creative commons

COMMONS DEED

저작자표시-비영리-변경금지 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.

다음과 같은 조건을 따라야 합니다:

저작자표시. 귀하는 원저작자를 표시하여야 합니다.

비영리. 귀하는 이 저작물을 영리 목적으로 이용할 수 없습니다.

변경금지. 귀하는 이 저작물을 개작, 변형 또는 가공할 수 없습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 이용허락규약(Legal Code)을 이해하기 쉽게 요약한 것입니다.

Disclaimer

dCollection

Ph.D.Dissertation

Design Techniques for Stochastic Frequency Detector Based Referenceless Clock and Data Recovery

통계적 주파수 검출기 기반 기준 주파수를 사용하지 않는 클록 및 데이터 복원 회로의 설계 방법론

by

Hong Seok Choi

August, 2022

Department of Electrical and Computer Engineering
College of Engineering
Seoul National University

Design Techniques for Stochastic Frequency Detector Based Referenceless Clock and Data Recovery

지도 교수 정 덕 균

이 논문을 공학박사 학위논문으로 제출함
2022 년 8 월

서울대학교 대학원
전기·정보공학부
최 홍 석

최홍석의 공학박사 학위논문을 인준함
2022 년 8 월

위 원 장 _____ (인)

부위원장 _____ (인)

위 원 _____ (인)

위 원 _____ (인)

위 원 _____ (인)

Design Techniques for Stochastic Frequency Detector Based Referenceless Clock and Data Recovery

by

Hong Seok Choi

A Dissertation Submitted to the Department of
Electrical and Computer Engineering
in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy
at
SEOUL NATIONAL UNIVERSITY

August, 2022

Committee in Charge:

Professor Jaeha Kim, Chairman

Professor Deog-Kyoon Jeong, Vice-Chairman

Professor Woo-Seok Choi

Professor Dongsuk Jeon

Professor Jaeduk Han

Abstract

In this thesis, a design of a high-speed, power-efficient, wide-range clock and data recovery (CDR) without a reference clock is proposed. A frequency acquisition scheme using a stochastic frequency detector (SFD) based on the Alexander phase detector (PD) is utilized for the referenceless operation. Pattern histogram analysis is presented to analyze the frequency acquisition behavior of the SFD and verified by simulation. Based on the information obtained by pattern histogram analysis, SFD using autocovariance is proposed. With a direct-proportional path and a digital integral path, the proposed referenceless CDR achieves frequency lock at all measurable conditions, and the measured frequency acquisition time is within 7μs. The prototype chip has been fabricated in a 40-nm CMOS process and occupies an active area of 0.032 mm^2. The proposed referenceless CDR achieves the BER of less than 10^{-12} at 32 Gb/s and exhibits an energy efficiency of 1.15 pJ/b at 32 Gb/s with a 1.0 V supply.

Keywords : Acquisition time, bang-bang phase detector (BBPD), bang-bang phase-frequency detector (BBPFD), clock and data recovery (CDR), dual-loop, frequency acquisition, low power, phase-locked loop (PLL), referenceless, stochastic frequency detector (SFD), stochastic phase-frequency detector (SPFD), unlimited frequency detection.

Student Number : 2016-20984

Contents

ABSTRACT **I**

CONTENTS **II**

LIST OF FIGURES **IV**

LIST OF TABLES **VIII**

CHAPTER 1 INTRODUCTION **1**

1.1 MOTIVATION 1

1.2 THESIS ORGANIZATION 13

CHAPTER 2 BACKGROUNDS **14**

2.1 CLOCKING ARCHITECTURES IN SERIAL LINK INTERFACE.....14

2.2 GENERAL CONSIDERATIONS FOR CLOCK AND DATA RECOVERY 24

2.2.1 OVERVIEW 24

2.2.2 JITTER 26

2.2.3 CDR JITTER CHARACTERISTICS 33

2.3 CDR ARCHITECTURES.....39

2.3.1 PLL-BASED CDR – WITH EXTERNAL REFERENCE CLOCK 39

2.3.2 DLL/PI-BASED CDR.....44

2.3.3 PLL-BASED CDR – WITHOUT EXTERNAL REFERENCE CLOCK.....47

2.4 FREQUENCY ACQUISITION SCHEME 50

2.4.1 TYPICAL FREQUENCY DETECTORS 50

2.4.1.1 DIGITAL QUADRICORRELATOR FREQUENCY DETECTOR 50

2.4.1.2 ROTATIONAL FREQUENCY DETECTOR 54

2.4.2 PRIOR WORKS.....56

CHAPTER 3 DESIGN OF THE REFERENCELESS CDR USING SFD 58

3.1 OVERVIEW.....58

3.2 PROPOSED FREQUENCY DETECTOR.....62

3.2.1 MOTIVATION.....62

3.2.2 PATTERN HISTOGRAM ANALYSIS.....68

3.2.3 INTRODUCTION OF AUTOCOVARIANCE TO STOCHASTIC FREQUENCY DETECTOR 75

3.3 CIRCUIT IMPLEMENTATION 83

3.3.1 IMPLEMENTATION OF THE PROPOSED REFERENCELESS CDR 83

3.3.2 CONTINUOUS-TIME LINEAR EQUALIZER (CTLE) 85

3.3.3 DIGITALLY-CONTROLLED OSCILLATOR (DCO).....87

3.4 MEASUREMENT RESULTS 89

CHAPTER 4 CONCLUSION 99

APPENDIX A DETAILED FREQUENCY ACQUISITION WAVEFORMS OF THE PROPOSED SFD 100

BIBLIOGRAPHY 108

초 록 122

List of Figures

FIG. 1.1 THE NEW COVID-19 BROADBAND HOSEHOLD AND THE TRAFFIC GROWTH IN THE EARLY PHASE OF THE COVID-19 PANDEMIC (2020) [1] 2

FIG. 1.2 ZOOM TRAFFIC OVER THE EARLY PHASES OF THE COVID-19 PANDEMIC [1] 3

FIG. 1.3 GLOBAL INTERNET TRAFFIC SHARE [2] 4

FIG. 1.4 INCREASING VIDEO DEFINITION [3] 5

FIG. 1.5 GLOBAL INTERNET USER GROWTH [3] 6

FIG. 1.6 GLOBAL DEVICE AND CONNECTION GROWTH [3] 6

FIG. 1.7 PER-LANE DATA-RATE VS. YEAR FOR A VARIETY OF COMMON I/O STANDARDS [6] 7

FIG. 1.8 MARKET SHARES 2022 ACCORDING TO HMS NETWORKS [7] 8

FIG. 1.9 LATEST INTERFACES AND NOMENCLATURE [8] 9

FIG. 1.10 RELATIVE FORM FACTOR FOOTPRINTS [10] 10

FIG. 2.1 BLOCK DIAGRAM OF THE SERDES INTERFACE 15

FIG. 2.2 SYNCHRONOUS CLOCKING ARCHITECTURE – FORWARDED CLOCKING 18

FIG. 2.3 MESOCHRONOUS CLOCKING ARCHITECTURE – FORWARDED CLOCKING 19

FIG. 2.4 MESOCHRONOUS CLOCKING ARCHITECTURE – COMMON REFERENCE CLOCK 20

FIG. 2.5 PLESIOCHRONOUS CLOCKING ARCHITECTURE 22

FIG. 2.6 ROLE OF A CDR CIRCUIT IN RETIMING DATA [68] 25

FIG. 2.7 OPTIMUM SAMPLING OF NOISY DATA [68] 25

FIG. 2.8 THE NORMAL DISTRIBUTION [70] 27

FIG. 2.9 THE CONVOLUTION OF THE SUM OF TWO DELTA FUNCTIONS SEPARATED BY DJ AND GAUSSIAN RJ DISTRIBUTION OF WIDTH Σ [73] 30

FIG. 2.10 APPLICATION OF THE DUAL-DIRAC MODEL (DJ: SINUSOIDAL DISTRIBUTION) [73] .. 31

FIG. 2.11 APPLICATION OF THE DUAL-DIRAC MODEL (DJ: UNIFORM DISTRIBUTION) [73]..... 31

FIG. 2.12 ACCUMULATION OF CYCLE-TO-CYCLE JITTER IN A PHASE-LOCKED OSCILLATOR:.... 34

FIG. 2.13 JITTER TRANSFER MASK [69] 35

FIG. 2.14 SONET JITTER TOLERANCE MASK 120[69] 37

FIG. 2.15 OIF-CEI JITTER TOLERANCE MASK [74]..... 38

FIG. 2.16 PLL-BASED CDR ARCHITECTURE WITH DUAL VCO. 40

FIG. 2.17 PLL-BASED CDR ARCHITECTURE USING SEQUENTIAL LOCKING. 42

FIG. 2.18 ADPLL-BASED CDR ARCHITECTURE 43

FIG. 2.19 PLL-BASED CDR ARCHITECTURE WITH DUAL VCOS 43

FIG. 2.20 DLL-BASED CDR ARCHITECTURE 45

FIG. 2.21 DLL-BASED CDR ARCHITECTURE : FORWARDED CLOCKING 45

FIG. 2.22 DLL-BASED CDR ARCHITECTURE 46

FIG. 2.23 DLL-BASED CDR ARCHITECTURE : FORWARDED CLOCKING 46

FIG. 2.24 DUAL-LOOP REFERENCELESS CDR ARCHITECTURE 48

FIG. 2.25 SINGLE-LOOP REFERENCELESS CDR ARCHITECTURE..... 48

FIG. 2.26 POTTBACKER FD (KNOWN AS DQFD) 50

FIG. 2.27 TIMING DIAGRAMS OF POTTBAKER FD WHEN THE CLOCK FREQUENCY IS (A) LOWER AND (B) HIGHER THAN INPUT DATA RATE..... 51

FIG. 2.28 BLOCK DIAGRAM OF THE CDR USING DIGITAL QUADRICORRELATOR[82] 53

FIG. 2.29 OPERATION PRINCIPLE OF ROTATIONAL FD 54

FIG. 2.30 TIMING DIAGRAM OF ROTATIONAL FD WHEN THE CLOCK FREQUENCY IS (A) HIGHER THAN THE DATA RATE AND (B) LOWER THAN THE DATA RATE 55

FIG. 3.1 (A) THE REPRESENTATIVE PATTERN HISTOGRAMS (B) AND CORRESPONDING

EARLY/LATE DECISION AND WEIGHT USED IN [52] 60

FIG. 3.2 SIMULATED (A) PHASE DETECTION GAIN CURVE AND (B) FREQUENCY DETECTION CURVE OF [52] 61

FIG. 3.3 CIRCUIT IMPLEMENTATION OF REFERENCELESS CDR WITH STOCHASTIC FPD [52] ... 62

FIG. 3.4 32GB/S 7-DB CHANNEL LOSS PRBS7 FREQUENCY ACQUISITION BEHAVIOR OF SPFD WITH 3-BIT PATTERN WEIGHT (-1, -1, 3, 3) 63

FIG. 3.5 ZOOMED VIEW (0 ~ 100MS) OF FIG. 3.4 64

FIG. 3.6 ZOOMED VIEW (100MS ~ 200MS) OF FIG. 3.4 65

FIG. 3.7 OPEN-LOOP PATTERN HISTOGRAM 68

FIG. 3.8 FD GAIN CURVE WITH 3-BIT PATTERN WEIGHTS (-1, -1, 3, 3) 69

FIG. 3.9 BBPD FD GAIN IN THE DIRECT-PROPORTIONAL PATH 71

FIG. 3.10 DIRECT-PROPORTIONAL PATH FREQUENCY CAPTURE RANGE 71

FIG. 3.11 OPEN-LOOP PATTERN HISTOGRAM INCLUDING DIRECT-PROPORTIONAL PATH 72

FIG. 3.12 FD GAIN CURVE WITH 3-BIT PATTERN WEIGHTS (-1, -1, 3, 3) INCLUDING DIRECT-PROPORTIONAL PATH 73

FIG. 3.13 CONCEPTUAL DIAGRAM OF CALCULATING AUTOCOVARIANCE OF THE 3-BIT PATTERN : *DN0* CASE 77

FIG. 3.14 BLOCK DIAGRAM OF THE SFD LOGIC 78

FIG. 3.15 PATTERN HISTOGRAMS OF THE AUTOCOVARIANCE FOR 3-BIT PATTERNS 79

FIG. 3.16 (A) THE FD CURVE AND (B) WEIGHTS OF THE PROPOSED SFD 80

FIG. 3.17 32GB/S 7-DB CHANNEL LOSS PRBS7 FREQUENCY ACQUISITION BEHAVIOR OF THE PROPOSED SFD 81

FIG. 3.18 SYSTEM BLOCK DIAGRAM OF THE PROPOSED CDR ARCHITECTURE 83

FIG. 3.19 CIRCUIT DIAGRAM OF THE CTLE 85

FIG. 3.20 POST-LAYOUT SIMULATED FREQUENCY RESPONSE OF CTLE 86

FIG. 3.21 BLOCK DIAGRAM OF THE 8-PHASE DCO 87

FIG. 3.22 CHIP PHOTOMICROGRAPH 89

FIG. 3.23 MEASUREMENT SETUP. 91

FIG. 3.24 MEASURED DCO GAIN CURVE 92

FIG. 3.25 MEASURED TRANSIENT RESPONSE OF THE PROPOSED CDR @ 28 GB/S PRBS7 93

FIG. 3.26 MEASURED FREQUENCY ACQUISITION BEHAVIOR AT VARIOUS DATA RATES 94

FIG. 3.27 MEASURED JTOL AT 32 GB/S (BER < 10^{-12}) 96

FIG. 3.28 MEASURED JITTER HISTOGRAM OF RECOVERED CLOCK AT 32GB/S 97

FIG. A.1 32GB/S 7-DB CHANNEL LOSS PRBS7 FREQUENCY ACQUISITION BEHAVIOR OF THE PROPOSED SFD (KI=2, 3) 101

FIG. A.2 32GB/S 7-DB CHANNEL LOSS PRBS7 FREQUENCY ACQUISITION BEHAVIOR OF THE PROPOSED SFD (KI=4) 102

FIG. A.3 32GB/S 7-DB CHANNEL LOSS PRBS7 FREQUENCY ACQUISITION BEHAVIOR OF THE PROPOSED SFD (KI=5) 103

FIG. A.4 32GB/S 7-DB CHANNEL LOSS PRBS7 FREQUENCY ACQUISITION BEHAVIOR OF THE PROPOSED SFD (KI=6) 104

FIG. A.5 32GB/S 7-DB CHANNEL LOSS PRBS7 FREQUENCY ACQUISITION BEHAVIOR OF THE PROPOSED SFD (KI=7) 105

FIG. A.6 32GB/S 7-DB CHANNEL LOSS PRBS7 FREQUENCY ACQUISITION BEHAVIOR OF THE PROPOSED SFD (KI=8) 106

FIG. A.7 32GB/S 7-DB CHANNEL LOSS PRBS7 FREQUENCY ACQUISITION BEHAVIOR OF THE PROPOSED SFD (KI=9) 107

List of Tables

TABLE 2.1 CLASSIFICATION OF SIGNAL-CLOCK SYNCHRONIZATION [66] 16

TABLE 2.2 VALUES OF Q_{BER} 32

TABLE 2.3 SONET JITTER GENERATION [69] 33

TABLE 3.1 DETAILED POWER BREAKDOWN OF THE PROPOSED CDR 90

TABLE 3.2 DETAILED AREA OF THE PROPOSED CDR 90

TABLE 3.3 PERFORMANCE SUMMARY AND COMPARISON 98

Chapter 1

Introduction

1.1 Motivation

Recently, global data traffic is increasing unprecedentedly as smartphones have become more common and the use of video media platforms such as YouTube and Netflix has increased. In addition, lifestyle changes due to the COVID-19 pandemic are also accelerating this trend. For example, a global shutdown increased various data traffic volumes: High volume downstream video streaming, video conferencing, and game downloads mix with upstream video conferencing, social network live streams and uploads, as well as lower volume work and messaging applications [1] . This phenomenon is well descripted in Fig 1.1 and Fig 1.2. The leading cause of the exploding internet traffic is video streaming. According to Sandvine, in the first half of

Traffic Growth	1 Feb – 1 Mar Growth	1 Mar – 1 Apr Growth	1 Apr – 19 Apr Growth	Total 1 Feb – 19 Apr
Upstream	+2.07%	+123.18%	-2.95%	+121%
Downstream	-2.39%	+11.72%	+13.67%	+23%
Total	-1.74%	+28.69%	+9.28%	+38%

Fig. 1.1 The new COVID-19 broadband hosehold and the traffic growth in the early phase of the COVID-19 pandemic (2020) [1]

2021, bandwidth traffic was dominated by streaming video, accounting for 53.72% of overall traffic, with Youtube, Netflix, and Facebook video in the top three [2] . Fig 1.3 shows this trend well. Furthermore, this trend will intensify as the video definition increases. CISCO predicts that two-thirds of the installed flat-panel TV sets will be Ultra-High-Definition (UHD) or 4K by 2023, which is shown in Fig. 1.4 [3] . UHD as a percentage of IP VoD traffic will be higher at 35% by 2022 compared to 2017 [4] .

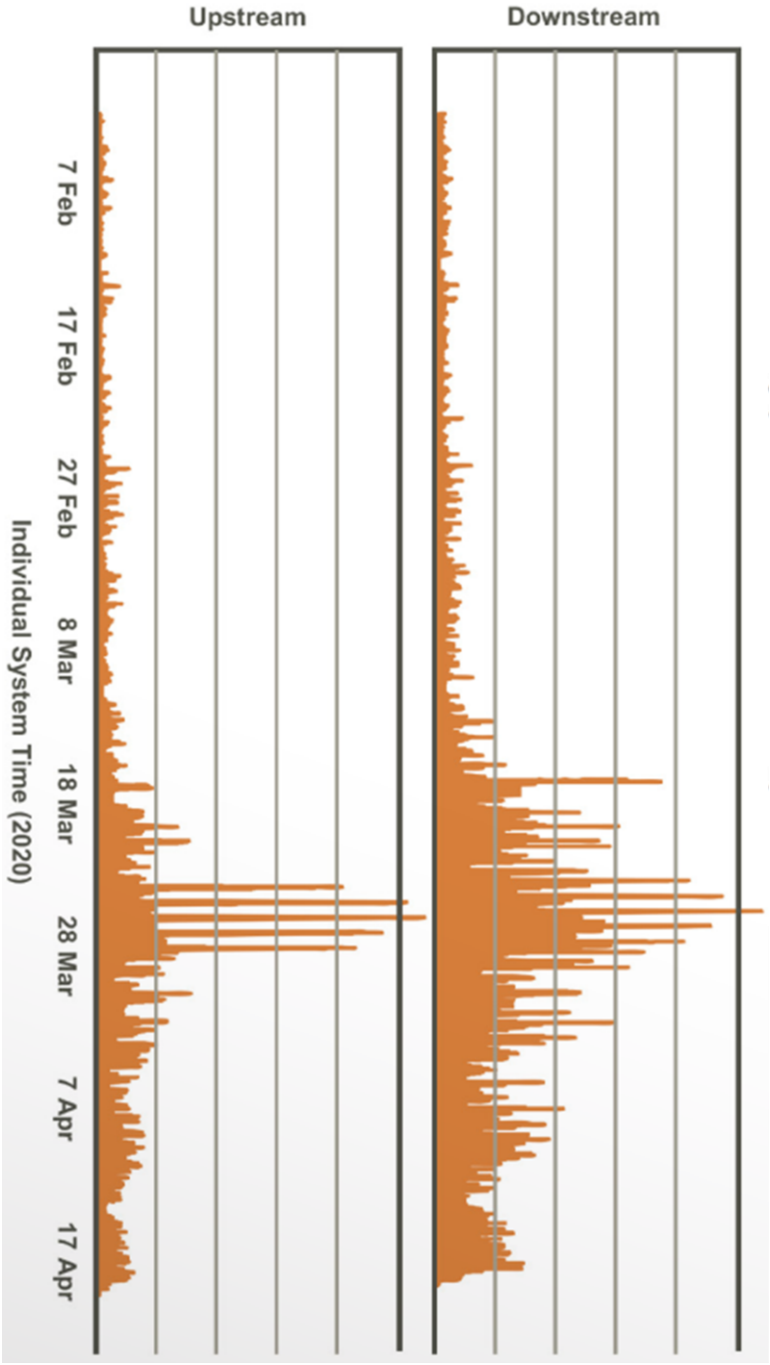


Fig. 1.2 Zoom traffic over the early phases of the COVID-19 pandemic [1]

CATEGORY TRAFFIC SHARE

TOTAL TRAFFIC

	Category	Total Volume
1	Video	53.72%
2	Social	12.69%
3	Web	9.86%
4	Gaming	5.67%
5	Messaging	5.35%
6	Marketplace	4.54%
7	File Sharing	3.74%
8	Cloud	2.73%
9	VPN	1.39%
10	Audio	0.31%

GLOBAL APP TRAFFIC SHARE

TOTAL TRAFFIC

	Category	Total Volume
1	YouTube	14.61%
2	Netflix	9.39%
3	Facebook	7.39%
4	Facebook video	4.20%
5	Tik Tok	4.00%
6	QUIC	3.98%
7	HTTP	3.58%
8	HTTP Media Stream	3.57%
9	BitTorrent	2.91%
10	Google	2.79%

GLOBAL APP TRAFFIC SHARE

DOWNSTREAM TRAFFIC

	Category	Total Volume
1	YouTube	16.37%
2	Netflix	10.61%
3	Facebook	7.67%
4	Facebook video	4.83%
5	TikTok	4.48%
6	HTTP Media Stream	4.07%
7	Generic QUIC	4.03%
8	HTTP	2.63%
9	Playstation Download	2.27%
10	iTunes Store	2.12%

GLOBAL APP TRAFFIC SHARE

UPSTREAM TRAFFIC

	Category	Total Volume
1	BitTorrent	9.70%
2	HTTP	9.05%
3	Google	8.02%
4	Facebook	5.77%
5	Wordpress	5.01%
6	YouTube	4.45%
7	iCloud	4.09%
8	Generic QUIC	3.70%
9	Netflix	3.00%
10	Facebook Messenger	2.37%

Fig. 1.3 Global internet traffic share [2]

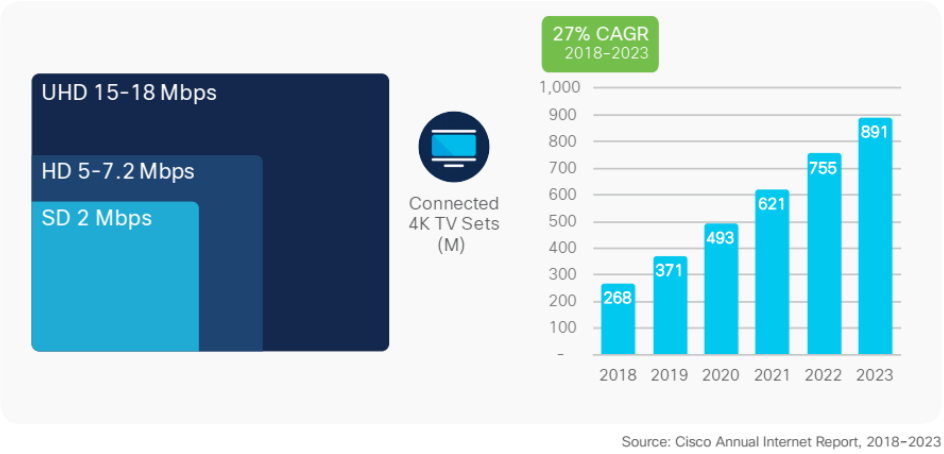


Fig. 1.4 Increasing video definition [3]

In short, the amount of data that each user consumes per month will grow continuously. However, not only the data bandwidth which each user consumes but also the total number of internet users and device connections is increasing. Global internet user growth is depicted in Fig. 1.5, and global device and connection growth is illustrated in Fig 1.6. Thus, this never-ending global datasphere is inevitable. IDC predicts that the global datasphere will grow to 175 zettabyes by 2025 [5] .

The total data bandwidth for wireline interface must rise at a corresponding rate to handle the ever-increasing data traffic. A simple way to increase the data bandwidth is increasing the number of I/O pins. On the other hand, this solution requires a proportional physical area and has limitations. Thus, it is suitable for linear growth but not for exponential growth shown by recent data traffic. For this reason, various wireline interface standards are evolving in the direction of raising per-lane data transfer rate. This trend is shown in Fig. 1.7.

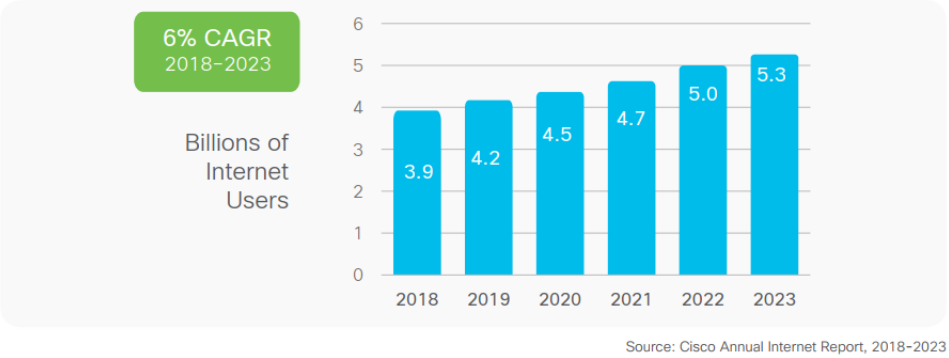


Fig. 1.5 Global Internet user growth [3]

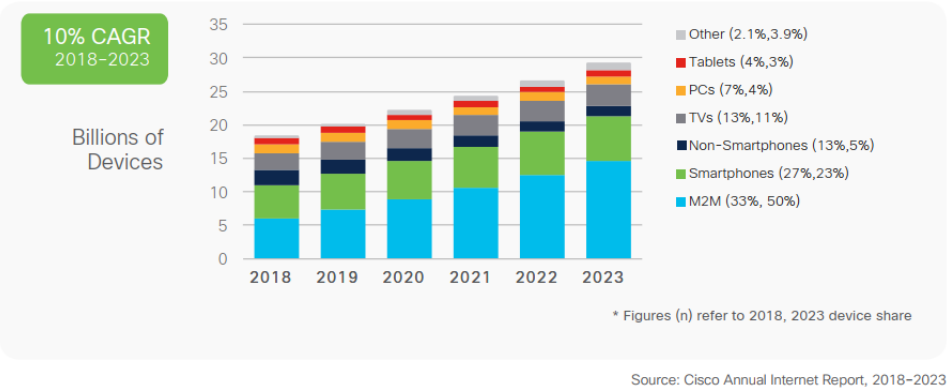


Fig. 1.6 Global device and connection growth [3]

Currently, the standard that is mainly used in the industrial networks is OIF-CEI based Ethernet. According to HMS networks, the industrial network market is expected to grow by 8% in 2022, and 66% of all new installed nodes are Ethernet, which is shown in Fig. 1.8 [7] . The most recently established standard is 400 Gigabit Ethernet (400GbE) and 200GbE, and Ethernet standards currently in commercial use are 1GbE, 10GbE, 40GbE and 100GbE. As the data transmission rate rises, the distance

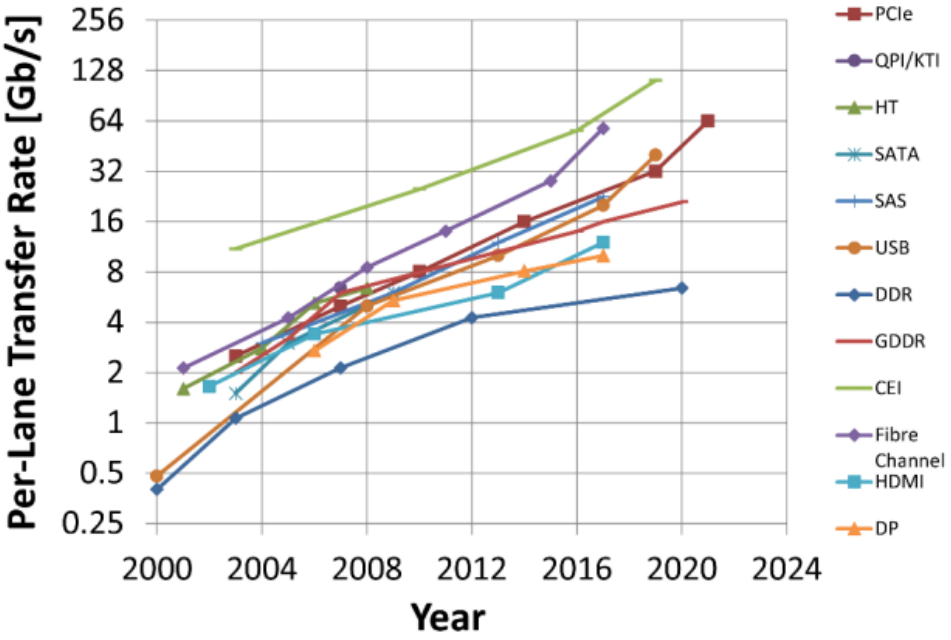


Fig. 1.7 Per-lane data-rate vs. year for a variety of common I/O standards [6]

PCIe: Peripheral Component Interconnect express; QPI/KTI: QuickPath Interconnect/Keizer Technology Interconnect; HT: HyperTransport; SATA: Serial Advanced Technology attachment; SAS: Serial Attached Small computer system interface; USB: Universal Serial Bus; DDR: Double Data Rate SDRAM; GDDR: Graphics DDR; CEI: Common Electrical I/O; HDMI: High Definition Multimedia Interface; DP: DisplayPort.

