Sampled-Data IIR Filtering Via Time-Mode Signal Processing

Michael M. Guttman

Department of Electrical and Computer Engineering

McGill University

Montréal, Québec

July 2009

A thesis submitted to McGill University in partial fulfillment of the requirements of the degree of Master of Engineering

ACKNOWLEDGMENTS

First, I would like to thank my supervisor, Professor Gordon Roberts, for giving me the opportunity to study in his research group. I would not have been able to complete this thesis without his guidance, his enthusiasm, and his dedication. He has always been able and available to offer me both professional and academic advice, for which I will always be grateful. To Dong (Hudson) An, I would like to offer my appreciation for his advice and support in the preparation and testing of my design. His expertise made my experience that much more informative and clear. To all my friends at McGill, I thank you for making my experience both exciting, enjoyable, and at times, intellectually-stimulating. I would like to thank my parents Joseph and Hinda, as well as my brother Robert for their love and support. Finally, I would like to acknowledge the Natural Sciences and Engineering Research Council of Canada (NSERC) for their financial support, as well as the Canadian Microelectronics Corporation (CMC) for the use of their integrated circuit fabrication and testing facilities.

ABSTRACT

In this work, the design of sampled-data infinite impulse response filters based on time-mode signal processing circuits is presented. Time-mode signal processing (TMSP), defined as the processing of sampled analog information using time-difference variables, has become one of the more popular emerging technologies in circuit design. As TMSP is still relatively new, there is still much development needed to extend the technology into a general signal-processing tool. In this work, a set of general building block will be introduced that perform the most basic mathematical operations in the time-mode. By arranging these basic structures, higher-order time-mode systems, specifically, time-mode filters, will be realized. Three second-order time-mode filters (low-pass, band-reject, high-pass) are modeled using MATLAB, and simulated in Spectre to verify the design methodology. Finally, a damped integrator and a second-order low-pass time-mode IIR filter are both implemented using discrete components.

ABRÉGÉ

Dans ce mémoire, la conception de filtres de données-échantillonnées ayant une réponse impulsionnelle infinie basée sur le traitement de signal en mode temporel est présentée. Le traitement de signal dans le domaine temporel (TSDT), définie comme étant le traitement d'information analogique échantillonnée en utilisant des différences de temps comme variables, est devenu une des techniques émergentes de conception de circuits des plus populaires. Puisque le TSDT est toujours relativement récent, il y a encore beaucoup de développements requis pour étendre cette technologie comme un outil de traitement de signal général. Dans cette recherche, un ensemble de blocs d'assemblage capable de réaliser la plupart des opérations mathématiques dans le domaine temporel sera introduit. En arrangeant ces structures élémentaires, des systèmes en mode temporel d'ordre élevé, plus spécifiquement des filtres en mode temporel, seront réalisés. Trois filtres de deuxième ordre dans le domaine temporel (passe-bas, passe-bande et passe-haut) sont modélisés sur MATLAB et simulé sur Spectre afin de vérifier la méthodologie de conception. Finalement, un intégrateur amorti et un filtre passe-bas IIR de deuxième ordre en mode temporel sont implémentés avec des composantes discrètes.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS

$\Delta\Sigma$: Delta Sigma

AC: Alternating Current

ADC: Analog-to-Digital Converter

AWG: Arbitrary Wave Generator

BRTF: Band-Reject Time-Mode Filter

CMOS: Complementary Metal Oxide Semiconductor

DC: Direct Current

HPTF: High-Pass Time-Mode Filter

IC: Integrated Circuit

IIR: Infinite Impulse Response

LPTF: Low-Pass Time-Mode Filter

PCB: Printed Circuit Board

PSD: Power Spectral Density

SDR: Signal-to-Distortion Ratio

SNDR: Signal-to-Noise and Distortion Ratio

SNR: Signal-to-Noise Ratio

SoC: System-on-Chip

TAMP: Time-Mode Amplifier

TATN: Time-Mode Attenuator

TDC: Time-to-Digital Converter

TMSP: Time-Mode Signal Processing

TMU: Time-Measurement Unit

TVC: Time-to-Voltage Converter

VCDU: Voltage-Controlled Delay Unit

VTC: Voltage-to-Time Converter

WDTA: Weighted Delayed Time-Mode Adder

WDTS: Weighted Delayed Time-Mode Subtractor

Chapter 1

Introduction

1.1 Motivation

Over the years, it is the demand for full electronic systems on a single chip (i.e. SoC) that has been the driving force behind the evolution of CMOS technology. This has prompted the integrated circuit (IC) industry to develop transistors with smaller dimensions to allow for the integration of a greater number of analog, digital and mixed-signal circuits onto a single piece of silicon. In addition to the increased packing density, the shrinking of CMOS technology has led to an increase in digital switching speeds, as well as a lowering of the supply voltage.

The rapid shrinking of CMOS technology is not without its challenges. With the shrinking of transistor dimensions comes the reduction of the transistor gate oxide thickness, resulting in an increase of the leakage current through the transistor gate [1]. The lowering of the supply voltage has lead to reduced voltage swings, limiting our the ability to stack transistors in order to obtain reduced sensitivity as well as other advantageous properties. A decreased voltage range also impacts analog and digital circuit performance by increasing their susceptibility to voltage

domain and switching noise. Finally, the decrease in the supply voltage has been offset by the need for larger driving currents in many of these high speed designs, and as a result, the power dissipation of these circuits has shown little improvement.

As a result of these factors, there is a need to redesign conventional circuits for each emerging technology, specifically those that are analog or sampled-data in nature. For the IC industry, this means increased costs for the development and testing of new implementations of the same circuits. In order to solve some of these design challenges, a new circuit technology is being investigated, referred to as time-mode signal processing or TMSP.

With TMSP, sampled analog information is instead carried by the time-difference between two digital signals, known as time-difference variables. In doing this, the traditional sampled-data system composed of analog components can be replaced with a design based on digital-like circuits. These designs will offer all the benefits of digital CMOS technology, specifically, the ability to operate at higher speeds, while consuming less power and less silicon area. Furthermore, since propagation delay is inherent in all materials, and since there is a one-to-one correspondence between voltage signals and time signals, there is the potential for TMSP to provide a means of implementing *any* sampled-data analog signal processing function in *any* technology. With that said, it is important to note that this work sets out to achieve a small step in this general direction. Specifically, this thesis demonstrates for the first time how to implement infinite-impulse response (IIR) filtering algorithms based on time-difference variables. In the process of demonstrating this technique, an analog circuit is used in the prototyped rather than one based on a digital-like circuit. This does not diminish the overall objective of time-mode signal processing but simply tackles

one issue at a time. Finally, the fact that the time resolution resulting from the propagation delay of a transistor (researchers have reported maximum jitter values in the picoseconds range [2]) is superior to any voltage resolution achievable on-chip is a very attractive property for the design of high performance, high resolution measurement systems [3].

TMSP is a sampled-data technique. As the method is general it can be applied to any type of sampled-data application such as those presently implemented using the switched-capacitor technique. Such sampled-data circuits are used in part as anti-aliasing and anti-imaging filters for data conversion, as well as part of the data conversion circuit, to name just a few applications.

Several works have begun to investigate the potential of utilizing time-mode signals in electronic systems, including the designs of a voltage comparator, a single slope and a delta-sigma analog-to-digital converter (ADC), a finite impulse response (FIR) filter, as well as circuits that convert between the voltage and time domains. Most of the systems listed are designed to perform a distinct function, that is, there are very few blocks in these circuits that can be used to design other types of circuits. As TMSP is still relatively new, the technology does not include the same basic building blocks that are available in analog signal processing. In this work, the inherent property of delay is generalized in order to design a broad set of building blocks that perform basic signal processing operations in the time-mode. These blocks will then be organized to construct second-order time-mode infinite impulse response (IIR) filters. Finally, the sections will be cascaded, allowing us to obtain a methodology for higher-order IIR filter design.

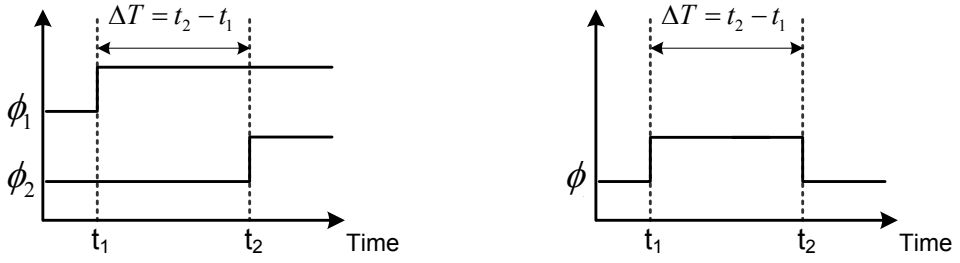


Figure 1–1: Representing time-difference signals: (a) two-signal representation, and (b) one-signal representation.

1.2 Time-Mode Signal Representation

The basic unit of TMSP, known as a time-difference variable, ΔT, is defined as the quantity of time between one event and a second reference event or time, where an event refers to digital clock transition, that is, either a rising clock edge (0-to-1 transition) or a falling clock edge (1-to-0 transition). Figure 1–1 demonstrates two ways of representing time-differences. In (a), the time interval or phase difference between two clock edges is used to represent the time-difference ΔT. In (b), time-difference is measured by the duration of a pulse. In the context of this work, the representation in Figure 1–1(a) will be used to represent time-mode signals and differences. Note that although these events are digital in nature, the time-mode variables they define are actually discrete samples of analog information.

We will henceforth refer to the time-different variable $\Delta T(n)$ as the quantity of time between a *rising edge* of one clock signal ϕ with respect to the *rising edge* of a second reference clock ϕ_{REF} at instance n. The sampling instance n will refer to the discrete number of periods of the reference clock (running at some sampling frequency F_S).

1.3 Thesis Overview

The remainder of this thesis is organized as follows. In Chapter 2, an overview of past works in the field of TMSP will be summarized. This includes the design of cicrcuits that manipulate information directly in the time-mode, as well as those that convert information between the time-mode and the voltage and digital domains. In Chapter 3, the basic TMSP building blocks used to construct time-mode IIR filters are described. The goal of this work is to demonstrate how these basic building blocks can be combined to design higher-order time-mode systems, and thus, Chapter 4 focuses on the design methodology of second-order and higher-order time-mode filters. This chapter will also present the MATLAB and simulation results of the transistor level schematic in a 0.18-μm process, as well as an implementation of both a damped integrator circuit as well as a second-order low-pass time-mode IIR filter using discrete components. Finally, Chapter 5 offers a summary of this work and a discussion of future advancements of this study.

Chapter 2

Time-Mode Circuits and Applications

In the sections that follow, the current designs which implement TMSP will be described. This includes those circuits that manipulate information directly in the time-mode, as well as those that convert information between the time domain and the analog or digital domain.

2.1 Delay Elements

The fundamental operation in time-mode signal processing is the delay of time information. Those elements that produce delay can be classified into two different flavours: constant delay units and controlled delay units. Constant delay units are circuits in which the input-output propagation delay is a single value. Controlled delay units, on the other hand, provide more flexibility by allowing the designer to vary the delay over a specific range through a control variable such as a voltage or digital logic. The block diagrams that will be used to represent these type of circuits, as well as their time-domain signal operation, are illustrated in Figures 2–1(a) and 2–1(b). A variety of delay elements have been proposed in previous works [4]. This

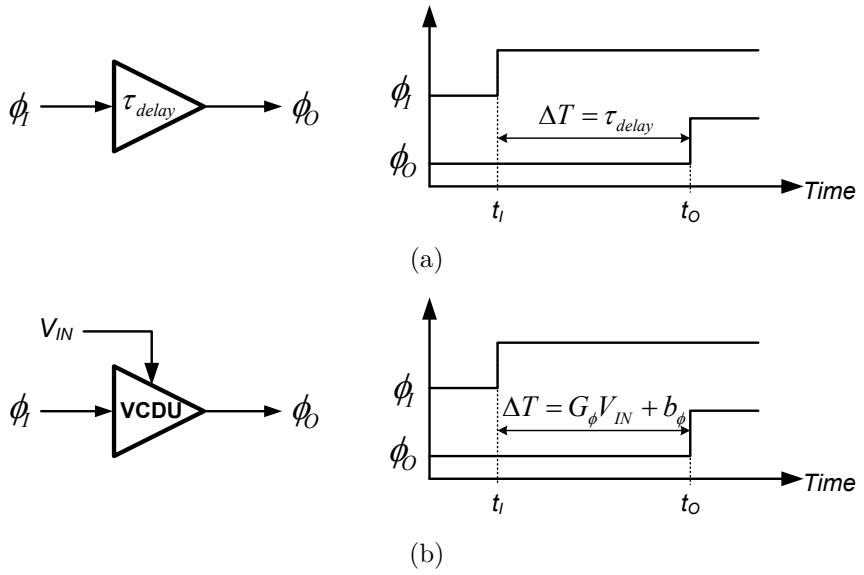


Figure 2–1: Block diagram and signal representation for (a) constant delay units, and (b) linear voltage-controlled delay units.

