Fractionally Spaced Equalization for High-Speed Links

by

Sanquan Song

B.S., EE, Tsinghua University, 2002
M.S., ECE, University of Massachusetts Amherst, 2005

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 2011

© Massachusetts Institute of Technology 2011. All rights reserved.

Author

Department of Electrical Engineering and Computer Science

January 6, 2011

Certified by

Vladimir M. Stojanović
Associate Professor
Thesis Supervisor

Accepted by

Terry P. Orlando
Chairman, Department Committee on Graduate Students
Fractionally Spaced Equalization for High-Speed Links

by

Sanquan Song

B.S., EE, Tsinghua University, 2002
M.S., ECE, University of Massachusetts Amherst, 2005

Submitted to the Department of Electrical Engineering and Computer Science
on January 6, 2011, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Electrical Engineering and Computer Science

Abstract

As high-speed links enter the multi-Gb/s era, equalization and clock recovery designs become much more challenging. For conventional links, these two loops are separate with different performance metrics, resulting in sub-optimal performance. Fractionally spaced equalization (FSE) inherently unifies these two functions, and therefore is proposed for joint equalization and synchronization in this thesis.

At the system level, this thesis introduces new adaptation techniques for both mesochronous and plesiochronous applications. For mesochronous systems, the divergence issue of the low-cost sign-sign least-mean-square (SSLMS) adaptive algorithm is solved by using update conditioning to effectively increase the quantization resolution. For plesiochronous systems, a digitally-controlled bit-skipping scheme is proposed for frequency offset compensation.

At the circuit level, the voltage-time conversion technique is redesigned to build high-speed, linear and energy-efficient FSE filter taps, which are scalable to advanced technology nodes. All the information is processed by linear current integration, with all integrated currents independent of the channel voltages, avoiding the non-linear voltage-current transformation.

Based on different voltage-to-time converter designs, two proof-of-concept FSE implementations have been fabricated in a 90-nm CMOS process. The first implementation is a 2-way interleaved 2-tap FSE, operating at 4.0 Gb/s, with 2.0 pJ/bit energy-efficiency and 4.3 bits of linearity, showing immunity to the sampling phase. Operating at higher rates (6.25 Gb/s), the second implementation is designed as a 4-way interleaved 2-tap FSE with a 1-tap DFE, which achieves 3.6 pJ/bit energy-efficiency and over 4.0 bits of linearity, demonstrating the convergence of the modified sign-sign least-mean-square (M-SSLMS) algorithm. A third implementation has been designed with on-chip coefficient adaptation loop and bit-skipping scheme for plesiochronous systems.

Thesis Supervisor: Vladimir M. Stojanović
Title: Associate Professor
Acknowledgments

I would like to show my deep appreciation to my advisor, Professor Vladimir M. Stojanović, for leading me to the high-speed links area, guiding my research and supporting me financially. When I joined his team 5 years ago, I was a pure communications student, knowing almost nothing about integrated circuits. It was he who helped me with great patience to build a solid background to do interesting work in this very competitive direction. I cannot forget the experience that I worked for my first tape-out day and night in my cubicle while he worked in his office simultaneously with his door open, ready to help at any time. I am fortunate to have such a helpful, knowledgeable and dedicated advisor.

I want to thank my committee members, Prof. Hae-Seung Lee and Prof. Joel L. Dawson. They took time to have my RQE presentation, read my thesis proposal carefully and serve as my committee members. Their feedback helped me to polish my work. I took course 6.775 from Prof. Hae-Seung Lee and 6.776 from Prof. Joel L. Dawson in my early years at MIT. They are great lecturers and I benefited a lot from them.

I am grateful to all the current and former ISG members, including Fred Chen, Byungsub Kim, Ranko Sredojevic, Ajay Joshi, Jonathan Leu, Michael Georgas, Yan Li, Wei An, Ben Moss, Hossein Fariborzi, Oguzhan Uyar, Zhipeng Li, Fabian Lim, Maxine Lee and Natasa Blitvic. They are my friends and teachers. Our discussions often brought new ideas to my project. The inner-group paper reviews were very helpful to improve my paper quality. Particularly, I owe Fred Chen a debt of gratitude for numerous design aids.

I appreciate National Semiconductor and CICS for financial support. I worked as an intern for two summers in National Semiconductor Laboratory. I was blessed to work with Jerry Socci, Jim Wieser, Mounir Bohsali, Hsinho Wu and Ali Djabbari.

This thesis is for my mom, who has been supporting me consistently and fully for three decades. This thesis is for my wife, Jie Xiong, who encouraged me to overcome the difficulties during the five years to complete my Ph.D program. This thesis, for its worth, is dedicated to all my family.
## Contents

1 Introduction
   1.1 Application and problem overview ............................................... 11
   1.2 Introduction to fractionally spaced equalization .......................... 17
   1.3 Thesis contributions ................................................................. 19
      1.3.1 System level analysis ......................................................... 19
      1.3.2 Circuit level implementation ............................................... 20
   1.4 Thesis organization ................................................................. 21

2 FSE System Study ...................... 23
   2.1 Comparison of FSE and SSE ......................................................... 24
      2.1.1 Analytical Comparison ....................................................... 24
      2.1.2 Numerical comparison ......................................................... 29
   2.2 FSE adaptation ................................................................. 31
      2.2.1 FSE adaptive algorithm overview ....................................... 31
      2.2.2 SSLMS algorithm convergence ............................................. 32
      2.2.3 M-SSLMS algorithm for convergent FSE adaptation .................. 34
   2.3 FSE for plesiochronous systems .............................................. 39
      2.3.1 The need for a frequency offset compensation loop ............... 39
      2.3.2 FSE with bit-skipping scheme for plesiochronous systems ......... 41
   2.4 FSE with a conventional CDR .................................................. 46
      2.4.1 Jitter analysis for conventional SSE with CDR receivers ........... 46
      2.4.2 FSE and CDR combination .................................................... 49
   2.5 Summary ................................................................. 53
3 FSE Tap Circuit Design

3.1 FSE tap design challenges ........................................ 55
3.2 Voltage-time conversion principle ................................. 57
  3.2.1 $V_{TH}$-based V2T design .................................. 57
  3.2.2 $V_M$-based V2T design .................................. 61
  3.2.3 T2V design .................................................. 68
3.3 Comparison of VT-based design and CML-based design ............ 74
3.4 Summary ........................................................ 76

4 FSE Implementation .................................................. 77

4.1 $V_{TH}$-based FSE - 1st generation implementation ................. 78
  4.1.1 Micro-architecture ........................................ 78
  4.1.2 $V_{TH}$-based V2T converter ................................ 79
  4.1.3 T2V converter .............................................. 79
  4.1.4 Slicer ........................................................ 82
  4.1.5 Measurements ............................................... 84
4.2 $V_M$-based FSE - 2nd generation implementation .................. 86
  4.2.1 Micro-architecture ........................................ 88
  4.2.2 $V_M$-based V2T converter ................................ 88
  4.2.3 T2V converter .............................................. 91
  4.2.4 DFE and Adaptation Blocks Implementation .................. 92
  4.2.5 Measurements ............................................... 92
  4.2.6 Comparison of the receive linear equalizers ................ 96
4.3 On-chip adaptation design and estimation - 3rd generation implementation 98
  4.3.1 Coefficient adaptation design ............................ 98
  4.3.2 Bit-skipping algorithm design ............................ 99
  4.3.3 Implementation estimation ............................... 100
4.4 Summary ........................................................ 102

5 Conclusions .................................................................. 105

5.1 FSE system study ................................................... 105
5.2 FSE circuit study ................................................... 106
5.3 Future work ........................................................ 107
A A threshold tunable sense amplifier with high immunity to the input common-mode variation

A.1 Impact of the common-mode variation ........................................ 109
A.2 Compensation scheme for the input common-mode variation .......................... 112
A.3 Summary .............................................................................. 114
Chapter 1

Introduction

1.1 Application and problem overview

There are lots of high-speed link applications in today's electronic systems, which are usually the most important components determining the overall performance. Some examples, such as the CPU-memory interfaces in computers and backplane links in servers and routers, are shown in Figure 1-1. To keep the integrated system performance scaling as the transistor density increases in advanced technologies, the data rate on these links must increase. However, these bandwidth limited electrical copper channels introduce more attenuation as the data rate grows, requiring more sophisticated equalization and synchronization circuits. As shown in Figure 1-2, these losses appear as large inter-symbol interference (ISI) in the time domain, which is the main reason for bit errors in high-speed links, challenging both transmitter and receiver designs.

The channel loss is mainly due to the skin effect and dielectric loss, characterized by channel physical properties, such as length. The notches in the transfer curves are due to channel discontinuities, such as impedance mismatches in connector-board interfaces. The channel’s frequency response varies depending on these properties, which is shown in Figure 1-3. For example, one type of channel can be dramatically different from another type due to different lengths. For the same type of channel, the frequency responses may be different from board to board due to manufacturing variations. Even for a fixed channel, its transfer function may still vary as the temperature or humidity changes. These variations impact the link performance on different levels, hence the adaptation and synchronization loops are needed for compensation.
Figure 1-1: Examples of high-speed link applications. (1) MCH buses connecting CPU with memory and peripherals through north/south bridges. The frontside bus clock can run up to 800 MHz (Pentium 4 HT Extreme Edition). (Courtesy of Jon Stokes) (2) QPI protocol buses connecting CPU with CPU and memory. It operates at 3.2 GHz with double data rate. (Courtesy of Intel) (3) Router and server backplanes for 10 Gb/s applications. (Courtesy of Rambus)

Figure 1-2: Channel loss introduces ISI and limits the achievable signalling speed. As the speed increases, the channel loss is more severe and more ISI shows-up in the time domain, which reduces the receiver signal-to-noise ratio (SNR) and increases the bit-error rate (BER). The channel shown here is 20 inches long.
Figure 1-3: Channel transfer functions vary depending on different sources of uncertainty. (1) Channel variation due to trace routing variation, such as length and via position variations [1]. (2) Channel variation due to manufacturing variation [2]. (3) Channel variation due to environmental changes [3].
A conventional link transceiver is presented in Figure 1-4 (a). At the transmitter (Tx) end, pre-emphasis [4–8] has been applied to reduce the ISI by attenuating the low frequency signal content and boosting the high frequency content. Hence, the overall system transfer function becomes flatter, as shown in Figure 1-5. According to the Nyquist Criterion\(^1\), the system suffers less from the ISI [9,10].

With two samples per symbol (data sample \(d_n\) and edge sample \(e_n\)), the conventional link receiver includes a voltage recovery loop (or equalization loop) [11–14] and a timing recovery loop (CDR loop or synchronization loop) [15–21]. The voltage recovery loop uses the data sample \(d_n\) to recover the transmitted bits. To maximize the output eye opening, \(d_n\) should be well aligned with the input maximal eye opening phase. A drift from that phase will result in sub-optimal performance or even bit errors. In other words, the voltage recovery loop is sensitive to the sampling phase and requires a precise timing, which is provided by the timing recovery loop. For the popular bang-bang CDR design [6, 22], by comparing the signs of the data sample \(d_n\) and edge sample \(e_n\), the edge clock (for \(e_n\) sampling) is adjusted dynamically to lock to the input zero crossings, and the data clock (for \(d_n\) sampling) is derived from the edge clock by a 0.5 symbol-time (or unit interval, UI) delay. These two loops work with different performance metrics: vertical eye opening amplitude and edge tracking capability.

This conventional receiver architecture operates reasonably well when the input eye is open and symmetric. A low rate eye diagram is shown in Figure 1-4, where the edge moves in a small range with a relatively steep slope. Hence, these edges can be tracked easily. Since the eye is symmetric and the maximal eye open phase is 0.5 UI away from the edge mean, the derived data clock is optimal.

However, when the input eye is closed and asymmetric, which is much more common for high-speed links that operate at rates much higher than the channel bandwidth [8, 13, 23],

\(^1\)Nyquist’s Criterion [10]
The necessary and sufficient condition for \(x(t)\) to satisfy:

\[
x(nT) = \begin{cases} 
1 & n = 0 \\
0 & n \neq 0
\end{cases}
\]

is that its Fourier transform \(X(f)\) satisfies:

\[
\sum_{m=-\infty}^{\infty} X(f + m/T) = T,
\]

where \(T\) is the symbol period.
Figure 1-4: (a) A conventional high-speed link with independent voltage recovery loop (EQ) and timing recovery loop (CDR) at receiver (Rx) end. (b) The proposed high-speed link with an FSE receiver, which unifies EQ and CDR.

Figure 1-5: Equalization principle. The equalizer flattens the system (channel & equalizer) transfer function.
the performance degrades quickly. For the closed eye case, the edge moves in a range larger than 1 UI with a relatively gentle slope, raising the tracking error. Even if perfect edge timing recovery is assumed, the derived data clock may still be non-optimal because the distance between the edge and the maximal eye opening phase is no longer 0.5 UI due to the asymmetric eye shape.

This problem is even more severe for those receivers with a loop-unrolled decision-feedback equalizer (DFE) [6, 24]. A loop-unrolled DFE calculates the ISI introduced by the previous bits and biases the data slicer's offset accordingly. It is an effective way to compensate for part of the ISI without explicit signal subtraction from the channel output. However, it changes the transmitter equalization strategy: instead of zeroing all the ISI, the transmitter allows a certain number of post-cursors because they are correctable at the receiver end. By zeroing the DFE-uncorrectable cursors only, the pre-emphasis attenuates the signal less, which improves the overall performance. Such a transmitter-receiver strategy is effective for voltage recovery. However, it alters the eye diagram greatly and impacts the timing recovery performance. In return, the data clock is disturbed and the voltage recovery performance is degraded.

An eye diagram of a 1-tap loop-unrolled DFE is given in Figure 1-4. It consists of two eye diagrams (dependent on the previous bit) offset by 2α, where α is the DFE cursor amplitude. Since the zero-crossings are also pulled up or down by some amount (generally speaking, it is not α), it is quite challenging to figure out what new level the edge slicer should be biased at and what phase the edge clock should be locked to. Mostly, the recovered edge timing is ambiguous and the derived data timing is sub-optimal. Even for the direct DFE systems [25], the feedback signal may disturb the zero crossings of the edges, impact the recovered timing, and degrade the sampled eye openings. Solutions, such as separate DFE correction for edge samples, have been introduced at the cost of doubled receiver complexity [26].

In all, the voltage and timing recovery loops, which are originally designed to work separately with different performance metrics, are actually highly coupled. This discrepancy results in sub-optimal performance or even a closed eye. Thus, a combination of the voltage/timing recovery loops is desired to do joint equalization and synchronization, which is the goal of this thesis.
1.2 Introduction to fractionally spaced equalization

We propose to use fractionally spaced equalization (FSE) to do both voltage and phase recovery. The proposed FSE receiver for mesochronous systems\textsuperscript{2} is demonstrated in Figure 1-4 (b). Because the FSE is able to recover the voltage at any sampling phase through the interpolation among the 0.5 UI spaced samples, it recovers the phase implicitly, hence does not require a separate timing recovery loop.

An FSE is a finite-impulse-response (FIR) linear receive equalizer, sharing the same FIR architecture as the symbol-spaced equalizer, which is shown in Figure 1-6. The channel outputs are sampled and all the values are multiplied with the filter coefficients. The multiplication outputs are summed up as the equalizer's output, which feeds the slicer. For the symbol-spaced equalizer, the input samples $d_1, d_2, \cdots d_n$ are spaced at 1 UI. To maximize the eye opening, the timing recovery loop is required to align these values to the maximal eye opening positions. For the FSE, the input samples $d_1, d_2, \cdots d_n$ are spaced at 0.5 UI. By tuning the tap weights $W$, this FIR filter is able to interpolate among the samples to achieve almost the same vertical eye opening at any sampling phase. Therefore, only a frequency lock is required for an FSE receiver because the residual phase can be compensated by the FSE itself. To replace a separate frequency lock loop, we also propose a digitally-controlled bit-skipping algorithm to compensate for the frequency offset in plesiochronous systems.

In theory, the over-sampling ratio should be at least 2 (preferably higher), to minimize the aliasing due to the excess signal bandwidth. However, as this ratio increases, the system

\textsuperscript{2}Mesochronous system: system clocks run at the same frequency but unknown phase at different blocks [11, 27]. For high-speed links, it means that the receiver has exactly the same clock frequency (same source) as the transmitter, but unknown phase.

Another term that is used frequently in this thesis is plesiochronous system, referring to the system with clocks running at different frequencies at different blocks. For high-speed links, it means that the receiver has a different clock frequency from the transmitter.
Figure 1-7: Pulse response based FSE analysis. (1) Analog channel pulse response. (2) 2x over-sampled channel pulse response. (3) 2x over-sampled joint channel and FSE pulse response. (4) Symbol-spaced channel & FSE pulse response, optimized assuming a 1-tap DFE is applied.

becomes more power hungry while the performance improves only marginally, since in links we already operate much above the channel bandwidth. Therefore, we only focus on the 2x over-sampled FSE in this thesis.

The analysis of the 2x over-sampled FSE is similar to the analysis of symbol-spaced equalization (SSE). As shown in Figure 1-7, the continuous channel pulse response is sampled at 2x rate and the FSE input pulse response is generated. After convolving with the FSE taps, a 2x (sampling) rate pulse response of the channel and an FSE is derived. Since the data slicer operates at the symbol rate, another 2x down-sampling is taken to produce the channel and FSE pulse response at the symbol rate. If there is no DFE after the FSE, ignoring the random noise, the FSE taps should be tuned to minimize the ISI that appears in the symbol-spaced pulse response. If a DFE is applied, the FSE taps should be tuned to allow the DFE taps, while minimizing the other ISI cursors.

Though named as 2x over-sampled equalization, an FSE-based receiver samples at the same rate as a symbol-spaced equalization (SSE) receiver. The difference is that the SSE
receiver has data-type samples $d_n$ and edge-type samples $e_n$. The latter is quantized into only one bit to recover timing in the widely adopted bang-bang CDR designs. For the FSE, the samples are processed identically: the information from all the samples is used to create a symbol-rate output for the slicer. Though effectively the FSE processes the input samples with a higher resolution, it does not require a timing recovery loop, hence can potentially improve the BER or energy-efficiency in comparison with conventional SSE receivers.

### 1.3 Thesis contributions

There are two main contributions in this thesis: system level investigation and circuit innovations. At the system level, it is the first report of using an FSE to do joint equalization and synchronization in high-speed links. The challenges of applying an FSE for both mesochronous systems and plesiochronous systems are studied and solved. At the circuit level, inspired by the development of zero-crossing-based circuits (ZCBC) [28], a voltage-time conversion based FIR tap implementation is developed. In comparison with the conventional current mode logic (CML) based designs, it achieves higher rates and better linearity with less power. Potentially, this technique can be used for a broad range of high-speed mixed-signal filtering and signal-processing applications.

#### 1.3.1 System level analysis

Existing FSE receiver designs focus only on the voltage recovery function [29, 30]. This thesis is the first to use an FSE to recover the signals at unknown phases. As an oversampling receiver FIR filter, the FSE is able to reconstruct the desired decision voltages regardless of the sampling phase. This aggregates the equalization and phase interpolation functions into a single structure, tuned by a common performance criterion: the output vertical eye opening. A 2x over-sampled fractionally spaced equalizer samples twice per symbol, synchronizes the phase, and equalizes the channel by filtering these samples. In comparison with the SSE receivers, it requires the same number of samples per symbol but with a higher resolution (5-bit for adaptation, compared to the 1-bit resolution requirement with edge samples for the SSE). Since the FSE unifies the system performance metric and avoids a separate timing recovery loop, it can potentially achieve a higher vertical eye opening with less power dissipation.

To better develop design intuition, we adopted a simple model to compare the FSE with the SSE. This comparison shows that an FSE is immune to the sampling phase while an
SSE is not, which is consistent with our simulations and experiments. However, the FSE coefficients must be dynamically adjusted to track the changes in the channel characteristics that impact both voltage and timing domains. The standard SSLMS algorithm is widely used in links [6], and is investigated in this thesis for its low implementation cost, since it requires only one bit of quantized information from the channel and FSE outputs. We have shown that because of the quantization noise, the SSLMS algorithm converges to a balanced solution, which is different from the minimum mean square error (MMSE) solution given by the least-means-square (LMS) algorithm. For the worst quantization noise case, it even diverges. To overcome the divergence issue, a modified sign-sign least-mean-square (M-SSLMS) algorithm is developed. Without additional hardware, it introduces one more quantization level to the input signals, reduces the quantization noise, and hence keeps the adaptation converging. The adaptive algorithm development enables the use of FSE in mesochronous link systems.

