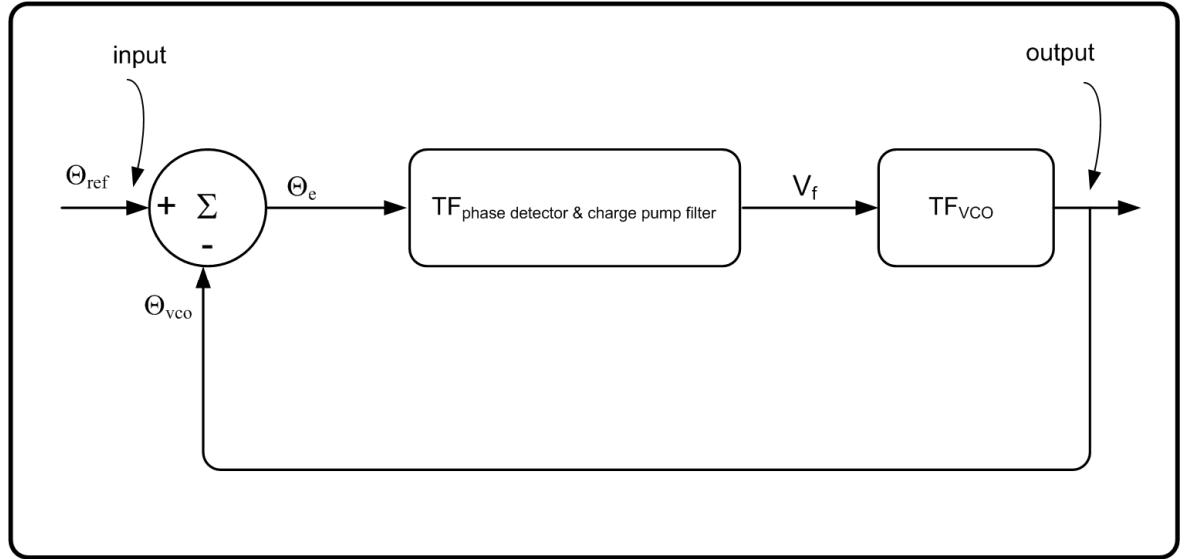
Figure 3-6: Frequency domain block diagram of the charge pump PLL.

The open *G(s)* and closed loop *H(s)* transfer functions of the PLL will be giving as follow:

$$G(j\omega) = TF_{pd+cp} \cdot TF_{vco} = \frac{I_p R}{2\pi} \cdot \frac{\tau s + 1}{\tau s} \cdot \frac{k_{vco}}{s} = k \cdot \frac{\tau s + 1}{\tau s^2} \tag{3.28}$$

$$H(s) = \frac{k/\tau(1+s\tau)}{s^2 + ks + k/\tau} \tag{3.29}$$

Where $k = k_{vco} \dfrac{I_p}{2\pi} R$, setting $k = 2\xi\omega_n$ *and* $\dfrac{k}{\tau} = \omega_n^2$

Equations 3.28 and 3.29 become:

$$G(s) = k\frac{(k/\omega_n^2)s+1}{(k/\omega_n^2)s^2} = \frac{ks+\omega_n^2}{s^2} \tag{3.30}$$

$$H(s) = \frac{\omega_n^2 + 2\xi\omega_n s}{s^2 + 2\xi\omega_n s + \omega_n^2} \tag{3.31}$$

3.3.4 Bode stability analysis of the charge pump PLL

Substituting s by ($j\omega$) in the open loop transfer function $G(s)$ of the charge pump PLL (CP-PLL), the magnitude of that function will be giving as follow:

$$|G(j\omega)| = \sqrt{\frac{k^2\omega^2 + \omega_n^4}{\omega^4}} \qquad (3.32)$$

Solving the equation $|G(j\omega)|=1$ with respect to (ω) give us the cut-off frequency (ω_{3dB}) of the CP-PLL.

$$\omega_{-3dB} = \omega_n \sqrt{\sqrt{1 + 4\zeta^4} + 2\xi^2} \qquad (3.33)$$

The phase margin of the function $G(s)$ is giving as follow:

$$\phi_{m\arg in} = \arctan\{2\xi\sqrt{\sqrt{1 + 4\zeta^4} + 2\xi^2}\} \qquad (3.34)$$

For convenience we rewrite the equations describing the main characteristics of the CP-PLL in term of the design parameters R, C and I_p, and in term of the stability parameter such as the damping factor ξ and the natural frequency (ω_n). The open $G(s)$ and closed loop $H(s)$ transfer functions, the cut-off frequency (ω_{3dB}) and the phase margin (ϕ_{margin}) are giving respectively by the following relationships:

$$G(s) = k\frac{ks + \omega_n^2}{s^2}, \quad H(s) = \frac{\omega_n^2 + 2\xi\omega_n s}{s^2 + 2\xi\omega_n s + \omega_n^2} \qquad (3.35)$$

$$\omega_{-3dB} = \omega_n\sqrt{\sqrt{1 + 4\zeta^4} + 2\xi^2}, \quad \phi_{m\arg in} = \arctan\{2\xi\sqrt{\sqrt{1 + 4\zeta^4} + 2\xi^2}\}$$

$$\text{Where, } k = k_{vco}\frac{I_p}{2\pi}R, \quad k = 2\xi\omega_n, \quad \frac{k}{\tau} = \omega_n^2 \text{ and } \tau = RC$$
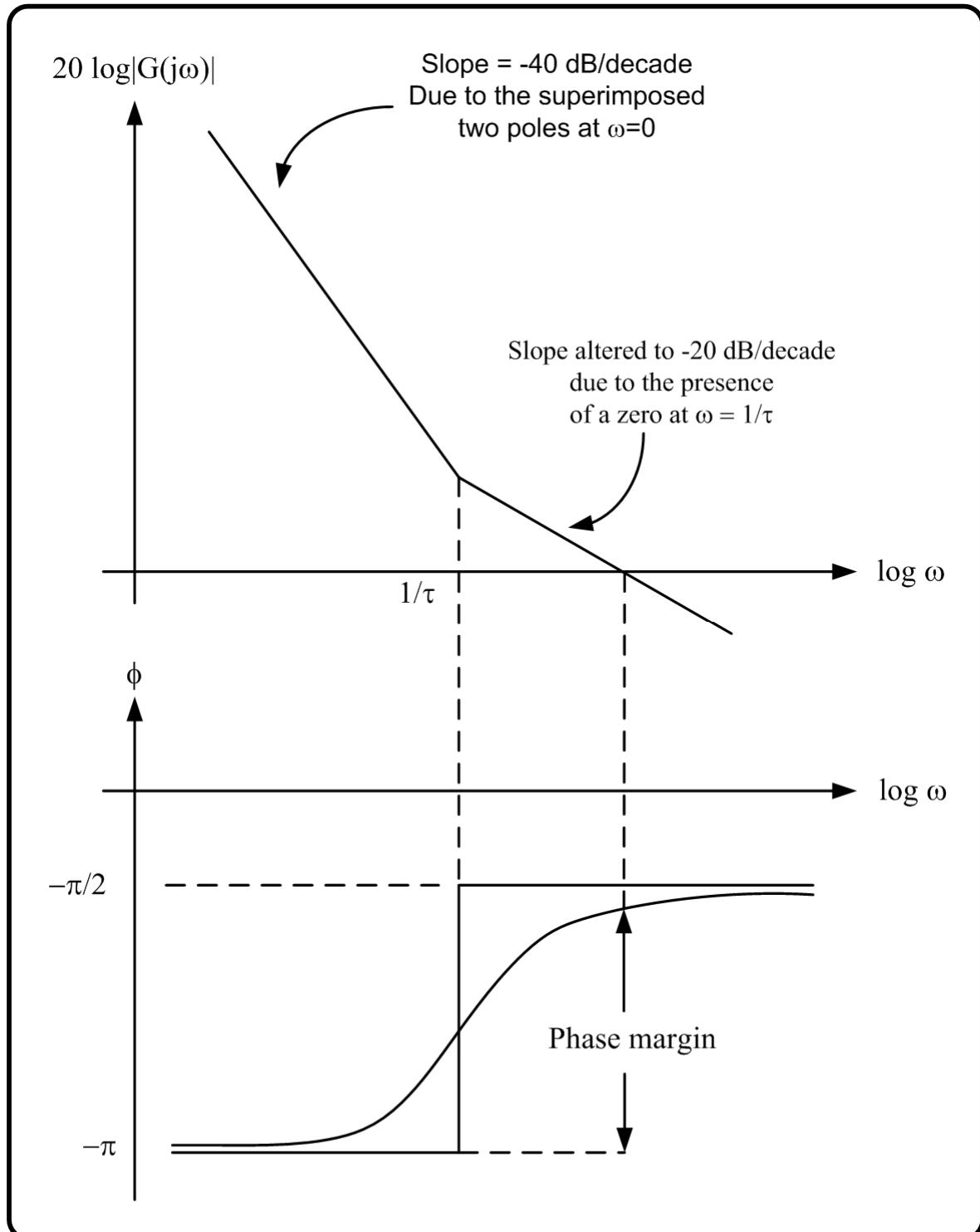
Figure 3-7: Bode diagram of the CP-PLL with a simple RC filter.

3.4 Phase Noise and Jitter in PLL-Based CDR Circuits

The design of reliable communication circuits and systems normally concerns the reduction of phase noise and jitter. These two undesirable effects are closely related, and sought to be considered in the context of oscillators and PLL's.

3.4.1 Oscillator Phase Noise

In order to study and estimate the impact of phase noise on an oscillator's output, let us consider, for instance an ideal oscillator producing a sinusoidal signal at frequency $\omega_0 = 2\pi f_0 = 2\pi/T_0$. Its output waveform can be expressed as $V_{out}(t) = V_0 \cos\omega_0 t$ and its frequency spectrum-as illustrated in Fig. 3-8(a)-consists of two impulses at $\omega = \pm\omega_0$. Since this sinusoid is an ideal one, its zero-crossing points occur at integer multiples of (T_0). Also, the spectrum indicates that the signal carries no energy at any frequency other than (ω_0).
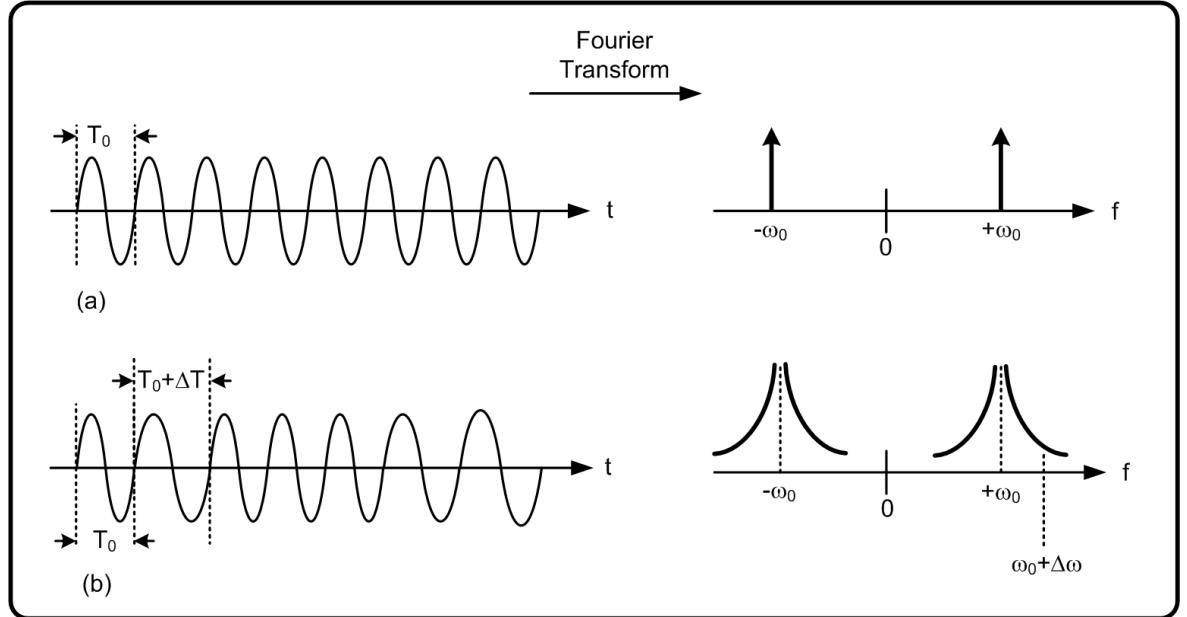
Figure 3-8: (a) Spectrum of a noiseless sinusoid, and (b) noisy sinusoid.

In real oscillator, its internal devices and the circuits surrounding it will randomly vary its oscillation period -as if the oscillator occasionally operates at frequencies other than ω_0- as shown in Fig. 3-8(b). In this case, the zero-crossing points do not necessarily occur at

integers multiples of (T_0) and the output spectrum spreads out around the peaks, revealing that the signal carries finite energy at ($\omega_0 + \Delta\omega$).

In order to find a mathematical expression for this phase noise, we suppose that the amplitude of the output signal is constant and unaffected by the noise. Since the instantaneous frequency varies randomly, the oscillator signal can be written as:

$$V_{out}(t) = V_0 \cos[\omega_0 t + \phi_n(t)] \tag{3.36}$$

Where $\phi_n(t)$ is a small random phase component with zero average. Thus, the zero crossing points of the signal $V_{out}(t)$ occur randomly because they appear at instants given as:

$$t = \frac{k(\pi/2) - \phi_n(t)}{\omega_0} \tag{3.37}$$

Where (k) is an odd number. Equivalently, the oscillation period varies from one cycle to the next. The frequency spectrum of the signal $\phi_n(t)$ is called the phase noise and is denoted by $S_{\phi n}$. Since $\phi_n(t)$ is typically very small, we therefore can assume:

$$\phi_n(t) << 1 \; rad \,, \; then \; \cos\phi_n \approx 1 \; and \; \sin\phi_n \approx \phi_n \tag{3.38}$$

Thus, simplifying Eq. (3.36) to

$$\begin{aligned} V_{out}(t) &= V_0 \cos\omega_0 t.\cos[\phi_n(t)] - V_0 \sin\omega_0 t.\sin[\phi_n(t)] \\ &\approx V_0 \cos\omega_0 t - V_0\phi_n(t)\sin\omega_0 t \end{aligned} \tag{3.39}$$

Eq. 3.39 dictates that the spectrum of V_{out} consists of impulses at $\omega = \pm\omega_0$ and that the spectrum of $\phi_n(t)$ translated to $\pm\omega_0$ as illustrated in Fig. 3-8(b). To quantify the phase noise $S_{\phi n}$, we measure the average power carried in $\Delta f = 1$ Hz in the phase noise area of Fig. 3.8(b). Since the intensity of ϕ_n is frequency dependant, the power must be measured at a consistent specified frequency offset ($\Delta\omega$) from (ω_0) as shown in Fig. 3-9. Also, the measured power in 1 Hz at $\Delta\omega$ must be normalized to the carrier power, P_c (i.e. the power

carried by the impulses at ω_0), this normalization allows comparison between different oscillators. Based on Eq. 3.39, the carrier power is equal to $V_{rms}^2 = V_0^2 / 2$.
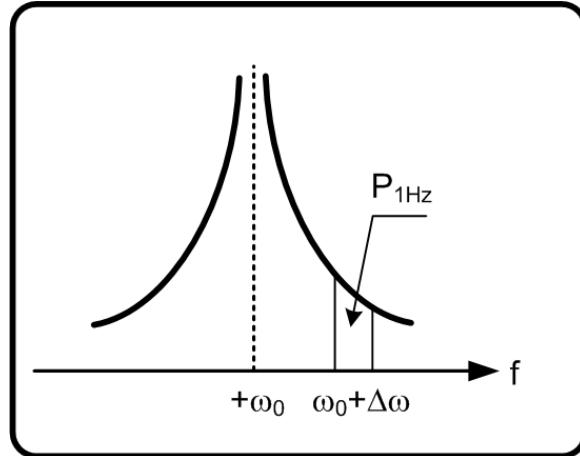
Figure 3-9: Illustration of phase noise.

The relative phase noise will be defined as

$$\operatorname{Re}lative\ Phase\ Noise\ |_{\Delta\omega} = 10\log\frac{P_{1Hz}\ |_{\Delta\omega}}{P_c}\ dBc\,/\,Hz \qquad (3.40)$$

Where the unit *dBc/Hz* denotes the decibels with respect to the carrier emphasizing the normalization [29]. As an example, suppose the phase noise spectrum of an oscillator is giving by

$$P_{1Hz}\ |_{\Delta\omega} = \frac{(50\ mV_{rms})^2}{\Delta\omega^2}$$

If the oscillation amplitude is equal to 0.5 V_{rms}, the relative phase noise at 100 kHz offset will be

$$S_{\phi n}(2\pi\times100\ kHz) = 6.333\times10^{-15}\ V^2\,/\,Hz$$

Normalizing this value to the carrier power, $(0.5V_{rms})^2$, we obtain

$$\frac{S_{\phi n}(2\pi \times 100 \; kHz)}{(0.5 \; V_{rms})^2} = 2.533 \times 10^{-14} \; . \; \text{Thus,}$$

$$\text{Relative Phase Noise} = 10 \log(2.533 \times 10^{-14}) \approx -136 \; dBc \, / \, Hz$$

3.4.2 Oscillator Jitter

The signal jitter is defined as the deviation of the zero crossings from their ideal position in time, or alternatively could be defined as the deviation of each period from the ideal value. Consider a noisy oscillator operating at a nominal frequency $\omega_0 = 2\pi f_0 = 2\pi/T_0$ with its output compared against an ideal square wave with period T_0 [Fig. 3.10(a)].

To estimate the jitter, we measure the deviation of each positive (or negative) transition point of $x_2(t)$ from its corresponding point in the ideal signal $x_1(t)$, i.e., ΔT_1, ΔT_2, ..., ΔT_N. This type of jitter is called "absolute jitter" because it results from comparison with an ideal reference. Since the measured deviations are random, we therefore measure a very large number of deviations (i.e. ΔT) and evaluate the root mean square value of absolute jitter as:

$$\Delta T_{rms}^{abs} = \lim_{N \to \infty}[\frac{1}{N}\sqrt{\Delta T_1^2 + \Delta T_1^2 + ... + \Delta T_N^2}] \qquad (3.41)$$

Another type of jitter which does not require a reference signal and is called "cycle-to-cycle" jitter. It is obtained by measuring the difference between each two consecutive cycles of the waveform, and taking the root mean square of the values [Fig. 3.10(b)]:
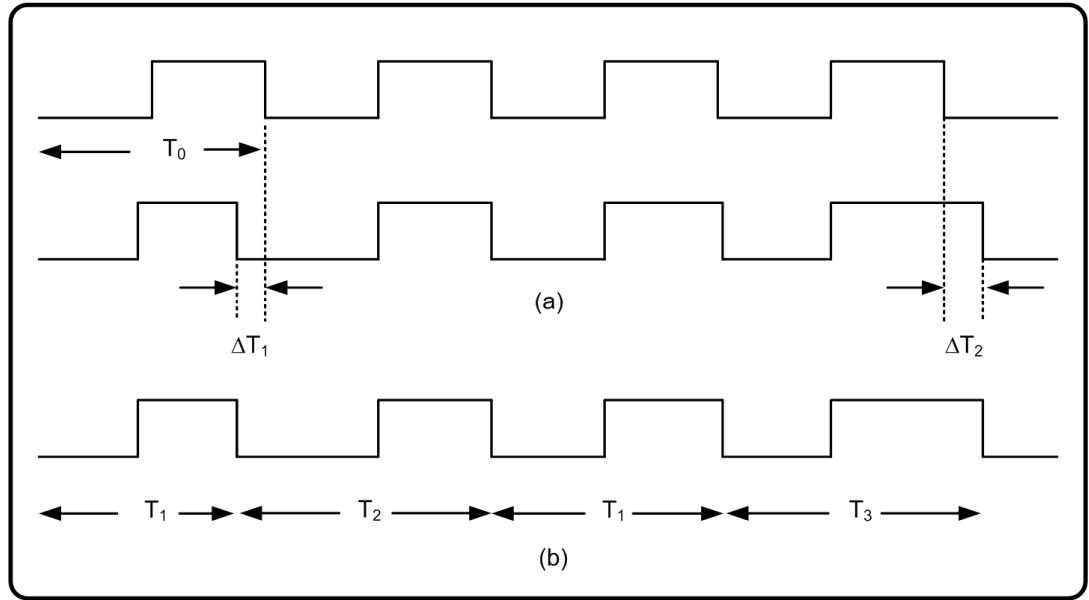
Figure 3-10: (a) Cycle-to-cycle jitter, and (b) variable cycles.

$$\Delta T_{rms}^{cc} = \lim_{N \to \infty}[\frac{1}{N}\sqrt{(T_2 - T_1)^2 + (T_3 - T_2)^2 + ... + (T_N - T_{N-1})^2}\,] \qquad (3.42)$$

Absolute and cycle-to-cycle jitters are generally used to characterize the quality of signals in time domain. A third type of jitter called "period jitter" is defined as the deviation of each cycle from the average period of the signal, $\overline{T}$:

$$\Delta T_{rms}^{p} = \lim_{N \to \infty}[\frac{1}{N}\sqrt{(\overline{T} - T_1)^2 + (\overline{T} - T_2)^2 + ... + (\overline{T} - T_N)^2}\,] \qquad (3.43)$$

3.4.3 Relationship Between Oscillator Phase Noise and Jitter

The oscillator phase noise can be more easily simulated and measured in the laboratory compared to the jitter. It is therefore desirable to establish a relationship between the two quantities. For absolute jitter, a comparison between the actual signal and an ideal reference is required noting that the deviation of each zero crossing is $\Delta T_j = (2\pi/T_0)\phi_{n,j}$, where $\phi_{n,j}$ denotes the value of ϕ_n in radians of the zero crossing number (j). Thus,

$$\Delta T_{abs,rms}^2 = \lim_{N \to \infty} \frac{1}{N}\sum_{j=1}^{N}\Delta T_j^2 = (\tfrac{2\pi}{T_0})^2 \lim_{N \to \infty} \sum_{j=1}^{N}\phi_{n,j}^2$$

The summation can be approximated by an integral:

$$\Delta T^2_{abs,rms} = (\tfrac{2\pi}{T_0})^2 \lim_{T\to\infty} \frac{1}{T} \int_{-T/2}^{+T/2} \phi_n^2(t)dt$$

The limit represents the average power of ϕ_n and, from Parceval's theorem [29], is equivalent to the area under the spectrum of ϕ_n:

$$\Delta T^2_{abs,rms} = (\tfrac{2\pi}{T_0})^2 \int_{-\infty}^{+\infty} S_{\phi n}(f)df \qquad\qquad (3.44)$$

3.5 Jitter in CP-PLL Based CDR Circuits

CDR circuits used for wireline, optical and other communication systems must satisfy certain jitter criteria specified by the standard associated to a particular type of communication system. In this section, a description and estimation of the CDR jitter characteristics will be carried out. The main CDR jitter characteristics are, jitter transfer, jitter generation and jitter tolerance. Each type of jitter will be studied and related through analytical expressions to the PLL parameters ξ, ω_n, ω_{3dB}, R, C, and I_p.

3.5.1 Jitter Transfer

The jitter transfer function of a CDR circuit represents the output jitter as a function of the input one, when the input jitter is varied at different rates. If, for example, the input jitter varies slowly and therefore the waveform zero-crossing points move slowly around their ideal positions then the output can follow the input to ensure phase locking. On the other hand, if the input jitter varies rapidly, the CDR circuit must filter the jitter, i.e., the output tracks the input to a lesser extent. Thus, the jitter transfer exhibits a low-pass characteristic, as in the case of the PLL. The jitter transfers required by communication standards must generally meet difficult specifications. First, the CDR bandwidth should be small enough to attenuate jitter components above the CDR bandwidth. Second, the amount of peaking in the jitter transfer (jitter peaking) must be also small to avoid any eventual instability. Reducing the CDR bandwidth requires a reduction its CP-PLL bandwidth (ω_{3dB}), giving as (Eq. 3.35)

$$\omega_{-3dB} = \omega_n \sqrt{\sqrt{1+4\zeta^4} + 2\xi^2} \text{ , Where, } k = k_{vco}\frac{I_p}{2\pi}R, \quad k = 2\xi\omega_n, \quad \frac{k}{\tau} = \omega_n^2 \text{ and } \tau = RC$$

$$\text{Or, } \omega_n = \sqrt{I_p k_{vco}/(2\pi C)} \text{ and, } \xi = (R/2)\sqrt{I_p C k_{vco}/(2\pi)} \qquad (3.45)$$

To reduce (ω_{3dB}), either (ξ) or (ω_n) must be reduced. However, loop stability requires that ξ stays higher than 0.707, leaving (ω_n) as the only parameter which may be reduced. Lowering (k_{vco}) and (I_p) will reduce (ω_n), but will also reduce (ξ). Thus, C is the principal parameter that can be increased to decrease the loop bandwidth (ω_{3dB}) while increasing the damping factor (ξ).

For, $\xi >> 1$, the CP-PLL bandwidth expression (Eq. 3.45) can be reduced to

$$\omega_{\text{-3dB}} = 2\xi\omega_n = \frac{RI_p k_{vco}}{2\pi}$$

Now, in order to reduce the jitter peaking, the damping factor ξ should have a large value, and careful attention must be paid to the poles and zeros of the closed loop transfer function. The closed loop transfer function of the CP-PLL is giving by (Eq. 3.35)

$$H(s) = \frac{\omega_n^2 + 2\xi\omega_n s}{s^2 + 2\xi\omega_n s + \omega_n^2}$$

Its zero is given by

$$\omega_z = -\frac{\omega_n}{2\xi} = -\frac{1}{RC} \qquad (3.46)$$

And the poles are equal to

$$\omega_{p1,p2} = (-\xi \pm \sqrt{\xi^2 - 1})\omega_n = (-1 \pm \sqrt{1 - \frac{1}{\xi^2}})\xi\omega_n$$

For a large damping factor (i.e. $\xi >> 1$), the square root function can be approximated as

$$\sqrt{1-\varepsilon} \approx 1 - \frac{\varepsilon}{2} - \frac{\varepsilon^2}{8}$$

$$\omega_{p1,p2} = [-1 \pm (1 - \frac{1}{2\xi^2} - \frac{1}{8\xi^4})]\xi\omega_n$$

It follows that
$$\begin{cases} \omega_{p1} = -\frac{\omega_n}{2\xi} - \frac{\omega_n}{8\xi^3} \\ \omega_{p2} = -2\xi\omega_n + \frac{\omega_n}{2\xi} + \frac{\omega_n}{8\xi^3} \end{cases}$$
(3.47)
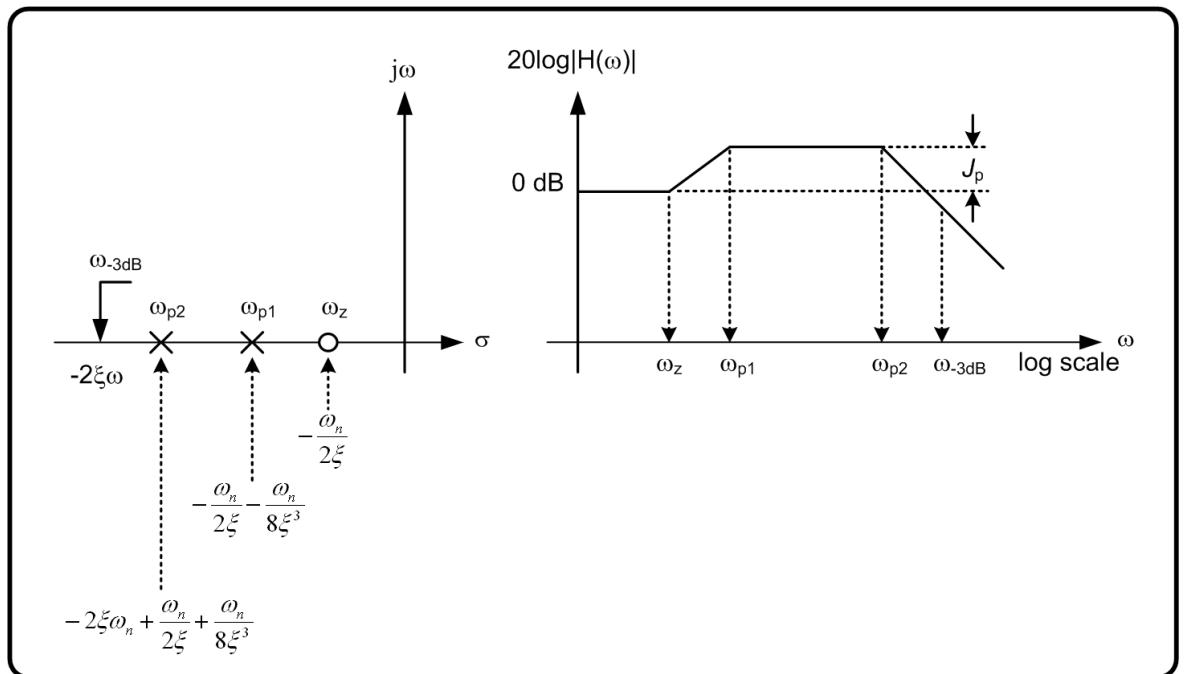
Figure 3-11 (a) Poles and zeros position of the CP-PLL, (b) corresponding jitter transfer function.

The poles and zero are illustrated graphically in Fig. 3.11(a). The expressions 3.46 and 3.47 yield several interesting results. First, the zero appears always before the poles, leading to an inevitable peaking in the jitter transfer function. Second, since the damping factor is large (i.e. $\xi>>1$), the zero and the first pole differ in magnitude only by a small value (i.e. $\omega_n/8\xi^3$). Third, for $\xi>>1$, the pair (ω_z-ω_{p1}) falls well below the second pole (ω_{p2}) because ($\omega_n/2\xi$)$<< 2\xi\omega_n$. Fourth, ω_{p2} is slightly lower than (ω_{-3dB}) by an amount equal to the magnitude of ω_{p1}. Figure 3-11(b) shows a Bode plot of the magnitude of the closed-loop transfer function $H(\omega)$. At $\omega = \omega_z$, the magnitude starts to rise at 20 dB/decade, then it

assumes a constant value for $\omega > \omega_{p1}$, and begins to fall at 20 dB/decade at $\omega > \omega_{p2}$, dropping to -3 dB at $\omega = 2\xi\omega$. With logarithmic scales, the value of jitter peaking (J_p) can be written as [29]

$$20\log J_p = 20\log \omega_{p1} - 20\log \omega_z$$

That is,
$$J_p = \frac{\omega_{p1}}{\omega_z} \approx 1 + \frac{1}{4\xi^2}$$

Expressing the jitter peaking J_p in decibels, we can write

$$20\log J_p = 20\ln J_p \cdot \log_e e = 8.686\ln(1 + \frac{1}{4\xi^2})$$

Which, for $\xi >> 1$. Hence $4\xi^2 >> 1$, can be reduced to

$$20\log J_p \approx \frac{8.686}{\xi^2} = \frac{2.172}{\xi^2} \tag{3.48}$$

Using expression 3.45, Eq. 3.48 can be expressed in terms of the CP-PLL design parameters k_{vco}, R, C, and I_p

$$20\log J_p = \frac{8.686}{R^2 I_p C k_{vco}} \tag{3.49}$$

If the resistor value R is lowered to reduce the jitter bandwidth (ω_{3dB}), then the capacitor value C must be raised substantially to maintain J_p constant.