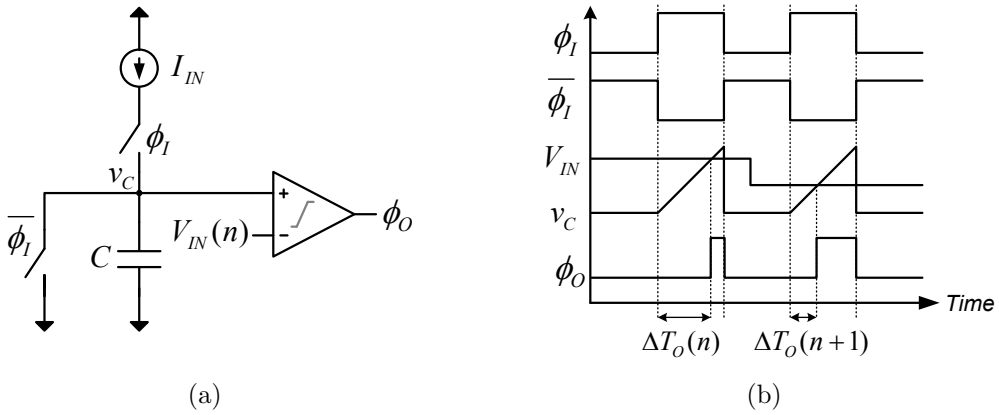


Figure 2–2: Voltage-controlled delay unit (VCDU) (a) functional diagram, and (b) signal representation.

section will focus on those circuits capable of producing controlled linear delays, otherwise known as Voltage Controlled Delay Units (VCDUs).

2.1.1 Direct Voltage-Controlled VCDU

A direct voltage-controlled VCDU can be constructed using a current source, a capacitor, and an asynchronous voltage comparator as illustrated in Figure 2–2(a). Initially ϕ_I is low, resetting the capacitor voltage to ground and thus forcing ϕ_O into a logic low state. On the rising edge of the input ϕ_I, the capacitor C will charge up linearly by the constant current source I_{IN}, resulting in a voltage ramp at the node v_C:

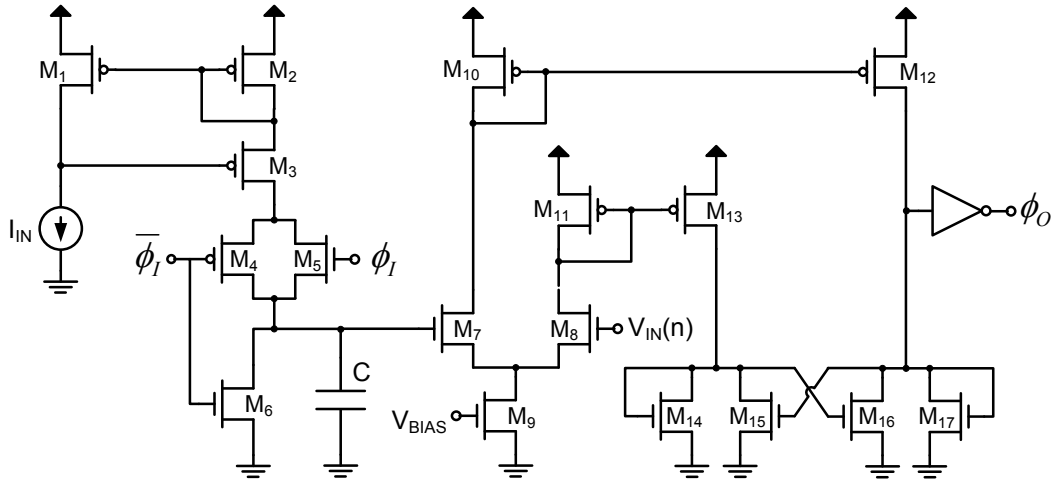
$$v_c(t) = \int_0^t \frac{I_{IN}(t)}{C(t)} dt = \frac{I_{IN}}{C} \int_0^t dt = \frac{I_{IN}}{C} t \tag{2.1}$$

The voltage comparator, which senses the difference between the input reference voltage and the capacitor voltage, will at its output switch from a logic 0 to a logic 1 when $v_C(t)$ reaches $V_{IN}(n)$. The equation which describes the conversion of the input voltage $V_{IN}(n)$ into the output time-difference $\Delta T_O(n)$ is

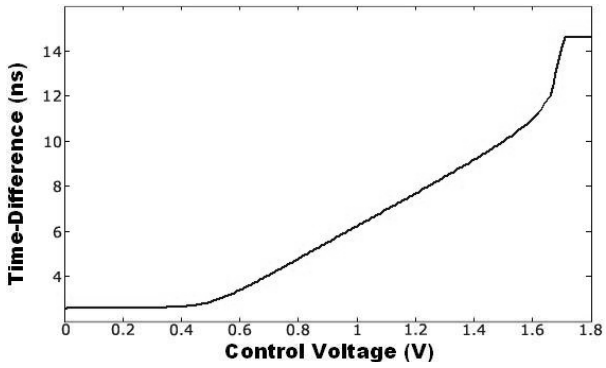
$$\Delta T_O(n) = \frac{C}{I_{IN}} V_{IN}(n) = G_\phi \cdot V_{IN}(n) \tag{2.2}$$

where here the voltage-to-time conversion factor G_ϕ is the ratio of the current and capacitance, $G_\phi = C/I_{IN}$. The operation of the VCDU is illustrated in the time and voltage signal graph of Figure 2–2(b). During the low phase of the input clock ϕI, the capacitor voltage v_C is reset to ground. When the rising edge of ϕI arrives, the voltage v_C begins to rise linearly with a slope $1/G_\phi$. When the voltage v_C reaches the input $V_{IN}(n)$, the comparator output switches from zero to one, producing an edge at ϕO.

An example of a transistor level VCDU using the direct voltage-controlled strategy is illustrated in the current-steering design of Figure 2–3(a). A Wilson current mirror, comprised of transistors M1-M3, is used to generate the constant current I_{IN}.



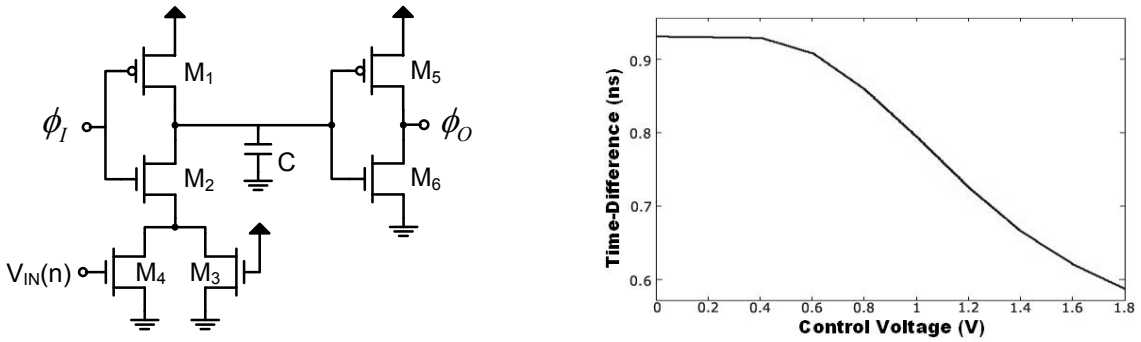
(a)



(b)

Figure 2–3: Current-steered VCDU (a) transistor schematic; (b) transfer characteristic.

Initially, when ϕ_I is low, the capacitor C is reset to ground through the transistor M6. On the rising edge of the input ϕ_I, the capacitor C will begin charging through the transmission gate formed by M4 and M5. The comparator is implemented via a current-steering amplifier (transistors M7-M13) which senses the difference between $V_{IN}(n)$ and the voltage on the capacitor C. The final logic decision is made by the output latching circuit (M14-17).



(a) (b)

Figure 2–4: Current-starved VCDU (a) transistor schematic; (b) transfer characteristic.

The transfer characteristic of a current-steered VCDU is illustrated in Figure 2–3(b). As depicted in the transfer characteristic, direct voltage-controlled VCDUs offer the advantages of very good linearity (errors smaller than 1% have been reported) and a good input voltage range (0.6 V - 1.4 V). These advantages are attributed to the fact that the capacitance and charging current are strongly independent of the input signal and the charging voltage v_C. Also note that the voltage-to-time conversion gain is positive for this design, which is expected since the ramping voltage v_C will take longer to surpass a higher input reference voltage.

2.1.2 Current-Controlled VCDU

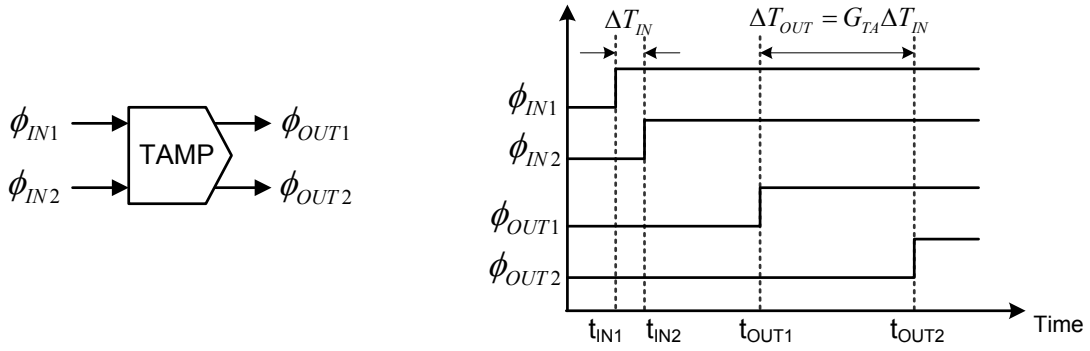
Many different designs for current-controlled VCDUs (also known as current-starved VCDUs) have been proposed. For example, researchers have investigated the design of these circuits in which the delay of the cell is controlled directly by the current [5], or indirectly by an analog voltage [6], via digital control bits [7], or by the sizing of a load transistor [8].

A current-controlled VCDU can be constructed with two inverter-type circuits, as illustrated in Figure 2–4(a). This first inverter circuit is voltage-controlled such that the propagation delay is proportional to the control voltage V_{IN}, while the second inverter acts as the comparator with a fixed threshold of V_{TH}. When the input edge ϕ_I is low, the capacitor C is pulled up to the supply rail resulting in a logic 0 at the output ϕ_O. On the rising edge of ϕ_I, the capacitor begins to discharge through the load transistors M3 and M4 at a rate that is controlled by the input voltage V_{IN}. The second stage inverter compares the voltage on the capacitor to the inverter threshold, producing a rising edge when the voltage over C drops below V_{TH}.

The transfer characteristic of a current-steered VCDU is illustrated in Figure 2–4(b). Unlike the voltage-controlled VCDUs, the current starved VCDU has a negative voltage-to-time conversion gain. While the linearity of the current starved VCDU has a smaller range than the voltage-controlled VCDU in Figure 2–3(a), it does boast better power consumption, smaller silicon area usage, and a higher bandwidth of operation.