The possibility of applying the FSE in plesiochronous systems is also studied and we developed a digitally-controlled bit-skipping algorithm for frequency offset compensation. By monitoring the phase movement through the input sign correlation, the algorithm selects between the FSE outputs adaptively and keeps the residual phase within a fraction of 1 UI, which can be compensated by the coefficient adaptation loop. As an alternative approach, the FSE with a conventional bang-bang CDR loop is also analyzed. Immune to the sampling phase, the FSE adds one more degree of freedom into the original timing-recovery design space and is shown to outperform the SSE & CDR systems under various jitter conditions.

1.3.2 Circuit level implementation

At the circuit level, we extend the voltage-time conversion technique from analog-to-digital converter (ADC) design [28] to build filter taps for high-speed links. In this new technique, the channel voltage is first converted into time domain and then converted back into voltage domain. Both conversions are completed by linear current integration, with all integration currents independent of the channel voltages, avoiding the non-linear voltage-current transformation through input device transconductance. Two voltage-to-time converters and one time-to-voltage converter design styles are presented in this thesis. In contrast with the conventional CML-based implementations, the voltage-time conversion based multiplier shows better power-linearity performance.
Based on different voltage-to-time converter designs, two proof-of-concept FSE implementations have been fabricated in a 90-nm CMOS process. The first implementation is a 2-way interleaved 2-tap FSE, operating at 4 Gb/s, with 2 pJ/bit energy-efficiency, and 4.3 bits of linearity, showing robustness to the sampling phase. The second implementation is designed as a 4-way interleaved 2-tap FSE with a 1-tap DFE, which operates at 6.25 Gb/s, with 3.6 pJ/bit energy-efficiency, and over 4.0 bits of linearity, demonstrating the convergence of the \textit{M-SSLMS} algorithm. A third implementation has been designed with on-chip coefficient adaptation and bit-skipping schemes for plesiochronous systems.

1.4 Thesis organization

This chapter gives an overview of the high-speed link channels and the conventional transceiver designs. The coupling issue between the voltage and timing recovery loops is discussed, revealing the motivation of using the FSE to combine them into one block with a unified performance metric.

In Chapter 2, a detailed system-level study of the FSE is presented. An analytical comparison between the FSE and SSE shows that through tap tuning, a 2-tap FSE can cancel all the ISI from an RC dominant channel at any phase, while the SSE cannot. Since the optimal FSE taps are functions of the sampling phase, the FSE adaptation is further studied. The conventional low-cost low-complexity \textit{SSLMS} algorithm is shown to converge differently from the \textit{LMS} algorithm due to the quantization noise, and even diverges in the worst case. As a solution, the \textit{M-SSLMS} algorithm is proposed to reduce the quantization noise and make the adaptation converge, which enables FSE application for mesochronous systems. To overcome the frequency offset in plesiochronous systems, a bit-skipping scheme is introduced, which keeps the residual phase bounded by dynamically selecting between the two outputs, and solves the frequency offset issue. As an alternative approach, the FSE combined with a conventional CDR system is also studied. It is shown to outperform the SSE with CDR system under various jitter conditions.

In Chapter 3, the FSE tap circuit design principle is investigated. The voltage-to-time (V2T) conversion and time-to-voltage (T2V) conversion, which are the two key steps of voltage-time (VT) conversion based FSE implementation, are analyzed. Both the \textit{VTH}-based (i.e. NMOS-based) V2T converter and the \textit{VM}-based (i.e. inverter-based) V2T converter are investigated. Some new circuit techniques are proposed for further perfor-
mance improvement in future designs. A comparison with the CML-based implementation is conducted in this chapter, showing the advantages of voltage-time (VT) conversion based circuits.

In Chapter 4, for mesochronous systems, two generations of 2-tap FSE implementation and the corresponding measurements are reported. With different V2T converter designs, these VT conversion based implementations show better power-linearity performance than the CML-based designs. These implementations also demonstrate the immunity against sampling phase and the convergence of the $M$-SSLMS adaptive algorithm. The FSE design for plesiochronous systems is further discussed in this chapter, and the estimated costs of power and area for the on-chip FSE back-end are presented.

Chapter 5 concludes this thesis and puts forward some thoughts for future research.
Chapter 2

FSE System Study

The FSE is studied at the system level in this chapter. To demonstrate the FSE's phase immunity, the performance of the FSE is compared with the SSE both analytically and numerically in Section 2.1. It is verified that by tuning the coefficients, an FSE can generate a flat eye opening regardless of the sampling phase.

However, to tune the coefficients automatically, an adaptive algorithm is required. The FSE adaptation is covered in Section 2.2. The conventional SSLMS algorithm is suitable for the FSE receivers, because it needs one bit quantization on both input and error samples. However, due to the large quantization noise, it converges to a balanced solution, which can be different from the MMSE solution. In the worst sampling phase, this large quantization noise can make the balanced solution optimum quite flat, causing the algorithm to keep increasing the coefficients until they saturate, i.e. the algorithm diverges. To overcome this problem, we have developed the M-SSLMS algorithm by introducing another quantization level through signal conditioning at no additional cost, which enables FSE application for mesochronous systems.

We also study the application of the FSE for plesiochronous systems. Section 2.3 presents a digitally-controlled bit-skipping scheme, which compensates for the frequency offset without a conventional CDR loop. We also investigate the benefits that the FSE can bring to a receiver with a conventional CDR loop in Section 2.4. We show that this combination outperforms the SSE and CDR loop system under various jitter situations.
2.1 Comparison of FSE and SSE

To show the advantages of the FSE, it is compared with the SSE under different timing conditions in this section. In order to build intuition about the difference, a short 2-tap FSE is first studied analytically, in contrast with a 2-tap SSE under an RC dominant channel. Then, a numerical verification is conducted with a measured channel and a 4-tap FSE/SSE.

2.1.1 Analytical Comparison

Comparison setup

Without loss of generality, an RC-dominant channel model is adopted to simplify the comparison. The corresponding normalized channel pulse response is shown in Figure 2-1. The rising and falling edges are exponential with parameters $\alpha$ and $\beta$ respectively. The analog channel pulse response is:

$$ p(t) = \begin{cases} 
0 & t \in (-\infty, 0) \\
1 - e^{-\alpha t} & t \in [0, T_0] \\
1 - e^{-\alpha T_0} & t \in (T_0, +\infty), 
\end{cases} $$

where $T_0$ is the symbol period. As a simplified model, it covers a broad range of scenarios. For example, the open-eye case can be modeled with a combination of large $\alpha$ and large $\beta$, and the closed eye case can be modeled with a combination of small $\alpha$ and small $\beta$.

To keep the derivations tractable, a 2-tap SSE and a 2-tap FSE are investigated. The FSE’s output is down-sampled by 2x, and only the pulse response observed by the slicer is studied. The same method can be extended to the study of arbitrarily long equalizers.

Comparison at the optimal phase

For easier understanding, the comparison starts from the optimal phase case, where the peak of the analog pulse response is sampled as one cursor of the symbol-spaced pulse response observed by the SSE. As shown in Figure 2-1, the resulting discrete pulse response $P_{SSE}(n)$ is exponential with parameter $\beta$:

$$ P_{SSE}(n) = e^{-n\beta T_0}, \quad \text{where } n = 0, 1, 2, \ldots $$
Figure 2-1: FSE and SSE achieve the same equalization performance at the optimal phase.

A 2-tap SSE \( w_{SSE}(n) = w_1 \delta(n) + w_2 \delta(n-1) \) can equalize this pulse response with tap settings \( w_1 = 1, w_2 = -e^{-\alpha T_0} \) because the convolution of \( P_{SSE}(n) \) and \( w_{SSE}(n) \) is:

\[
P_{SSE}(n) \ast w_{SSE}(n) = \delta(n).
\]

Thus, the SSE equalizes the channel completely, removing all ISI.

At the same phase, the 2x over-sampled pulse response \( P_{FSE}(n) \) observed by the FSE is:

\[
P_{FSE}(n) = \begin{cases} 
0 & n = 0 \\
\frac{1}{1 + e^{-\alpha T_0/2}} & n = 1 \\
\frac{e^{-(n-1)\beta T_0/2}}{1 - e^{-\alpha T_0/2}} & n = 2, 3, 4, \ldots.
\end{cases}
\]

A 2-tap FSE \( w_{FSE}(n) = w_1 \delta(n) + w_2 \delta(n-1) \) can equalize it with tap settings \( w_1 = \gamma, w_2 = -\gamma e^{-\beta T_0/2} \), where \( \gamma = 1 + e^{-\alpha T_0/2} \). The convolution of the channel 2x over-sampled pulse response and the FSE filter pulse response is:

\[
P_{FSE}(n) \ast w_{FSE}(n) = \begin{cases} 
1 + e^{-\alpha T_0/2} - e^{-\beta T_0/2} & n = 0, 2, 4, \ldots \text{ even} \\
\delta(n-1) & n = 1, 3, 5, \ldots \text{ odd}.
\end{cases}
\]

Although the FSE can generate two outputs per symbol, the slicer needs only one to make a symbol decision. In this comparison, the odd output is selected for the slicer and the corresponding channel with FSE symbol-spaced pulse response is \( \delta(n), n = 0, 1, 2, \ldots. \) Thus, the FSE cancels all the ISI, showing the same voltage recovery performance as the SSE.

Interestingly, at this optimal phase, the FSE can also generate the main cursor of value
Figure 2-2: SSE is sensitive to the sampling phase: there always exists non-zero residual ISI no matter what equalization optimization strategy is taken. The FSE is immune to the phase movement: zero residual ISI for all sampling phases.

1 at the even output with a combination of the first two cursors. There is also no residual ISI, but the resulting dominant taps are offset by 0.5 UI. Thus, the FSE provides one more degree of freedom here.

Comparison at non-optimal phases

The SSE's performance degrades significantly at other sampling phases, while the FSE's performance is robust. A scenario is shown in Figure 2-2. Given a phase offset $\epsilon$, the pulse response observed by the SSE is:

$$P_{SSE}(n) = \begin{cases} 
1 - e^{-\alpha \epsilon} & n = 0 \\
\frac{1 - e^{-\alpha T_0}}{1 + e^{-\beta (n+\epsilon-1) T_0}} & n = 1, 2, 3, \ldots
\end{cases}$$

The SSE taps can be tuned according to several criteria: zero-forcing the pre-cursor ISI, zero-forcing the post-cursor ISI, or minimizing the ISI in the mean-square sense. However, in all cases the residual ISI is nonzero due to the SSE's limited ability to handle the aliasing. For highly lossy channels (small $\alpha$ and $\beta$), the residual ISI may close the eye opening, causing bit errors.

At the same sampling phase, the 2x over-sampled channel pulse response observed by
the FSE is:

$$P_{FSE}(n) = \begin{cases} 
1 - e^{-\alpha(nT_0/2+\epsilon)} & n = 0, 1 \\
\frac{1 - e^{-\alpha T_0}}{1 - e^{-\alpha T_0/2+\epsilon}} & n = 2, 3, 4, \ldots 
\end{cases}$$

The FSE can generate the desired main cursor of value 1 and zero-out the residual ISI. A calculation shows that the corresponding coefficient settings are \( w_1 = \gamma, w_2 = -\gamma e^{-\beta T_0/2}, \)
where

$$\gamma = \frac{1 - e^{-\alpha T_0}}{1 - e^{-\alpha T_0/2+\epsilon} - e^{-\beta T_0/2} (1 - e^{-\alpha \epsilon})}.$$ 

The convolution of the channel pulse response and the FSE filter \( w_{FSE}(n) = w_1 \delta(n) + w_2 \delta(n-1) \) is:

$$P_{FSE}(n) \ast w_{FSE}(n) = \begin{cases} 
\gamma((1 - e^{-\alpha \epsilon}) \delta(n) + \gamma \left(\frac{e^{-\beta \epsilon} (1 - e^{-\alpha T_0}) - (1 - e^{-\alpha (T_0/2+\epsilon)})}{1 - e^{-\alpha T_0}}\right)) \delta(n-2) & n = 0, 2, 4, \ldots \\
\delta(n-1) & n = 1, 3, 5, \ldots.
\end{cases}$$

Thus, the FSE zeros all the ISI with the odd output \((n = 1, 3, \ldots)\), the same as the optimal phase case. This example shows that the FSE is immune to the sampling phase while the SSE is not.

**Comparison summary**

This comparison reveals several interesting properties of the FSE. Firstly, though the period covered by the 2-tap FSE is only 0.5 UI, half of the 2-tap SSE, it provides equal (at the optimal phase) or even better (at a non-optimal phase) equalization performance\(^1\). Secondly, the sampling phase offset \( \epsilon \) degrades the SSE’s performance, while showing no impact on the FSE’s performance. Thirdly, by generating the dominant cursor with the first two cursors of the 2x over-sampled channel pulse response, an FSE is effective in canceling the pre-cursor ISI. Obviously, this may lead to larger equalizer coefficients, i.e., a higher gain requirement at the receiver end. But this effect is generally mild. As shown in Figure 2-2, for the worst channels with \( \alpha \approx 0 \) and \( \beta \approx 0 \), the largest coefficient \( \gamma \) is less than 2. On the other hand, for the SSE case, if the dominant cursor is just derived from the first cursor of the (symbol-spaced) channel pulse response to avoid the pre-cursor ISI, the SSE coefficient \( w_1 \) has to be \( \frac{1 - e^{-\alpha T_0}}{1 - e^{-\alpha \epsilon}} \), which becomes impractically large when \( \epsilon \) approaches

\(^1\)A similar result has been reported in the communications community [9].
zero. One side effect of the large coefficients is the noise enhancement issue [9]. Given a
fixed main tap\(^2\), the output signal power is fixed, while the output noise power increases
as the equalizer coefficients grow in amplitude, worsening the equalizer output SNR. Thus,
there exists a trade-off between the ISI cancellation performance and the noise enhancement
effect. For ISI dominant channels, which fit most of the high-speed link applications, the
FSE coefficients shown in Figure 2-1 and Figure 2-2 are optimal in terms of minimizing the
BER.

The performance difference against the sampling phase between the SSE and FSE can
be explained in the frequency domain. On these link channels, symbol-spaced sampling
cannot avoid aliasing. Any offset from the optimal phase will introduce an extra linear
phase to the Fourier transform of the sampled values, which exacerbates the aliasing result,
i.e. contaminates the flatness of the folded spectrum. As shown in Figure 2-3, processing the
signal after the aliasing has occurred, the SSE has to flatten the aliased channel response,
which leads to potentially large ISI and poor performance. On the other hand, aliasing is
significantly mitigated after the 2x over-sampling. The phase offset only appears as an extra
linear phase term in the Fourier transform of the sampled pulse response. Furthermore,
there is no aliasing during the FSE filtering, and the aliasing happens only when the FSE’s
output is down-sampled by a factor 2. Since the FSE filtering happens before the aliasing,
it can work with aliasing actively: instead of flattening the spectrum of the FSE’s output
signal, the FSE can “adapt” to different amounts of aliasing by tuning the coefficients
to flatten the folded spectrum after the down-sampler. Thus, the FSE achieves better

\(^2\)It is 1 in this analysis.

Figure 2-3: Aliasing happens before the SSE filtering, and it happens after the FSE filtering.
equalization performance, such as small residual ISI.

In summary of this example, the SSE relies on the synchronization loop to produce an optimal sampling phase, so that the samples are well aligned with the maximal eye opening position. Any phase drift may result in performance degradation or even eye closure. On the other hand, a properly tuned FSE can keep the receiver output well equalized at any sampling phase.

### 2.1.2 Numerical comparison

It is difficult to do a similar analytical comparison with measured channels and arbitrarily long equalizers under various sampling phase conditions. To verify the analytical statements based on the simplified model, a 4-tap FSE at 6.0 Gb/s is studied numerically with a measured 20-inch legacy backplane channel. The channel frequency response and the eye diagram are shown in Figure 2-4(1-2). There is a 17 dB loss at 3 GHz Nyquist frequency, and the eye is completely closed. The Rx clock phase relative to Tx clock phase is swept within 1 UI. The FSE coefficients are calculated by convex programming for each phase\(^3\), and the corresponding worst case eye openings are shown in Figure 2-4(3-4). The worst case eye opening is flat with ±5% variation over all the phases, while for the 4-tap SSE, the worst case eye opening varies on a much larger scale and even closes in some phase range. The properties of the optimal coefficients of the SSE/FSE over 1 UI phase offset range are given in Table 2.1. In comparison with the SSE, the FSE requires a wider amplitude range. However, the extra gain is no more than 2, which is consistent with the analytical conclusion on the 2-tap FSE/SSE case.

<table>
<thead>
<tr>
<th></th>
<th>Maximal coefficient</th>
<th>Maximal (L^1) norm(^4)</th>
<th>Maximal (L^2) norm(^5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSE</td>
<td>1.6</td>
<td>4.0</td>
<td>2.2</td>
</tr>
<tr>
<td>FSE</td>
<td>3.1</td>
<td>7.0</td>
<td>4.2</td>
</tr>
</tbody>
</table>

This simulation illustrates that the FSE can equalize the channel at any sampling phase with almost identical worst-case eye opening. Obviously, the optimal coefficients vary as the sampling phase varies, and an adaptation scheme is needed for any practical implementation.

\(^3\)The goal is to maximize the worst-case equalized vertical eye opening \([31,32]\].

\(^4\)The \(L^1\) norm of the optimal coefficients \(\{w_1, w_2, \cdots w_n\}\) is defined as: \(\sum_{k=1}^{n} |w_k|\).

\(^5\)The \(L^2\) norm of the optimal coefficients \(\{w_1, w_2, \cdots w_n\}\) is defined as: \(\sqrt{\sum_{k=1}^{n} |w_k|^2}\).
Figure 2-4: A 4-tap FSE is robust to the variation in sampling phase. (1) Channel transfer function, 17 dB attenuation at 3.0 GHz (Nyquist frequency). (2) Eye diagram at 6.0 Gb/s with no equalization. (3) Optimal tap weights for each phase offset. (4) FSE and SSE worst case eye openings given the optimal tap weights for each phase. The fluctuation of the FSE eye opening is less than ±5%. The SSE eye varies on a much larger scale and even closes at some phases.
This adaptive equalization loop can replace both the coefficient adaptation loop and the clock-data recovery (phase-synchronization) loop needed by the SSE based receivers for mesochronous systems.

2.2 FSE adaptation

2.2.1 FSE adaptive algorithm overview

To demonstrate the phase robustness of the FSE, an adaptation engine is needed to tune the taps dynamically, because the optimal coefficients depend on the sampling phase, which is at best (unknown) static or slowly varying in time. An adaptation structure is shown in Figure 2-5. It combines the information of the input and error signals to adjust the equalizer coefficients $W$ accordingly [33].

As one of the most widely used adaptive algorithm, the LMS algorithm [34-36] and its variants, including sign-LMS algorithm, sign-regressor LMS algorithm and sign-sign LMS (SSLMS) algorithm, are studied in the context of the FSE adaptation. The underlying principle in all these algorithms is to iteratively change the tap weights by a small amount $\delta$ as a function of the received signal vector $d = [d_1, d_2, \ldots, d_n]^T$ and the corresponding error $e$, which orthogonalizes the error from $d$ and leads to a stochastic approximation of the mean-square Wiener filter solution. In peak-power constrained high-speed links, this solution is suboptimal compared to the convex programming solution (Figure 2-4 (3)), but at the slight loss in performance, which is well justified by the simplicity of the iterative adaptation loop.

The adaptation equations from update instant $k$ to $k + 1$ for these algorithms are given in the order of decreasing hardware complexity:

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Update Equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>LMS</td>
<td>$W^{(k+1)} = W^{(k)} + \alpha d^{(k)} e^{(k)}$</td>
</tr>
<tr>
<td>Sign LMS</td>
<td>$W^{(k+1)} = W^{(k)} + \alpha d^{(k)} \text{sgn} \left( e^{(k)} \right)$</td>
</tr>
<tr>
<td>Sign regressor LMS</td>
<td>$W^{(k+1)} = W^{(k)} + \alpha \text{sgn} \left( d^{(k)} \right) e^{(k)}$</td>
</tr>
<tr>
<td>Sign-sign LMS</td>
<td>$W^{(k+1)} = W^{(k)} + \alpha \text{sgn} \left( d^{(k)} \right) \text{sgn} \left( e^{(k)} \right)$</td>
</tr>
</tbody>
</table>

Here, $\alpha$ is the step size, a small positive number, and $W^{(k)}$ are the corresponding equalizer coefficients at update instant $k$.