3.5.2 Jitter Generation

Jitter generation refers to the jitter produced by the CDR circuit itself, when the input random data is jitter free. The source of jitter in CDR circuits can be summarized as follow:

- VCO phase noise due to the electronic noise of its constituent devices

- Ripple on the VCO control voltage issued from the filter

- Coupling of data switching to the VCO through the phase and frequency detectors

- Power supply and substrate noise

To estimate the VCO noise contribution to CDR jitter, an expression relating PLL jitter to the jitter of the free-running VCO must be derived. The phase noise and cycle-to-cycle jitter of the free-running VCO are related by the following equation [45]

$$\Delta T_{cc}^2 \approx \frac{4\pi}{\omega_0^3} S_\phi(\Delta\omega)\,\Delta\omega^2 \tag{3.50}$$

Where ω_0 denotes the oscillation frequency and $S_\phi(\Delta\omega)$ represents the relative phase noise power at an offset frequency ($\Delta\omega$) [45]. The jitter given by Eq. 3.50 will be shaped due to the PLL effect. As illustrated in Fig. 3.12, it can be assumed that for a loop bandwidth of $2\pi f_u$, the jitter rises with the square root of time until the instant $t_1 = 1/2\pi f_u$ and saturates thereafter [46]. The total jitter accumulated over time t_1 by a free-running oscillator is equal to [45]

$$\Delta T_1 = \sqrt{\frac{f_0}{2}}\,\Delta T_{cc}\sqrt{t_1} \tag{3.51}$$

Substituting (3.50) in (3.51) yields the closed loop jitter

$$\Delta T_{PLL} = \frac{1}{2\pi}\sqrt{S_\phi(\Delta\omega)}\,\frac{\Delta\omega}{\omega_0} \tag{3.52}$$

Now, if the value of (ΔT_{PLL}) must be less than 0.25 ps at 40 GHz and $f_u = 20$ MHz, then $S_\phi(\Delta\omega)$ must not exceed -79 dBc/Hz at 1-MHz offset [29].
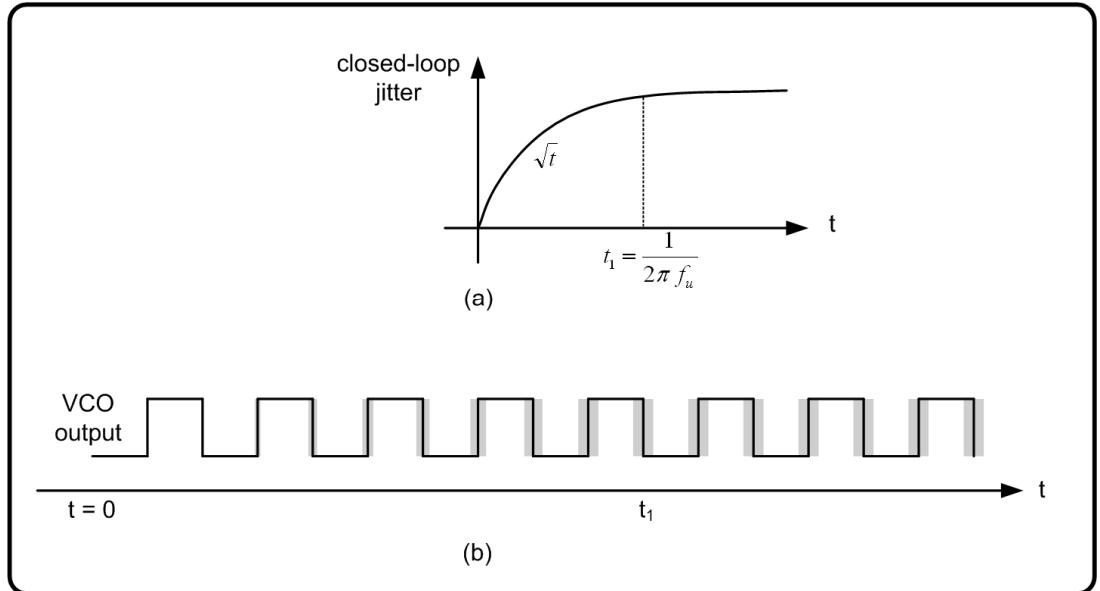
Figure 3-12 Accumulation of cycle-to-cycle jitter in a phase-locked oscillator: (a) actual behavior and (b) resultant waveform.

Another jitter source in CDR circuits is the ripple on the control voltage. Any mismatch in the charge pump design circuit can lead to a net charge injection into the loop filter on every phase comparison instant even if the loop is locked, hence modulating the VCO control voltage and generating jitter. Random data transitions can also generate ripple on the VCO control voltage through the phase and frequency detector. Thus, modulation resulting from the ripple may be significant thus yielding large jitter.

For a simple case of periodic modulation of a VCO, it is possible to estimate the output jitter. Assuming a sinusoid modulation $V_m cos\omega_m t$, the cycle-to-cycle jitter is giving by [45]

$$\Delta T_{cc} = \frac{V_m k_{vco}}{f_0^2}\sqrt{1-\cos\frac{\omega_m}{f_0}} \qquad (3.53)$$

If the modulation frequency is much smaller than the oscillation frequency (i.e. $f_m \ll f_0$), the expression (3.53) can be reduced to

$$\Delta T_{cc} = \frac{V_m k_{vco}\omega_m}{\sqrt{2}f_0^3} \qquad (3.54)$$

3.5.3 Jitter Tolerance

The analysis of jitter properties in the previous sections has been so far focused on their effects on the recovered clock. However, as the data stream at the CDR circuit output will be used for further processing, hence, the retimed data quality is also important. The CDR circuit should normally behave with respect to the jittered input data stream as follow:

- For slowly varying jitter at the input, the recovered clock usually tracks the phase variations, always sampling the data in the middle of the bit [Fig. 3-13(a)] and guaranteeing a low bit error rate (BER).

- For rapidly varying jitter, the clock cannot completely track the input phase variations, failing to sample the data optimally [Fig. 3-13(b)] and hence creating a large BER.
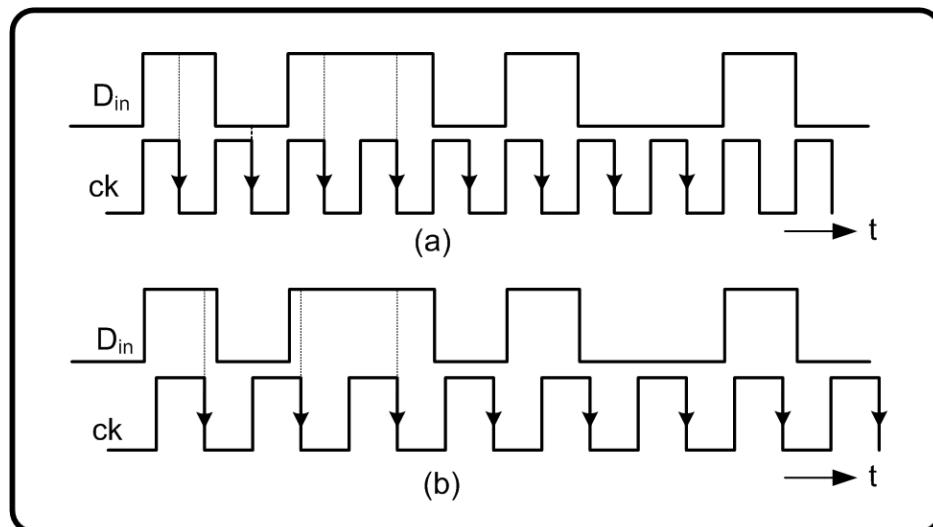
Figure 3-13: Effect of (a) slow and (b) fast jitter on data retiming.

The above two properties are natural for the CDR circuits but they must still conform to certain requirements posed by the communication standards. Communication standards often express jitter in terms of the bit period, also called the unit interval (UI). For example, a jitter of 0.01 UI (10 mUI) refers to 1% of the bit period. The jitter tolerance is defined as how much input jitter a CDR circuit must tolerate without increasing the BER. As illustrated in Fig. 3-14, the specification is typically described by a mask as a function of the jitter frequency. For example, the CDR circuit must withstand a peak-to-peak jitter of 15 UI if the jitter varies at a rate below 100 Hz. The tolerance test is normally performed

with a random bit sequence whose phase is modulated at different rates for different parts of the mask.
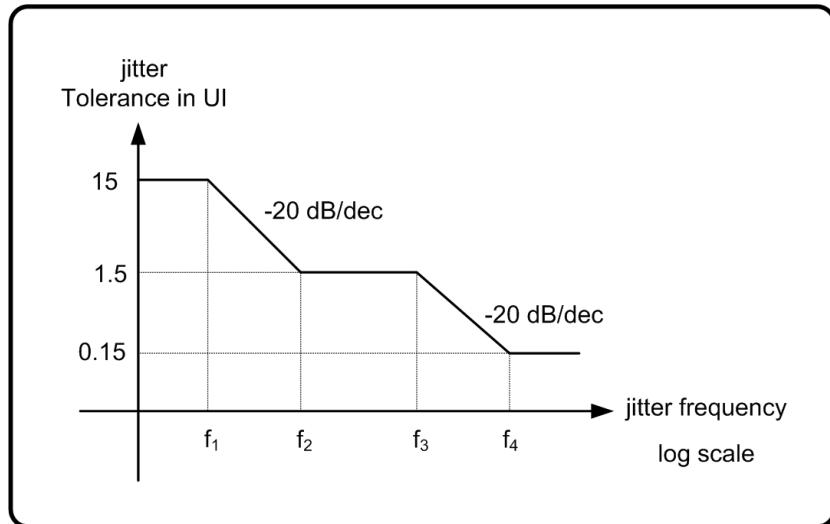
Figure 3-14: Example of jitter tolerance mask.

In the next section, the jitter tolerance of a typical CDR circuit will be quantified and compared with the mask shown in Fig. 3-14. At, a given jitter frequency, the magnitude of the input phase ϕ_{in} must be increased until the BER begins to rise. This occurs when the phase error, ϕ_{in}-ϕ_{out}, approaches one-half unit interval, bringing the sampling edge of the clock close to the zero-crossing points of data. Thus, an approximate condition to avoid increasing the BER is

$$\phi_{\text{in}} - \phi_{\text{out}} \langle \frac{1}{2} UI$$

Or, equivalently,
$$\phi_{in}[1 - H(s)] \langle \frac{1}{2} UI$$

Where, $H(s) = \phi_{out}/\phi_{in}$, and hence

$$\phi_{in} \langle \frac{0.5\,UI}{[1 - H(s)]} \qquad (3.55)$$

We, therefore can express the jitter tolerance as

$$G_{JT}(s) \langle \frac{0.5}{[1 - H(s)]} \qquad (3.56)$$

Where, $G_{JT}(s)$ denotes he largest phase modulation at the input that increases the BER negligibly. For a CDR loop based on the CP-PLL studied in the previous section, hence

$$G_{JT}(s) = \frac{1}{2} \frac{s^2 + 2\xi\omega_n s + \omega_n^2}{s^2} \qquad (3.57)$$

Where, the closed loop transfer function $H(s)$ is giving by (Eq. 3.35)

$$H(s) = \frac{\omega_n^2 + 2\xi\omega_n S}{s^2 + 2\xi\omega_n s + \omega_n^2}$$

The function $G_{JT}(s)$ contains two poles at the origin and two zeros coincident with the poles of $H(s)$. Consequently, as depicted in Fig. 3-15, the Bode plot of the function $|G_{JT}(s)|$ (i.e. $20\log|G_{JT}(s)|$) falls at a rate of 40 dB/decade for $\omega < \omega_{p1}$ and at 20 dB/decade for $\omega_{p1} < \omega < \omega_{p2}$, approaching 0.5 UI for $\omega > \omega_{p2}$.
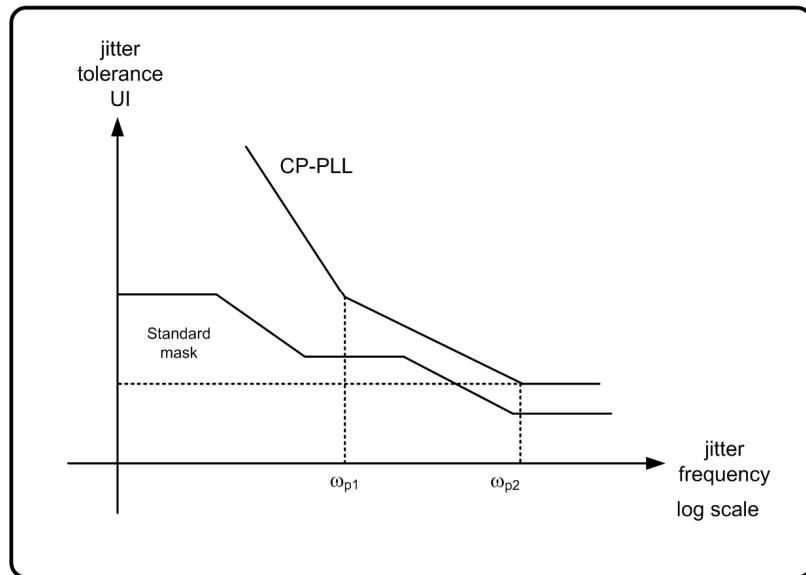
Figure 3-15: Jitter tolerance for CP-PLL.

Some few interesting remarks could be deduced from the previous results. First, the magnitude of the function $G_{JT}(s)$ at $s = j|\omega_{p1}|$ can be calculated as

$$|G_{JT}(s=j\,|\,\omega_{p1}\,|)\,|^2 = \frac{1}{4}\frac{(\omega_n^2 - \omega_{p1}^2)^2 + 4\xi^2\omega_n^2\omega_{p1}^2}{\omega_{p1}^4}$$

But, using Eq. (3.47), we have $\omega_n \approx 2\xi\,|\,\omega_{p1}\,|$ for $\xi \gg 1$ and hence

$$|G_{JT}(s=j\,|\,\omega_{p1}\,|)\,|^2 = 8\xi^4$$

That is,
$$|G_{JT}(s=j\,|\,\omega_{p1}\,|)\,| \approx 2\sqrt{2}\,\xi^2\,UI \qquad\qquad (3.58)$$

Also, Eq. (3.47) yields $|\,\omega_{p2}\,| \approx 2\xi\omega_n$ for $\xi \gg 1$.

Figure 3.16(a) plots $|G_{JT}(s)|$ for two different values of ξ, revealing that if ξ increases and ω_n remains constant, the required jitter tolerance is easily met. Moreover, Fig. 3.16(b) suggests that as ω_n increases while ξ remains constant, jitter tolerance improves.
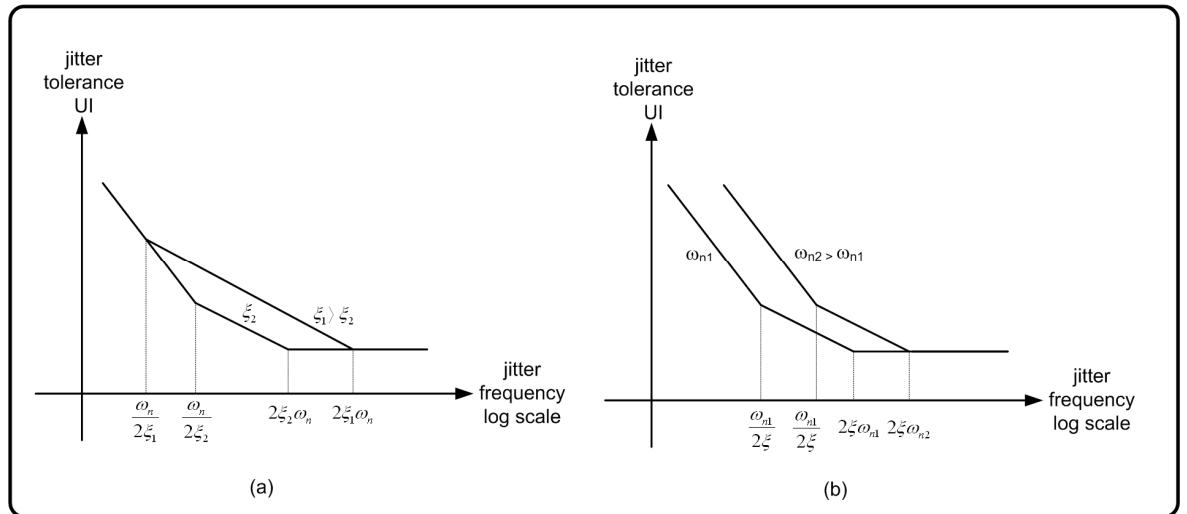
Figure 3-16: Jitter tolerance for different values of (a) ξ and (b) ω_n.

3.5.4 R, C, and I_p Value Optimization Algorithm and Performance Comparison of the PLL and the CP-PLL

After studying the time domain tracking property of the PLL, and the stability analysis of the PLL and the CP-PLL incorporating a simple RC filter, we will now look for the optimized value of R, C and I_p to obtain reasonable value of the loop parameters (ϕ_m, ξ and ω_{-3dB}). Once the optimized value is obtained, a performances comparison of the PLL and the CP-PLL will be carried out. To do this, we will start from initial value of the design parameters R, C, and I_p. The VCO's conversion gain k_{vco} is taken from the transistor level design of the PLL.

Equations for the PLL are:

$$\omega_{-3dB} = \omega_n \sqrt{\sqrt{1 + 4\zeta^4} - 2\xi^2} \ , \ \phi_{m\,arg\,in} = \frac{\pi}{2} - \arctan[\frac{2\xi}{\sqrt{\sqrt{1 + 4\zeta^4} - 2\xi^2}}]$$

Where, $$2 \cdot \xi \cdot \omega_n = \frac{1}{\tau} , \ \omega_n^2 = \frac{k}{\tau} \quad and \quad \xi = \frac{1}{2\sqrt{k \cdot \tau}}$$

And for the CP-PLL are:

$$\omega_{-3dB} = \omega_n \sqrt{\sqrt{1 + 4\zeta^4} + 2\xi^2} \ , \ \phi_{m\,arg\,in} = \arctan\{2\xi\sqrt{\sqrt{1 + 4\zeta^4} + 2\xi^2}$$

Where, $$k = k_{vco}\frac{I_p}{2\pi}R, \quad k = 2\xi\omega_n, \quad \frac{k}{\tau} = \omega_n^2 \ and \ \tau = RC$$

We have, $k_{vco} = 2\pi (1.7x10^9) \ rad/V.sec$. The values of the parameters resulting from the optimization are the following: $R = 370 \ \Omega, C = 2.3 \ nF, I_p = 30 \ \mu A$.

Parameter	ξ	ω_n (rad/sec)	ω_{-3dB} (rad/sec)	ϕ_{margin} (degrees)	Jitter Peaking (dB)	Jitter Tolerance (UI)
PLL	0.0312	18.86×10^6	18.84×10^6	58	2.24×10^3	2.7577×10^{-3}
CP-PLL	2	4.71×10^6	18.84×10^6	86.48	0.543	11.314

Table 3-1: PLL and CP-PLL loop parameters for the optimized value of R, C and Ip.

Table 3-1 shows clearly that the CP-PLL is much better than the PLL in term of damping factor, phase margin, jitter peaking and jitter tolerance.

3.6 Summary

In this chapter, a simplified time-domain analysis of the PLL in the locked state has been carried out illustrating the tracking property of the PLL. In order to properly select the low pass filter components (i.e. *R* and *C*), a frequency-domain stability analysis of the PLL and the CP-PLL has been carried out, this analysis results in analytical expression relating the stability parameters to *R* and *C*. Finally, as the jitter is predominant parameters in the *CDR* circuits, a study of the jitter in the CP-PLL and its relation to *R* and *C* has been carried out.
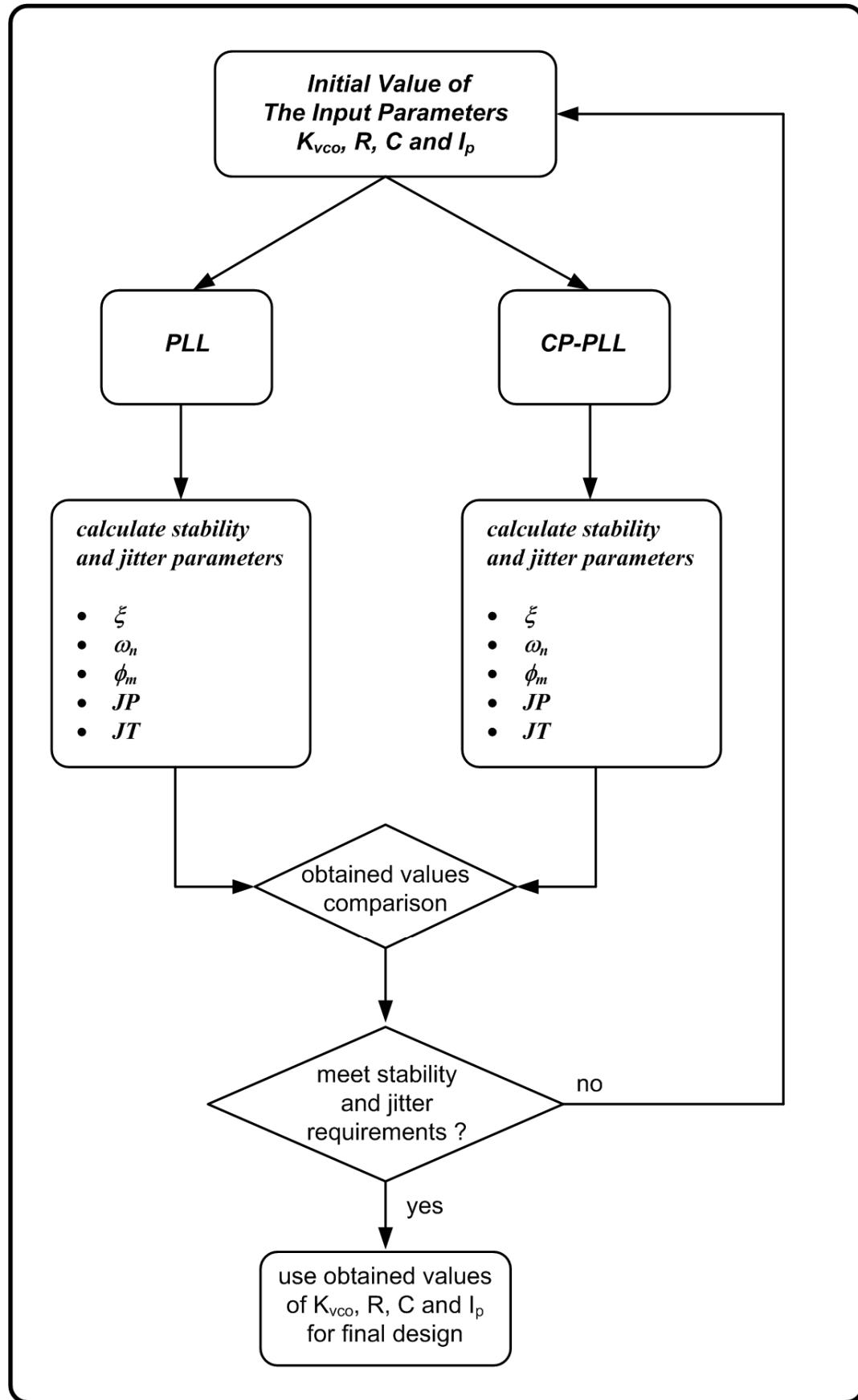
Figure 3 17: Optimization algorithm for selecting the value of R, C, and I$_p$.

4 Inter Chip Communication and Verilog-A System Modelling

The Verilog-A language is relatively new. It is an extension of SPICE; hence they have a compatible simulation environment. In this work, we have adopted an efficient bottom-up extraction approach to build and simulate a gate-level model for the clockless SerDes-based serial link for an asynchronous inter-chip communication system [34, 65, 66, 67]. First, the dynamic (e.g. Latch, DFF, DETFF) and static gates (e.g. AND, OR, XOR) were designed at transistor level using the resistively loaded MOS current mode logic, then the characteristic parameters (e.g. delay, rise and fall time) of those gates were extracted. Finally all the extracted parameters were incorporated into the behavioral model of the reciprocal gates. In order to verify the accuracy of the quarter-rate concept, a 10 Gb/s point-to-point based serial link interfacing two 8 bits chips will be implemented using the Verilog-A language [34]. The proposed serial link will be incorporating the proposed quarter-rate PLL-based CDR circuits. Based on the diagram illustrated on Figure 4-2, the optimization implementation and simulations of this link will be carried out as follow:

1. Optimization, implementation and time domain simulation of the 8-to-1 serializer, the serializer data input is 8 parallel PRBS data streams at 1.25 Gb/s each, its output will be a single data stream at 10 Gb/s (section 4.2.1).

2. Optimization, implementation and time domain simulations of the 1-to-8 deserializer, the deserializer data input is a single 10 Gb/s PRBS data stream, its output will be 8 parallel data streams at 1.25 Gb/s each. The quarter-rate circuit will be incorporated in the deserializer circuit (section 4.2.2).

3. Optimization, implementation and time domain simulations of the complete serial link involving the serializer and the deserializer. The link input at the serializer side will be 8 parallel PRBS data streams at 1.35 Gb/s each, whereas its output at the deserializer should 8 parallel data streams at 1.35 Gb/s each, if the 8 parallel data streams at the input are the same as the 8 parallel data streams at the output, hence the serial link is working properly and therefore the concept of quarter-rate PLL-based CDR is a working one. (section 4.2.3)

4.1 Dedicated Point-to-Point Serial Link

In a serial bus or link, a circuit called SerDes (Serializer/Deserializer) interfacing for example two VLSI chips is used to transmit and receive data over clockless serial link as shown in Figure 4-1.
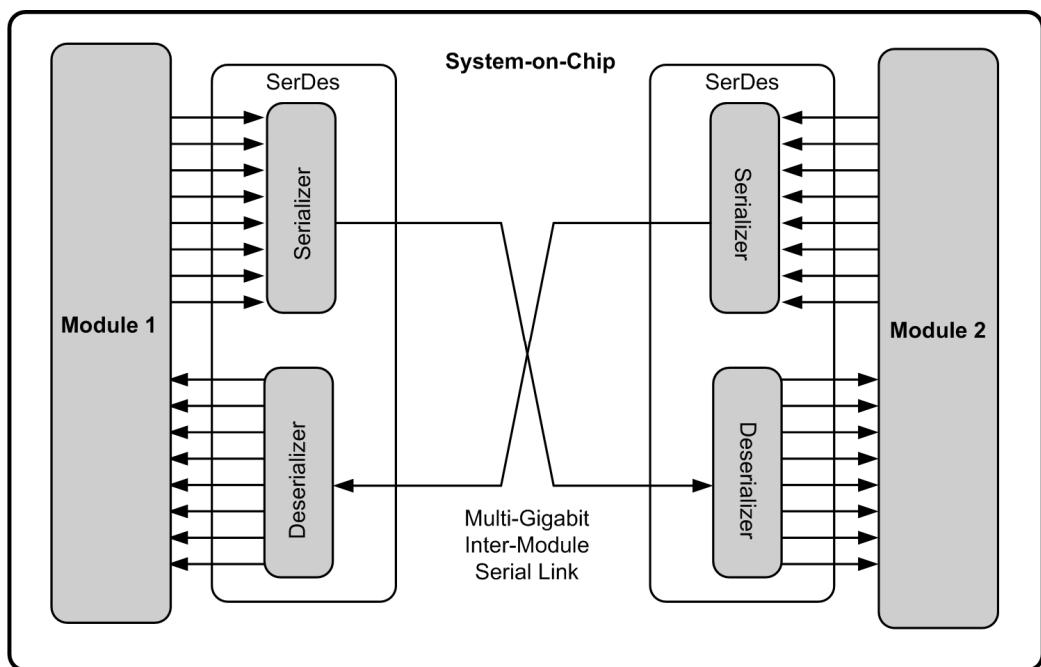
Figure 4-1: SerDes system as used in chip-to-chip serial data communication.

In essence, a SerDes is a serial transceiver which converts parallel data into a serial data stream on the transmitter side and converts the serial data back to parallel on the receiver side. The timing skew problem encountered usually in a parallel bus is eliminated by embedding the clock signal into the data stream. Since there is no separate clock signal in a serial link, timing skew between clock and data no longer exist. As a result, a serial bus or link can usually operate at a much higher data rate than a parallel bus in a comparable system environment. Since the data is sent without the clock signal, one therefore needs a circuit in the receiver to extract that clock signal from the serial data stream itself and sample the last stream using the extracted clock, the clock extraction and data retiming is actually refer to the clock and data recovery (CDR) operation.

4.2 Serializer/Deserializer (SerDes) System

As discussed earlier, a SerDes circuit performs two functions, serialization and deserialization in a lossy and noisy environment. As shown in Figure 4-2, the serializer converts the 8 bits parallel data streams into a single serial data stream. The conversion is done with the clocks generated from the transmitter's clock generator. Usually a high speed clock running at the serial data rate is required. A practical and cost effective solution is to generate this high speed clock from an off-chip low frequency quartz crystal oscillator. As a result, a PLL based frequency multiplier is required in the transmitter side; another important design challenge for the PLL is to maintain a minimum amount of clock jitter despite all the switching noise generated by the surrounding circuits.
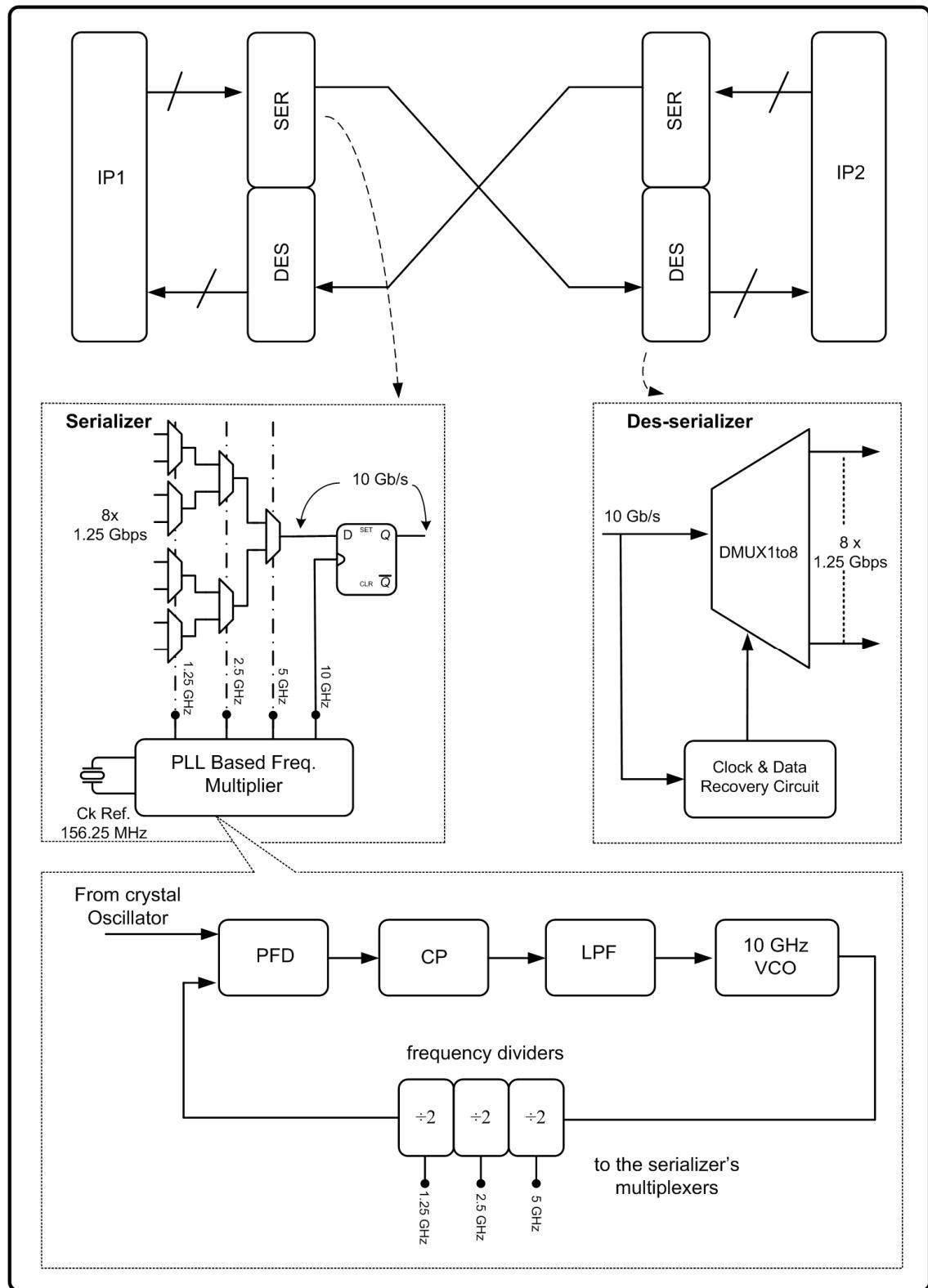
Figure 4-2: Simplified SerDes block diagram.