2.2 Time Amplifier

In much the same manner as programmable gain amplifiers are used to increase the resolution of an ADC, time amplification can be used as a pre-processing step to improve the dynamic range of many time-mode systems. A time amplifier, depicted in Figure 2–5(a), is a circuit that takes two input time-mode signals and produces two outputs which differ in time by a scaled version of the input time-difference. The operation of time-domain amplification can thus be described by the difference



(a) (b)

Figure 2–5: Time amplifier (a) block diagram; (b) timing diagram.

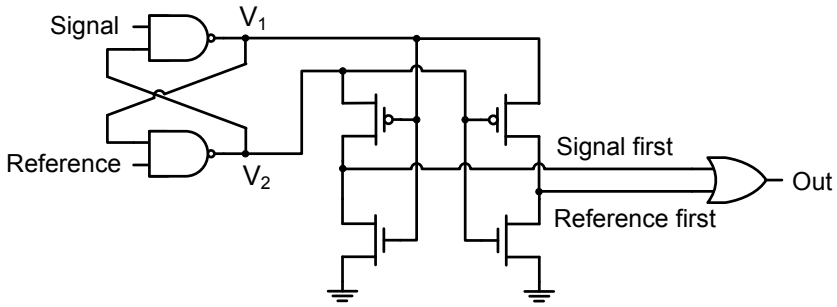


Figure 2–6: A MUTEX time amplifier.

equation

$$\Delta T_O(n) = G_{TA}\Delta T_{IN}(n) \tag{2.3}$$

where G_{TA} is the gain of the time amplifier and ΔT_I and ΔT_O are the time-differences between the rising edges of the two inputs ϕ_{IN1}, ϕ_{IN2} and the two outputs ϕ_{OUT1}, ϕ_{OUT2}, respectively. The process of time amplification is depicted in the timing diagram of Figure 2–5(b).

The first time amplifier [9] proposed was based on the mutual-exclusive circuit illustrated in Figure 2–6. In this circuit, the cross-coupled NAND gates form a

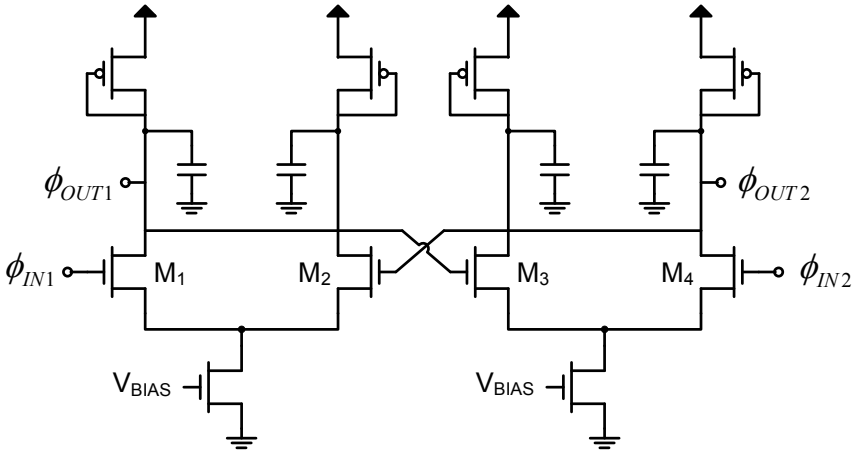


Figure 2–7: A single-stage time amplifier.

bistable circuit while the output transistors switch for a certain difference ΔV between voltages V_1 and V_2. This switching action is detected by the OR-gate at the output. When the time difference at the input is small enough, meta-stability is caused in the bistable, resulting in time amplification.

A second method proposed for time amplification [10] is shown in Figure 2–7. This circuit is composed of two cross-coupled differential pairs, each loaded by a capacitance and resistance. Upon arrival of the rising edges of ϕ_{IN1} and ϕ_{IN2} the amplifier bias current is steered around the differential pairs and into the passive loads. As a result, the voltage at the drains of M1 and M2 will be equal at a specific time, and that of M3 and M4 at a short time later. By detecting these voltages with a voltage comparator, a time-difference can be produced which is proportional to the input time-difference.

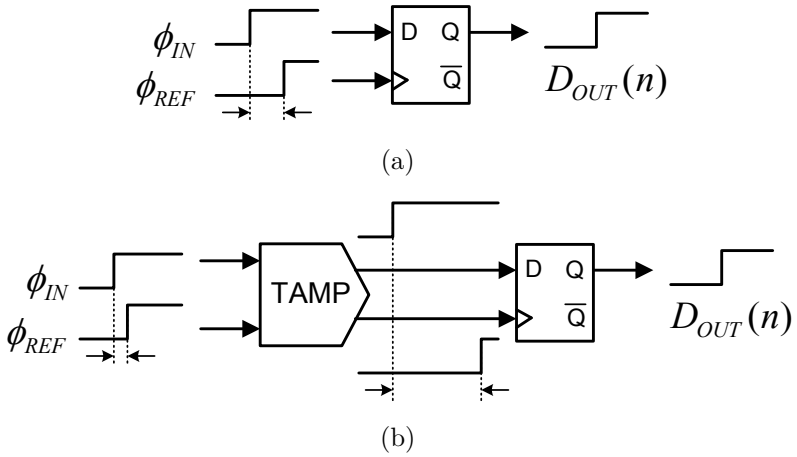


Figure 2–8: Time-mode comparator implemented using (a) a D-type flip-flop (b) and a time amplifier for pre-amplification.

2.3 Time-Based Comparators

A time-mode comparator can be implemented using an edge-triggered D flip-flop, as illustrated in Figure 2–8(a). The circuit compares the rising edge of the input ϕ_{IN} to that of the reference ϕ_{REF} and outputs a logic 1 if ϕ_{IN} is leading the reference, or a logic 0 if ϕ_{IN} is lagging it.

Similar to a voltage comparator, the resolution of the circuit in Figure 2–8(a) will be limited by the metastability of the flip-flop when attempting to make a correct decision between two edges which are close together. An approach which reduces the effects of flip-flop metastability thus leading to greater resolution is illustrated in Figure 2–8(b). In this circuit, the input time-mode signals are first fed through a time amplifier, where their time-difference is first stretched. The new larger time-difference is then passed to the arbiter (D flip-flop) which easily make the correct logic decision.

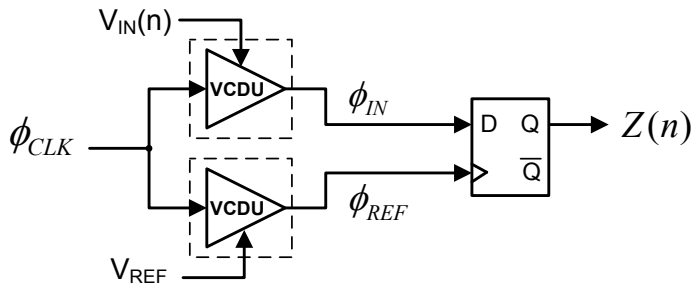


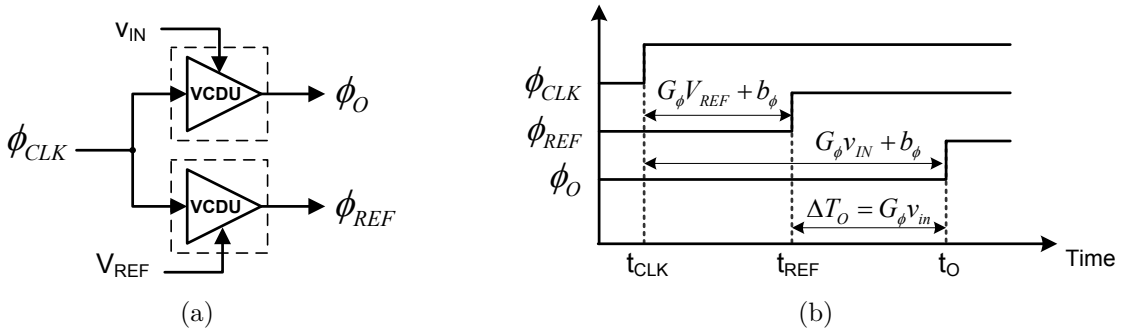
Figure 2–9: Time-mode voltage comparator.

By cascading two VCDUs to a D-type edge-triggered flip-flop, a time-mode voltage comparator [11] can be implemented. Shown in Figure 2–9, this circuit operates by first converting the two input voltages into delayed versions of a common clock ϕ_{CLK} through the two VCDUs. The two resulting time-mode signals, delayed proportionally to their respective control voltages, are then compared using the time comparator described above. Assuming that ΔT is greater than the metastability window of the flip-flop, the circuit will produce a correct logic output. Furthermore, a time amplifier may be added to this circuit to minmize the effects of this metastability. Note that to obtain higher resolution, calibration is necessary to ensure good matching of the voltage-to-time transfer characteristics of the two VCDUs. With calibration, the circuit in Figure 2–9 has achieved 8-bits of resolution while consuming only 0.9 mW of power [11].

2.4 Time-to-Voltage/Voltage-to-Time Circuits

2.4.1 Time-to-Voltage Converter

A time-to-voltage converter (TVC) is used to convert the time-difference between two edges into a voltage variable that can then be later converted into digital



(a) (b)

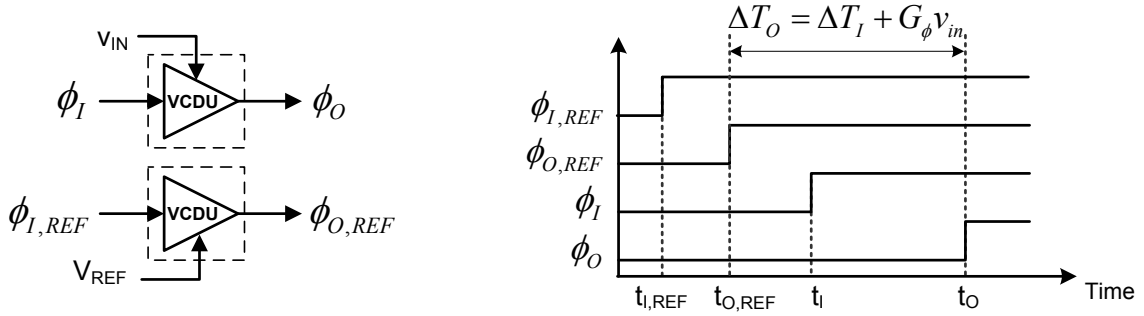
Figure 2–10: Voltage-to-Time Converter (a) functional diagram, and (b) timing diagram.

information through an ADC [12][13] or time-measurement unit (TMU) [14]. In these systems, the width of the input pulse controls how long a capacitor C is charged by a current I through transistor switches. The voltage change on the capacitor is then captured by either an ADC or a TMU and stored in digital form.

2.4.2 Voltage-to-Time Converter

For all voltage applications, a voltage-to-time converter (VTC) is first required to transform input voltage date into the time domain before processing. This is achieved using two VCDUs with a single common input clock ϕ_{CLK} and two outputs, ϕ_{REF} and ϕ_O, as illustrated in Figure 2–10(a). The top VCDU converts the input voltage $v_{IN}(n)$ into a time-mode signal while the bottom VCDU generates the reference time variable from the analog reference V_{REF}.

The timing diagram in Figure 2–10(b) illustrates how the input clock is delayed with respect to both the input and reference voltages, resulting in two output signals ϕ_O and ϕ_{REF}. The edge transitions for these outputs can be written in terms of the



(a) (b)

Figure 2–11: Voltage-to-Time Adder/Subtractor (a) block diagram, and (b) timing diagram.

rising edge of the input clock as:

$$t_O = t_{CLK} + G_\phi v_{IN} + b_\phi \tag{2.4}$$

$$t_{REF} = t_{CLK} + G_\phi V_{REF} + b_\phi \tag{2.5}$$

By subtracting Equations (2.4) and (2.5), the resulting time-difference equation for the VTC is obtained:

$$\Delta T_O = t_O - t_{REF} = G_\phi v_{in} = G_\phi (v_{IN} - V_{REF}) \tag{2.6}$$

where v_{in} represents the difference between the input and reference voltages $v_{IN} - V_{REF}$.