The LMS algorithm converges on average to the minimum mean square error (MMSE) solution [37]. However, it is difficult to implement at multi-GHz frequency because of the
Figure 2-5: Block diagram for the receiver equalizer adaptation. For the LMS algorithm, both the input and error quantizers are with infinite resolution. For the sign LMS algorithm, the input quantizer has infinite resolution while the error sign quantizer is 1-bit. For the sign-regressor LMS algorithm, the input quantizer is 1-bit while the error quantizer is with infinite resolution. For the SSLMS algorithm, both the input and error quantizers are 1-bit.

relatively high resolution requirements in both $d$ and $e$. Sign LMS algorithm and sign-regressor LMS algorithm simplify the LMS algorithm by quantizing either error $e$ or input $d$ into one bit. Thus, they reduce hardware complexity at the cost of more quantization error, which impacts the performance, such as the increase in steady-state error. However, they are still too complicated to be deployed for links because of the relatively high resolution requirement in either $d$ or $e$. Sign-sign LMS algorithm quantizes both input $d$ and error $e$ into one bit, which is practical for high-speed implementations. Although it can leverage the simple comparator circuits that already exist in the high-speed link receivers, the sign-sign LMS algorithm introduces large quantization error, leading to a higher averaging requirement and a lower convergence speed. The adaptation equation for a block-window $N$-averaging sign-sign LMS algorithm is:

$$ W^{(k+1)} = W^{(k)} + \alpha \left( \sum_{i=0}^{N-1} \text{sgn} \left( d^{(kN+i)} \right) \text{sgn} \left( e^{(kN+i)} \right) \right) , \ k = 0, 1, 2, \cdots \ 	ag{2.1} $$

2.2.2 SSLMS algorithm convergence

By comparing the coefficient evolutions, we find that the steady-state of the SSLMS adaptive algorithm is different from that of the LMS algorithm. Based on a 2-tap FSE, the evolution traces are shown in Figure 2-6(1), where the $x$-axis and $y$-axis correspond to the
tap coefficients $w_1$ and $w_2$ respectively. For each FSE coefficient setting, the corresponding mean square error from a preset level (500 mV) is shown as the background, indicating that the MMSE solution is at $W = [w_1, w_2]^T = [0.75, -0.05]^T$. Given the same initial conditions, the LMS algorithm converges to the MMSE solution while the SSLMS algorithm converges to a different solution.

To study the steady-state of the SSLMS algorithm, the balance function is defined. Assuming that the equalizer tap weights $W^{(k)}$ at $kT$ are $[w_1^{(k)}, w_2^{(k)}]^T$ (considering a 2-tap FSE with 1-averaging scheme), the balance function $f(W^{(k)})$ is defined as:

$$f(W^{(k)}) = f(w_1^{(k)}) \times f(w_2^{(k)})$$

where

$$f(w_1^{(k)}) = \max \left( \frac{P\left(e^{(k)}d_1^{(k)} > 0\right)}{P\left(e^{(k)}d_1^{(k)} < 0\right)}, \frac{P\left(e^{(k)}d_1^{(k)} > 0\right)}{P\left(e^{(k)}d_1^{(k)} < 0\right)} \right)$$

$$f(w_2^{(k)}) = \max \left( \frac{P\left(e^{(k)}d_2^{(k)} > 0\right)}{P\left(e^{(k)}d_2^{(k)} < 0\right)}, \frac{P\left(e^{(k)}d_2^{(k)} > 0\right)}{P\left(e^{(k)}d_2^{(k)} < 0\right)} \right)$$

$P(\cdot)$ is the probability function. For example, $P\left(e^{(k)}d_1^{(k)} > 0\right)$ denotes the probability that the product of $e^{(k)}$ and $d_1^{(k)}$ is positive. This definition can be extended to the equalizers of any length. It is obvious that:

$$f(W^{(k)}) \geq 1.$$

By examining the SSLMS adaptation equation in Table 2.2, we discover that the SSLMS algorithm always converges to a balanced solution $W^{(\infty)}$ satisfying:

$$f(W^{(\infty)}) = 1.$$

Equivalently, the balanced solution $W^{(\infty)}$ satisfies:

$$P\left(\text{sgn}(d^{(\infty)})\text{sgn}(e^{(\infty)}) = 1\right) = P\left(\text{sgn}(d^{(\infty)})\text{sgn}(e^{(\infty)}) = -1\right) = 0.5. \quad (2.2)$$

Equation 2.2 shows that at the balanced solution $W^{(\infty)}$, the probability for coefficients to increase is equal to the probability to decrease, so that the coefficients can stay at this solution on average. This convergence is verified by the simulations. As shown in Figure 2-
Figure 2-6: Convergence difference between the LMS and the SSLMS algorithms with the MSE contour background (1) and the balance function contour background (2). Given the same initial state \( (w_1 = 1, w_2 = 0) \), the LMS algorithm converges to the MMSE solution \( (w_1 = 0.75, w_2 = -0.05) \), while the SSLMS algorithm converges to the balanced solution \( (w_1 = 1.1, w_2 = -0.45) \).

6(2), for each coefficient combination \((w_1, w_2)\), the balance function is evaluated and plotted as the background. The SSLMS algorithm converges to the balanced solution, which is shown as dark blue in the plot.

The SSLMS algorithm converges to a balanced solution instead of the MMSE solution, which results in sub-optimal performance. Further study demonstrates that the averaging scheme (shown in Equation 2.1) does not change the steady-state. Therefore, it is incapable of reducing the gap between the balanced and MMSE solutions. One effective method is to increase the quantization resolution of the input signal \( d \). Following this idea, we developed the \( M \)-SSLMS algorithm, which increases the effective input signal resolution through signal conditioning, with no additional hardware cost.

2.2.3 \( M \)-SSLMS algorithm for convergent FSE adaptation

The large quantization noise not only alters the steady-state solution but also puts the SSLMS algorithm in danger of divergence [38, 39]. According to the system-level analysis and simulations, the SSLMS algorithm converges to a balanced solution, which is usually close to the MMSE solution. But if there is one sample near the zero-crossing phase (for instance, \( d_2 \) in Figure 2-7), divergence may happen. For such a scenario, the samples have 50% probability to be nearly zero, assuming that the incoming bits are independent and equally distributed between \{-1, +1\}. Quantizing such a sample into two levels \{-1, +1\}
Figure 2-7: For a 3-tap FSE, sign-sign LMS algorithm diverges when there are samples close to the zero-crossing points ($d_2$ in this figure, which is aligned with tap weight $w_2$). The corresponding eye is closed.

Figure 2-8: An illustration of the M-SSLMS adaptive algorithm. Signs of the samples are applied to tune the FSE taps. By detecting the absence of the edge transition, the signs of sample $d_0[n]$'s neighbors, i.e. sign ($d_1[n-1]$) and sign ($d_1[n]$), provide indications whether sign ($d_0[n]$) is reliable or not. The M-SSLMS algorithm adapts only with those reliable signs.
introduces the worst case quantization error, and leads to divergence, as shown in Figure 2-7. To reduce the quantization error and approach the LMS algorithm, we developed a modified sign-sign LMS algorithm, which is shown in Figure 2-8: a tap $w_k$ is updated if and only if the corresponding sample $d_k$ has the same sign as its two neighbors ($d_{k-1}$ & $d_{k+1}$). In other words, a tap is updated only on those input values with high amplitude, which are still reliable after quantization. For those with low amplitude, the tap is updated with 0 equivalently. Thus, $d_k$ is effectively quantized into three levels $\{-1, 0, +1\}$, and the quantization error is reduced. The $M$-SSLMS algorithm convergence is demonstrated in Figure 2-9. Further simulations demonstrate that the $M$-SSLMS algorithm converges at any sampling phase with performance comparable to the LMS algorithm, as shown in Figure 2-10.

Based on a measured channel, the effective quantization level of the $M$-SSLMS algorithm is shown in Figure 2-11. For the open-eye case, when the sampling phase is close to the zero crossings (around 2/3 UI in Figure 2-11(1)), the samples within $\pm 0.5$ are quantized into 0. The adaptation of the corresponding tap is enabled only when the samples are above 0.5 or below -0.5, where the quantization is less noisy. Thus, through digital signal conditioning, two quantization levels are effectively set at $\pm 0.5$. However, the quantization situation changes as the phase moves. When the sampling phase is at the middle of the eye (around 1/4 UI in Figure 2-11(1)), the samples within [0.3, 0.75] are quantized into 0 or 1 according to the signs of two neighbor samples. The samples are more likely to be quantized to 0 when they are close to 0.3, and quantized into 1 when they are close to 0.75.
Figure 2-10: M-SSLMS algorithm converges at any phase for a 4-tap FSE at 6.25 Gb/s. Top: the steady-state coefficients vs. phase for LMS algorithm, M-SSLMS algorithm, and SSLMS algorithm. Bottom: the corresponding steady-state worst case eye openings vs. phase. The M-SSLMS algorithm performance is close to the conventional LMS algorithm, and the conventional SSLMS algorithm diverges at phases 0.4 UI and 0.8 UI. When the phase is around 0.4 UI, the taps \( w_1 \) and \( w_3 \) are aligned with the data transitions (same as \( w_1 \) in Figure 2-7); when the phase is around 0.8 UI, the taps \( w_2 \) and \( w_4 \) are aligned with the data transitions.

Thus, the quantization noise is still reduced.

For the closed eye case, shown in Figure 2-11(2), the M-SSLMS algorithm is still able to reduce the quantization noise. At phase around 1 UI (close to the edge), it quantizes the values within \([-0.5, 0.5]\) into 0, reducing the error effectively. Even at phase around 1/2 UI (close to the eye), it quantizes a fraction of the values within \([-0.5, 0.5]\) into 0, limiting the quantization error. In all, without additional slicers, the M-SSLMS algorithm effectively introduces another quantization level by digital signal conditioning, which lowers the quantization noise and avoids the divergence issue with the SSLMS algorithm.

The M-SSLMS algorithm is compared with the true 1.5-bit SLMS algorithm\(^6\), at one of the critical sampling phases, showing a similar steady-state. The frequency response curves of the filters with steady-state coefficient values are shown in Figure 2-12. The difference between M-SSLMS algorithm and the true 1.5-bit SLMS algorithm is negligible. In all, as a coarser version of the LMS algorithm, the M-SSLMS algorithm achieves similar performance as the 1.5-bit SLMS algorithm.

\(^6\)Same as the conventional SSLMS algorithm (Equation 2.1), except that the input is quantized into three levels \([-1, 0, +1]\) instead of 2 levels \([-1, +1]\).
Figure 2-11: The M-SSLMS algorithm effective quantization levels vs. phase. The positive inputs above the blue curve are quantized into +1 with probability 1. The negative inputs below the blue curve are quantized into -1 with probability 1. The signals between the red curves are quantized into 0 with probability 1. The positive signals between the blue/red curves are quantized into \{0, +1\} randomly, which is determined by the neighbor sign conditions. Similarly, the negative signals between the blue/red curves are quantized into \{-1, 0\} randomly.

Figure 2-12: The transfer functions of the FSE filter with steady-state coefficient values after the adaptation of the LMS, M-SSLMS, and 1.5-bit SLMS algorithms. The FSE transfer function after the M-SSLMS algorithm adaptation is almost the same as that after the 1.5-bit SLMS algorithm adaptation. Conventional SSLMS algorithm diverges under the same conditions.
The analyses and simulations above demonstrate that the *M-SSLMS* adaptive algorithm can compensate for the phase offset, making the FSE operational in mesochronous systems. In plesiochronous systems, on the other hand, the phase offset moves without limit because of the frequency offset\(^7\). The adaptive algorithm cannot compensate for it alone, because its phase offset compensation range is limited by its total tap length. Therefore, to apply the FSE to plesiochronous systems, a separate frequency offset compensation mechanism is required.

### 2.3 FSE for plesiochronous systems

For plesiochronous systems, Tx and Rx clock frequencies are different by as much as several hundred parts per million (ppm) \([40,41]\) due to various reasons, such as the crystal reference clock frequency mismatch between Tx and Rx boards. Therefore, the receiver sampling phase keeps accumulating without limit, leading to bit errors. In conventional designs, this problem is solved by applying a CDR loop \([16-20]\), which is used to tune the Rx clock to track the Tx clock frequency and phase simultaneously. Since the FSE is able to work regardless of the sampling phase, the conventional CDR loop is no longer a must.

An FSE can generate two outputs per symbol period \([9]\), and these outputs are offset by 0.5 UI. An intuitive idea is to simply switch between them to keep the residual phase small, which can be further compensated by the coefficient adaptation loop. Based on this idea, a digitally-controlled bit-skipping scheme is developed to handle the frequency offset issue in plesiochronous systems.

#### 2.3.1 The need for a frequency offset compensation loop

The coefficient adaptation loop cannot compensate for the frequency offset by itself because the accumulated phase is unbounded. One example is demonstrated in Figure 2-13. At time zero, a 2-tap FSE generates an even output with a combination of samples \(d_0 \& d_1\), and an odd output with a combination of samples \(d_1 \& d_2\). Without loss of generality, the even output is selected as the receiver's output, illustrated in Figure 2-13 (1). Since the samples \(d_0 \& d_1\) carry enough information about the transmitted bit \(b_{T,0}\), the generated bit \(b_{R,0}\) matches \(b_{T,0}\) with probability 1. The Tx-Rx phase\(^8\) between the \(b_{T,0}\) and \(b_{R,0}\) is defined

\(^7\)This is not an issue in mesochronous systems, simply because the FSE recovers the bit close to the input samples by default. Thus, no matter what the nominal value is, the actual phase is always within 1 UI. On the other hand, in the plesiochronous systems, the actual phase is unbounded.

\(^8\)Or Tx-Rx delay, defined as the delay between the recovered bit \(b_{R,k}\) and the corresponding transmitted bit \(b_{T,k}\).
Figure 2-13: The frequency offset impact and the principle of bit-skipping scheme for a 2-tap FSE. (1) Initial state. Samples $d_0$ and $d_1$ are used to generate the FSE's output $b_{Rx,0}$, matching the transmitted bit $b_{Tx,0}$. The Tx-Rx phase between the transmitted $b_{Tx,0}$ and the corresponding recovered bit $b_{Rx,0}$ is defined as 0 UI. (2) Combination of $d_{2n}$ and $d_{2n+1}$ is used to generate $b_{Rx,n}$, matching the transmitted bit $b_{Tx,n}$. The accumulated Tx-Rx phase (between $b_{Tx,n}$ and $b_{Rx,n}$) is 0.5 UI. (3) After running for another $k$ cycles, the Tx-Rx phase becomes $(m - k)$ UI. The combination of samples $d_{2n+2k}$ and $d_{2n+2k+1}$, which are actually close to the transmitted bit $b_{Tx,n+m}$, is used to recover the transmitted bit $b_{Tx,n+k}$. A bit error is generated with around 50% probability when the difference between $k$ and $m$ is large (for example, $k = m + 10$). (4) Even/odd output switching: in the $(n + 1)^{th}$ cycle, the combination of samples $d_{2n+1}$ and $d_{2n+2}$ is used to recover the transmitted bit $b_{Tx,n+1}$. The phase between $b_{Tx,n+1}$ and $b_{Rx,n+1}$ is 0 UI, the same as the initial state.
as 0 UI at this state.

The 2-tap FSE operates at the receiver clock rate and generates one output per sample pair \(d_{2n} \& d_{2n+1}\). However, the channel output operates at the transmitter clock speed and the Tx-Rx phase between \(b_{Tx,n}\) and \(b_{Rx,n}\) moves. One possible scenario is shown in Figure 2-13 (2). The accumulated phase becomes 0.5 UI. However, since the samples \(d_{2n}\) and \(d_{2n+1}\) are still around \(b_{Tx,n}\), the generated bit \(b_{Rx,n}\) is still able to match \(b_{Tx,n}\) with probability 1 through proper coefficient setting.

However, the phase keeps accumulating. As shown in Figure 2-13 (3), the phase between \(b_{Tx,n+k}\) and \(b_{Rx,n+k}\) is \((m - k)\) UI. If \((m - k)\) is large, the generated bit \(b_{Rx,n+k}\) is unable to match \(b_{Tx,n+k}\) properly, simply because the two samples \(d_{2n+2k}\) and \(d_{2n+2k+1}\) carry little information about \(b_{Tx,n+k}\). Thus, the frequency offset leads to bit errors, no matter what the FSE coefficients are.

In all, besides the coefficient adaptation loop, the FSE requires an additional control loop to compensate for the frequency offset.

### 2.3.2 FSE with bit-skipping scheme for plesiochronous systems

A comparison between Figure 2-13 (1) and (2) shows the intuition to compensate for the frequency offset. The Tx-Rx phase accumulates indefinitely due the frequency offset, leading to bit errors. However, as long as the receiver switches between the two outputs of the FSE, the residual phase can be kept within 0.5 UI, which can be compensated further by the coefficient adaptation loop alone.

As demonstrated in Figure 2-13 (4), during the \((n + 1)^{th}\) cycle, instead of sticking to the even output \(b_{Rx,n+1} = b_{even}^{Rx,n+1}\), the receiver switches to the odd output \(b_{Rx,n+1} = b_{odd}^{Rx,n+1}\). The Tx-Rx phase becomes 0 UI and the system is reset to the initial state. The system keeps running without switching any more until the accumulated phase grows to 0.5 UI again. Thus, the residual phase is always between 0 - 0.5 UI, which can be handled by the FSE coefficient adaptation loop itself, hence the frequency offset is compensated completely. The scheme of dynamic selection between FSE's even and odd outputs is called the **bit-skipping** scheme.

However, this scheme needs an indicator to detect the accumulated phase in order to trigger the switching properly. To fully reuse the existing circuit, the input signal sign
Figure 2-14: The correlation of the input signal sign indicates the accumulated phase. The system is at 5.0 Gb/s with a 100 kHz frequency offset.

correlation is applied. As shown in Figure 2-14, given a 100 kHz frequency offset, the input sign correlation is periodic with period 10 \( \mu s \). It reflects the residual phase information, hence can serve as the indicator for switching so that the FSE always operates in the desired phase range.

This bit-skipping scheme can operate seamlessly especially with the long-tap-length FSE, for which the eyes with both the even and odd outputs are flat and open while the Tx-Rx phase accumulates for several symbol-times. During such a period, the even/odd output switching can happen at any time to compensate for the accumulated phase with almost no penalty, since it is equivalently a re-selection between two well-behaved outputs.

For ease of implementation, we only focus on the 2-tap FSE in this section. The behaviors of a 2-tap FSE's even and odd outputs are shown in Figure 2-15. The coefficient and performance evolutions are almost identical, except at a 0.5 UI phase offset. The input

\[ \text{corr}(d_{2n}, d_{2n+1}) = E(\text{sign}(d_{2n}) \times \text{sign}(d_{2n+1})) \]

when the even output (generated by \( d_{2n} \) and \( d_{2n+1} \)) is selected as the FSE's output, where the expression \( E(x) \) means the expectation value of \( x \). When the odd output (generated by \( d_{2n+1} \) and \( d_{2n+2} \)) is selected as the FSE's output, it is defined as

\[ \text{corr}(d_{2n+1}, d_{2n+2}) = E(\text{sign}(d_{2n+1}) \times \text{sign}(d_{2n+2})) \].

In theory, the input signal sign correlation is between -1 and 1. For an FSE, it is between 0 and 1 due to the samples positive correlation. It can be shown that:

\[ \text{corr}(d_{2n}, d_{2n+1}) + \text{corr}(d_{2n+1}, d_{2n+2}) = 1. \]
Figure 2-15: The FSE coefficients, output and input sign correlation evolutions of a 2-tap FSE for a plesiochronous system at 5.0 Gb/s with 100 kHz frequency offset (without switching). Left: the even output is selected as the FSE’s output for the coefficient adaptation, data recovery and correlation calculation (sign correlation is between $d_{2n}$ and $d_{2n+1}$). Right: the odd output is selected as the FSE’s output (sign correlation is between $d_{2n+1}$ and $d_{2n+2}$).
signal sign correlations are also shown. The eye opening is good when the correlation is low; and the eye opening is bad when the correlation is high. When the correlation is high, the signs of the input samples are more likely to be the same. Therefore, the coefficient changes are more likely to be identical (see Equation 2.1). Thus, the adaptation becomes initial state dependent, causing performance degradation. In all, the FSE should switch to the other output as the correlation reaches some threshold\textsuperscript{10}.