4.2.1 Serializer Principle and time domain simulations

As illustrated in Figure 4-2 and 4-4, the serializer in this work contains 7 units of 2-to-1 multiplexers and a PLL-based frequency multiplier circuit. A single 2-to-1 multiplexer comprises 5 latches and a selector (Figure 4-3). The lathes guarantee the synchronization of data edges to the clock edges for proper operation of the multiplexer. As shown in the timing diagram of Figure 4-3, a 2-to-1 multiplexer generally combines 2 low-speed (i.e. input bit width is equal to 800 ps) parallel channels into a higher speed stream of serial data (i.e. output bit width is equal to 400 ps).
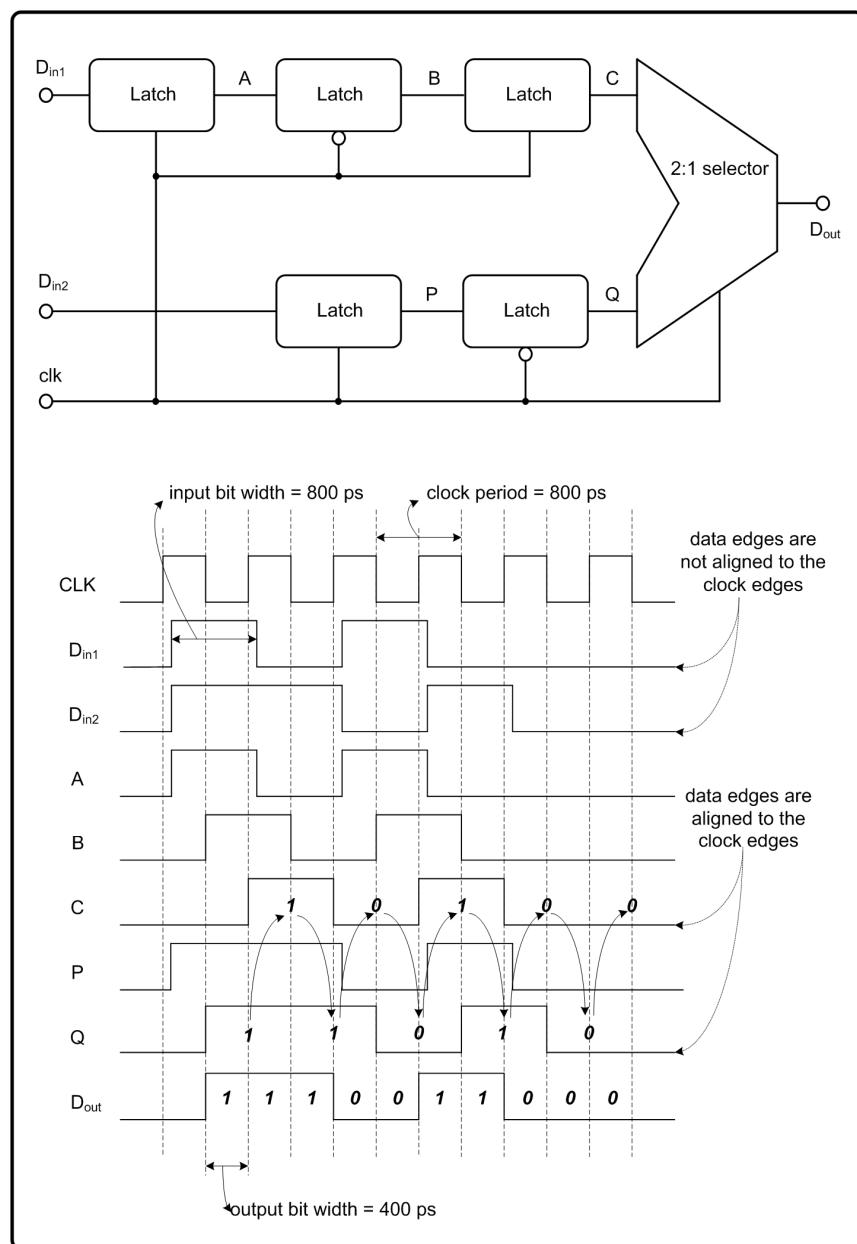
Figure 4-3: A multiplexer (a) and, its timing diagram (b).
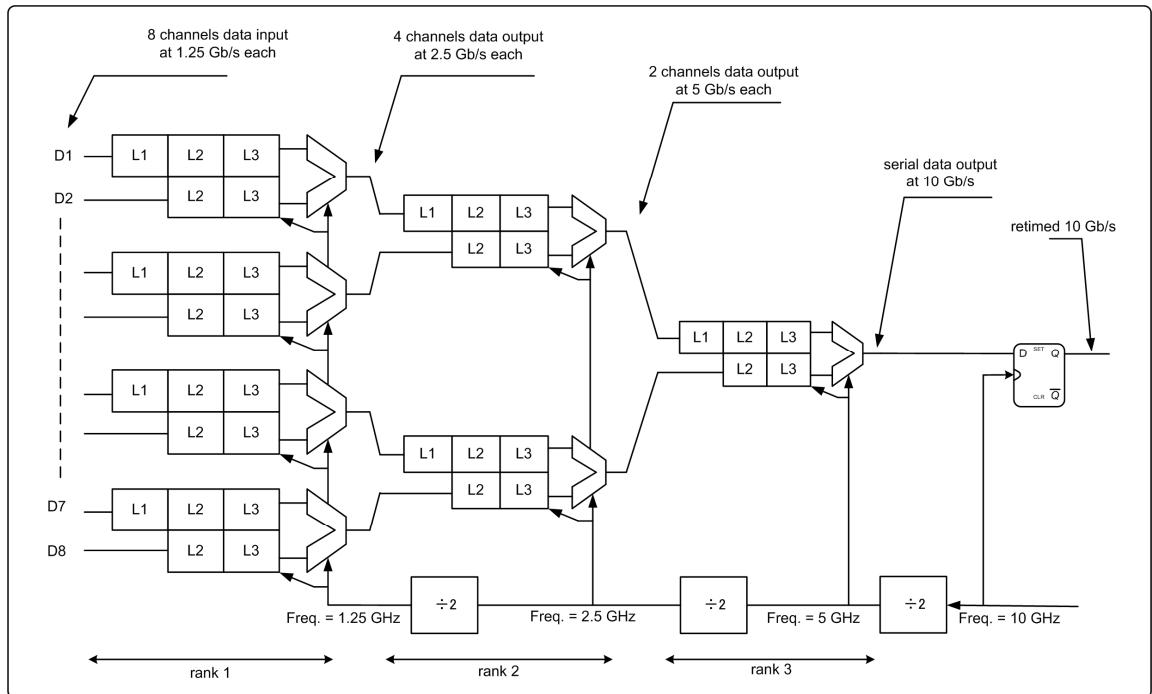
Figure 4-4: A tree architecture of the 8-to-1 serializer.

Based on the 2-to-1 multiplexer developed above, one can use this architecture to implement a multiplexer with a larger number of input channels. Most multiplexers are based on the topology of tree structure. As illustrated in Figure 4-4, the tree structure is a natural extension of the 2-to-1 multiplexer, the idea is to group the input channels in pairs and multiplex each pair, reducing the number by a factor of two after each rank. In this architecture, the flip-flop is driven by a clock frequency fck, the multiplexer in rank 3 is driven by a clock frequency $fck/2$, the multiplexers in rank 2 are driven by a clock frequency $fck/4$, and those in the rank1 are driven by a clock frequency $fck/8$.
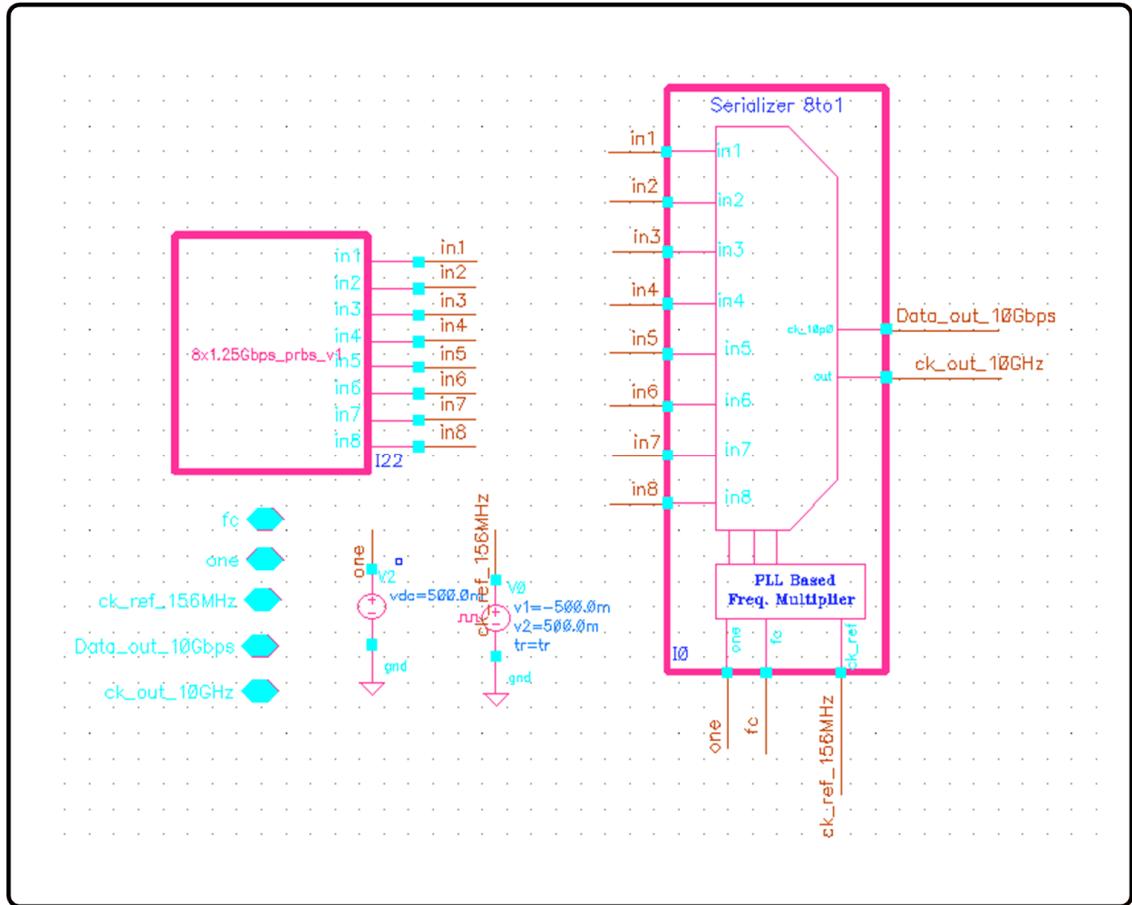
Figure 4-5: Serializer test bench circuit.

The test bench of the serializer circuit is shown in Figure 4-5. The input signal of the serializer circuit is 8 parallels channels of PRBS (pseudo-random bit sequence generator) at 1.25 Gb/s data rate each representing for example the output of an 8-bits microprocessor, this former communicates with a hard-drive disk or a memory. The serializer output should be a single data stream at 8X1.25 Gb/s (=10 Gb/s). The serializer time domain simulations results are shown in Figure 4-6, once the PLL in the serializer reached the steady state, and for data input bits width of 800 ps (red signal in Figure 4-6(a)), the serializer data output bits have a width of 100 ps (blue signal in Figure 4-6(b)), and the clock issued from the PLL in the serializer has a period width of 100 ps (red signal in Figure 4-6(b)) which confirms the operation accuracy of the serializer.
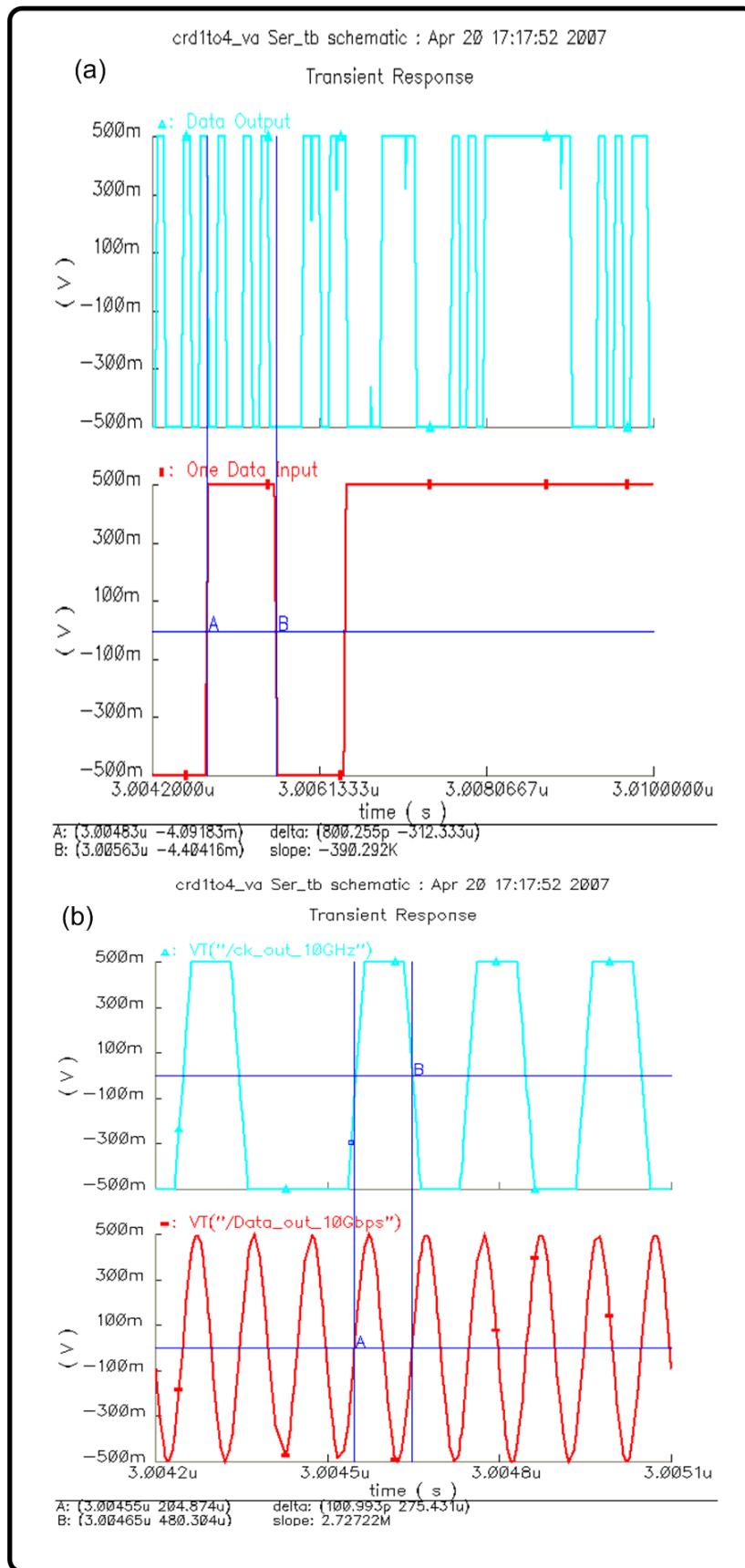
Figure 4-6: Serializer time domain results, data bit input width is 800 ps (a) and, (b) output bit width is 100 ps.

4.2.2 Deserializer Principle and Time Domain Simulations

Since our proposed CDR will automatically demultiplex (1-to-4) the serial data stream, hence an additional demultiplexer (4-to-8) will be required. Figure 4-7(a) show the block diagram of the 4-to-8 de-multiplexer that will normally be driven by the CDR. Figure 4-7(c) show the timing diagram of the 1-to-2 demultiplexer implemented in five-latch architecture (Figure 4-7(b)).
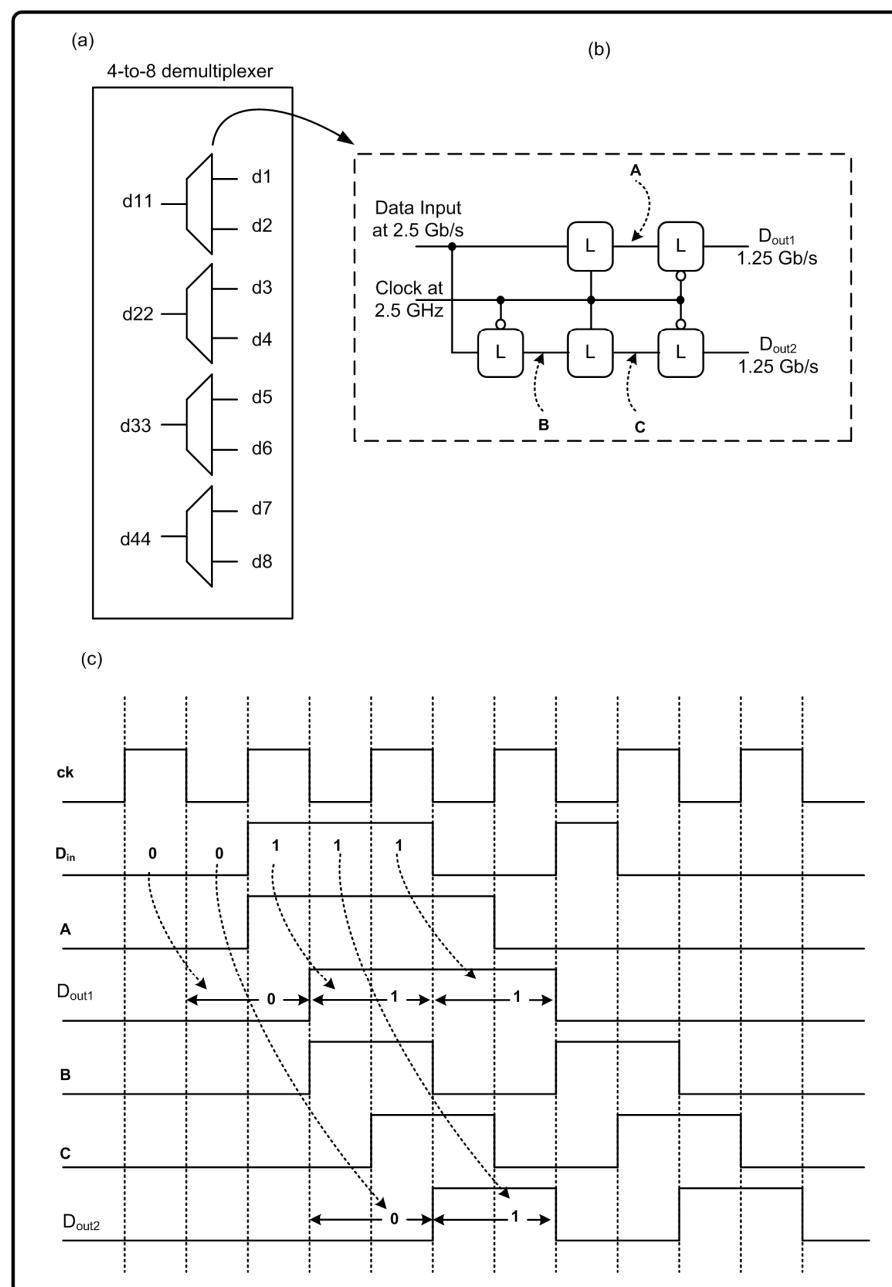
Figure 4-7 Block diagram of the 4-to-8 demultiplexer (a), five-latch architecture of the 1-to2 demultiplexer (b), and timing diagram of the demultiplexer (c).

As shown in Figure 4-8, the deserializer circuit, comprise our proposed quarter-rate PLL based clock and data recovery circuit. Its input is a one PRBS serial data stream at 10 Gb/s data rate without any clock signal associated to it. The VCO frequency in the CDR was 2.45 GHz, which is 50 MHz below the required frequency of 2.5 GHz (i.e. quarter-rate of the data rate). The task of the deserializer is to, extracts the clock signal embedded in the data stream, demultiplexes (1-to-4) the former one and simultaneously retime (sample) them for further processing. In our case an additional demultiplexing (4-to-8) is required in order to compare the 8 inputs of the serializer (section 4.2.1) to the 8 outputs of the deserializer. As shown in Figure 4-9(a,) the PLL in the deserializer reached the steady state within 2.3 μs, and extracted clock has a frequency of 2.5 GHz (Figure 4-9(b)).
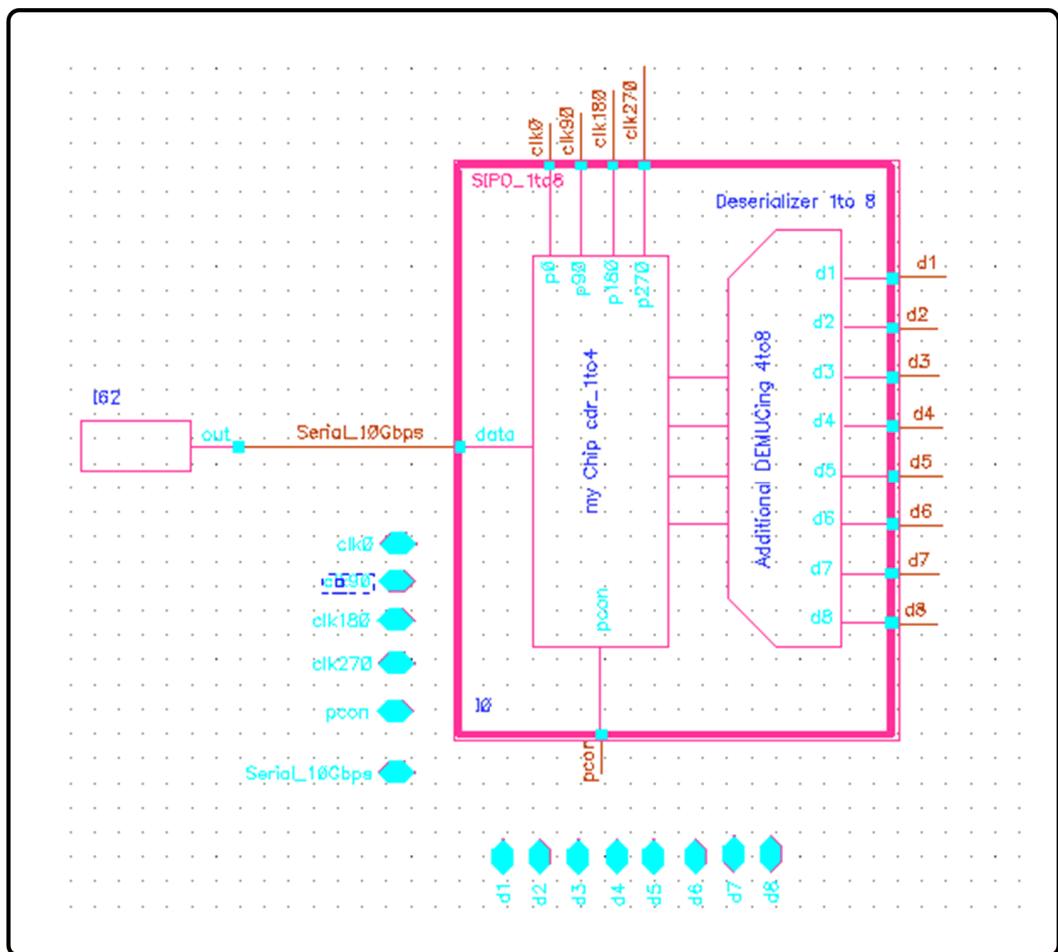
Figure 4-8: Deserializer test bench circuit.
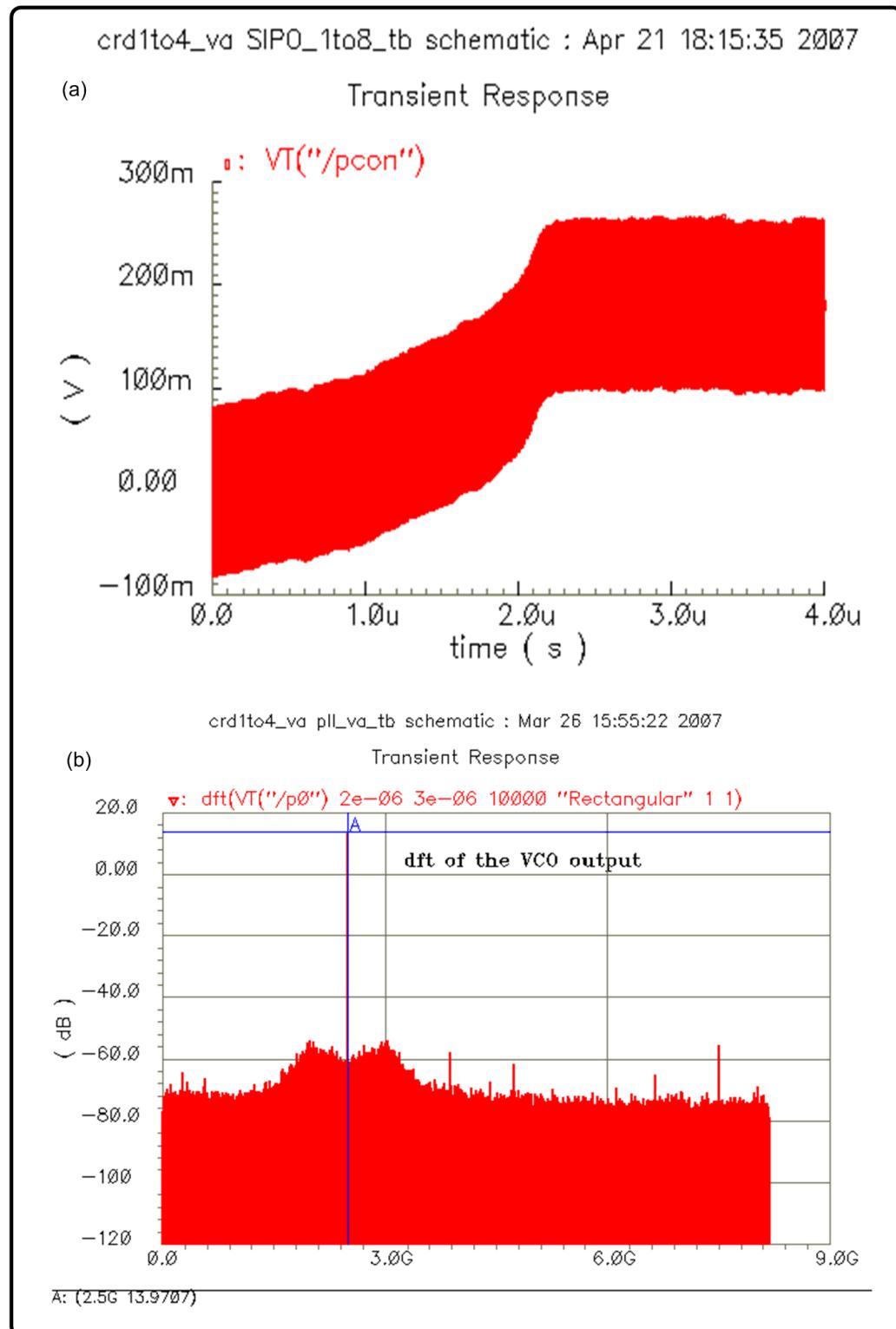
Figure 4-9: Low pass filter output showing the deserializer PLL locking process (a) and, (b) DFT of the quarter-rate recovered clock output signal.