2.4.3 Voltage-to-Time Adder/Subtractor

Similarly to a VTC, a voltage-to-time adder/subtractor is implemented with two VCDUs as shown in Figure 2–11(a). In this case, instead of a common input clock, the VCDUs are fed with two input signals ϕ_I and $\phi_{I,REF}$ which are delayed an amount proportional to the two input voltages v_{IN} and V_{REF} respectively. The

edge transition equations for the two outputs can be written as:

$$t_O = t_I + G_\phi v_{IN} + b_\phi \tag{2.7}$$

$$t_{O,REF} = t_{I,REF} + G_\phi V_{REF} + b_\phi \tag{2.8}$$

By taking the difference between the two output signals, the output time delay is:

$$\Delta T_O = t_O - t_{O,REF} = t_I - t_{I,REF} + G_\phi (v_{IN,1} - V_{REF}) \tag{2.9}$$

The expression above can be re-written in terms of the time-difference at the input of the VCDUs and the difference between the two input voltages:

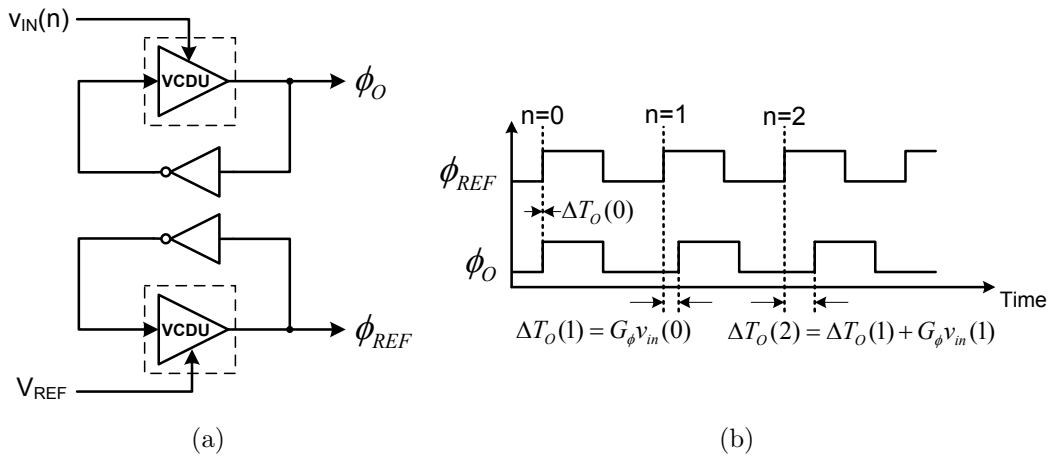
$$\Delta T_O = \Delta T_{IN} + G_\phi v_{in} \tag{2.10}$$

where $\Delta T_{IN} = t_I - t_{I,REF}$ and $v_{in} = v_{IN} - V_{REF}$.

If positive gain VCDUs are implemented in Figure 2–11(a) above such that $G_\phi > 0$, Equation (2.10) describes a voltage-to-time adder, while using negative gain VCDUs ($G_\phi < 0$) implements a voltage-to-time subtractor. The timing diagram for the voltage-to-time adder is shown in Figure 2–11(b).

2.4.4 Voltage-to-Time Integrator

A circuit that performs voltage-to-time integration is shown in Figure 2–12(a). The circuit is equivalent to a voltage-to-time adder/subtractor in which the outputs of the VCDUs are fed back to the inputs via an inverter [15]. The VCDUs are implemented by taking the current-starved topology depicted in Figure 2–4(a) and AND-ing the output with the input signal ϕ_I to eliminate any dependency of the falling edge's propagation delay on the control voltage.



(a) (b)

Figure 2–12: Voltage-to-time Integrator (a) block diagram, and (b) timing diagram.

The operation of the circuit is demonstrated in Figure 2–12(b). At the first sampling instance, the output of the integrator, $\Delta T_O(0)$, is set to zero. This time-difference is fed back to the VCDU, and during the following time instance ($n = 1$), is delayed by the input voltage at that instance $v_{IN}(1)$ scaled by G_ϕ. The operation of delaying proportionally to the input voltage and feeding back the signal can be described by the difference equation for voltage-to-time integration:

$$\Delta T_O(n) = \Delta T_O(n-1) + G_\phi v_{in}(n-1) \tag{2.11}$$

Note that the circuits in Figure 2–12(a) are equivalent to two voltage controlled ring oscillators, and thus there is no external clock signal required in the system. The frequency of operation is instead determined by the device sizing of the transistors in the blocks.

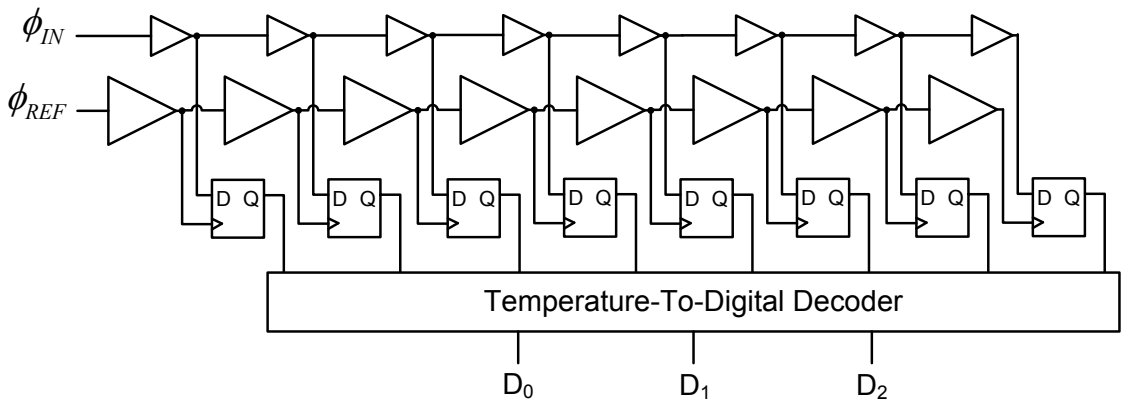


Figure 2–13: 3-bit Vernier Delay Flash Time-to-Digital Converter.

2.5 Time-to-Digital Converter

A time-to-digital converter (TDC) is essential to time-mode processing because it allows for the conversion of very small time-differences which are difficult to measure with modern-day test equipment into digital numbers which are easily captured. This circuit is used extensively as the core building block of many high speed measurement systems.

A simple and popular TDC [16] known as a Vernier delay flash converter is shown in Figure 2–13. This circuit operates by comparing time-delayed versions of the input signal to time-delayed versions of the reference clock. The delays of the buffers in the signal path are designed to be slightly smaller than those in the reference edge path. Thus at each adjacent stage the input signal will be compared with a reference edge which has been displaced by one additional small time interval. The flip-flop outputs create a thermometer code which is converted into a digital code and represents the time-difference between the reference and signal edges.

In general, the delay stages utilized in the Vernier delay line are voltage controlled to allow for a range of tune-ability. The delay blocks are as well arranged

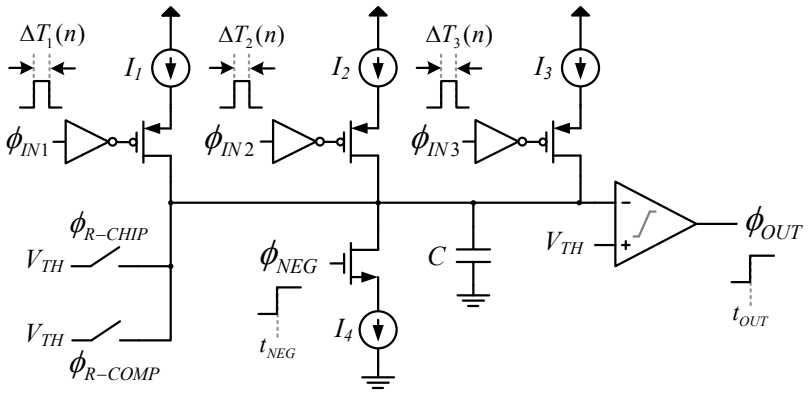


Figure 2–14: 3-tap Finite Impulse Response Filter.

in a negative feedback configuration, known as a delay-locked loop, which is much more robust to noise and jitter effects. The circuit in Figure 2–13 is just one of the many TDC designs that have been proposed. Many different styles exist, such as those described in [17], [18] and [19].

2.6 Finite Impulse Response Filter

The difference equation that defines the output $y(n)$ of a FIR filter in terms of its input $x(n)$ is given by the relation

$$y(n) = b_0x(n) + b_1x(n-1) + ... + b_Nx(n-N) \tag{2.12}$$

where b_i represent the coefficients and N the order of the filter. An implementation of a time-mode FIR filter, originally presented in [20] (the authors note that it is also in this study that the term "time-mode signal processing" first appears), is shown in Figure 2–14. The structure is based on the direct voltage controlled VCDU circuit presented earlier in this chapter (Figure 2–2(a)), except that in this case, there are multiple branches connected to one of the comparator terminals. Each of the top

branches, in which a time-mode signal ϕ_{INx} controls the charging of the capacitor by a current I_x, represents a single tap of the filter. During the first stage of operation, the signals ϕ_{1-3} turn on the three transistors, causing the capacitor to be charged by the currents I_{1-3} for the duration of the pulses ΔT_{1-3}. The final voltage over the capacitor is:

$$V_C = V_{TH} + \frac{1}{C}(I_1\Delta T_1 + I_2\Delta T_2 + I_3\Delta T_3) \tag{2.13}$$

Next, the rising edge of ϕ_{NEG} occurs at $t = t_{NEG}$, and the capacitor begins to discharge by the current I_4. The output ϕ_{OUT} remains low until the capacitor voltage reaches V_{TH} at some $t = t_{OUT}$, at which point the comparator output switches to a high logic state (producing a step at the output). In this case, the output time-difference, defined between the rising edges of the signals ϕ_{NEG} and ϕ_{OUT}, can be written as:

$$\Delta T_{OUT} = t_{OUT} - t_{NEG} = \frac{I_1}{I_4}\Delta T_1 + \frac{I_2}{I_4}\Delta T_2 + \frac{I_3}{I_4}\Delta T_3 \tag{2.14}$$

To implement a FIR filter, the inputs in Figure 2–14 are fed with different delayed versions of a common signal ϕ_{IN}. Each subsequent input is delayed exactly one sample from the previous input. Therefore, if M_1 is switched on for the duration of the pulse of ϕ_{IN} at sampling instance n, then M_2 and M_3 will be fed by the corresponding pulse at instances $(n-1)$ and $(n-2)$ respectively. Replacing these into Equation (2.14) above yields

$$\Delta T_{OUT} = \frac{I_1}{I_4}\Delta T_{IN}(n) + \frac{I_2}{I_4}\Delta T_{IN}(n-1) + \frac{I_3}{I_4}\Delta T_{IN}(n-2) \tag{2.15}$$

which is the time-mode equivalent of a second-order FIR filter as described at the beginning of this section. The extension of this theory to design FIR filters of higher order simply requires an extra sourcing branch to be added for each additional term

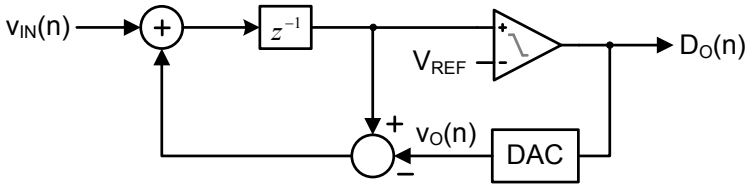


Figure 2–15: First-order single-bit $\Delta\Sigma$ ADC error-feedback model.

in the equation (and N-order filter requires $N+1$ taps). Note that the circuit implementation of Figure 2–14 will require additional computational blocks to implement required to produce the delayed outputs $\Delta T_{IN}(n-1)$, $\Delta T_{IN}(n-2)$, etc.

2.7 Delta-Sigma Analog-to-Digital Converter

A time-mode delta-sigma ADC can be designed [15] based on the block diagram of the error-feedback structure for a $\Delta\Sigma$ modulator which is illustrated in Figure 2–15. This system is composed of a dual-input integrator, a delay cell, and a voltage comparator. To implement a time-mode $\Delta\Sigma$ modulator, we replace the traditional blocks with their time-mode equivalents, that is, the dual-input integrator is replaced by the voltage-to-time integrator with dual inputs, while the voltage comparator is replaced with a time-mode comparator. Finally, if full swing voltages are being used, the DAC may be eliminated.