Though the input signal sign correlation monitors the phase, it cannot tell whether the Rx clock frequency is lower or higher than the Tx clock frequency. If the receiver clock frequency is lower, the first recovered bit after switching (bit $b_{Rx,n+1}$ in Figure 2-13) is matched with the transmitted bit ($b_{Tx,n+1}$), and hence should be kept in the recovered data stream. Thus, $b_{Rx,n}$ is 0.5 UI spaced from $b_{Rx,n+1}$. As a result, the recovered bit stream is equivalently at the Tx clock rate, higher than the Rx clock rate. On the other hand, if the receiver clock frequency is higher, the first recovered bit after switching is a duplicate of the previous bit, and hence should be discarded. As a result, the last recovered bit before switching is 1.5 UI spaced from the next valid bit. Hence, the recovered bit stream is equivalently at the Tx clock rate, lower than the Rx clock rate. In all, the first bit after each switching should be flagged, and a back-end processing unit is required to decide whether to discard it or not\textsuperscript{11}.

The performance of the bit-skipping scheme is presented in Figure 2-16. Since the Rx clock frequency is 100 kHz lower than that of the Tx, the accumulated phase grows by 1 UI per 10 μs. When the correlation reaches the threshold (0.75), the receiver switches the output selection, and the FSE coefficients are re-initialized\textsuperscript{12}. As a result, the eye opening

\textsuperscript{10}Data correlation concept has been applied in the high-speed links to turn ON/OFF the adaptation in order to prevent bad data-pattern dependent erroneous coefficients wandering [24]. Similar to the bang-bang CDR, we use the data-edge correlation to monitor the phase movement.

\textsuperscript{11}This can be completed at a higher layer through coding/decoding. For example, at the Tx end, an $N$-bit sequence $(b_1, b_2, \ldots, b_N)$ is coded and transmitted as an $(N + 2)$-bit sequence: $(b_1, b_2, \ldots, b_N, 0, 1)$. If the Rx clock frequency is lower, the same $(N+2)$-bit sequence is recovered, even if the switching happens once in this procedure. On the other hand, if the Rx clock frequency is higher, the received $(N + 2)$-bit sequence would be $(b_1, b_2, \ldots, \hat{b}_k, \ldots, b_{N+1}, 0)$, where the bit $\hat{b}_k$ is flagged. Since the last two bits are no longer '01', the flagged bit $\hat{b}_k$ is a duplicate, and hence should be discarded. Thus, the transmitted bit sequence is received properly. Clearly, the number $N$ should be limited so that there is at most one flagged bit in the $(N + 2)$ bit cycles. For example, if the frequency offset is 1000 ppm, $N$ should be no more than 498, large enough to ignore the related coding overhead issue.

\textsuperscript{12}To improve the performance, the coefficients should be updated to the optimal for the new phase
Figure 2-16: Bit-skipping algorithm performance for plesiochronous systems at 5.0 Gb/s with a frequency offset of 20 ppm (100 kHz). The receiver output switches when the correlation is above 0.75.

Figure 2-17: Simulated jitter tolerance of the bit-skipping algorithm for plesiochronous systems at 5.0 Gb/s with a frequency offset of 400 ppm (2 MHz). The correlation threshold for the switching is 0.75.
is quite flat with 10% perturbation. This example shows that the bit-skipping scheme is able to compensate for the frequency offset, hence work for plesiochronous systems.

The jitter tolerance of the bit-skipping scheme is demonstrated in Figure 2-17. Given a frequency offset of 400ppm (2 MHz) and a minimal 300mV differential worst case eye opening requirement, the proposed bit-skipping algorithm can meet the required single-tone sinusoidal jitter tolerance of the XAUI standard.\(^{13}\)

### 2.4 FSE with a conventional CDR

An alternative approach to compensate for the frequency offset is to apply a conventional CDR, as shown in Figure 2-18, to the FSE receiver. Since an FSE itself has the ability to compensate for the sampling phase, which overlaps partially with the CDR’s working space, it is interesting to understand the potential advantages of this novel FSE with CDR system compared to the conventional SSE with CDR system under various jitter conditions.

#### 2.4.1 Jitter analysis for conventional SSE with CDR receivers

A conventional high-speed link receiver includes an SSE and a CDR. It is the CDR’s role to compensate for jitter. Depending on the sources, the jitter in links can be classified into three categories: transmitter-induced (Tx-induced) jitter, channel-induced jitter and receiver-induced (Rx-induced) jitter. Different types of jitter require different CDR strategies to optimize the system performance.

---

\(^{13}\)IEEE std 802.3ae-2002 PP-278, XAUI is a standard for extending the XGMII (10 Gigabit Media Independent Interface) between the MAC and PHY layer of 10 Gigabit Ethernet.
Figure 2-19: A high-bandwidth CDR loop is required to track the Tx-induced jitter. The simulated Tx-induced jitter is periodic at 100 kHz frequency with 2 UI_{pp} amplitude. An ideal CDR is assumed to lock to the zero crossings in real time and derive the data clock by 0.5 UI delay instantly. The channel-induced jitter can be ignored in this example.

**Tx-induced jitter**

The Tx-induced jitter\(^{14}\) includes the Tx reference clock jitter as well as the thermal noise and supply-noise induced jitter in both the Tx phase-locked loop (PLL) and the clock buffers. It causes signal timing uncertainties before the signals are introduced into the channel medium, which reduces the eye opening. To alleviate the impact, the CDR loop bandwidth is required to be as high as possible so that the receiver tracks the transmit jitter simultaneously. One example is shown in Figure 2-19. A transmitter periodic jitter almost closed the eye opening at the receiver end. However, the eye opening moves in the same way as the edges. In other words, the edge position is correlated with the optimal data sampling position. As long as the CDR has enough bandwidth to track the transmitter jitter, the 0.5 UI delayed data clock is close to the optimal.

**Channel-induced jitter**

The channel induced jitter is from two sources: channel noise and channel ISI. For most link channels, the ISI is much larger than the channel noise, and the ISI-induced jitter is by far the most dominant component of the channel-induced jitter [42]. Therefore, we only focus on the ISI-induced jitter here.

\(^{14}\)Only lower frequency jitter, which is constant over several bit cycles, is discussed. The high frequency jitter, which varies from cycle to cycle, is highly attenuated at the Rx end due to the channel loss, and therefore ignored.
Data sampling phase from a low bandwidth CDR

1. RX eye diagram w/ mild ISI induced jitter
2. Worst case (sampled) eye opening assuming an infinite bandwidth ideal CDR

Figure 2-20: A low-bandwidth CDR is required to filter the ISI induced jitter. (1) A very low bandwidth CDR will lock at 0.85 UI phase and derive the data clock at 1.35 UI. The resulting eye opening is 256 mV. (2) If the CDR loop bandwidth is infinitely wide and it adjusts the data clock right after each income edge, the resulting sampled eye opening is only 165 mV.

Varying from cycle to cycle, the ISI-induced jitter is a high frequency jitter. Being data-dependent, it impacts the data timing differently from the edge timing. The ISI from the previous bits (post-cursor) may move the edge/data timing forward with different amplitudes, and the ISI from the following bit (pre-cursor) may move them backward with different amplitudes. Since the ISI contributions at these two timing points are randomized by the surrounding data bits, it is worthless (or even harmful) to track the data timing movement through the edge movement [42].

One example is demonstrated in Figure 2-20. A low bandwidth CDR locking at 0.85 UI phase provides the data sampling phase at 1.35 UI with a worst case eye opening of 256 mV. On the other hand, an infinitely high-bandwidth CDR, which locks to each incoming zero-crossing and derives the data clock in real time, provides a worst case eye opening of 165 mV, 40% less than the low-bandwidth CDR. This example shows that to block the damaging ISI-induced jitter, the CDR bandwidth should be lower.

**Rx-induced jitter**

Rx-induced jitter is mainly due to the Rx PLL or VCO clock jitter, which drifts the data sampling phase from the optimal position and reduces the SNR of the received signal. Techniques for limiting the PLL output jitter can be found in [43], which will not be discussed here. To minimize the damage of the Rx-induced jitter, a high-bandwidth CDR...
loop is required to dynamically adjust the phase so that the data clock does not significantly drift away from the optimal point.

**Summary of the jitter analysis**

In the above analysis we reviewed the desired CDR loop behavior under various jitter sources and frequency characteristics. The lower frequency jitter from the transmitter moves the edge/data timing equally, hence should be tracked by a high-bandwidth CDR loop. On the other hand, the data-dependent high-frequency jitter from the channel moves the edge/data timing differently, hence should be blocked by a low-bandwidth CDR loop. Both the low frequency and high frequency jitter from the receiver PLL/VCO output move the receiver data clocks away from the optimal, hence should be tracked by a high-bandwidth CDR loop.

A summary of the jitter and the feasible CDR bandwidth is shown in the Table 2.3. To optimize the performance, the CDR speed should be tuned according to the jitter conditions. For example, for the ISI induced jitter dominant case, the CDR has to run slowly to mitigate the ISI jitter impact. On the other hand, for the Rx PLL jitter dominant case, the CDR has to run fast to hold the Rx clock at the optimal phase. This creates an optimal (adaptive) CDR bandwidth point for any signalling environment, depending on the relative contributions of these various jitter components.

<table>
<thead>
<tr>
<th>Jitter</th>
<th>Tx-induced</th>
<th>Channel-induced</th>
<th>RX-induced</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strategy</td>
<td>track</td>
<td>block</td>
<td>track</td>
</tr>
<tr>
<td>Desired CDR bandwidth</td>
<td>high</td>
<td>low</td>
<td>high</td>
</tr>
</tbody>
</table>

### 2.4.2 FSE and CDR combination

It is assumed that the timing recovery is done by the CDR only in the previous discussions. This assumption is true for the SSE with CDR systems because the SSE is worthless in timing recovery. However, it is not true for the FSE with CDR case. Since an FSE is able to recover the voltage regardless of the sampling phase, it partially overlaps with the CDR’s role. This section focuses on the performance of the FSE with CDR systems in various jitter environments.
Comparison benchmark

With a 20-inch channel showing 13 dB attenuation at 3.0 GHz, a 4-tap FSE with a conventional bang-bang CDR loop at 6.0 Gb/s is studied numerically. A 4-tap SSE with the same bang-bang CDR loop is simulated as well. The CDR model is shown in Figure 2-18. A first order CDR is applied ($K_i = 0$) in these examples without averaging (i.e. $N = 1$). The feed-forward gain $K_p$ is swept to adjust the CDR loop bandwidth\(^{15}\). To simplify the analysis, both the FSE and SSE adaptation engines are based on the $LMS$ algorithm with a fine step size ($< 10$ mV by default), and adapt at a quarter rate\(^{16}\). The output SNRs of the FSE and SSE are shown in Figure 2-21, which set the benchmark for the comparison.

With Tx-induced jitter

A sinusoidal jitter at 250 kHz with 1 UI\(_{pp}\) amplitude is added at the transmitter end. The CDR bandwidth is swept, and the equalization output SNR is shown in Figure 2-21. Clearly, this additional jitter decreases the SNR for both cases. For the SSE, the worst case degradation is up to 10 dB. As a result, the output SNR is only around 12.5 dB when the CDR bandwidth is low ($< 160$ kHz), and the eye is already closed. For the FSE, on the other hand, the degradation is only about 4 dB, 6 dB better than the SSE. Furthermore, the worst case SNR is well above 20 dB and the eye is still open.

This simulation shows that the FSE is able to track the Tx-induced jitter while the SSE cannot. In order to optimize the output SNR, the Tx-induced jitter should be well tracked. In this example, this task is left to the equalization block when the CDR bandwidth is limited ($< 40$ kHz). Since the SSE cannot handle this issue, its performance degrades significantly. The FSE, on the other hand, tracks the Tx-induced jitter well, and its per-

\(^{15}\)As a non-linear system, the bang-bang CDR loop bandwidth is defined through linearization [22, 44]:

$$f_{-3dB_{bb}} = (TD)K_pK_{VCO}K_{pd_{bb}},$$

$K_{pd_{bb}}$ is the linearized gain of the bang-bang phase detection, which is about 2 V/UI in this setup. $(TD)$ is the transition density of the input data [42, 45]. It is the ratio of the number of the transitions to the number of the bits transmitted. For a clock pattern input, it is 1. For random patterns, such as PRBS, it is close to 0.5. $K_{VCO}$ is 10 kHz/V. $K_p$ is swept from 4 to 512, hence, the bang-bang loop bandwidth is swept from 40 kHz to 5.12 MHz.

\(^{16}\)The FSE tunes the coefficients averaging by 4, i.e. $N = 4$ in the notation of the windowed $LMS$ adaptive algorithm:

$$W^{(k+1)} = W^{(k)} + \alpha \sum_{i=0}^{N-1} d^{(kN+i)} e^{(kN+i)}, k = 0, 1, 2, \cdots . \tag{2.3}$$

Since the data rate is 6 Gb/s, the adaptation speed is: $6G/4 = 1.5$ GHz.
Figure 2-21: A comparison of the sensitivity to the Tx-induced jitter (a sinusoidal jitter at 250 kHz with 1 UIpp amplitude) with different CDR loop bandwidth. The FSE with CDR system is less sensitive than the SSE with CDR system. The FSE adapts at 1.5 GHz for all except the one which is at 60 MHz.

Performance degrades on a much smaller scale. A faster FSE adaptation loop can reduce the performance degradation even further.

The FSE adaptation rate determines the tolerance to the Tx-induced jitter. Given the current condition, a 60 MHz FSE adaptation rate is fast enough to track the jitter, and provide reasonably good output, which is shown in Figure 2-21. On the other hand, at a 1.5 GHz adaptation speed, the jitter frequency can be up to 5.0 MHz, and the FSE system still operates well with different CDR bandwidth settings.

**With channel-induced jitter**

To alter the channel ISI induced jitter, the data rate needs to be increased. However, the rate increase also disturbs the data samples, making the comparison unfair. Instead of changing the data rate, the adaptation speed is updated to observe the impact of channel-induced jitter.

The analysis in the Section 2.4.1 demonstrates that a low bandwidth CDR is required to block the high frequency channel-induced jitter. Equivalently, a low speed FSE adaptation loop is desired, which is confirmed by the simulation results shown in Figure 2-22. For both the SSE and FSE cases, the performance is improved by increasing the adaptation...
averaging window from 4 to 32. Effectively, the adaptation speed is reduced from 1.5 GHz to about 200 MHz\textsuperscript{17}. This example shows that the adaptation rate should be kept lower to alleviate the ISI-induced jitter’s impact on the FSE loop. Equivalently, the CDR bandwidth should be set low.

**With Rx-induced jitter**

To mimic the Rx-induced jitter, the original clean Rx VCO is replaced by a noisy VCO with -94 dBc noise level at 10 MHz frequency offset. The simulation result is shown in Figure 2-23. As the CDR loop bandwidth reduces, the bandwidth of the transfer function from the VCO phase noise to the CDR output increases [43]. Therefore, the CDR output clock is noisier, which further contaminates the equalizer output signal.

However, the FSE is able to track part of the CDR output jitter to keep the performance relatively constant. For example, the performance degradation is only 4 dB when the CDR bandwidth is at 40 kHz. On the other hand, the degradation for SSE is over 8 dB, which is worse by 4 dB. In all, the FSE with CDR combination is less sensitive to Rx-induced jitter. To improve the FSE’s performance further, a faster adaptation loop is desired.

\textsuperscript{17}To keep the quantization error constant, the step size is scaled up by \(2\sqrt{2}x\) in this procedure.
Figure 2-23: A comparison of the sensitivity to the Rx-induced jitter. The FSE with CDR system is less sensitive than the SSE with CDR system.

Summary of the FSE with CDR system

In this section, the FSE with CDR system is studied under various jitter conditions and the proper FSE adaptation rate and CDR loop bandwidth are discussed. Since the FSE can do phase interpolation besides equalization, it introduces one more degree of freedom for system tuning, and improves the performance. Most importantly, the FSE with CDR combination outperforms the SSE with CDR combination for all the jitter conditions.

2.5 Summary

The analytical and numerical analyses show that FSE is immune to the sampling phase offset. Regardless of the sampling phase, it interpolates between the 0.5 UI spaced samples, recovering both the phase and voltage simultaneously. For mesochronous systems, the FSE does not require an extra CDR, which simplifies the receiver design greatly.

To work with unknown phases, a digital adaptation is required to tune the coefficients automatically. The SSLMS algorithm is practical for its low implementation cost. However, due to the noise with the 1-bit quantization, the SSLMS algorithm converges to a balanced solution instead of the MMSE solution, leading to performance degradation. In the worst case, the SSLMS algorithm diverges. Without additional hardware, our new M-SSLMS algorithm solves this issue through signal conditioning, which effectively introduces another
quantization level, and make the adaptation similar to the true 1.5-bit SLMS algorithm.

The bit-skipping scheme is also proposed here for the frequency offset compensation. By dynamically selecting between the even and odd outputs of the FSE, it keeps the residual phase within sub-1 UI, which is compensable by the coefficient adaptation loop. As an alternative approach, an FSE with CDR system is investigated. It surpasses the conventional SSE with CDR system under various jitter conditions, making the FSE implementation more attractive.
Chapter 3

FSE Tap Circuit Design

An FSE simplifies the receiver design by unifying the timing and voltage recovery loops, which can potentially reduce the power consumption and improve link performance over those highly lossy channels. However, it requires two relatively high resolution samples in each symbol cycle for effective adaptation. The conventional SSE with bang-bang CDR receiver, on the other hand, requires only one high resolution sample (data sample $d_n$ for voltage recovery in Figure 1-4), while the other sample (edge sample $e_n$ for timing recovery) is only 1-bit quantized. Thus, the FSE should be designed carefully so that the extra high-accuracy sampling and processing circuit does not destroy the benefits provided by the system architecture simplification.

The challenge of the FSE circuit design is to implement an FSE tap, which is fundamentally a multiplication and summation unit. The existing CML-based multiplier designs and their challenges are discussed in Section 3.1. Inspired by the ZCBC technique [28], two voltage-time (VT) conversion based designs are presented in Section 3.2. A comparison of this new technique with the CML-based design is given in Section 3.3.

3.1 FSE tap design challenges

The FSE tap design challenges include speed, power and linearity requirements. Most of the high-speed links operate in the multi-GHz range. Designing such a high-speed FIR tap with good energy-efficiency and a relatively high resolution (5-bit for effective adaptation) is challenging.

Most of the recent receiver FIR filter designs for high-speed links, including the SSE [11,46] and FSE designs [13,29,30], are CML-based. The basic tap circuit is shown in Figure
3-1 (1), which is fundamentally a Gilbert cell [47]. The voltage is converted into current first by the transconductance of input devices. Then, the current is converted back into voltage through the output resistors. Relying on voltage-time (VI) conversion, the output equals to the product of the input and the gain $g_m r_o$ to the first order, where $g_m$ is the transconductance of input devices and $r_o$ is the output resistance, including the resistor $R$ in parallel with the output resistance of the NMOS stack. Since $g_m$ can be tuned through changing the tail current, it functions as a multiplier.

There are several issues with the CML-based designs: tight power-linearity tradeoff, poor performance with the supply scaling and limited input range. Firstly, for the circuit shown in Figure 3-1, the input pair and the tail current determine the overall system linearity. For good linearity performance, the tail current has to be reasonably large$^1$ [46]. Circuit modifications like source degeneration [46] improve the linearity by reducing the gain and common-mode rejection. In order to keep the gain constant, the tail current has to increase, affecting the energy-efficiency. Secondly, for good linearity performance, both

---

$^1$The linearity can also be improved by increasing the current density of the input devices ($M_3$-$6$) through reducing their width. However, to keep the gain constant, the output resistors must increase. As a result, the bandwidth of the output nodes is decreased. Unfortunately, for this type of link receivers, the poles of the output nodes dominate for two reasons: (1) The output nodes are shared by multiple taps and follow-on sense amplifiers, hence the load and wire capacitance is large (tens of femto-farad). (2) To keep a feasible output range, the output resistors are generally large (multi kilo-Ohm). Hence, the space for current density tuning is limited. To improve the linearity without scarifying the speed, the tail current must increase.
the tail current device and the input device have to be saturated. Being stacked, they limit the output headroom and lead to smaller output range as the supply scales. Thirdly, the input range must be limited. The inputs with large range may push the input devices into linear or subthreshold region, affecting the linearity performance.