4.2.3 Complete Serial Link (SerDes) Time Domain Simulations

The test bench circuit of the serial link is shown in figure 4-10. This circuit includes 8 PRBS parallel data channels at 1.35 Gb/s each, the PLL-based 8-to-1 serializer and the PLL-based 1-to-8 deserializer (our proposed 1-to-4 CDR plus an additional 4-to-8 DEMUX). The VCO minimum frequency in the CDR was 2.6 GHz, which is mean 100 MHz below the required one (2.7 GHz). Figure 4-11 (a and b) illustrates the transient simulation results. The serializer reaches the steady-state within 1.2 μs, followed by the deserializer in less than 2 μs later. As shown in Figure 4-12(a) the serial link is working properly, because the deserializer outputs *d1* and *d2* are the same as the serializer inputs *in1* and *in2*.
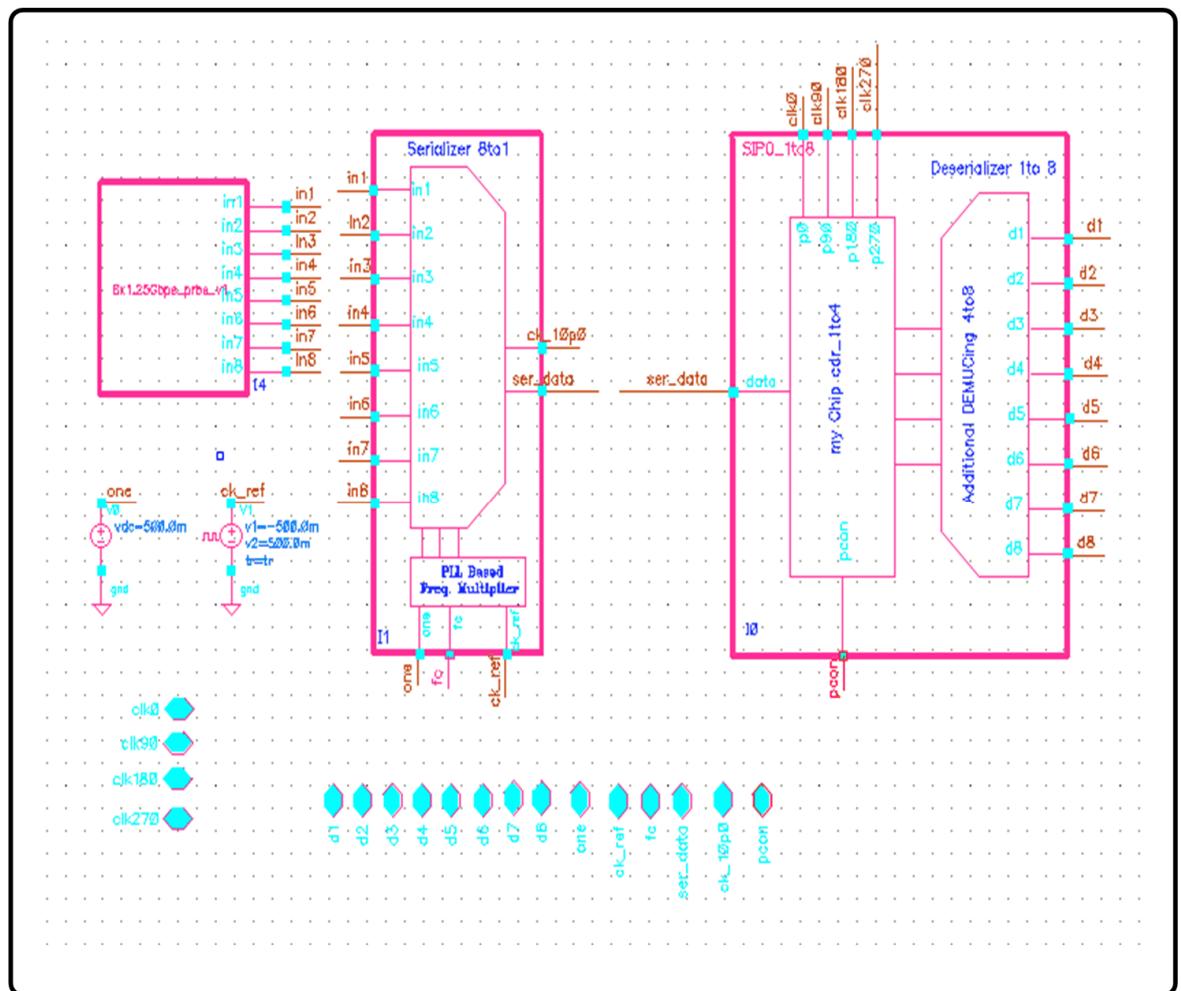
Figure 4-10: SerDes circuit test bench.
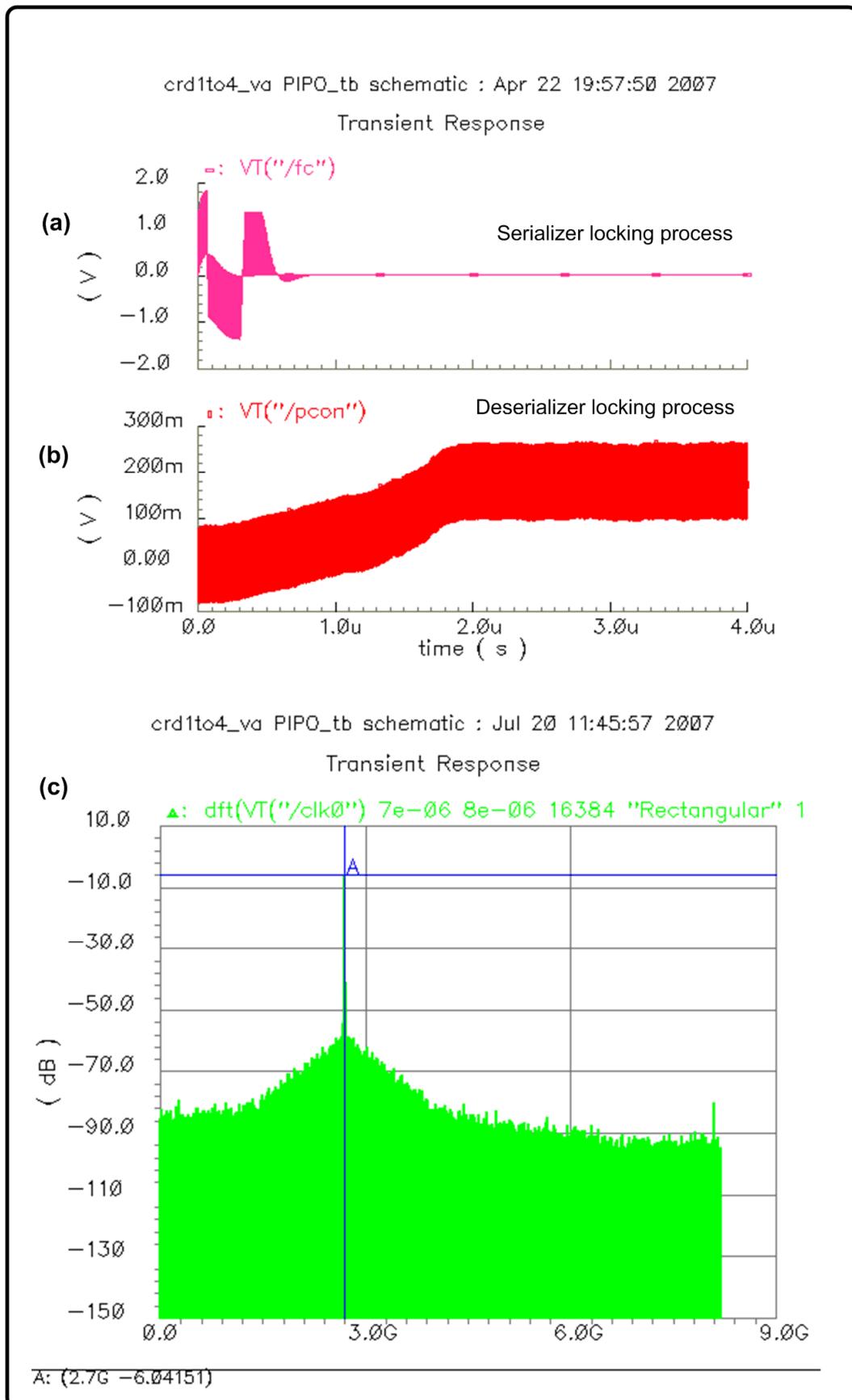
Figure 4-11: Low-pass filter output voltage showing the serial link locking process (a and b), and the DFT of the recovered clock in the deserializer (c).
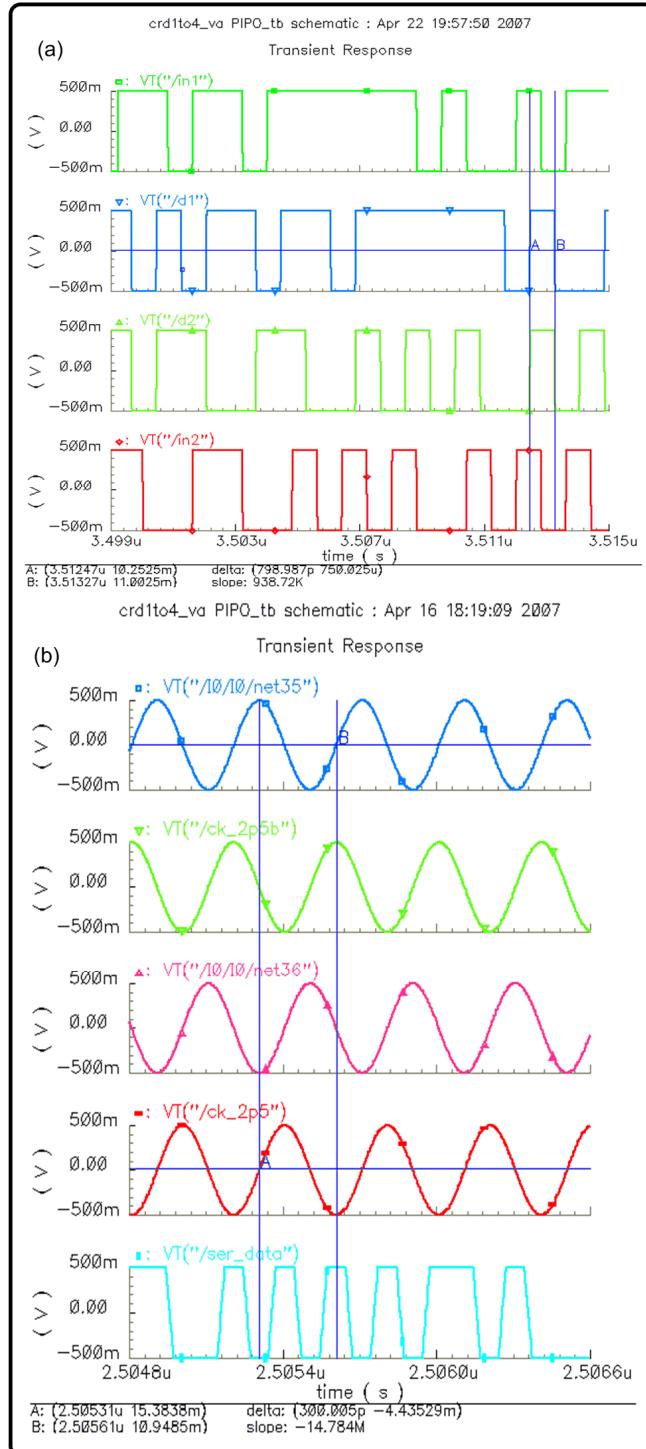
Figure 4-12: Serial link data input and output (a) and, serializer data and clock output (b).

In this chapter, we proved using the serial link schematic view and the Verilog-A language that our proposed quarter-rate concept PLL-Based CDR is a working concept for a point-to-point clockless based serial link interfacing two chips communicating serially at 10 Gb/s data rate.

5 Building Blocks Circuit Design

This chapter describes the transistor level design using the complementary metal-oxide-semiconductor (CMOS) current-mode logic (CML) of the blocks comprising the PLL-based CDR circuit, such as the ELPD, the DQFD, the VCO and the charge-pump.

5.1 Static and Dynamic Logic Gates Design

In this work the static logic gates (e.g. AND, OR, XOR and MUX) and the clocked dynamic elements (e.g. Latch, DFF, DETFF) were designed using CML in CMOS technology. The CMOS-CML (MCML) circuits were first used in [48] to implement gigahertz MOS adaptive pipeline technique. Since then, it has been extensively used to implement high-speed buffers [49, 50], latches [51], multiplexers and demultiplexers [52], and frequency dividers [53]. CML circuits use the current switching concept to represent the digital binary states. This type of circuit defines the logic states, high or low by the presence or absence of the current in the output branches. The basic MCML gate structure is shown below in Figure 5-1.
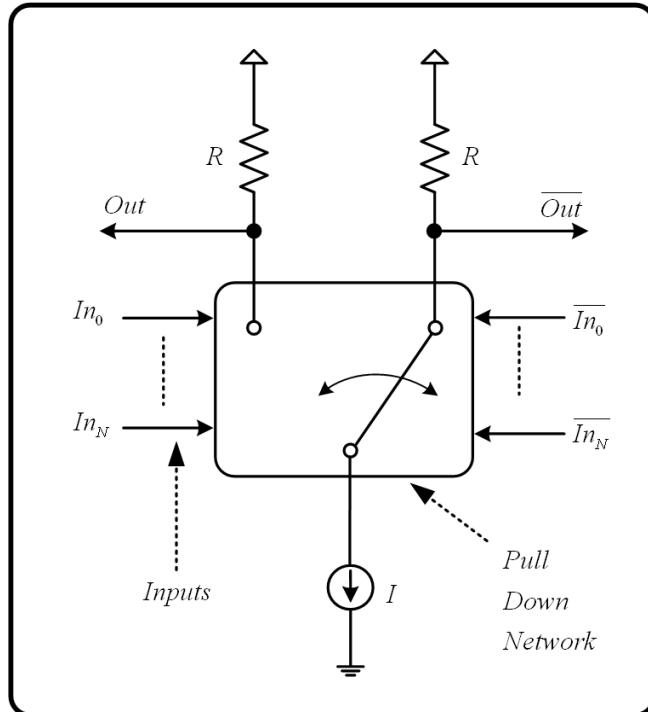
Figure 5-1: Basic CML gate.

The MCML gates are fully differential and steer current between the two pull-up resistors. The total voltage swing, $\Delta V = R.I_{tail}$, is set by adjusting the resistance of the pull-up devices for a given current. The voltage swing ΔV is not rail to rail but in fact is much less, of the order of several hundred millivolts.

5.1.1 CML Circuit Design Advantages and Comparison

CML circuits are widely known to have advantages over their CMOS counterparts which are especially useful for this application. First, CML circuits operate differentially; hence inherently rejecting any common-mode noise introduced by the power supply and the surrounding environment. Also, due to its reduced logical voltage swing, propagation delays are shorter [54], which will be translated to a faster switching circuit. Although CML logic style is known to suffer from more static power dissipation than the CMOS logic, properly designed CML gates can consume less power than the CMOS style at higher frequency of operation [55]. Especially, CML gates reduce current spikes and peak during logical transitions, which in turn reduce the effects of power supplies bouncing. CML circuits are mainly designed for low power and high frequency applications such as communication transceivers and serial links; they are usually incorporating resistive load rather than PMOS active load devices because the PMOS transistors severely limit the maximum operating frequency of the circuit [56, 57]. There are generally several techniques to implement logic circuits and functions in CMOS technology such as complementary MOS logic (CMOS), the MOS current mode logic (MCML), folded source mode logic (FSCL), domino logic, and complementary pass logic (CPL). The most popular design styles are based on the CMOS logic for digital circuits. The CMOS logic design style is known for being robust to the variations of fabrication process, hence producing reliable integrated circuits. In the other hand, the resistively loaded MCML circuits are sensitive to process variations and mismatch. For example, certain type of resistors can vary up to 30% in CMOS technology, which may affect the proper functionality of the logic circuits. Due to the popularity of using the CMOS logic style in VLSI systems, the MCML characteristics will be compared to it. The main parameters to be compared between the two logic styles are generally the delay, the power consumption and power-delay product.

Let's assume that our circuit is composed of an integer number N of identical gates connected in series, all with load capacitance C. The total propagation delay through the N gates will be given as follow [82]:

$$D_{MCML} = NRC = NC\frac{\Delta V}{I_{tail}} = N\frac{C \cdot \Delta V}{I_{tail}}$$

While the CMOS logic gates dissipate static and dynamic power, the MCML gates are drawing constant current over time and independent of the switching activity. Based on the above assumption, expressions for power, power-delay product can be written as follow:

$$P_{MCML} = N \cdot I_{tail} \cdot V_{DD}$$

$$PD_{MCML} = N \cdot I_{tail} \cdot V_{DD} \cdot N\frac{C \cdot \Delta V}{I_{tail}} = N^2 \cdot C \cdot \Delta V \cdot V_{DD}$$

The delay of static CMOS logic gates is given by:

$$D_{CMOS} = N \cdot C \cdot \frac{V_{DD}}{I_{sat}}$$

Where I_{sat} is the saturation sourcing or sinking current provided respectively by a PMOSFET or NMOSFET transistor in the CMOS gate. One core advantage of CMOS is that it draws minimal power under quiescent conditions. However, for our application of high speed data transfer we make the assumption that the CMOS is never quiescent, but continually switching. Under these conditions, the CMOS gate is continually charging or discharging a capacitor C between 0 and V_{DD}, and we can write the following expression:

$$C \cdot V_{DD} = I \cdot D_{CMOS}$$

Hence, $$I = \frac{C \cdot V_{DD}}{D_{CMOS}}$$

$$P_{CMOS} = N \cdot I \cdot V_{DD} = N \cdot V_{DD} \cdot \frac{C \cdot V_{DD}}{D_{CMOS}} = N \cdot C \cdot V_{DD}^2 \cdot \frac{1}{D_{CMOS}}$$

$$PD_{CMOS} = P_{CMOS} \cdot D_{CMOS} = N \cdot C \cdot V_{DD}^2 \cdot \frac{1}{D_{CMOS}} \cdot D_{CMOS} = N \cdot C \cdot V_{DD}^2$$

This CMOS power-delay product, under conditions of continual dynamic load, is higher than that for the MCML gate by a factor $V_{DD}/\Delta V$, where ΔV is the lower voltage swing of the MCML system. In effect the MCML circuit trades noise margin for a significantly improved power-delay product. If for instance the MCML tail current I_{tail} is equal to the PMOSFET (NMOSFET) saturation current in the CMOS gate, one therefore can compare the delay in both logics.

$$\frac{D_{CMOS}}{D_{MCML}} = N \cdot C \cdot \frac{V_{DD}}{I} \cdot \frac{1}{N \cdot \dfrac{C \cdot \Delta V}{I_{tail}}} = \frac{V_{DD}}{\Delta V}$$

As an example, for the CMOS technology provided by UMC (United Microelectronics Corporation) and having 130 nm feature size, $V_{DD} = 1.2$ Volts and $\Delta V = 0.2$ Volts, hence,

$$\frac{D_{CMOS}}{D_{MCML}} = \frac{1.2}{0.2} = 6$$

The selected value of ΔV above is the minimum voltage swing required to make the NMOSFET differential pairs in the MCML gate switch properly between on and off. The example above clearly shows that the delay is larger in CMOS than its counterpart in MCML logic and hence the operation frequency is higher in MCML, thus retaining only the delay (or frequency) parameter and ignoring the other, the MCML logic style is more suitable than CMOS logic for our particular high speed applications [82]. Another interesting point to be compared between the two logic styles is actually the common supply lines fluctuations during the bits transitions in digital integrated circuits. Since the voltage swing in MCML is much less than its CMOS counterpart, therefore the supplied lines current fluctuations for the CMOS inverter for example is higher than those in the MCML buffer. This reduced fluctuations in common supply lines, decrease in turn the amount of jitter propagated throughout the integrated circuit.

5.2 Oscillator Fundamentals

Oscillators are an essential part of many electronic systems, and they are generally embedded in the PLL circuits. They have a wide range of applications, such as clock generation in microprocessors and frequency synthesizers in cellular telephones, therefore requiring different oscillator topologies and performance parameters. In the following sections, the analysis and CMOS design of oscillators and more specifically the VCO will be carried out.

5.2.1 Negative Feedback Based Oscillator

The oscillator is a negative feedback system has no input and producing a periodic output, usually in the form of voltage. Let us consider the unity-gain feedback system shown in Figure 5-2, comprising an amplifier represented by its transfer function *H(s)*, the closed-loop gain of that system is then given by

$$\frac{V_{out}}{V_{in}}(s) = \frac{H(s)}{1 + H(s)} \tag{5.1}$$
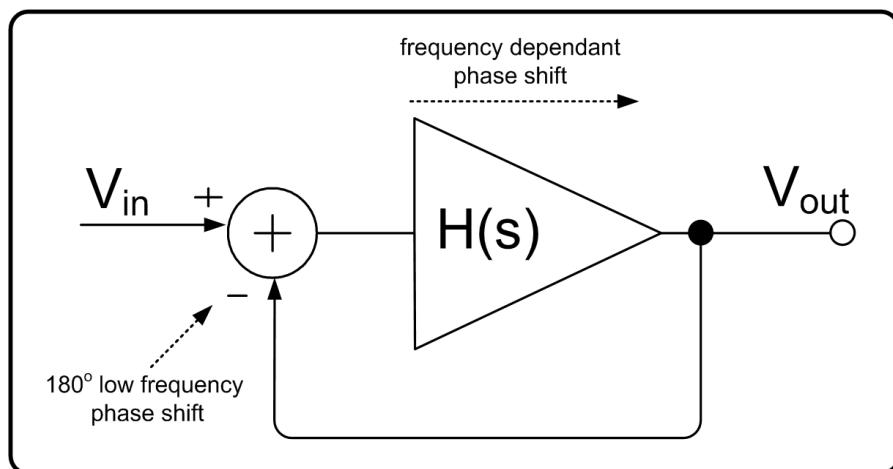
Figure 5-2: Negative feedback system.

If the amplifier introduces a frequency dependant phase shift such that the overall feedback becomes positive, then oscillation may occur. More precisely, if for $s = j\omega_0$, $H(j\omega_0) = -1$, then the closed-loop gain approach infinity at ω_0.

Under this condition, the circuit amplifies its own noise components at ω_0 indefinitely. As conceptually illustrated in Figure 5-3, a noise component at ω_0 having a total gain of unity and a phase shift of 180°, returning to the subtractor as a negative replica of the input. Upon subtraction, the input and feedback signals produce a larger difference. Thus, the feedback system amplifies continuously the noise component and hence generating a periodic signal at ω_0.
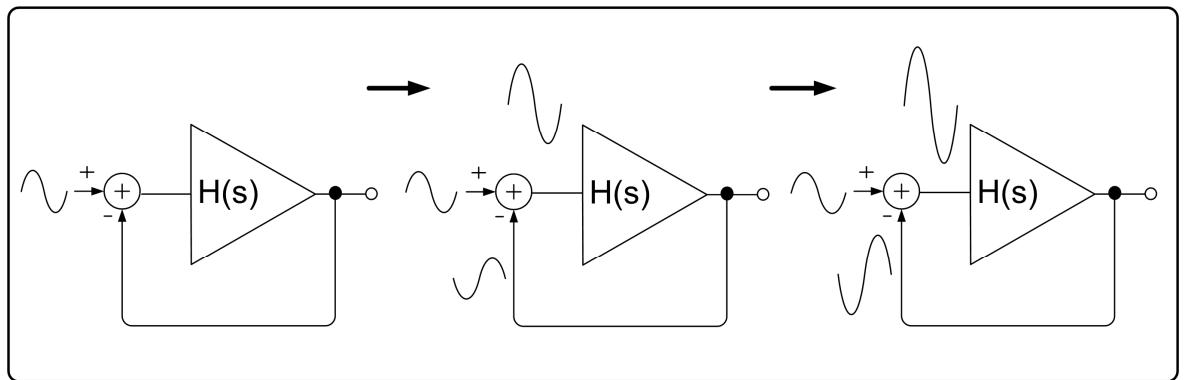
Figure 5-3: Oscillator and generation of periodic signal.

In summary, if a negative feedback circuit has an open loop gain *H(jω)* that satisfies the following two conditions:

$$| H(j\omega_0) | \geq 1$$
$$\angle H(j\omega_0) = 180^o$$

$$(5.2)$$

Then oscillation may occur at ω_0. The conditions described by Equation 5.1 are normally called Barkhausen criteria, these conditions are necessary but not sufficient [29]. To ensure the starting of oscillation in the presence of temperature and process variations, the open loop gain should be at least twice or three times the required value. The oscillation normally occurs when the total phase shift around the loop is equal 360°; this total phase shift is composed of two components, a low frequency phase shift of 180° represented by the subtractor, and a frequency dependant component of 180° introduced by the amplifier transfer function *H(jω)*. CMOS oscillators of today's technology are typically implemented as ring or LC oscillators; we focus more on ring type as it will be used in our CDR system.

5.2.2 Negative Resistance Based Oscillator

An alternative way of generating oscillation is to employ the concept of negative resistance, as the Colpitts oscillator [70-74] or LC-based oscillators [78-80]. To properly explain this concept, let us consider a simple tank composed of a coil with an inductance L_p, a capacitor with a capacitance C_p, and a resistor with a resistance R_p connected in parallel and excited by a current impulse as depicted in Figure 5-4(a). The tank responds with a decaying oscillatory behavior because, in every cycle, some of the energy that transferred between the capacitor and inductor is lost in the form of heat in the resistor. As shown in Figure 5-4(b), if a negative resistor equal to $-R_p$ is placed in parallel with R_p and the experiment is repeated, now since $R_p||(-R_p) = \infty$, then the tank will oscillate indefinitely. One of the methods to produce negative resistance is to use a positive feedback around a source follower [29]. As shown in Figure 5-5(a), the feedback is implemented by using a common gate stage and add the current source I_b to provide the bias current of M_2. From the equivalent circuit in Figure 5-5(b), we have

$$I_X = g_{m2}V_2 = -g_{m1}V_1$$

Where, g_{m1} and g_{m2} are the transconductance of transistors M_1 and M_2 respectively.

And,
$$V_X = V_1 - V_2 = -\frac{I_X}{g_{m1}} - \frac{I_X}{g_{m2}} = -I_X\left(\frac{1}{g_{m1}} + \frac{1}{g_{m2}}\right)$$

And, if $g_{m1} = g_{m2} = g_m$, then

$$Z_X = \frac{V_X}{I_X} = -\frac{2}{g_m} \tag{5.3}$$

Since $g_m > 0$, then $Z_X < 0$. In other word, if the input voltage in Figure 5-5(a) is increased, so does the source of the transistor M_1, reducing the drain source voltage of M_2 thus reducing the drain current of M_2, and allowing part of I_b to flow back to the input source hence reducing it. One of the negative resistance based oscillator design is shown in Figure 5-6(a). Here, L_p provides the bias current to M_2 and R_p denotes the equivalent parallel resistance of the tank and, for oscillation to occur $R_p - 2/g_m \geq 0$.
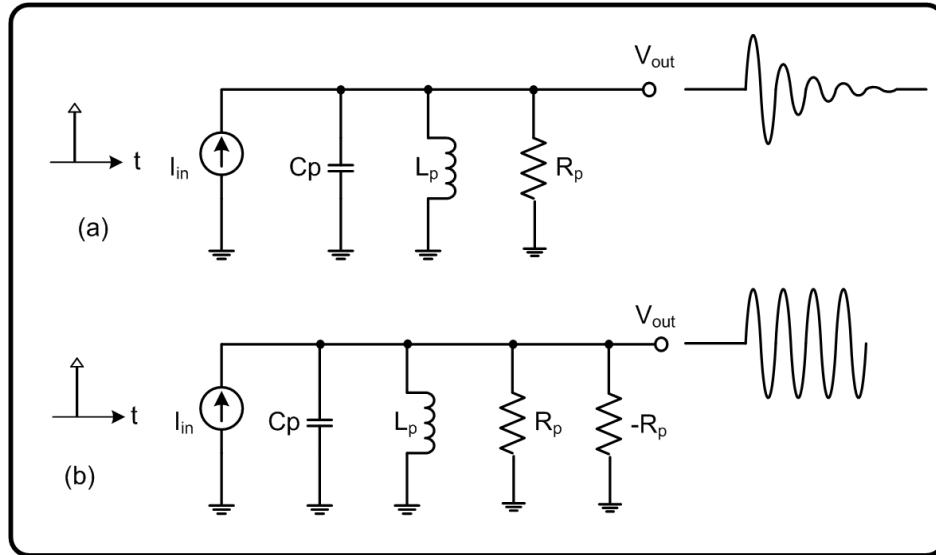
Figure 5-4: (a) Decaying impulse response of a tank, (b) addition of negative resistance to cancel loss in R_p.

If the small signal negative resistance presented by M_1 and M_2 to the tank is less than R_p, then the circuit experiences large swings such that each transistor is nearly off for part of the period, thereby yielding an average resistance of $-R_p$. The design of Figure 5-6(a) is a single ended and it can be modified to obtain a differential design as shown in Figure 5-6(b), merging the two tanks into one, we obtain the design shown in Figure 5-7. In order to generate oscillation in the circuit of Figure 5-7, the cross coupled pair (M_1 and M_2) must provide a negative resistance of $-R_p$ between the nodes X and Y. As proven earlier, the cross coupled pair resistance is equal to $-2/g_m$ and hence it is necessary to have $R_p \geq 1/g_m$ for the oscillation to occur.
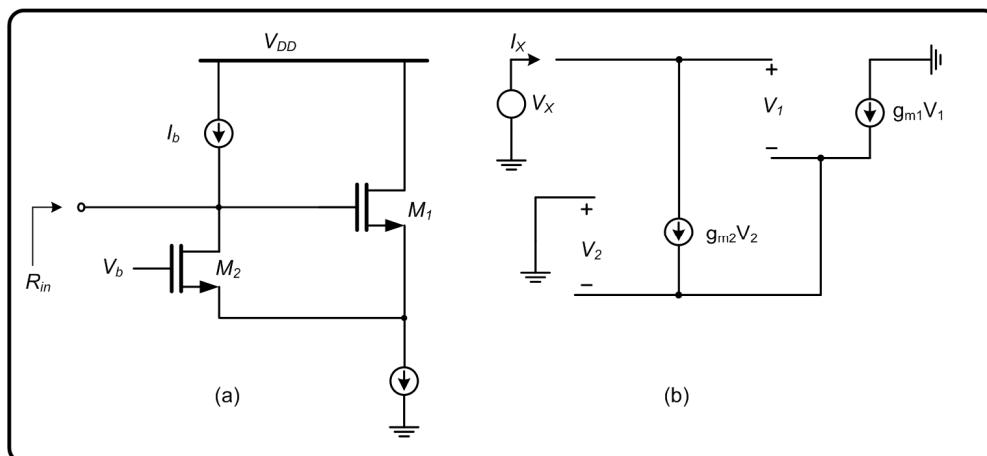
Figure 5-5: (a) Source follower with positive feedback to create negative impedance, (b) equivalent circuit of (a).
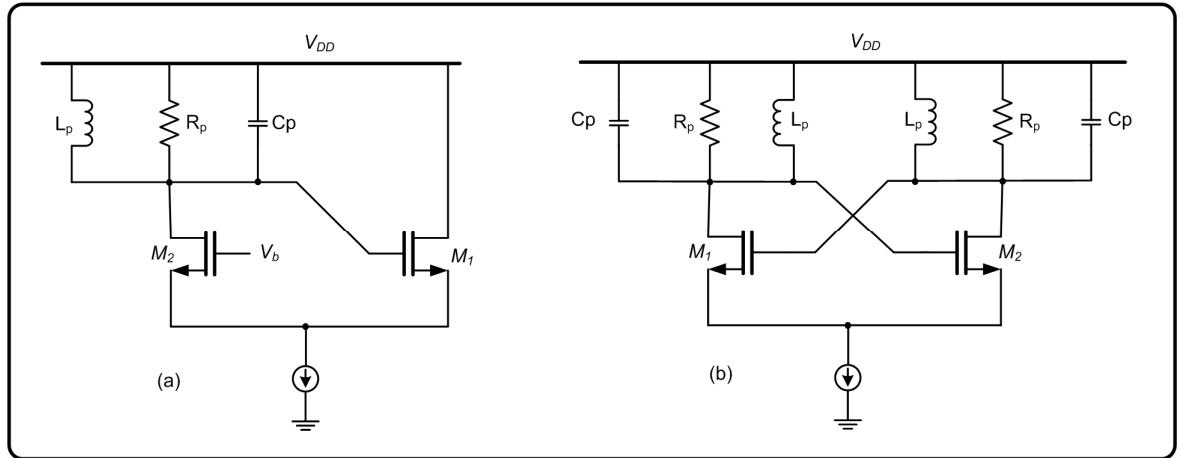
Figure 5-6: (a) Single and, (b) differential ended negative resistance based oscillator.
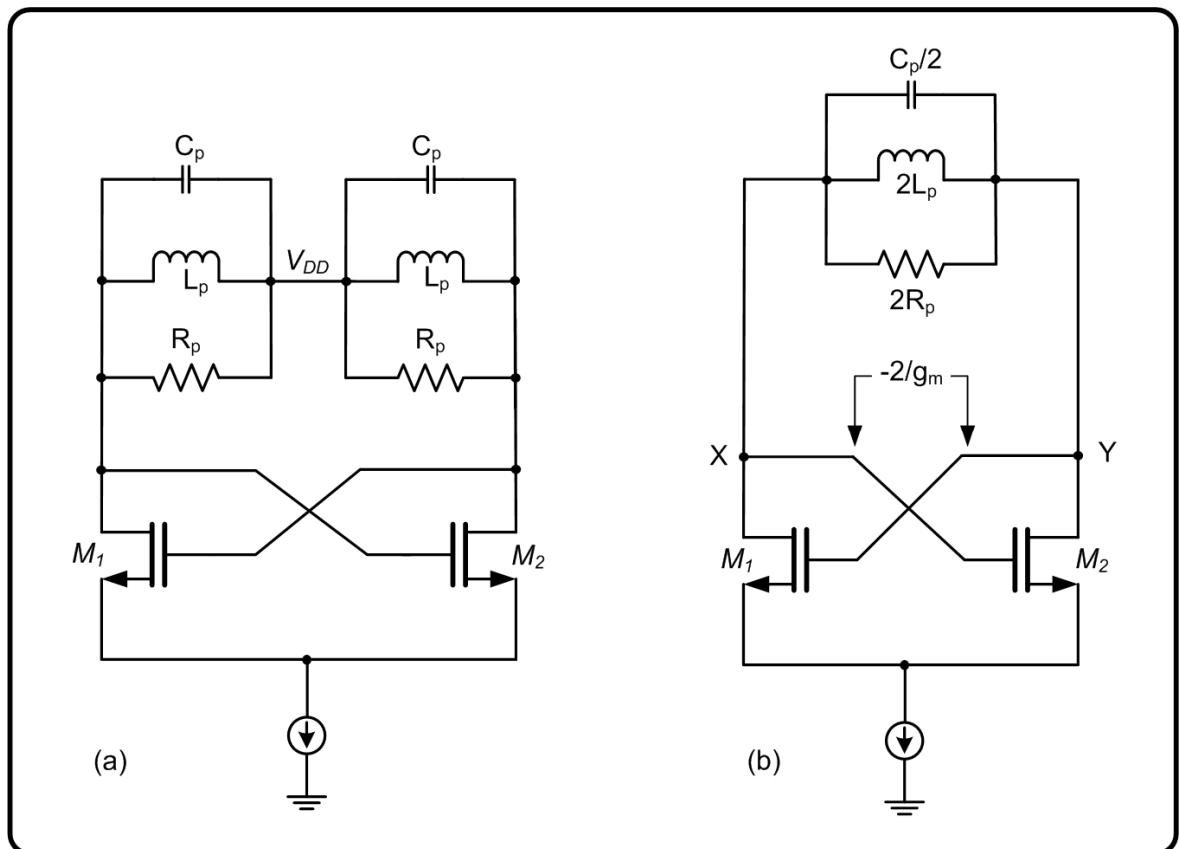
Figure 5-7: (a) Oscillator and, (b) its equivalent circuit.