The block diagram of the first-order time-mode $\Delta\Sigma$ modulator is shown in Figure 2–16(a) and consists of two dual-input integrators and a D-type edge-triggered flip-flop. The bottom dual-input integrator is required to create the reference phase ϕ_{REF} while the top integrator sums the input voltage with the inverse of the digital output voltage. The equivalent full circuit schematic, depicted in Figure 2–16(b), is obtained by replacing the current-starved VCDU of Figure 2–4(a) into the block

diagram. The time-difference equation which describes the operation of this circuit is

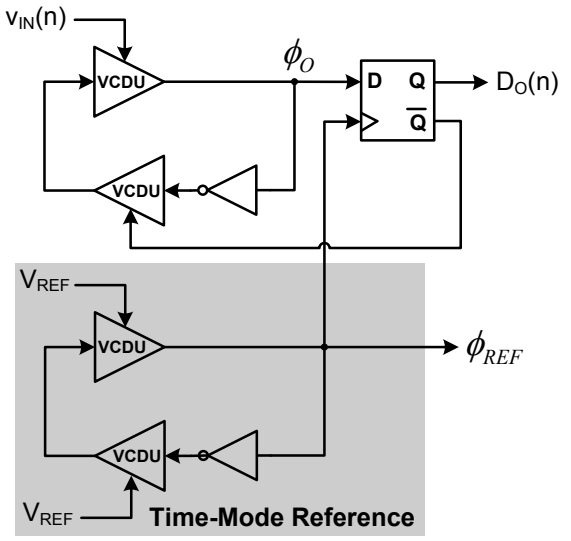
$$v_O(n) = v_{IN}(n-1) + \frac{1}{G_\phi}(\Delta T_\epsilon(n) - \Delta T_\epsilon(n-1)) \tag{2.16}$$

where G_ϕ is the voltage-to-time conversion factor of the VCDU and $\Delta T_\epsilon(n)$ is the error signal made by the flip-flop comparison with respect to the time reference.

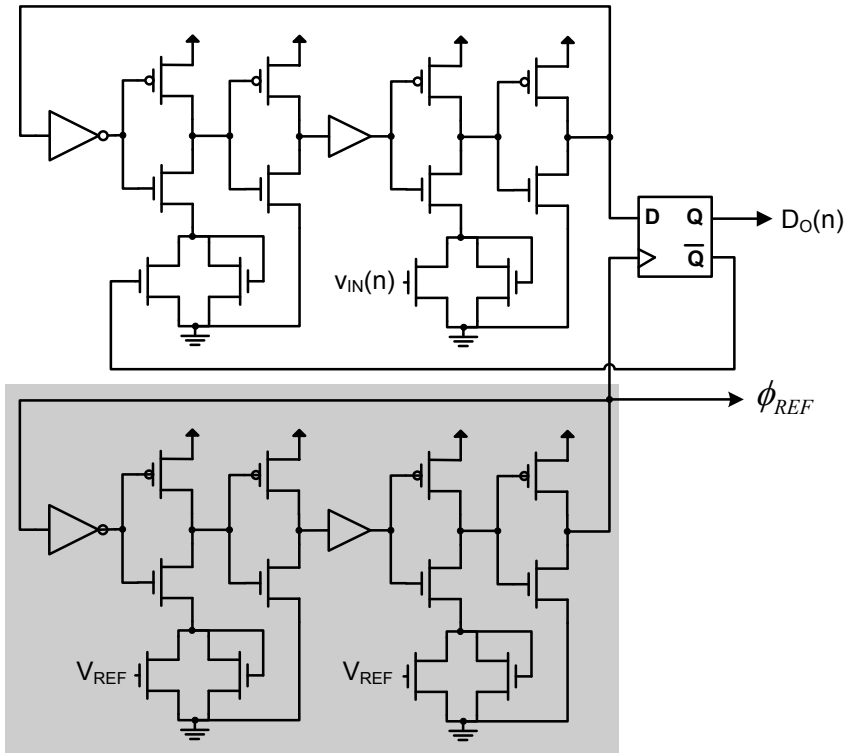
The single-ended first-order $\Delta\Sigma$ modulator implemented in [15] achieved a peak signal-to-noise ratio (SNR) of 49 dB while only consuming 475 μW of power over an area of 375 μm^2. A differential version of the circuit has been built [15] in which the reference oscillator is replaced with the complement of the input oscillator. The resultant peak SNR was measured to be 60 dB with a power consumption of 780 μW. Although the resolution of time-mode $\Delta\Sigma$ modulator is not as high as other reported designs, these numbers represent some of the smallest area and most power efficient designs.

2.8 Summary

In this chapter, various CMOS circuits which implement TMSP were presented. Included in these designs were circuits which convert voltage or digital information into time-difference signals, circuits which process information directly in the time-mode, and circuits that transform time-difference variables back to the voltage and digital domains. As illustrated by the designs introduced, TMSP benefits from the fact that time-differences are technology independent, and as a result, they do not suffer from the dynamic range limitations imposed by the supply voltages. In addition, due to their digital implementation, TMSP circuits will consume less area and operate at faster speeds as technology scales down.



(a)



(b)

Figure 2–16: First-order single-bit time-mode $\Delta\Sigma$ ADC (a) block diagram; (b) transistor schematic

While some of the circuits presented in this chapter were used as bases for larger systems, most were designed for specific applications. The subsequent chapters aim to develop time-mode circuits which implement basic mathematical operations, thus making them applicable for a broader use.

Chapter 3

Building Blocks for Time-Mode Signal Processing

While Chapter 2 focused on time-based designs that performed various signal processing functions, note that the delay unit was the only component that was used in multiple systems. As time-mode signal processing is still relatively new, the technology does not include the same basic mathematical building blocks that are available in analog signal processing. The motivation behind creating a basic set of building blocks is that it allows for the abstraction of the design of large time-mode systems through the creation of different hierarchical levels. By simply replacing the basic functional block in a circuit's block diagram by the building blocks described here, any signal processing circuit could be constructed and later utilized itself as a component in a higher level system.

In this chapter, a set of time-mode signal processing building blocks will be introduced. The circuits described here are based on the work presented in [21] and [20], but in this case, additional current-capacitor branches are added to the system to remove the constant offset term that appears in the input-output expression for each these circuits. An additional constraint which arises from the properties of IIR systems is the need for a common input and output reference clock. This is

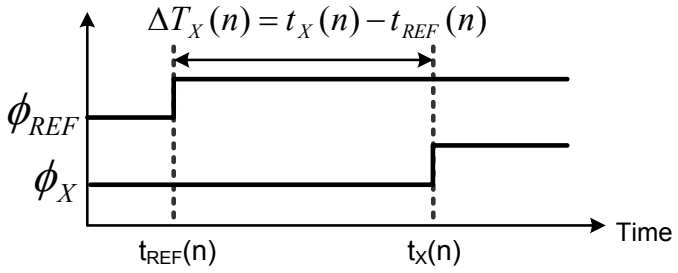


Figure 3–1: Timing diagram representing a time-difference variable $\Delta T_X(n)$.

essential for the implementation of negative feedback in time-mode systems, as it ensures that the feedback signal and the input signal will always stay in sync.

For the remainder of this work, a time-different variable, $\Delta T_X(n)$ will be defined as the quantity of time between the rising edge of a digital signal ϕ_X with respect to the rising edge of a second reference signal ϕ_{REF}. Figure 3–1 shows two digital signals, ϕ_{REF} and ϕ_X, and the resultant time-difference $\Delta T_X(n)$ taken between their rising edges at $t = t_{REF}(n)$ and $t = t_X(n)$.

3.1 Frequency Domain Interpretation of TMSP

In all of the TMSP designs that have been proposed, the system's operation was described by a sampled-data difference equation with respect to the sampling instance n. This representation of time-mode signals is synonymous to a sampled-data sequences of analog values $x(n)$, and therefore we can apply the same frequency domain representations and transforms to time-mode signals that exist for sampled-data time signals. Specifically, we can define the z-transform for a time-difference signal $\Delta T(n)$ as

$$\Delta T(z) \equiv \sum_{n=-\infty}^{\infty} \Delta T(n) z^{-n} \tag{3.1}$$

where z is a complex variable.

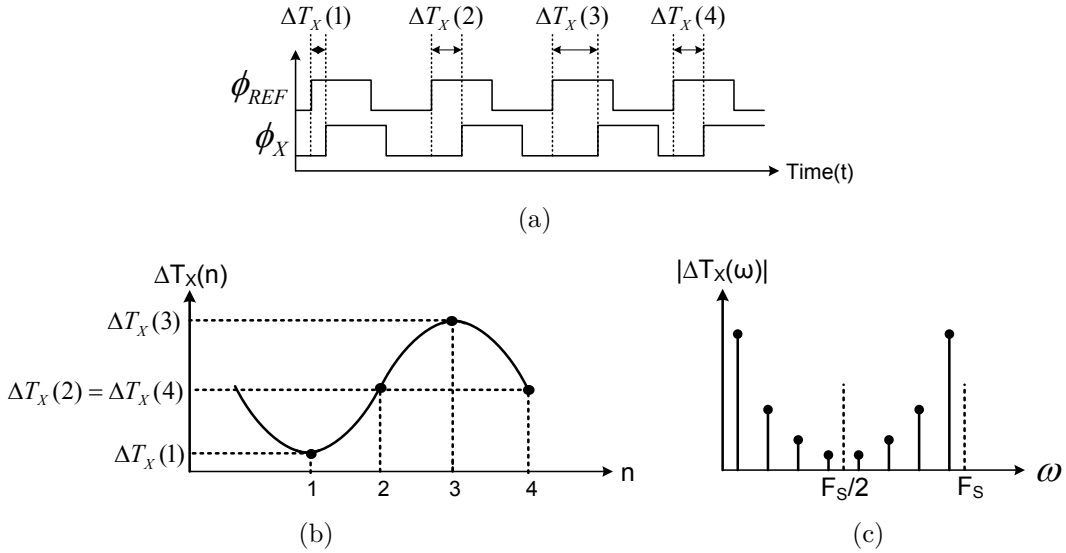


Figure 3–2: Frequency domain representation of TMSP.

For a graphical interpretation of the frequency domain representation of time-difference signals, consider a digital signal ϕ_X and its reference clock ϕ_{REF}, as illustrated in Figure 3–2(a). We can plot the time-difference between the rising edges of the signals, $\Delta T_x(n)$, as a function of the sampling instance n. In the time-difference plot of Figure 3–2(b), $\Delta T_x(n)$ is simply a sampled-data signal, in units of seconds (s), and therefore, we can take the discrete-time Fourier transform $\Delta T_x(\omega)$ of $\Delta T_x(n)$, the magnitude of which is presented in Figure 3–2(c). As with general sampled-data signals, the Fourier transform $\Delta T_x(\omega)$ is simply the frequency domain representation of the time-difference signal $\Delta T_x(n)$. Also note that since we are dealing with a sampled-data system, the Nyquist conditions are still applicable.