3.2 Voltage-time conversion principle

To achieve better power-speed-linearity trade-off and enable performance scaling in future low-supply processes, the basic ideas of ZCBC technique [28] for ADC designs are applied to the FSE tap design, in order to complete all FSE signal processing through linear current integration. In our design, the VT conversion based multiplier operates in two steps: a linear voltage-to-time conversion step and a linear time-to-voltage conversion step. In the first step, the voltage is converted into time domain by charging a capacitor with a constant current and sensing the output with a voltage threshold crossing detector. Thus, the voltage information is transferred into time domain linearly, carried by the edge-position modulated rail-to-rail signals. In the second step, the timing signal is transferred back into voltage domain by controlling the period for discharging a capacitor with a programmable current from a current digital-to-current converter (current DAC). This time-to-voltage conversion is proportional to the input timing and the DAC current. Since the timing is proportional to the channel voltage, a multiplier of the channel voltage and the T2V DAC current is created, realizing one FSE tap. To generate a multi-tap FIR filter, the output nodes of all the taps are connected directly, implementing the final summation/subtraction among taps directly through integration on the shared output capacitors.

This technique includes a voltage-to-time (V2T) converter and a time-to-voltage (T2V) converter. Two V2T converter designs and one T2V converter design are discussed in following sub-sections.

3.2.1 $V_{TH}$-based V2T design

A $V_{TH}$-based (NMOS-based) V2T converter pair transfers a signal pair from voltage domain into time domain proportionally by integrating constant currents and sensing the voltages with NMOS transistors, as shown in Figure 3-2.
Figure 3-2: Schematic and timing diagram of a conceptual $V_{TH}$-based V2T converter pair.
Operation principle

This V2T converter operates in two phases, namely track & preset phase, and evaluation phase. During the track & preset phase, clock $\Phi$ is high and nodes $V_{S+}/V_{S-}$ are precharged to VDD. Simultaneously, the sampling capacitors $C_{S+}/C_{S-}$ track the input signals $V_{in+}/V_{in-}$.

At the beginning of the evaluation phase, the falling edge of $\Phi$ pulls the bottom plate of the sampling capacitors $C_{S+}/C_{S-}$ from VDD down to GND. Simultaneously, there are two identical voltage drops at nodes $V_{X+}/V_{X-}$ caused by the capacitor divider circuit between the sampling capacitors $C_{S+}/C_{S-}$ and the parasitic capacitance $C_{X+}/C_{X-}$ of nodes $V_{X+}/V_{X-}$:

$$V_{X+} = V_{in+} - \frac{C_{S+}}{C_{S+} + C_{X+}} \times VDD,$$

$$V_{X-} = V_{in-} - \frac{C_{S-}}{C_{S-} + C_{X-}} \times VDD.$$

This divider circuit operates as a level shifter to guarantee that the voltages at nodes $V_{X+}/V_{X-}$ are initially lower than the threshold voltage $V_{TH}$ of transistors $N+/N-$ so that these two devices are OFF at the beginning of the evaluation phase.

Afterwards, two identical currents $I_{charge+}/I_{charge-}$ charge $V_{X+}/V_{X-}$ respectively. The voltages ramp up linearly with an identical rate until they cross the threshold voltages ($V_{TH+}/V_{TH-}$) of $N+/N-$ one after another. Thus, the output node $T_{O+}$ is pulled down at time:

$$t(T_{O+}) = \frac{(C_{S+} + C_{X+}) \times (V_{TH+} - V_{X+})}{I_{charge+}} + t_{detect,dl+}$$

$$= \frac{(C_{S+} + C_{X+}) \times (V_{TH+} - V_{in+})}{I_{charge+}} + \frac{C_{S+} \times VDD}{I_{charge+}} + t_{detect,dl+},$$

where $t_{detect,dl+}$ is the delay introduced by the threshold crossing detector. Similarly, the output node $T_{O-}$ is pulled down at time:

$$t(T_{O-}) = \frac{(C_{S-} + C_{X-}) \times (V_{TH-} - V_{in-})}{I_{charge-}} + \frac{C_{S-} \times VDD}{I_{charge-}} + t_{detect,dl-},$$

Assuming perfect matching ($C_{X+} = C_{X-} = C_X$, $C_{S+} = C_{S-} = C_S$, $I_{charge+} = I_{charge-} = I_{charge}$ and $t_{detect,dl+} = t_{detect,dl-} = t_{detect,dl}$), the timing difference between these two events
\[ \Delta t_O = t (T_{O+}) - t (T_{O-}) \]
\[ = \frac{(C_S + C_X) \times (V_{in+} - V_{in-})}{I_{charge}} \]
\[ = \frac{-C_S + C_X}{I_{charge}} \times \Delta V_{in}. \]  

Thus, the voltage information is proportionally transferred into time domain with a transfer gain of \(-\frac{C_S + C_X}{I_{charge}}\).

**Detailed analysis**

Based on the above analysis, the V2T conversion linearity is determined mostly by the linearity of the current \(I_{charge}\) (assuming relatively linear metal capacitors \(C_{S+}/C_{S-}\) and negligible parasitics). However, it matters only when the charge voltages \(V_{X+}/V_{X-}\) are lower than the threshold voltage \(V_{TH}\), which is as low as 300 - 350 mV in today’s sub-100 nm processes. The current source devices are deeply saturated in this region, achieving relatively good linearity performance.

The speed of this V2T conversion circuit is determined by the crossing detection speed. Given an input value \(V_{in+} = V_{TH} + \frac{C_{S+}}{C_{S+} + C_{X+}} \times V_{DD}\), because of the level shifting, the voltage at \(V_{X+}\) is \(V_{TH}\) after the level shifting. Device \(N+\) is turned ON immediately and starts to discharge \(T_{O+}\), generating a falling edge with a delay \(t_{detect,dl+}\), which contributes as part of the common delay in the time domain, limiting the system speed. Positive feedback PMOS \(P+/P-\) are placed to accelerate the \(V_{X+}/V_{X-}\) charging. When \(T_{O+}\) voltage is discharged to be lower than \(V_{DD}-V_{TH,P+}\), \(P+\) is turned ON. It helps to pull node \(V_{X+}\) up faster, and reduce the detection delay \(t_{detect,dl+}\) by 30% according to simulations\(^2\).

This V2T conversion circuit is relatively insensitive to the clock jitter. In each cycle, the uncertainty of the clock \(\Phi\) moves the output edges \(T_{O+}\) and \(T_{O-}\) in the same direction by the same amount, while the differential timing is unchanged. Hence, the clock jitter only shows as the common-mode timing variation, which can be rejected by the following T2V converter. On the other hand, the jitter of the sampling clock \(\Phi_S\) may impact the sampling accuracy. To minimize this effect, the sampling circuit should be designed with a proper

\(^2\)It is about 75 ps with \(P+/P-\). The nominal fan-out of 4 (FO4) delay in this technology is about 25 ps.
aperture (bandwidth) for a given application.

This V2T conversion circuit is relatively insensitive to the low to mid-frequency supply noise, in a way similar to the robustness to the jitter with clock $\Phi$. The low frequency supply noise moves the timing signals in the same direction with the same amount, leaving the differential timing unchanged. However, the high frequency supply noise that varies within each clock cycle can affect the timing edges differently as they propagate through the V2T converter stage, and hence get converted into a differential voltage offset. The range of timing signals has to be reasonably large to tolerate this effect$^3$.

In the analysis above, for the sake of simplicity, perfect matching ($C_{X+} = C_{X-} = C_X$, $t_{\text{detector,dl}+} = t_{\text{detector,dl}-} = t_{\text{detect,dl}}$, $V_{TH+} = V_{TH-} = V_{TH}$, $C_{S+} = C_{S-} = C_S$, $I_{\text{charge}+} = I_{\text{charge}-} = I_{\text{charge}}$) is assumed. However, the FSE tap shows different degree of sensitivity to these mismatches. The mismatch of detection delays $t_{\text{detector,dl}+}/t_{\text{detector,dl}-}$ results in a data-independent differential offset in the time domain. After the time-to-voltage conversion, it shows as a DC differential offset in the voltage domain. This differential offset can be canceled by shifting the threshold of the slicers. However, because it is dependent on the FSE tap value, the slicer threshold has to be tuned accordingly. The mismatch between the detector threshold voltages $V_{TH+}/V_{TH-}$ leads to a DC differential offset in the time domain, the same as the detection delay mismatch case. Hence, it can be compensated in the same way. The ratios $\frac{C_{S+} + C_{X+}}{I_{\text{charge}+}}$ and $\frac{C_{S-} + C_{X-}}{I_{\text{charge}-}}$ have to match well to keep the gains for the positive/negative legs equal. The mismatch between them can lead to input common-mode signal to differential output signal conversion. To circumvent this, a separate adjustment of $I_{\text{charge}+}$ and $I_{\text{charge}-}$ is required.

### 3.2.2 $V_M$-based V2T design

Aiming at higher performance, a $V_M$-based (inverter-based) V2T design is put forward to increase the crossing detection speed, enlarge the input range and enable the threshold mismatch compensation. A $V_M$-based V2T converter pair with timing diagram is shown in Figure 3-3.
Figure 3-3: Schematic and timing diagram of a conceptual $V_M$-based V2T converter pair.
Operation principle

Similar to the NMOS based V2T design, the inverter based V2T converter operates in two phases, namely *track & hold* phase and *evaluation* phase. During the *track & hold* phase, the back plate of the sampling capacitor $C_{S+}$ tracks the channel voltage $V_{in+}$, while the top plate tracks the inverter switching threshold voltage $V_{M+}$ by shorting the inverter’s input and output. In the *evaluation* phase, the back plate is first switched to GND. Due to the existing capacitor divider circuit, a voltage drop is generated at node $V_{CST+}$, which is a function of the sampling capacitance $C_{S+}$, load and parasitic capacitance $C_{LP+}$ of the node $V_{CST+}$:

$$\Delta V_{CST+} = \frac{C_{S+}}{C_{S+} + C_{LP+}} V_{in+}.$$  

Thus, the voltage at node $V_{CST+}$ is shifted to:

$$V_{ini,CST+} = V_{M+} - \frac{C_{S+}}{C_{S+} + C_{LP+}} V_{in+}, \quad (3.4)$$

which guarantees that $V_{CST+}$ is lower than $V_{M+}$ and the inverter output voltage is initially high. Afterwards, a constant current $I_{charge+}$ starts to charge the node $V_{CST+}$, and the corresponding voltage ramps up linearly. When it crosses the inverter switching voltage $V_{M+}$, a falling edge is generated at the inverter output with timing:

$$t (T_{O+}) = \frac{(C_{S+} + C_{LP+}) \times (V_{M+} - V_{ini,CST+})}{I_{charge+}} + t_{detect,dl+}$$

$$= \frac{C_{S+}}{I_{charge+}} V_{in+} + t_{detect,dl+}, \quad (3.5)$$

where $t_{detect,dl+}$ is the delay introduced by the threshold crossing detector$^4$. Similarly the other leg of the pseudo-differential V2T stage converts the other channel output $V_{in-}$ into time domain:

$$t (T_{O-}) = \frac{C_{S-}}{I_{charge-}} V_{in-} + t_{detect,dl-}. \quad (3.6)$$

$^3$At the follow-on T2V stage, these timing signals are used to control the time of integration. As the range of the timing signals increases, the integration may last for a longer time, hence more supply noise are integrated on the output capacitors. Therefore, it requires a small range of the timing signals. In all, there exists a tradeoff to determine the range of the timing signals between jitter tolerance and supply noise reduction.

$^4$About 50 ps in simulation.
Assuming perfect matching \((C_S^+ = C_S^- = C_S, I_{\text{charge}^+} = I_{\text{charge}^-} = I_{\text{charge}}\) and \(t_{\text{detect,d}l^+} = t_{\text{detect,d}l^-} = t_{\text{detect,d}l}\), the input differential voltage is converted into a differential timing signal:

\[
\Delta t_O = T_{O^+} - T_{O^-} = \frac{C_S}{I_{\text{charge}}} (V_{in^+} - V_{in^-}) = \frac{C_S}{I_{\text{charge}}} \Delta V_{in},
\]

which controls the current switches of the follow-on differential T2V converter.

Similar to the \(V_{TH}\)-based design, the voltage information is converted into time domain. The transfer factor amplitude \(\frac{C_S}{I_{\text{charge}}}\) is different from that for the \(V_{TH}\)-based design.

**Detailed analysis**

The \(V_M\)-based V2T converter shows three advantages over the \(V_{TH}\)-based V2T converter: higher operation rate, greater input range and auto-zeroing mechanism for threshold mismatch compensation.

To compare the speed, \(g_{mn} (V_{in})\) and \(g_{mp} (V_{in})\) are used to denote the transconductance of the NMOS and PMOS transistors respectively, and \(C_L\) is used to denote the load capacitance. When the input voltage exceeds the threshold, the output node bandwidth of the \(V_{TH}\)-based design is:

\[
\frac{|g_{mn} (V_{TH})|}{C_L},
\]

while the bandwidth of the \(V_M\)-based design is:

\[
\frac{|g_{mn} (V_M)| + |g_{mp} (V_M)|}{C_L}.
\]

Assuming the same transistor size and load capacitance, with \(V_{TH}\) as the gate voltage, the transistor is only weakly ON and the corresponding transconductance \(|g_{mn} (V_{TH})|\) is smaller than \(|g_{mn} (V_M)| + |g_{mp} (V_M)|\). Therefore, the \(V_M\)-based V2T converter provides greater bandwidth, reduces the detection delay \(t_{\text{detect,d}l}\), and increases the operation rate.

The \(V_M\)-based V2T converter can operate with a greater input range than the \(V_{TH}\)-based V2T converter because \(V_M > V_{TH}\). For a 90-nm CMOS process used in this thesis,
the NMOS threshold voltage $V_{TH}$ is about 350 mV, and the inverter switching voltage $V_M$ is about 600 mV, with a 1.2 V supply. For the $V_{TH}$-based implementation, to make sure that the inverter input voltage is no less than 0 mV after the level shifting operation, the maximal channel voltage is VDD and the minimal channel voltage is VDD - $V_{TH}$ (assuming zero parasitics). Therefore, the input range is $\pm V_{TH} = \pm 350$ mV. For the $V_M$-based implementation, to meet the same criteria, the maximal channel voltage is 600 mV and the minimal channel voltage is 0 mV. Therefore, the maximal differential input range is $\pm 600$ mV, which is 1.7x of the $V_{TH}$-based case. In the $V_M$-based design, due to the parasitic capacitance at the detector input node, the input signal is attenuated after level shifting, as shown in Equation 3.4. Therefore, the maximal input voltage can be even larger. In the implementation, the level shifting ratio $\frac{C_{S+}}{C_{S+} + C_{L+}}$ is around 0.75x. Thus, the input signal range for the $V_M$-based design can be up to $\pm 800$ mV. As a result, the $V_M$-based V2T converter is compatible to a broader range of the high-speed link transmitters, especially those designed in the older technology with large output range. Furthermore, it can operate under the environment of no transmitter equalization, where the signal range is relatively large due to the existence of ISI.

Implemented with an auto-zeroing scheme, the $V_M$-based design can compensate for the detector threshold mismatch. As shown in Equation 3.5 and Equation 3.6, both $V_{M+}$ and $V_{M-}$ are canceled, hence their mismatch does not impact the output timing. This compensation reduces the tap value dependent offset at the FSE output stage, and enhances the system sensitivity to the input.

The impact of supply noise, jitter and mismatch to the $V_M$-based converters is very similar to that of the $V_{TH}$-based converters. Therefore, the related discussion is omitted here.

**Improvement of the $V_M$-based design**

The $V_M$-based V2T converter design is not as energy-efficient as the $V_{TH}$-based design due to the power-hungry threshold mismatch compensation scheme. To compensate for the threshold mismatch, the detector inverter has the input connected with the output during the *track* and *preset* phase. In this procedure, the inverter input settles to the switching voltage $V_M$. Since $V_M$ is close to VDD/2, both the NMOS and PMOS of the inverter are strongly *ON*. They conduct current from supply directly to the ground, and increase the
Figure 3-4: Schematic and timing diagram for an improved $V_{th}$-based V2T converter. It compensates for the threshold mismatch with zero power consumption.
power dissipation.

To improve the energy-efficiency, an improved $V_M$-based design is proposed, as shown in Figure 3-4. The threshold detector is designed as a segmented inverter stage, with zero static power dissipation and programmable switching threshold. It avoids the direct path from supply to ground during the track and preset phase in the original design and achieves better energy-efficiency.

Same as the two predecessors, it operates in two phases, i.e. track & preset phase and evaluation phase:

- **track & preset phase**

  Clock $\Phi$ is high. The sampling capacitor $C_S$ tracks the channel voltage. $V_{\text{dischar}}$ is preset high to set $E_{\text{buf}}$ low and $T_{O+}$ high.

- **evaluation phase**

  Clock $\Phi$ is low. A constant current discharges node $V_{\text{dischar}}$, which is connected to the sampling capacitor $C_S$ through a switch. When the voltage at $V_{\text{dischar}}$ crosses the inverter switching voltage $V_{M+}$, a rising edge is generated at $E_{\text{buf}}$ and a falling edge is derived at the output. The output edge timing $t(T_{O+})$ is:

  $$t(T_{O+}) = \frac{C_S}{I_{\text{discharge}+}} \times (V_{in+} - V_{M+}) + t_{\text{detect},dl+},$$  

  \hspace{1cm} (3.8)

  where $t_{\text{detect},dl+}$ is the delay introduced by the threshold crossing detector. Similarly, the output edge timing of the other leg $t(T_{O-})$ is:

  $$t(T_{O-}) = \frac{C_S}{I_{\text{discharge}-}} \times (V_{in-} - V_{M-}) + t_{\text{detect},dl-}.$$  

  \hspace{1cm} (3.9)

Assuming perfect matching ($C_{S+} = C_{S-} = C_S$, $I_{\text{discharge}+} = I_{\text{discharge}-} = I_{\text{discharge}}$, $V_{M+} = V_{M-} = V_M$ and $t_{\text{detect},dl+} = t_{\text{detect},dl-} = t_{\text{detect},dl}$), the input differential voltage $\Delta V_{in}$ is converted into a differential timing signal:

  $$\Delta t_O = t(T_{O+}) - t(T_{O-})$$

  $$= \frac{C_S}{I_{\text{discharge}}} (V_{in+} - V_{in-})$$

  $$= \frac{C_S}{I_{\text{discharge}}} \Delta V_{in},$$  

  \hspace{1cm} (3.10)
which is almost the same as the original $V_M$-based design (Equation 3.7), except the charge/discharge current term.

To work with high common-mode channel signaling, a pull-down scheme is applied in this design. In the previous implementations, the detector input voltages are pulled up linearly to cross the threshold in each \textit{evaluation} phase. Since the input voltage cannot be higher than 800 mV\textsuperscript{5}, the first two V2T converter designs prefer relatively low input common signals so that detector initial input voltage is below the threshold at the beginning of the \textit{evaluation} phase. For this new design, the detector input voltage is pulled down to cross the detector threshold during each \textit{evaluation} phase. It is compatible with the high common-mode transmitters\textsuperscript{6}, which are widely adopted in the high-speed link applications [6,48]. Furthermore, the level shifting operation is eliminated.

3.2.3 T2V design

A time-gated integration stage is needed to implement the final summation and subtraction of the FIR filter. Equivalently, this circuit can be viewed as a time-to-voltage (T2V) converter with shared output capacitors among all taps, performing summation/subtraction through charge redistribution.

A differential one-tap T2V converter is demonstrated in Figure 3-5. Switches are placed between VDD and the output nodes to preset the state in each cycle. The timing signal controls current switches to select the output nodes to discharge. The tail current source controls the discharge current amplitude. The T2V converter operates in \textit{preset} and \textit{evaluate} phases. At any time, the V2T converters and T2V converters are either both in the \textit{preset} phase or both in the \textit{evaluation} phase. During the \textit{preset} phase, the output nodes $V_{O+}/V_{O-}$ are precharged to VDD. Since the current paths are clock gated, there is no significant crowbar current during the \textit{preset} phase.

During the \textit{evaluation} phase, the two output nodes are first discharged with a shared current $I_1$ until the first timing signal arrives. Without loss of generality, $T_{O-}$ is assumed

\textsuperscript{5}From Equation 3.4, assuming that $V_M$ is 600 mV and the attenuation factor $\frac{C_{S+}}{C_{S+} + C_{LP+}}$ is 0.75x.