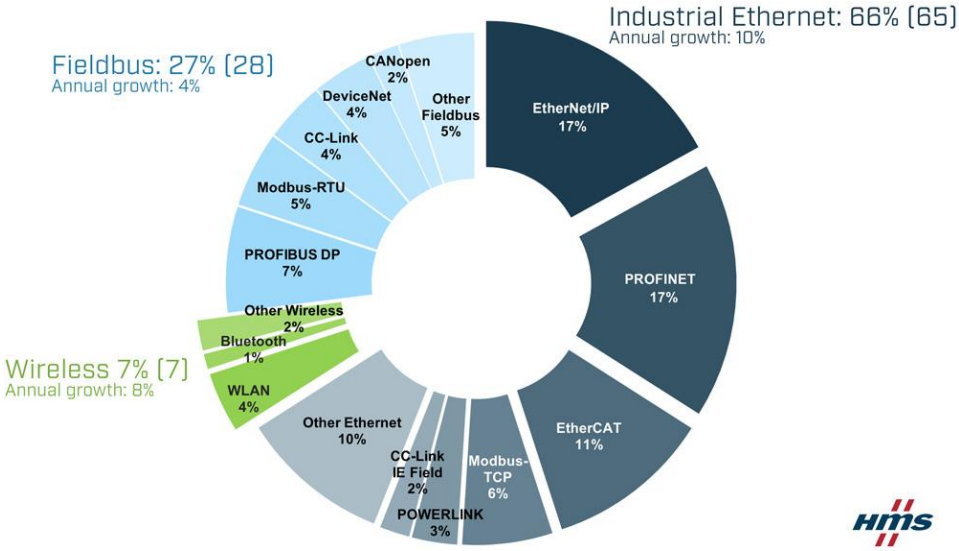


Fig. 1.8 Market shares 2022 according to HMS Networks [7]

that can be transmitted by copper wire is getting shorter. Thus, traditional copper wire-line networks are becoming less and less common in Ethernet standards, and fiber-optic networks are instead increasing in proportion. This trend is illustrated in Fig. 1.9. Since silicon chips cannot produce optical signals, there must be an electrical-to-optical signal converter in the interface system to use fiber-optic networks. Moreover, the new Ethernet standard must be compatible with the previous standards, so Ethernet interface systems must support both electrical and optical cables. For these reasons, there are specified form factors that suit Ethernet standards: small form-factor pluggable (SFP) transceiver. The SFP converts the serial electrical signals to serial optical signals and vice versa, so it can be connected with a fiber-optic cable and a copper cable both [9] . Fig. 1.10 shows an example for 400GbE form factors. By the way,

	Backplane	Twinax Cable	15-40m(OT) Single Twisted Pair	>100m (OT) Single Twisted Pair	100m (IT) Twisted Pair (2/4 Pair)	MMF	500m PSM4	2km SMF	10km SMF	20km SMF	40km SMF	80km SMF	Electrical Interface	Pluggable Module
10BASE–	T1S		T1S	T1L	T									
100BASE–			T1	T1L*	T									
1000BASE–			T1		T									
2.5GBASE–	KX		T1		T									
5GBASE–	KR		T1		T									
10GBASE–			T1		T				BIDI Access	BIDI Access	BIDI Access			
25GBASE–	KR1 KR	CR1 CR/CR-S	T1		T (30m)	SR			LR EPON BIDI Access	EPON BIDI Access	ER BIDI Access		25GAUI	SFP
40GBASE–	KR4	CR4			T (30m)	SR4/eSR4	PSM4	FR	LR4				XLAUI XLPPI	QSFP
50GBASE–	KR2 KR	CR2 CR	T2			SR		FR	EPON BIDI Access LxR	EPON BIDI Access	BIDI Access ER		LAUI-2/50GAUI-2 50GAUI-1	SFP/QSFP
100GBASE–	KR4 KR2 KR1	CR10 CR4 CR2 CR1	T4			SR10 SR4 SR2 VR1 SR1	PSM4 DR	CWDM4 FR1	LR4 4WDM-10 LR1	4WDM-20	ER4 4WDM-40	ZR	CAUI-10 CPPI CAUI-4/100GAUI-4 100GAUI-2 100GAUI-1	SFP QSFP/QSFP-DD OSFP
200GBASE–	KR4 KR2	CR4 CR2 CR1*				SR4 VR2 SR2	DR4 1pair*	FR4 1 pair*	LR4		ER4		200GAUI-4 200GAUI-2 200GAUI-1*	QSFP/QSFP-DD SFP-DD
400GBASE–	KR4*	CR4 CR2*				SR16 SR8/SR4.2 VR4 SR4	DR4 2 pair*	FR8 FR4 400G-FR4	LR8 LR4-6 400G-LR4-10		ER8	ZR	400GAUI-16 400GAUI-8 400GAUI-4 400GAUI-2*	QSFP/QSFP-DD OSFP
800GBASE-	ETC-KR8 KR8*	ETC-KR8 CR8* CR4*				VR8* SR8*	8 pair* 4 pair*	8 pair* 4 pair* 4 lambda*	TBD*		TBD*		800GAUI-8* 800GAUI-4*	
1.6TBASE-		CR8*					8 pair*	8 pair*					1.6TAUI-16* 1.6TAUI-8*	QSFP/QSFP-DD OSFP/OSFP-XD

Gray Text = IEEE Standard Red Text = In Task Force Green Text = In Study Group
Blue Text = Non-IEEE standard but complies to IEEE electrical interfaces * Note: As of publication, subject to change

Fig. 1.9 Latest interfaces and nomenclature [8]

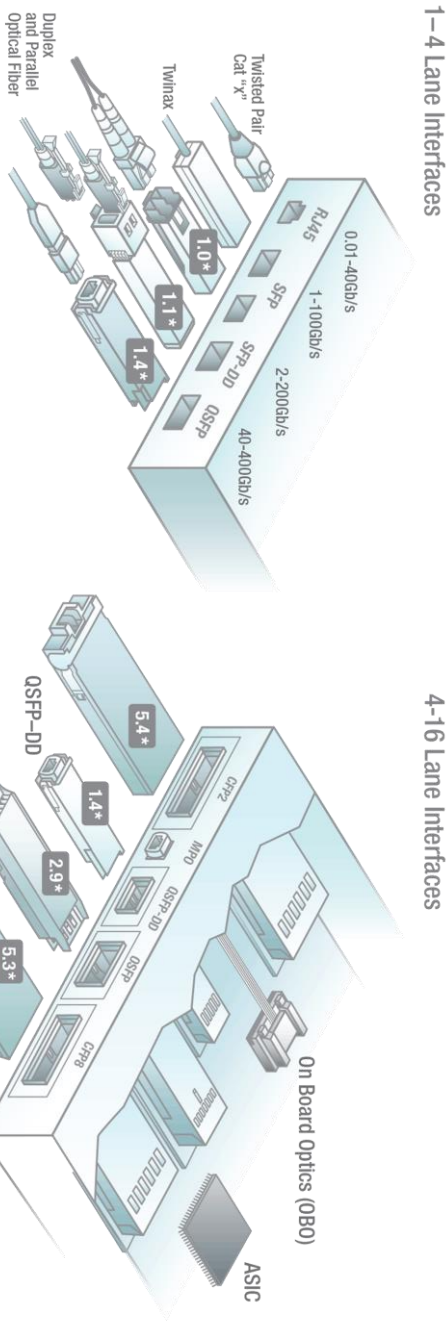


Fig. 1.10 Relative form factor footprints [10]

there is no clock pin in the SFP or QSFP electrical pin-out, so SFP and QSFP transceivers need clock and data recovery (CDR) to regenerate the transmitted data. In addition, there is no space in SFP or QSFP modules to place a reference clock generator. Hence, SFP and QSFP must adopt referenceless CDR.

Implementing a referenceless CDR is a challenging task because the addition of a frequency tracking scheme requires a local oscillator and an additional loop to control frequency acquisition. Furthermore, referenceless CDR has difficulty in wide-frequency range operation because it has a limited frequency capture range. Despite these shortcomings, referenceless CDR is widely used in the wireline communications [11] - [52] . The main reason for using referenceless CDR is the reduced number of I/O pins, which is directly related to the system cost. A system like the HDMI interface employs forwarded clock architecture to lower the complexity of the receiver. However, it requires three additional pins, including one ground shield pin, to transmit the differential clock. A system like PCIe interface adopts an external 100MHz differential clock reference. Note that PCIe Gen 2 and 3 also supported data clocked reference clock architecture, which is the same as referenceless clock architecture, but this specification is deleted from PCIe Gen 4. In this case, three additional pins, including one ground shield pin, are assigned, and the clock generator should be placed in the same printed-circuit-board (PCB). Therefore, referenceless CDR is used when there is no space to allocate additional pins to packages such as connectors or when it is challenging to attach additional reference clock generators to the board. Repeaters and active cables are representative examples.

Systems that use referenceless CDR architecture do not receive external reference clocks, so they need a way to generate their own clocks and synchronize them

with input data. Therefore, a method of obtaining frequency information from input data is required for the referenceless CDR, which significantly affects the performance of the CDR. In this thesis, we propose a design method for referenceless CDR using a stochastic frequency detector. We analyze data histograms to predict CDR lock behavior and introduce the way to choose optimal coefficient values for robust operation.

1.2 Thesis Organization

This thesis is organized as follows. In Chapter 2, the backgrounds of the clocking architectures of high-speed serial link and general considerations of CDR are explained. The basic operations and architectures of the general CDR are provided. Also, frequency acquisition schemes for referenceless CDR are described.

In Chapter 3, a referenceless CDR architecture based on the stochastic frequency detector is presented. First, the basic concept of the stochastic frequency detector (SFD) is described, and a method for analyzing the frequency acquisition behavior of SFD is presented. According to the proposed analysis method, it is possible to infer the points at which false lock occurs, thereby enabling a stable SFD design without a false lock. Secondly, simulation results of the proposed referenceless CDR are presented. Then, the circuit implementation of the proposed referenceless CDR is described. Finally, the measurement results of the implemented CDR are presented.

Chapter 4 summarizes the proposed work and concludes this thesis.

Chapter 2

Backgrounds

2.1 Clocking Architectures in Serial Link Interface

Serial link interface is a communication interface between two digital systems. Serial link transmitter serializes low-speed on-chip parallel data into high-speed data and transmits high-speed data stream through wireline. Likewise, serial link receiver receives transmitted data stream and deserializes high-speed input data into low-speed parallel data to make data processable in a digital domain. This interface system is also referred to as the SerDes interface. Fig. 2.1 shows the simplified block diagram of the SerDes interface.

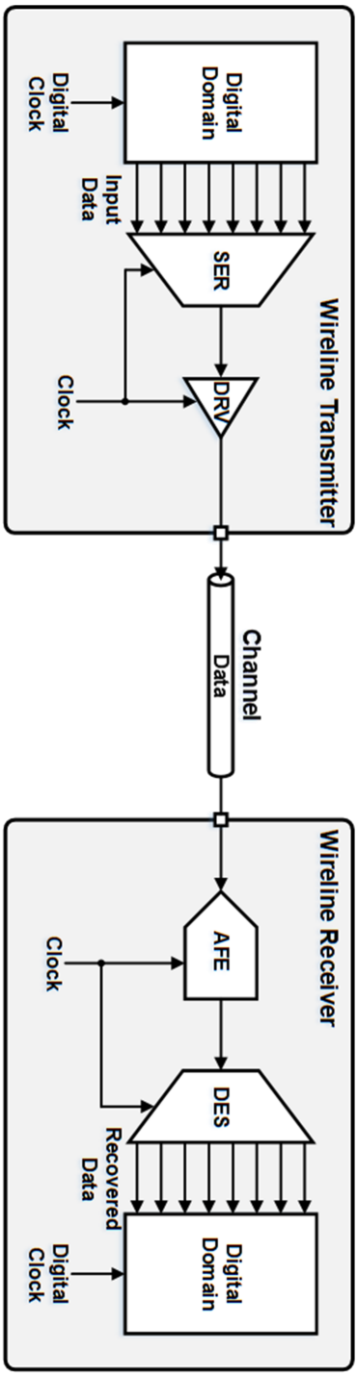


Fig. 2.1 Block diagram of the SerDes interface

As the amount of data to process continuously rises, most the wireline communication systems use the SerDes interface to cope with increased throughput. As a result, the required bandwidth per pin keeps increasing, and several issues arise from the high-speed data rate. The signal-to-noise ratio (SNR) of the transmitted data is degraded by high-frequency channel loss and crosstalk. Various equalization schemes are adopted to compensate for channel loss and crosstalk [53] - [65] . Additionally, the timing margin of the receiver becomes tight, so the robust design of timing circuits is directly related to the performance of the receiver. Therefore, in order to understand the timing circuit properly, understanding the clocking architecture of SerDes is a priority.

To accurately receive and decode transmitted data, the transmitter and receiver clocks must be synchronized or capable of operating asynchronously. The SerDes clocking architecture is classified according to the types of synchronization scheme between transmitter and receiver clocks. Table 2.1 shows the classification of signal-clock synchronization [66] . Among the five types of signal-clock synchronization, three types are mainly adopted in SerDes architecture: synchronous, mesochronous, and plesiochronous.

Table 2.1 classification of signal-clock synchronization [66]

Classification	Synchronous	Mesochronous	Plesiochronous	Periodic	Asynchronous
Periodicity	O	O	O	O	X
$\Delta\varphi$	0	φ_c	Varies	-	-
Δf	0	0	$f_d < \varepsilon$	$f_d > \varepsilon$	-

Synchronous clocking architecture is a clocking scheme with no frequency and phase differences between the receiver clock and the transmitted data. It means that there is no need to additionally adjust the clock phase for data sampling. Typically, synchronous clocking is implemented with forwarded-clocking architecture. Fig. 2.2 is an example of synchronous clocking architecture with a forwarded-clocking scheme. In order to operate synchronous clocking architecture reliably, the delay of the data channel and the clock channel should be matched, and the delay of the transmitter side (clock to driver delay) and the receiver side (clock buffer delay) should be matched or within a reasonable range. If the aforementioned delays are controlled tight enough to guarantee robust operation, synchronous clocking architecture is a simple and effective way to construct a SerDes interface with low cost. However, it is difficult to take a synchronous clocking architecture in the modern SerDes interface. As the data rate goes up, the timing margin for the robust sampling decreases while the clock skew and propagation delay remain the same. Furthermore, synchronous clocking architecture is sensitive to delay variations in on-chip circuits and PCB backplane. For these reasons, synchronous clocking architecture is used in relatively low-speed applications.

Mesochronous clocking architecture is a clocking scheme with only a phase difference between the receiver clock and the transmitted data. Almost all systems using forwarded clocking architecture belong to this classification. Fig. 2.3 shows a typical case of a mesochronous SerDes system adopting forwarded clocking architecture. Since the frequency is the same as TX, only a delay adjustment is required to ensure optimal sampling timing. Such a delay adjustment method may be implemented manually or automatically. If the SerDes system has a pre-training sequence,

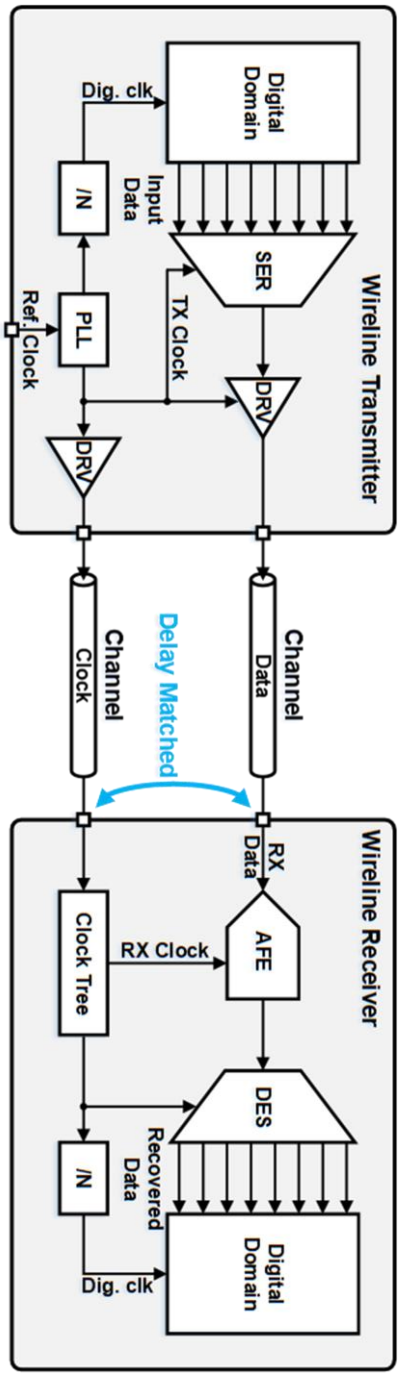


Fig. 2.2 Synchronous clocking architecture – forwarded clocking

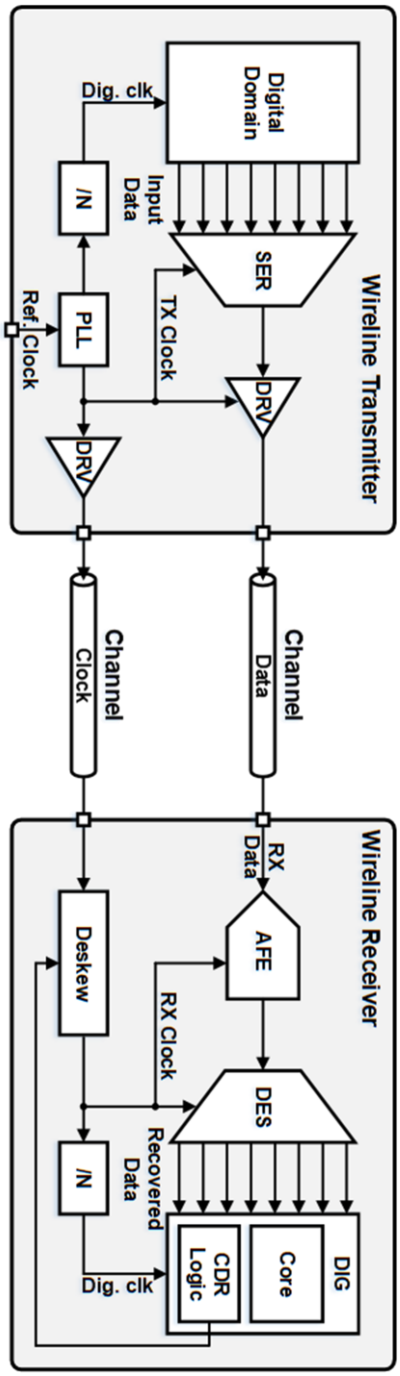


Fig. 2.3 Mesochronous clocking architecture – forwarded clocking

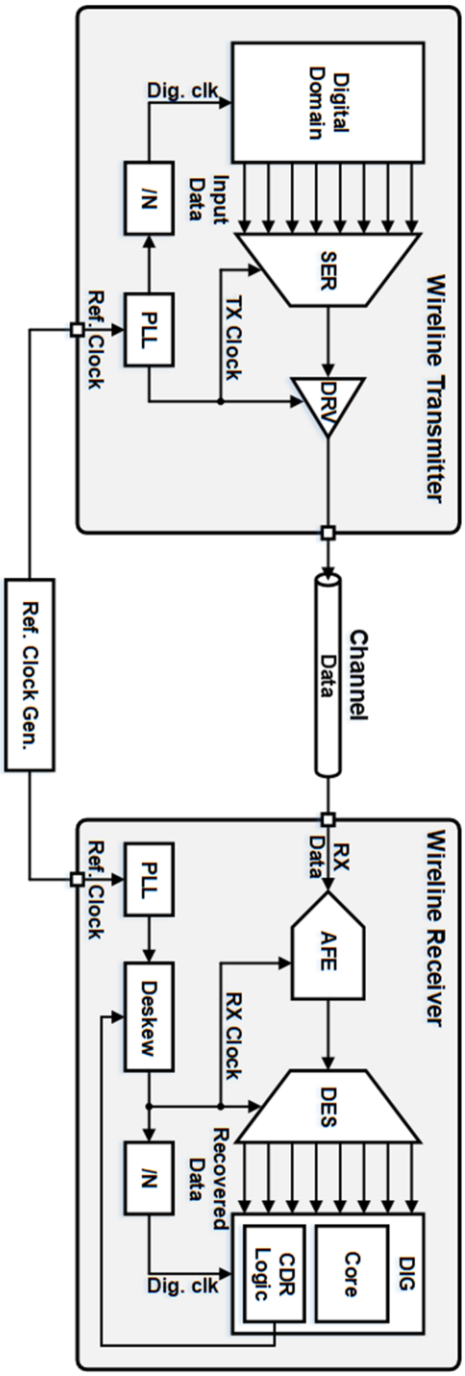


Fig. 2.4 Mesochronous clocking architecture – common reference clock

CDR is not required because the controller can perform delay adjustments. In this case, a variable delay line or phase-interpolator (PI) is sufficient. In contrast, CDR is required if the system does not have a controller. CDR can be performed with a simple deskewing circuit such as a delay-locked loop (DLL) or a PI. Meanwhile, the mesochronous system may also be configured with a common clock architecture rather than forwarded clocking architecture. Fig. 2.4 is an example of the mesochronous system using common clock architecture. Instead of a transmitted clock, the receiver phase-locked-loop (PLL) generates a clock having the same frequency as the transmitter. Then, CDR with the aforementioned deskewing circuit adjusts the clock phase. Common clock architecture is widely used in PCIe applications [67]. Although mesochronous clocking architecture requires an additional clock channel or external reference clock generator, it is widely used in some applications because of its simple CDR structure.

In the plesiochronous clocking architecture, there is a slight frequency difference between the transmitted data and the receiver clock. Fig. 2.5 shows an example of plesiochronous clocking architecture. Since there is no common clock, the transmitter and receiver generate clocks with their own reference clock. Thus, each clock cannot be synchronized, resulting in a slight frequency difference. This frequency difference causes a continuous drift in the relative phase of the clocks. An elastic buffer can be used to resolve this problem, but it is valid when there is little attenuation of data. Most receivers need to regenerate the attenuated data; thereby, CDR is required in general. Unlike mesochronous systems, CDR in plesiochronous systems must track not only phase but also frequency. Consequently, frequency acquisition ability must be added to the receiver, which makes the design of the receiver challenging.

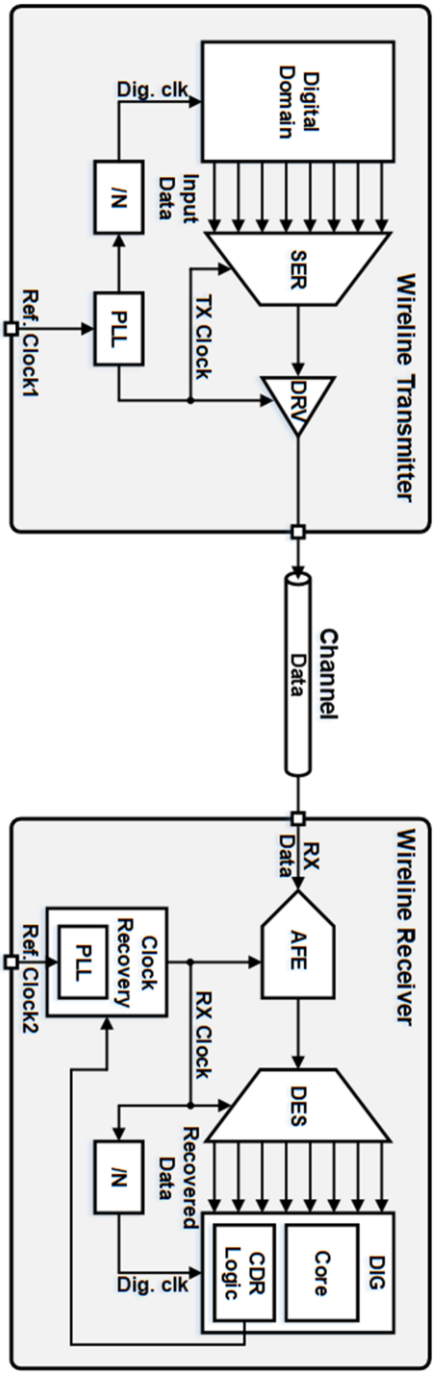


Fig. 2.5 Plesiochronous clocking architecture

Still, plesiochronous clocking architecture is generally used in the system where the addition of a clock channel is costly or impossible.

2.2 General Considerations for Clock and Data Recovery

2.2.1 Overview

One of the critical tasks in building a high-speed analog SerDes interface is getting the receiver clock to be properly aligned to the incoming data. Since most of the SerDes interface is mesochronous or plesiochronous clocking architectrues, in order to perform synchronous operations such as retiming and demultiplexing on random data, receivers must generate a clock synchronized with input data. Clock alignment is usually done using a feedback system called clock and data recovery (CDR) block. The CDR circuitry creates a clock signal that is aligned to the phase and, to some extent, the frequency of the transmitted signal. Fig 2.6 illustrates a role of a CDR circuit. The clock recovery circuit senses the data and produces a periodic clock. As shown in Fig. 2.7, the decision circuit samples the noisy data at the rising edge of the recovered clock. This series of processes, called CDR, allows the receiver to regenerate noisy data into clean data. In order for the CDR to operate properly, the recovered clock must satisfy three conditions [68] . First, the recovered clock must have a frequency equal to the data rate. Second, it must have a certain phase relationship with respect to data, enabling optimal sampling of the input data. Third, it must exhibit a small jitter as it is the principal contributor to the retimed data jitter.

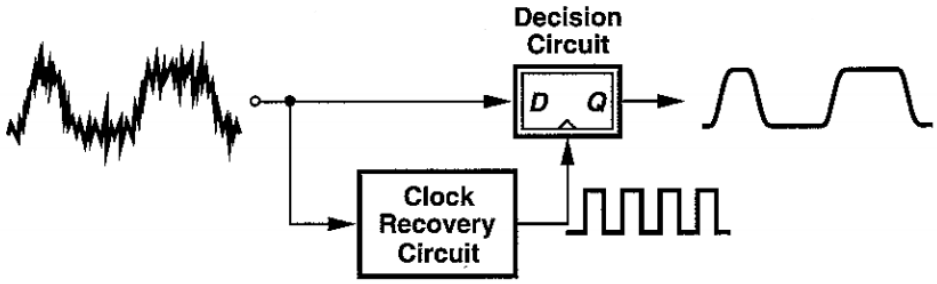


Fig. 2.6 Role of a CDR circuit in retiming data [68]

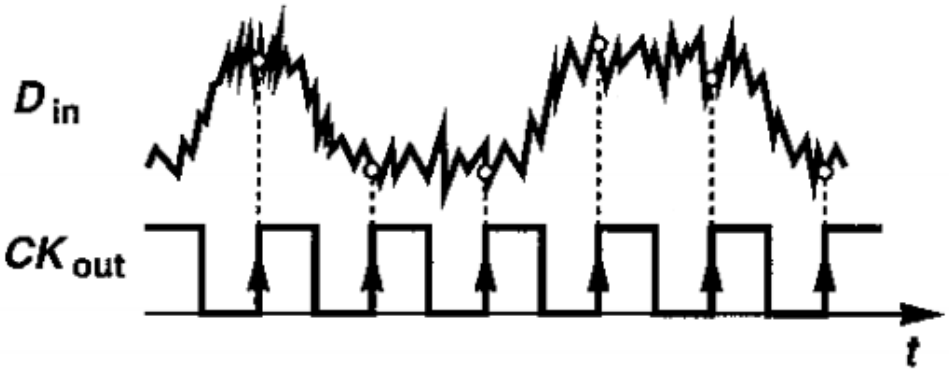


Fig. 2.7 Optimum sampling of noisy data [68]

2.2.2 Jitter

According to the SONET standard, jitter is defined as the short-term variations of a digital signal's significant instants from their ideal positions in time [69] . In short, jitter is timing errors within a system. The presence of jitter changes the edge positions with respect to the sampling point. An error will occur when a data edge falls on the wrong side of a sampling instant. Jitter is mainly expressed in unit-interval (UI), which is the minimum time interval between conditional changes in the data transmission signal, also known as pulse time or symbol duration time.

Since jitter has a random component, it cannot be accurately predicted at a specific time. Therefore, jitter must be specified using statistical terms such as mean value and standard deviation. Jitter analysis is important to estimate the performance of the receiver because the data-eye opening can be estimated as the amount of total jitter at a specific bit error rate (BER). To accurately estimate the amount of total jitter, the total jitter must be decomposed into random jitter (RJ) and deterministic jitter (DJ).

2.2.2.1 Random Jitter (RJ)

Random jitter comes from processes that are truly random noise sources, such as thermal noise and flicker noise. Since random jitter cannot be accurately predicted in a given cycle, statistical approaches are used to analyze the amount of random jitter. The most useful method to measure the random jitter is finding the standard deviation of the random distribution, which is the same as the root mean square (RMS) jitter.

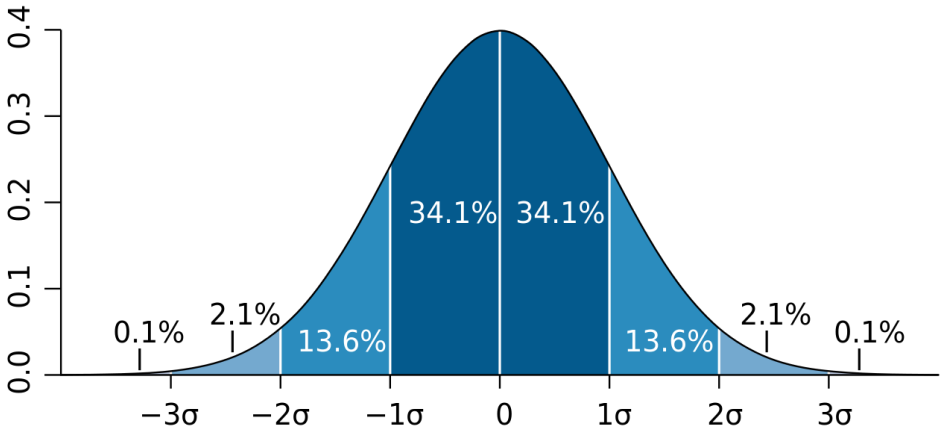


Fig. 2.8 The normal distribution [70]

As depicted in Fig. 2.8, random jitter is assumed to follow Gaussian distribution and to be uncorrelated with other system noise sources. The two tails extending away from the center of the curve asymptotically approach zero, but not zero. Therefore, a probability density function (pdf) of random jitter is considered to be unbounded as test time increases. The BER caused by the random jitter is proportional to its standard deviation σ. Moreover, if there are multiple sources of uncorrelated random jitter, the total amount of random jitter is the RMS of each random jitter.

$$\sigma_{\text{total}} = \sqrt{\sigma_1^2 + \sigma_2^2 + \cdots + \sigma_n^2}$$

2.2.2.2 Deterministic Jitter (DJ)

Deterministic jitter is jitter with a non-Gaussian pdf. It has identifiable causes and is always bounded in amplitude. The sources of deterministic jitter are generally related to imperfections in the behavior of a device or transmission medium: limited bandwidth of the signal path, signal reflection, clock duty-cycle distortion, switching power supply noise, crosstalk from neighboring signals, and electromagnetic interference (EMI) of surroundings. Deterministic jitter consists of four components; data-dependent jitter (DDJ), duty cycle distortion (DCD), bounded uncorrelated jitter (BUJ), and sinusoidal jitter (SJ). The way to quantify DJ is to use peak-to-peak value since it is bounded [72] .

Data-dependent jitter (DDJ), also known as pattern jitter or inter-symbol interference (ISI), is the timing error correlated with the bit sequence in a data stream. The terms "data-dependent jitter" and "pattern jitter" are used to describe the effects of jitter in the time domain, while the terms "ISI" are more commonly applied in the frequency domain. DDJ appears in the form that the response of the current bit is affected by the response of the previous bit. The primary source of DDJ is the limited bandwidth of the signal path. The high frequency signal has a shorter settling time than the low-frequency signal. This changes the starting conditions of the transition at different frequencies and results in timing errors depending on the data pattern applied.

Pulse width distortion (PWD) or duty cycle distortion (DCD) is the deviation in duty cycle value from the ideal value. For many serial data systems, this value is equal to the bit time deviation of 1 and 0 bits. It can also be defined as the difference in

propagation delay between high and low latency at low latency. PWD causes distortion in the eye diagram where the eye crossings are offset up or down from the vertical midpoint of the eye. The main source of the PWD is timing differences between rising and falling edges within a system. It can also be caused by ground shift if the signal is single-ended or by voltage offsets between differential inputs if the signal is differential.

Bounded uncorrelated jitter (BUJ) is any deterministic jitter that is bounded in amplitude but uncorrelated to the data pattern. Generally, the source is interference from other signal sources such as EMI, capacitive and inductive coupling, and power supply switching noise.

Sinusoidal jitter (SJ) or periodic jitter (PJ) refers to the sinusoidal or periodic variations of the rising and falling edges of the signal within time. The primary source of the SJ is interference from signals that are related to the data pattern. In addition, ground bounce and other power supply variations are common causes, although the levels of sinusoidal jitter normally encountered are very low.

2.2.2.3 Total Jitter (TJ)

Total jitter (TJ) is defined as the amount of eye closure due to jitter at a particular BER [71] . The peak-to-peak total jitter can be obtained by subtracting the eye-opening at a specific BER level at the bit interval. Since total jitter is composed of deterministic jitter (DJ) and random jitter (RJ), TJ pdf is the convolution of the Gaussian distribution (RJ pdf) and the non-Gaussian distribution (DJ pdf). Typically, TJ pdf is

modeled using a dual-Dirac model. In the dual-Dirac model, DJ is assumed to follow a distribution formed by two Dirac-delta functions. Then, TJ distribution can be considered as two Gaussian distributions over two tail regions, which is shown in Fig. 2.9. In other words, TJ model is Gaussian approximation to the outer edges of the jitter distribution displaced by DJ(δδ), where DJ(δδ) is $|\mu_L - \mu_R|$ [73] .

Alternatively, DJ distribution can be modeled as sinusoidal or uniform distribution. Fig. 2.10 and 2.11 shows the resulting distribution of the total jitter. Note that the central part of the dual-Dirac distribution may not match the actual distribution. However, it does not matter because an important feature of the dual-Dirac distribution is that the Gaussian tails can be matched to the tail of the actual jitter, which enables the estimation of the BER.