The z-transform for time-mode signals possesses the same properties as those for discrete-time z-transforms. The two properties that will be of most interest for this work are the properties of linearity and time shifting. Written in terms of time-difference variables, these properties imply that if $\Delta T_1(z)$ and $\Delta T_2(z)$ are the

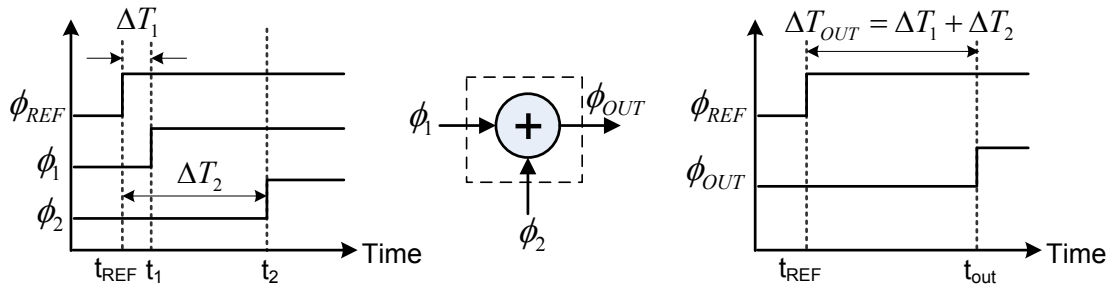


Figure 3–3: Timing diagram and functional block representing time addition.

z-transforms of $\Delta T_1(n)$ and $\Delta T_2(n)$ respectively, then

$$a_1\Delta T_1(n) + a_2\Delta T_2(n) \overset{z}{\leftrightarrow} a_1\Delta T_1(z) + a_2\Delta T_2(z) \tag{3.2}$$

$$\Delta T_1(n-k) \overset{z}{\leftrightarrow} z^{-k}\Delta T_1(z) \tag{3.3}$$

are the z-transforms of the linear combination and time shifted versions of the sampled-data time-mode signals.

For the remainder of this thesis, we will refer to time-difference variables using both their sampled-data as well as their frequency z-domain representations. Since most large systems are characterized by their impulse response $H(z)$, the frequency domain representation of time-difference signals is essential to the development of a design methodology for higher order systems in the time-mode.

3.2 Building Blocks for TMSP

3.2.1 Delay Operators

The delay element is the fundamental element of time-mode signal processing. Multiple implementations of the delay unit that produce both constant and controlled delays were presented in Chapter 2.

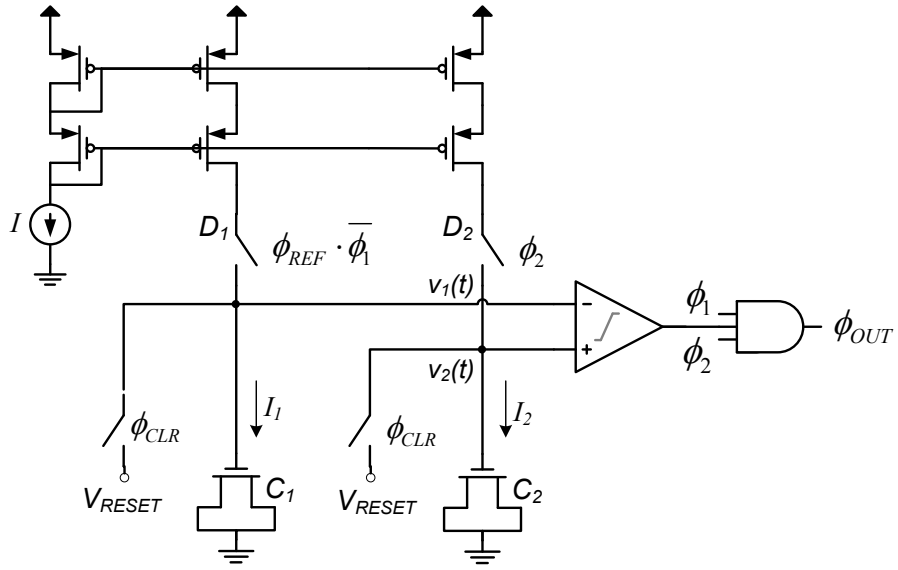


Figure 3–4: Functional diagram of a time adder.

3.2.2 Time-Mode Addition

A timing diagram illustrating the process of time addition as well as the associated functional block are shown in Figure 3–3. A time-mode adder is a circuit that takes two input time-differences and produces at its output two edges which differ in time by the sum of those differences. Therefore, if ΔT_1 and ΔT_2 represent two input time-differences, the sampled-data time-difference equation for ideal time addition is

$$\Delta T_{OUT} = \Delta T_1 + \Delta T_2 \tag{3.4}$$

The functional diagram of a circuit that implements a time adder is illustrated in Figure 3–4. The circuit is composed of a cascode current mirror, an asynchronous analog voltage comparator, two charging capacitors C_1 and C_2 implemented with NMOS transistors, and some analog switches. The output of the comparator is AND-ed with the edges ϕ_1 and ϕ_2 to remove any glitches at the output of the comparator

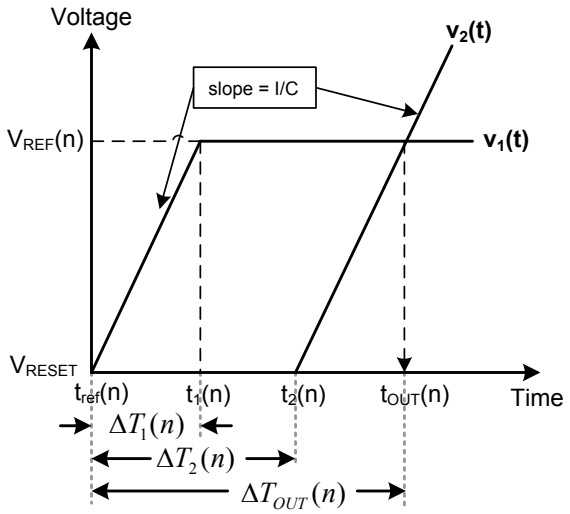


Figure 3–5: Time-voltage graph for a time adder.

due to comparator offsets or glitches in the input time-mode signals. During the low phase of the reference clock and the high phase of ϕ_{CLR}, the voltage on the capacitors are reset to V_{RESET} and the circuit output remains low. When the rising edge of ϕ_{REF} arrives at time $t = t_{REF}$, switch D1 is closed and capacitor C_1 is charged up by the current I_1. The voltage $v_1(t)$ will increase linearly at a rate of I_1/C_1 until the rising edge of ϕ_1 arrives at time $t = t_1$, at which point the switch D1 is open-circuited and the voltage at v_1 is held at

$$v_1(t_1) = V_{REF} = \frac{I_1}{C_1}(t_1 - t_{REF}) + V_{RESET} \tag{3.5}$$

Similarly, at some time $t = t_2 > t_1$ the rising edge of ϕ_2 closes the switch D2, resulting in a voltage that increases linearly at $v_2(t)$ according to the equation

$$v_2(t) = \frac{I_2}{C_2}(t - t_2) + V_{RESET} = \frac{I_2}{C_2}((t - t_{REF}) - (t_2 - t_{REF})) + V_{RESET} \tag{3.6}$$

The output of the comparator will remain low until some time $t = t_{OUT}$ when the voltage at v_2 reaches the reference V_{REF}, resulting in a zero-to-one transition (i.e. rising edge) at the output of the comparator. Equating (3.5) and (3.6) at $t = t_{OUT}$, we can write

$$\frac{I_1}{C_1}(t_1 - t_{REF}) + V_{RESET} = \frac{I_2}{C_2}((t_{OUT} - t_{REF}) - (t_2 - t_{REF})) + V_{RESET} \tag{3.7}$$

In the special case where $I_1 = I_2 = I$ and $C_1 = C_2 = C$, the conversion factors I/C will cancel in (3.7) above, yielding the equation

$$t_1 - t_{REF} = (t_{OUT} - t_{REF}) - (t_2 - t_{REF}) \tag{3.8}$$

The equation above can be rewritten in terms of time-difference variables using the definition of time-mode signals provided in Figure 3–1, that is,

$$\Delta T_X(n) = t_X(n) - t_{REF}(n) \tag{3.9}$$

By using Equation (3.9) with (3.8), the sampled-data version of the time addition relationship described in (3.4), known as time-difference equation for a time adder becomes

$$\Delta T_{OUT}(n) = \Delta T_1(n) + \Delta T_2(n) \tag{3.10}$$

By taking the z-transform of both sides of (3.10), we obtain the frequency domain time-difference equation for the time adder:

$$\Delta T_{OUT}(z) = \Delta T_1(z) + \Delta T_2(z) \tag{3.11}$$

In addition to the timing diagram, the operation of the time adder can be summarized in one voltage vs time graph, as illustrated in Fig. 3–5. This graph

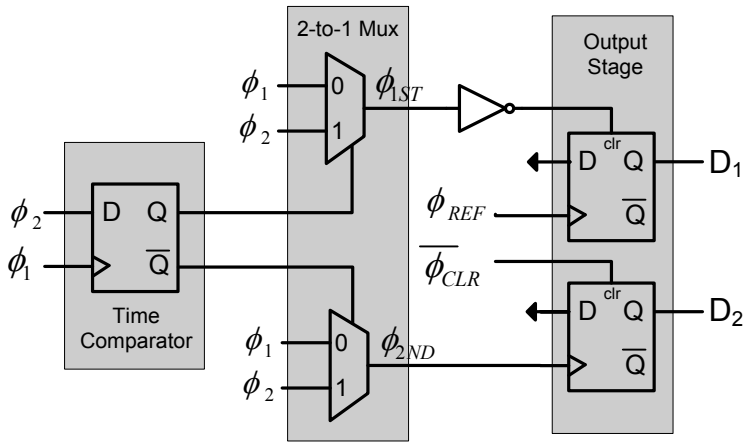


Figure 3–6: Control logic for analog switches.

shows the relationship between the voltages $v_1(t)$ and $v_2(t)$ (y-axis) and the rising edges of the various signals (x-axis) in the circuit for a specific discrete period n of the reference clock.

Note that for the operation described above to be valid, it is required that the rising edge of ϕ_1 occurs before that of ϕ_2, that is, $t_1 < t_2$. To eliminate the circuit's dependency on which edge arrives first, the digital logic of Figure 3–6 is included in the control signals for the switches D1 and D2. The digital logic first compares the arrival of the rising edges of ϕ_1 and ϕ_2 using a time comparator (the operation of which was described in Chapter 2). The input signals are then routed to the switches D1 and D2 based on the comparator output (i.e. which one arrives first) through two multiplexers. The output stage ensures that that D1 turns on only if ϕ_{REF} precedes ϕ_{1ST} and D2 only turns on if ϕ_{2ND} precedes $\overline{\phi_{CLR}}$.

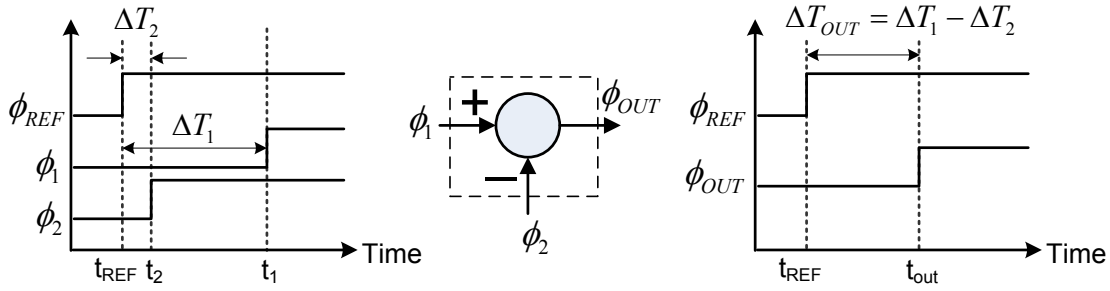


Figure 3–7: Timing diagram and functional block representing ideal time subtraction.