\textsuperscript{6}On the other hand, to make this design work in the low common-mode systems, only two modifications are needed: (1) change the preset circuit so that node $V_{\text{dischar}}$ is preset low in each cycle, (2) change the current source type from discharging to charging.
Figure 3-5: Schematic and timing diagram of a one-tap differential T2V converter.
to be the first timing edge. The edge of \( T_{O-} \) floats the output node \( V_{O-} \) at:

\[
V_{O-} = VDD - \frac{t(T_{O-})}{C_{T2V-} + C_{T2V-}} \times I_1.
\]

The node \( V_{O+} \) continues to discharge at a double rate until the \( T_{O+} \) edge arrives. At this point, \( V_{O+} \) is:

\[
V_{O+} = VDD - \frac{t(T_{O-})}{C_{T2V+} + C_{T2V-}} \times I_1 - \frac{(t(T_{O+}) - t(T_{O-}))}{C_{T2V+}} \times I_1.
\]

Assuming that the output node capacitors are matched, \( C_{T2V+} = C_{T2V-} = C_{T2V} \), the T2V converter output is:

\[
\Delta V_O = V_{O+} - V_{O-} = -\frac{I_1}{C_{T2V}} \times \Delta t_O.
\]  

(3.11)

Since then, the discharging is stopped and the different output is kept till the end of evaluation phase. Thus, the multiplication of the input timing and the tail current is realized by current integration.

This new design can achieve better linearity than conventional integrating receivers [25, 49]. For conventional designs, the input information is in the voltage domain and it is transferred into current by the device transconductance, which is non-linear especially when the range of the input signal is large. For this new design, the input information is in the time domain, controlling the current switches that are either fully \( ON \) or \( OFF \). Thus, the transconductance nonlinearity is mitigated and the voltage headroom of the integrator output is increased, leading to better linearity performance\(^7\).

According to Equation 3.3 and Equation 3.7, the relationship between the T2V differential output voltage \( \Delta V_O \) and the channel output voltage is:

\[
\Delta V_O = \begin{cases} 
\frac{C_S + C_X}{C_{T2V}} \times \frac{I_1}{I_{charge}} \Delta V_{in} & \text{for } V_{TH} \text{-based V2T} \\
-\frac{C_S}{C_{T2V}} \times \frac{I_1}{I_{charge}} \Delta V_{in} & \text{for } V_M \text{-based V2T}.
\end{cases}
\]  

(3.12)

The negative sign in Equation 3.12 can be flipped by exchanging the terminal connections of \( V_{O+} \) and \( V_{O-} \). Thus, relying on the voltage-time conversion, a programmable FIR filter\(^7\).

\(^7\)The linearity of generating the static tail current \( I_1 \) also impacts the system linearity. However, since it is usually from a digital-to-analog converter (DAC), its linearity improves as the DAC resolution increases.
tap is built, with $I_1$ as the gain tuning knob. In addition, tuning any of $C_S$, $C_{T2V}$ and $I_{\text{charge}}$ also alters the transformation gain. These extra degrees of freedom can be further used for mismatch compensation. For example, the mismatch between $C_{S+}$ and $C_{S-}$ can be corrected by tuning $I_{\text{charge+}}$ and $I_{\text{charge-}}$ (or $I_{1+}$ and $I_{1-}$) or vice versa, while keeping:

$$\frac{C_{S+}}{I_{\text{charge+}}} = \frac{C_{S-}}{I_{\text{charge-}}}.$$  

However, the output common-mode voltage is

$$V_{CM} = \frac{V_{O+} + V_{O-}}{2} = VDD - \frac{t(TO_+) \times I_1}{2C_{T2V}},$$

which varies depending on the edges $TO_+$ and $TO_-$. This variation can alter the effective bias level of the following slicer, reducing the slicer input sensitivity. Fundamentally, this problem is due to the fact that the discharging is stopped after the latest edge. Thus, the total integrated charge is $t(TO_+) \times I_1$ or $t(TO_-) \times I_1$, which varies from cycle-to-cycle. To keep the output common-mode voltage constant, design modifications are required. One method is to keep discharging after the latest edge, pulling the two output nodes $V_{O+}/V_{O-}$ at the same slope $\frac{I_1}{2C_{T2V}}$, as shown in Figure 3-6. Thus, the differential output is still kept, but the tail current is effectively ON for the whole cycle $T_{eva}$. The total integrated charge is always $T_{eva} \times I_1$ and the output common-mode voltage becomes:

$$V_{CM} = VDD - \frac{T_{eva} \times I_1}{2C_{T2V}}, \text{ } T_{eva} \text{ is the period of the evaluation phase.}$$

Thus, the output common-mode is a constant as long as the tap weight $I_1$, is a constant\(^8\).

Relying on the linear current integration, an FSE tap is built by converting the channel voltage into time domain and then converting the timing signal back into voltage domain with weighting. For a multi-tap FSE implementation, all the taps share the same two output capacitors $C_{T2V+}$ and $C_{T2V-}$. The summation and subtraction are automatically completed in the charge domain.

\(^8\)For a multiple-tap FSE, as long as the sum of the taps is a constant, the output common-mode does not vary.
Figure 3-6: Timing diagram of a one-tap differential T2V converter with a constant common-mode output. The discharge continues with the two output nodes after the latest edge.

**Improvement of the T2V converter design**

The existing T2V design suffers from two issues:

- *No rejection of the common-mode signal in time domain*

  The input common-mode signal includes two parts: the period before the first edge and the period after the last edge. In these two periods, the two output nodes are both pulled down linearly and no differential output is derived. However, the valuable output voltage headroom is consumed and the space for voltage differentiating is reduced. It limits the range of the FSE tap gain and deteriorates the system sensitivity to the input. One method that can alleviate this issue is to keep the V2T gain \( \frac{C_S}{I_{\text{charge}}} \) as large as possible to increase the differential timing signal’s range and effectively reduce the common-mode signal in time domain. However, it cannot solve this issue completely because the detection delay \( t_{\text{detection, dl}} \) and the signal propagation delay (from V2T converter to T2V converter) are unchanged, yet contribute non-negligible parts of the common-mode signal in time domain. A T2V converter with rejection of the input common-mode timing signal is desired.

- *Limited time for the slicer to settle*

  The T2V output nodes are reset to VDD immediately after the *evaluation* phase, right at the same time as the slicer detects the voltage difference and tries to generate a
digital output ‘0’ or ‘1’. It only leaves a narrow window for the slicer to settle, and then destroys the T2V output quickly, which worsens the system sensitivity to the input. The reset current can be designed low to expand the slicing window. However, the output nodes may not be charged fully to VDD at the end of the preset phase, putting the system in danger of hysteresis issues.

To overcome these problems, an improved design is proposed. As shown in Figure 3-7, the improvements include:

- **Rejection of the common-mode signal in time domain**

  The discharge happens only when the input timing signals carry useful information, namely their voltage levels are different: it discharges positive output node $V_{O+}$ as $T_{O+}$ is low and $T_{O-}$ is high; it discharges the negative output node $V_{O-}$ as $T_{O+}$ is high and $T_{O-}$ is low. For all the other cases, the discharge current is bypassed. A summary of the discharge logic is shown in Table 3.1.

<table>
<thead>
<tr>
<th>Scenario</th>
<th>$T_{O+}$</th>
<th>$T_{O-}$</th>
<th>Node to discharge</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scenario1</td>
<td>high</td>
<td>low</td>
<td>$V_{O+}$</td>
</tr>
<tr>
<td>Scenario2</td>
<td>low</td>
<td>high</td>
<td>$V_{O-}$</td>
</tr>
<tr>
<td>Scenario3</td>
<td>high</td>
<td>high</td>
<td>None (bypass)</td>
</tr>
<tr>
<td>Scenario4</td>
<td>low</td>
<td>low</td>
<td>None (bypass)</td>
</tr>
</tbody>
</table>

  Thus, the output nodes are pulled down only when input timing carries the data information. Equivalently, it rejects the common-mode signal in the time domain and extracts the differential timing information to generate a differential voltage output. No output headroom is sacrificed for the common-mode part of the input timing signal.

- **Extended period for slicing**

  A 0.5 UI delayed clock $\Phi_{delay}$ is introduced to defer the starting time of preset. Thus, the T2V output will be kept for an extra 0.5 UI, which is lent to the slicer to resolve the voltage difference. Thus, the slicer’s sensitivity to the FSE’s output is increased.

One potential issue with the modified design is the output common-mode variation. For the one-tap case, which functions as a gain stage, a differential output signal $\Delta V_O$ shows
as $VDD$ and $VDD - \Delta V_O$ on the two legs respectively. Thus, the output common-mode voltage is $VDD - \Delta V_O/2$, which varies as the output differential voltage changes. It disturbs the slicer’s input referred offset, and reduces the slicer’s input sensitivity. This topic and a compensation circuit are described further in Appendix A.

### 3.3 Comparison of VT-based design and CML-based design

The VT-based design outperforms the CML-based design in power scaling. For the CML-based design, most of the power dissipation is caused by the analog tail current. For the VT-based design, on the other hand, about half of the power is consumed by the clocking\(^9\). In addition, most of the internal signals in CML-based design are analog, while those for the VT-based design are ‘digital’\(^10\). Hence, the power of the VT-based implementation scales similarly to that of a digital system, better than the CML-based implementation.

The VT-based design outperforms the CML-based design in linearity performance. To keep the output bandwidth, the linearity of the CML-based design is determined by the tail current amplitude, highly coupled with the power. The linearity of the VT-based design, on

---

\(^9\)According to the post-layout simulation of the $V_M$-based FSE receiver design.

\(^10\)They are digital (rail-to-rail) in voltage domain and analog in time domain.
the other hand, is determined by the quality of the current sources, i.e. the independence to the voltage across them. Thus, the performance is no longer tightly power-coupled and scales reasonably well with the process technology development. In addition, the following property further improves the linearity of the VT-based design further.

Both of V2T and T2V current devices are designed to be in deep saturation region so that the output currents are quite constant. Thus, relatively good linearity performance is achieved. For example, in the $V_M$-based design, $I_{\text{charge}}$ is required to be linear only when the detector input voltage $V_{CST}$ is lower than the threshold $V_M$, where the current source device is in deep saturation region and hence can be viewed as a relatively constant current source. Similarly, assuming that the T2V differential output range is $\pm VDD/2$, the minimal voltage at one leg is $VDD/2$, enough to keep the device for current $I_1$ in deep saturation region\textsuperscript{11}.

Given the same power budget, the VT-based design can achieve better linearity. Equivalently, given the same linearity requirement, the VT-based design can achieve better energy-efficiency. Simulations show that given the 5-bit linearity and unity gain requirements, with a 800 mV differential input at 1.0 V supply, the $V_M$-based voltage-time conversion technique\textsuperscript{12} provides approximately 50% power savings\textsuperscript{13} over the tail-degenerated CML-based implementation.

The VT-based design outperforms the CML-based design in the compatibility to the range of the input signals. For the CML-based design, the input’s range is determined by the linearity requirement. A large differential input can push one input device into linear region and push the other one into subthreshold region, degrading the transform linearity. For the VT-based designs, on the other hand, the feasible input range is determined by the detector threshold settings. For example, the input range for the $V_{TH}$-based design is $\pm V_{TH}$, while that for the $V_M$-based design is $\pm V_M$,\textsuperscript{14} which is about $\pm VDD/2$. By tuning the threshold lower than VDD/2, the feasible input range for the improved $V_M$-based V2T converter can be even greater. Obviously, to expand the input range, the charge current\textsuperscript{15}

\textsuperscript{11}In the CML-based design, on the other hand, to keep the good linearity performance, both the tail current device and the input devices must be in deep-saturation region, limiting the output range.
\textsuperscript{12}This comparison is done with $V_M$-based design, which is the least energy-efficient of all three designs that we described. Even greater savings can be expected from the $V_{TH}$-based and improved $V_M$-based designs.
\textsuperscript{13}Clock power included.
\textsuperscript{14}Assuming no parasitics. Actually, the parasitics increase the feasible input range further, which can be observed from Equation 3.4.
\textsuperscript{15}For the improved $V_M$-based design, it is the discharge current.
in the V2T converter must increase.

3.4 Summary

To relax the tight coupling between power and linearity in the conventional CML-based FSE tap designs, the VT-based tap circuits are developed in this chapter. In this new technique, the channel voltage is first converted into time domain and then converted back into voltage domain with weighting. Both conversions are completed by linear current integration, with all integration currents independent of the input channel voltage, which avoids the nonlinear voltage-current transformation through input device transconductance.

Two voltage-to-time converter types are discussed in this chapter. In the first design, an NMOS is applied as the threshold detector, and its threshold voltage $V_{TH}$ is the detector threshold. In the second design, an inverter with auto-zeroing function is used as the threshold detector, and its switching voltage $V_M$ is the detector threshold. The inverter based design increases the speed, enlarges the input range, and enables the compensation for the detector threshold mismatch. However, to track the threshold $V_M$, the compensation scheme generates a direct current path from VDD to ground, worsening the energy-efficiency performance. To improve the energy-efficiency, an improved $V_M$-based V2T converter also developed. In the improved V2T converter, the segmented inverter is used as a comparator with digitally programmable switching threshold and no crowbar current.

A time-to-voltage converter design is also presented in this chapter, which is implemented as a digitally-controlled, current-switched integrator. The current switches control the discharge period and the current source controls the discharge current amplitude. Since the discharge period is proportional to the channel voltage, a multiplier of the channel voltage and the T2V discharge current is created. An improved T2V design saves the output headroom by rejecting the common-mode part of the input timing signals. It also keeps the output voltage constant for 0.5 UI providing enough hold time for the slicer to settle.

The V2T converter and the T2V converter form an FSE tap. In contrast with the conventional CML-based implementations, the voltage-time conversion based multiplier shows better power-linearity performance. Their advantages are verified through simulations in this chapter. In the next chapter, we describe the experimental infrastructure and proof-of-concept designs that further quantify the functionality of the proposed circuits.
Chapter 4

FSE Implementation

System level investigation shows that the FSE unifies the timing and voltage recovery loops, providing a much simplified high-speed link receiver architecture than conventional designs. Circuit level study of the VT-based design shows a novel way to build an FIR tap with better power-linearity performance than the conventional CML-based implementation. Due to the high-speed and mixed-signal nature of these ideas, proof-of-concept experiments are needed to illustrate their feasibility. In this section, we describe two voltage-time conversion based FSE implementations to demonstrate:

- FSE performing equalization and synchronization simultaneously\(^1\)
- M-SSLMS algorithm convergence during FSE adaptation
- VT-based designs achieve better power-linearity performance than the CML-based designs

As discussed in Chapter 3, the essential circuit component is the threshold detector. In the 1\(^{st}\) generation implementation, an NMOS is applied as the crossing detector, and the details are covered in Section 4.1. For higher data rates, the 2\(^{nd}\) implementation applies an inverter as the crossing detector, and the details are covered in Section 4.2. Both designs are for mesochronous systems. The 3\(^{rd}\) implementation, which is for plesiochronous systems, is presented in Section 4.3. The estimated performance is given and compared with the recent state-of-art timing recovery designs.

\(^1\)We focus on the FSE's immunity against the sampling phase, which is not addressed in the previous FSE implementations [13,29,30].
Figure 4-1: Architecture of a 2-way time-interleaved 2-tap FSE. Scan-chain and snapshot are applied for in situ link characterization. Clock pair \((\Phi_0, \Phi_{180})\) and the clock pair \((\Phi_{90}, \Phi_{270})\) offset by a quadrature delay are from off-chip. The sampling clocks \((\Phi_{S1} - \Phi_{S4})\) are generated on-chip.

4.1 \(V_{TH}\)-based FSE — 1

4.1.1 Micro-architecture

In the VT-based design, all of the information is processed through the voltage-to-time and time-to-voltage conversions. The range of the timing signals determines the upper bound of the conversion speed. For example, in the 90-nm process that we used, the detection delay plus the signal propagation delay is about 150 ps. Assuming that the timing signal range is \(\pm 100\) ps, the clock cycle must be \(2 \times (150 + 100) = 500\) ps, setting the highest rate at 2 Gb/s. To overcome this issue, a 2-way (even and odd) time-interleaving technique is applied to double the data rate to 4.0 Gb/s.

The block diagram of the implemented 2-tap 2x over-sampled FSE receiver is shown in Figure 4-1. Four clocks \((\Phi_0, \Phi_{90}, \Phi_{180}, \Phi_{270})\) in quadrature phases are supplied from off-chip,
and four sampling phases ($\Phi_{S1} - \Phi_{S4}$) with 25% duty cycle are derived locally.

A $V_{TH}$-based V2T converter samples and converts the channel voltage signal into a position modulated edge pair, which includes a rising edge and a falling edge crossing at $V_{DD}/2$. Two V2T converter pairs process two sampled differential voltages, corresponding to the two FSE taps. A follow-on 2-tap T2V converter completes the summation/subtraction and multiplication operations.

Scan-chains and snapshots are placed around the core circuit to configure each block and bring the data out of the chip for further analysis.

4.1.2 $V_{TH}$-based V2T converter

The implemented V2T converter is shown in Figure 4-2. The threshold crossing detector is an NMOS transistor and the detector threshold is the transistor threshold voltage $V_{TH}$, which is about 350 mV for the 90-nm process. The V2T converter pairs are pseudo differential. The sources of the input NMOS devices are tied together for better matching.

Instead of sending the timing signal $T_{O+}/T_{O-}$ directly to the T2V converter, a buffer chain is placed on the signal path. The first two inverters derive a rising edge and a falling edge from the falling edge $T_{O+}$ or $T_{O-}$, which are required by the differential T2V converter. The following sign muxes alter the polarity of V2T conversion to switch between summation and subtraction operations. The inverters and sign muxes also steepen the edge, improving the timing accuracy and the robustness against the supply noise. However, they introduce extra delay, which becomes an extra part of the common-mode signal in the time domain, reducing the headroom for the differential timing output.

4.1.3 T2V converter

The implemented one-tap differential T2V converter is shown in Figure 4-3. Made by two current pairs, it takes 2 pairs of edges as the input and converts the time domain signal $\Delta T$ into a voltage domain signal $\Delta V_O$. There are two phases of operation: a preset phase and an evaluation phase. In the preset phase, the output nodes are precharged to VDD. In the evaluation phase, the output nodes are discharged by $I_1$. The input timing determines the period of discharge, and the tail current determines the discharge current amplitude. Thus, the multiplication of timing and current is completed. The timing signal is converted back into voltage through current integration at the end of the evaluation phase. The analysis and conclusions on the conceptual design shown in Figure 3-5 are still valid for this
Figure 4-2: Schematic and timing diagram of a V2T converter pair. It converts two sampled voltages (differential) into two timing signals (differential).
implementation except that the effective tail current is $2I_1$ instead of $I_1$ in this design.

There are three devices in the stack from the outputs $V_{O+}/V_{O-}$ to the ground, which would require large voltage headroom and limit the output range to keep the devices in the saturation region, as would be the case for the CML-type designs, such as those used in integrating link receivers [49]. However, in this implementation, two out of three devices are switches, controlled by the rail-to-rail timing signals and clock signal. Their low drain-source voltage drop allows enough headroom on the tail current source transistor even at reduced supplies. Furthermore the voltage headroom for the output is well kept.

For better linearity performance, the tail current devices should be always ON. Instead of bypassing the current to the supply in the preset phase, the current sources are redirected to the other way (even/odd) to improve the energy-efficiency performance. However, the parasitic capacitance on the tail node may result in hysteresis, which could increase the ISI equivalently and degrade the performance. This problem can be overcome by increasing the output node capacitance, at the cost of lower energy-efficiency. An alternative approach is to keep the parasitic capacitance as small as possible. Thus, the related wire has to be kept as short as possible in the layout. It is doable for the 2-way interleaved case, but quite
challenging for the 4-way interleaved case. Therefore, we discard this inter-way current sharing scheme in the later 4-way interleaved implementations.

A 2-tap T2V converter is shown in Figure 4-4. It consists of two 1-tap differential T2V converters that share the output nodes. Since the multiplication is processed in the charge domain, the summation and subtraction are all completed automatically by charging and discharging the common output capacitors.

4.1.4 Slicer

The slicer design is shown in Figure 4-5. It consists of a threshold tunable strong-arm sense amplifier, an SR latch and a synchronization latch [50]. The slicer detects a small-swing signal and produces a digital bit ‘0’ or ‘1’.

The sense amplifier operates in two phases, precharge and evaluation phases. In the precharge phase, the clock $\Phi$ is low. The output nodes $V_{O+}/V_{O-}$ are precharged to VDD. Since the tail NMOS $M_{10}$ is OFF, there is no leakage current. The equalization PMOS $M_9$ is ON to reduce the voltage difference between $V_{O+}/V_{O-}$. In the evaluation phase, the clock $\Phi$ is high. The precharge PMOS $M_7/M_8$ are OFF and the tail NMOS $M_{10}$ is ON. The input NMOS $M_1/M_2$ pull down $V_{O+}/V_{O-}$. Without loss of generality, the input $V_{in+}$
is assumed to be higher than $V_{i_{in}}$. Then, $V_{O-}$ is discharged with a higher current and its voltage drops at a higher rate. As $V_{O-}$ reaches the switching threshold first, $V_{O+}$ is pulled high. Due to the positive feedback mechanism of the cross-coupled inverter, $V_{O-}$ is pulled low simultaneously and a digital output '1' is generated.