5.2.3 Ring Type Oscillator

A ring oscillator consists generally of a number of gain stages in a closed loop [75-77]. For the proper operation of our CDR circuit, eight clock phases separated by 22.5^{o} and their complements will be required; obtaining such clock phases requires a differential ring type oscillator comprising eight gain stages as shown in Figure 5-8(a). To simplify the analysis, let us consider the half-circuit equivalent depicted in Figure 5-8(b), and calculate the minimum voltage gain that is necessary for the oscillation to occur. In this oscillator design, the eight gain stages are identical, in which, R_D and C_L represents the total resistance and total capacitance seen by the output node of each gain stage.
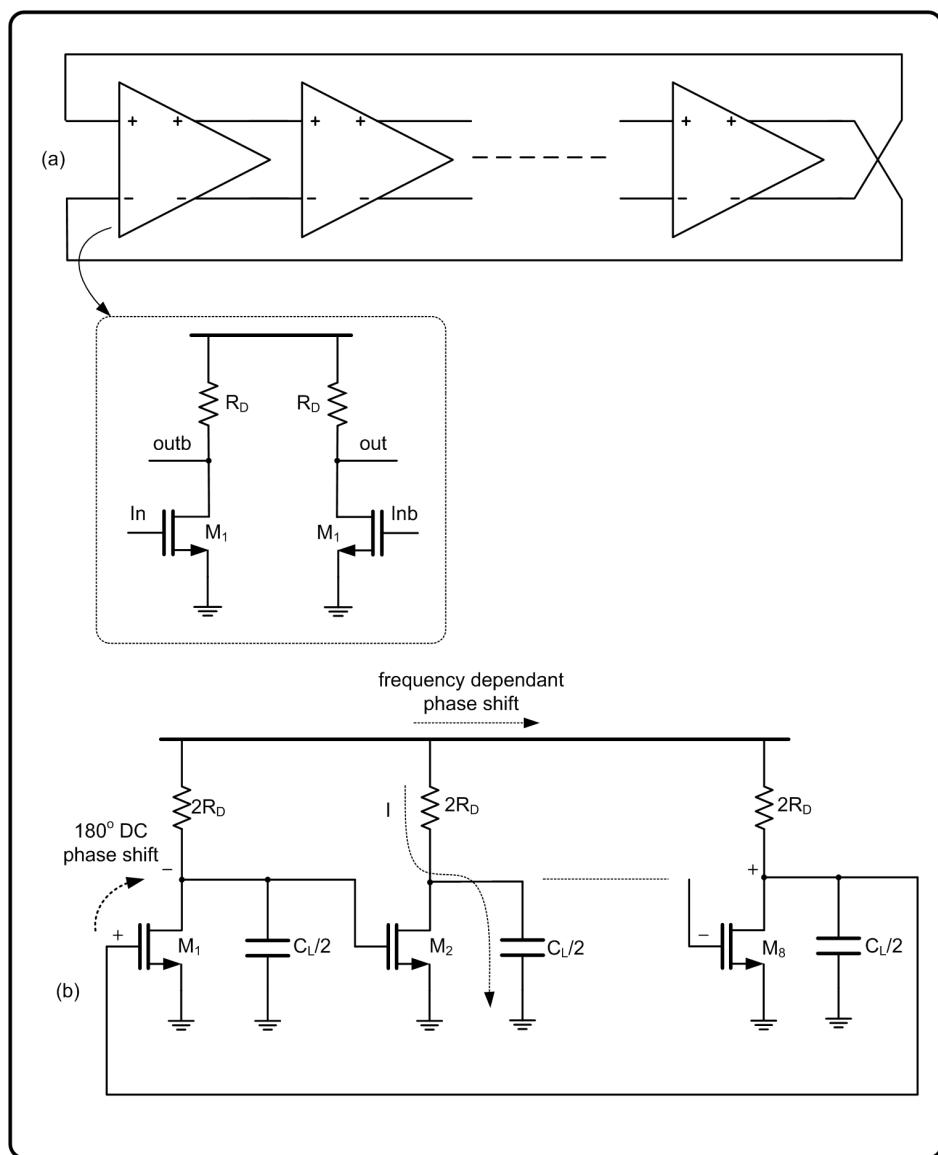
Figure 5-8: Differential eight gain stages ring oscillator (a) and (b) its half circuit equivalent.

If the transfer function of each gain stage is $H_0(s)$, then the open loop transfer function of the eight gain stages will be given by

$$H(s) = H_0(s).H_0(s).......H_0(s) = \frac{-A_0}{(1+\dfrac{s}{\omega_0})} \cdot \frac{-A_0}{(1+\dfrac{s}{\omega_0})} \cdots\cdots \frac{-A_0}{(1+\dfrac{s}{\omega_0})} = \frac{A_0^8}{(1+\dfrac{s}{\omega_0})^8}$$

$$\text{Where, } \omega_0 = \frac{1}{2R_D \cdot \dfrac{C_L}{2}} = \frac{1}{R_D C_L}$$

Hence,

$$H(s) = \frac{A_0^8}{(1+\dfrac{s}{\omega_0})^8} \tag{5.4}$$

The oscillation will start only if the total frequency dependant phase shift equal 180°, or if each stage contributes 22.5^o ($=180^o/8$). The frequency at which this occurs is given by

$$\tan^{-1}\frac{\omega_{osc}}{\omega_0} = 22.5^o \tag{5.5}$$

And hence,

$$\omega_{osc} = 0.45\omega_0 \tag{5.6}$$

The minimum voltage gain per stage must be such that the magnitude of the open loop gain at ω_{osc} is equal to unity:

$$\frac{A_0^8}{[\sqrt{1+(\dfrac{\omega_{osc}}{\omega_0})^2}\,]^8} = 1 \tag{5.7}$$

It follows from Eq. 5.5 and Eq. 5.6 that

$$A_{0,min} = 1.1 \tag{5.8}$$

In summary, an eight-stage ring oscillator requires a low frequency gain of 1.1 per stage, and it oscillates at a frequency of $0.45\omega_0$, where ω_0 is the -3dB bandwidth of each stage.

The waveforms at the eight nodes of the oscillator of Figure 5-8 are depicted on Figure 5-9. Each stage of the oscillator contribute a frequency dependant phase shift of 22.5^o as well as a low frequency signal inversion, hence the waveform at each output node is 202.5^o ($=180^o + 22.5^o$) out of phase with respect to its previous and next nodes. The ability of generating multiple phases is a very useful property of ring oscillators, because those phases are required for the proper operation of our CDR circuit.
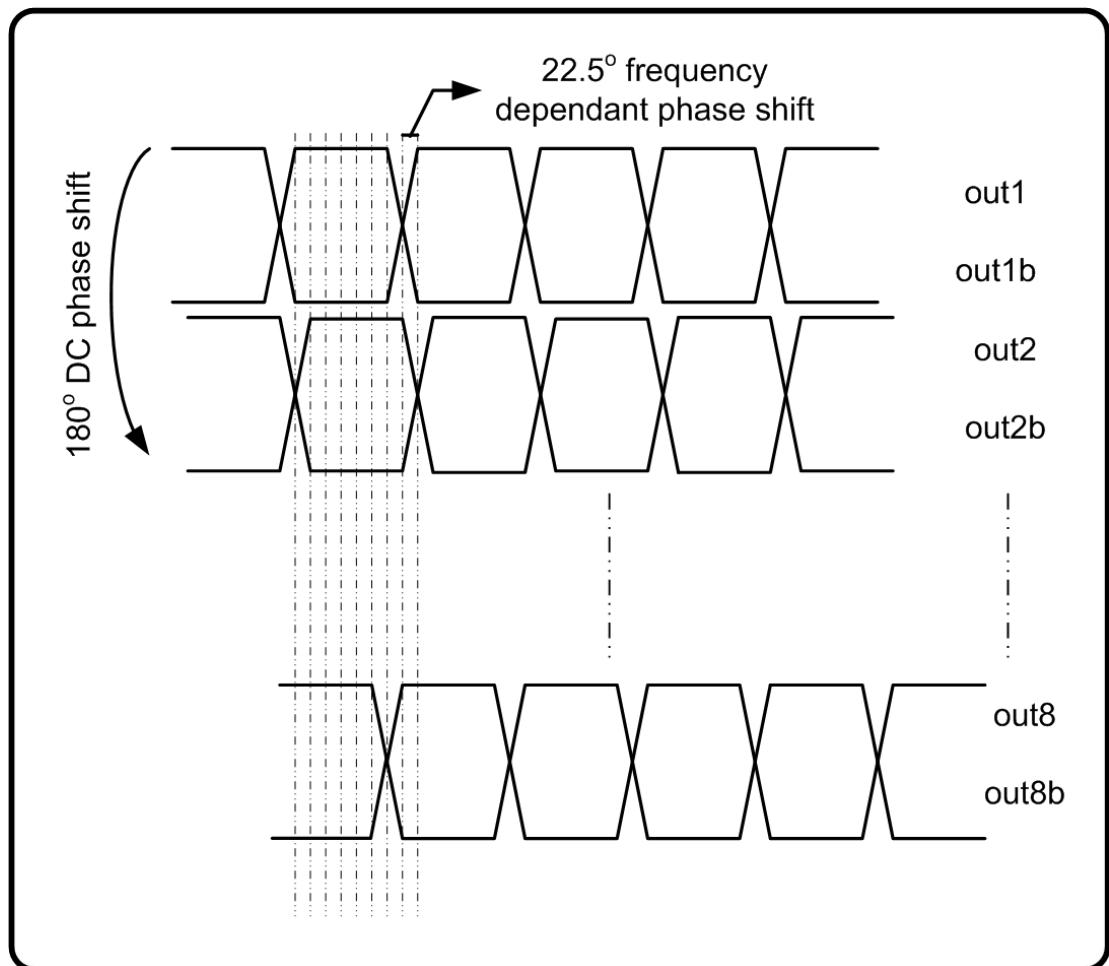
Figure 5-9: Waveforms of an eight-stage ring oscillator.

One of the practical implementation of eight stages ring oscillator is depicted on Figure5-10 and called the current steering based differential ring oscillator. If the gain per stage is well above 2 ($A_{0,min}=1.1$), then the amplitude grows until each differential pair experiences complete switching, that is , until the current I_s is completely steered to one side every half cycle. As a result, the swing at the output node is equal to I_sR_D.
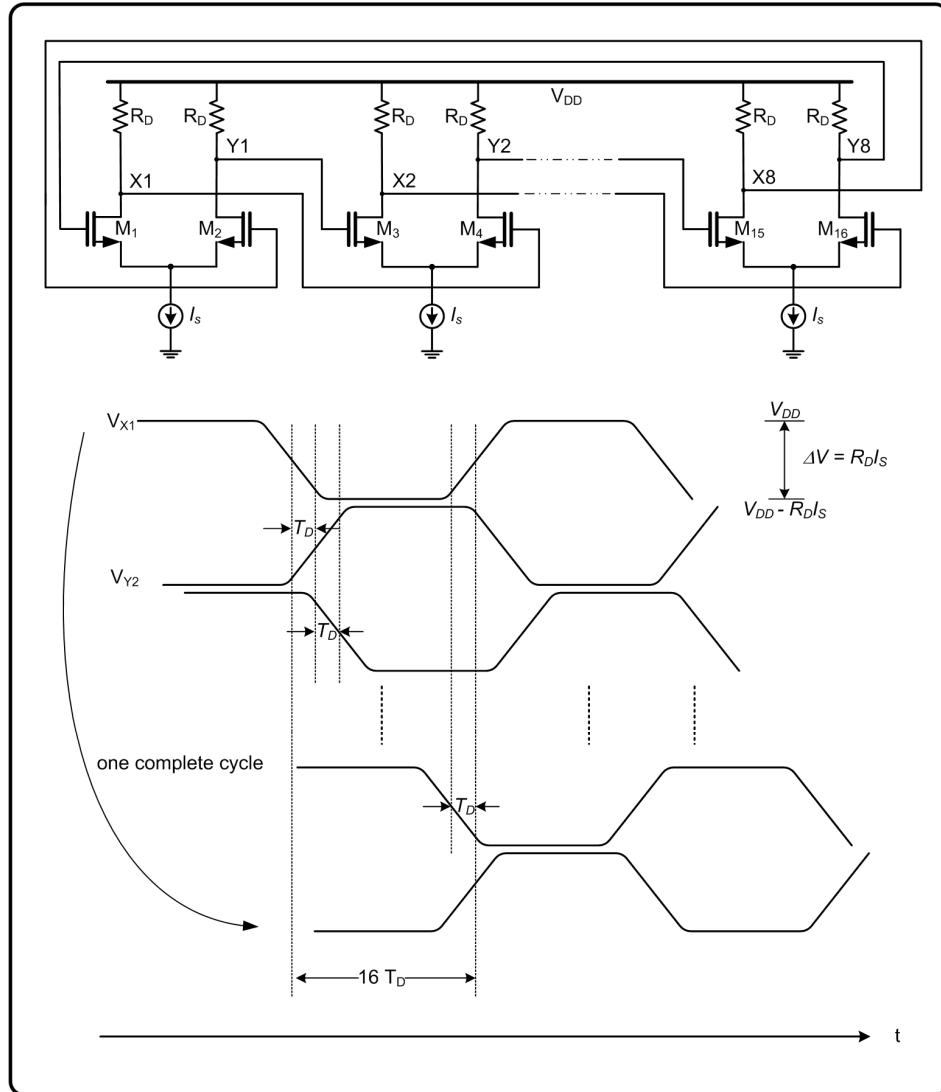
Figure 5-10: Differential current steering ring oscillator and its waveforms.

If the number of stages is N and the delay per stage is T_D, thus the circuit completes one period of oscillation in a lap of time equal to $2NT_D$ and hence the circuit oscillates at a frequency equal to $1/2NT_D$. In summary, an eight-stage (N = 8) differential ring oscillator has a small-signal oscillation frequency equal to $0.45\omega_0$ (Eq. 5.3) and a large-signal value equal to $1/16T_D$. Since ω_0 is determined by the small-signal output resistance and capacitance of each stage whereas T_D is results from the large signal, nonlinear current drive and capacitance of each stage, therefore the large-signal frequency is less than the small-signal one. In other word, the eight-stage ring oscillator starts oscillating with a frequency of $0.45\omega_0$ but, as the amplitude grows and the circuit becomes nonlinear, the frequency shifts to the lower value of $1/16T_D$.

5.3 Voltage-Controlled Oscillators

Most applications require that oscillators be tunable, i.e., their output frequency is a function of a control input, usually a voltage. As shown in Figure 5-11, an ideal voltage controlled oscillator is a circuit in which the output frequency is a linear function of its input control voltage and it is described by the following equation:

$$\omega_{vco} = \omega_0 + k_{vco}.v_f \qquad (5.9)$$

Where, ω_0 is the frequency corresponding to $V_f = 0$ and K_{vco} denotes the sensitivity of the VCO expressed in rad/V.S. The linear range, ω_2-ω_1, is called the tuning range.
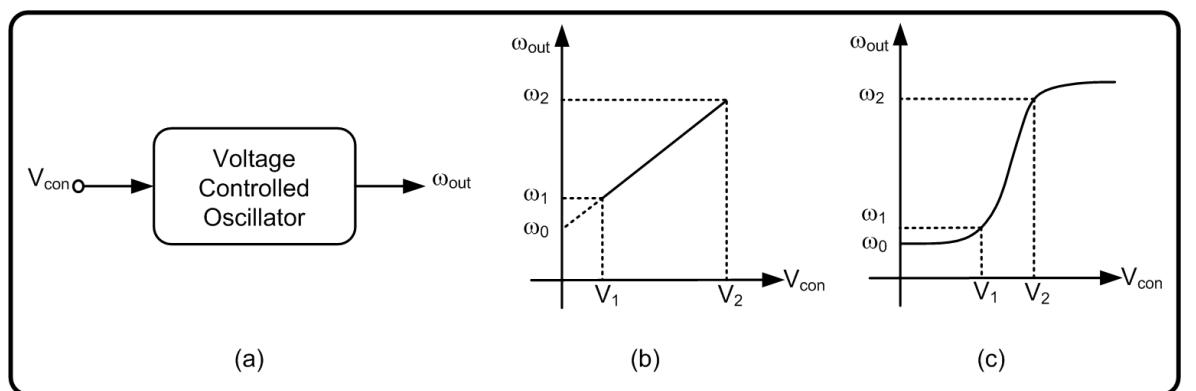
Figure 5-11: Definition of a VCO (b) ideal and, (c) real.

5.3.1 Tuning in Ring Oscillators

As seen earlier, the oscillation frequency of an N-stage ring oscillator is equal to 1/2ND, where T_D represents the large signal delay of each stage. Therefore, to tune the frequency, T_D should be varied. As an example of tuning, consider the differential pair of Figure 5-12(a) as one stage of a ring oscillator. Here, M_3 and M_4 are operating in the triode region and are acting as voltage variable resistors controlled by V_{cont}. As V_{cont} becomes more positive, the on-resistor of M_3 and M_4 increases, thus raising the time constant τ and hence lowering the oscillation frequency f_{osc}. If M_3 and M_4 are and remain in deep triode region, therefore

$$R_{on3,4} = \frac{1}{\mu_p C_{ox} (\frac{W}{L})_{3,4} (V_{DD} - V_{cont} - |V_{thp}|)} \tag{5.10}$$

Where, $R_{on3,4}$ is the on-resistance of the PMOS transistors M_3 and M_4. Thus

$$\tau = R_{on3,4} C_L = \frac{C_L}{\mu_p C_{ox} (\frac{W}{L})_{3,4} (V_{DD} - V_{cont} - |V_{thp}|)} \tag{5.11}$$

Where, C_L represents the total capacitance seen by each output to ground including the input capacitance of the following stage. The total delay in the circuit is proportional to the delay in each stage, hence

$$f_{osc} = \frac{1}{2NT_D} = \frac{\mu_p C_{ox} (\frac{W}{L})_{3,4} (V_{DD} - V_{cont} - |V_{thp}|)}{2NC_L} \tag{5.12}$$

Eq. 5.4 shows that the frequency of oscillation f_{osc} of an N stages ring oscillator is linearly proportional to the control voltage V_{cont} and inversely proportional to the number of stages N of the oscillator.

5.3.2 Delay Variation by Positive Feedback

An alternative tuning technique is based on the current controlled negative resistance. As seen earlier, a cross coupled transistor pair such that of Figure 5-7 exhibits a negative resistor of $-2/g_m$, a value that can be controlled by the bias current of the cross coupled transistors. If a negative resistance R_N is placed in parallel with a positive resistance R_P the equivalent resistor $R_{eq.}$ will be given by

$$R_{eq.} = R_N \| R_P = \frac{R_N R_P}{R_N + R_P}$$

If for example $|R_N| \rangle |R_P|$, then $R_{eq.}$ is less negative and it has therefore a higher value. This concept can be used in each stage of a ring oscillator as illustrated in Figure 5-12(b).

Here, the load of the differential pair M_1-M_2 consists of resistors R_1 and R_2 ($R_1 = R_2 = R$) and the cross coupled pair M_3-M_4.

As I_{cc} increases, the small signal differential resistance $-2/g_{m3,4}$ becomes less negative and, from the half circuit of Figure 5-12(c), the equivalent resistance

$$R_{eq.} = R_P \, || \, (\frac{-1}{g_{m3,4}}) = \frac{R_P}{1 - g_{m3,4}R_P}$$ increases, thereby lowering the frequency of oscillation.
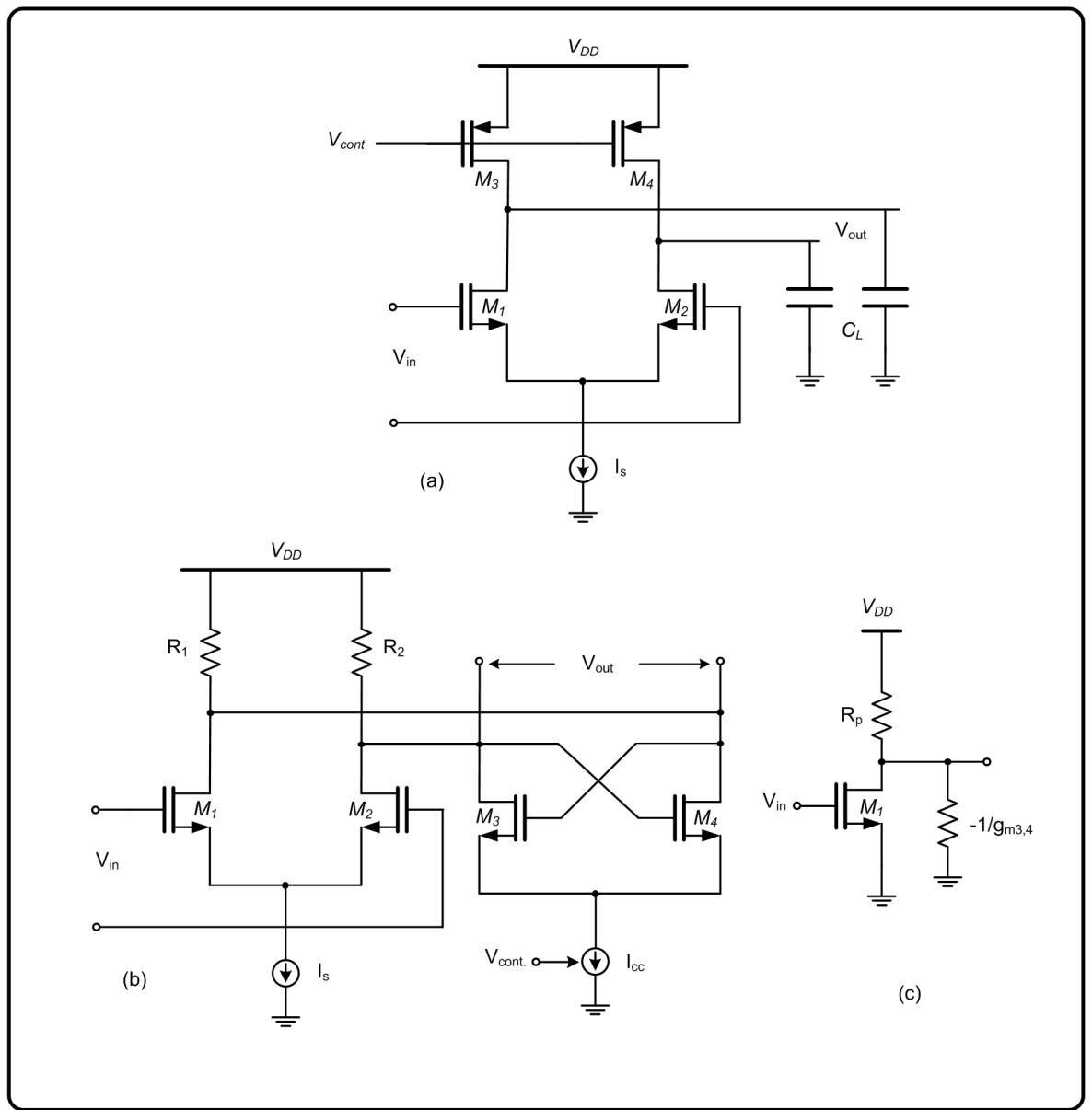

Figure 5-12: (a) Tuning with voltage variable resistors, (b) differential stage with variable negative resistance load, (c) half circuit equivalent of (b).

A drawback in the circuit of Figure 5-12 is that as I_{cc} varies, so does the current steered by the pair M_3-M_4 through R_1 and R_2. Thus, the output voltage swing is not constant across the tuning range. To reduce this effect, I_s can be varied in the opposite direction of I_{cc} such that the total current steered between R_1 and R_2 remains constant. In other words, it is preferable to vary I_{cc} and I_s differentially while their sum is fixed, this property is normally provided by a differential pair. As illustrated in Figure 5-13, the idea is to use the differential pair M_5-M_6 to steer I_T between the two pairs M_1-M_2 and M_3-M_4 such that the expression $I_T = I_s + I_{cc}$ is always verified. Since I_T must flow through R_1 and R_2, if M_1-M_4 experience complete switching in each cycle of oscillation, then I_T is steered to R_1 (through M_1 and M_3) in half a period and to R_2 (through M_2 and M_4) in the other half, giving a differential swing of $2R_pI_T$. The control voltages V_{cont1} and V_{cont2} in the circuit of Figure 5-13 can be viewed as differential control lines if they vary by equal and opposite amounts. Differential topology provides normally higher noise immunity for the control input than if V_{cont} is single ended.

As V_{cont2} increases and V_{cont1} decreases, the transconductance of the cross coupled pair increases, increasing the time constant τ and hence reducing the frequency of oscillation. A drawback of circuit in Figure 5-13 is that when the current I_T is completely steered by M_6 through the pair M_3-M_4. Since the pair M_1-M_2 carries no current at all, hence the gain of each stage will fall eventually to zero, preventing oscillation. To avoid the occurrence of this situation, a small constant current I_{bias} is added to the pair M_1-M_2, thereby ensuring M_1 and M_2 remain always on. We calculate the required minimum value of I_{bias} in Figure 5-13 to guarantee a low frequency gain of 1.1 (for N = 8) when all of I_T is steered to the cross coupled pair M_3-M_4. The small signal gain of the circuit 5-13 is given by [29]

$$A_{\min} = \frac{g_{m1,2}R_p}{1-g_{m3,4}R_p} = \frac{R_p\sqrt{\mu_nC_{ox}(\frac{W}{L})_{1,2}I_{bias}}}{1-R_p\sqrt{\mu_nC_{ox}(\frac{W}{L})_{3,4}I_T}} \geq 1.1$$

That is,
$$I_{bias} \geq \frac{1.21\,[1-R_p\sqrt{\mu_nC_{ox}(\frac{W}{L})_{3,4}I_T}\,]^2}{\mu_nC_{ox}(\frac{W}{L})_{1,2}I_{bias}R_p^2} \qquad (5.13)$$
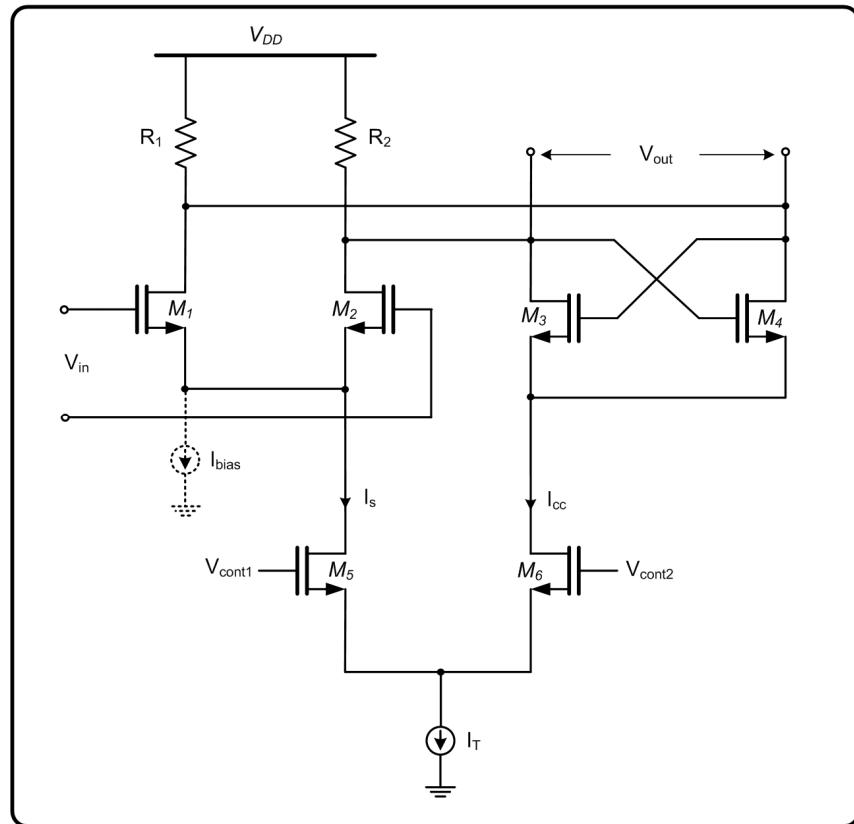
Figure 5-13: Differential pair used to steer current between M_1-M_2 and M_3-M_4.

5.4 A Novel Quarter-Rate Early-Late Phase-Detector

Before presenting our novel quarter-rate early-late *PD* (*ELPD*), we will briefly explain the concept of full-rate (i.e. the clock frequency is equal the data rate) *ELPD* that is originally proposed by Alexander [24]. Figure 5-14, illustrates the concept of early-late detection method. Using three data samples taken by three consecutive clock edges, the PD can determine whether a data transition is present, and whether the clock leads or lags the data. In the absence of data transitions, all three samples are equal and no action is taken. If the clock leads (it is early), the first sample S_1, is unequal to the last two (i.e. S_2 and S_3). Conversely, if the clock lags (it is late), the first two samples, S_1 and S_2, are equal but unequal to the last S_3. Thus, $S_1 \oplus S_2$, and $S_2 \oplus S_3$ provide the early-late information:

- If $S_1 \oplus S_2$ is high and $S_2 \oplus S_3$ is low, the clock is late.
- If $S_1 \oplus S_2$ is low and $S_2 \oplus S_3$ is high, the clock is early.
- If $S_1 \oplus S_2$ is equal to $S_2 \oplus S_3$, no data transition is present.

Based on the above observations, the Table 5-2, and Figure 5-14 can be constructed.

S_1	S_2	S_3	$Y = S_1 \oplus S_2$	$X = S_2 \oplus S_3$	Detection (Action)
0	0	0	0	0	no decision (no action)
0	0	1	0	1	early (slow down)
0	1	0	1	1	no decision (no action)
0	1	1	1	0	late (speed up)
1	0	0	1	0	late (speed up)
1	0	1	1	1	no decision (no action)
1	1	0	0	1	early (slow down)
1	1	1	0	0	no decision (no action)

Table 5-1: Truth table representing all states of the Alexander ELPD.
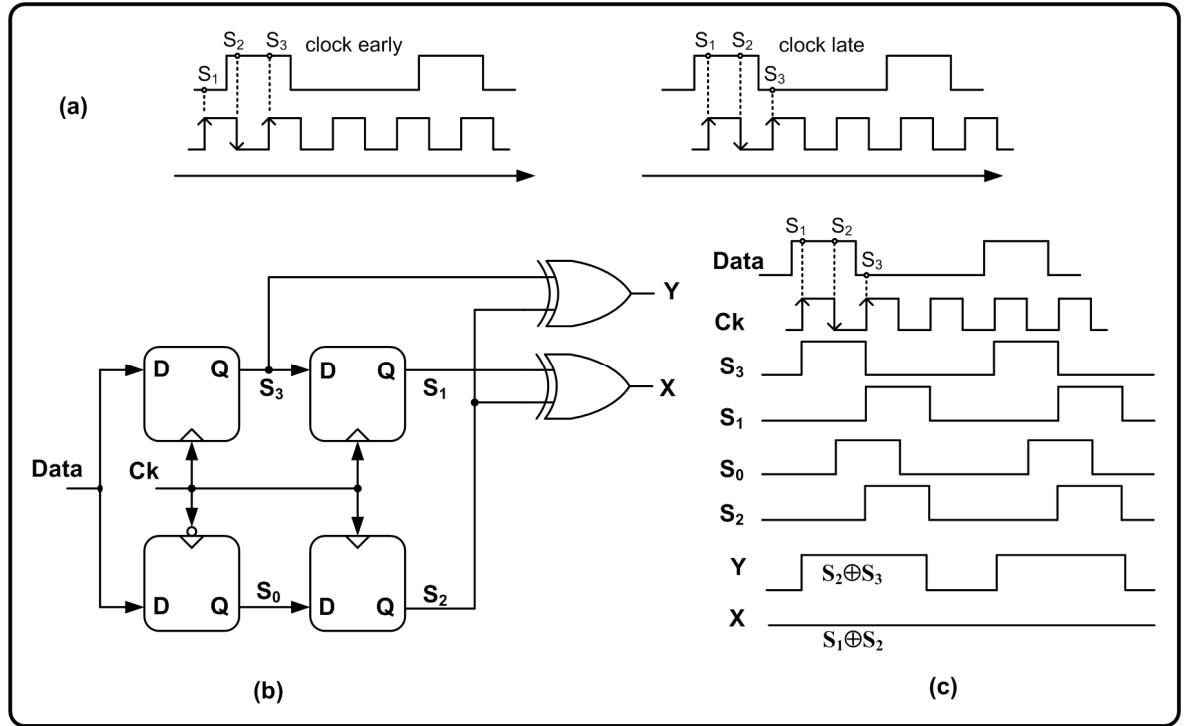
Table 5-14: (a) Three points sampling of data by clock, and (b) an Alexander ELPD.