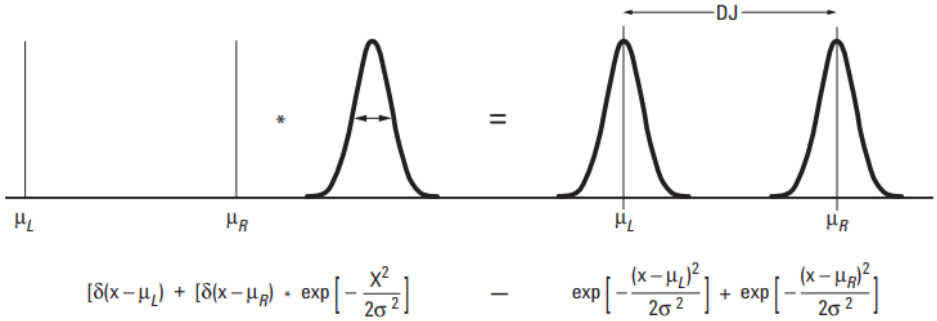


Fig. 2.9 The convolution of the sum of two delta functions separated by DJ and Gaussian RJ distribution of width σ [73]

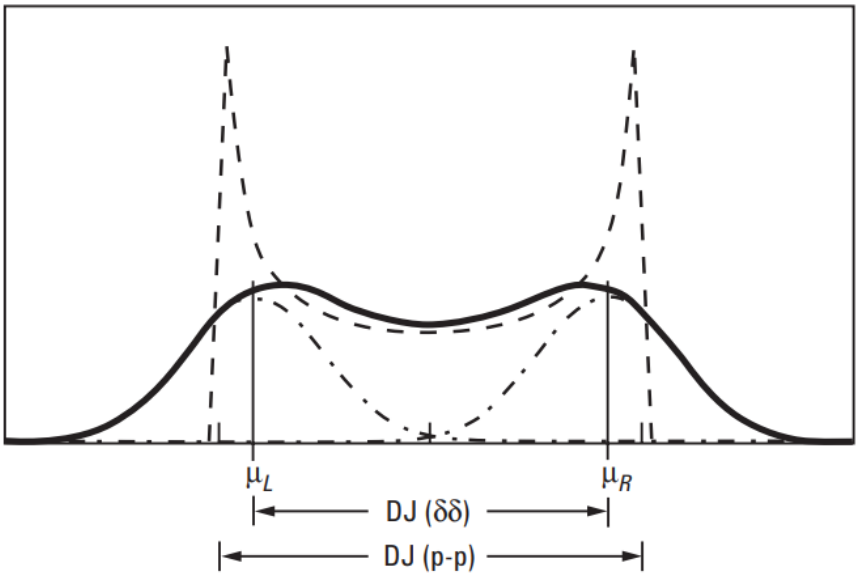


Fig. 2.10 Application of the dual-Dirac model (DJ: sinusoidal distribution) [73]

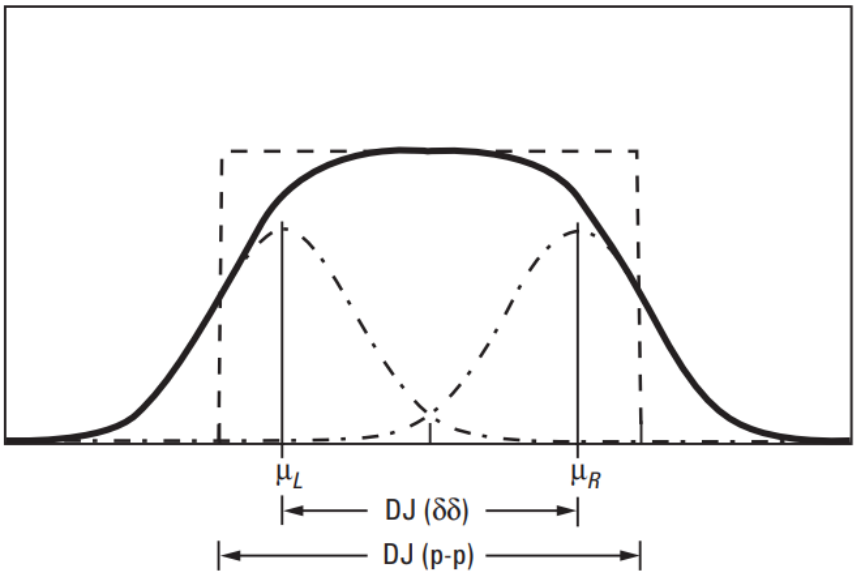


Fig. 2.11 Application of the dual-Dirac model (DJ: uniform distribution) [73]

Therefore, if the standard deviation of the RJ (σ) and the DJ(δδ) is given, peak-to-peak total jitter can be approximated as follows

$$\mathrm{TJ(BER)} = DJ(\delta\delta) + 2Q_{BER} \times \sigma$$

where Q_{BER} is calculated from the complementary error function, as given in Table 2.2.

Table 2.2 Values of Q_{BER}

BER	**10^{-5}**	**10^{-6}**	**10^{-7}**	**10^{-8}**	**10^{-9}**	**10^{-10}**	**10^{-11}**	**10^{-12}**	**10^{-13}**
Q	**4.26**	**4.75**	**5.20**	**5.61**	**6.00**	**6.36**	**6.71**	**7.03**	**7.35**

2.2.3 CDR Jitter Characteristics

2.2.3.1 Jitter Generation

Jitter generation refers to the amount of jitter produced by a CDR circuit itself when jitter-free data is applied to its input. Generally, as shown in table 2.3, jitter specifications stated in wireline standards are 0.1UI peak-to-peak and 0.01UI RMS. Factors that generate jitter are as follows: First, device noise, including voltage-controlled oscillator (VCO) intrinsic phase noise; Second, ripple on the control voltage; Third, supply and substrate noise. Fig. 2.12 shows the closed-loop jitter of the oscillator. The jitter rises with the square root of time until $t_1 = \frac{1}{2\pi f_u}$, where f_u is the loop bandwidth. Then, it is saturated thereafter with the effect of noise shaping by PLL.

Table 2.3 SONET Jitter Generation [69]

OC-N/STS-N Level	High-Pass Filter Cutoff	Low-Pass Filter Cutoff (B_3)	Jitter Generation Limit
1	12 kHz	400 kHz	0.1 UI_{pp} and 0.01 UI_{rms}
3	12 kHz	1.3 MHz	0.1 UI_{pp} and 0.01 UI_{rms}
12	12 kHz	5 MHz	0.1 UI_{pp} and 0.01 UI_{rms}
48	12 kHz	20 MHz	0.1 UI_{pp} and 0.01 UI_{rms}
192	20 kHz	80 MHz	0.3 UI_{pp}
	4 MHz		0.1 UI_{pp}
768[a]	20 kHz	320 MHz	1.2 UI_{pp}
	16 MHz		0.1 UI_{pp}

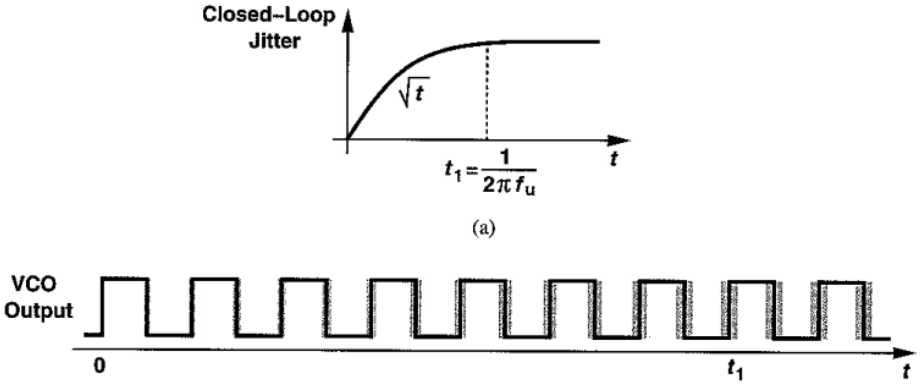


Fig. 2.12 Accumulation of cycle-to-cycle jitter in a phase-locked oscillator:

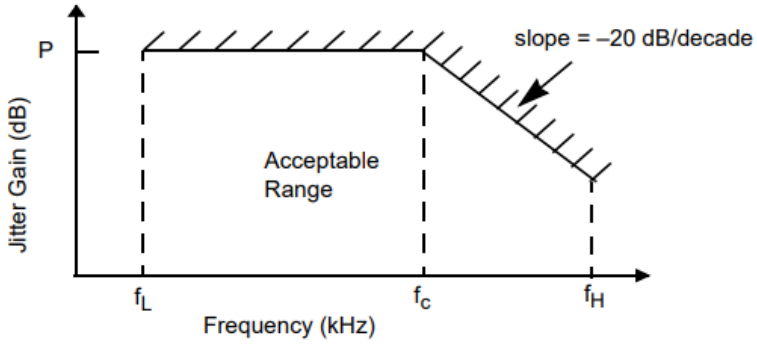
(a) actual behavior (b) conceptual behavior [68]

2.2.3.2 Jitter Transfer

The jitter transfer function represents the amount of jitter attenuation at a particular offset frequency. In other words, jitter transfer is a measure of the amount of jitter transferred from the input to the output of the system. Jitter transfer is important for cascaded clock recovery circuits in long-distance transmission systems because if the jitter is amplified, it should not exceed the jitter tolerance of the subsequent system. Hence, jitter transfer measurement is required to confirm that there is no amplification of jitter by network elements in the transmission system.

From the CDR point of view, the jitter transfer function shows a frequency response at which the input jitter starts to be filtered. For example, if the input jitter varies slowly, the output must follow the input to ensure phase-locking. On the other hand, if the input jitter varies rapidly and exceeds the bandwidth of the CDR circuit,

the CDR cannot keep up with the input jitter and filters it. Thus, the jitter transfer exhibits the same low-pass characteristics as PLL. Fig. 2.13 shows the jitter transfer mask stated in SONET.



OC-N/STS-N Level	f_L (kHz)	f_c (kHz)	f_H (MHz)	P (dB)
1	0.4	40	0.4	0.1
3	1.3	130	1.3	0.1
12	5	500	5	0.1
48	20	2000	20	0.1
192	10	1000	80	0.1
768[a]	40	4000	320	0.1

Fig. 2.13 Jitter Transfer Mask [69]

2.2.3.3 Jitter Tolerance

Jitter tolerance (JTOL) is the amount of peak-to-peak jitter that can be applied at a particular BER to the extent that the PLL does not lose its lock. The jitter tolerance is affected by both the loop bandwidth and the offset frequency. If the frequency of the jitter is lower than the CDR loop bandwidth, the PLL simply passes through the jitter with little attenuation, resulting in a high jitter tolerance. Likewise, a PLL with a low loop bandwidth will have lower jitter tolerance than a wideband PLL. Here, a trade-off relationship is established in the CDR design. If the CDR bandwidth is decreased to lower the jitter transfer bandwidth, then jitter tolerance degrades substantially at high jitter frequencies.

The approximate condition of maximum phase error of the recovered clock to prevent BER increase is 0.5UI, which is very close to the zero-crossing points of data. It can be mathematically expressed as follows.

$$\varphi_{in} - \varphi_{out} < \frac{1}{2}\mathrm{UI} \tag{2.1}$$

Since $\varphi_{out} = \varphi_{in} H(s)$, where $H(s)$ is the jitter transfer function, equation 2.1 can be modified as

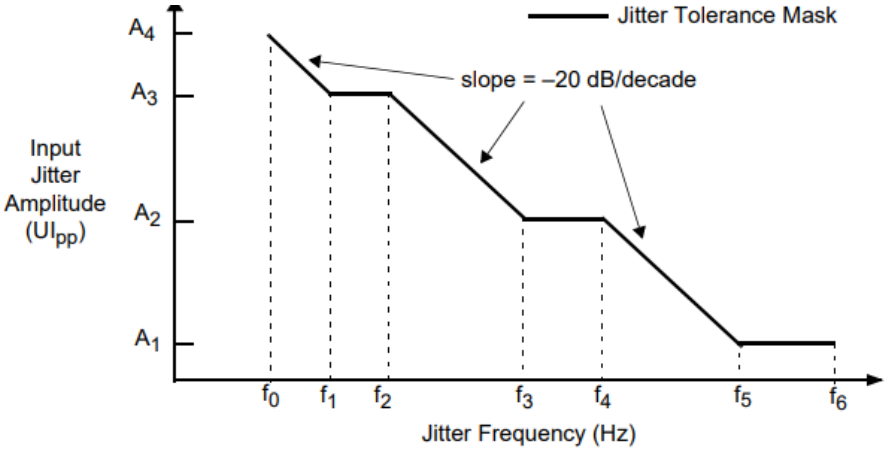
$$\varphi_{in}[1 - H(s)] < \frac{1}{2}\mathrm{UI}$$

$$\varphi_{in} < \frac{0.5\mathrm{UI}}{1 - H(s)}$$

Therefore, the mathematical expression of jitter tolerance is

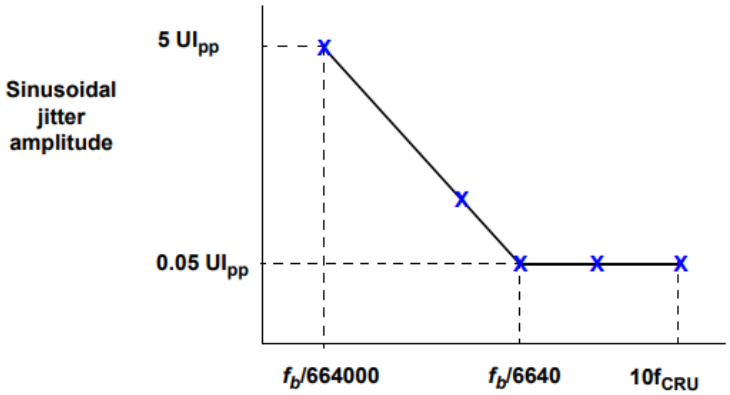
$$JTOL(s) = \frac{0.5}{1 - JTF(s)}$$

Fig. 2.14 and Fig. 2.15 shows the jitter tolerance mask specified in SONET and OIF-CEI, respectively.



OC-N/ STS-N Level	f_0 (Hz)	f_1 (Hz)	f_2 (Hz)	f_3[a] (Hz)	f_4 (kHz)	f_5[a] (kHz)	f_6[a] (MHz)	A_4 (UI_{pp})	A_3 (UI_{pp})	A_2[a] (UI_{pp})	A_1[a] (UI_{pp})
1	10	NA	41.3	100	2	20	0.4	NA	3.63	1.5	0.15
3	10	NA	68.7	500	6.5	65	1.3	NA	10.9	1.5	0.15
12	10	18.5	100	1000	25	250	5.0	27.8	15	1.5	0.15
48	10	70.9	500	5000	100	1000	20	106.4	15	1.5	0.15
192	10	296	2000	20000	400	4000	80	444.6	15	1.5	0.15
768[b]	10	1184	8000	20000	400	16000	320	1776	15	6.0	0.15

Fig. 2.14 SONET Jitter Tolerance Mask 118[69]



Frequency Range	Sinusoidal jitter, peak-to-peak (UI)
$f < f_b/664000$	Not Specified
$f_b/664000 < f \leq f_b/6640$	$5*f_b/(664000*f)$
$f_b/6640 < f \leq 10f_{CRU}$	0.05

Fig. 2.15 OIF-CEI Jitter Tolerance Mask [74]

- **f_b : maximum baud-rate supported by channel**
- **f_{CRU} : jitter corner frequency given by $f_b/6640$**

2.3 CDR Architectures

CDR architectures can be classified by the existence of feedback control. It is related to the phase relationship between the received input data and the local clock at the receiver, which is described in Chapter 2.1. Commonly used CDR topologies may be divided into three major categories: Topologies using feedback phase tracking, an oversampling-based topology without feedback phase tracking, and topologies using phase alignment but without feedback phase tracking [75] . However, except for the first category, it has been rarely used in wireline systems recently. In this chapter, we will discuss some types of CDRs in the first category that are still widely used today.

2.3.1 PLL-based CDR – with external reference clock

The PLL-based CDRs can be classified according to the existence of a reference clock. The CDR without a reference clock will be discussed later to see the PLL-based CDR with a reference clock first.

In order to lower the VCO jitter from the ripple of the control voltage, it is desirable to decompose the VCO control into coarse and fine inputs [76] . A CDR architecture utilizing such a scheme is depicted in Fig. 2.16. In this CDR architecture, two voltage-controlled oscillators (VCOs) are used for the two tracking loops. VCO2 is a replica of VCO1, and the gain of VCO2 is greater than the gain of VCO1. Here, the phase tracking loop with a phase detector (PD) locks the phase of VCO1 into input data

through fine control. Because the gain of VCO1 is relatively lower than that of a conventional single-loop PLL, ripples of the fine control voltage introduce less jitter than the conventional one. The frequency tracking loop with the phase-frequency detector (PFD) locks the phase and frequency of VCO2 output to the phase and frequency of the input reference clock. In general, since the frequency of the reference clock is lower than the input data rate, a divider is used to increase the frequency of the output clock by N times than the reference clock. Since VCO1 and VCO2 are identical, the control voltage of the frequency tracking loop can be used in the coarse control of the phase tracking loop.

CDR architecture using dual VCO has an advantage in reducing ripples, whereas

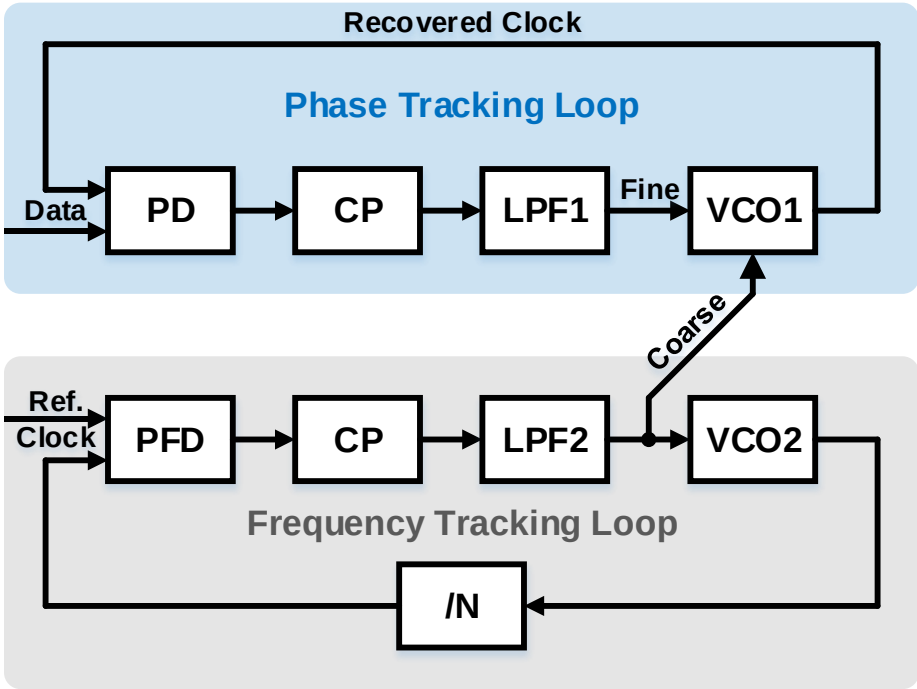


Fig. 2.16 PLL-based CDR architecture with dual VCO.

it has several design issues. The first one is the center frequency mismatch due to random mismatches between the two VCOs. Even if two VCOs are identical and share the same coarse control voltage, there can be a difference in the operating frequency due to PVT variation and mismatch. For this reason, the phase tracking loop must still achieve a sufficiently wide capture range to guarantee locking even with the initial frequency difference [76] . The second issue is the frequency mismatch between incoming data rate and lock frequency (Nf_{REF}) of the frequency tracking loop. There can be an error of 5 to 10 ppm between the frequency of transmitted data and Nf_{REF}. As a consequence, VCO1 and VCO2 may pull each other through the substrate or supply lines. If the bandwidth of the frequency tracking loop is larger than that of the phase tracking loop, the pulling caused by VCO2 can be corrected, but the pulling caused by VCO1 cannot be fixed. Another design issue is area overhead. Employing two PLL requires a large area, and this area issue may worsen the aforementioned mismatch problem. If the VCO type is an LC oscillator, this problem becomes worse.

Fig. 2.17 describes a relatively simple CDR architecture that acquires frequency and phase in two steps. In this architecture, the frequency tracking loop is enabled first to lock the oscillator to Nf_{REF}. When the frequency error between the PFD inputs falls to a sufficiently low level, the lock detector detects it and determines the locked state. Then, the frequency tracking loop is disabled, and the phase tracking loop is enabled to lock the VCO to the data. Though the frequency lock is achieved, the lock detector should continue to operate as the CDR may lose the lock due to unexpected noise. On the other hand, this CDR architecture has a critical design issue which needs to be verified thoroughly: loop transition. After the frequency lock, loop transition

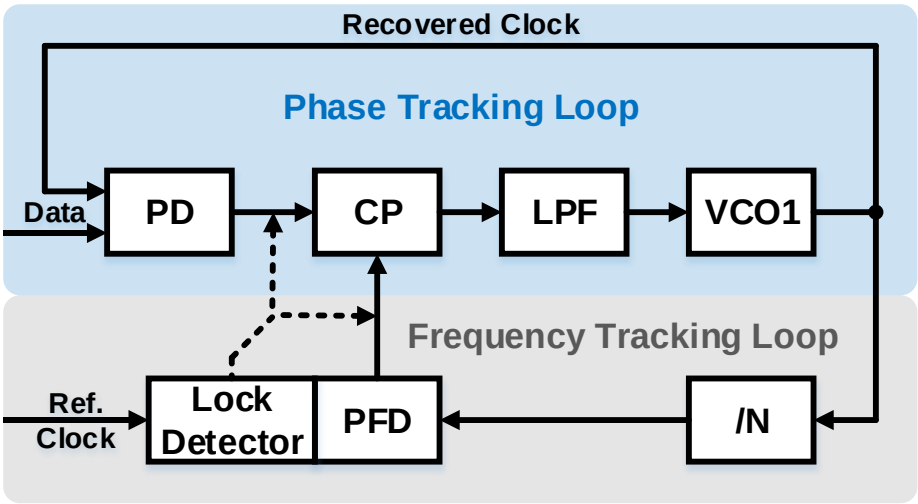


Fig. 2.17 PLL-based CDR architecture using sequential locking.

occurs from frequency tracking loop to phase tracking loop. Here, if the transition disturbs the control voltage significantly, VCO frequency may jump out of the capture range of the phase tracking loop, resulting in a failure of phase locking.

The aforementioned CDR architectures use analog PLL, which adopts a charge pump (CP), analog low-pass loop filter and VCO. Unlike the CDR architectures above, an all-digital PLL (ADPLL) based CDR uses a digital loop filter (DLF) and a digitally-controlled oscillator (DCO) instead of CP, analog LPF, and VCO. Fig. 2.18 shows the simplified block diagram of ADPLL-based CDR. Compared to analog PLL-based CDR, ADPLL-based CDR has several advantages. Since the loop components are mostly in the digital domain, the CDR loop becomes less vulnerable to process, voltage and temperature (PVT) variations and external noise sources. In addition, using digital logic rather than the analog components can minimize the required layout

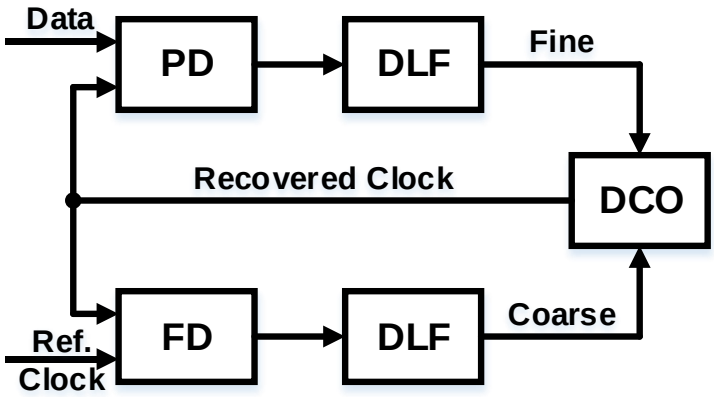


Fig. 2.18 ADPLL-based CDR architecture

area because it can be combined with other digital elements of the receiver, such as the decision-feedback equalizer (DFE) adaptation loop. Moreover, programmability resulting from digital implementation makes it easy to control loop parameters such as proportional gain and integral gain after chip manufacturing. However, the digital implementation of the CDR loop also has drawbacks. Compared to analog PLL-based CDR, ADPLL-based CDR shows longer latency because the input data and PD outputs must be deserialized to be processed in a digital domain. A long latency in the digital loop introduces a larger limit cycle as well as the jitter; thus, it can degrade the phase and frequency tracking ability and the stability of the loop [77] . Moreover, dithering due to the finite resolution of the DCO is translated to jitter.

2.3.2 DLL/PI-based CDR

Delay-locked loop (DLL)-based CDR and phase interpolator (PI)-based CDR are similar to PLL-based CDR using dual VCOs. Instead of VCO1, DLL or PI adjusts the clock phase to fit the input data. Instead of the control voltage, the frequency tracking loop passes the frequency-adjusted clock to the DLL or PI. As a result, DLL or PI-based CDR avoids the drawbacks of multi-VCOs: pulling and mismatch.

One of the most significant merits of using DLL or PI is the stability of the system. By directly controlling the clock phase, the pole is not generated in the closed-loop transfer function of PLL. Furthermore, DLL and PI loops are 1-st order systems; there is no jitter peaking in the transfer function. For these reasons and the advantage of sharing a frequency tracking loop, DLL or PI-based CDR is appropriate for multi-channel integration.

Fig. 2.20 shows an example of DLL-based CDR. A frequency tracking loop is required because DLL does not work with a frequency offset. In the forwarded clocking system, the frequency tracking loop can be removed, as depicted in Fig. 2.21. However, DLL suffers from several issues, including harmonic locking and stuck problems. These problems stem from the initial condition and limited delay range of the voltage-controlled delay line (VCDL).

Fig. 2.22 and 2.23 illustrates an example of PI-based CDR. It has the same dynamics and structure as DLL-based CDR, but there are several differences. First, the loop filter and control block is implemented in the digital domain. PI-based CDR also requires a frequency tracking loop, but it also works when the frequency offset is small. Furthermore, the delay range is unlimited with the phase-rotator.

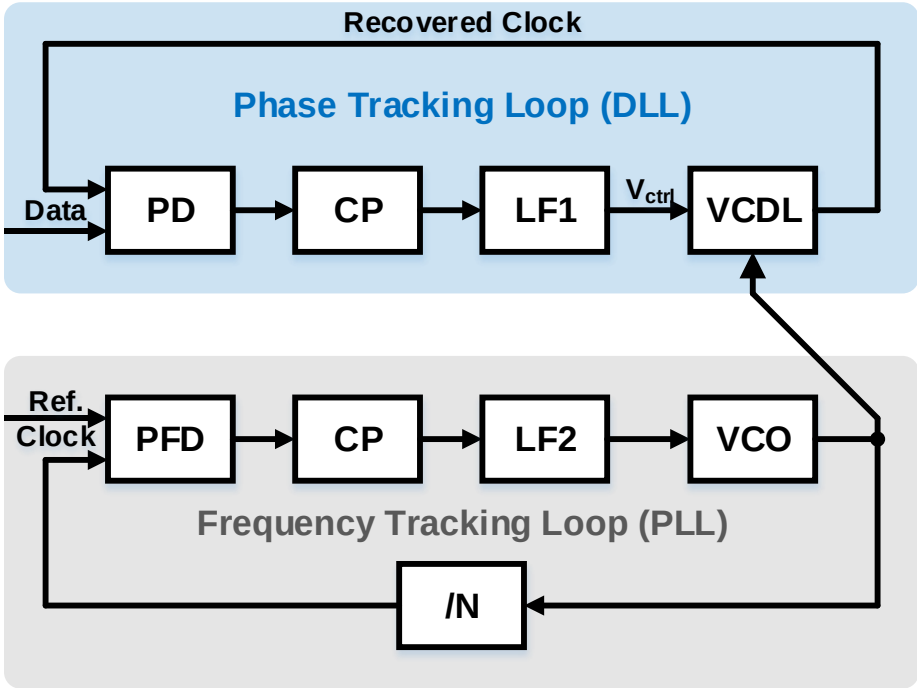


Fig. 2.20 DLL-based CDR architecture

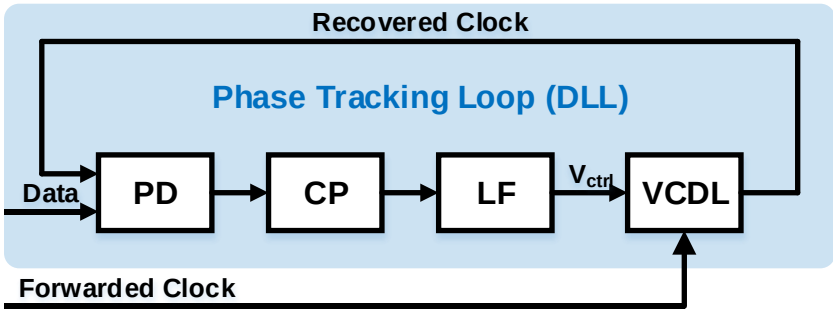


Fig. 2.21 DLL-based CDR architecture : forwarded clocking

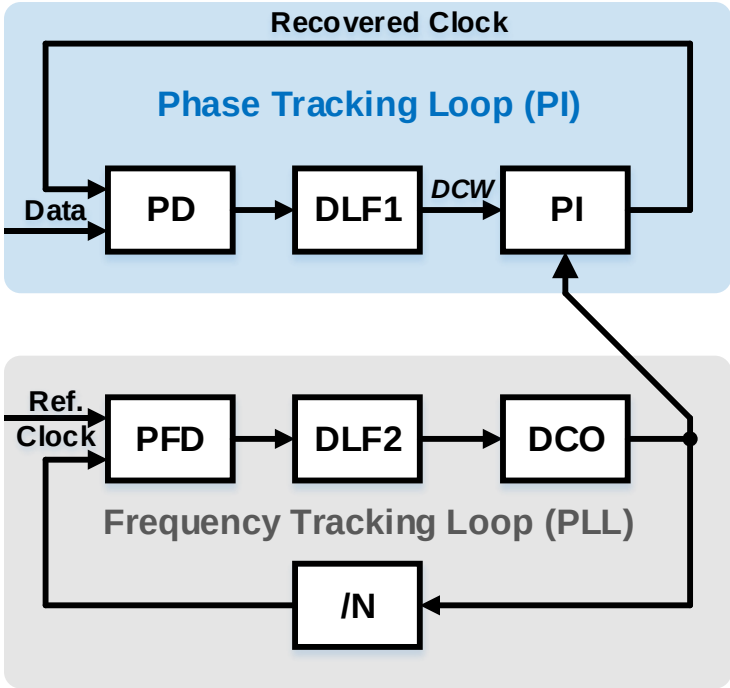


Fig. 2.22 DLL-based CDR architecture

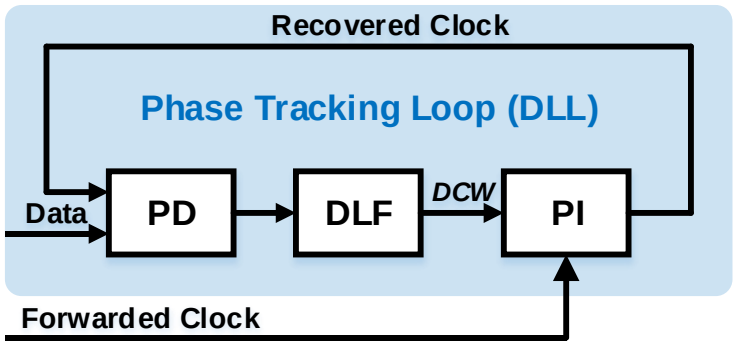


Fig. 2.23 DLL-based CDR architecture : forwarded clocking

2.3.3 PLL-based CDR – without external reference clock

PLL-based CDR architecture without an external reference clock is referred to as a referenceless PLL-based CDR or referenceless CDR. While frequency detectors (FDs) in PLL-based CDR with an external reference compare the frequency of the VCO and the reference clock, frequency detectors in referenceless CDR compare the frequency of the input data and the recovered clock. That is, the referenceless CDR extracts the frequency information from the incoming data stream and automatically adjusts the recovered clock frequency to the input data rate.