The tail current $I_{TH}$ and digital signals $V_{TH+}/V_{TH-}$ control the threshold of the sense amplifier. The bias current $I_{TH}$ pulls down one of the output nodes, and generates a small voltage difference between $V_{O+}/V_{O-}$ in the *precharge* phase. To compensate for this imbalance and keep the output binary distributed with 50% probability, a non-zero differential input is required, which is the input referred offset, i.e. the slicer's threshold. Furthermore, the amplitude of $I_{TH}$ determines the amount of the offset, and the digital signals $V_{TH+}/V_{TH-}$ determine polarity. A more detailed study of this circuit is presented in Appendix A.

An SR latch follows the sense amplifier. It passes the SA output during the *evaluation* phase and locks the SA output during the *precharge* phase. A clocked latch synchronizes the SR latch output for the follow-on digital blocks, such as the snapshot circuit.
4.1.5 Measurements

The design is fabricated in a 90-nm CMOS process. The chip layout is shown in Figure 4-6. It includes the $V_{TH}$-based 2-tap 2x over-sampled FSE with additional support blocks, such as scan-chains and high-speed data snapshots to enable link tuning and *in situ* performance characterization. The FSE receiver core area is 140 $\mu$m x 115 $\mu$m.

The main goal of this chip is to demonstrate the FSE robustness (immunity) to phase offset. To achieve this goal, the delay between the input clock and the channel output data in the measurement setup must be tunable. The measurement setup is shown in Figure 4-7, an Agilent 81133A provides 4 phases of clock at 2.0 GHz. RocketIO board delivers a random data sequence at 4.0 Gb/s. The BERT 70843 synchronizes these two sources and provides a Tx-Rx phase offset with 1.0 ps precision over a 1.0 ns range.

A mild channel is applied in the testing, which includes a 30-inch coaxial cable, a bias-T with 6.0 GHz bandwidth, an SMB connector with 4.0 GHz bandwidth, a 6.4 cm long on-board trace with 30 ohm characteristic impedance\(^2\), and a low speed socket with estimated 500 MHz bandwidth at -1 dB attenuation. Figure 4-8(a) shows the on-chip eye shape observed by the symbol-spaced slicers at 4.0 Gb/s. The vertical eye reaches 320 mV at phase 0.1 UI, and closes at the phase 0.7 UI, where the SSE cannot recover the voltage any more.

For the 2-tap FSE, tuning the tap weights guarantees a 200 mV flat open eye within

---

\(^2\)Due to a design error. It is supposed to be 50 ohm.
Figure 4-7: Measurement setup. The delay between the input clock and the channel output data are tunable.

Figure 4-8: Input (a), and FSE output (b) eye opening vs. sampling phase.
±5% variation for any sampling phase, as shown in Figure 4-8(b). It shows that the FSE is able to recover the signal regardless of timing. Since the T2V converter has no rejection of the common-mode signal in time domain, the range of the tap gain is limited. For the case presented by Figure 4-8, an attenuation of $\frac{2}{3}$ is observed. Gain mismatch between tap $I_1$ and tap $I_2$ is found, which is due to the current mirror ratio mismatches. In this setup, the tap weight $I_1$ and $I_2$ are set manually for each phase. In practical applications, an adaptation engine is desired to tune the taps automatically.

This FSE design achieves 4.3 bits of linearity with monotonic gain in tap weights, as shown in Figure 4-9. This work has been presented in [51].

### 4.2 $V_M$-based FSE – 2nd generation implementation

Although the $V_{TH}$-based implementation achieves good energy-efficiency, its speed is limited by the relatively large delay of the crossing detection, and its feasible input range is constrained by the low threshold setting of the V2T converter. To solve these issues, in the 2nd generation implementation, an inverter is applied as the threshold crossing detector, which reduces the detection delay, and enlarges the feasible input range.
Figure 4-10: Architecture of the 4-way time-interleaved 2-tap 2x over-sampled receive FSE and 1-tap loop-unrolled DFE with adaptation sensors. Scan-chains and snapshots are applied for \textit{in situ} link characterization.
4.2.1 Micro-architecture

Although an FSE samples twice per symbol, each track & hold circuit operates at the symbol rate with 0.5 UI offset from its neighbors. The FSE also produces the output at the symbol rate, because the slicer runs at the same rate to match the transmitter throughput. For example, at 6.25 Gb/s, the FSE samples effectively at 12.5 GSamp/s and processes them at 6.25 Gsym/s with 5-bit linearity for an effective adaptation. However, even though all the blocks run at the symbol rate, designing such an energy-efficient front-end at this high rate is challenging. In this second generation design, we further interleave the FSE into 4 ways to relax the timing requirements.

The implemented architecture is a 4-way time-interleaved FSE, including two feed-forward taps and one feedback tap in each way, as shown in Figure 4-10. The DFE is implemented by extending the latch-based architecture [50] to a 4-way coupled loop-unrolling structure. At 6.25 Gb/s, 4 external clock phases at 3.125 GHz with quadrature phases, are converted on-chip to 8 sampling clock phases at 1.5625 GHz each with 0.5 UI phase offset, as shown in Figure 4-11.

4.2.2 $V_M$-based V2T converter

The circuit and timing diagrams of the implemented V2T converter are shown in Figure 4-12. The sampling clock $\Phi_s$ is in the same phase as $\Phi$ for the first tap and 0.5 UI earlier than $\Phi$ for the second tap so that the two samples are spaced by 0.5 UI. In comparison with Figure 3-3, an inverter chain with two sign-muxes is applied between the V2T converters and T2V converter to both sharpen the edges and derive an edge pair (including a rising edge and a falling edge) to control the differential current switch pair in the T2V block. The delay from node $V_{CST+}$ to node $T_{OF+}$ is designed to match with that from node $V_{CST+}$ to node $T_{OR+}$ so that the rising and falling edges carry the same timing information. Each output mux (sign-mux) selects the rising or falling edge as the output, depending on whether a summation or subtraction is required. All of the sign-muxes can be configured independently.

The $I_{\text{charge}}$ is generated by a 5-bit gated current DAC, which can be programmed to adjust the V2T conversion gain. In order to increase the time domain range, the evaluation clock $\Phi_{EVA}$ is offset by 0.5 UI to amortize for the delay of the first inverter along the signal path. When clock $\Phi_{EVA}$ is high, MN0 pulls node $V_{R3}$ low and the output nodes are initialized. It blocks the threshold crossings at node $V_{F2}$ earlier than the falling edge of
Figure 4-11: 8-phase-clock generator and the clock assignment for the 4-way time-interleaving architecture. At 6.25 Gb/s data rate, each input clock is at 3.125 GHz and each output clock is at 1.5625 GHz.
Figure 4-12: Schematic and timing diagram for an implemented $V_M$-based programmable V2T converter.
Figure 4-13: Schematic and timing diagram of a latch-based 4-way loop unrolling DFE with threshold-tunable sense amplifiers.

\[ \Phi_{EVA} \] Due to the intrinsic delay of the first inverter, the falling edge of \( V_{F2^+} \) is later than the falling edge of \( \Phi_{EVA} \). Therefore, it does not limit the timing range. On the other hand, it allows threshold-crossings at node \( V_{CST^+} \) to be as late as \( t_{detect,dl} \) before the rising edge of \( \Phi_{EVA} \), which extends the timing range by 0.5 \( \text{UI} \).

Due to a design error, relatively large leakage current between MPO and MNO occurs during the track & hold phase, when the MPO gate voltage approaches \( V_M \) while the MNO gate voltage is at VDD (Figure 4-12). This crowbar current represents half of the measured V2T power consumption. Substitution of MN0 with an inverter eliminates this problem, which improves the energy-efficiency of the design with no performance degradation.

### 4.2.3 T2V converter

The design of the T2V converter is nearly identical to the T2V design in the first generation (see Figure 4-4). However, in this design, the tail current sources for \( I_1 \) and \( I_2 \) are not reused among the interleaved ways to avoid the large parasitic capacitance introduced by wiring the tail node between 4 ways. As shown in Figure 4-10, the FSE tap weights \( I_1 \) and \( I_2 \) are given by two 7-bit current DACs. Tuning the current DAC code can effectively change the FSE tap values. The current DACs for \( I_1 \) of all four ways share a different external current reference than that for \( I_2 \) of all four ways, which introduces one more degree of freedom in tap tuning.
4.2.4 DFE and Adaptation Blocks Implementation

The DFE is implemented with a 4-way interleaved version of the latch-based loop-unrolling structure [50], as shown in Figure 4-13. For each interleaved way, the output of the T2V blocks goes into two sampling sense amplifiers with threshold voltages biased to $\pm \alpha$ respectively, where $\alpha$ is the feedback tap weight. Both sense amplifiers are followed by latches. These two branches share one latch multiplexer, which is controlled by the output bit from its neighbor.

Sense amplifier and latch designs are shown in Figure 4-5. By tuning the tail current $I_{TH}$ and selecting the polarity through $V_{TH-}$ and $V_{TH+}$, the sampling sense-amplifier threshold voltage is adjusted. $I_{TH}$ is generated by an on-chip 7-bit current DAC, as shown in Figure 4-10.

For each way, two input-sign sense amplifiers are placed at the input of the FSE to sense the sign of the input signal for adaptation. An additional sense-amplifier is placed in parallel with data samplers, and biased at one of the data levels to extract the sign of the error. Snapshot and scan-chains are designed to take the information out of chip for further processing, such as adaptation, statistical eye diagram characterization and BER tests. For example, the $M$-SSLMS adaptive algorithm implemented off-chip utilizes these error and data streams to optimize the FSE taps. Coefficients in all four ways can be set independently, allowing for separate gain/phase mismatch compensation.

4.2.5 Measurements

This 4-way time-interleaved $V_M$-based two-tap FSE with one-tap loop unrolling DFE is fabricated in a 90-nm process and occupies 0.03 mm$^2$ active area, as shown in Figure 4-14. Scan-chains and data snapshots are placed around the core circuit to facilitate the external adaptation and in situ system characterization. The coefficient and threshold settings are scanned into the chip to program the core circuit. The snapshots capture the received data, input signal signs and error signs to load the scan-chains for off-chip analysis and adaptation.

System level simulations show that to achieve the adaptation convergence, the DNL linearity of the taps for both the signal and the coefficients should be no less than 4.0 bits\(^3\). To test the linearity versus input voltage, the current DAC code for tap weight $I_1$ is set to

\(^3\)Otherwise, the steady state noise can reduce the eye opening by 50%.
be 40 (out of 127), the other tap \( I_2 \) is set to be zero and input differential voltage is swept from 0 to 400 mV. The error-sign sampler's threshold is swept with the help of the 7-bit reference current DAC to sense the FSE output level for each input. The DNL and INL of the FSE output voltage vs. FSE input voltage are shown in Figure 4-15 (left two plots). At 6.25 Gb/s, the worst-case DNL is about 2 LSB out of 128 quantization levels and the linearity with input voltage is about 6 bits.

To test the linearity of the FSE tap weights, the input voltage is set to be 200 mV and the FSE tap code \( I_1 \) is swept from 0 to 96. Same as in the input voltage linearity test, the error-sign sampler is applied to detect the FSE output voltage for each tap value. The DNL and INL of the FSE output voltage vs. FSE tap value are shown in Figure 4-15 (right two plots). At 6.25 Gb/s, the worst-case DNL is about -3.6 LSB out of 128 quantization levels and the linearity with FSE tap value is over 4.0 bits. Thus, the linearity with the input voltage and tap weights are both higher than 4-bit, satisfying the adaptation requirements.

Figure 4-16 shows the convergence of the \( M\text{-SSLMS} \) algorithm at two different phase positions offset by 0.5 UI, where one of the taps dominates. This algorithm makes use of the input-sign samplers and output level error-sign sampler of the FSE conditioned on the neighbor input-sign sampler's outputs, as shown in Figure 2-8. On the other hand, the conventional \( SSLMS \) algorithm diverges for the same settings.

At 6.25 Gb/s, tuning the FSE tap weights achieves a flat open eye within 5% for any delay between data and clock (Figure 4-17). In the same plot, we illustrate the large deviation of the eye opening in the SSE case by disabling the second tap. Unlike the
Figure 4-15: FSE Linearity vs. input voltage (at tap weight of 40) and tap weight (at input voltage of 200 mV) from nominal tap resolution of 7 bits.

Figure 4-16: Tap convergence for two different sampling phases with 0.5 UI offset. The sampling phases are at 0.1 UI and 0.6 UI, as shown in Figure 4-17(a).
FSE eye, which always stays open with minimal degradation, at some phases, the SSE eye collapses.

Because the adaptive engine is external and the scan-chain speed is limited (at 2 MHz rate with 992-bit length), the adaptive loop bandwidth is limited to kilohertz range. Hence, in Figure 4-18, we illustrate the simulated sinusoidal jitter tolerance (Jtol) of the FSE adaptation loop, at 5.0 Gb/s data rate with 5-bit resolution (31.25 mV per step) and different conditions. In Figure 4-18(1), the worst case eye opening is required to be no less than 200 mV, and the corresponding Jtol curves with different FSE loop bandwidth settings are demonstrated. In Figure 4-18(2), the FSE loop bandwidth is fixed at 200 MHz, and the corresponding Jtol curves with different worst case eye opening requirements are plotted.

Low bandwidth settings provide good quantization noise filtering at the expense of poor jitter tracking. On the other hand, high bandwidth settings show poor quantization noise filtering and good jitter tracking. For the mesochronous system, this provides another degree of freedom to track/filter jitter outside the CDR bandwidth. For example, the CDR bandwidth can be set high to filter out the VCO noise, while the residual jitter can be handled by the FSE adaptation loop, which can run at rates up to around 200 MHz when implemented in situ. The detailed discussion on this topic is in Section 2.4.1.
Figure 4-18: Simulated sinusoid jitter tolerance performance of a 2-tap FSE at 5.0 Gb/s with the same channel shown in Figure 2-4. (a): Jitter tolerance with different loop bandwidth (condition: the worst case eye opening is no less than 200 mV). (b): Jitter performance degradation at 200 MHz loop bandwidth.

At 6.25 Gb/s, the core circuit consumes 22.5 mW power, which includes the V2T, T2V, DFE, sign slicers and the clock tree, corresponding to 3.6 pJ/bit energy-efficiency. The power breakdown is shown in Figure 4-19. Of the 3.6 pJ/bit total power dissipation, 56 percents are consumed by the FSE clock drivers, and 34 percents are consumed by the V2T converters, including the 17 percent power taken by the direct path from supply to the ground in the track and preset phase due to the design mistake.

4.2.6 Comparison of the receive linear equalizers

Table 4.1 presents a comparison of recent receive linear FIR equalizer designs for links in the literature, including the FSE and SSE designs, with the designs presented in this thesis. In comparison with CML-based designs, which are mostly done in older technologies and rely on voltage-to-current conversion linearity with higher supplies, the voltage-time conversion technique provides better linearity performance and better energy-efficiency. The linearity of the voltage-to-current conversion technique depends on the transconductance linearity, which improves as power increases. The linearity of the VT-based design, on the other hand, is determined by the quality of the current sources, i.e. the independence to the voltage across them. Therefore, the correlation between linearity and power is relaxed. The $V_M$-based design shows 60-130% energy-efficiency improvement over CML-based designs with over 0.7 bits of linearity gain.
Figure 4-19: Simulated power partition of this FSE design @ 6.25 Gb/s.

Table 4.1: A comparison of the receive linear equalizer designs with the works of this thesis

<table>
<thead>
<tr>
<th>Works</th>
<th>[11]</th>
<th>[29]</th>
<th>[12]</th>
<th>$V_{TH}$</th>
<th>$V_M$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process</td>
<td>CMOS 130-nm</td>
<td>CMOS 250-nm</td>
<td>SiGe BiCMOS 180-nm</td>
<td>CMOS 90-nm</td>
<td>CMOS 90-nm</td>
</tr>
<tr>
<td>Supply</td>
<td>1.7 V</td>
<td>2.5 V</td>
<td>1.8 V</td>
<td>1.2 V</td>
<td>1.2</td>
</tr>
<tr>
<td>Technique</td>
<td>VI</td>
<td>VI</td>
<td>VI</td>
<td>VT</td>
<td>VT</td>
</tr>
<tr>
<td>EQ type</td>
<td>4-tap SSE</td>
<td>4-tap FSE</td>
<td>7-tap FSE</td>
<td>2-tap FSE</td>
<td>2-tap FSE</td>
</tr>
<tr>
<td>Interleaving</td>
<td>8-way</td>
<td>NO</td>
<td>NO</td>
<td>2-way</td>
<td>4-way</td>
</tr>
<tr>
<td>Speed (Gb/s)</td>
<td>8.0</td>
<td>2.5 - 3.5</td>
<td>10.0</td>
<td>4.0</td>
<td>6.25</td>
</tr>
<tr>
<td>Efficiency (pJ/bit)</td>
<td>18.9</td>
<td>38 @ 2.5G</td>
<td>4</td>
<td>2</td>
<td>3.6</td>
</tr>
<tr>
<td>$^2$Normalized Efficiency (pJ/bit)</td>
<td>8.3</td>
<td>5.9 @ 2.5G</td>
<td>–</td>
<td>2</td>
<td>3.6</td>
</tr>
<tr>
<td>Tap Linearity (bit)</td>
<td>&lt; 3.3</td>
<td>N/A</td>
<td>N/A</td>
<td>4.3</td>
<td>&gt; 4.0</td>
</tr>
<tr>
<td>Active Area (mm$^2$)</td>
<td>0.13</td>
<td>0.095</td>
<td>0.016</td>
<td>0.016</td>
<td>0.03</td>
</tr>
<tr>
<td>$^5$Normalized Active Area (mm$^2$)</td>
<td>0.066</td>
<td>0.034</td>
<td>–</td>
<td>0.016</td>
<td>0.03</td>
</tr>
</tbody>
</table>
4.3 On-chip adaptation design and estimation – 3rd generation implementation

Although the coefficient adaptation with the M-SSLMS algorithm is demonstrated in the 2nd generation design, its phase tracking ability is limited due to the low loop bandwidth, which is determined by the 2 MHz communication path with the off-chip adaptation engine. Furthermore, the bit-skipping algorithm discussed in Section 2.3 also requires a reasonably fast loop to compensate for the high frequency offset. The off-chip setup in the 2nd generation implementation definitely cannot satisfy these requirements. Therefore, an on-chip adaptation engine with bit-skipping scheme is needed to test the FSE jitter tolerance, frequency compensation performance, etc.

As a 4-way interleaved 2-tap 2x over-sampled FSE with a 1-tap DFE, the 3rd generation implementation is based on the first two implementations with on-chip coefficient adaptation and bit-skipping blocks. It also includes the V2T converter modifications discussed in Section 3.2.2, the T2V converter modifications discussed in Section 3.2.3, and the SA modification discussed in Appendix A.

4.3.1 Coefficient adaptation design

With the help of Cadence RTLCompiler and SOC Encounter, the coefficient adaptation block is implemented on the chip, using the synthesis and place-and-route ASIC flow. The block diagram for one way is shown in Figure 4-20. The signal sign slicers SS1/SS2 are aligned with the corresponding FSE taps TAP1/TAP2 by sharing the clocks respectively. An error sign slicer ES is placed at the output of the FSE to monitor the error sign situation.