The proposed quarter-rate (i.e. the clock frequency is one quarter of the data rate) phase detector is an ELPD (Alexander) based design. As shown in Figure 5-15, the *ELPD* samples the input data stream at $0^{°}$, $45^{°}$, $90^{°}$, $135^{°}$, $180^{°}$, $225^{°}$, $270^{°}$ and $315^{°}$ of the clock phase, producing the eight signals D_0, D_{45}, D_{90}, D_{135}, D_{180}, D_{225}, D_{270} and D_{315} at the *DFF* outputs. The last eight signals are used to generate the *UP* and *DN* signals that indicate the relative clock edge positions with respect to the data edges. The required logic to produce the *UP* and *DN* signals are as follow:

$$UP_1 = D_0 \oplus D_{45}, UP_2 = D_{90} \oplus D_{135}, UP_3 = D_{180} \oplus D_{225}, and\ UP_4 = D_{270} \oplus D_{315}.$$

$$DN_1 = D_{45} \oplus D_{90}, DN_2 = D_{135} \oplus D_{180}, DN_3 = D_{225} \oplus D_{270}, and\ DN_4 = D_{315} \oplus D_0.$$

In order to simplify the charge pump circuit design, the signals UP_1-UP_4 and DN_1-DN_4 are serialized using the clock phases 22.5^{o}, 67.5^{o}, 112.5^{o}, 157.5^{o}, 247.5^{o}, and 292.5^{o} as illustrated in Figure 5-15. When the CDR is in the locked state, the half-quadrature clock signal edges are aligned with the data transitions and, hence, D_0, D_{90}, D_{180} and D_{270} will be the recovered demultiplexed data.
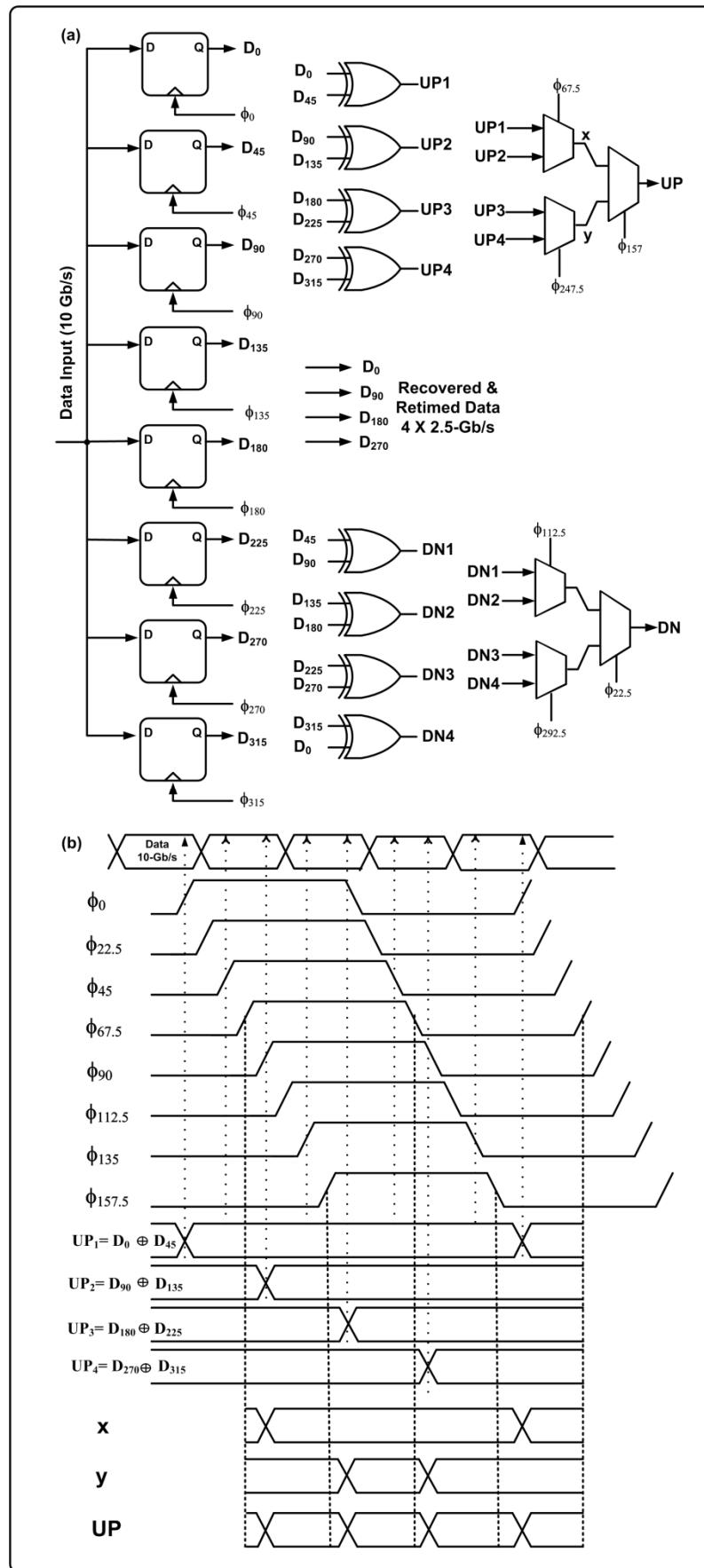
Figure 5-15: (a) Block diagram of the proposed quarter-rate ELPD, and (b) its operation.

5.5 A Novel Quarter-Rate Frequency Detector

The loop bandwidth of the PLL based CDR circuit (ω_{-3dB}) is generally has to be small to improve the noise performances [58-60]. However, it results in small pull-in range PLL. CDRs without frequency acquisition techniques may require additional reference clock [59] or external off chip tuning [60]. Digital quadricorrelator frequency detectors (DQFD) [62-63] have been widely used in frequency acquisition loops because they are more reliable and tolerant to process, voltage and temperature variations. However, the conventional DQFD [63] could be used only for full rate clocks. To lower the power consumption, clock relaxing techniques [58-60] have been used to achieve higher bit rate transmission with lower clock rate.
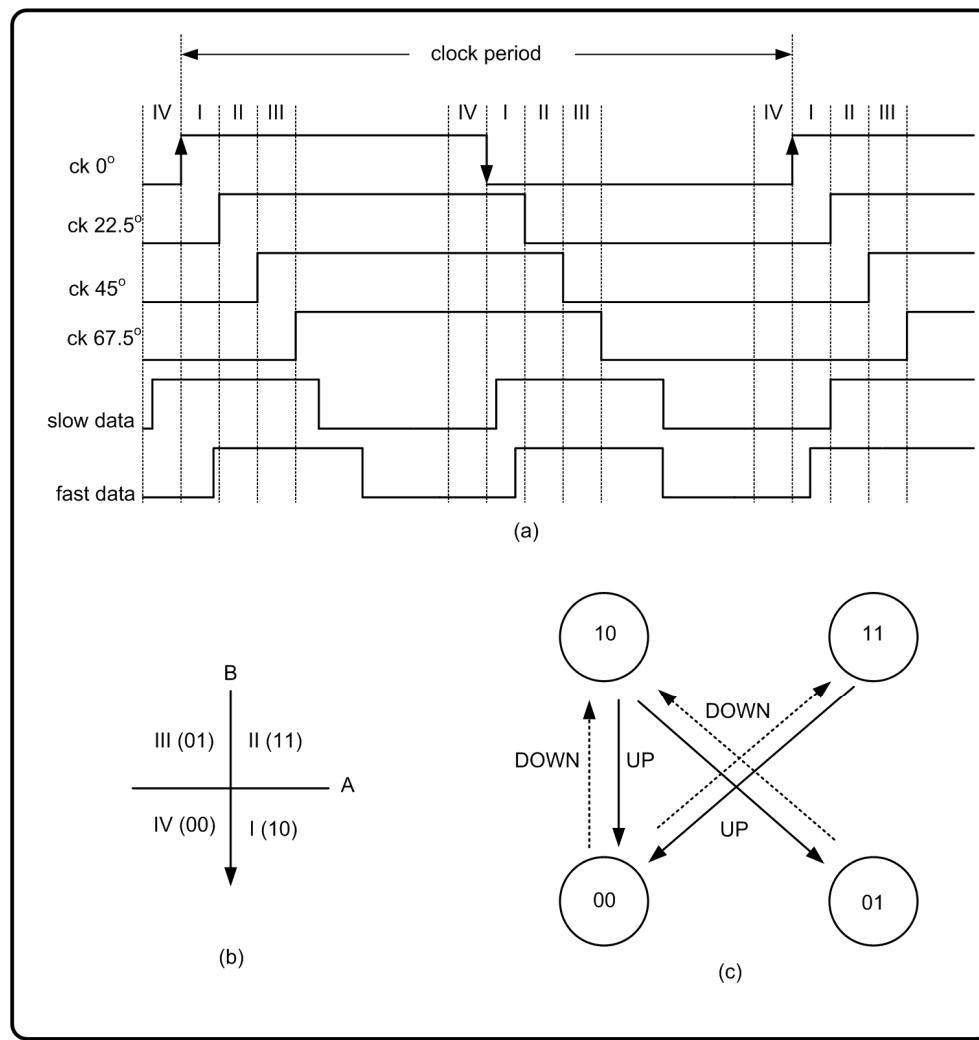
Figure 5-16: Timing diagram for (a) slow and fast data, (b) state representation and, (c) finite state diagram.

In this work, we propose a quarter-rate DQFD [34, 64], the proposed architecture comprises eight DFFs, two XOR gates, and combinational logics as shown in Figure 5-17. The combinational logics truth table of the proposed quarter-rate DQFD is shown in Table 5-1. Clocks 0°, 22.5^o, 45° and 67.5^o are first sampled by input data, each half of a clock period (i.e. 200 ps) is divided into four states, I, II, III, and IV as shown in Figure 5-16(b). In the proposed DQFD four DFFs triggered by rising and falling edges of the clock 0° will store the sampled values and record the states. The arrow in Figure 5-16(b) represents the rising or falling edge of the clock 0° to appear at the boundary between the states IV and I. the operational Principle of the proposed quarter-rate DQFD will be discussed in the following. For a slow periodic data stream as shown in Figure 5-16(a), suppose that the first rising edge of the data appears at the boundary between the states III and IV. Then the second rising edge crosses the boundary between the states IV and I and appears in state I. The state transition rotated from state IV to I would be detected. This state transition indicates that the clock is faster than quarter the data rate and frequency down pulses are generated. For a fast data periodic data as shown in Figure 5-16(a), the first rising edge appears at the boundary between the states I and II. Then the second rising edge crosses the boundary between the states IV and I and appears in state IV. The last state transition indicates that the clock is slower than quarter the data rate and frequency up pulses are generated. The truth table 5-1 represents the states transition of the proposed quarter-rate DQFD.

Q_5Q_6 → Q_7Q_8 ↓	State I 10	State II 11	State III 01	State IV 00
State I (10)	X	X	DOWN	DOWN
State II (11)	X	X	X	DOWN
State III (01)	UP	X	X	X
State IV (00)	UP	UP	X	X

Table 5-3: Truth table of the proposed quarter-rate DQFD.

To find the required combinational logics circuit, we write the equations describing the frequency up and down pulses. From table 5-3:

$$Freq.\_DOWN = Q_7.\overline{Q_8}.\overline{Q_5} + \overline{Q_5}.\overline{Q_6}.Q = \overline{Q_5}.Q_7.\overline{Q_6Q_8}$$

$$Freq.\_UP = Q_5.\overline{Q_6}.\overline{Q_7} + \overline{Q_7}.\overline{Q_8}.Q_5 = Q_5.\overline{Q_7}.\overline{Q_6Q_8}$$

(5.14)

From the above equations, the implementation of the required combinational logic circuit is shown in Figure 5-17.
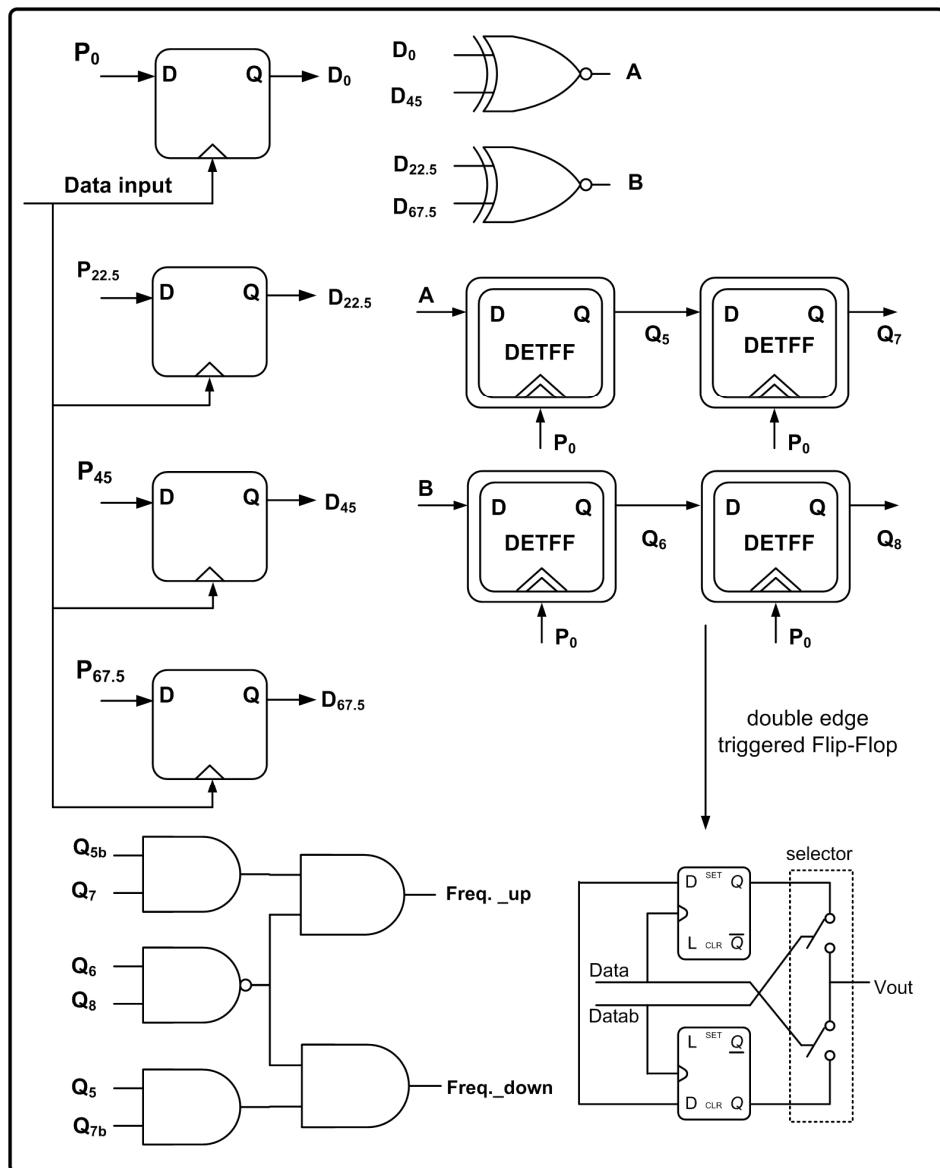
Figure 5-17: Schematic of the proposed quarter-rate DQFD.

5.6 Charge-Pump Principle

A better understanding of a three-state charge pump is achieved when it is considered in conjunction with a periodic signal based conventional three-state phase and frequency detector as shown in Figure 5-18. The charge pump circuit itself consists of two switched current sources controlled by the signals Q_A and Q_B that are issued from the PFD. In this circuit, the switch S_1 and the current source I_1 are implemented using a PMOSFET transistor, whereas the switch S_2 and the current source I_2 are implemented with an NMOSFET transistor. If a pulse of width T appears on Q_A, I_1 deposits a charge equal to $I.T$ on the capacitor C_p.
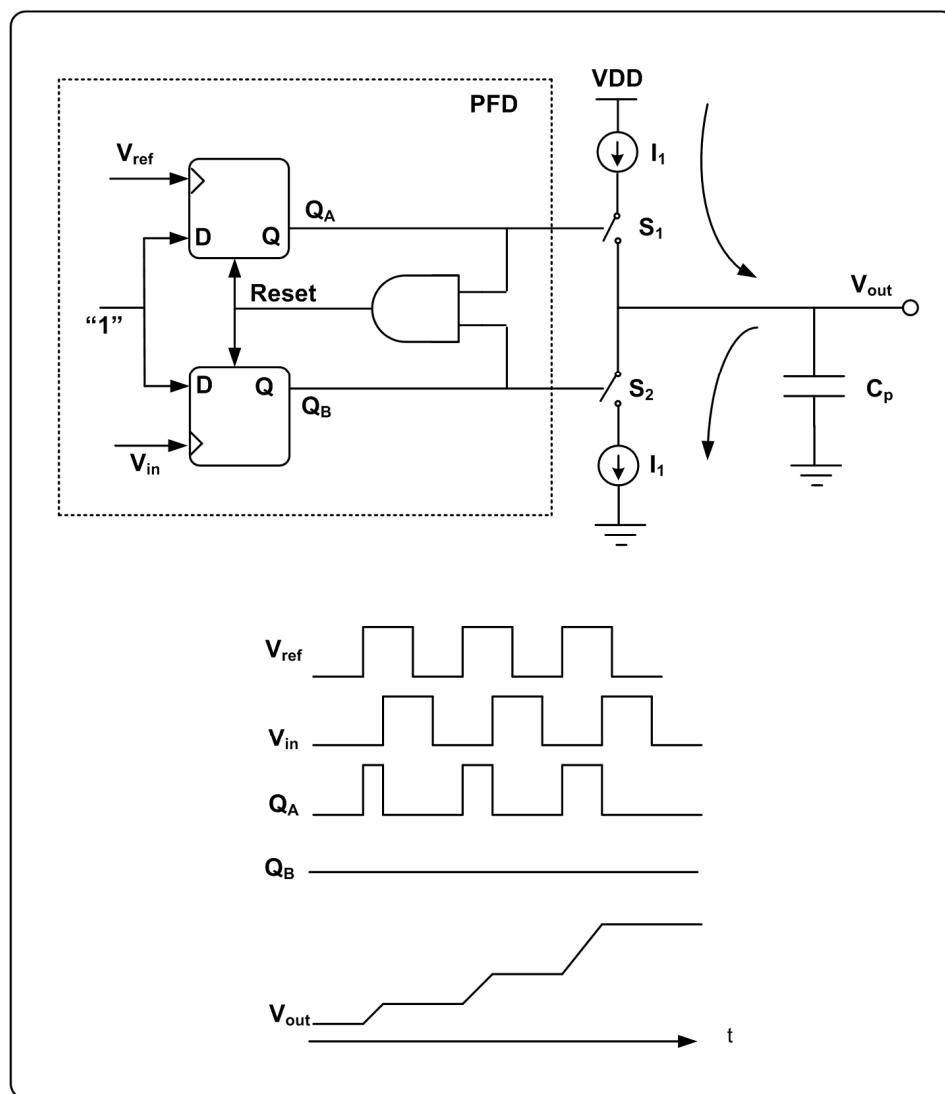
Figure 5-18: Charge pump and its output signal in conjunction with a periodic signal based phase and frequency detector.

5.7 Charge-Pump and Loop Filter Circuit Design

As shown in Figure 5-19, the phase and frequency detectors output signals control a charge-pump and a loop filter (R and C) to provide the required input differential voltage (V_{up}-V_{down}) to the VCO. Transistors M_1 and M_2 are controlled by the phase detector output signals and transistors M_3 and M_4 are controlled by the frequency detector output signals. These transistors determine the current sources I_{pd} and I_{fd} through the loop filter. Transistors M_5 and M_6 provide the pull-up current I_c. The relationship between the magnitudes of the above current sources is given by

$$I_c \gg I_{fd} > I_{pd} \qquad\qquad (5.15)$$

The common mode voltage of V_{up} and V_{down} is compared to a reference voltage V_{ref} by the comparator. If the common mode voltage level is increased, the drain currents of transistors M_7 and M_8 are decreased and the common mode voltage is pulled up by the current source I_c.
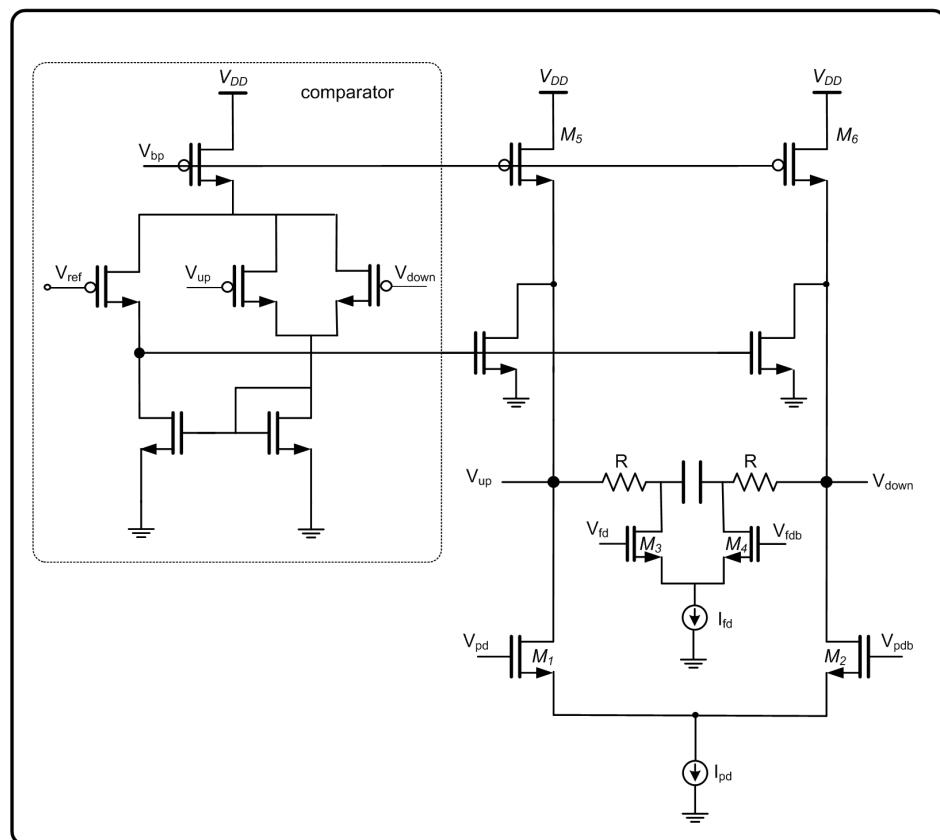
Figure 5-19: Schematic of the charge-pump and loop filter.

6 PLL-Based CDR Circuit Implementation

The results of design and transistor level simulation of a novel architecture for PLL-based clock and data recovery (CDR) circuit are presented in this chapter. The proposed PLL-based CDR is a referenceless quarter-rate design (i.e., the clock frequency is quarter the input data rate), comprising a novel quarter-rate phase detector, a novel quarter-rate frequency detector and can be used in a deserializer as part of the Serializer/Deserializer (SerDes) device usually utilized in inter-chip communication networks [34]. The proposed CDR circuit is designed in a standard $0.13\,\mu\mathrm{m}$ CMOS technology, and simulated at transistor level to verify its accuracy as well as to evaluate its characteristics and performances.

6.1 Voltage Controlled Oscillator

For proper operation of the phase and frequency detectors, eight clock signals and their complements (separated by 22.5°) are required. Due to its wide tuning range an eight-stage ring oscillator structure was chosen. As shown in Figure 6-1, the VCO consists of eight stages, each one of them comprising a delay cell and a control circuit for generating differential control voltages V_{inc} and V_{dec} for the delay cell. The controlling signals V_{inc} and V_{dec} can be viewed as differential control lines and hence providing higher noise immunity to the VCO controlled input. The dimensions of transistor $M7$ and the voltage at its gate V_{bias} should be carefully adjusted such that proper VCO gain, linearity and tuning range will be obtained. The tuning technique in this architecture is already described in 5.3.2 and based on the concept of bias current controlled negative resistance [64]. As the bias current of the cross-coupled pair of transistors ($M3$ and $M4$) increases, their negative small-signal resistance becomes less negative; hence the total resistance seen by the outputs nodes *out* and *outb* increase, thereby lowering the oscillation frequency. The eight clock signals generated by the VCO are shown in Figure 6-2(a).
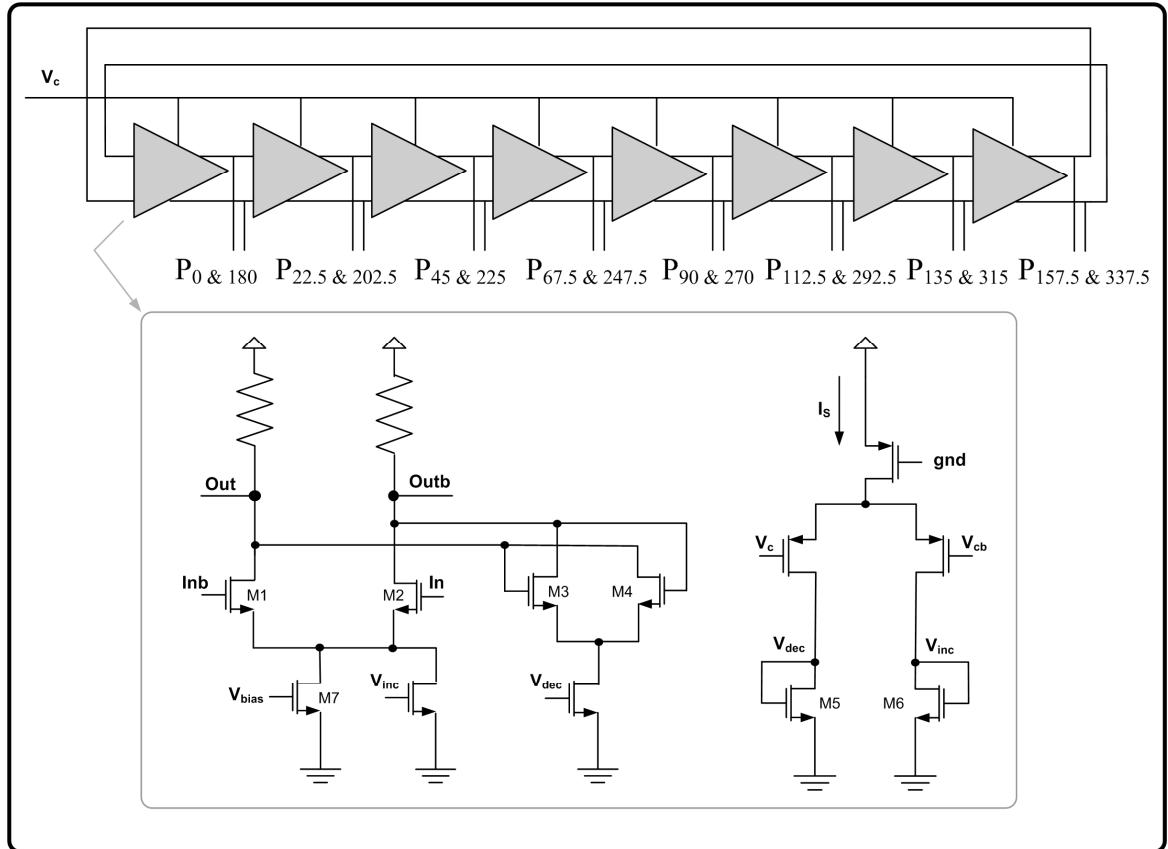
Figure 6-1: The eight-stage voltage-controlled ring oscillator.

In summary, and based on the post layout simulation results, the proposed VCO has the following features:

- A $\pm$ 18% tuning range around the centre frequency 2.75 GHz.

- A conversion gain of 492 MHz/V.

- Generating eight clock signals separated by a required phase shift of 22.5^{o}.

- The generated clock signals are differential which gives it a good supply and substrate noise rejection and yield 50% duty cycle in the oscillating signals.
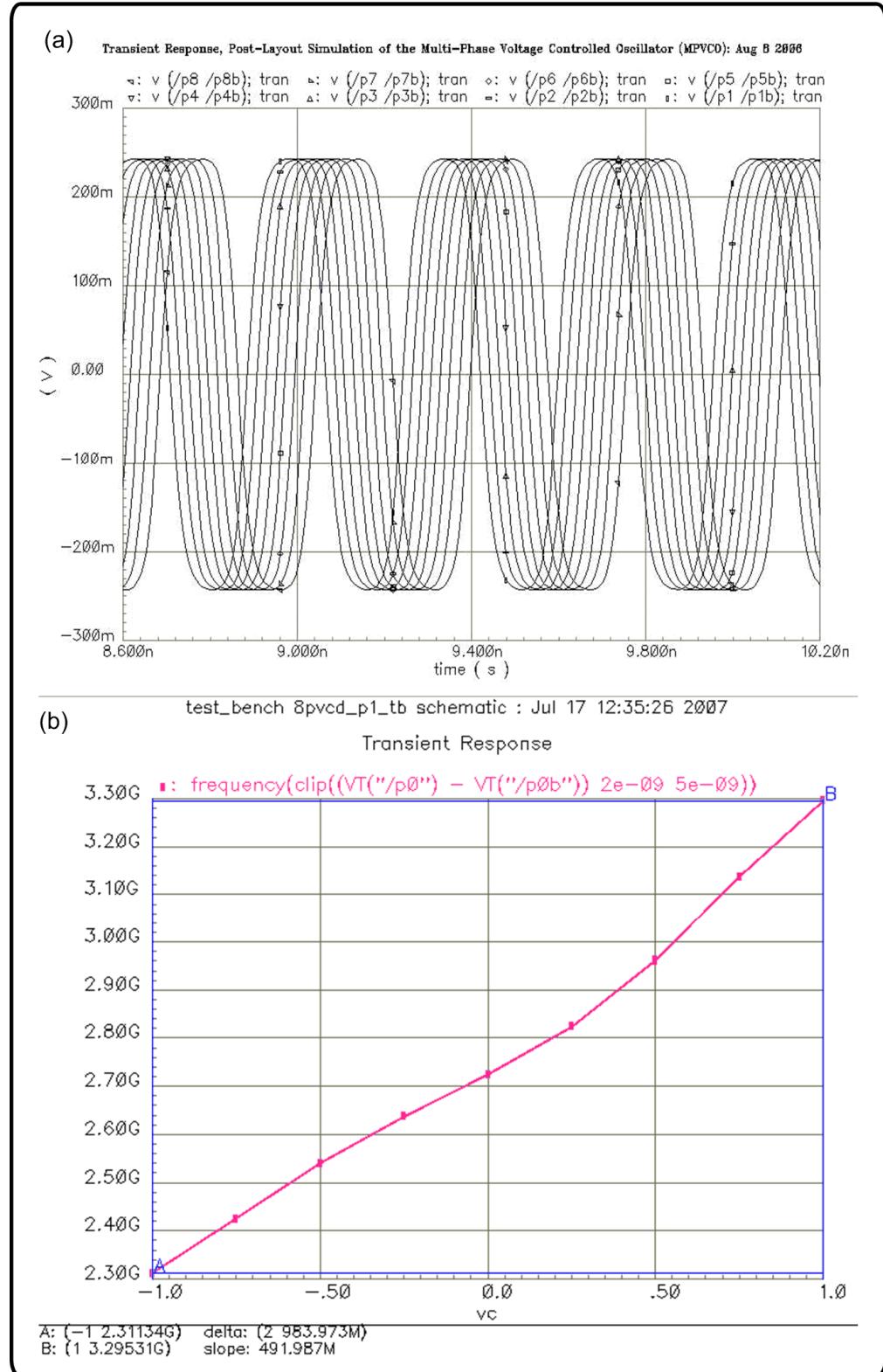
Figure 6-2: Post-layout simulation, (a) the clock signals generated by the VCO and, (b) the VCO's conversion gain.