There are two types of referenceless CDR architecture: dual-loop and single-loop referenceless CDR. Fig. 2.24 and Fig. 2.25 show the dual-loop and single-loop referenceless CDR architecture, respectively. As shown in Fig. 2.24, dual-loop architecture use a separate FD to achieve the frequency acquisition [13] [15] [17] [18] [23] [32] , [48] [80] . During CDR initiation or phase lock loss, the FD is activated to generate a control signal through the digital loop filter (DLF) to move the DCO oscillation frequency to the data rate. When the frequency difference falls within the capture range of the phase tracking loop, the PD takes over and adjusts the clock phase to the data phase. The rotational FD and the Pottbacker FD are the most popular choice for the dual-loop referenceless CDR [78] . The detailed operation of the FD is stated in Chater 2.4.1. The bandwidth of the frequency tracking loop should be much smaller than that of the phase tracking loop because short-term spectral lines of random input data, which are close but unequal to the nominal data rate, may occasionally appear,

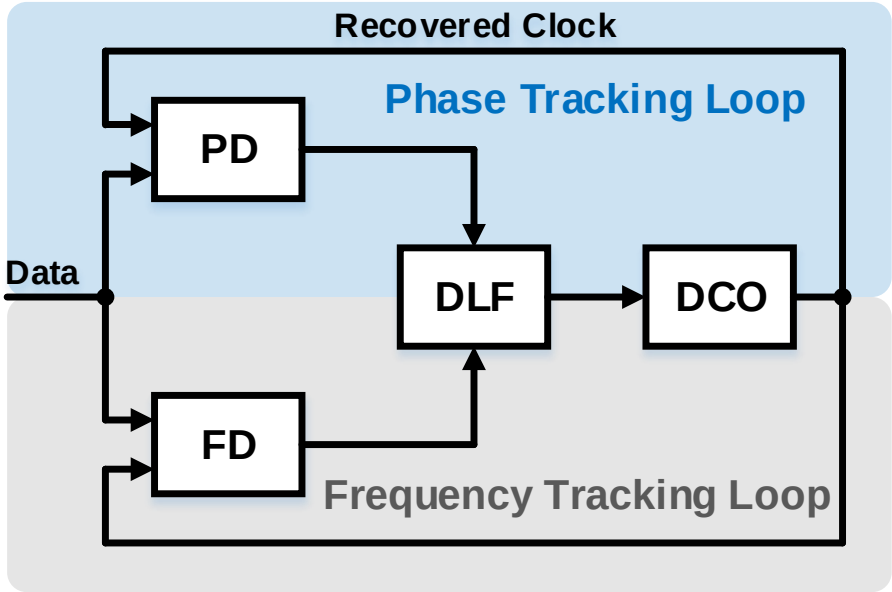


Fig. 2.24 Dual-loop referenceless CDR architecture

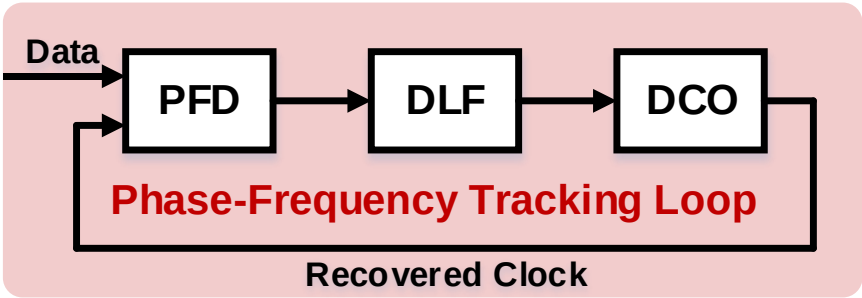


Fig. 2.25 Single-loop referenceless CDR architecture

possibly confusing the frequency detector [76] . However, most separate FDs are sensitive to input jitter, and dual-loop architectures suffer from increased hardware complexity and inherent loop interference. Therefore, as shown in Fig. 2.25, the single-loop referenceless CDR has been widely explored in recent works [19] - [21] , [24] - [26] , [33] , [43] [45] [47] , [50] - [52] , [79] . It has the advantage of simplicity in comparison to the dual-loop architecture.

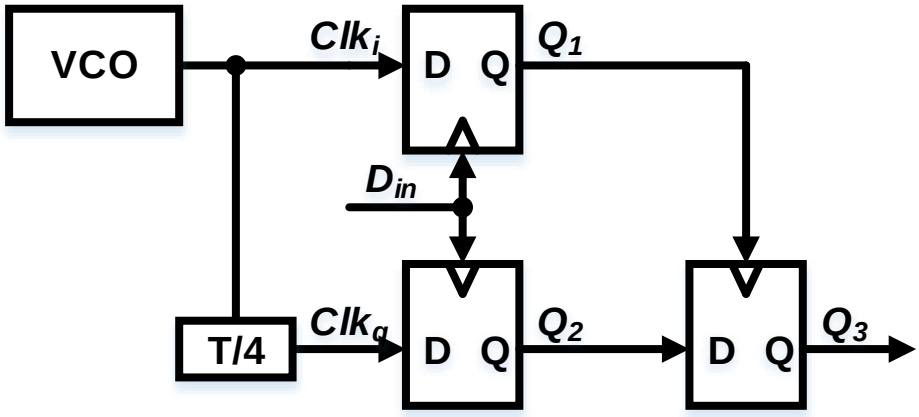
Besides the structural aspects described above, the frequency acquisition scheme is the most challenging task in the referenceless CDR. Several frequency acquisition schemes for the referenceless operation are introduced in Chapter 2.4.2.

2.4 Frequency Acquisition Scheme

2.4.1 Typical Frequency Detectors

2.4.1.1 Digital Quadricorrelator Frequency Detector

Digital quadricorrelator frequency detector was proposed by A. Pottbacker in 1992 [78] . Fig. 2.26 shows the block diagram and the function of the Pottbacker

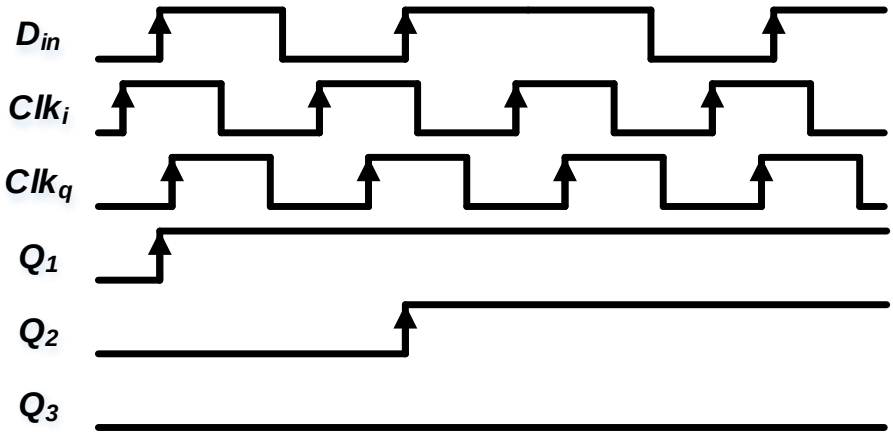


(a) Block Diagram

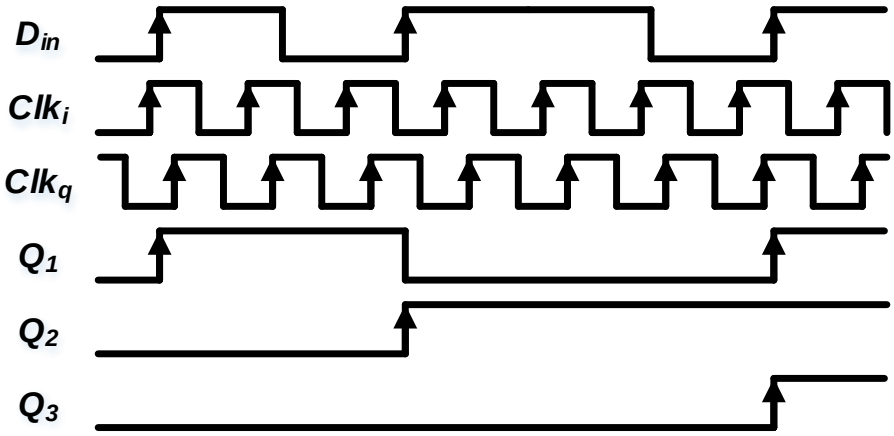
$f_{VCO} < f_D$	Q_1 leads Q_2	Q_3 : -1(low)
$f_{VCO} > f_D$	Q_1 lags Q_2	Q_3 : $+1$(high)

(b) Relationship among Q1, Q2, Q3.

Fig. 2.26 Pottbacker FD (known as DQFD)



(a)



(b)

Fig. 2.27 Timing diagrams of Pottbaker FD when the clock frequency is (a) lower and (b) higher than input data rate

FD. The FD consists of 3 flip-flops. The input non-return-to-zero (NRZ) data stream samples an in-phase clock and a quadrature-phase clock, which is delayed 1/4 of the clock period. If the VCO clock frequency is lower than the input data rate, Q_1 leads to Q_2, and output Q_3 keeps a logic low value. Likewise, if the VCO clock frequency is higher than the input data rate, Q_1 lags Q_2, and output Q_3 exports a logic high value. As a result, the average of Q_3 over a predetermined time window represents the polarity of frequency offset in the corresponding time window. A timing diagram to help understand the operation of Pottbaker FD is shown in Fig. 2.27.

The principle of quadricorrelator frequency detector is shown in Fig. 2.28. With the quadrature-phase of clocks, two PD (flip-flops) converts the clock and input data into $\cos(\Delta\omega t)$ and $\sin(\Delta\omega t)$ where $\Delta\omega = \omega_{clk} - \omega_{data}$. Thus, if $\Delta\omega$ is positive, then Q_1 leads Q_2, and if $\Delta\omega$ is negative, Q_1 lags Q_2 because $\sin(-\Delta\omega t) = -\sin(\Delta\omega t)$. In short, the phase difference between Q_1 and Q_2 shows the sign of frequency difference.

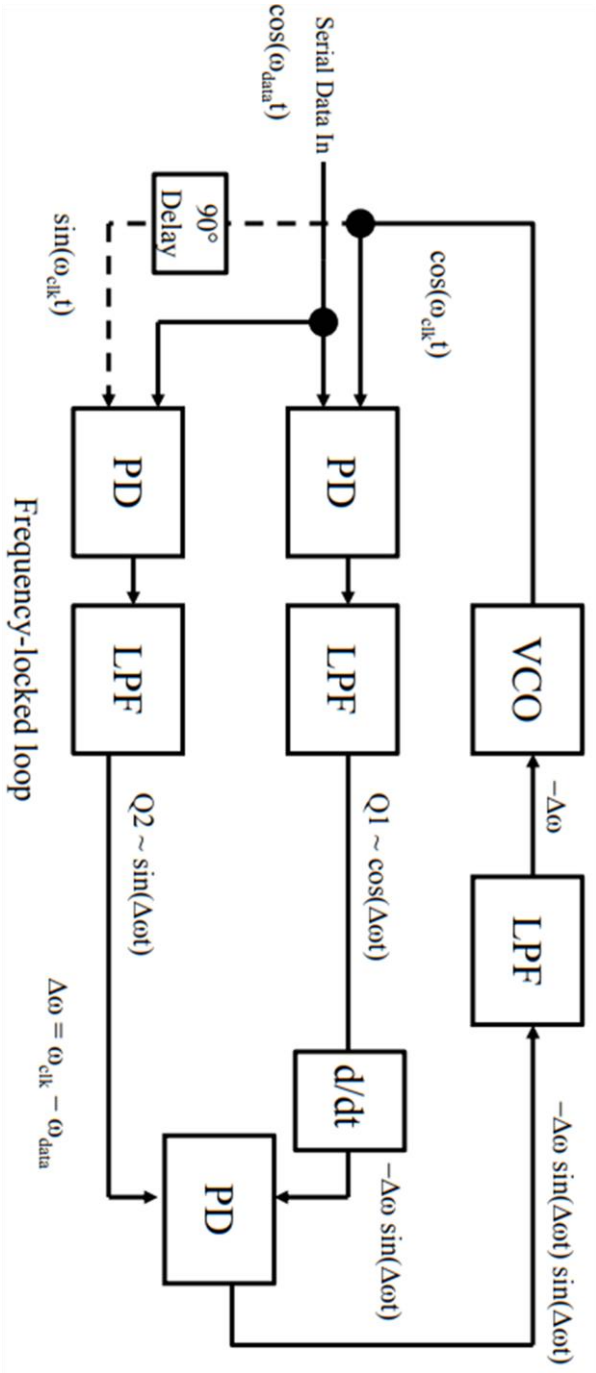


Fig. 2.28 Block diagram of the CDR using digital quadricorrelator[82]

2.4.1.2 Rotational Frequency Detector

In contrast to the quadricorrelator, the rotational FD includes no filter and differentiator [83] . Hence, it is particularly suitable for certain applications but has a more limited operating frequency range than quadricorrelator. Fig. 2.29 shows the operation principle of rotational FD. The rotational FD detects the rotating direction of the data edge in the phasor domain. When the clock frequency is higher than the data rate, the data edge in the phasor domain rotates clockwise. On the contrary, when the clock

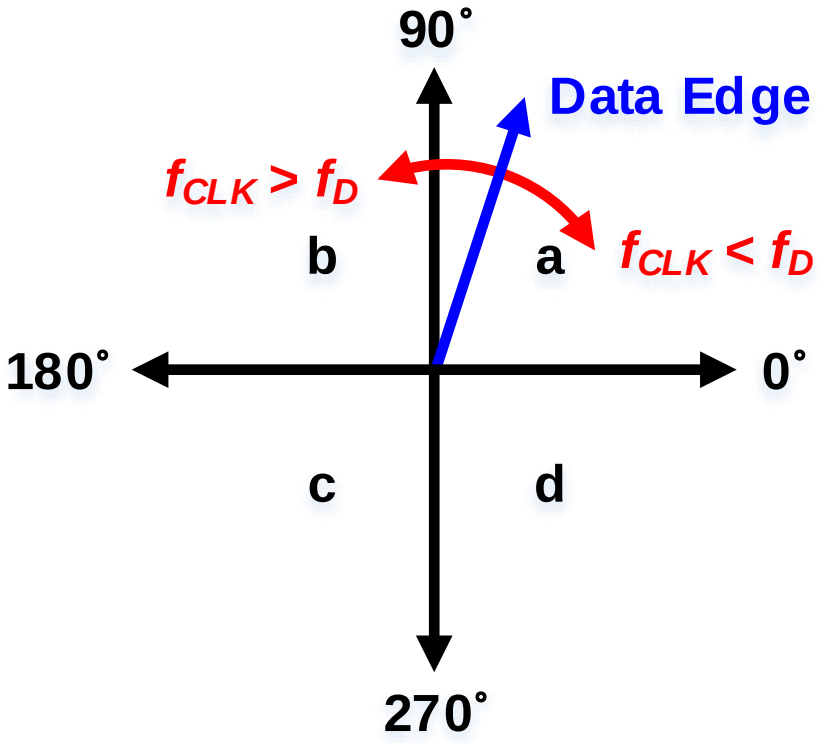


Fig. 2.29 Operation principle of rotational FD

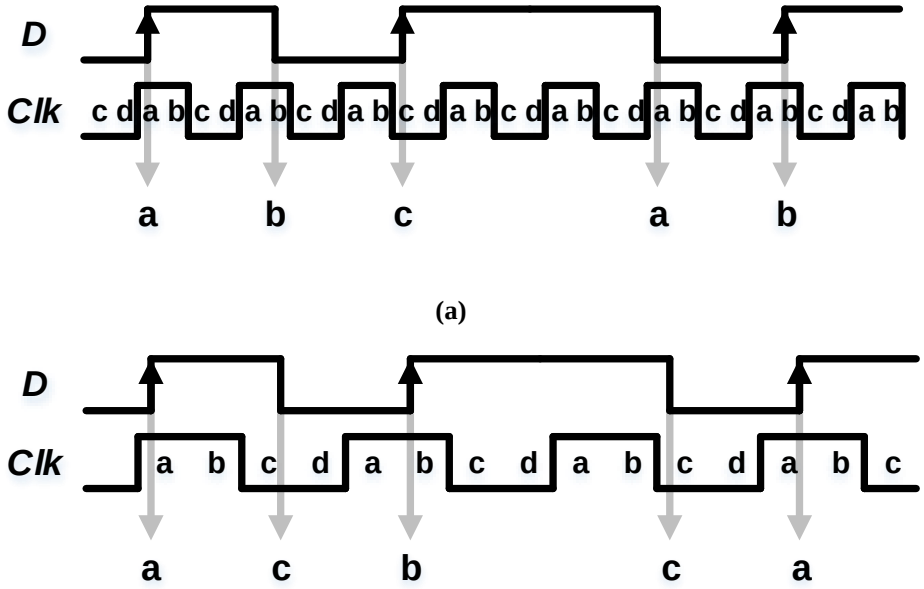


Fig. 2.30 Timing diagram of rotational FD when the clock frequency is (a) higher than the data rate and (b) lower than the data rate

frequency is lower than the data rate, the data edge in the phasor domain rotates counter-clockwise. Detailed timing diagram of the rotational FD is shown in Fig. 2.30.

2.4.2 Prior Works

The frequency acquisition scheme is the most critical task in referenceless CDR, and various schemes have been studied to achieve high energy efficiency, fast acquisition time, and a wide frequency capture range. However, in general, FD has a trade-off between frequency capture range, acquisition time, and power consumption, making the optimal design of referenceless CDR difficult. In [19] and [46] , a wide frequency capture range is achieved, but they use a pre-training sequence during the frequency tracking phase. [18] achieves a wide frequency capture range with stochastic subharmonic frequency extraction, but it shows a long frequency acquisition time due to its low operating frequency. [31] uses the number of consecutive late or early signals from bang-bang phase detector (BBPD) outputs to detect the frequency difference. This scheme has the advantage of being able to lock even if the density of input data transition is low, but it also takes a long time to reach the frequency lock and requires the lowest reset of the DCO frequency.

As discussed in Chapter 2.4.1, frequency information can be obtained through a phase difference of 90 degrees. Therefore, most of the previously proposed schemes are based on oversampling structure. In [22] , a data delay buffer generates quadrature-phase data instead of a quadrature clock. It can reduce the overhead of the quadrature-phase clock generation and distribution. Though, this method should be able to know because of analog delay buffers and is not suitable for a wide-range operation. In [20] , [21] , [25] , [48] and [80] , the high-speed quadrature clock is sampled by the input data. In this case, the frequency tracking loop becomes sensitive to the ISI of the input

data, which is a factor that has become difficult to use in the recent high-speed wireline system. In [47] , [50] , and [79] , the incoming data is sampled by quadrature clocks. [47] and [50] use BBPD to generate frequency up/dn signals. [50] achieved unlimited frequency capture range by generating additional blocking signals. [79] uses modified BBPD that uses an additional quadrature clock. With the observation of stochastic occurrence, reduced the sampling point of the additional quadrature phase to one and implemented FD with relatively simple hardware.

On the other hand, employing a quadrature clock faces several challenges in high-speed applications. Commonly, the maximum data rate and the structure of the system are determined by the maximum frequency of the oscillator. If an LC oscillator is used for a clock generator, generating multi-phase clocks requires additional hardware since the LC oscillator outputs a differential clock. In this case, the multi-phase clock generator introduces large jitter, thereby reducing the advantage of the LC oscillator. If a ring oscillator is used for a clock generator, the number of stages must be increased to generate multi-phase clocks, which leads to a decrease in the maximum operating frequency. Moreover, skew between multi-phase clocks and increased power consumption due to the increased number of samplers and clock trees are problems. In [43] and [45] , baud-rate referenceless CDR is implemented with rotational FD. With additional data samplers, it divides the data level into three sections and detects the frequency information by rotation direction of the data. However, it shows a long frequency lock time compared to oversampling-based referenceless CDR architectures. In [52] , phase and frequency detection is done by a single Alexander PD. Through stochastic observations of pattern histograms, phase and frequency detection are achieved with minimal hardware overhead.

Chapter 3

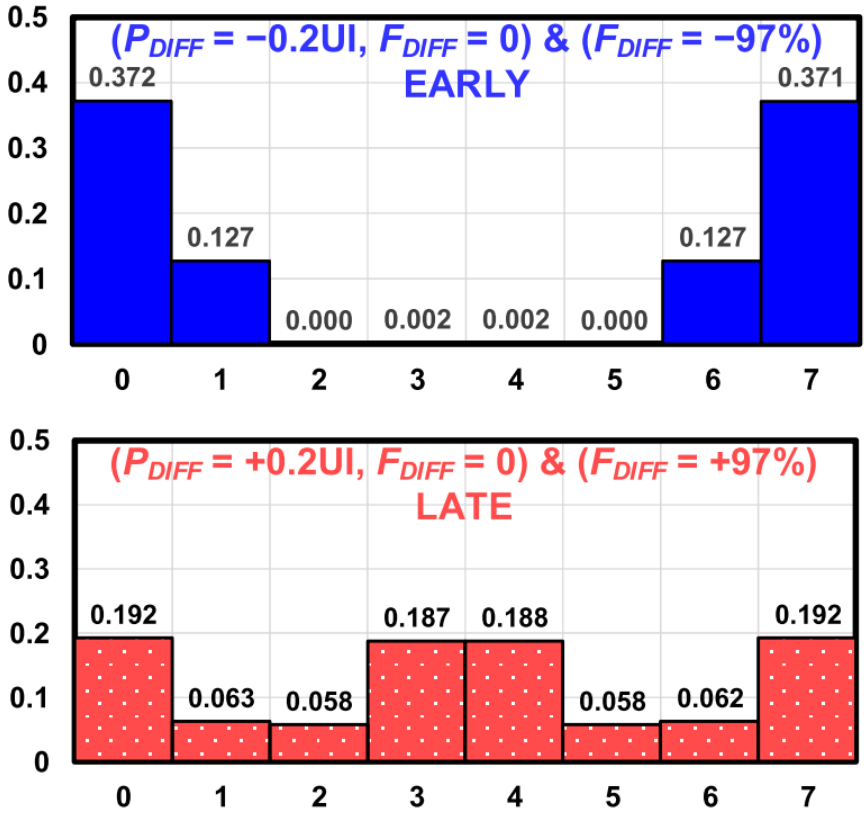
Design of the Referenceless CDR Using SFD

3.1 Overview

Referenceless CDR refers to a CDR in which a frequency lock is performed without an external reference clock. Thanks to cost savings obtained by not using external reference clocks, referenceless CDR is widely adopted in wireline communication systems using embedded clock structures such as repeaters and active cables. Since systems employing referenceless CDR do not receive external reference clocks, they must generate clocks with their own PLL, and they need the means to synchronize

their own generated clocks with input data. Accordingly, the referenceless CDR requires a scheme for extracting frequency information from input data, and the scheme dramatically influences the performance of the CDR. As previously discussed, most of the previously proposed frequency acquisition schemes have a tradeoff between capture range, frequency acquisition time, and power consumption. If high power efficiency is achieved, the frequency acquisition time is long, or the capture range is small, and if the capture range is extensive, the power consumption is largely due to increased hardware. Meanwhile, a frequency acquisition scheme based on stochastic PFD is proposed, which achieves both frequency and phase detection with a low power consumption [52] . In [52] , three consecutive sequential data-edge-data samples are monitored for the case where the clock is early and late. Among the various phase and frequency offset conditions, the phase difference of ± 0.2UI and the frequency difference of $\pm 97\%$ are selected as the final representative histogram conditions. Fig. 3.1 (a) shows the corresponding representative pattern histograms. By using these histograms, the weights are calculated by the Bayes' theorem. Fig. 3.1 (b) shows the decision and the calculated weights. As shown in Fig. 3.2, the stochastic PFD (SPFD) proposed in [52] achieves a wide range of frequency acquisition operations without interfering with PD operation. However, selecting the representative condition of the phase and frequency offset is a little ambiguous.

In this paper, we introduce a method for analyzing stochastic frequency detectors. In addition, the concept of autocovariance was introduced into SPFD to propose referenceless CDR so that there are no false locking points. As a result, we propose a design methodology for designing robust SFD-based referenceless CDR.



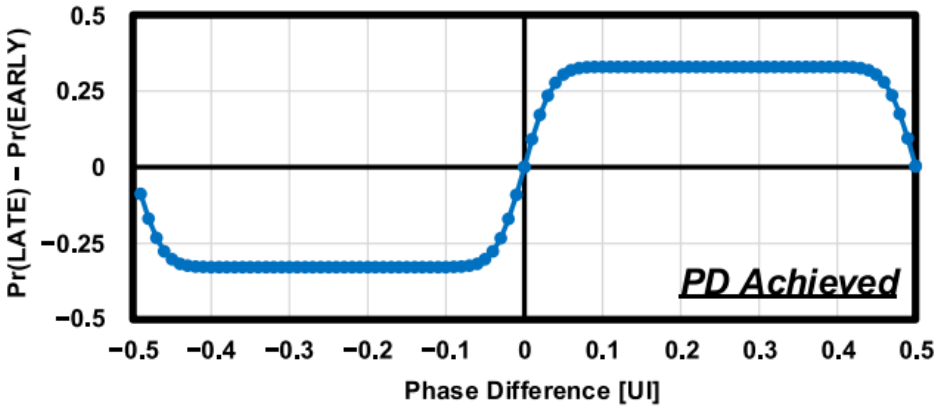
(a)

Rep. Histograms: (P_{DIFF} = ±0.2UI, F_{DIFF} = 0) & (F_{DIFF} = ±97%)								
Pattern, N	0	1	2	3	4	5	6	7
Pr(N| EARLY)	0.372	0.127	0	0.002	0.002	0	0.127	0.371
Pr(N| LATE)	0.192	0.063	0.058	0.187	0.188	0.058	0.062	0.192
Weight, W_N	−0.32	−0.34	+1	+0.98	+0.98	+1	−0.34	−0.32
Decision	E	E	L	L	L	L	E	E

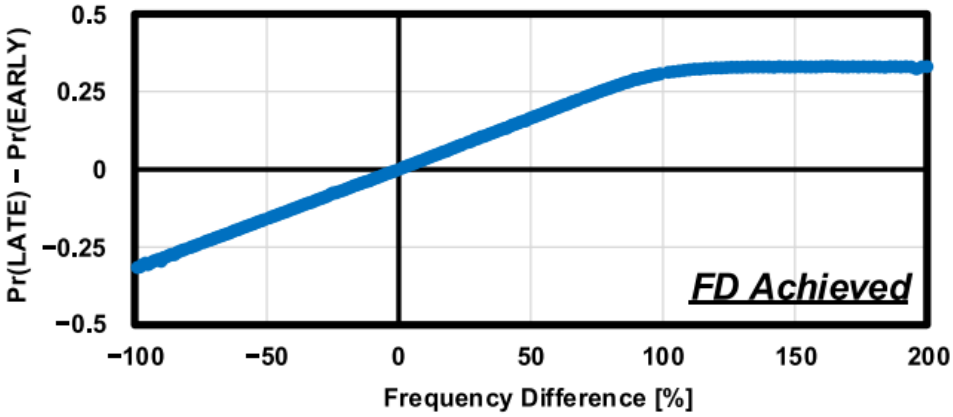
E: EARLY L: LATE

(b)

Fig. 3.1 (a) The representative pattern histograms (b) and corresponding early/late decision and weight used in [52]



(a)



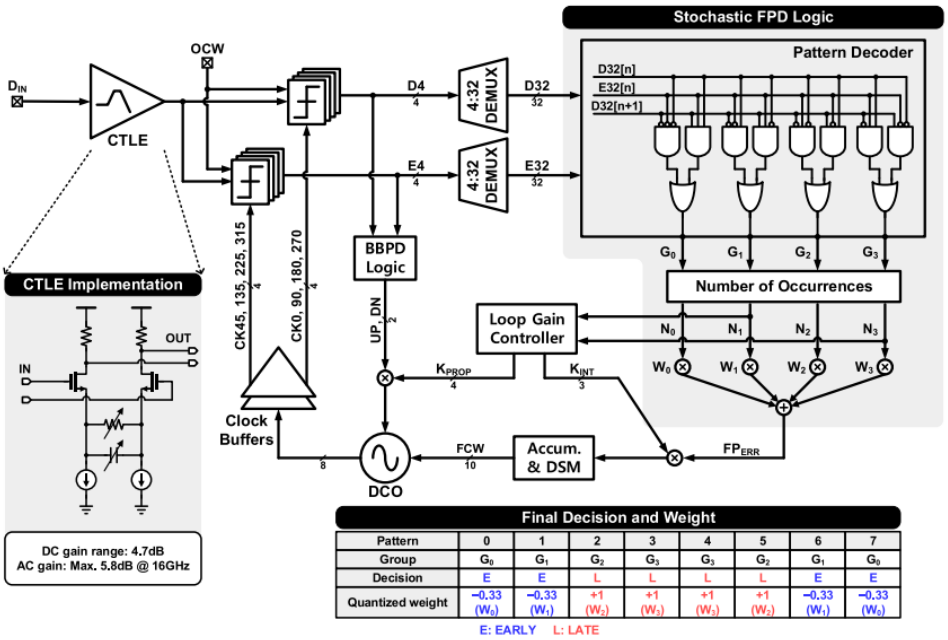
(b)

Fig. 3.2 Simulated (a) phase detection gain curve and (b) frequency detection curve of [52]

3.2 Proposed Frequency Detector

3.2.1 Motivation

Since the proposed frequency acquisition scheme is extended from the stochastic PFD in [52] , we will start by reviewing the frequency acquisition behavior of the previous SPFD. Fig. 3.3 shows the circuit implementation of [52] . 3-bit DED patterns are counted in the digital domain, and analog BBPD is adopted for the direct-proportional path of DCO. To see the detailed frequency acquisition behavior of this system,



Pattern	0	1	2	3	4	5	6	7
Group	G_0	G_1	G_2	G_3	G_3	G_2	G_1	G_0
Decision	E	E	L	L	L	L	E	E
Quantized weight	−0.33 (W_0)	−0.33 (W_1)	+1 (W_2)	+1 (W_3)	+1 (W_3)	+1 (W_2)	−0.33 (W_1)	−0.33 (W_0)

Fig. 3.3 Circuit implementation of referenceless CDR with stochastic FPD [52]

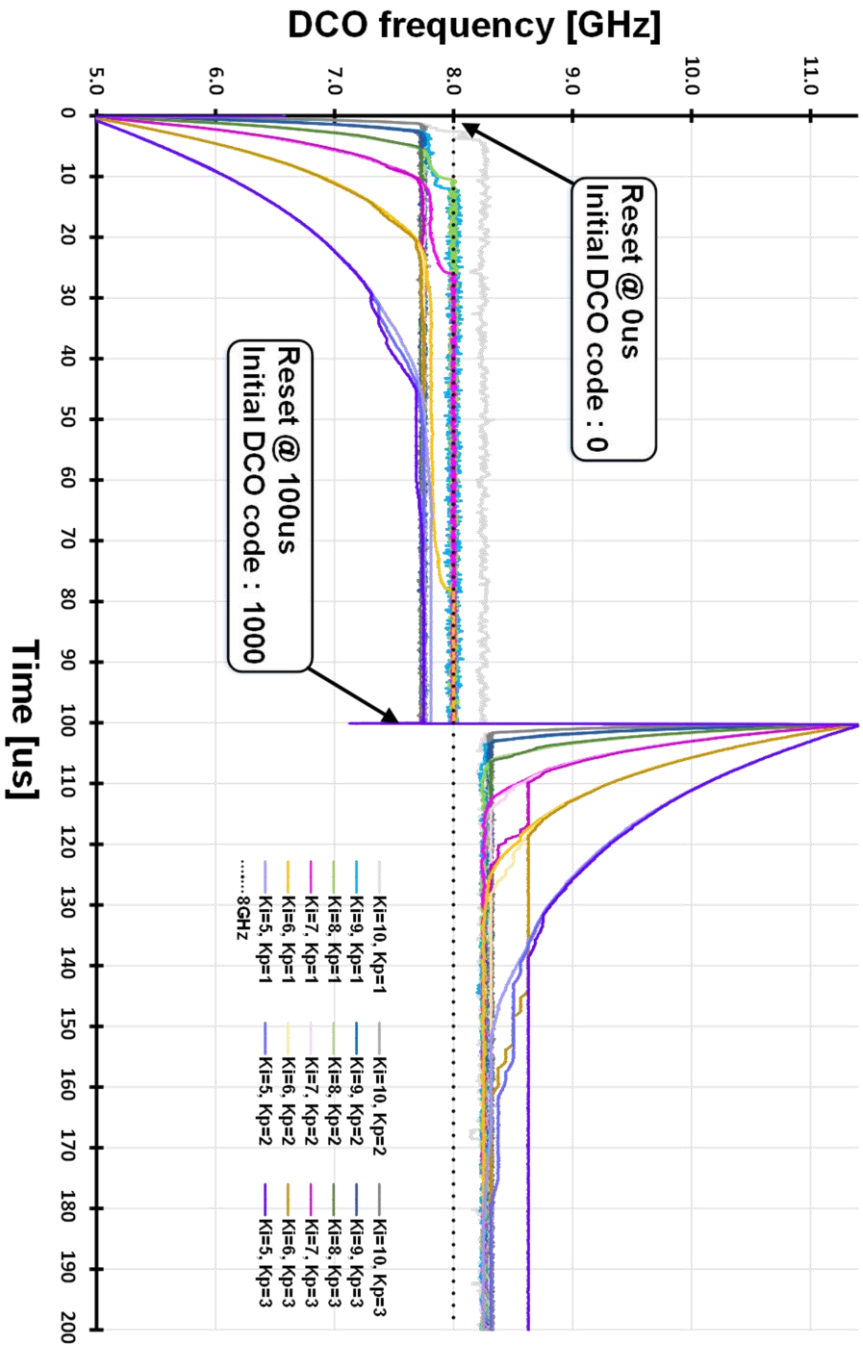


Fig. 3.4 32Gb/s 7-dB channel loss PRBS7 frequency acquisition behavior of SPFD with 3-bit pattern weight (-1, -1, 3, 3)