The coefficient adaptation engine averages the signs of the input signals and FSE output

\[ P_{\text{Normal}} = \frac{P_{\text{Digital}} \times 1.2^2}{VDD^2} \times \frac{90 \text{ nm}}{\text{Process minimal gate length}} + \frac{P_{\text{Analog}} \times 1.2}{VDD} \times \frac{90 \text{ nm}}{\text{Process minimal gate length}}. \]

1Not presented in the paper. It is based on our best estimation.
2Normalized to the 90-nm process with a 1.2V supply as the following equation:
3From simulation.
4The digital and analog power partition is not presented in the paper. To our best estimation, 80% of the total power goes to the analog blocks.
5Normalized to the 90-nm process quadratically.
6Estimated. It occupies 4.5 mm² including the pads.
Figure 4-20: Block diagram of the on-chip coefficient adaptation block. Each tap requires several clocks and only the sampling clock is presented to address the timing.

error signals to tune the coefficients automatically:

$$W^{(k+1)} = W^{(k)} + \begin{cases} \alpha & \text{if } \sum_{i=1}^{N-1} \text{Quant}_3 \left( d^{(kN+i)} \right) \text{sgn} \left( e^{(kN+i)} \right) \geq N_{TH} \\ 0 & \text{if } N_{TH} > \sum_{i=1}^{N-1} \text{Quant}_3 \left( d^{(kN+i)} \right) \text{sgn} \left( e^{(kN+i)} \right) \geq -N_{TH} \quad (4.1) \\ -\alpha & \text{if } \sum_{i=1}^{N-1} \text{Quant}_3 \left( d^{(kN+i)} \right) \text{sgn} \left( e^{(kN+i)} \right) < -N_{TH} \end{cases}$$

where Quant$_3$ ($\cdot$) is the 3-bit quantization function introduced by the M-SSLMS algorithm, which is illustrated in Figure 2-11. The counter is disabled if the two input neighbors are with opposite signs. Thus, the adaptation is conditional and the M-SSLMS algorithm is implemented. In comparison with the 2-level quantization scheme of the conventional algorithm (Equation 2.1), the counter output is also quantized into 3 levels for coefficient adjustment. It stops the coefficients from changing when the counter output is close to zero (i.e. within $[-N_{TH}, +N_{TH}]$) and is therefore less reliable. The digital quantization level $N_{TH}$ is set by the scan-chain, which is configured by an off-chip controller.

### 4.3.2 Bit-skipping algorithm design

Although the bit-skipping algorithm switches dynamically from the FSE’s even and odd outputs, it only picks up one as the output. If two blocks are applied to generate the even and odd outputs respectively, half of the hardware is idle at any given time, wasting the power and area. Since the two blocks are identical except that the clocks are shifted by 0.5
Figure 4-21: Conceptual block diagram of the 4-way interleaved receiver with bit-skipping algorithm. Only the sampling clock is presented for each block. Clock muxes are placed locally for clear explanation. They are merged and centralized in the implementation. A correlation counter and comparator is built in way0.

UI, they can be combined into one with a mux controlling the clocks.

A conceptual block diagram is shown in Figure 4-21. For mesochronous systems, the sampling phase is fixed and the input signal sign correlation is constant. Therefore, the selection bit for the clock muxes never changes, hence it operates in the same way as the 2\textsuperscript{nd} implementation with a coefficient adaptation engine running on-chip. For plesiochronous systems, the correlation counter keeps monitoring the sampling phase. When the input signal sign correlation is higher than a preset value, the comparator output is flipped. It alters the clocks for all four ways and equivalently switches between the FSE's even and odd outputs.

### 4.3.3 Implementation estimation

As a proof-of-concept design, a 4-way interleaved 2-tap FSE with a 1-tap DFE system is built with both the on-chip coefficient adaptation and the bit-skipping mechanism. As shown in Figure 4-22, the core area is about 0.008 mm\textsuperscript{2} per way, similar to the 2\textsuperscript{nd} generation implementation. The coefficient adaptation block takes an area 0.0045 mm\textsuperscript{2} per way. The bit-skipping block takes an area of 0.0035 mm\textsuperscript{2}. 

100
According to the post-layout simulation with extracted parasitic capacitance, at 8.0 Gb/s with 1.2 V supply, the FSE core circuit operates with about 1.2 pJ/bit energy-efficiency, including the V2T, T2V, slicer and clock power. With 500 MHz loop bandwidth, the digital back-end with coefficient adaptation and bit-skipping blocks consumes about 1.2 pJ/bit power. Since these two blocks are purely digital, their energy-efficiency (with respect to the data rate) improves linearly as the loop bandwidth decreases.

A comparison of the timing recovery loop performance is shown in Table 4.2. In comparison with the most recent digitally-assisted CDR designs [53,54], this works is over 15x more energy-efficient and over 9.6x more area-efficient. In comparison with the state-of-the-art conventional CDR design [7], it achieves about 3.3x higher energy-efficiency and 8.1x better area-efficiency.
Table 4.2: A comparison of the existing CDR designs with this work

<table>
<thead>
<tr>
<th>Works</th>
<th>[53]</th>
<th>[54]</th>
<th>[42]</th>
<th>[7]</th>
<th>This work (est.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Architecture</td>
<td>ADC-based SSE &amp; CDR</td>
<td>Feedforward CDR</td>
<td>2nd order CDR &amp; DFE</td>
<td>Coefficient adapt &amp; bit-skipping FSE</td>
<td></td>
</tr>
<tr>
<td>Process</td>
<td>CMOS 65-nm</td>
<td>CMOS 65-nm</td>
<td>CMOS 250-nm</td>
<td>CMOS 90-nm</td>
<td>CMOS 90-nm</td>
</tr>
<tr>
<td>Supply (V)</td>
<td>1.0</td>
<td>–</td>
<td>2.5</td>
<td>7.12/1.0</td>
<td>1.2 V</td>
</tr>
<tr>
<td>Speed (Gb/s)</td>
<td>10.0</td>
<td>5.0</td>
<td>3.125</td>
<td>10</td>
<td>8.0</td>
</tr>
<tr>
<td>Efficiency (pJ/bit)</td>
<td>850</td>
<td>35.68</td>
<td>48</td>
<td>13</td>
<td>91.2</td>
</tr>
<tr>
<td>Normalized Efficiency (pJ/bit)</td>
<td>100</td>
<td>–</td>
<td>4.0</td>
<td>13</td>
<td>91.2</td>
</tr>
<tr>
<td>Area (mm²)</td>
<td>113</td>
<td>0.51</td>
<td>116</td>
<td>0.432</td>
<td>0.053</td>
</tr>
<tr>
<td>Normalized Area (mm²)</td>
<td>5.75</td>
<td>0.98</td>
<td>0.78</td>
<td>0.432</td>
<td>0.053</td>
</tr>
</tbody>
</table>

4.4 Summary

Based on the voltage-time conversion technique, three FSE receivers are designed and the two for mesochronous systems are measured. The 1st generation implementation is a 2-tap 2x over-sampled FSE. As the key circuit component of the filter tap, the threshold detector is an NMOS and its corresponding threshold voltage $V_{TH}$ is the detection threshold. At 4.0 Gb/s, the FSE achieves 2 pJ/bit energy-efficiency and 4.3 bits of linearity with immunity to the sampling phase. Targeting higher performance, the threshold detector is replaced by an inverter in the 2nd generation implementation and the detection threshold is raised to the inverter switching voltage $V_{M}$, which is generally around $V_{DD}/2$. The 2nd generation design operates at 6.25 Gb/s with 3.6 pJ/bit energy-efficiency and over 4.0 bits linearity with immunity to the sampling phase. The convergence of the $M$-SSLMS algorithm is verified in hardware, under real circuit nonlinearities. Both of the implementations verify that an FSE can recover both the voltage and the timing information, unifying the separate equalization

---

7 Two voltage domains at 1.2V and 1.0 V respectively.
8 Including the ADC power, which is 33 pJ/bit, excluding the DSP power.
9 Power of the coefficient adaptation and bit-skipping blocks at 500 MHz.
10 Assuming the digital circuits dominate the power consumption (based on our best estimation), normalized to the 90-nm process with a 1.2V supply as the following equation:

$$P_{\text{Normal}} = \frac{P_{\text{Digital}} \times 1.2^2}{V_{DD}^2} \times \frac{90 \text{ nm}}{\text{Process minimal gate length}}.$$ 

11 Including the pads.
12 Normalized to the 90-nm process quadratically.
block and CDR block into one unique block. In comparison with the CML-based designs, these implementations show obvious improvement of the speed-power-linearity performance.

Based on the first two implementations, the 3rd generation design integrates the coefficient adaptation control block on chip and modifies the framework for the bit-skipping scheme realization. It also includes the modified V2T converter, T2V converter and sense amplifier’s design. Based on simulations, our design operates at 8.0 Gb/s with 1.2 pJ/bit efficiency for the FSE itself and extra 1.2 pJ/bit power consumption for the two adaptation blocks.
Chapter 5

Conclusions

Voltage recovery and timing recovery loops are the two essential parts of high-speed links. As the data rate increases, the coupling between them exacerbates the link performance and challenges the link receiver design. Fractionally spaced equalization inherently combines the two loops into an FIR filter to maximize the output vertical eye opening through FIR-filtering and digital adaptation. In this thesis, we proposed to use the FSE for joint equalization and synchronization, and studied the FSE both at the system level and the circuit implementation level.

5.1 FSE system study

We first studied the phase immunity of the FSE analytically, by comparing a 2-tap FSE with a 2-tap SSE at any sampling phase. It was proved that the FSE is able to equalize an RC dominant channel and cancel all the ISI completely regardless of the phase, while the SSE can achieve the same performance only at the optimal phase.

Since the optimal FSE coefficients vary according to the sampling phase, we studied the adaptive algorithms in this thesis to tune the FSE coefficients automatically for system performance optimization. The conventional SSLMS adaptive algorithm was shown to converge to a balanced solution instead of the MMSE solution because of the quantization noise. For the worst noise case, it even diverges. To solve this issue, we developed the M-SSLMS adaptive algorithm, which effectively introduces an additional quantization level, hence reduces the quantization noise. It is implemented through signal conditioning, requiring no extra hardware.

To extend FSE applications to the plesiochronous systems, we proposed a digitally-
controlled bit-skipping algorithm to compensate for the frequency offset. It monitors the phase movement by checking the input signal sign correlation, and keeps the residual phase within sub-1 UI though dynamic selection of the FSE's two outputs. Since the residual phase can be compensated by the coefficient adaptation loop alone, the frequency offset issue is solved completely. As an alternative approach, the FSE with CDR system was also studied. It surpasses the conventional SSE with CDR system under various jitter conditions.

5.2 FSE circuit study

At the circuit level, we developed a novel VT-based filter tap for the FSE implementations. In this new FSE technique, the channel voltage is first converted into time domain and then converted back into voltage domain with weighting. Both conversions are completed by linear current integration, with all integration currents independent of the input channel voltage, which avoids the non-linear voltage-current transformation through input device transconductance.

Two voltage-to-time converter designs were discussed in this thesis. In the first design, an NMOS is applied as the threshold detector for higher energy-efficiency. For higher data rate, in the second design, an inverter is used as the threshold detector, which also increases the input range and enables the compensation for the detector threshold mismatch. A time-to-voltage converter design is also presented in this chapter, which is implemented as a digitally-controlled, current-switched integrator. Compared to the conventional CML-based implementations, our new FSE tap design shows better power-linearity performance.

Two proof-of-concept FSE chips were built in a standard 90-nm CMOS process for mesochronous applications. The 1st generation implementation is a 2-way interleaved 2-tap FSE with the $V_{TH}$-based V2T converter design. It operates at 4.0 Gb/s with 2 pJ/bit energy-efficiency and 4.3 bits of linearity, showing a flat open eye at all the sampling phases. For the sake of higher rates, the 2nd generation implementation is interleaved in 4 ways, and an inverter is used as the threshold detector in the V2T converter design. As a 2-tap FSE with a 1-tap DFE link receiver, it operates at 6.25 Gb/s with 3.6 pJ/bit energy-efficiency and over 4.0 bits of linearity. Based on the 2nd generation implementation, we not only showed the FSE's immunity to the sampling phase, but also demonstrated the convergence of the $M$-SSLMS algorithm.

We also studied the problem of applying an FSE to the plesiochronous systems in this
thesis. By dynamically muxing the clocks, the receiver jumps between the FSE's two outputs, and equivalently compensates for the frequency offset. According to our best estimation, its power consumption is much lower than the recent digitally-assisted CDR designs and the state-of-the-art conventional CDR design. At a very high update rate, the power and area footprint of these adaptation and bit-skipping blocks is comparable to that of the front-end.

5.3 Future work

In this thesis, we proposed to use FSE to do joint equalization and synchronization. Two proof-of-concept chips have been fabricated and measured to verify this idea. To understand the FSE system better and improve the FSE's performance further, there are several open problems to be solved.

At the system level, the bit-skipping algorithm was developed to compensate for frequency offset. In addition to the coefficient adaptation loop, it requires an additional sign correlation loop to monitor the phase, at the cost of the energy-efficiency and system complexity. There could be other options. For example, since the coefficients are adapted according to the sampling phase, the coefficients themselves already contain the timing information. Thus, a second order coefficient adaptation loop could potentially provide the accumulated phase information, replace the separate sign correlation loop, and simplify the FSE system. How to design such a second order coefficient adaptation loop is an open problem.

At the circuit level, a more complete model for the impact of the circuit non-idealities, such as jitter, clock feed-through, supply noise, is needed. The optimization of the coefficient adaptation speed, the bit-skipping scheme operation speed, and correlation threshold setting for better jitter tolerance performance is also worthy of future investigation.
Appendix A

A threshold tunable sense amplifier with high immunity to the input common-mode variation

A strong-arm sense amplifier (SA) boosts an analog input signal to generate a digital signal. As a common component in a wide range of IC designs, such as ADC and high-speed links receiver, it is widely applied and analyzed [55–58]. One key design issue is to compensate for the threshold offset due to the device mismatch or to bias the threshold on purpose for sensing signals with a built-in offset. Different offset compensation and biasing schemes have been reported, such as the low-speed switched-capacitor based comparator by Shih [59], high-speed programmable capacitive load comparator by Lee [60], and high-speed current-steering comparator by Kim [50]. The principle of latter two is to set imbalance at the output nodes to alter the SA’s input referred offset. However, the offset setting is vulnerable to the input common-mode variation. The impact behavior of the common-mode variation is studied in this Appendix and a circuit solution is given.

A.1 Impact of the common-mode variation

The SA input referred biased threshold $\alpha$ is sensitive to the input common-mode variation. To simplify the analysis, a capacitively loaded SA is studied in this section, which is shown in Figure A-1. An extra load $C_s$ is connected with the negative output node $V_{O-}$ to bias the input referred threshold at $\alpha$ when the input common-mode level is at $V_{in,cm}$. Then, if
the inputs are:

\[ V_{in^+} = V_{in,cm} + \alpha/2 \]
\[ V_{in^-} = V_{in,cm} - \alpha/2, \]

the SA becomes meta-stable and cannot resolve the input to a valid digital output in finite time. Hence, at any time \( t \) during the evaluation phase (clock \( \Phi \) is high), the output node voltages are substantially the same:

\[ V_{O^+}(t) \approx V_{O^-}(t). \]

It is equivalent to claim that at any time \( t \) during sensing:

\[ \frac{C_L + C_S}{C_L} \approx \frac{I_+(t)}{I_-(t)}. \] (A.1)

Assuming that the input devices are both in the saturation region, the discharge currents at time \( t \) are:

\[ I_+(t) = \frac{\mu_n C_O X}{2} \left( \frac{W}{L} \right) (V_{GS+}(t) - V_{TH})^\beta (1 + \lambda V_{DS+}(t)) \]
\[ I_-(t) = \frac{\mu_n C_O X}{2} \left( \frac{W}{L} \right) (V_{GS-}(t) - V_{TH})^\beta (1 + \lambda V_{DS-}(t)), \] (A.2)

where the power number \( \beta \) is between 1 and 2 for the advanced technologies. Combine them with Equation A.1:

\[ \frac{C_L + C_S}{C_L} \approx \frac{(V_{GS+}(t) - V_{TH})^\beta}{(V_{GS-}(t) - V_{TH})^\beta} \]
\[ = \frac{(V_{in^+} - V_{TH})^\beta}{(V_{in^-} - V_{TH})^\beta} \]
\[ = \frac{(V_{in,cm} + \alpha/2 - V_{TH})^\beta}{(V_{in,cm} - \alpha/2 - V_{TH})^\beta}. \] (A.3)
Define $\gamma = \sqrt[\theta]{\frac{C_L}{C_L + C_0}}$, then obviously $\gamma > 1$, and

$$\frac{V_{in,cm} + \alpha / 2 - V_{TH}}{V_{in,cm} - \alpha / 2 - V_{TH}} \approx \gamma.$$ 

It yields:

$$\alpha \approx \frac{2(\gamma - 1)}{\gamma + 1} (V_{in,cm} - V_{TH}). \quad (A.4)$$

Since $\gamma > 1$, it is proved that the input referred bias level $\alpha$ increases as the input common-mode voltage $V_{in,cm}$ increases. Intuitively, given the same differential input, as the input common-mode voltage increases, the discharge current difference between $I_+$ and $I_-$ increases in the absolute value, but decreases relatively to either of them. To keep the SA from generating a solid ‘1’ or ‘0’ at the output, the ratio between $I_+$ and $I_-$ must be a constant. Thus, to keep the SA output ambiguous, the differential input must increase. In other words, the input referred offset increases as the input common-mode voltage increases.
A.2 Compensation scheme for the input common-mode variation

To keep the SA bias level tunable in a wider range without occupying too much area, the current steering SA is applied in this project. The topology of a current steering biased SA is shown in A-2(1). The digital parameters $V_{TH-}$ and $V_{TH+}$ determine the bias polarity. The tail current $I_{TH}$ determines the bias amplitude by setting different initial voltages at the output nodes for the *evaluation* phase. For the output nodes to reach the cross-coupled inverter’s switching voltage at the same time, the ratio of the discharge currents on the two legs has to be a constant, which is determined by the initial voltage offset. The similar requirement can be found in Equation A.1. Thus, the conclusion in Section A.1 can be applied: given the same setup of $I_{TH}$, $V_{TH-}$ and $V_{TH+}$, the input referred bias offset increases as the input common-mode increases.

This conclusion is consistent with the simulation, which is shown in Figure A-3(1). For the bias current $I_{TH3}$, the input referred bias level varies between 125 mV and 175 mV as the input common-mode voltage varies from 900 mV to 1.10 V, corresponding to 40% fluctuation. This effect reduces the system sensitivity greatly, especially when the input common-mode signal varies in a large range.

Obviously, the compensation scheme should tune the bias current $I_{TH}$ effectively based on the input common-mode: if the input common-mode voltage is low, $I_{TH}$ increases; if the

---

1For example, the output signal of the improved T2V converter design discussed in Section 3.2.3.
input common-mode voltage is high, $I_{TH}$ decreases. Furthermore, the feed-forward loop has to be fast enough to track the common-mode voltage variation from cycle to cycle. A high-speed common-mode feed-forward compensation circuit is shown in Figure A-2(2).

The output voltages control two parallel NMOS devices, which work as resistors. Intuitively, as the input common-mode voltage increases, the parallel resistance reduces and bypasses a greater fraction of $I_{TH}$. Thus, the effective bias current for SA is smaller and the input referred offset is constant. Similarly, as the input common-mode voltage decreases, the parallel resistance increases and bypasses a smaller fraction of $I_{TH}$. Thus, the effective bias current for SA is larger and the input referred offset is unchanged. $I_{CMP}$ is another knob to control the bypass current, which makes this scheme work for a wider range of $I_{TH}$.

The compensation performance is demonstrated in Figure A-3, which shows the effect of a fixed compensation current $I_{CMP}$ for a set of bias values $I_{TH}$. It flattens the input referred offset against the input common-mode voltage. For example, after compensation, the offset (with $I_{TH3}$) is at 115 mV with fluctuation of 10 mV as the common-mode voltage varies between 900 mV and 1.10 V. In comparison with the non-compensated case, the variation is reduced from 40% to about 8.0% variation, corresponding to over 5.0X improvement.

Another possible solution is to extract the common-mode signal first and then use the extracted signal to control the compensation circuit, just like the resistive or capacitive common-mode feedback (CMFB) techniques [47]. However, in comparison with the resistor-based CMFB design, our idea avoids the extra pole introduced by the resistors, and therefore
works at higher rates. In comparison with the capacitor-based technique, this design is also more area-efficient.

Similar to the conventional current steering SA design, this new design scheme is sensitive to the supply noise because only one output is connected to the bias circuit at any time. Thus, the supply noise only impacts one output, though the feed-forward compensation path can mitigate this issue by increasing the bypass current simultaneously. The supply noise reduces the system sensitivity and impacts the minimal eye opening that the SA can demodulate. Supply noise rejection technique that further circumvents this issue can be found in [61].

A.3 Summary

A biased SA with immunity to the input common-mode voltage variation is studied in this appendix. An analytical investigation on the capacitively biased SA shows that the input referred offset increases as the input common-mode voltage increases. The conclusion is extended to the current steering SA and verified by simulations. Based on the analysis, a fast common-mode feed-forward compensation circuit is proposed.
Bibliography