3.2.3 Time-Mode Subtraction

Ideal time subtraction is the process of taking the difference of two time-mode variables, as shown in the timing diagram of Figure 3–7. Thus, if we assume two positive input time-differences ΔT_1 and ΔT_2, the corresponding output difference will be:

$$\Delta T_{OUT} = \Delta T_1 - \Delta T_2 \tag{3.12}$$

As mentioned earlier in this work, all input and output signals are taken with respect to a common reference clock. Under this condition, it would be impossible to implement direct time subtraction or reduction as in both cases, the law of causality would be violated. As illustrated by Equation (3.12), the operations of subtraction (or division) always lead to an output value that is smaller than at least one of the input operands. Extending this idea to time-mode signals, this corresponds to an output time-difference that would be smaller than at least one of the input time-difference signals. However, as all time-difference in this work are referenced to a common clock ϕ_{REF}, this means that the rising edge of the output would have to occur before one or both of the input rising edges in time. This therefore leads to

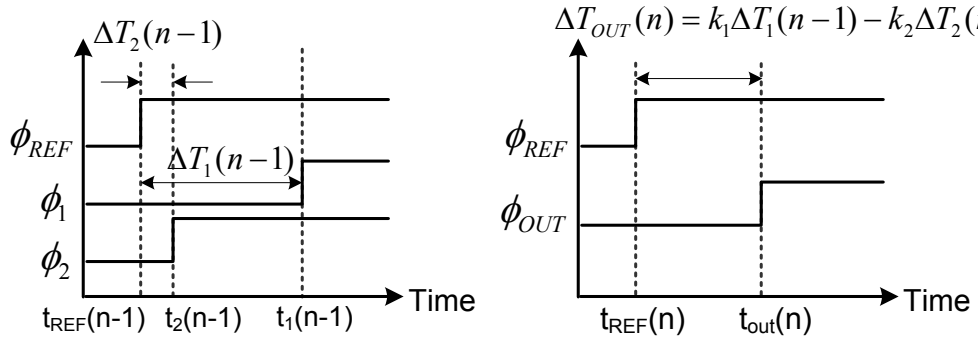


Figure 3–8: Timing diagram for weighted delayed time subtraction (WDTS).

a non-causal situation. This work will then instead present a building block that combines time subtraction with delay and time scaling.

3.2.4 Time-Mode Weighted Delayed Subtraction

In this section, a building block that combines a time subtraction with delay and time multiplication/division is presented. The resultant circuit is known as a weighted delayed time-mode subtractor (WDTS). A time-mode WDTS solves the problem of a common input-output reference clock by delaying the output time-difference by the duration of a clock period, as illustrated in the timing diagram in Figure 3–8. Therefore, the time-difference output defined at sampling instance n will be a function of the input time-differences from the previous sampling instance $(n-1)$, described by the behavioural equation for a WDTS

$$\Delta T_{OUT}(n) = k_1\Delta T_1(n-1) - k_2\Delta T_2(n-1) \tag{3.13}$$

where k_1 and k_2 are the scaling factors for ϕ_1 and ϕ_2, respectively.

A circuit which implements a WDTS is depicted in Figure 3–9. Assuming that both capacitors are initially reset to V_{RESET}, when the rising edge of ϕ_{REF} arrives

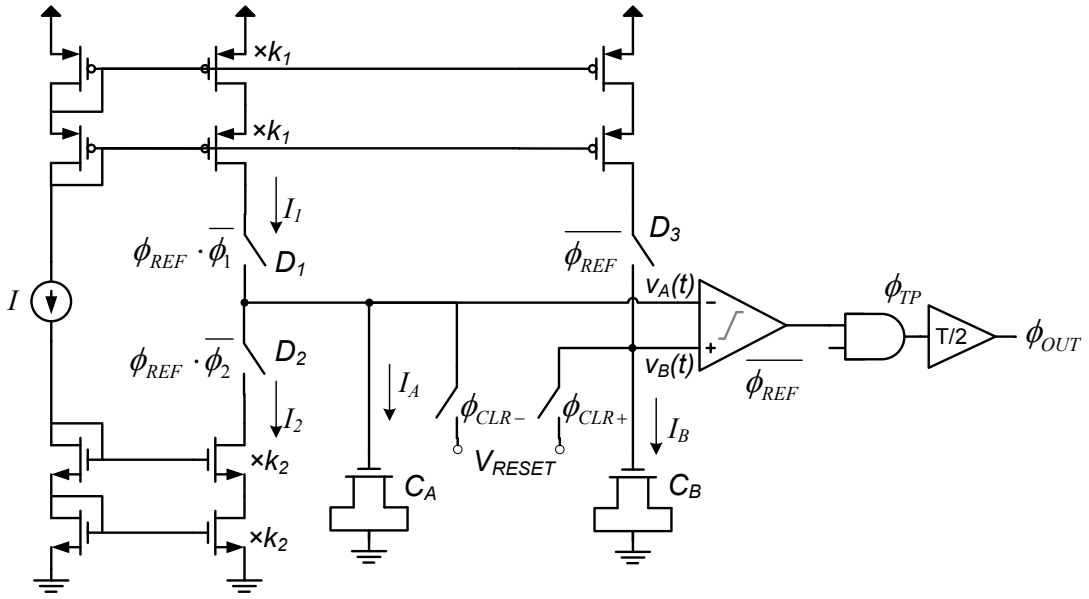


Figure 3–9: Functional diagram of a weighted delayed time subtractor (WDTS).

at time $t = t_{REF}(n)$, switches D1 and D2 are closed and the capacitor C_A will begin to charge by the difference of the currents $I_A = I_1 - I_2$ until either of the rising edges of ϕ_1 or ϕ_2 arrives at $t = t_1(n)$ or $t = t_2(n)$ respectively. It can be shown that regardless of which edge arrives first, the final voltage $V_{REF}(n)$ that will be held on the capacitor C_A after both switches D1 and D2 open is

$$V_{REF}(n) = \frac{I_{SUB}}{C_A}(k_1 \Delta T_1(n) - k_2 \Delta T_2(n)) + V_{RESET} \tag{3.14}$$

where k_1 and k_2 are defined as the ratio of the transistors in the push-and-pull current mirror, and $\Delta T_1(n)$ and $\Delta T_2(n)$ are given by (3.9). The circuit remains in hold mode until ϕ_{REF} experiences a high-to-low transition at $t = \overline{t_{REF}}(n) = t_{REF}(n) + T/2$, where T is defined as the period of the reference clock. At this point, switch D3 closes and the voltage $v_B(t)$ increases linearly by the current I_B. The output of the comparator will remain low until some time $t = t_{TP}(n)$ when the voltage at $v_B(t)$

reaches the reference $V_{REF}(n)$, at which point we can write the voltage over C_B as

$$v_B(tp(n)) = \frac{I_{SUB}}{C_B}(t_{TP}(n) - \overline{t_{REF}}(n)) + V_{RESET} \tag{3.15}$$

Equating (3.14) and (3.15) with $C_A = C_B$ yields the time-difference equation

$$t_{TP}(n) - \overline{t_{REF}}(n) = k_1\Delta T_1(n) - k_2\Delta T_2(n) \tag{3.16}$$

Finally, the rising edge occurring at the output of the comparator/AND gate at time $t = t_{TP}(n)$ is delayed in time by half the period of the reference clock to re-reference it with the next rising edge of the clock. The relationship between the rising edges of ϕ_{TP} occurring at time $t = t_{TP}(n)$ and that at ϕ_{OUT} is thus

$$t_{TP}(n) = t_{OUT}(n+1) - T/2 \tag{3.17}$$

Combining (3.16) and (3.17) with $\overline{t_{REF}}(n) = t_{REF}(n) + T/2$ yields the final relationship between the input time-differences and the output time-difference of the WDTS

$$t_{OUT}(n+1) - (t_{REF}(n) + T) = k_1\Delta T_1(n) - k_2\Delta T_2(n) \tag{3.18}$$

Finally, since the reference in this case is a digital clock with period T, we can replace $t_{REF}(n+1) = t_{REF}(n) + T$ in the equation above. The left side of Equation (3.18) can now be written in terms of a time-difference $\Delta T_{OUT}(n+1)$, yielding the the final time-difference equation for a WDTS:

$$\Delta T_{OUT}(n+1) = k_1\Delta T_1(n) - k_2\Delta T_2(n) \tag{3.19}$$

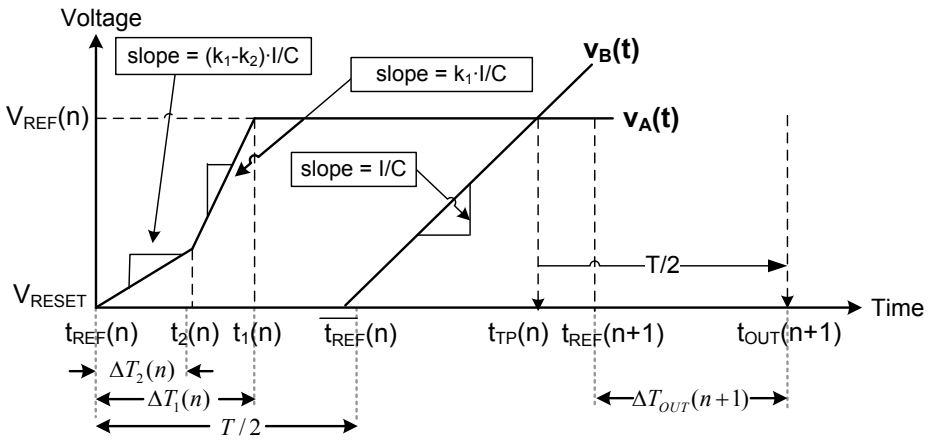


Figure 3–10: Time-voltage graph for a WDTS.

Similarly to the time adder, the frequency domain time-difference equation for the WDTS is obtained by taking the z-transform of Equation (3.19)

$$\Delta T_{OUT}(z) = k_1 z^{-1} \Delta T_1(z) - k_2 z^{-1} \Delta T_2(z) \tag{3.20}$$

where it is assume that all initial conditions on $\Delta T_1(z)$ and $\Delta T_2(z)$ are zero.

A time-voltage graph illustrating the relationship between the voltages $v_A(t)$ and $v_B(t)$ to the transition times of the input and output phases is shown in Figure 3–10. In this case, it is assumed that the rising edge of ϕ_1 occurs after that of ϕ_2, and as a result, the voltage $v_A(t)$ increases throughout the charging cycle. On the other hand, if the rising edge of ϕ_1 occurs before that of ϕ_2, there is an additional necessary condition that must be imposed regarding the voltage at v_A:

$$k_1 \Delta T_1(n) - k_2 \Delta T_2(n) > 0 \tag{3.21}$$

The condition above specifies that the voltage at v_A must increase during the charging phase, otherwise, the capacitor C_A will be discharged to or below the reference voltage

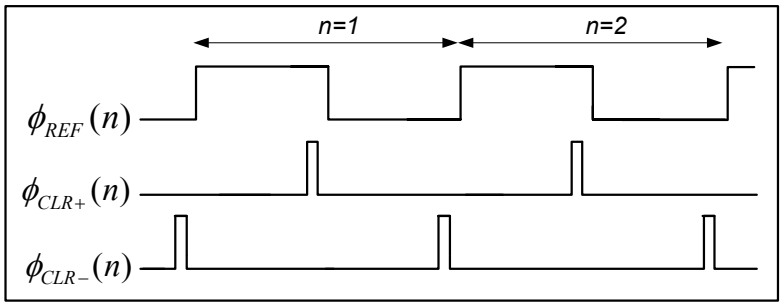


Figure 3–11: Timing diagram of the reference and two reset phases for the WDTS.

causing the output of the comparator to rise at an incorrect time. This behaviour is equivalent to a negative time-difference output, which can not be represented in our system.

While the time adder presented earlier in this chapter only uses the positive half cycle of the reference clock, the WDTS requires both the positive and negative half-cycles of the clock to operate. Since the capacitors C_A and C_B charge during different regions of the clock period, they can not be reset simultaneously, and thus two reset phases signals ϕ_{CLR+} and ϕ_{CLR-} are required. Voltage $v_A(t)$ is reset at the end of the negative half-cycle of the clock by ϕ_{CLR-} while $v_B(t)$ is reset at the end of the positive half-cycle of the clock by ϕ_{CLR+}. The timing diagram in Figure 3–11 shows the relationship between the reference clock and the two reset signals.