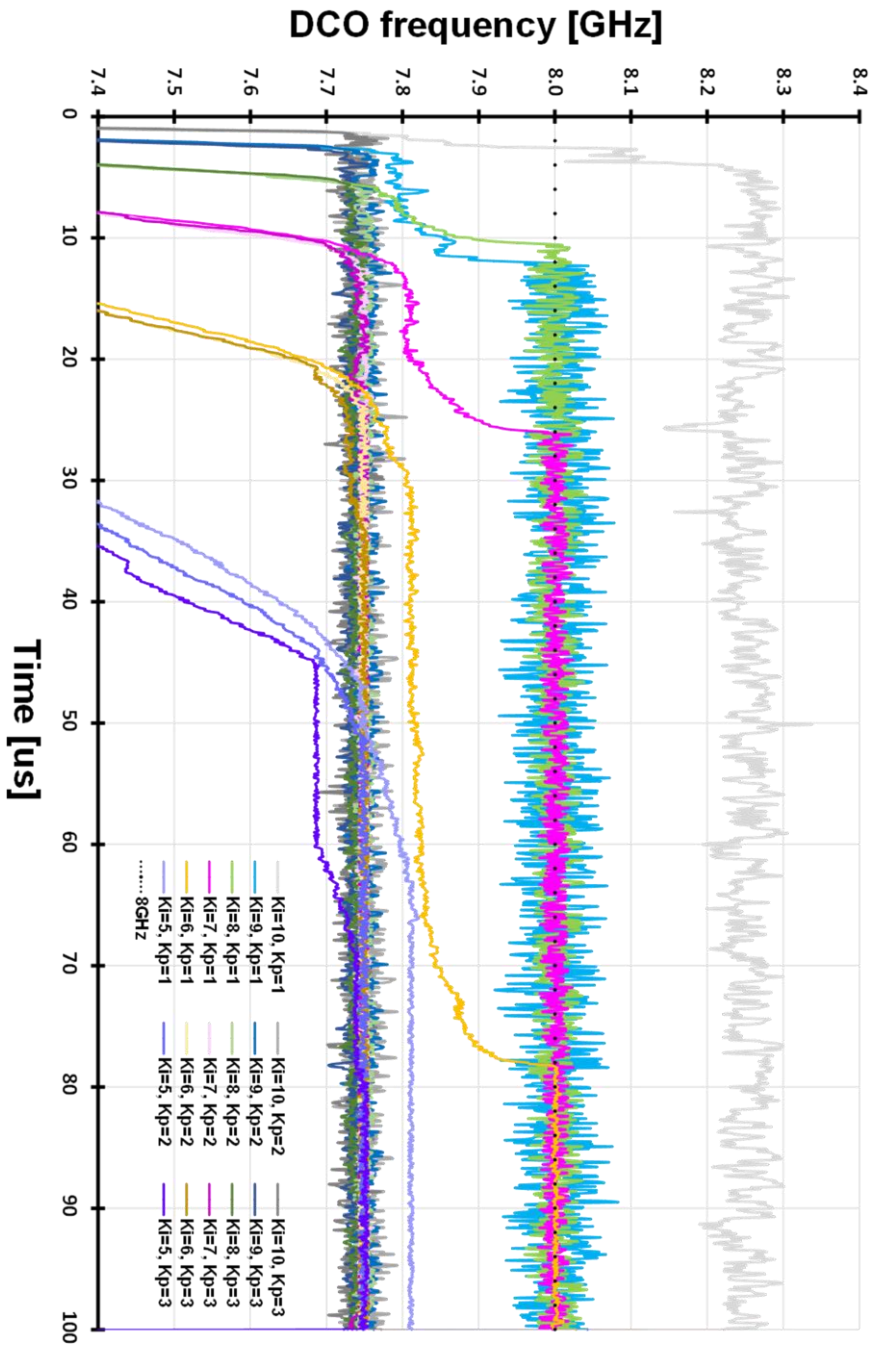


Fig. 3.5 Zoomed view (0 ~ 100μs) of Fig. 3.4

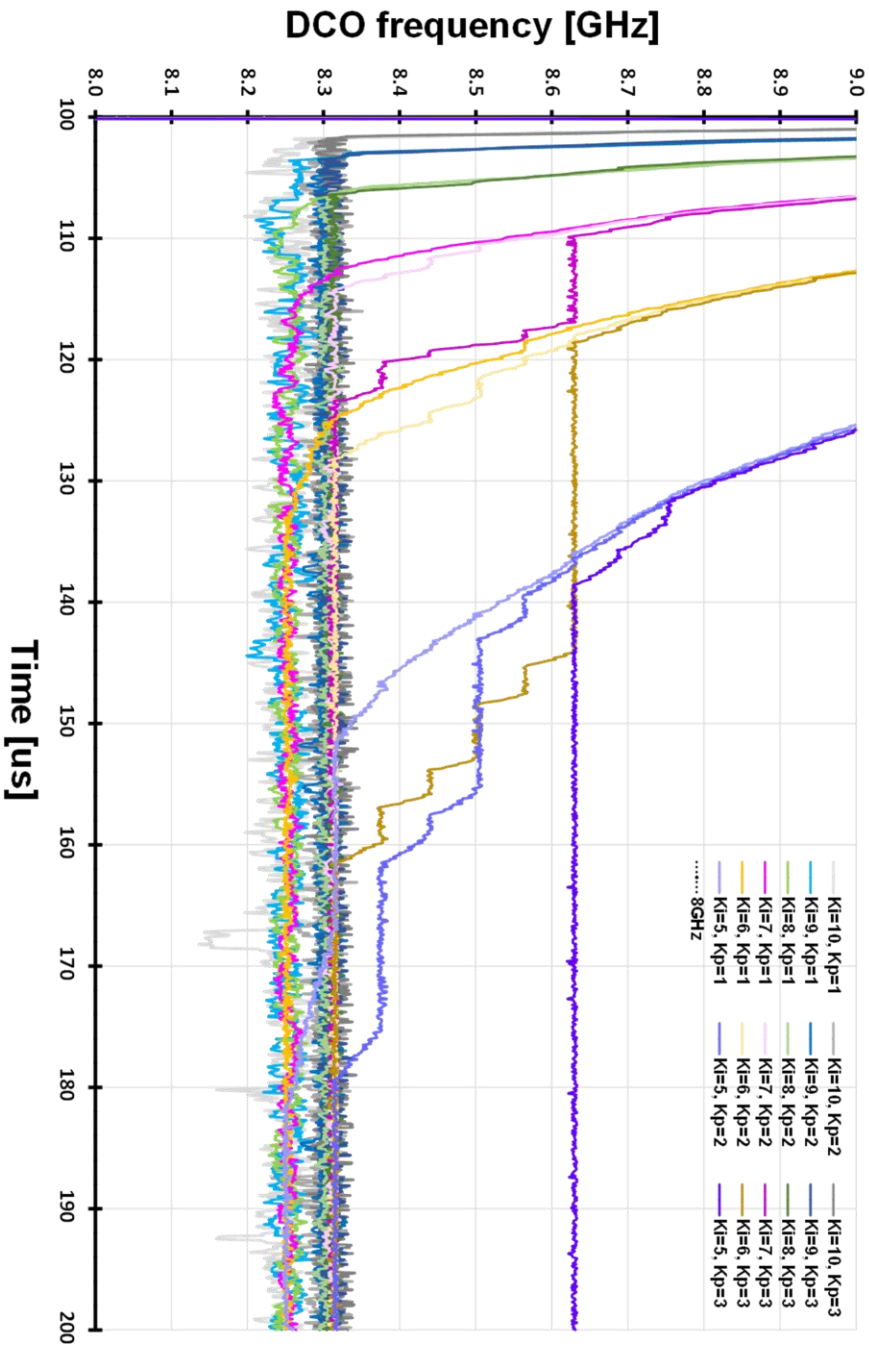


Fig. 3.6 Zoomed view (100μs ~ 200μs) of Fig. 3.4

we modeled the SPFD with system-Verilog. Fig. 3.4 shows a simulated frequency acquisition behavior of the SPFD in [52] at various loop gain conditions: direct-proportional path gain for DCO (Kp) and integral path gain for digital loof filter (Ki). Note that Kp is linear order and Ki is exponential order. 32Gb/s pseudo-random binary sequence 7 (PRBS7) traversing a 7-dB loss channel is applied as input data. The weights of the 3-bit patterns are determined as shown in Fig. 3.3: (w_{G0}, w_{G1}, w_{G2}, w_{G3})=(-1, -1, 3, 3). The phase noise of the DCO modeling is -80dBc/Hz at 1MHz offset with the center frequency of 8GHz. The simulation results show conditions in which frequency locking is not achieved. The dashed line displays an 8 GHz line. To show up-tracking and down-tracking cases, the initial DCO frequency is set to the lowest at 0μs and set to a high point at 100μs. Waveforms with the same Ki are plotted in the same color series. That is, the same color series waveforms are the result of simulation while fixing Ki and changing Kp.

Before 100μs, it can be seen that there are conditions for locking properly and conditions for false locking. Fig. 3.5 shows an enlarged view of the front part of the graph. In waveforms before 100us, only a small number of conditions succeed in achieving frequency lock at 8GHz, and those conditions are when Kp is 1. Even though Kp is 1, the conditions where the frequency lock does not occur at 8 GHz are when Ki is 5 and 10. Here we can assume that there is a set Ki that can achieve frequency lock. And the larger the integral gain Ki, the faster the acquisition time instead of the amplitude of dithering. What can be seen here is that if Kp is too large, the phase of the DCO controlled in the direct-proportional path becomes larger than the value generated by the frequency difference in the digital loop filter (DLF), and thus the frequency lock is not performed at the target frequency. Conversely, if Ki is too large, the frequency

lock is roughly performed near the target frequency, but the phase lock is not performed properly, and the DCO frequency is locked elsewhere.

On the other hand, no frequency lock is achieved under any conditions after 100μs. Instead, the frequency is stuck at a point slightly higher than 8GHz under all simulation conditions. Fig. 3.6 shows an enlarged view of the latter part of the graph. Under the condition of Ki=5 and Kp=3, the DCO frequency stucks at 8.63GHz, and the remaining conditions are stuck between 8.2GHz and 8.3GHz. In order to analyze this result, the frequency detection curve should be examined. Since the gain near zero was very small in the FD gain graph Fig. 3.2 (b), we need to zoom in on the FD gain curve near the zero frequency difference.

3.2.2 Pattern Histogram Analysis

The method of obtaining FD gain at each point is to fix the DCO frequency and average the updated value of DLF for a long time. Each point matches the code of the DCO. On the other hand, FD gain is calculated by adding the values obtained by multiplying the frequency of occurrence of 3-bit patterns by weight, so the frequency of occurrence of individual patterns is also plotted and analyzed.

Fig. 3.7 shows the open-loop pattern histogram of the each sequential data-edge-data patterns near the frequency difference of 0. *dn0* equals patterns 000 and 111, *dn1*

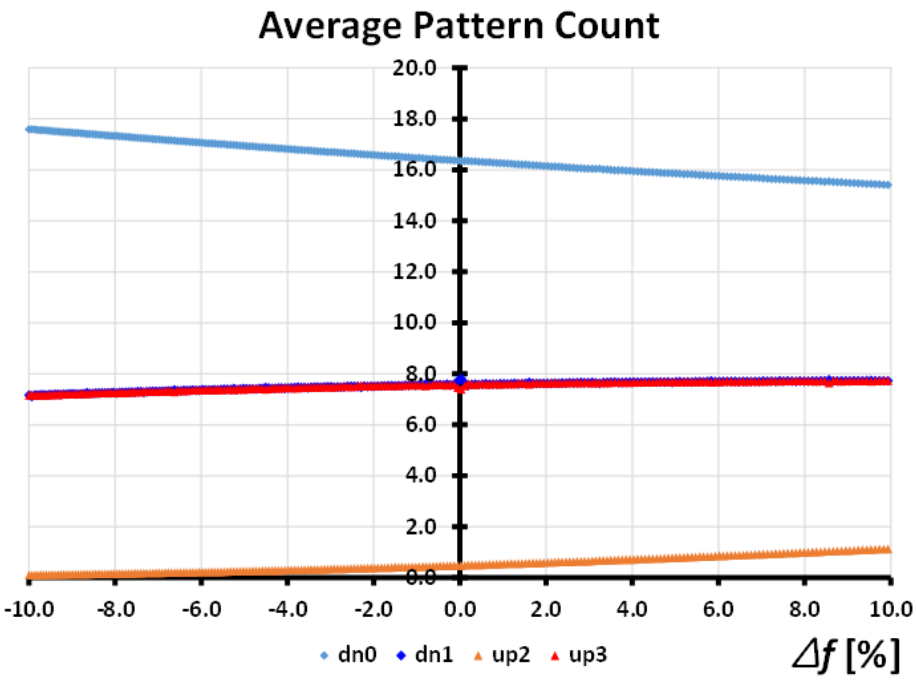


Fig. 3.7 Open-loop pattern histogram

equals patterns 001 and 110, *up2* equals patterns 010 and 101, and *up3* equals patterns 011 and 100. Δf is the normalized frequency difference between the DCO and input data ($\Delta f = (f_{data} - f_{DCO})/f_{DCO}$). As shown in FIG. 3.7, as the DCO frequency increases, the non-transition pattern *dn0* (000, 111) tends to increase, while the pattern *up2* (010, 101) in which two transitions occur tends to decrease. *dn1* and *up3* do not show a significant difference according to the change in DCO frequency.

Fig. 3.8 shows the simulated FD gain of the SPFD in [52] . The corresponding graph was obtained by multiplying the patterns in Figure 3.7 by -1, -1, -3, and 3. The overall shape of the graph is a straight line, but the spike pops down near zero.

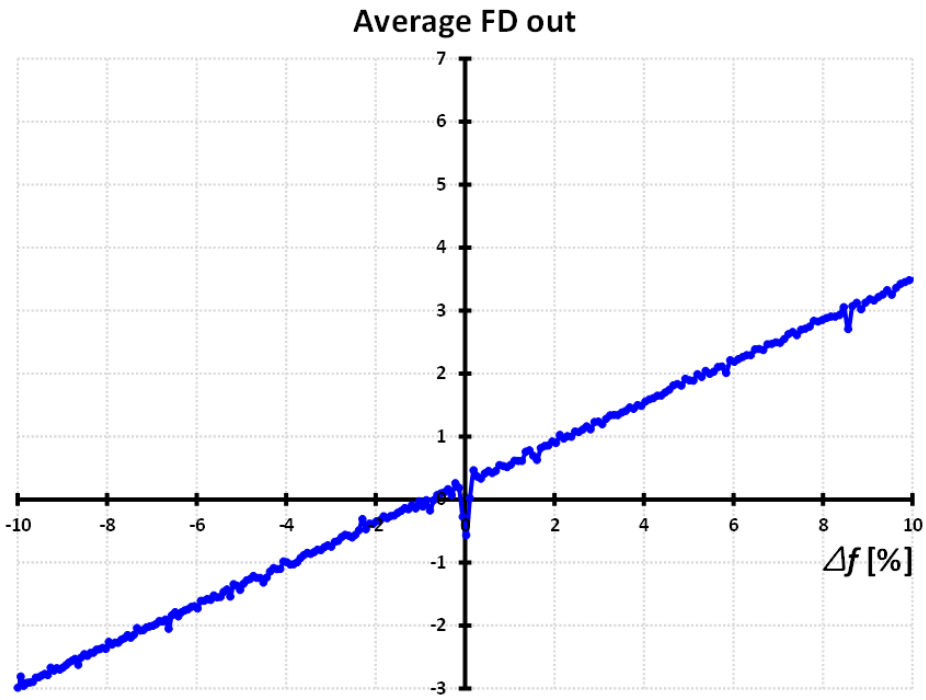


Fig. 3.8 FD gain curve with 3-bit pattern weights (-1, -1, 3, 3)

According to this FD gain curve, the frequency lock is achieved at the frequency difference of around 0% or -1%, but this is not the case. This means that there are incorrect assumptions in obtaining the graphs in Fig. 3.7 and Fig. 3.8.

The pattern histogram in Fig. 3.7 plots the values counted by the DLF in an open-loop state with the feedback path to the DCO disconnected. Under this condition, the direct-proportional path is also disconnected. In conventional digital PLLs or CDRs, direct proportional paths are employed to reduce the latency of updating DCO codes to reduce jitter [84] or increase loop stability [85] . On the other hand, BBPD-based PLL has a pull-in range which is the frequency range that the PLL can lock after some cycle-slipping [81] . In other words, BBPD has a frequency tracking capability at a small frequency difference. Fig. 3.9 shows the simulated FD gain curve of the BBPD in the direct-proportional path at various proportional gains. As the proportional gain increases, the peak of the FD gain moves outward, which means that the frequency capture range of the BBPD increases. It is also shown in Fig. 3.10. Based on the information so far, it can be seen that the waveforms in Fig. 3.5 achieved frequency locking due to the influence of BBPD in the direct-proportional path. Therefore, in order to accurately analyze the FD gain of SPFD, it is necessary to look at the results, including the influence of the direct-proportional path.

Fig. 3.11 shows the revised open-loop pattern histogram of each sequential data-edge-data pattern near the frequency difference of 0. The overall aspect of the 3-bit patterns has changed a lot. *dn1* and *up3* show a large gain near the frequency difference of 0, and their respective polarities are opposite to each other. Near the frequency difference 0, *dn0* shows a slightly reduced value, and *up2* shows a slightly increased value. The effect of direct-proportional path pulling and pushing phase

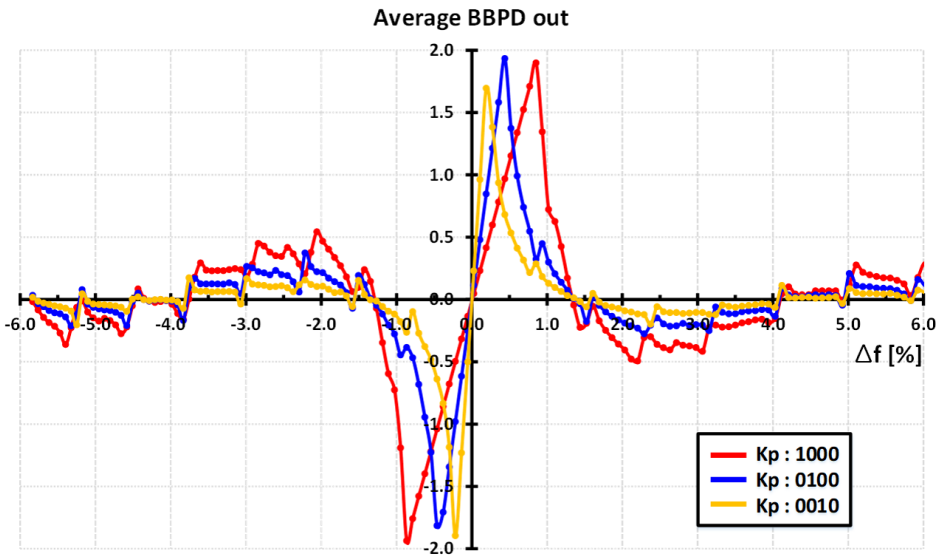


Fig. 3.9 BBPD FD gain in the direct-proportional path

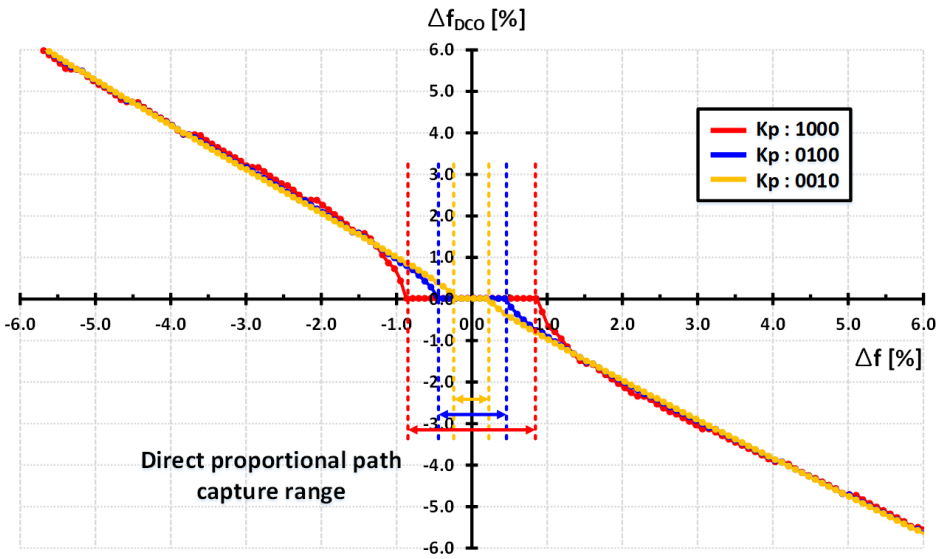


Fig. 3.10 Direct-proportional path frequency capture range

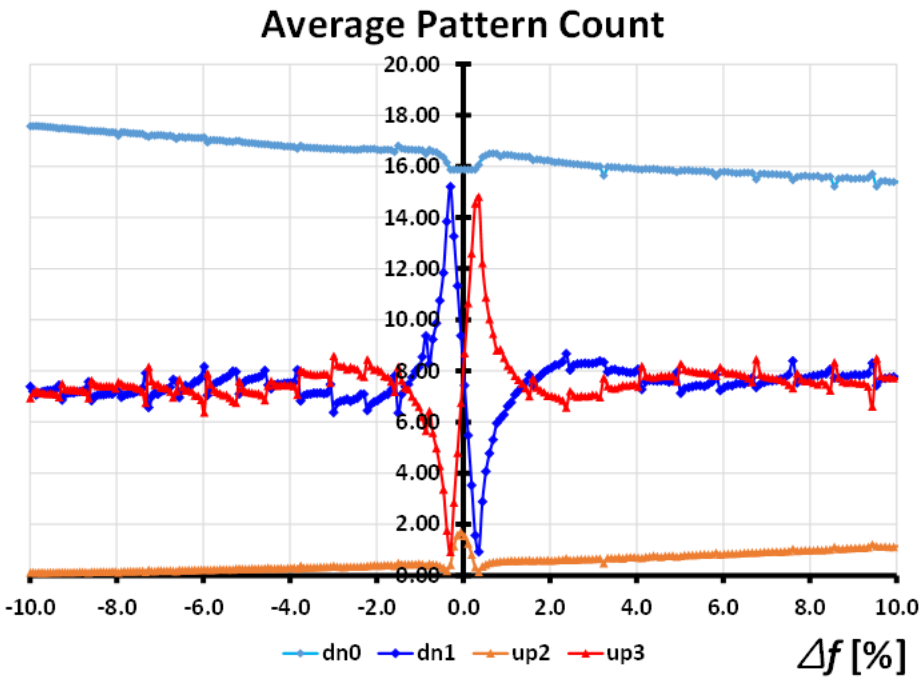


Fig. 3.11 Open-loop pattern histogram including direct-proportional path

every moment is no longer zero near frequency difference 0%. Fig. 3.12 shows the modified FD gain curve, including the effect of the direct proportional path. Due to the influence of BBPD, there is a change in the zero-crossing point. 4 zero-crossing points are added near zero. Among them, the frequency difference of -3.6%, which coresponds to 8.3GHz, is the same as the point where most waveforms lock after 100μs. And there's a point that goes down slightly below zero around +3%, where the clock frequency is 7.74 GHz; it corresponds to the part where the false lock occurs

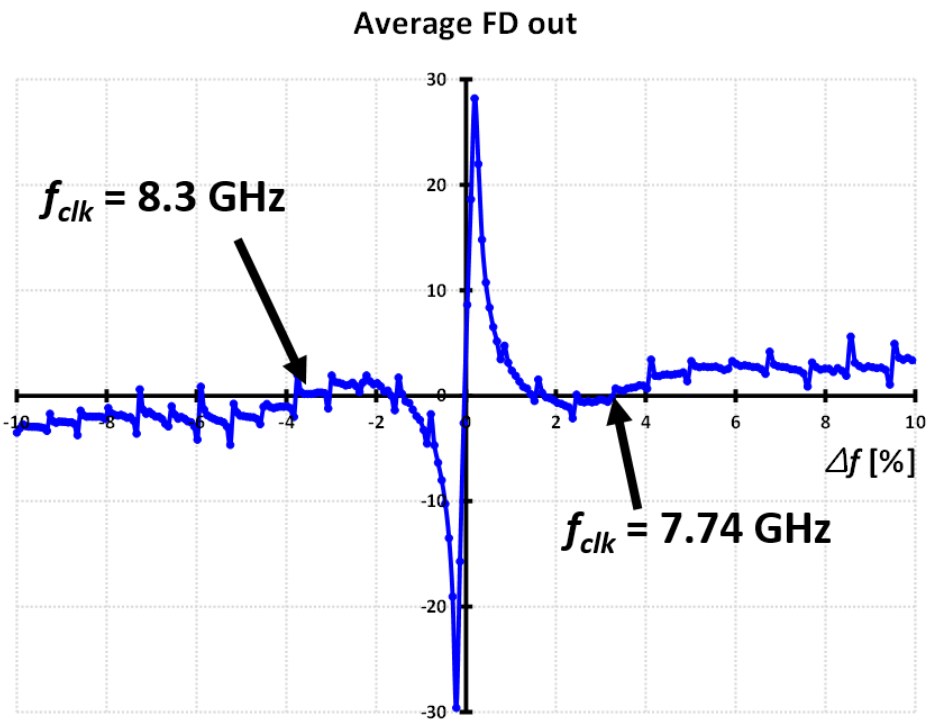


Fig. 3.12 FD gain curve with 3-bit pattern weights (-1, -1, 3, 3) including direct-proportional path

below 8 GHz when Kp is large in Fig. 3.5. If the integral gain is large, the code value that changes in the digital loop filter near 0% will be out of the capture range of the analog BBPD, which will cause locking near -4%. The graph in Fig. 3.12 can illustrate all of the frequency acquisition behaviors discussed earlier, so it can be seen that this graph accurately represents the gain of the SPFD near zero.

For SPFD to operate reliably without false lock, only one zero-crossing point must

exist in the FD gain curve at 0%. However, the linear combination of the patterns *dn0*, *dn1*, *up2*, and *up3* could not produce the corresponding graph. There are two conditions that must be satisfied when configuring stochastic FD with a linear combination of 3-bit pattern histograms. The first is that the weighted sum of histograms must be zero near the frequency difference of 0%, and the second is that there is only one zero-crossing point in the whole graph. The weight (-1, -1, 3, 3) presented above almost satisfies the first condition but does not satisfy the second condition. In order to satisfy the second condition, the slope of the overall FD curve must be increased. However, in increasing the slope, the first condition is not satisfied. Among the four patterns, the pattern with the largest slope according to frequency difference is *dn0*, followed by *up2*. Increasing the weight of *dn0* to increase the slope of the FD curve breaks the first condition, that the sum of histograms should be zero near 0% because dn0 has the highest average among the patterns. Increasing the weight of *up2* to increase the slope of the FD curve also breaks the first condition. *up2* shows a high frequency of occurrence near 0%, and the problem is that this is the maximum value in the vicinity. For these reasons, it is not possible to obtain an FD curve satisfying both of the above-described conditions by the linear combination of the sequential 3-bit patterns.

3.2.3 Introduction of Autocovariance to Stochastic Frequency Detector

To overcome the preceding limitations, we have considered the autocovariance which is used to analyze time-series data in statistics. Time-series data are not independent over time, so the current state is very closely related to the past and future states. The autocovariance function is used to indicate the degree of correlation of time-series data over time. With the usual notation E for the expectation operator, if the stochastic process $\{X_t\}$ has the mean function $\mu_t = \mathrm{E}[X_t]$, then the autocovariance is given by

$$\begin{aligned}\gamma(h) &= \mathrm{cov}(X_t, X_{t+h}) \\ &= \mathrm{E}[(X_t - \mu_t)(X_{t+h} - \mu_{t+h})]\end{aligned} \tag{3.1}$$

where h is the lag time, or the amount of time by which the signal has been shifted. If $\{X_t\}$ is a weak-sense stationary or wide-sense stationary (WSS) process whose mean function and correlation function do not change by shifts in time, equation (3.1) can be modified as follows.

$$\begin{aligned}\gamma(h) &= \mathrm{E}[(X_t - \mu_t)(X_{t+h} - \mu_{t+h})] \\ &= \mathrm{E}[X_t X_{t+h} - \mu_t X_{t+h} - \mu_{t+h} X_t + \mu_t \mu_{t+h}] \\ &= \mathrm{E}[X_t X_{t+h}] - \mu_t \mathrm{E}[X_{t+h}] - \mu_{t+h} \mathrm{E}[X_t] + \mu_t \mu_{t+h} \\ &= \mathrm{E}[X_t X_{t+h}] - \mu^2\end{aligned} \tag{3.2}$$

This concept can be similarly applied to the FD of the CDR. If the current state is a situation in which the DCO frequency is higher than the data rate, there is a high

probability that the state will be maintained even after a short time and vice versa. Thus, the introduction of the autocovariance concept into SFD will result in patterns that are more sensitive to current frequency differences. Using this, it will be possible to design an FD that satisfies both of the above-mentioned conditions. In this paper, we propose the stochastic frequency detector using autocovariance.

The lag time *h* of the proposed SFD is determined to be one digital clock cycle. When processing data in a digital domain, the incoming 32-bit deserialized data is processed as a unit. Since it takes several clock cycles for the current incoming data to be reflected in the DCO via digital logic, the frequency bias will remain somewhat the same for the current and the subsequent clock cycle data. By calculating the autocovariance of current digital data and then clock cycle data, we can extract data that is more sensitive to frequency information, and the corresponding data is expected to have a steeper slope on the FD curve. Fig. 3.13 shows a conceptual diagram for calculating the autocovariance of the 3-bit pattern in the digital domain. The data and edges are deserialized to 32-bits and sent to the digital domain. To classify consecutive D-E-D patterns, D[n] is shifted 1 bit to the left, and bitwise operations are performed with D[n] and E[n]. The autocovariance operation is performed on this classified pattern, with the corresponding pattern before one clock cycle.

If the input data is random and the DCO clock is also fixed with a specific code, statistical properties such as mean and correlation will not change over time. In this case, it can be assumed that the statistical properties of patterns sampled for a short period of time at a particular DCO frequency will not change over time, and each sequential three-bit pattern can be considered a WSS process. Therefore, equation (3.2) can be used to calculate autocovariance.