In any MOSFET technology, the parameters of individual components such as capacitors, resistors and transistors vary from wafer to wafer (inter-process variations) and from die to die (intra-process variations). The random distribution of identically drawn devices is caused by the variations in process parameters, e.g., impurity concentration densities, oxide thicknesses and diffusion depths. The change in the process parameters causes electrical parameters to vary, such as sheet resistance, capacitance and transistors threshold voltage, which in turn alters the performance of circuits from their desired values. Figure 6-3 illustrates the dependence of frequency and amplitude of oscillation of the VCO to the corners process of the transistors and resistors. The maximum relative change (i.e., max($\Delta f/f$)) of frequency due to the variations of the resistor corner process from the minimum to maximum corner is about 21% of the centre frequency, whereas the same type of change due to the variations of the transistor corner process is less than 1% of the centre frequency. Therefore the proposed VCO is more sensitive to the resistor than the transistor corner process. Figure 6-4 illustrates the layout of the proposed VCO.
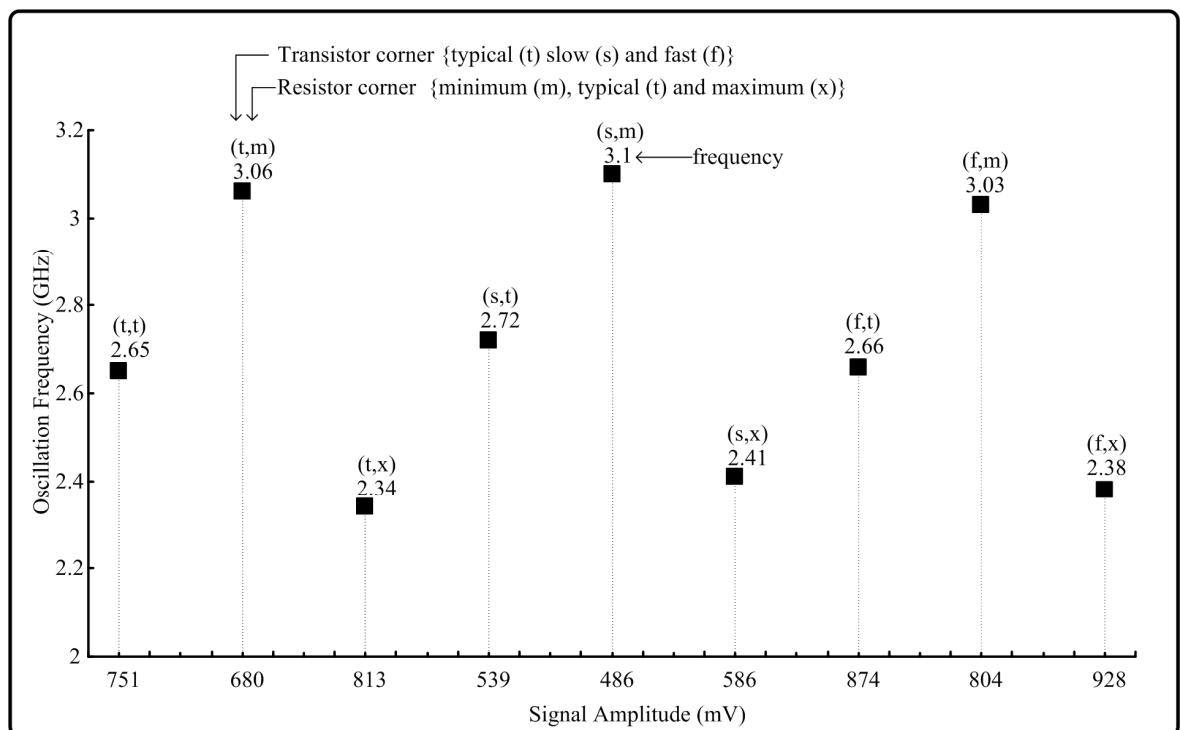

Figure 6-3: Process variations effects on the frequency centre and amplitude of the VCO.
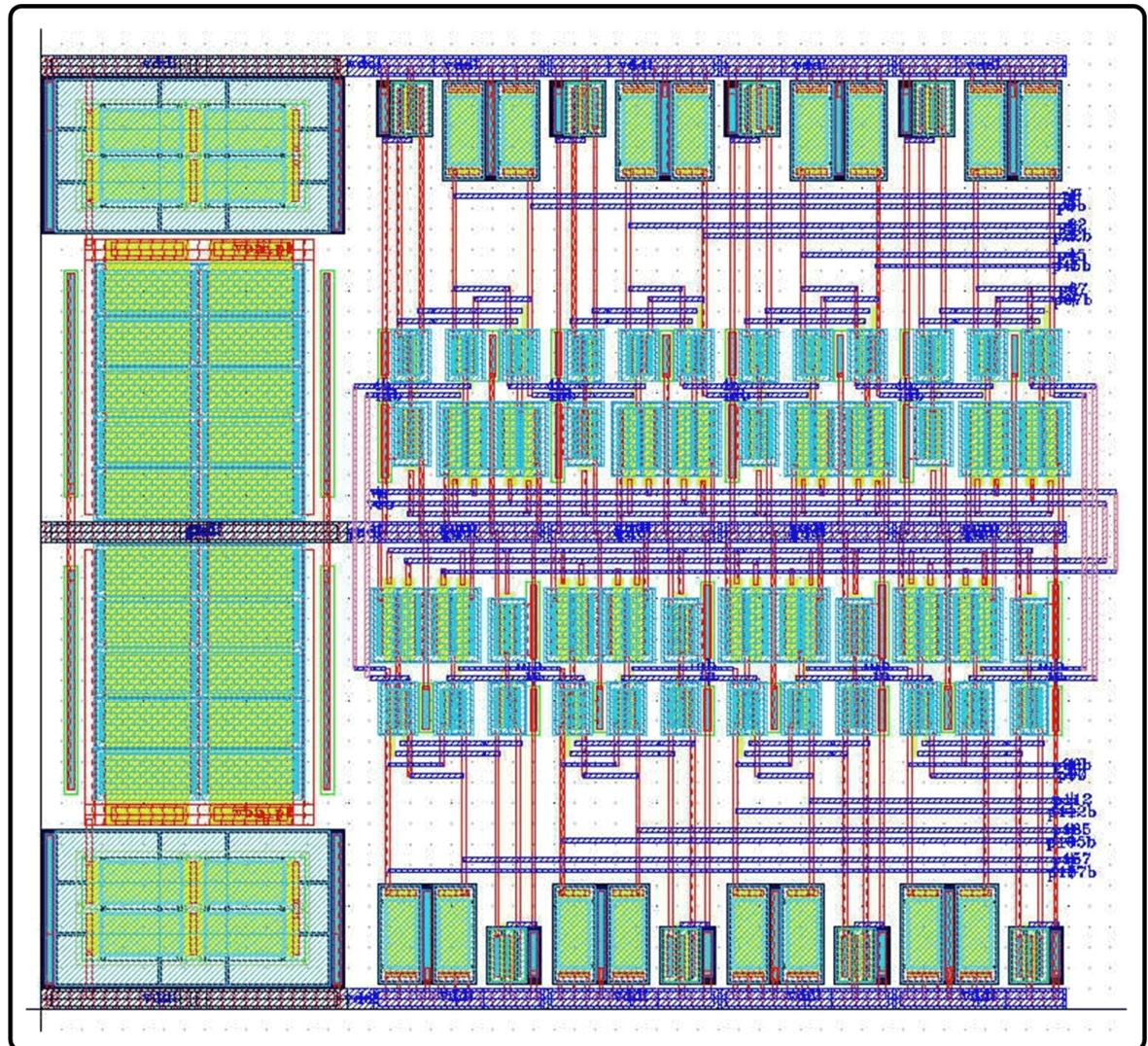
Figure 6-4: Layout of the proposed VCO.

6.2 Novel Quarter-Rate Three-State Early-Late Phase-Detector

The jitter generation of a PLL-based CDR is a function of the phase detector propagation delay [39]; hence it is desired to have a phase detector with a low propagation delay to improve the jitter performances of the PLL- based CDR circuit. Though a two states phase detector has generally less propagation delay [32] but its two states behaviour increases the data dependant jitter for long periods of data without any transitions. The proposed phase detector is shown in Figure 6-5; it is a three-state quarter-rate early-late based design (i.e. the data rate is four times the clock frequency). The operation of this phase detector is already explained in section 5-4. Figure 6-6 shows the phase detector output when the clock signal is leading the data by 10 ps. The layout of the proposed phase detector is shown in Figure 6-7.
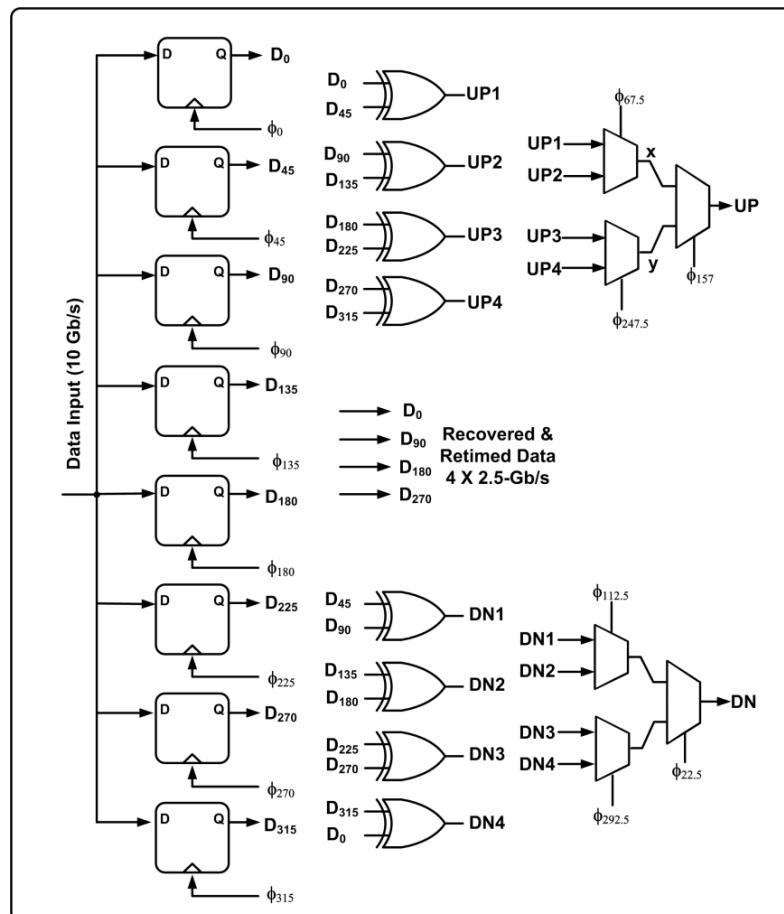
Figure 6-5: The proposed quarter-rate early-late type phase detector (D0, D90, D180 and D270) are the demultiplexed recovered data.
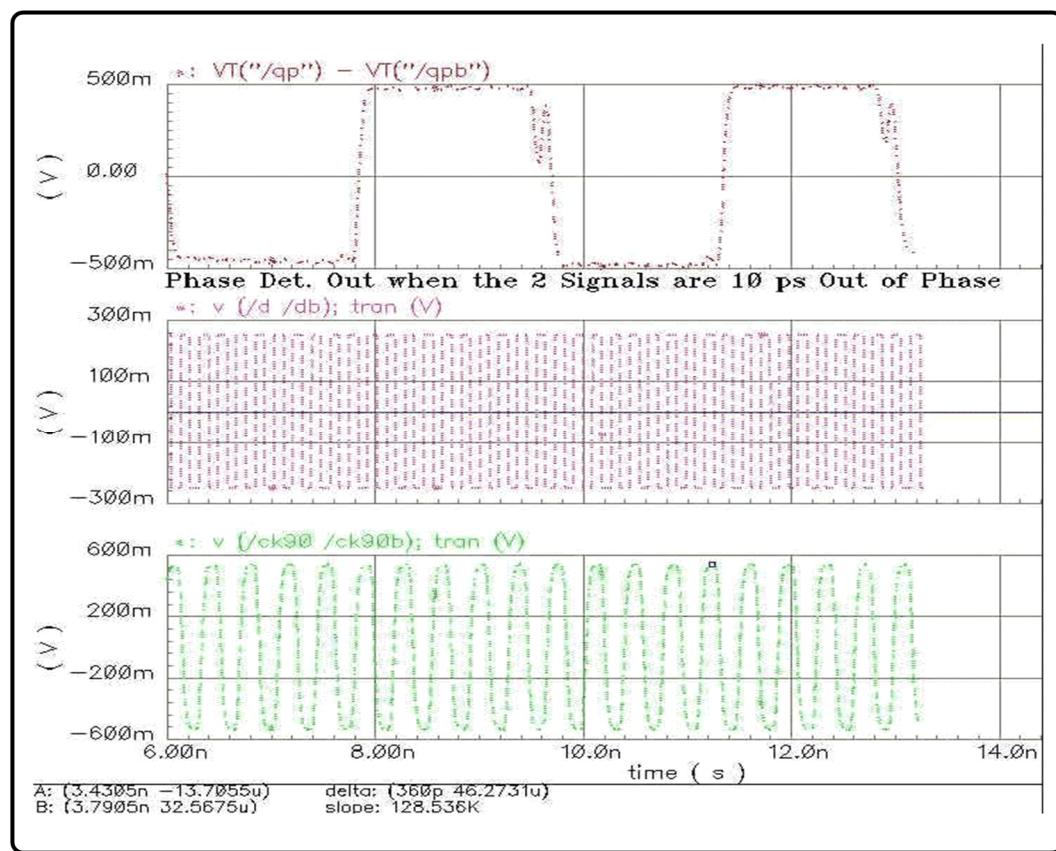
Figure 6-6: Phase detector output for 10 ps out of phase two signals at its input.
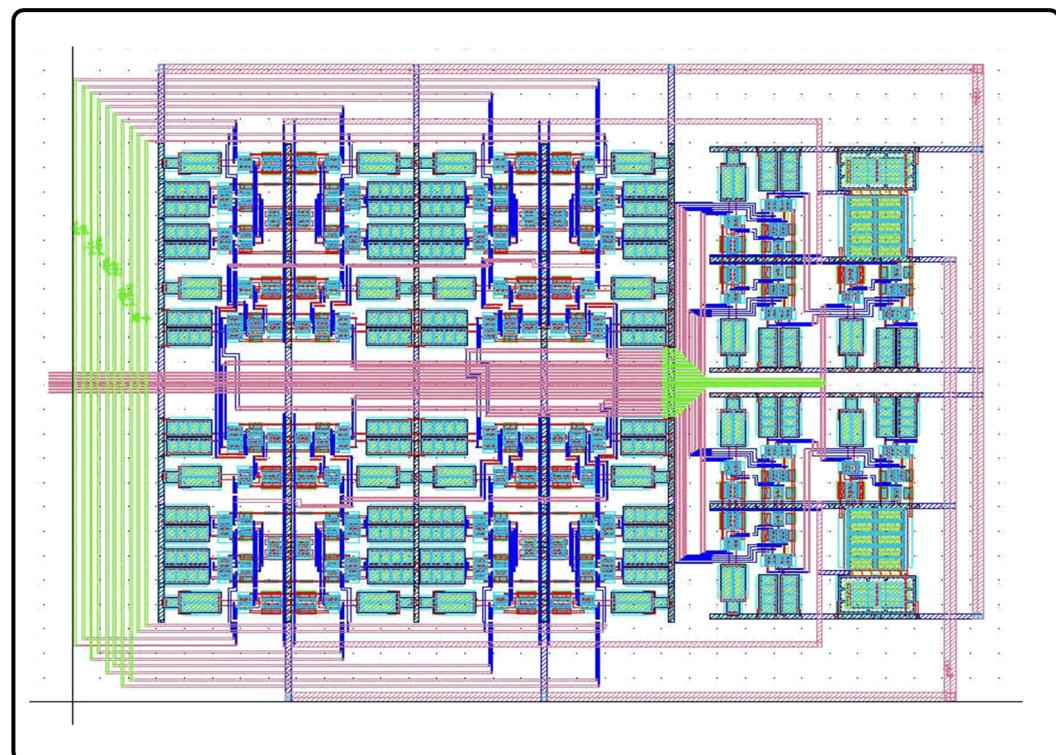
Figure 6-7: Layout of the proposed phase detector.

6.3 Novel   Quarter-Rate   Digital   Quadricorrelator
Frequency Detector

The communication standards generally require a small loop PLL-Based CDR. This results in a narrow capture range, hence a data-based frequency detector is required to increase the capture range. Once the frequency has been acquired the frequency detector must be disabled, hence generating no outputs, and then the phase detector must automatically take over to adjust the clock phase to the data properly [34]. The proposed frequency detector is a quarter-rate (i.e. the data rate is four times the clock frequency) is shown in Figure 6-8, its operation Principle is explained in section 5.5. As illustrated in Figure 6-9, the frequency detector generates a train of *freq_down* pulses when the clock frequency (f_{ck}) is higher than one fourth the input data rate (i.e., $f_{ck} > 0.25$ data_rate).
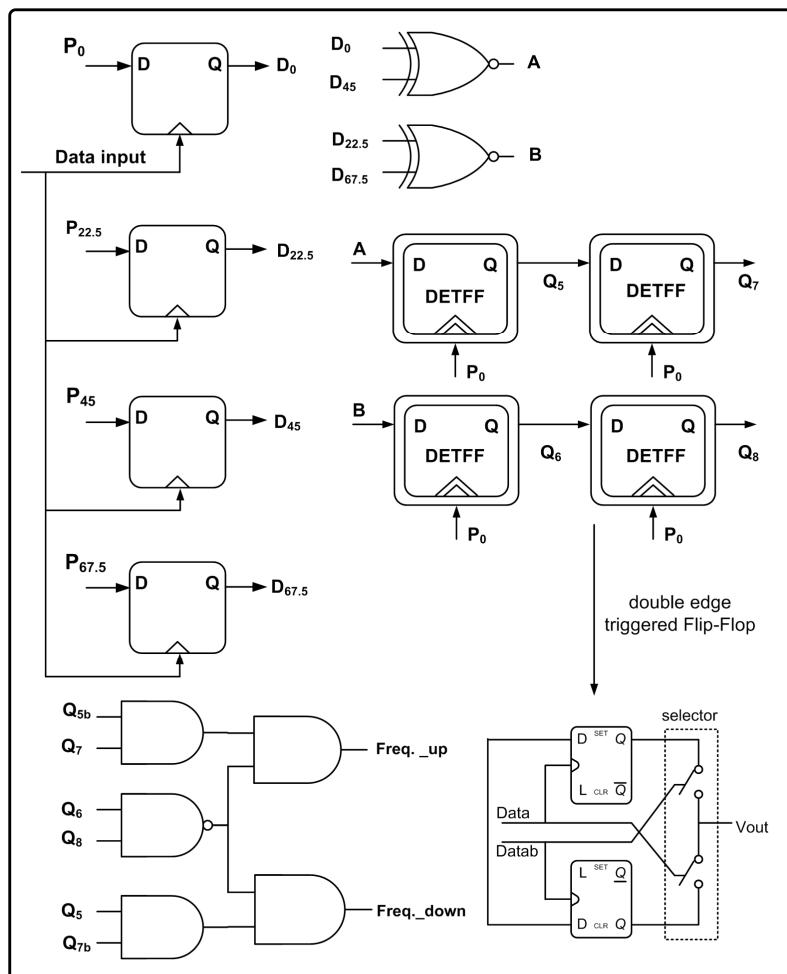
Figure 6-8: Architecture of the proposed frequency detector.
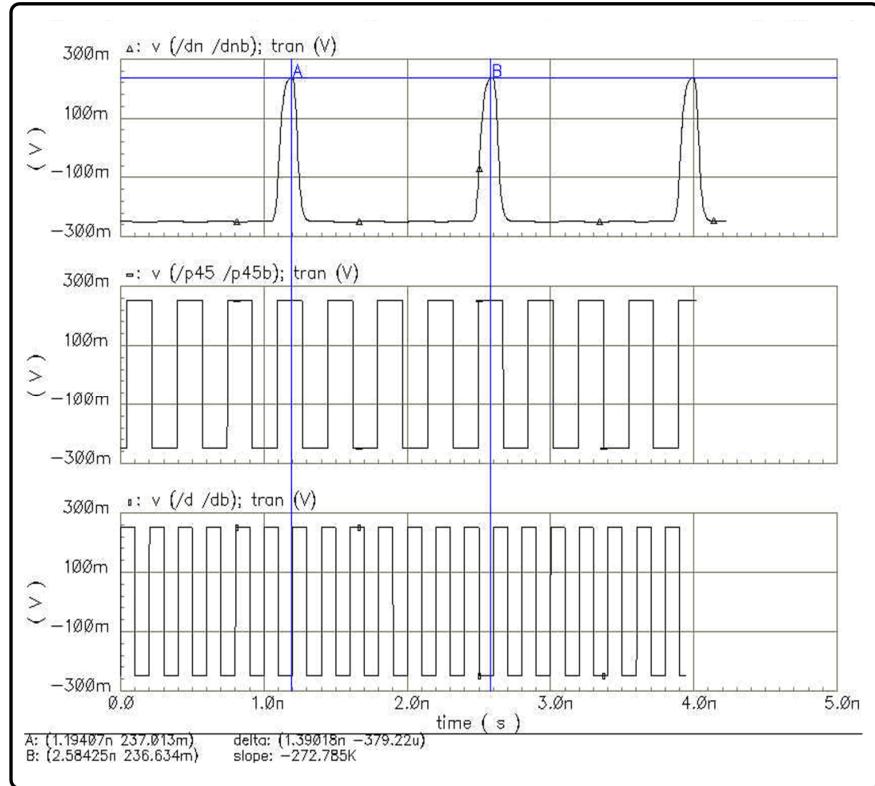
Figure 6-9: Frequency down pulses generated when the frequency of the VCO is higher that the frequency of the incoming data.
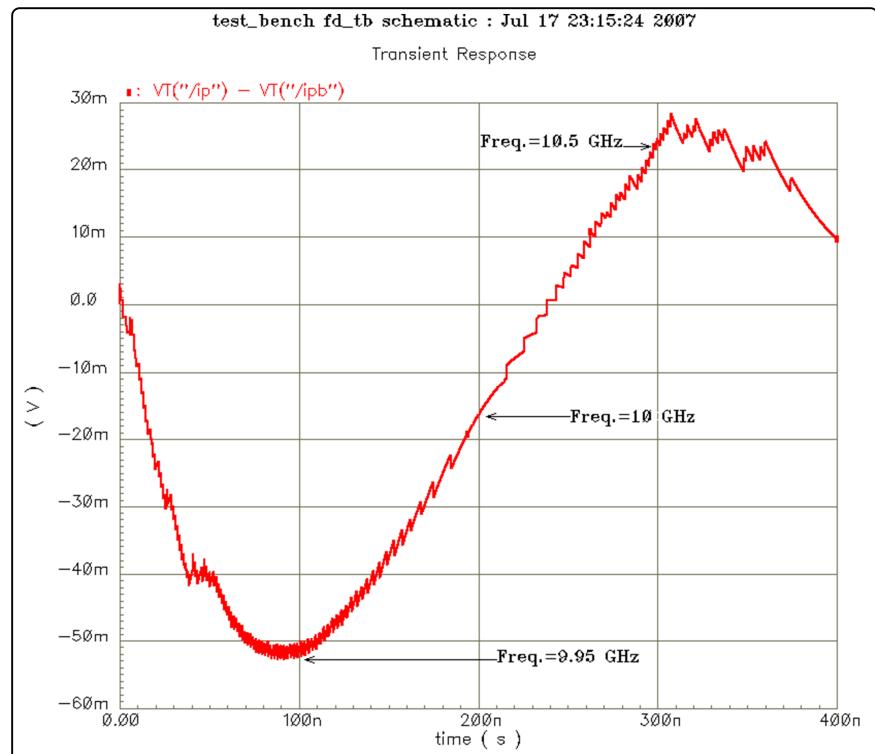
Figure 6-10: Operating range of the proposed frequency detector.

To determine the operating range of the proposed frequency detector, we apply two periodic signals to its inputs. One of them is considered as a reference and has a quarter-rate constant frequency (2.5 GHz) and the other signal is swept in frequency at a constant rate of 5 MHz/ns starting from 9 GHz and stopping at 11 GHz. The transfer curve of the proposed frequency detector is illustrated on Figure 6-10. It exhibits a 1 GHz operating range around the nominal frequency of 10 GHz.
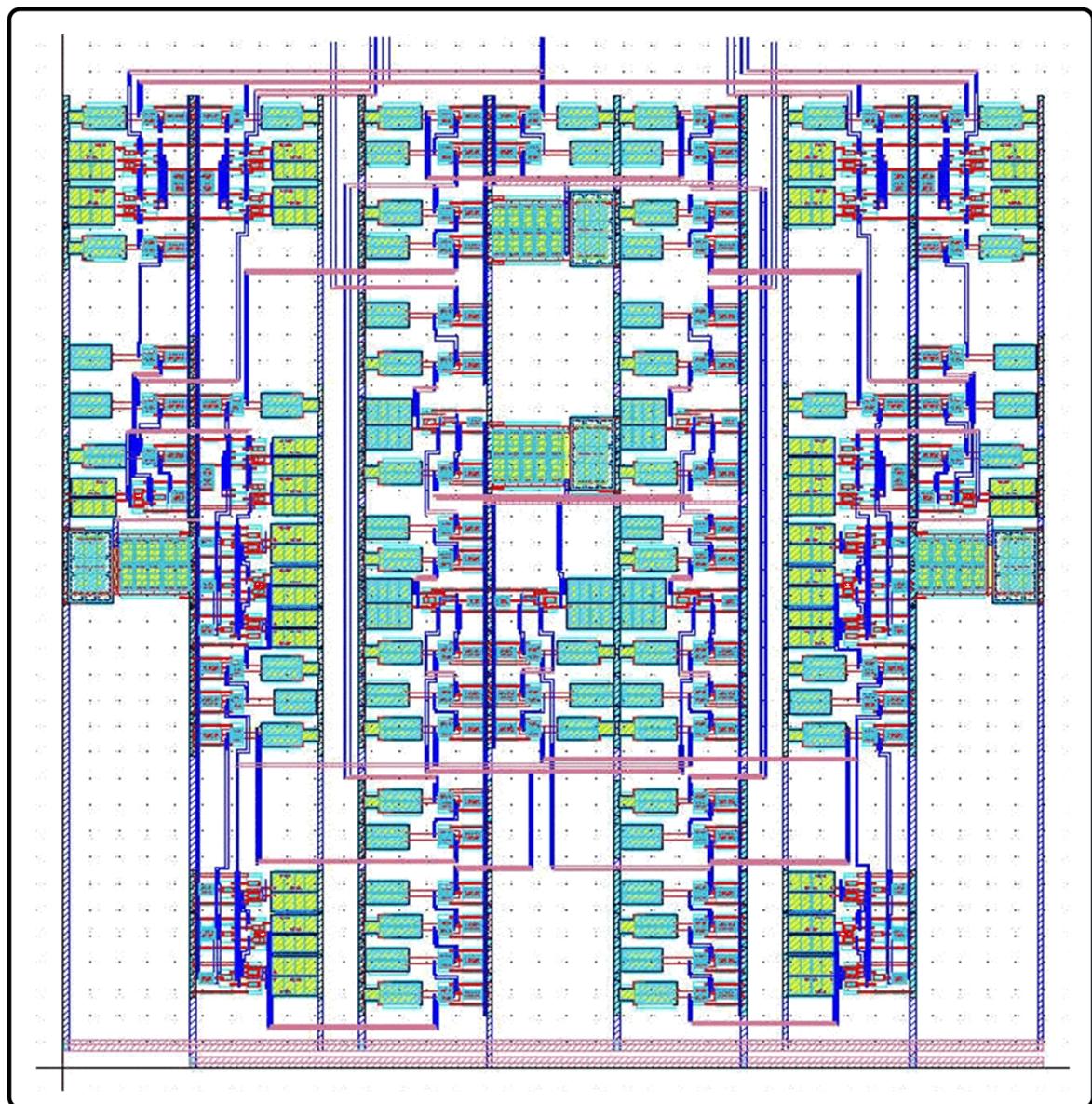
Figure 6-11: Layout of the proposed frequency detector.