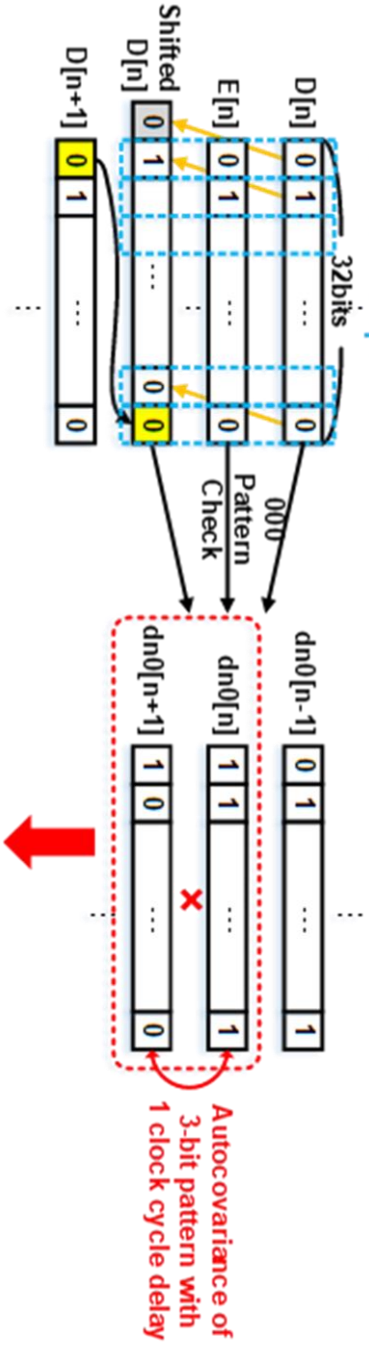


Fig. 3.13 Conceptual diagram of calculating autocovariance of the 3-bit pattern : *dn0* case

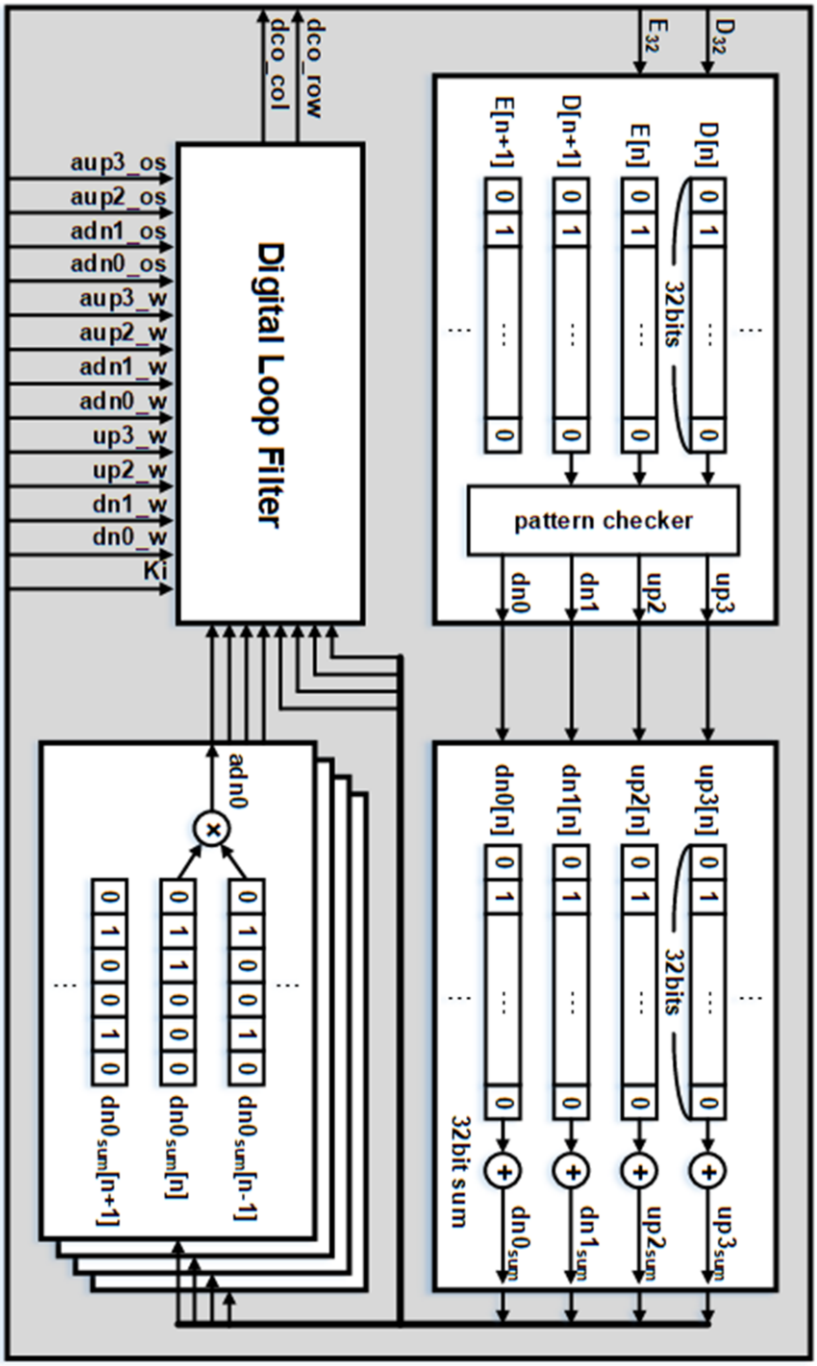


Fig. 3.14 Block diagram of the SFD logic

Fig 3.14 illustrates the detailed implementation of SFD logic. First, a 3-bit pattern classification of 32-bit data from the deserializer is performed. Thereafter, the pattern occurrence frequency is calculated by adding each digit of the classified pattern. The sum obtained is sent to the DLF and simultaneously sent to the autocovariance calculation block, multiplied by the sum prior to one clock cycle. The product obtained in this way is subtracted from the average value in the DLF. The average values and weight values of the patterns are input from the outside. Fig. 3.15 shows histograms of the autocovariance for each pattern. *adn0* is autocovariance of *dn0*, *adn1* is autocovariance of *dn1*, *aup1* is autocovariance of *up1*, *aup2* is autocovariance of *up2*.

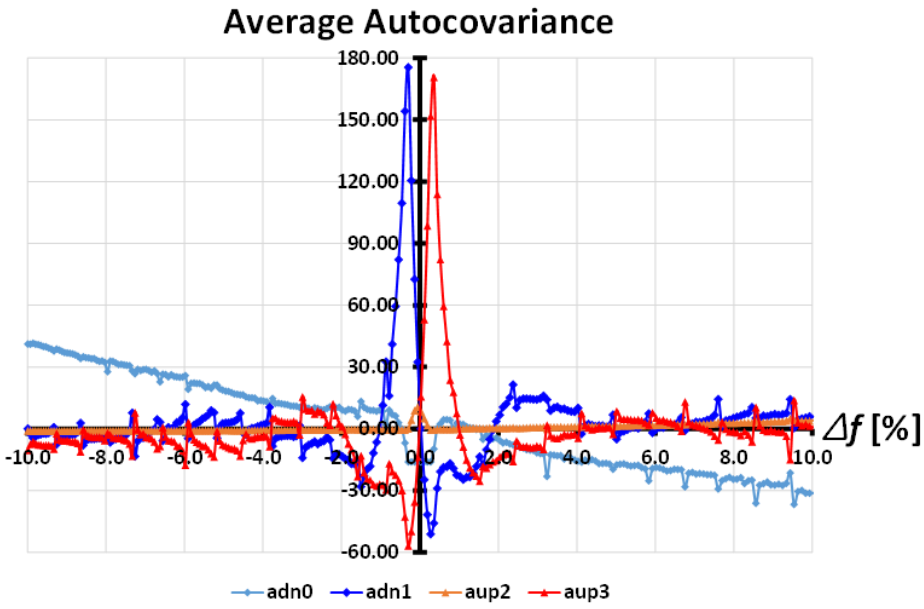
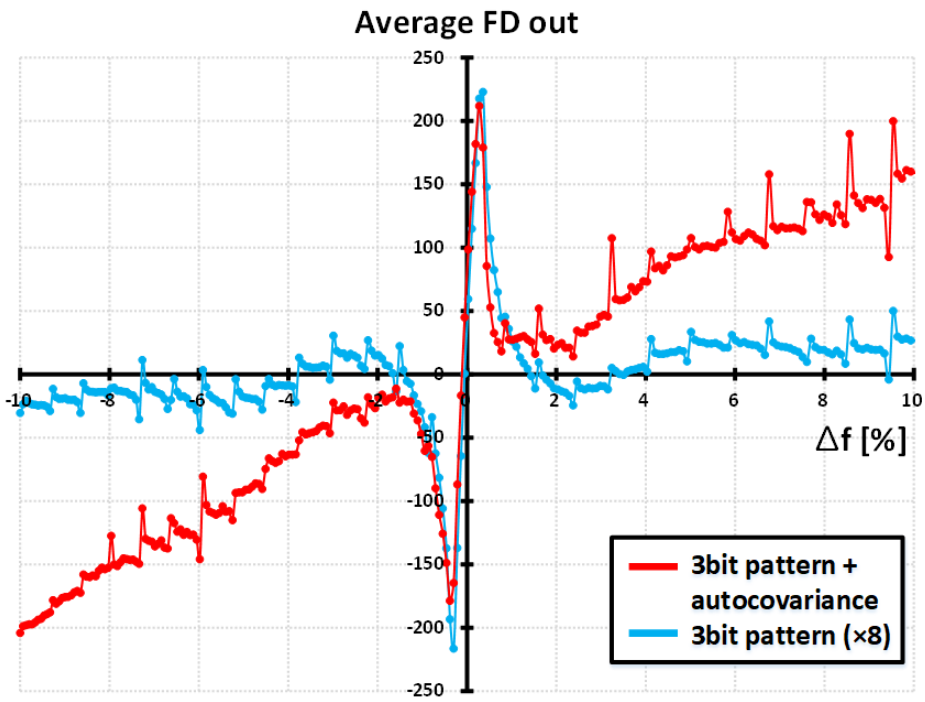


Fig. 3.15 Pattern histograms of the autocovariance for 3-bit patterns



(a)

Pattern	dn0	dn1	up2	up3
Weight	-1	-4	+1	+7
A.-Corr.	adn0	adn1	aup2	aup3
Weight	-5	-1	0	0

(b)

Fig. 3.16 (a) The FD curve and (b) weights of the proposed SFD

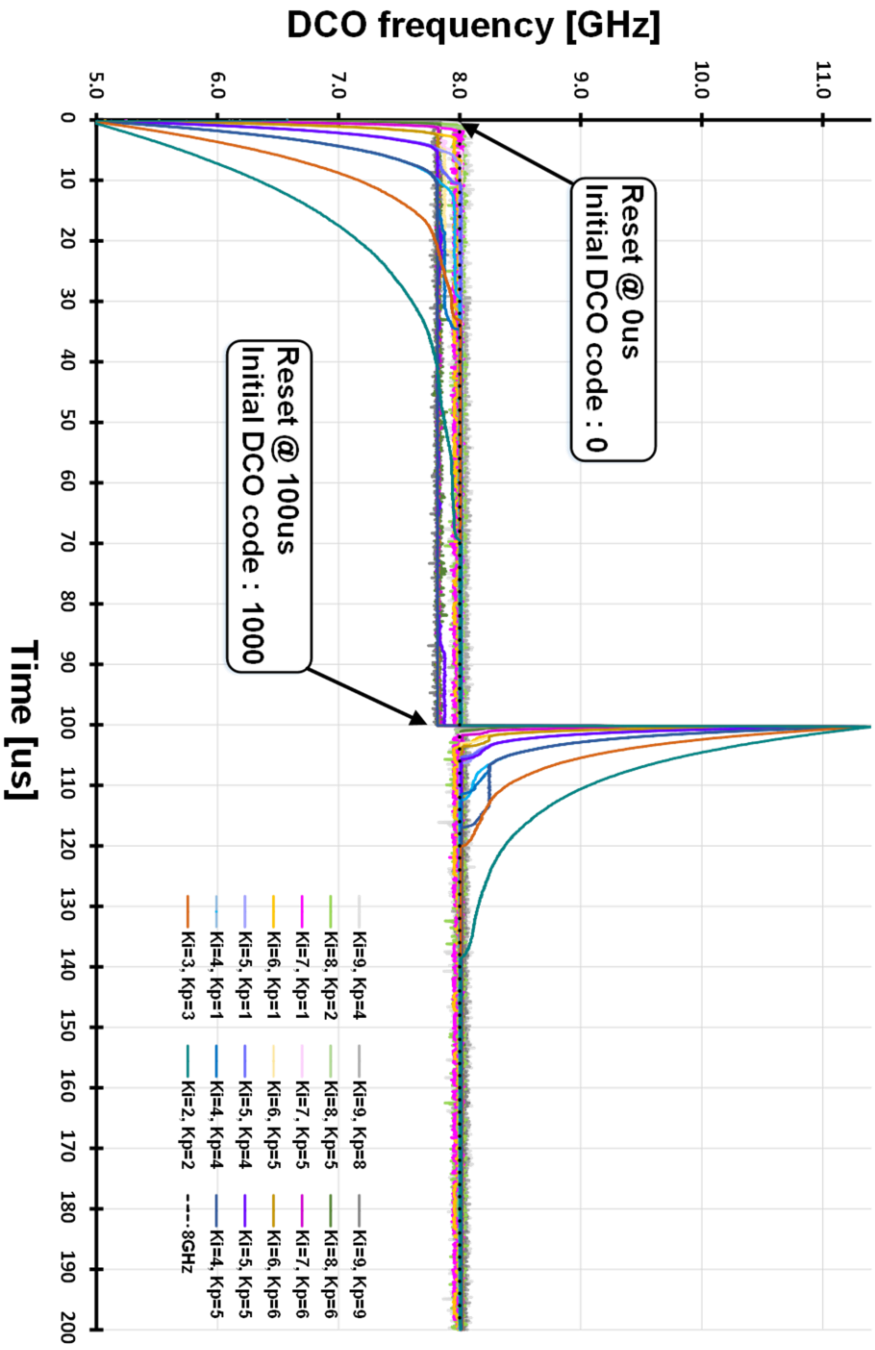


Fig. 3.17 32Gb/s 7-dB channel loss PRBS7 frequency acquisition behavior of the proposed SFD

The *aup2* pattern is originally a pattern that rarely occurred, so frequency gain is buried, and *adn1* and *aup3* show a large gain only near zero, and the slope of *adn0* is increased. Fig. 3.16 (a) shows a comparison of the FD curves of the proposed SFD and the previous SPFD in [52] . The FD gain of 3-bit pattern-based SPFD is increased by 8 times and plotted so that the peak values are similar to each other. The weight values used in the proposed SFD are shown in 3.16(b). Fig. 3.17 illustrates a simulated frequency acquisition behavior of the proposed SFD at various loop gain conditions. In most waveforms, frequency lock is achieved at 8 GHz. The graph shows the boundaries of Kp where frequency acquisition is achieved successfully for each Ki. Detailed waveforms of the proposed SFD are in Appendix A.

3.3 Circuit Implementation

3.3.1 Implementation of the Proposed Referenceless CDR

Fig. 3.17 illustrates the system block diagram of the proposed referenceless CDR. The proposed CDR is designed to operate up to 32Gb/s. Considering the operation range of the ring DCO, quarter-rate clocking architecture is adopted. The proposed CDR consists of continuous-time linear equalizer (CTLE), ring DCO, eight samplers

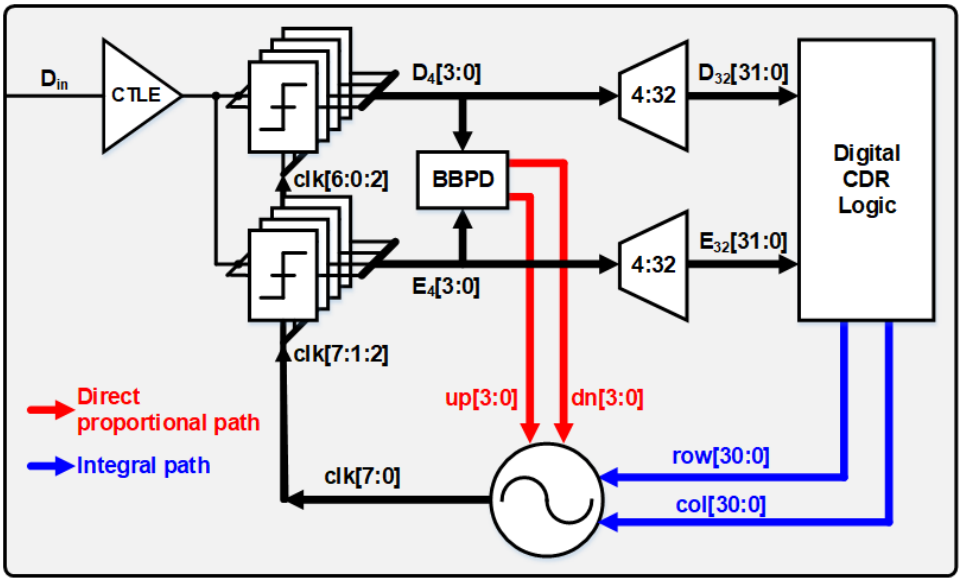


Fig. 3.18 System block diagram of the proposed CDR architecture

for quarter-rate data and edge sampling, analog BBPD, two 4:32 deserializers, and a digital block with an embedded stochastic frequency detector and loop filter. The ring DCO generates an 8-phase clock, four of which are sent to the data sampler and the other four to the edge sampler. A StrongArm latch is used as the sampler structure. The outputs of the samplers are deserialized at 4:32 deserializers and then sent to digital CDR logic. The analog BBPD generates four up/dn signals, and the signals are fed back to the DCO for a direct-proportional path. A digital loop filter generates 10-bit frequency control words (FCW) for an integral path. FCW is decoded into thermometer code to prevent the DCO frequency from jumping due to glitches. The DCO frequency is controlled by both integral and direct-proportional paths. The phase acquisition of the proposed CDR is entirely dependent on the direct-proportional path. Digital blocks are only involved in frequency acquisition. Therefore, the architecture of the proposed CDR can be viewed as a dual loop CDR with analog phase tracking loops and digital frequency tracking loops. Since the latency in the digital loop is much longer than that of the analog loop, the bandwidth of the analog loop is higher than that of the digital loop, which allows the CDR to reliably achieve phase-lock in an appropriate gain setting.

3.3.2 Continuous-Time Linear Equalizer (CTLE)

Fig. 3.19 depicts the circuit diagram of CTLE. CTLE used in the system is designed to boost up to 25 dB using the Cherry-Hopper structure. RC bank is designed to allow DC gain and pole position adjustment to be 3 bits, respectively. The frequency response of the CTLE is illustrated in Fig. 3.20, and The setting used for the measurement is highlighted in light green.

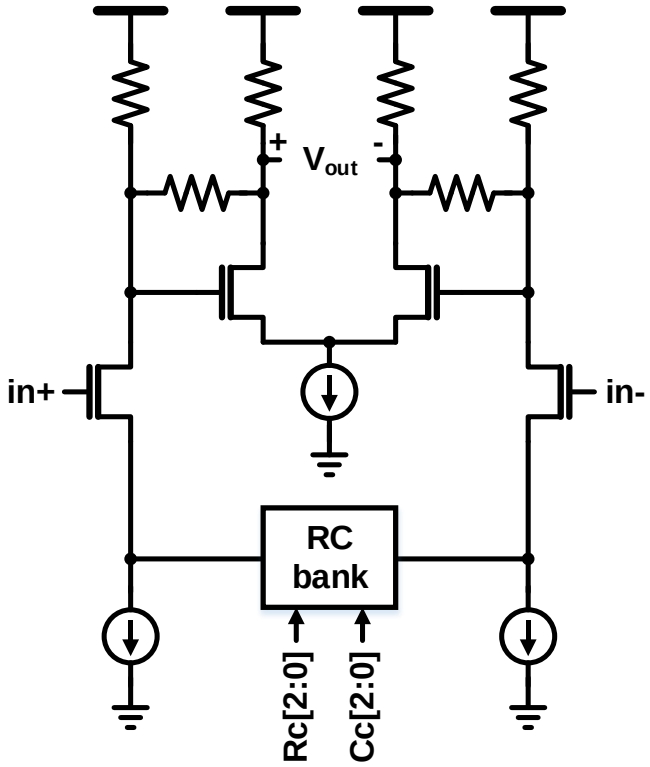


Fig. 3.19 Circuit diagram of the CTLE

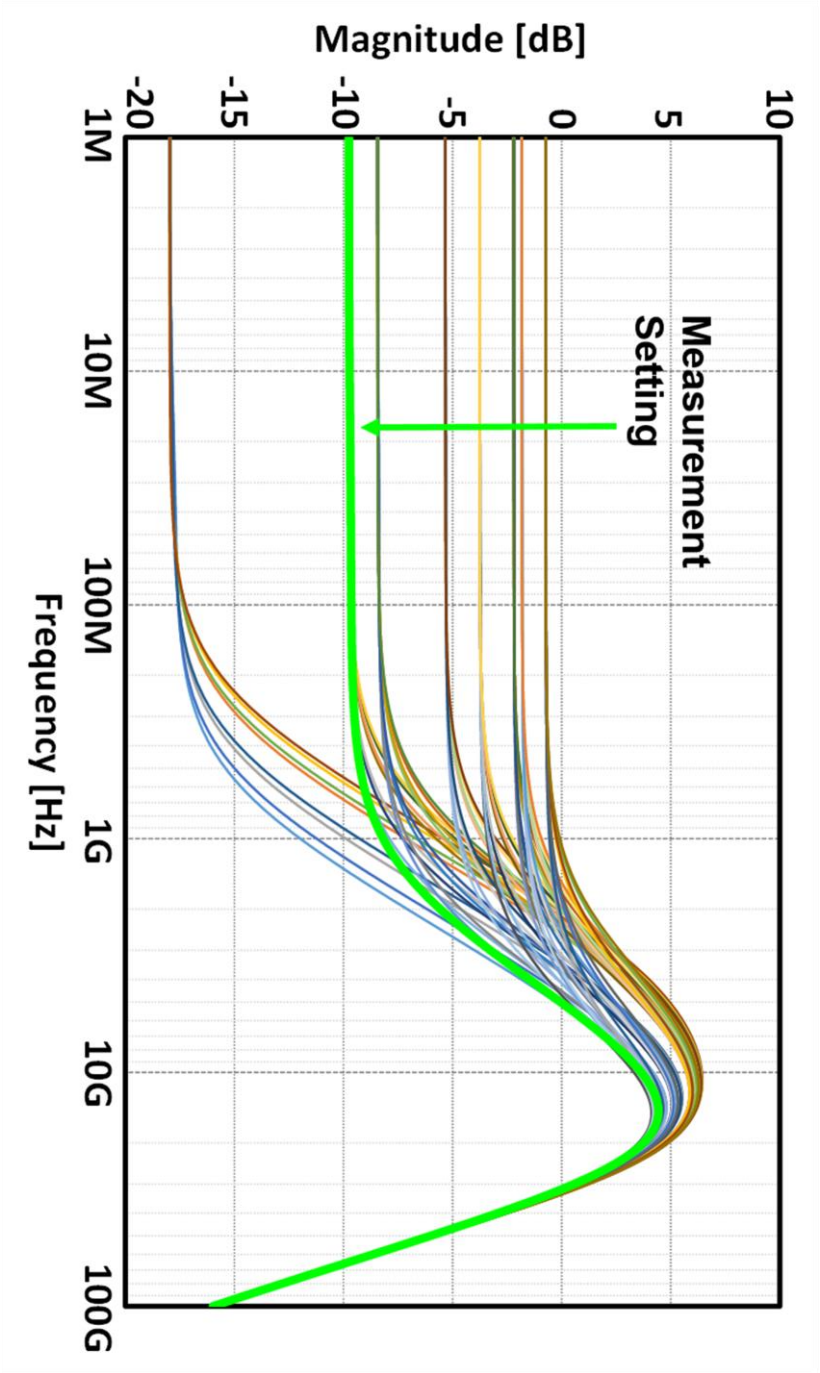


Fig. 3.20 Post-layout simulated frequency response of CTLE

3.3.3 Digitally-Controlled Oscillator (DCO)

Fig. 3.21 shows the block diagram of the 8-phase DCO. The DCO is composed of a digitally-controlled resistor (DCR), 4-stage pseudo-differential inverters, varactor loads with direct-proportional logic, and level shifters. A code controls the DCR from 0 to 1023 generated by the thermometer code Row/Col. The frequency adjustment method of the DCO is to adjust the delay of the four-stage inverter using the voltage VDD_DCO lowered through the DCR as the supply voltage. Level shifters are used

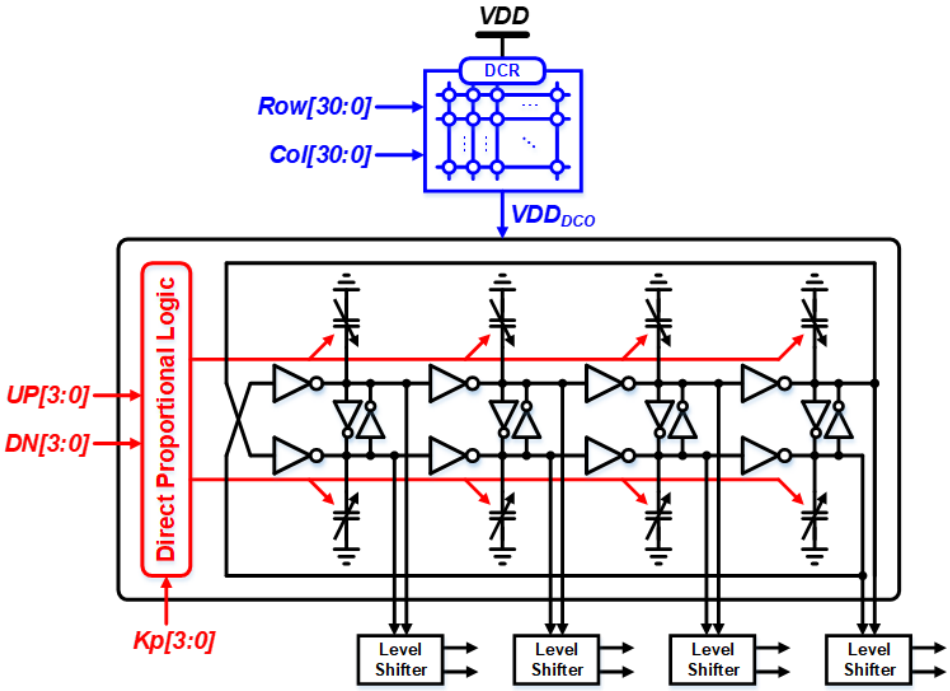


Fig. 3.21 Block diagram of the 8-phase DCO

to increase the low output swing to the full swing due to the lowered supply voltage by DCR. The phase error information from the analog BBPD is fed back to the varactor in proportion to the direct-proportional gain to temporarily change the DCO frequency.

3.4 Measurement Results

The proposed referenceless CDR prototype has been designed and fabricated in 40nm CMOS technology. The chip photomicrograph is shown in Fig. 3.22 with anctive area of 0.032 mm^2. The proposed CDR has two voltage domains separated into digital and analog, both using 1.0 V. As shown in Table 3.1, the digital domain includes DES and CDR digital blocks, while the analog domain includes DCO, CTLE, and BBPD.

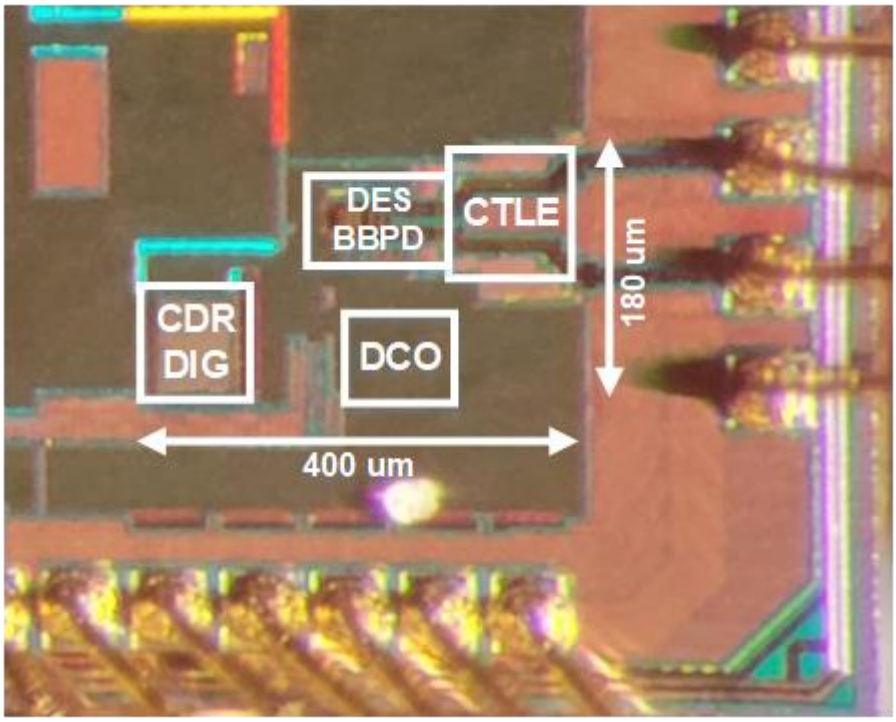


Fig. 3.22 Chip photomicrograph

Table 3.1 Detailed power breakdown of the proposed CDR

Supply	Block	Power [mW]
VDDDIG	CDR Digital	13.7
	Deserializer	
VDDANA	DCO	23.0
	CTLE	
	BBPD	

Total power consumption : 36.7mW

Table 3.2 Detailed area of the proposed CDR

Block	Area (mm^2)
CDR Digital	0.01525
Deserializer	0.00323
DCO	0.00736
CTLE	0.00516
BBPD	0.00136
Total	0.03236

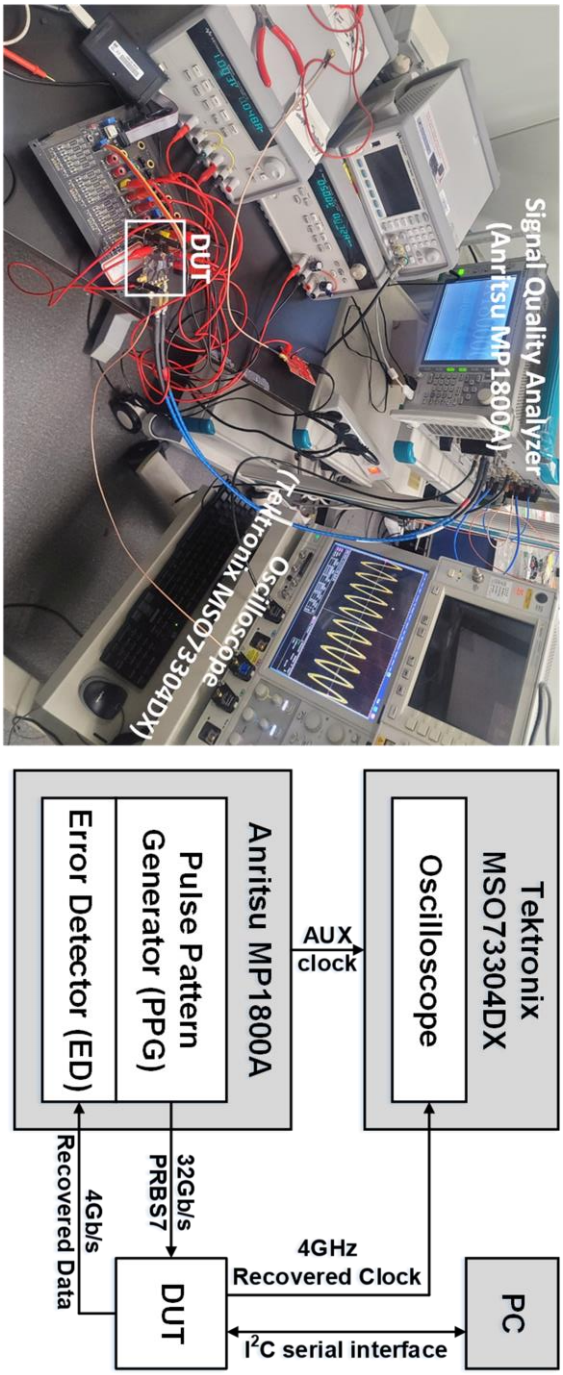


Fig. 3.23 Measurement setup.

Digital domain consumes 13.7 mW and analog domain consumes 23.0 mW at 32Gb/s. The prototype chip achieves and an energy efficiency of 1.15 pJ/b at 32Gb/s. The detailed area of the proposed CDR is described in Table 3.2.

Fig. 3.23 shows the measurement setup of the prototype chip. The pulse pattern generator module (PPG) of the Signal Quality Analyzer (Anritsu MP1800A) generates a PRBS7 pattern for the input data. For the jitter tolerance measurement, the Signal Quality Analyzer generates data to which random jitter is added, and the 4Gb/s recovered data from the device under test (DUT) is fed back to the error detector (ED) module of the Signal Quality Analyzer. The recovered clock from the DUT is monitored by the real-time oscilloscope (Tektronix MSO71604C). With the real-time oscilloscope, frequency acquisition behaviors and jitter histograms of the recovered clock can be measured.

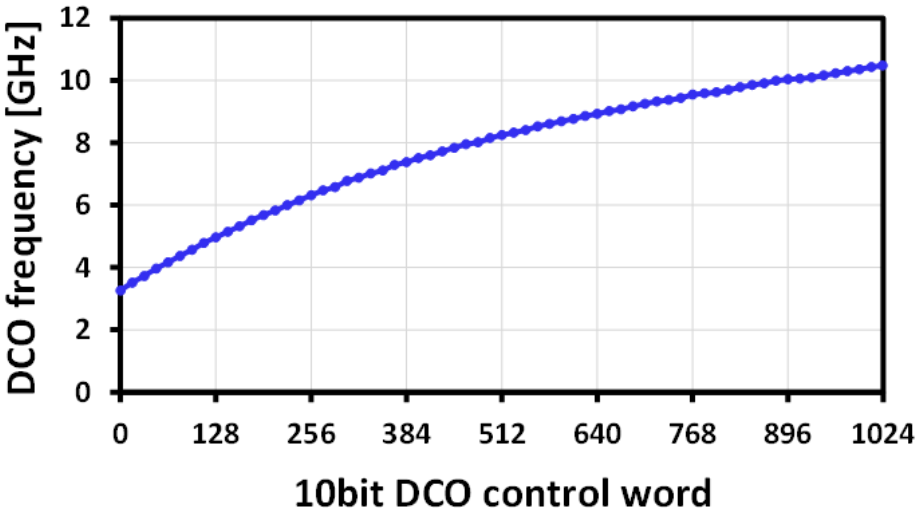


Fig. 3.24 Measured DCO gain curve

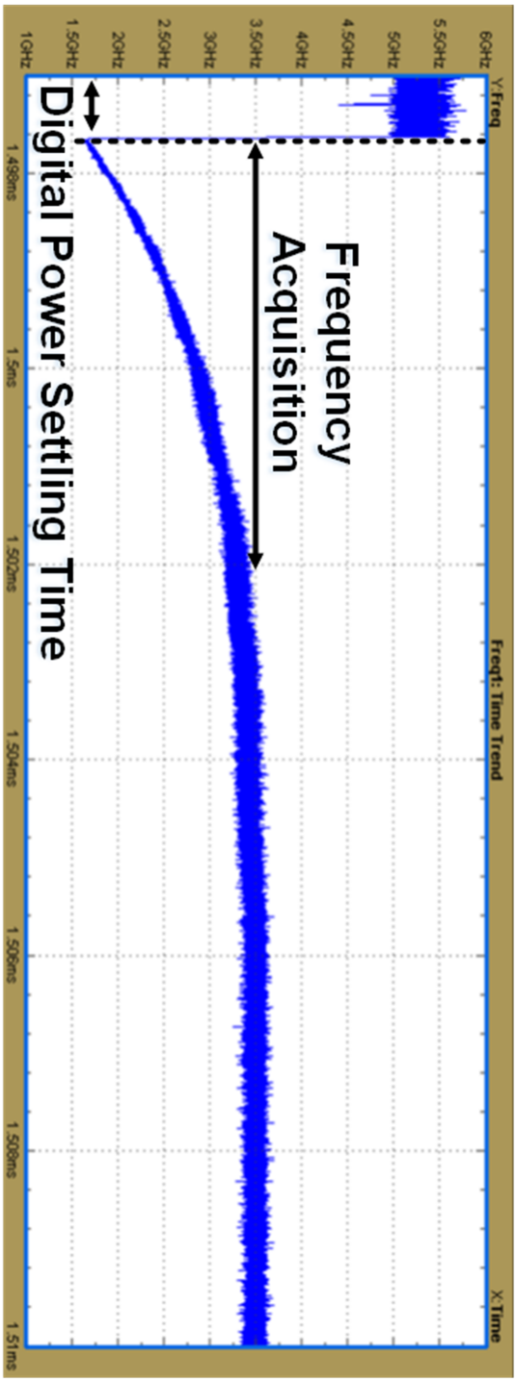


Fig. 3.25 Measured transient response of the proposed CDR @ 28 Gb/s PRBS7

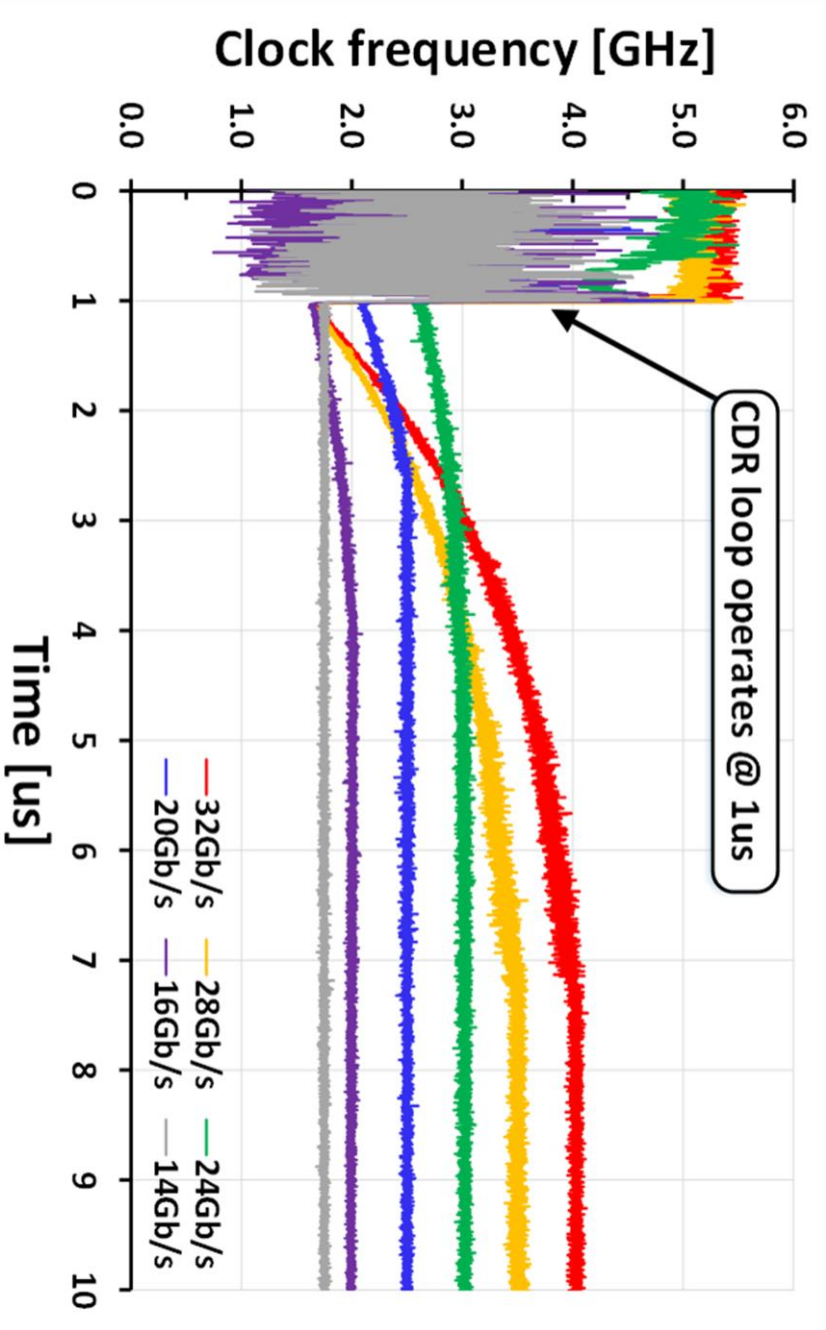


Fig. 3.26 Measured frequency acquisition behavior at various data rates

The measured frequency gain of the DCO is depicted in Fig. 3.24. The operating range of the DCO is from 3.26 GHz to 10.48 GHz. Thus, the prototype chip can operate from 13.04 Gb/s to 41.92 Gb/s. However, the maximum operating data rate is limited to the PPG equipment limit of 32Gb/s.

Fig. 3.25 shows a measured transient response of 28 Gb/s input data. Since there is no external trigger signal in the prototype chip, the trigger signal for the real-time oscilloscope is tied to the power supply of the digtal domain. The trigger signal rises to 1 as the power supply of the digital domain is turned on, and the oscilloscope starts measuring the transient response from the time the trigger rises to 1. As a result, after about 1.5 ms of time, the power supply of the digital domain is stabilized, and then a proper FCW value starts to enter the DCO. For this reason, even though the function of setting the initial frequency code of the DCO is implemented, the initial frequency starts from the low points when measuring the transient response. In order to show various results at once, we post-processed the measured transient responses, as shown in Fig. 3.26. Considering the equipment limit and the operation range of the DCO, it can be seen that frequency lock is achieved properly under all measurable conditions. In addition, the frequency lock is achieved under all conditions before 8us, so the frequency acquisition time of the proposed referenceless CDR can be considered to be less than 7us.

Fig. 3.27 shows the measured JTOL curve of the proposed referenceless CDR. The jitter tolerance is measured with 32 Gb/s input data at the bit error rate of 10^{-12}. At the jitter frequency lower than 31 MHz, the JTOL of the CDR exceeds the maximum jitter amplitude of the test equipment. The jitter tracking bandwidth is about 100MHz.

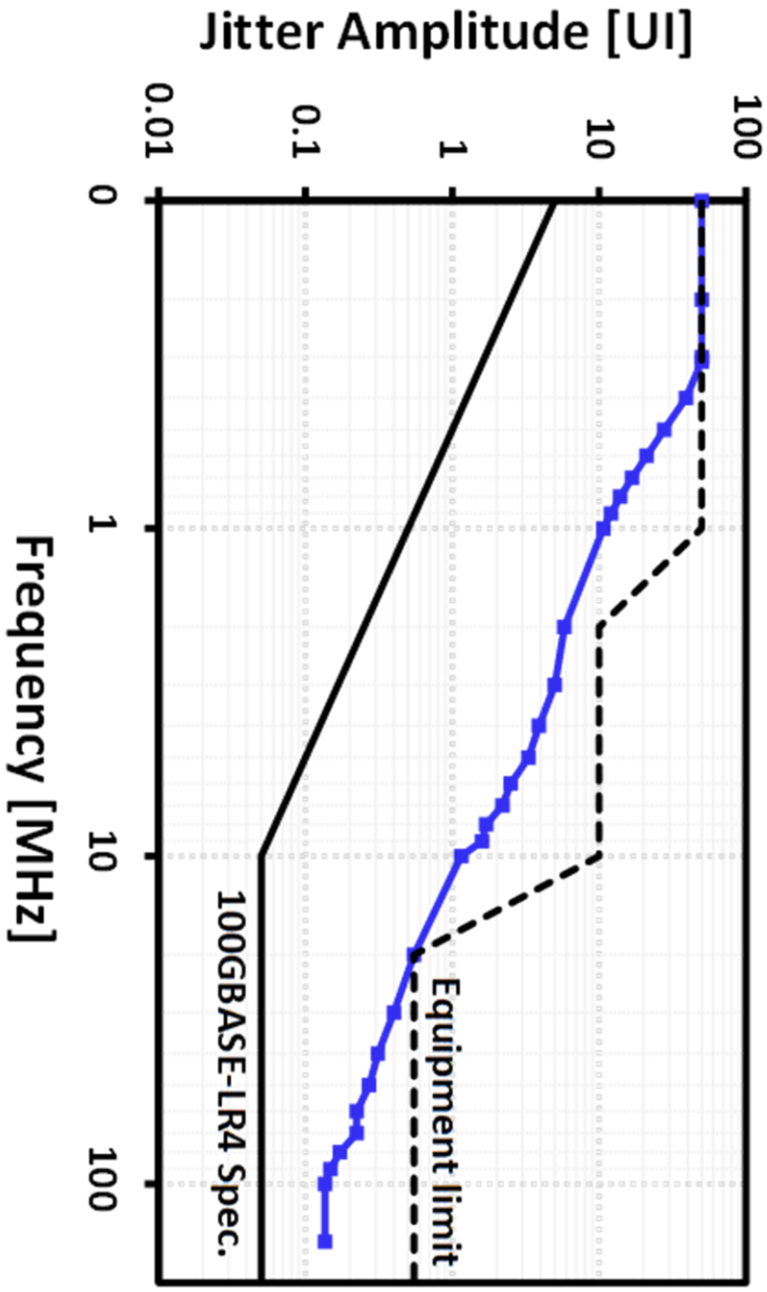


Fig. 3.27 Measured JTOL at 32 Gb/s (BER < 10^{-12})

Fig. 3.28 shows the measured jitter histogram of the recovered clock at the 32 Gb/s. The clock jitters are measured with the external jitter source disabled. The measured RMS and peak-to-peak jitters are 2.9 ps and 27.5 ps, respectively. Table 3.3 summarizes the overall performance of the proposed CDR and compares it with other referenceless CDRs.

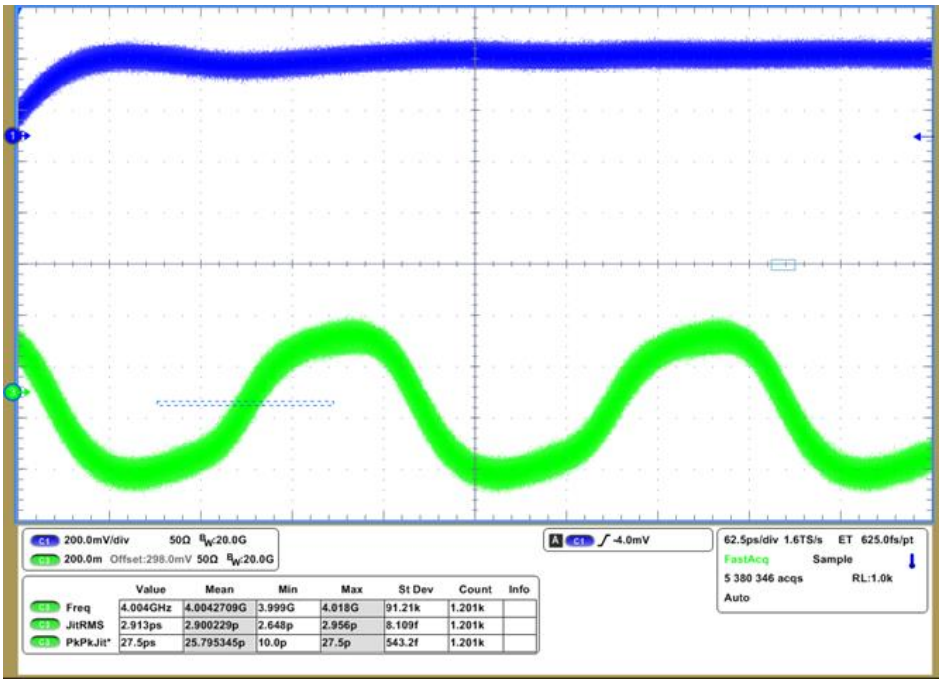


Fig. 3.28 Measured jitter histogram of recovered clock at 32Gb/s

Table 3.3 Performance summary and comparison

	TCAS-I '16	JSSC '17	JSSC '20	JSSC '21	This Work
Technology	65nm	28nm	28nm	40nm	40nm
Data Rate [Gb/s]	0.65-10.5	22.5-32	6.5-12.5	6.4-32	14-32
Architecture	Full-rate	Quarter-rate	Half-rate	Quarter-rate	Quarter-rate
FD Type	Rotational FD	Baud-rate Rotational FD	Quadri-correlator	3-bit Stochastic FD	3-bit Stochastic FD with autocovariance
Capture Range	Not reported	34%	175%	Unlimited	Unlimited
Lock Time [us]	< 50	< 10100	< 8	< 11	< 7
Active Area [mm^2]	0.21	0.2131	0.031	0.041	0.032
Supply Voltage [V]	1.0	0.9	1.0	1.1	1.0
Power [mW]	26	102.0	21.1	30.8	36.7
FoM [pJ/b]	2.6	3.19	2.11	0.96	1.15

Chapter 4

Conclusion

In this dissertation, design techniques of referenceless CDR based on the stochastic frequency detector are proposed. In addition, this thesis also proposes an analysis technique that can explain the frequency acquisition behavior of the SFD. Using the proposed analysis technique, the incomplete frequency acquisition behavior of the previous SPFD is fully solved by introducing the autocorrelation function.

The proposed referenceless CDR adopts a dual-loop architecture with a direct-proportional path and a digital integral path. The direct proportional path is configured by feeding the output of a separate analog BBPD back to the varactor of the DCO. The digital integrated path consists of a path through which the frequency control word of the DCO generated by the DLF adjusts the DCR of the DCO.

The proposed referenceless CDR shows a fast frequency lock time and unlimited capture range with a small active area of 0.032mm^2. The prototype chip is fabricated in 40-nm CMOS technology and achieves an energy efficiency of 1.15pJ/b at 32 Gb/s.

Appendix A

Detailed Frequency Acquisition Waveforms of the proposed SFD

This chapter contains graphs detailing the waveforms in Figure 3.17 for each Ki. The graphs are obtained with system-Verilog. For each Ki, there is an optimal Kp in which the dithering amplitude is minimized. As Ki increases, the dithering amplitude also increases, and it is difficult to say that frequency lock is performed from Ki 7. Ki used for the measurement is 4.

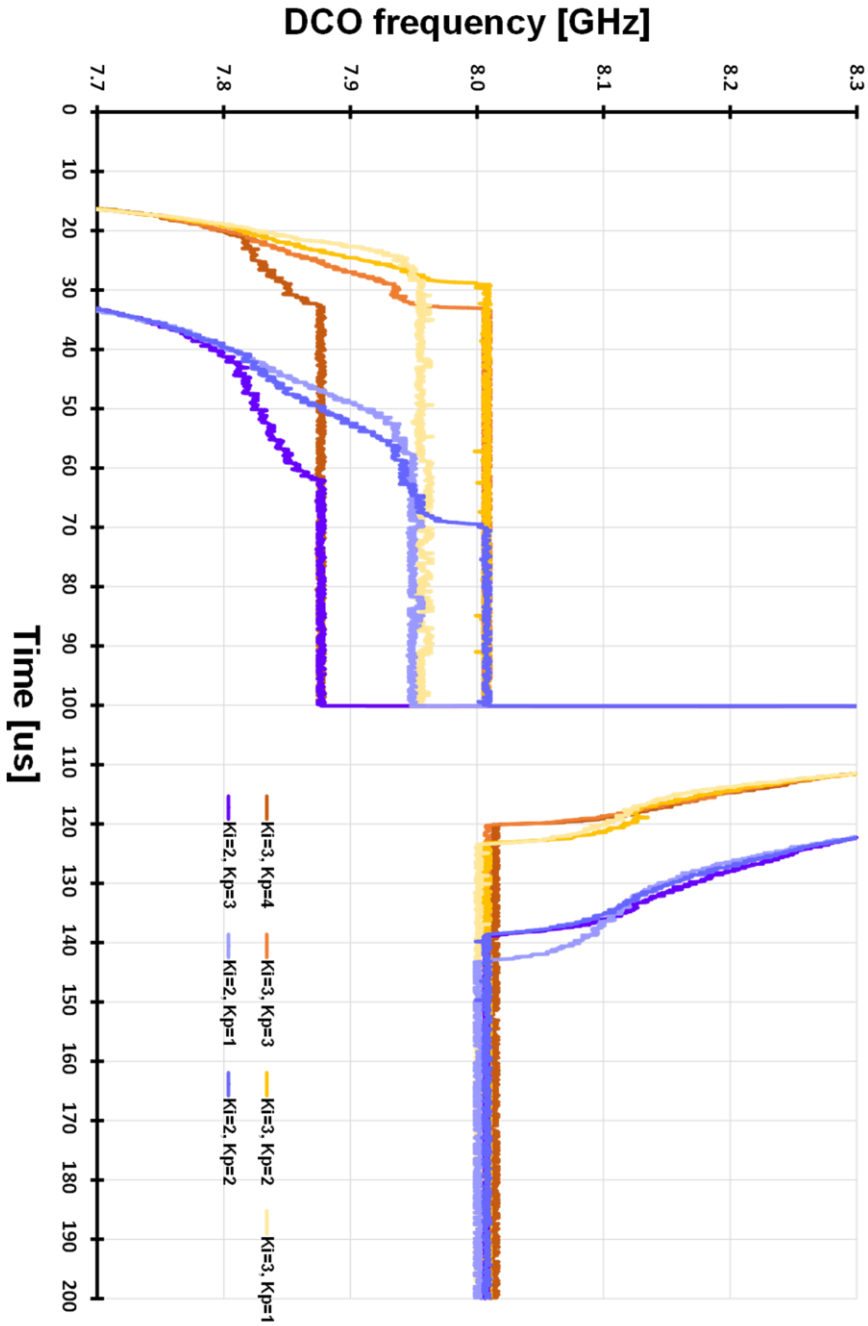


Fig. A.1 32Gb/s 7-dB channel loss PRBS7 frequency acquisition behavior of the proposed SFD (Ki=2, 3)

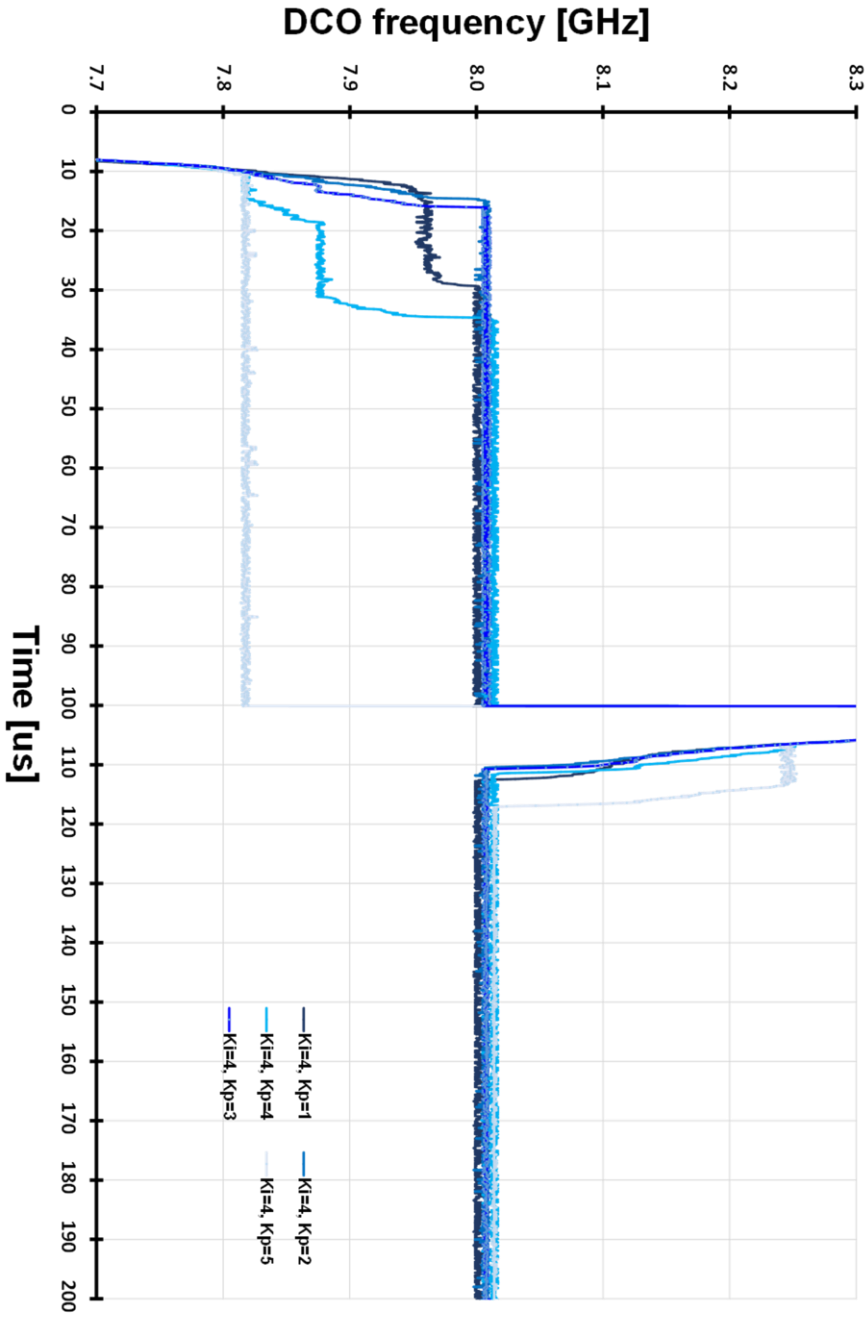


Fig. A.2 32Gb/s 7-dB channel loss PRBS7 frequency acquisition behavior of the proposed SFD (Ki=4)

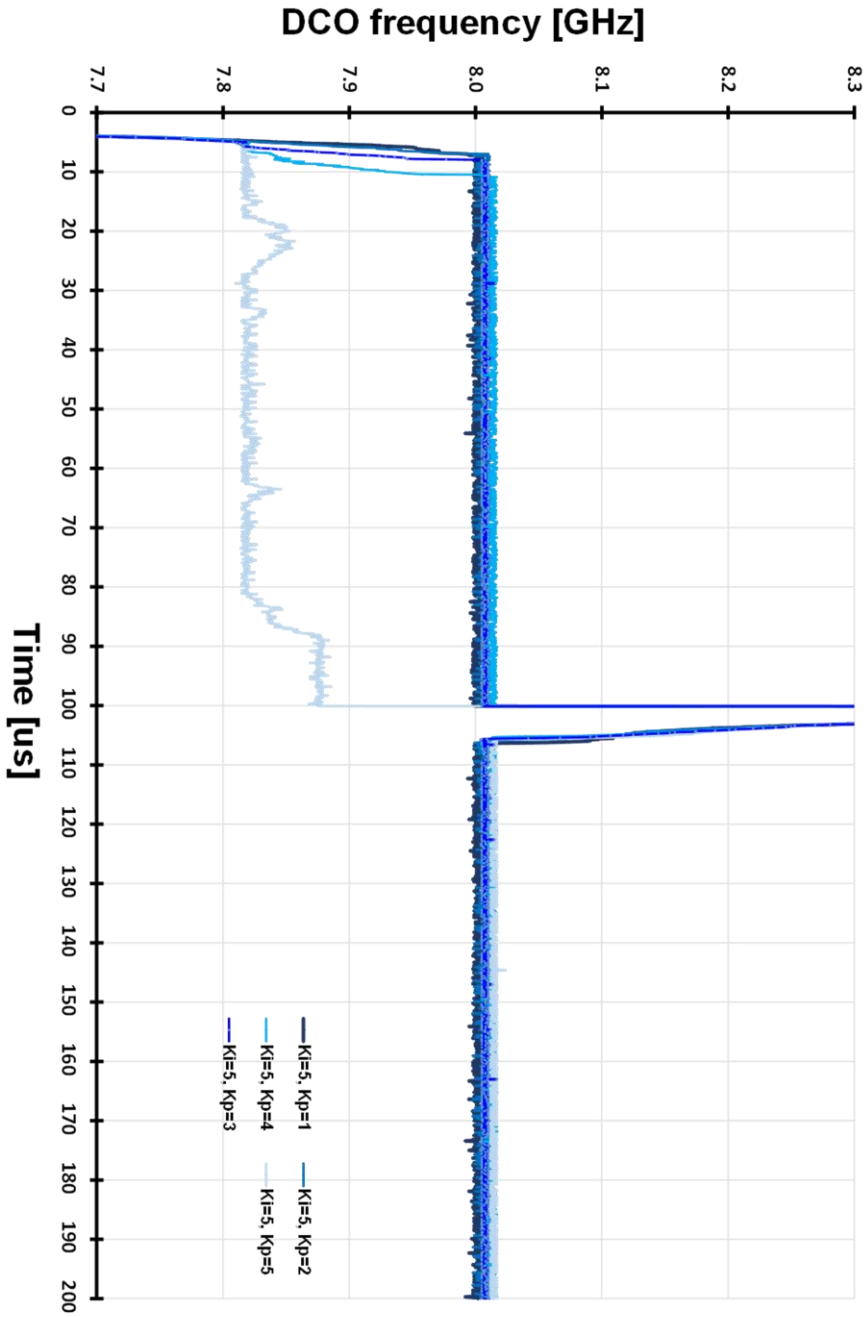


Fig. A.3 32Gb/s 7-dB channel loss PRBS7 frequency acquisition behavior of the proposed SFD (Ki=5)

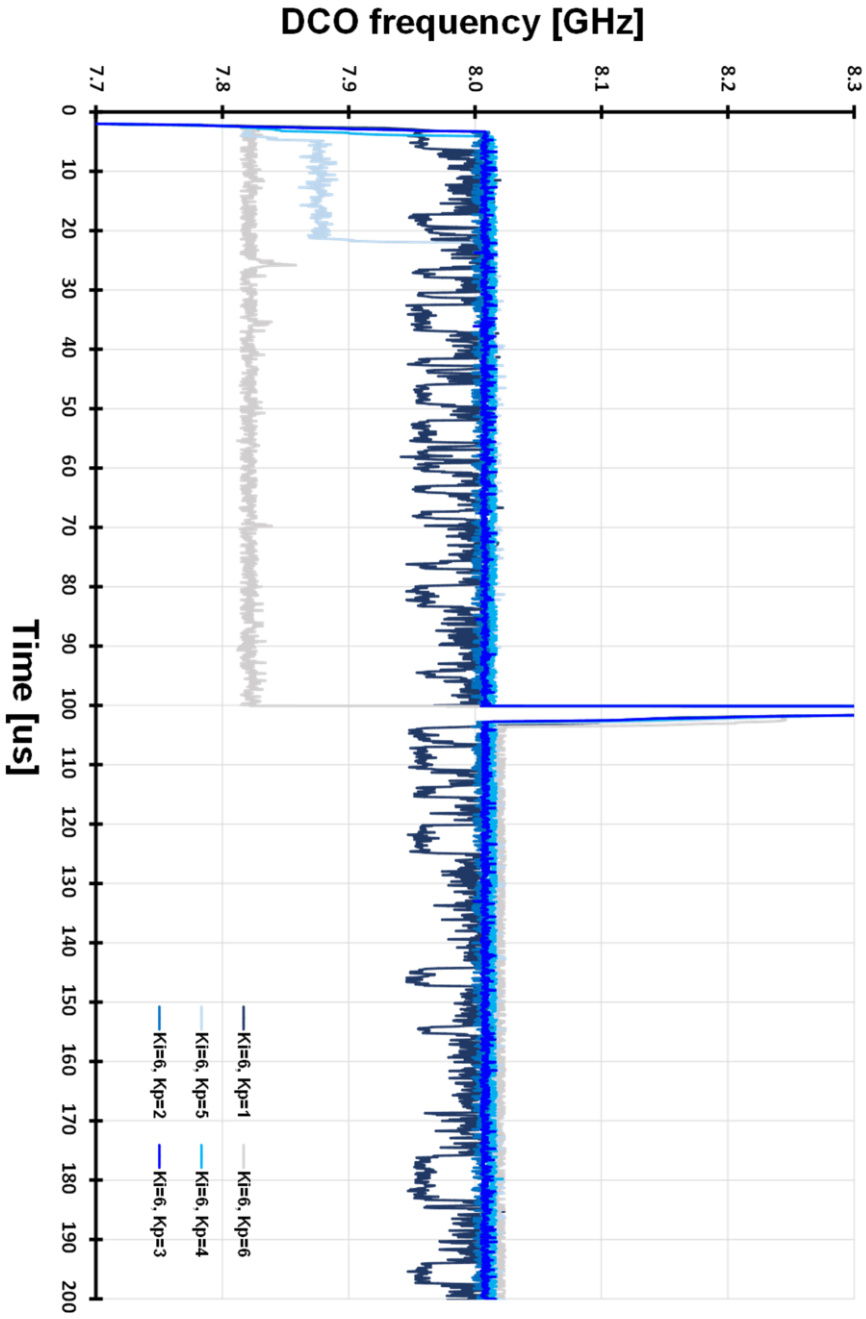


Fig. A.4 32Gb/s 7-dB channel loss PRBS7 frequency acquisition behavior of the proposed SFD (Ki=6)

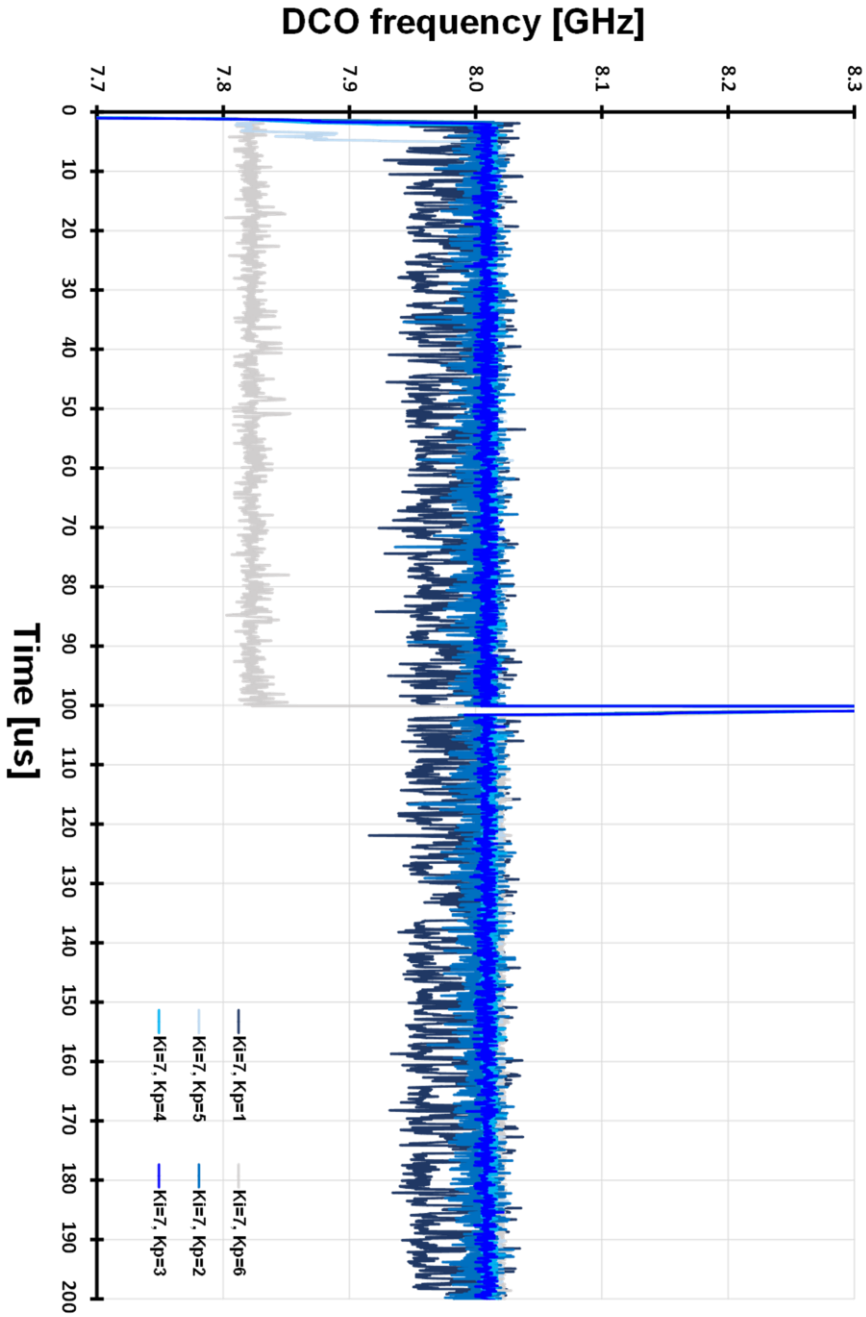


Fig. A.5 32Gb/s 7-dB channel loss PRBS7 frequency acquisition behavior of the proposed SFD (Ki=7)