6.4 Transistor Level Simulation of the Proposed PLL-Based Quarter-Rate Clock and Data Recovery Circuit

The proposed quarter rate PLL-CDR has been designed in UMC 0.13μm CMOS technology and simulated at transistor level using the schematic view of the CDR circuit [64]. Since we are using a quarter-rate based CDR topology, the input data rate should be four times the VCO centre frequency. Based on the VCO schematic simulation characteristic curve of Figure 6-12, the VCO centre frequency is about 5.5 GHz, therefore the data rate should be about 22 Gb/s. As shown in Figure 6-13, the input data signal is PRBS (N=32) with a data rate of 21.85 Gb/s. The data rate is 160 MHz below the required centre frequency of the VCO (i.e. 5.35 GHz). Figure 6-14(b), illustrates the transient simulation results of the circuit locking process, the PLL reaches the steady state within 500 ns. As shown in Figure 6-14(a), once the desired frequency has been acquired the frequency detector is disabled, hence generating no outputs. Table 6-1 summarizes the PLL-CDR circuits performances based on schematic view simulation results.
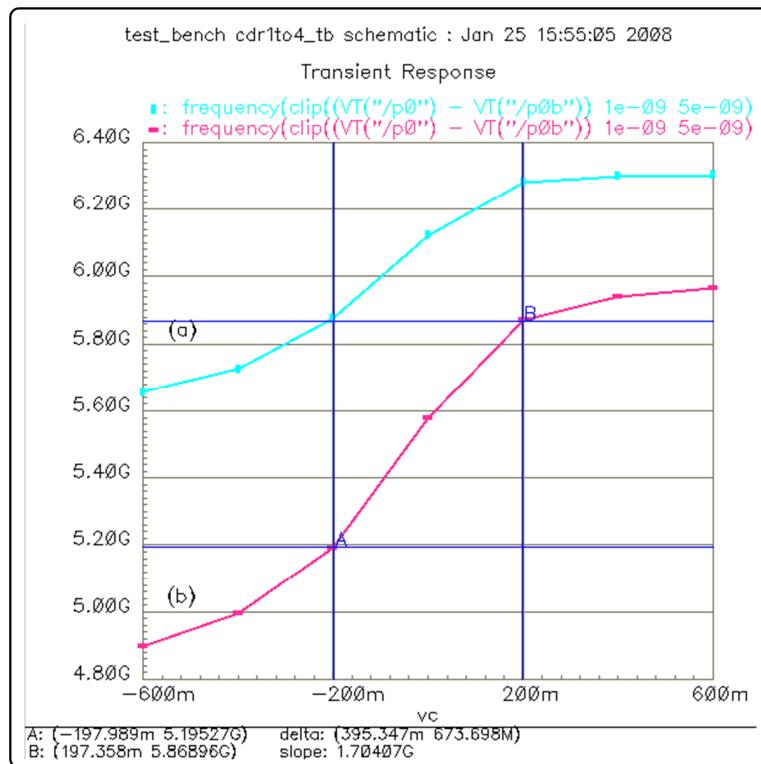
Figure 6-12: Frequency tuning range of the schematic view of the VCO for (a) V$_{bias}$ = 0.75 V and (b) V$_{bias}$ = 0.6V.
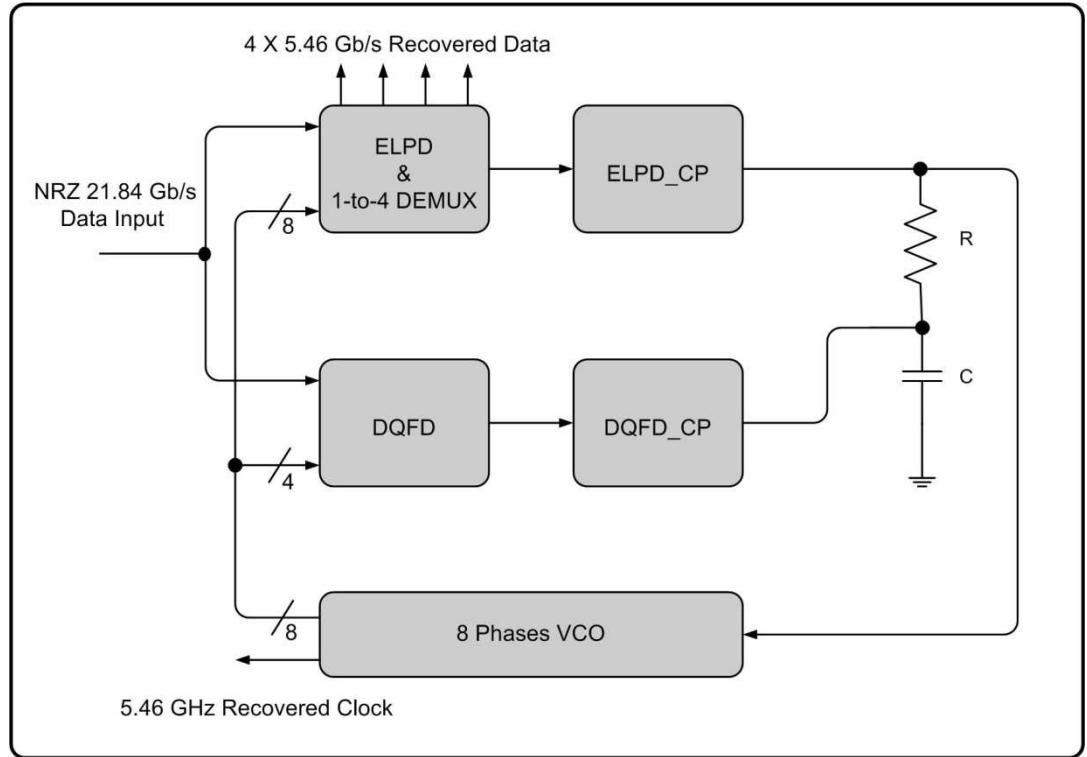
Figure 6-13: Block diagram of the proposed quarter-rate PLL-Based CDR circuit.

Table 6-2 : CDR characteristics table.

Parameter	Simulation
Input data rate	21.84 Gb/s
PRBS	2^{32}-1
VCO frequency range	4.9-6 GHz
VCO conversion gain	1.7 GHz/V
CDR bandwidth	3 MHz
Lock-in time	750 ns
Pull-in range	5.284-5.71 GHz
CDR power	97 mW
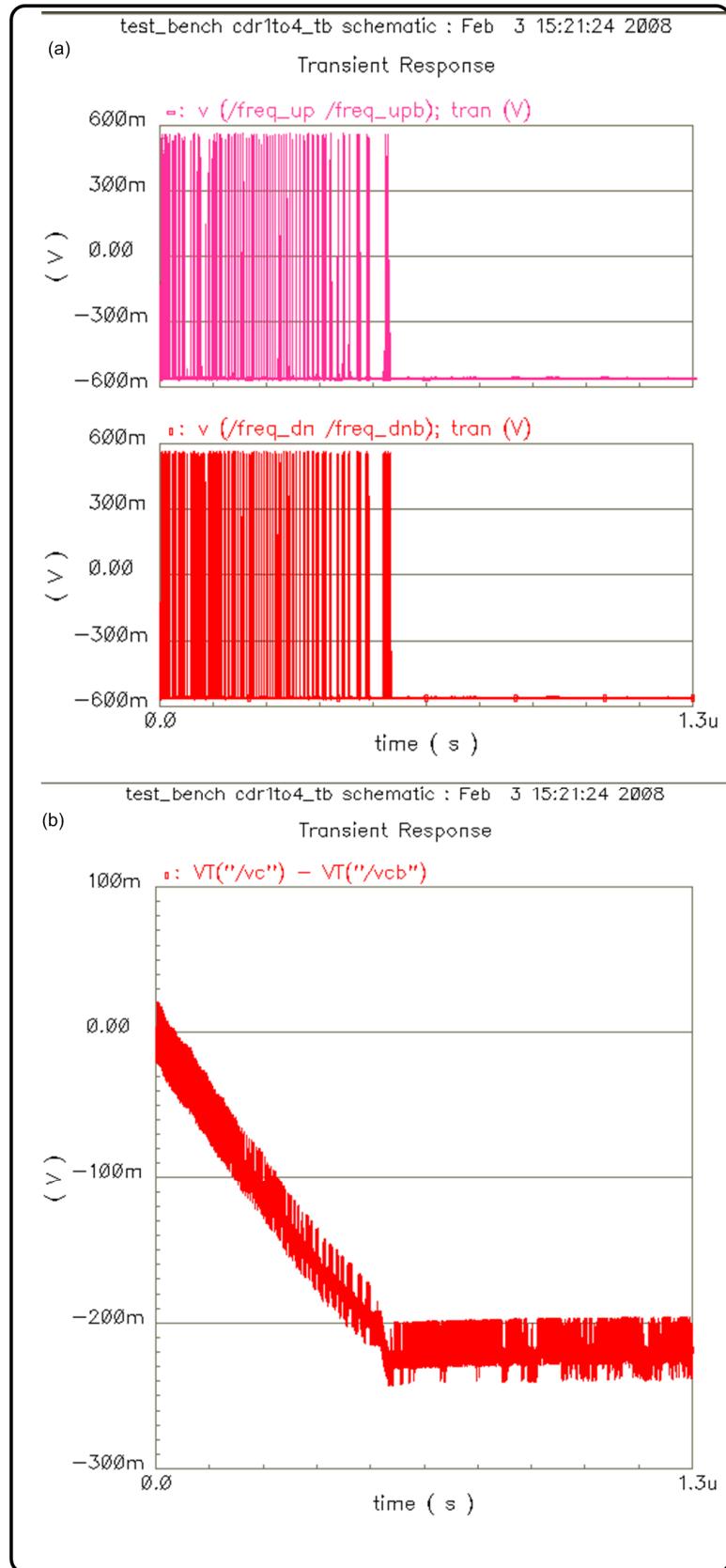
Figure 6-14: Frequency detector outputs (a) and output of the low pass filter showing the PLL locking process.

Based on the schematic view simulation results illustrated on Figure 6-14 (a) and (b), the quarter-rate PLL-based CDR is a working concept. Although the schematic view of the CDR circuit is working at around 22 Gb/s data rate, the fabricated chip is expected to work at about 10 Gb/s, because the VCO centre frequency is expected to be lower than the schematic one due to the presence of parasitic capacitors and resistors associated to the fabricated chip.
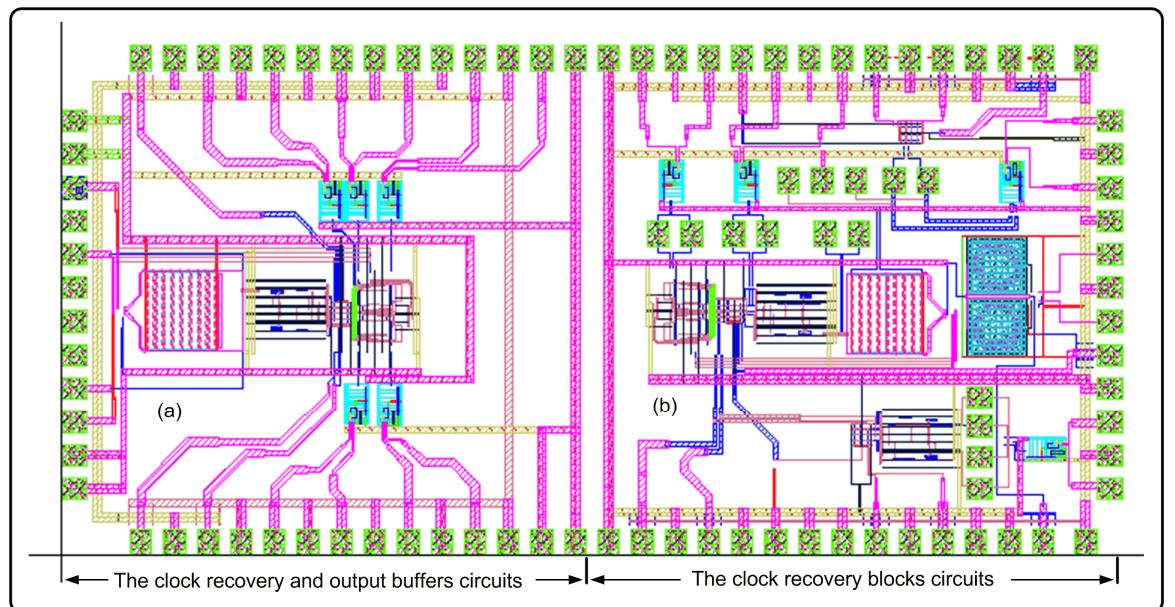
Figure 6-15: Layout of the complete PLL-Based CDR circuit and its constituting circuits.

As shown in Figure 6-15, the design occupies an area of 920 μm x 315 μm and is expected to dissipate approximately 97 mW, excluding the output buffers, at a supply voltage of 1.2 V according to the transistor level simulation results [64].

7 Conclusion and Future Work

In this thesis, we considered the design, modelling and implementation of a referenceless quarter-rate PLL-based clock and data recovery integrated circuit. Up to a certain extent this has been achieved at transistor level design, simulation and Verilog-A modelling, despite the fact that the chip was not working. This chapter will review the findings of this study and present some suggestions for future work.

7.1 Conclusions

Serial data communications are widely used in today's data communication systems such as fibre optic and wireline based communication links, they as well as are aggressively substituting the communication based on the source synchronous parallel links and the multi-bit parallel bus because they are more power and space efficient. Higher volume of transmitted data requires higher and higher bandwidth. CMOS technology is largely used and highly desired for monolithic implementation because of its advantages of low cost and wide availability. The primary goal of this dissertation is to implement a new concept of a clock and data recovery circuit in 130nm CMOS technology for 10 Gb/s operation, modelling it with the Verilog-A language and ultimately using it as part of the receiver in a chip-to-chip serial link transceiver, another advantage of the proposed concept is that, the serial data stream is inherently 1-to-4 demultiplexed.

The existing works of Gb/s clock and data recovery circuits are full, half data rate, reference or referenceless based architectures. The proposed architecture of this circuit is a referenceless quarter-rate PLL-based clock and data recovery circuit, it means that first, the circuit does not require a reference clock signal because it is internally generated from the VCO and, second for a 10 Gb/s incoming data rate, the internal parts of the circuit (i.e. VCO, DFFs and primitive gates) are actually working at a clock speed of 2.5 GHz. Working at quarter-rate relaxes the timing constraints of the dynamic elements and the static gates as well as reducing the dynamic power consumption resulting from the switching activities in the circuits.

The proposed topology contains two loops operating independently, the phase and frequency-locked loops; the frequency detector is for frequency acquisition only. Once the frequency lock is acquired (i.e. the clock frequency is equal to quarter of the data rate), the frequency detector is disabled and the phase detector will take over to properly adjust the clock phase with respect to the data stream (i.e. the clock edges occurs in the middle of the data bit). When the lock is lost, the frequency detector is automatically activated.

The proposed quarter-rate frequency detector has two advantages, first because the frequency detector is completely disabled when the lock is acquired, it does not contribute any jitter to the system, second because the gain or the operating range of the frequency detector is reasonably large, hence the process of frequency acquisition is faster while the loop dynamics of the phase locked loop and the jitter performance of the system are not disturbed. From the transistor level simulations, the frequency detector demonstrated a detecting range ±25% of the data rate. The proposed phase detector is a symmetric quarter-rate and nonlinear; because it is nonlinear, hence it has a large gain and therefore it is suitable for Gb/s data rate. An 8-stage differential ring oscillator was used for the voltage controlled oscillator (VCO). The differential architecture is widely used because it rejects noises from both the power lines and the substrate. Eight phases and their complements separated by 22.5^o are produced from the 8-stage ring oscillator and ready to use for proper operation of the phase and frequency detector. The chip was designed, transistor level simulated, modelled with the Verilog-A language and fabricated using the CMOS UMC 130nm technology process. The simulation results showed that the circuit has excellent performance in term of locking time (500 ns), small silicon area and power consumption (97 mW), having short acquisition time reduce the number of preamble or training bits required and results in higher efficiency. Unfortunately the fabricated chip was not working because the VCO was not generating any signal normally required for the proper operation of the phase and frequency detector. The VCO was not oscillating because the measured DC voltage level at the output of the VCO was much lower (0.2 V) than the simulated and expected value (0.8 V). Since the VCO architecture is a current mode based design, hence between the power supply (VDD) and the ground (GND), there is one load resistor cascoded (stacked) with two stacked transistors below it. Having low DC voltage level at the output of the load resistor makes the bottom two transistors below it completely off and hence preventing the VCO from oscillating.

7.2 Future Work

In semiconductor industries and research, the most important figure of merit of any new circuit design and system architecture is a working silicon implementation of the proposed circuit. Although our proposed concept or approach of the PLL-based clock and data recovery is a working concept at transistor level simulation and Verilog-A modelling, but we still need to have working silicon of such new concept. For a better chance of having a successful implementation in the future, we propose the following steps:

1. As a preliminary proof of concept the proposed idea could be implemented using an FPGA such as Altera DE2-70 or other.

2. Implementation of the new concept in a widely used and a well reputed technology such as Austria Mikro Systems (AMS) or Taiwan Semiconductor Microelectronic Corporation (TSMC).

3. Implementing of the new idea at lower data rate (e.g. 1 Gb/s) using the rail-to-rail CMOS logic and using as much as possible primitive logic cells and dynamic gates already available in the libraries provided by AMS or TSMC. Using the rail-to-rail logic alleviates the problem of proper biasing normally encountered in current mode logic.

4. Once the concept is proved to work in silicon, at a lower data rate using rail-to-rail logic, we can eventually move forward and implement the idea using the current mode logic for higher data rate (e.g. 10 Gb/s).

References

1. M. Horowitz et al., "High-Speed Electrical Signalling: Overview and Limitation", IEEE Micro, vol. 18, 12-24, Jan. /Feb. 1998.

2. K. Lee, S-J Lee, and H-J Yoo, "SILENT: Serialized Low Energy Transmission Coding for On-Chip Interconnection Networks", Proceedings of the 2004 IEE/ACM International Conference On Computer-Aided Design, 448-415, 2004.

3. S. Furber and J. Bainbridge, "Future Trends in SoC Interconnect", International Symposium on System-On-Chip, 183-186, Nov. 2005.

4. N. McKewon et al., "Tiny Tera: A Packet Switch Core", IEEE Micro, vol. 17, no.1, 26-33, Jan.-Feb. 1997.

5. F. Tobagi, "Fast Packet Switch Architectures for Broadband Integrated Services Digital Networks", Proceedings of the IEEE, vol. 78, no. 1, 133-167, Jan. 1990.

6. E. Rees et al., "A Phase-Tolerant 3.8 Gb/s Data-Communication Router for a Multiprocessor Supercomputer Backplane", IEEE International Solid-State Circuits Conference Digest of technical papers, 296-297, Feb. 1994

7. J. Kuskin et al., "The Stanford FLASH Multiprocessor", Proceedings of the 21st International Symposium on Computer Architecture, 302-313, Apr. 1994.

8. Scalable Coherent Interface (SCI), IEEE Standard 1596.

9. M. Galles et al., "Spider: A High-Speed Network Interconnect", IEEE Micro, vol. 17, no. 1, 34-39, Jan.-Feb. 1997.

10. A. Charlesworth, "Starfire: Extending the SMP Envelope", IEEE Micro, vol. 18, no. 1, 39-49, Jan.-Feb. 1998.

11. T. Takahashi et al. "A CMOS Gate Array With 600 Mb/s Simultaneous Bidirectional I/O Circuits", IEEE Journal of Solid-State Circuits, vol. 30, no. 12, 1544-1546, Dec. 1995.

12. K. Lee et al., "A Jitter-Tolerant 4.5 Gb/s CMOS Interconnect for Digital Display", IEEE International Solid-State circuits Conference Digest of Technical papers, 310-311, Feb. 1998

13. L.I. Anderson et al., "Silicon Bipolar Chipset for SONET/SDH 10-Gb/s Fiber-Optic Communication Links", IEEE Journal of Solid-State Circuits, vol. 30, no. 3, 210-218, Mar. 1995.

14. Y.M. Greshishchev et al., " A Fully Integrated SiGe Receiver IC for 10-Gb/s Data Rate", IEEE Journal of Solid-State Circuits, vol. 35, no. 12, 1949-57, Dec. 2000.

15. M. Meghelli et al., "SiGe BiCMOS 3.3-V Clock and Data Recovery Circuits", IEEE Journal of Solid-State Circuits, vol. 35, no. 12, 1992-5, Dec. 2000.

16. A. Momtaz et al., "Fully-Integrated SONET OC48 Transceiver in Standard CMOS", IEEE Journal of Solid-State Circuits Conference Digest of Technical Papers, 76-77, Feb. 2001.

17. S. Shioiri et al., "A 10 Gb/s SiGe Framer/Demultiplexer for SDH Systems", IEEE Journal of Solid-State Circuits Conference, 202-203, 1998.

18. A. X. Widmer, "Method of Coding to Minimize Delay at a Communication Node", U.S. patent 4665517, Assigned to International Business Machines, 1987.

19. Y.M. Greshishchev et al., "A 60-dB Gain, 55-dB Dynamic Range, 10-Gb/s Broadband SiGe HBT Limiter Amplifier", IEEE Journal of Solid-State Circuits, vol. 34, no. 12, 1914-1920, Dec. 1999.

20. W. Pohlmann, "A Silicon-Bipolar Amplifier for 10 Gbit/s with 45 dB Gain", IEEE Journal of Solid-State Circuits, vol. 29, no. 5, 551-556, May 1994.

21. A. Buchwald and K. Martin, "Integrated Fiber-Optic Receivers", Kluwer Academic Publishers, Fourth printing 2002.

22. C. A. Sharp, "A 3-State Phase Detector Can Improve Your Next PLL Design," EDN, pp. 55-59, Sept. 1976.

23. C. Hogge, "A Self-Correcting Clock Recovery Circuit", IEEE Journal of Light-Wave Technology, vol. LT-3, 1312-1314, Dec. 1985.

24. J. D. H. Alexander, "Clock Recovery from Random Binary Data," Electronics Letters, vol. 11, pp. 541-542, Oct. 1975.

25. R-J Yang et al., "A 3.125-Gb/s Clock and Data Recovery Circuit for 10-Gbase-LX4 Ethernet, IEEE Journal of Solid-State Circuits, vol. 39, 1356-1360, Aug. 2004.

26. J. Savoj and B. Razavi, "A 10-Gb/s CMOS Clock and Data Recovery Circuit with A Half-Rate Binary Phase/Frequency Detector", IEEE Journal of Solid-State Circuits, vol. 38, 13-21, January 2003.

27. J.E. Rogers and R. J. Long, "A 10-Gb/s CDR/DEMUX with LC Delay Line VCO in 0.18-µm CMOS, IEEE Journal of Solid-State Circuits, vol. 37, 1781-1789, Dec. 2002.

28. D. Richman, "Color-Carrier Reference Phase Synchronization Accuracy in NTSC Color-Television", Proc. IRE, vol. 22, 106-133, jan. 1954.

29. B. Razavi, "Design of Integrated Circuits for Optical Communications", McGraw-Hill Higher Education, 2003.

30. D. G. Messerschmitt, "Frequency Detector for PLL Acquisition in Timing and Carrier Recovery", IEEE Trans. Comm., vol. 27, 1288-1295, Sep. 1979.

31. L. DeVitto, "A Versatile Clock Recovery Architecture and Monolithic Implementation", Monolithic Phase-Locked Loops and Clock Recovery Circuits, B. Razavi, Ed., New York: IEEE Press, 1996.

32. J. Savoj and B. Razavi, "A 10-Gb/s CMOS Clock and Data Recovery Circuit with Frequency Detection", IEEE Journal of Solid-State Circuits Conference Digest of Technical Papers, 78-79, Feb. 2001.

33. G. Guetierrez et al., "2.485 Gb/s Silicon Bipolar Clock and Data Recovery IC for SONET (OC-48)", Proceedings of the Customs Integrated Circuits Conference, 575-578, May 1998.

34. M.ASSAAD and D. R. S. Cumming, "CMOS IC Design and Verilog-A Modeling of 10-Gb/s PLL-Based Deserializer for Inter-Chip Communication in SOC.", international symposium on system on chip 2007, Nov. 2007.

35. J. Savoj and B. Razavi, "A 10-Gb/s CMOS Clock and Data Recovery Circuit with a Half-Rate Linear Phase Detector". IEEE JSSC, vol. 36, 761-767, May 2001.

36. H. Djahanshahi and C. A. T. Salama, "Differential CMOS Circuits for 622-MHz/933-MHz Clock and Data Recovery Applications". IEEE JSSC, vol. 35, 847-855, June 2006.

37. C. J. Scheytt et al., "A 0.155-, 0.622-, and 2.488-Gb/s Automatic Bit-Rate Selecting Clock and Data Recovery IC for Bit-Rate Transparent SDH Systems". IEEE JSSC, vol. 34, 1935-1943, Dec. 1999.

38. K. Irvani et al., "Clock and Data Recovery for 1.25-Gb/s Ethernet Transceiver in 0.35-μm CMOS". IEEE CICC, 261-264, 1999.

39. R. C. Walker et al., "A Two-Chip 1.5GBd Serial Link Interface". IEEE JSSC, vol. 27, 1805-1811, Dec. 1992.

40. B. S. Anand and B. Razavi, "A CMOS Clock Recovery Circuit for 2.5-Gb/s NRZ Data", IEEE JSSC, vol. 36, 432-439, Mar. 2001.

41. Y. Qiu et al., "5-Gb/s 0.18- μm CMOS Clock Recovery Circuit". IEEE Int. Workshop VLSI Design & Video Tech., 21-23, May 28-30, 2005.

42. A. Rezayee and K. Martin, "A 9-16 GB/s Clock and Data Recovery Circuit with Three-State Phase Detector and Dual-Path Loop Architecture. ESSCIRC'03.

43. T-S Chen et al., "A 10Gb/s Clock and Data Recovery Circuit with Binary Phase/Frequency Detector Using TSMC 0.35−μm SiGe BiCMOS Process", IEEE Asia-Pacific conference on Circuit and Systems, Dec. 6-9, 981-984, 2004.

44. Razavi B., A 2.5-Gb/s 15-mW Clock Recovery Circuit. IEEE JSSC, vol. 31, pp. 472-480, April 1996.

45. F. Herzel, and B. Razavi, "A Study of Oscillator Jitter Due to Supply and Substrate Noise," IEEE Trans. Circuits and Systems, Part II, vol. 46, pp.56-62, 1999.

46. J. A. McNeill, "Jitter in Ring Oscillator," IEEE JSSC, vol. 32, pp. 870-879, 1997.

47. D. H. Wolaver, "Phase-Locked Loop Circuit Design," PTR Prentice Hall, 1991.

48. M. Mizuno et al., "A GHz MOS Adaptive Pipeline Technique Using MOS Current-Mode Logic," IEEE JSSC, vol. 31, pp. 784-791, June1996.

49. K. Irvani et. al., "Clock and data Recovery for 1.25 Gb/s Ethernet Transceiver in 0.35 μm CMOS," in Proc. IEEE Custom Integrated Circuits Conf., May 2001, pp. 261-264.

50. H.-T. Ng and D. J. Allstot, "CMOS Current Steering Logic for Low-Voltage Mixed-Signal Integrated Circuits," IEEE Trans. VLSI Syst., vol. 5, pp. 301-308, Sep. 1997.

51. A. Tanable et. al., "0.18-μm CMOS 10-Gb/s Multiplexer/Demultiplexer ICs Using Current Mode Logic with Tolerance to Threshold Voltage Fluctuation," IEEE JSSC, vol. 36, pp. 988-996, June 2001.

52. H.-D. Wohlmuth et. al., "A High Sensitivity Static 2:1 Frequency Divider up to 19 GHz in 120 nm CMOS," in Proc. IEEE Radio Frequency Integrated Circuits (RFIC) Symp., June 2002, pp. 231-234.

53. M. H. Anis and M. I. Elmasry, "Self-Timed MOS Current Mode Logic for Digital Applications," in Proc. IEEE Int. Conf. ASIC/SOC, 2002, pp. 193-197.

54. J. Musicer and J. Rabaey, "MOS Current Mode Logic for Low Power, Low Noise CORDIC Computation in Mixed-Signal Environments," Proc. ISPLPED, pp. 102-107, July 2000.

55. M. W. Allam and M. I. Elmasry, "Dynamic Current Mode Logic (DyCML), A New Low-Power High performance Logic style," IEEE JSSC, vol. 36, pp. 550-558, March 2001.

56. J. Rabaey, Digital Integrated Circuits: A Design perspective. Englewood Cliffs, NJ: Prentice-Hall, 1996.

57. B. Razavi, "Prospects of CMOS technology for high-Speed Optical Communication Circuits," IEEE JSSC, vol. 37, pp. 1135-1145, Sept. 2002.

58. S. J. Song et. al., "A 4-Gb/s CMOS Clock and Data Recovery Circuits Using 1/8-Rate Clock Technique," IEEE JSSC, vol. 38, pp. 1213-1219, July 2003.

59. S. H. Lee et. al., "A 5 Gb/s 0.25 µm CMOS Jitter-Tolerant variable-Interval Oversampling Clock/Data Recovery Circuit,", IEEE JSSC, vol. 37, pp. 1822-1830, December 2002.

60. J. E. Rogers and J. R. Long, "A 10 Gb/s CDR/DEMUX with LC Delay Line VCO in 0.18-µm CMOS," IEEE JSSC, vol. 37, pp. 1781-1789, May 2002.

61. J. Savoj and B. Razavi, "A 10-Gb/s CMOS Clock and Data recovery Circuit with Half-Rate Linear Phase Detector," IEEE JSSC, vol. 36, pp. 761-767, May 2001.

62. A. Pottbacker et. al., "A Si Bipolar Phase and Frequency Detector IC for Clock Extraction up to 8 Gb/s," IEEE JSSC, vol. 27, pp. 1747-1751, December 1992.

63. B. Stilling, "Bit Rate and Protocol Independent Clock and Data Recovery," Electron. Lett., vol. 36, pp. 824-825, April 2000.

64. M. Assaad and D. R. S. Cumming, "20 Gb/s Referenceless Quarter-Rate PLL-Based Clock Data Recovery Circuit in 130 nm CMOS Technology", 15th International Conference on Mixed Design of Integrated Circuits and Systems. MIXDES 2008. pp. 147–150, 2008.

65. C-C Kuo, Y-C Wang and C-N J. Liu "An Efficient Approach to Build Accurate Behavioural Models of PLL Designs," IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, vol. E89-A, pp. 391-398, February 2006.

66. L-X Liu, Y-T Yang, Z-M Zhu and Y. Li, "Design of PLL System Based Verilog-AMS Behavioural Models," IEEE Int. Workshop VLSI Design & Video Tech., pp. 67-70, May 2005.

67. T. Oura, Y. Hiraku, T. Suzuki, and H. Asai "Modelling and Simulation of Phase-Locked Loop with Verilog-A Description for Top-Down Design," IEEE Asia-Pacific conference on Circuit and Systems, vol. 1, pp. 549-552, December 2004.

68. G. Balamurugan and N. Shanbhag, Modelling and Mitigation of Jitter in High-Speed Source-Synchronous Inter-Chip Communication Systems. IEEE Computer Society, Proceedings of the 21st International conference on computer Design (ICCD'03).

69. E. Yeung and Horowitz Mark A., "A 2.4 Gb/s/pin Simultaneous Bidirectional Parallel link with Per-Pin Skew Compensation," IEEE JSSC, vol. 35, pp. 1619-1628, November 2000.

70. Y. Chen et. al., "A Novel technique to Enhance the Negative resistance for Colpitts Oscillators by Parasitic Cancellation," IEEE Conference on Electron Devices and Solid-State Circuits, pp. 425-428, December 2007.

71. K. Mayaram, "Output Voltage Analysis for the MOS Colpitts Oscillator," IEEE Transactions on Circuits and Systems I, vol. 47, pp. 260-263, February 2000.

72. C.-Y. Cha and S.-G. Lee, "A Complementary Colpitts Oscillator in CMOS Technology," IEEE Transactions on Microwave Theory and Techniques, vol. 3, pp. 881-887, March 2005.

73. U. Yodprasit and C. C. Enz, "Realization of Low-Voltage and low-Power Colpitts Quadrature Oscillator," IEEE International Symposium on Circuits and Systems, pp. 4289-4292, 2006.

74. J. Steinkamp et. al., "A Colpitts Oscillator design for a GSM Base Station Synthesizer," IEEE Radio Frequency Integrated Circuits (RFIC) Symposium, pp. 405-408, June 2007.

75. Y. A. Eken and J. P. Uyemura, "A 5.9-GHz Voltage-Controlled Ring Oscillator in 0.18-μm CMOS," IEEE JSSC, vol. 39, pp. 230-233, January 2004.

76. S.-J. Lee et. al., "A novel high-speed ring oscillator for multiphase clock generation using negative skewed delay scheme," IEEE JSSC, vol. 32, pp. 289-291, February 1997.

77. J.D. Van Der Tang et. al., "A 9.8-11.5-GHz quadrature ring oscillator for optical receivers," IEEE JSSC, vol. 37, pp. 438-442, March 2002.

78. A. Hajimiri and T. Lee, "Design Issues in CMOS Differential LC Oscillators," IEEE JSSC, vol. 34, pp. 717-724, May 1999.

79. P. Zhang, "Design of CMOS LC Oscillators," International Conference on Solid-State and Integrated Circuit Technology, pp. 1534-1537, October 2006.

80. J. Van Der Tang and A. Van Roermund, "A 5.3 GHz phase shift tuned I/Q LC oscillator with 1.1 GHz tuning range," IEEE MTT-S International Microwave Symposium Diegest, vol. 1, pp. A133-A136, June 2003.

81. R. Dobkin et al., "Parallel vs. Serial On-Chip Communication," CCIT TR674, EE Pub No. 1631, EE Dept., Technion, December 2007.

82. J. Musicer and J. Rabaey, "An Analysis of MOS Current Mode Logic for Low Power and High Performance Digital Logic," Proceedings ISLPED, July 2000.