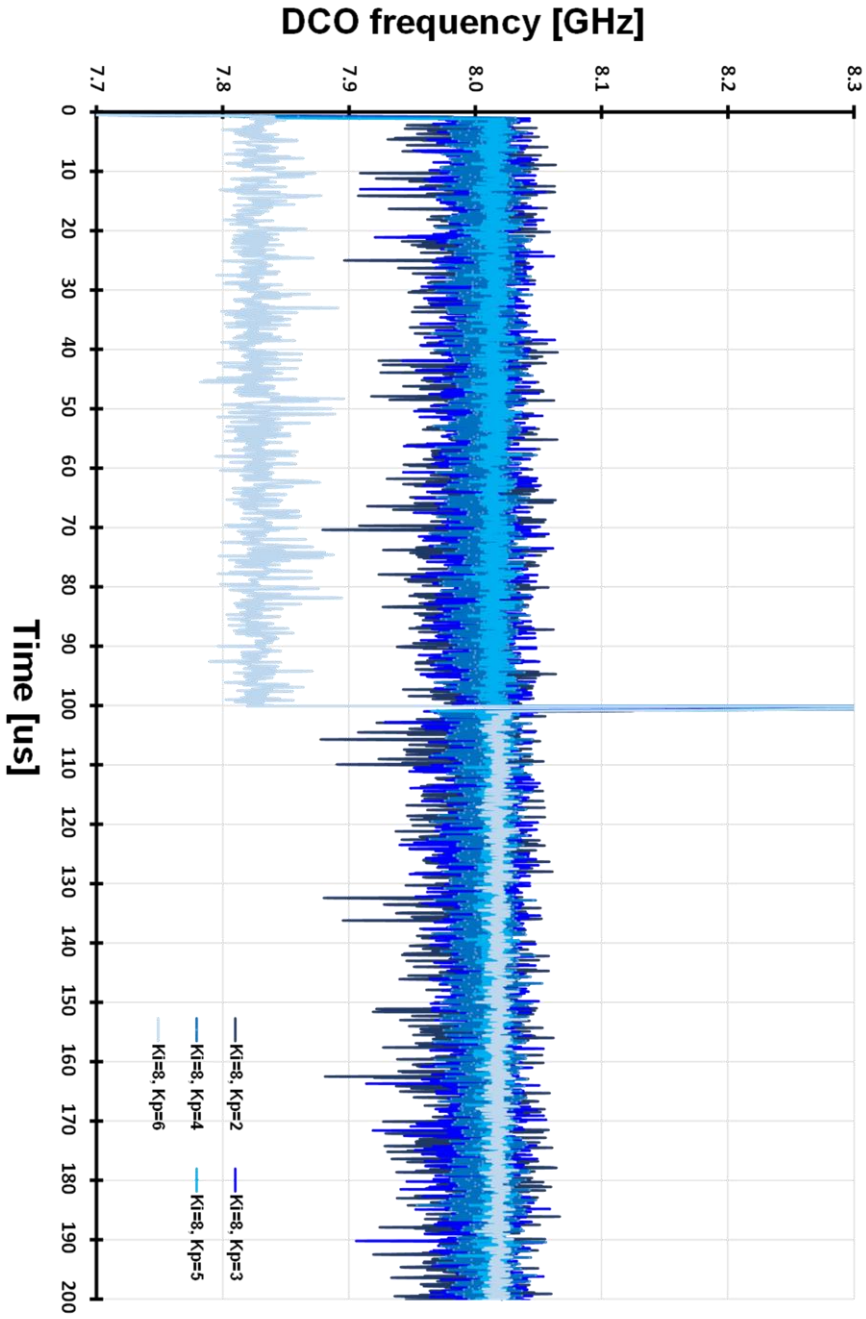


Fig. A.6 32Gb/s 7-dB channel loss PRBS7 frequency acquisition behavior of the proposed SFD (Ki=8)

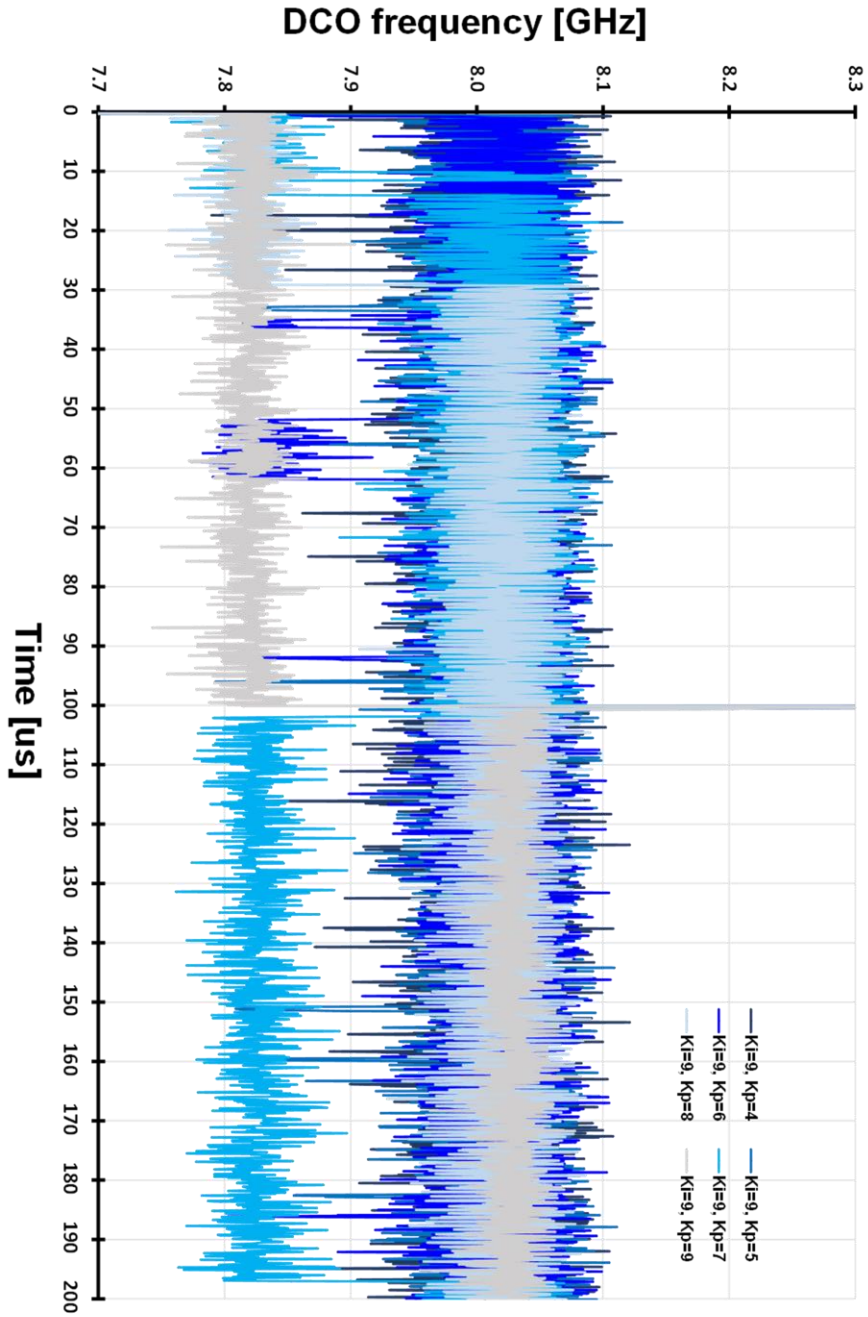


Fig. A.7 32Gb/s 7-dB channel loss PRBS7 frequency acquisition behavior of the proposed SFD (Ki=9)

Bibliography

[1] Sandvine, “The Global Internet Phenomena Report COVID-19 Spotlight”, Online(accessed Apr. 30, 2022), Available: *https://www.sandvine.com/hubfs/Sandvine_Redesign_2019/Downloads/2020/Phenomena/COVID%20Internet%20Phenomena%20Report%2020200507.pdf* (2020).

[2] Sandvine, “The Global Internet Phenomena Report”, Online(accessed Apr. 29, 2021), Available: *https://www.sandvine.com/hubfs/Sandvine_Redesign_2019/Downloads/2022/Phenomena%20Reports/GIPR%202022/Sandvine%20GIPR%20January%202022.pdf?utm_referrer=https%3A%2F%2Fwww.sandvine.com%2Fglobal-internet-phenomena-report-2022%3Fhs_preview%3DkhpPseNo-62343537839* (2022).

[3] Cisco, “Cisco Annual Internet Report (2018-2023)”, Online(accessed Apr. 30, 2022) Available: *https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.pdf* (2020)

[4] Cisco, “Cisco Visual Networking Index: Forecast and Trends, 2017–2022”, Online(Accessed: Apr. 30, 2022), Available: *https://twiki.cern.ch/twiki/pub/HEPIX/TechwatchNetwork/HtwNetworkDocuments/white-paper-c11-741490.pdf* (2019)

[5] IDC, “Data Age 2025: The Digitization of the World From Edge to Core”, Online(Accessed Apr. 30, 2022), Available: *https://www.seagate.com/files/www-content/our-story/trends/files/idc-seagate-dataage-whitepaper.pdf* (2018)

[6] ISSCC, “2022 Press Kit”, Online(accessed May. 02, 2022), Available: *https://www. isscc.org/past-conferences* (2022)

[7] HMS, “Inustrial networks keep growing despite challenging times”, Online(Accessed May. 3, 2022), Available: *https://www.hms-net-works.com/news-and-insights/news-from-hms/2022/05/02/industrial-net-works-keep-growing-despite-challenging-times*

[8] Ethernet alliance, “Ethernet Roadmap 2022”, Online(Accessed May. 04, 2022), Available: *https://ethernetalliance.org/wp-content/up-loads/2022/03/Ethernet-Roadmap2022-Final.pdf*

[9] "SFP Definition from PC Magazine Encyclopedia", Online(Accessed May. 04, 2022) Available: *https://www.pcmag.com/encyclopedia/term/sfp*

[10] ProLabs, “Comparing 200G/400G capable form factors (QSFP-DD vs. QSFP+/QSFP28/OSFP/CFP9/COBO)”, Online(Accessed May. 04, 2022) Available : *https://www.prolabs.com/industry/differences-between-qsfp-dd-and-qsfp-qsfp28-qsfp56-osfp-cfp8-cobo*

[11] A. Rezayee and K. Martin, “A 9–16Gb/s clock and data recovery circuit

with three-state phase detector and dual-path loop architecture," *in Proc. Eur. Solid-State Circuits Conf.*, Sep. 2003, pp. 683–686.

[12] J. Savoj and B. Razavi, "A 10-Gb/s CMOS clock and data recovery circuit with a half-rate binary phase/frequency detector," *IEEE J. Solid-State Circuits*, vol. 38, no. 1, pp. 13–21, Jan. 2003.

[13] D. Dalton et al., "A 12.5-Mb/s to 2.7-Gb/s continuous-rate CDR with automatic frequency acquisition and data-rate readback," *IEEE J. Solid-State Circuits*, vol. 40, no. 12, pp. 2713–2725, Dec. 2005.

[14] A. Kiaei, M. Bohsali, A. Bahai, and T. H. Lee, "A 10Gb/s NRZ receiver with feedforward equalizer and glitch-free phase-frequency detector," *in Proc. Eur. Solid-State Circuits Conf.*, Sep. 2009, pp. 372–375.

[15] S.-K. Lee et al., "A 650Mb/s-to-8Gb/s referenceless CDR circuit with automatic acquisition of data rate," *in IEEE ISSCC Dig. Tech. Papers*, Feb. 2009, pp. 184–185.

[16] C.-F. Liang, S.-C. Hwu, Y.-H. Tu, Y.-L. Yang, and H.-S. Li, "A reference-free, digital background calibration technique for gated-oscillator-based CDR/PLL," *in Symp. VLSI Circuits Dig. Tech. Papers*, 2009, pp. 14-15.

[17] J. Lee and K. C. Wu, "A 20-Gb/s full-rate linear clock and data recovery circuit with automatic frequency acquisition," *IEEE J. Solid-State Circuits*, vol. 44, no. 12, pp. 3590–3602, Dec. 2009.

[18] R. Inti, W. Yin, A. Elshazly, N. Sasidhar, and P. Hanumolu, “A 0.5-to-2.5-Gb/s reference-less half-rate digital CDR with unlimited frequency acquisition range and improved input duty-cycle error tolerance,” *IEEE J. Solid-State Circuits*, vol. 46, no. 12, pp. 3150–3162, Dec. 2011.

[19] W.-Y. Lee, K.-D. Hwang, and L.-S. Kim, “A 5.4/2.7/1.62-Gb/s receiver for DisplayPort version 1.2 with multi-rate operation scheme,” *IEEE Trans. Circuits Syst. I: Reg. Papers*, vol. 59, no. 12, pp. 2858–2866, Dec. 2012.

[20] M. S. Jalali, R. Shivnaraine, A. Sheikholeslami, M. Kibune, and H. Tamura, “An 8 mW frequency detector for 10 Gb/s half-rate CDR using clock phase selection,” *in Proc. IEEE CICC*, Sep. 2013, pp. 1–4.

[21] R. Shivnaraine, M. S. Jalali, A. Sheikholeslami, M. Kibune, and H. Tamura, “An 8–11 Gb/s reference-less bang-bang CDR enabled by “Phase reset”,” *IEEE Trans. Circuits Syst. I: Reg. Papers*, vol. 61, no. 6, pp. 2129–2138, Jun. 2013.

[22] N. Kocaman et al., “An 8.5–11.5-Gbps SONET transceiver with reference-less frequency acquisition,” *IEEE J. Solid-State Circuits*, vol. 48, no. 8, pp. 1875–1884, Aug. 2013.

[23] G. Shu et al., “A reference-less clock and data recovery circuit using phase-rotating phase-locked loop,” *IEEE J. Solid-State Circuits*, vol. 49, no. 4, pp. 1036–1047, Apr. 2014.

[24] F.-T. Chen et al., “A 10-Gb/s low jitter single-loop clock and data recovery circuit with rotational phase frequency detector,” *IEEE Trans. Circuits Syst. I:*

Reg. Papers, vol. 61, no. 11, pp. 3278–3287, Nov. 2014.

[25] M. S. Jalali, A. Sheikholeslami, M. Kibune, and H. Tamura, "A reference-less single-loop half-rate binary CDR," *IEEE J. Solid-State Circuits*, vol. 50, no. 9, pp. 2037–2047, Sep. 2015.

[26] S. Huang, J. Cao, and M. Green, "An 8.2 Gb/s-to-10.3 Gb/s full-rate linear referenceless CDR without frequency detector in 0.18 μm CMOS," *IEEE J. Solid-State Circuits*, vol. 50, no. 9, pp. 2048–2060, Sep. 2015.

[27] J.-Y. Kim, J. Song, J. You, S. Hwang, S.-G. Bae, and C. Kim, "A 250Mb/s to 6Gb/s reference-less clock and data recovery circuit with clock frequency multiplier," *IEEE Trans. Circuits Syst. II: Express Briefs*, vol. 64, no. 6, pp. 650–654, Nov. 2015.

[28] J.-H. Yoon, S.-W. Kwon, and H.-M. Bae, "A DC-to-12.5Gb/s 4.88mW/Gb/s all-rate CDR with a single LC VCO in 90nm CMOS," *in Proc. IEEE CICC*, Sep. 2015, pp. 1–4.

[29] Y. Tsunoda et al., "A 24-to-35Gb/s ×4 VCSEL driver IC with multi-rate referenceless CDR in 0.13μm SiGe BiCMOS," *in IEEE ISSCC Dig. Tech. Papers*, Feb. 2015, pp. 414–415.

[30] S. Choi et al., "A 0.65-to-10.5 Gb/s reference-less CDR with asynchronous baud-rate sampling for frequency acquisition and adaptive equalization," *IEEE Trans. Circuits Syst. I: Reg. Papers*, vol. 63, no. 2, pp. 276–287, Feb. 2016.

[31] G. Shu et al., "A 4-to-10.5 Gb/s continuous-rate digital clock and data recovery with automatic frequency acquisition," *IEEE J. Solid-State Circuits*, vol. 51, no. 2, pp. 428–439, Feb. 2016.

[32] T. Masuda et al., "A 12Gb/s 0.9mW/Gb/s wide-bandwidth injection-type CDR in 28nm CMOS with reference-free frequency capture," *IEEE ISSCC Dig. Tech. Papers*, Feb. 2016, pp. 188–189.

[33] S. Byun, "A 400 Mb/s~2.5 Gb/s referenceless CDR IC using intrinsic frequency detection capability of half-rate linear phase detector," *IEEE Trans. Circuits Syst. I: Reg. Papers*, vol. 63, no. 10, pp. 1592–1604, Oct. 2016.

[34] K. Lee, J.-Y. Sim, "A 0.8-to-6.5 Gb/s continuous-rate reference-less digital CDR with half-rate common-mode clock-embedded signaling," *IEEE Trans. Circuits Syst. I: Reg. Papers*, vol. 63, no. 4, pp. 482–493, Apr. 2016.

[35] J.-Y. Lee et al., "A 4×10-Gb/s referenceless-and-masterless phase rotator-based parallel transceiver in 90-nm CMOS," *IEEE Trans. VLSI Syst.*, vol. 24, no. 6, pp. 2310-2320, Jun. 2016.

[36] J. Song, S. Hwang, and C. Kim, "A 4× 5-Gb/s 1.12-μs locking time Reference-less receiver with asynchronous sampling-based frequency acquisition and clock shared subchannels," *IEEE Trans. VLSI Syst.*, vol. 24, no. 8, pp. 2768-2777, Aug. 2016.

[37] T. Iizuka, et al., "A 4-cycle-start-up reference-clock-less all-digital burst-mode CDR based on cycle-lock gated-oscillator with frequency tracking," *in Proc. European Solid-State Circuits Conf.*, 2016, pp. 301–304.

[38] T. Masuda et al., “A 12Gb/s 0.9mW/Gb/s wide-bandwidth injection-type CDR in 28nm CMOS with reference-free frequency capture,” *IEEE J. Solid-State Circuits*, vol. 51, no. 12, pp. 3204–3215, Dec. 2016.

[39] Y.-H. Kim, D. Lee, D. Lee, and L.-S. Kim, “A 10Gb/s reference-less baud-rate CDR for low power consumption with direct feedback method,” *IEEE Trans. Circuits Syst. II: Express Briefs*, vol. 65, no. 11, pp. 1539–1543, Oct. 2017.

[40] Y.-H. Kim, T. Lee, H.-K. Jeon, and L.S. Kim, “An input data and power noise inducing clock jitter tolerant reference-less digital CDR for LCD intra-panel interface,” *IEEE Trans. Circuits Syst. I: Reg. Papers*, vol. 64, no. 4, pp. 823–835, Apr. 2017.

[41] Y.-L. Lee, S.-J. Chang, Y.-C. Chen, and Y.-P. Cheng, “An unbounded frequency detection mechanism for continuous-rate CDR circuits,” *IEEE Trans. Circuits Syst. II: Express Briefs*, vol. 64, no. 5, pp. 500–504, May. 2017.

[42] C. H. Son, and S. Byun, “On frequency detection capability of full-rate linear and binary phase detectors,” *IEEE Trans. Circuits Syst. II: Express Briefs*, vol. 64, no. 7, pp. 757–761, Jul. 2017.

[43] W. Rahman et al., “A 22.5-to-32 Gb/s 3.2 pJ/b referenceless baud-rate digital CDR with DFE and CTLE in 28 nm CMOS,” *IEEE J. Solid-State Circuits*, vol. 52, no. 12, pp. 3517–3531, Dec. 2017.

[44] J.-H. Yoon, S.-W. Kwon, and H.-M. Bae, “A DC-to-12.5 Gb/s 9.76

mW/Gb/s all-rate CDR with a single LC VCO in 90 nm CMOS," *IEEE J. Solid-State Circuits*, vol. 52, no. 3, pp. 856–866, Mar. 2017.

[45] W. Rahman et al., "A 22.5-to-32 Gb/s 3.2 pJ/b referenceless baud-rate digital CDR with DFE and CTLE in 28 nm CMOS," *in IEEE ISSCC Dig. Tech. Papers*, Feb. 2017, pp. 120–121.

[46] K. Park et al., "A 55.1 mW 1.62-to-8.1 Gb/s video interface receiver generating up to 680 MHz stream clock over 20 dB loss channel," *IEEE Trans. Circuits Syst. II: Express Briefs*, vol. 64, no. 12, pp. 1432–1436, Dec. 2017.

[47] K. Park, W. Bae, and D.-K. Jeong, "A 27.1 mW, 7.5-to-11.1 Gb/s single-loop referenceless CDR with direct up/dn control," *in Proc. IEEE CICC*, Apr. 2017, pp. 1–4.

[48] J. Jin, X. Jin, J. Jung, K. Kwon, and J.-H. Chun, "A 0.75-3.0-Gb/s dual-mode temperature-tolerant referenceless CDR with a deadzone-compensated frequency detector," *IEEE J. Solid-State Circuits*, vol. 53, no. 10, pp. 2994–3003, Oct. 2018.

[49] B. Schell, R. Bishop, and J. Kenney, "A 3-12.5 Gb/s reference-less CDR for an eye-opening monitor," *in Proc. European Solid-State Circuits Conf.*, 2018, pp. 186–189.

[50] K. Park, W. Bae, J. Lee, J. Hwang, and D.-K. Jeong, "A 6.7-11.2 Gb/s, 2.25 pJ/bit, single-loop referenceless CDR with multi-phase oversampling PFD in 65-nm CMOS," *IEEE J. Solid-State Circuits*, vol. 53, no. 10, pp. 2982–3993,

Oct. 2018.

[51] K. Park et al., "A 4–20-Gb/s 1.87-pJ/b Continuous-Rate Digital CDR Circuit With Unlimited Frequency Acquisition Capability in 65-nm CMOS," *IEEE J. Solid-State Circuits*, vol. 56, no. 5, pp. 1597-1607, May 2021

[52] K. Park, M. Shim, H. -G. Ko, B. Nikolić and D. -K. Jeong, "Design Techniques for a 6.4–32-Gb/s 0.96-pJ/b Continuous-Rate CDR With Stochastic Frequency–Phase Detector," *IEEE J. Solid-State Circuits*, vol. 57, no. 2, pp. 573-585, Feb. 2022

[53] K. L. Chan et al., "A 32.75-Gb/s voltage-mode transmitter with three-tap FFE in 16-nm CMOS," *IEEE J. Solid-State Circuits*, vol. 52, no. 10, pp. 2663-2678, Oct. 2017.

[54] P.-J. Peng et al., "A 50-Gb/s quarter-rate voltage-mode transmitter with three-rap FFE in 40-nm CMOS," in *Proc. IEEE European Solid-State Circuits Conference*, Sep. 2018, pp. 174-177.

[55] A. Ramachandran et al., "A 16Gb/s 3.6pJ/bit wireline transceiver with phase domain equalization scheme: Integrated Pulse Width Modulation (iPWM) in 65nm CMOS," *in IEEE ISSCC Dig. Tech. Papers*, pp. 488-489, 2017.

[56] H. Ju et al., "A 64Gb/s 1.5pJ/bit PAM-4 transmitter with 3-tap FFE and Gm-regulated active-feedback driver in 28nm CMOS," ” *in Symp. VLSI Circuits Dig. Tech. Papers*, Jun. 2018, pp. 51–52.

[57] J. Lee et al., "A 0.1pJ/b/dB 1.62-to-10.8Gb/s video interface receiver with fully adaptive equalization using un-even data level," ” *in Symp. VLSI Circuits Dig. Tech. Papers*, Jun. 2019, pp. 198–199.

[58] T. Musah *et al*., "A 4-32 Gb/s bidirectional link with 3-tap FFE/6-tap DFE and collaborative CDR in 22nm CMOS," *IEEE J. Solid-State Circuits*, vol. 49, no. 12, pp. 3079–3090, Dec. 2014.

[59] S. Saxena et al., "A 5 Gb/s energy-efficient voltage-mode transmitter using time-based de-emphasis," *IEEE J. Solid-State Circuits*, vol. 49, no. 8, pp. 1827–1836, Aug. 2014.

[60] J. Lee, K. Park, K. Lee and D. Jeong, "A 2.44-pJ/b 1.62–10-Gb/s Receiver for Next Generation Video Interface Equalizing 23-dB Loss With Adaptive 2-Tap Data DFE and 1-Tap Edge DFE," *IEEE Trans. Circuits Syst. II: Express Briefs*, vol. 65, no. 10, pp. 1295-1299, Oct. 2018

[61] G. Jeong, B. Kang, H. Ju, K. Park and D. Jeong, "A Modulo-FIR Equalizer for Wireline Communications," *IEEE Trans. Circuits Syst. I: Reg. Papers*, vol. 66, no. 11, pp. 4278-4286, Nov. 2019

[62] H. -G. Ko, S. Shin, J. Oh, K. Park and D. -K. Jeong, "6.7 An 8Gb/s/μm FFE-Combined Crosstalk-Cancellation Scheme for HBM on Silicon Interposer with 3D-Staggered Channels," *in IEEE ISSCC Dig. Tech. Papers*, 2020, pp. 128-130

[63] B. Kang et al., "A 10 Gb/s PAM-4 Transmitter With Feed-Forward Implementation of Tomlinson-Harashima Precoding in 28 nm CMOS," *IEEE Access*, vol. 9, pp. 156789-156798, 2021

[64] K. Lee et al., "An Adaptive Offset Cancellation Scheme and Shared-Summer Adaptive DFE for 0.068 pJ/b/dB 1.62-to-10 Gb/s Low-Power Receiver in 40 nm CMOS," *IEEE Trans. Circuits Syst. II: Express Briefs*, vol. 68, no. 2, pp. 622-626, Feb. 2021

[65] H. Do *et al.* "A 64 Gb/s 2.09 pJ/b PAM-4 VCSEL Transmitter with Bandwidth Extension Techniques in 40 nm CMOS," in *IEEE Asian Solid-State Circuits Conf.*, Nov. 2021, pp. 1-3

[66] W. J. Dally and J. W. Poulton, *Digital System Engineering*, Cambridge University Press, 1988.

[67] Renesas Electronic Corp., "PCI Express Reference Clock Requirements", Online(Accessed May. 11, 2022), Available: *https://www.renesas.com/kr/en/document/apn/843-pci-express-reference-clock-requirements?language=en*

[68] B. Razavi, *Design of integrated circuits for optical communication*, McGraw-Hill Professional, 2002.

[69] Telcordia, "Synchronous Optical Network (SONET) Transport Systems: Common Generic Criteria, GR-253-CORE", Issue 4, December 2005.

[70] Wikipedia, Jitter. Online(Accessed May. 12, 2022), Available: *https://en.wikipedia.org/wiki/Jitter*

[71] T. J. Yamaguchi, K. Ichiyama, H. X. Hou and M. Ishida, "A robust method for identifying a deterministic jitter model in a total jitter distribution," *2009 International Test Conference, 2009*, pp. 1-10.

[72] M. P. Li, J. Wilstrup, R. Jessen and D. Petrich, "A new method for jitter decomposition through its distribution tail fitting," *International Test Conference 1999. Proceedings (IEEE Cat. No.99CH37034)*, 1999, pp. 788-794

[73] Agilent Technologies, “Jitter Analysis: The dual-Dirac Model, RJ/DJ, and Q-Scale”, Online(Accessed May. 14, 2022), Available: *https://people.engr.tamu.edu/spalermo/ecen689/jitter_dual_dirac_agilent.pdf*

[74] Common Electrical, I. O (CEI)—Electrical and Jitter Interoperability agreements for 6G+, 11G+bps, 25G+bps I/O and 56G+bps. OIF-CEI-04.0, 2017.

[75] M. Hsieh and G. E. Sobelman, “Architectures for multi-gigabit wire-linked clock and data recovery,” *IEEE Circuits and Systems Magazine*, vol. 8, no. 4, pp. 45–57, Dec. 2008.

[76] B. Razavi, “Challenges in the design of high-speed clock and data recovery circuits,” *IEEE Communications Magazine*, vol. 40, no. 8, pp. 94–101, Aug. 2002.

[77] S. Bashiri, S. Aouini, N. Ben-Hamida and C. Plett, "Analysis and Modeling of the Phase Detector Hysteresis in Bang-Bang PLLs," *IEEE Trans. Circuits Syst. I: Reg. Papers*, vol. 62, no. 2, pp. 347-355, Feb. 2015

[78] A. Pottbacker, U. Langmann, and H. Schreiber, “A Si bipolar phase and frequency detector IC for clock extraction up to 8 Gb/s,” *IEEE J. Solid-State Circuits*, vol. 27, no. 12, pp. 1747–1751, Dec. 1992.

[79] C. Yu, E. Sa, S. Jin, H. Park, J. Shin and J. Burm, "A 6.5–12.5-Gb/s Half-Rate Single-Loop All-Digital Referenceless CDR in 28-nm CMOS," *IEEE J. Solid-State Circuits*, vol. 55, no. 10, pp. 2831-2841, Oct. 2020

[80] J. Jin et al., "A 4.0-10.0-Gb/s Referenceless CDR with Wide-Range, Jitter-Tolerant, and Harmonic-Lock-Free Frequency Acquisition Technique," *ESSCIRC 2018 - IEEE 44th European Solid State Circuits Conference (ESSCIRC)*, 2018, pp. 146-149

[81] J. Kim, “Design of CMOS adaptive-supply serial links”, Stanford University, 2003.

[82] S. Tontisirin, “A Multi Gigabit Clock and Data Recovery Testchip fabricated in 0.18μm CMOS”, Online(Accessed May. 23, 2022), Available: *https://indico.gsi.de/event/4388/contributions/20205/attachments/14999/18990/DOC-2005-Oct-67-1.pdf*

[83] D. G. Messerschmitt, “Frequency detectors for PLL acquisition in timing

and carrier recovery," *IEEE Trans. Communications*, vol. COM-27, no. 9, pp. 1288-1295 , Sep. 1979

[84] N. Da Dalt, E. Thaller, P. Gregorius and L. Gazsi, "A compact triple-band low-jitter digital LC PLL with programmable coil in 130-nm CMOS," *IEEE J. Solid-State Circuits*, vol. 40, no. 7, pp. 1482-1490, July 2005

[85] D. -H. Oh, D. -S. Kim, S. Kim, D. -K. Jeong and W. Kim, "A 2.8Gb/s All-Digital CDR with a 10b Monotonic DCO," *in IEEE ISSCC Dig. Tech. Papers*, 2007, pp. 222-598

초 록

본 논문은 기준 클럭이 없는 고속, 저전력, 광대역으로 동작하는 클럭 및 데이터 복원회로의 설계를 제안한다. 기준 클럭이 없는 동작을 위해서 알렉산더 위상 검출기에 기반한 통계적 주파수 검출기를 사용하는 주파수 획득 방식이 사용된다. 통계적 주파수 검출기의 주파수 추적 양상을 분석하기 위해 패턴 히스토그램 분석 방법론을 제시하였고 시뮬레이션을 통해 검증하였다. 패턴 히스토그램 분석을 통해 얻은 정보를 바탕으로 자기공분산을 이용한 통계적 주파수 검출기를 제안한다. 직접 비례 경로와 디지털 적분 경로를 통해 제안된 기준 클럭이 없는 클럭 및 데이터 복원회로는 모든 측정 가능한 조건에서 주파수 잠금을 달성하는 데 성공하였고, 모든 경우에서 측정된 주파수 추적 시간은 7μs 이내이다. 40-nm CMOS 공정을 이용하여 만들어진 칩은 0.032 mm^2 의 면적을 차지한다. 제안하는 클럭 및 데이터 복원회로는 32 Gb/s 의 속도에서 비트에러율 10^{-12} 이하로 동작하였고, 에너지 효율은 32Gb/s 의 속도에서 1.0V 공급전압을 사용하여 1.15 pJ/b 을 달성하였다.

주요어 : 뱅뱅 위상 검출기, 뱅뱅 위상-주파수 검출기, 클럭 및 데이터 복원회로, 주파수 검출기, 통계적 주파수 검출기, 듀얼 루프, 위상 동기 루프, 기준 클럭 없는 시스템, 제한 없는 주파수 검출

학 번 : 2016-20984