3.2.5 Time-Mode Weighted Delayed Addition

By changing the direction of the current I_2 in the WDTS to source the capacitor C_A, a time-mode weighted delayed time-mode adder (WDTA) can be realized, shown in Figure 3–12. The operation of this circuit is identical to that of the WDTS, except that in the case of the WDTA, the capacitor C_A will be charged by both currents I_1 and I_2, as shown in the time-to-voltage graph in Figure 3–13. An analysis similar to

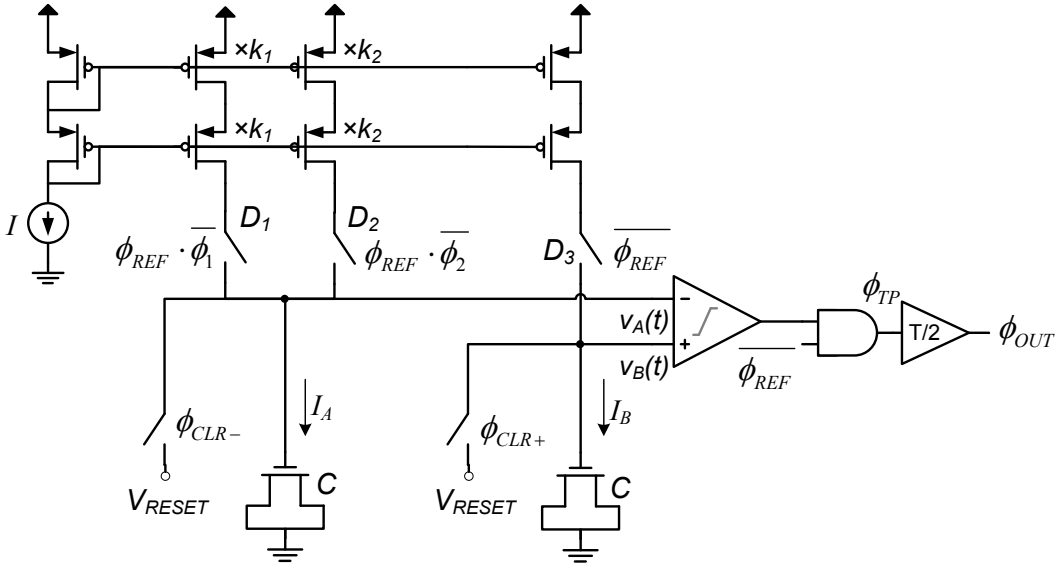


Figure 3–12: Functional diagram of a weighted delayed time adder (WDTA).

that performed for the WDTS yields the time-difference equation for the WDTA

$$\Delta T_{OUT}(n) = k_1 \Delta T_1(n-1) + k_2 \Delta T_2(n-1) \tag{3.22}$$

with corresponding frequency domain equation

$$\Delta T_{OUT}(z) = k_1 z^{-1} \Delta T_1(z) + k_2 z^{-1} \Delta T_2(z) \tag{3.23}$$

The equivalent z-domain block diagrams for the weighted delayed time adder and subtractor are shown in Figures 3–14(a) and 3–14(b).

3.2.6 Time-Mode Scaling

Section 2.2 described a circuit that takes two input time-mode signals and produces two outputs which differ in time by a scaled version of the input time-difference. Because of the nature of the circuit topology, the reference signals for the input time

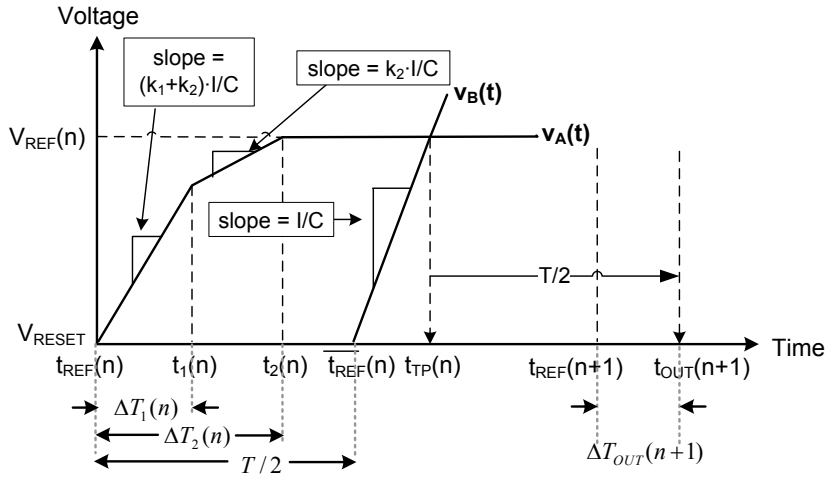


Figure 3–13: Time-voltage graph for a WDTA.

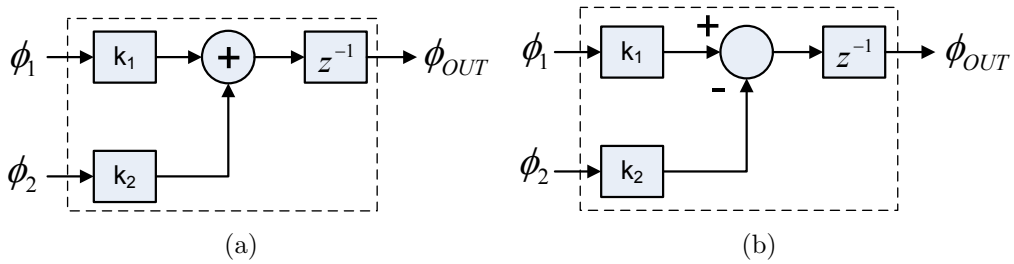


Figure 3–14: Block diagram for the time-mode (a) WDTS, and (b) WDTA.

difference and output time difference were separate, and thus this time amplifier could both increase (scale-up) or decrease (scale-down) the input time-difference.

Since our time-mode circuit topology uses a common input-output reference clock, the process of scaling-up and scaling-down time-mode differences must be divided into two distinct operations. For the context of this section and the remainder of this work, time amplification will be defined as the operation of scaling-up a time-difference by a factor $k \geq 1$, while time attenuation will be defined as the operation of scaling-down a time-difference by a factor $k < 1$. As with the WDTA and WDTS, due the constraint of a common input-output time reference, ideal time-attenuation

can not be performed due to the laws of causality. Therefore, time attenuation is inherently a time-delayed operation. Time amplification, on the other hand, does not require added delay since the output rising edge will always occur at a time after the rising edge of the input. The behavioural time-difference equations for time amplification and time attenuation are given by equations (3.24) and (3.25), respectively

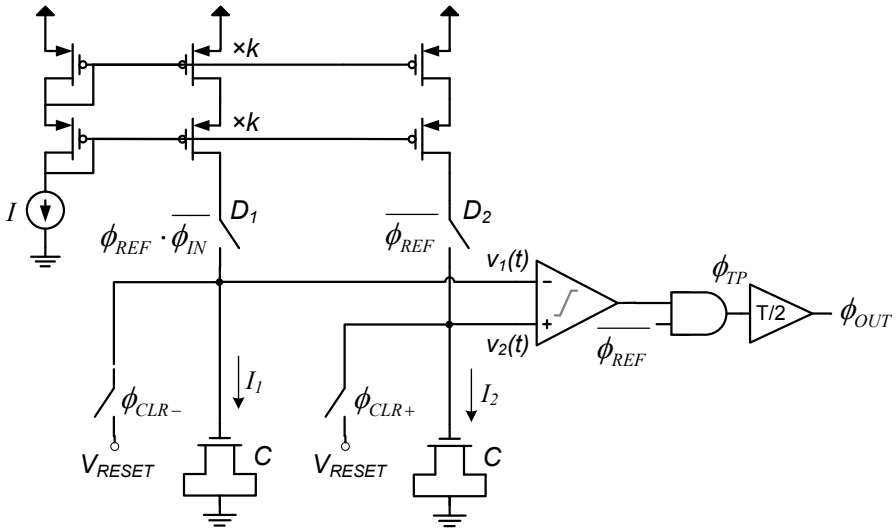
$$\Delta T_{OUT}(n) = k\Delta T_{IN}(n) \tag{3.24}$$

$$\Delta T_{OUT}(n) = k\Delta T_{IN}(n-1) \tag{3.25}$$

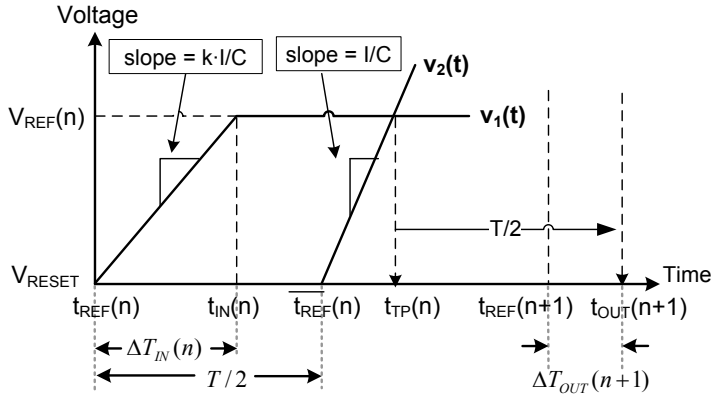
where $k \geq 1$ for time amplification and $k < 1$ for time attenuation.

The topology of the time-mode attenuator (TATN) circuit as well as its associated time-to-voltage relationship are shown in Figures 3–15(a) and 3–15(b), respectively. The operation of the time attenuator is equivalent to that of the WDTS where the sinking branch current I_2 is removed. The resultant analysis is obtained using the same methods presented in Section 3.2.4 and by setting $\Delta T_2(n) = 0$ to remove any contribution from the current sinking branch I_2.

Although omitted here, the topology of the time-mode amplifier (TAMP) is easily obtained by making two modifications to the TATN circuit. Since the output time-difference in the case of the TAMP is larger than its corresponding input (thus there are no causality concerns), the final $T/2$ delay cell can be removed. Secondly, for similar reasons as stated above, the edge used to control the switch D2 is changed from the falling edge of ϕ_{REF} in the case of the TATN to its rising edge in the TAMP.



(a)



(b)

Figure 3–15: Time-mode attenuator: (a) functional diagram, (b) time-voltage graph.

The frequency domain equations for the TAMP and TATN are found by taking the z-transforms of (3.24) and (3.25) respectively

$$\Delta T_{OUT}(z) = k\Delta T_{IN}(z) \tag{3.26}$$

$$\Delta T_{OUT}(z) = kz^{-1}\Delta T_{IN}(z) \tag{3.27}$$

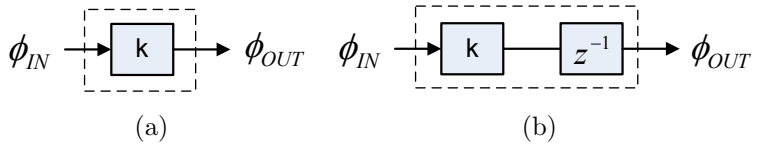


Figure 3–16: Block diagram for the time-mode (a) TAMP, and (b) TATN.

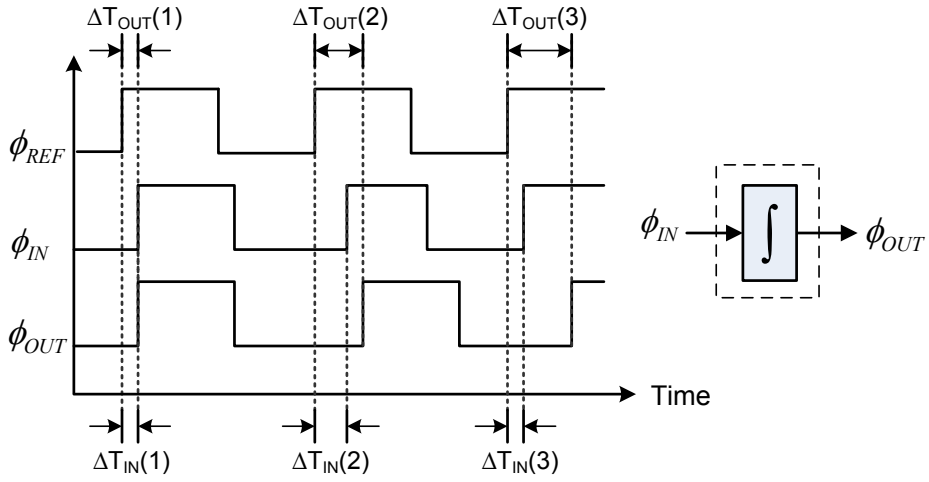


Figure 3–17: Timing diagram and functional block of time-mode integration.

The equivalent z-domain block diagrams for the time-mode amplifier and attenuator are shown in Figures 3–16(a) and 3–16(b).

3.2.7 Time-Mode Integration

A time-mode integrator is a circuit whose output time-difference is the sum of the output's previous and the input's current time-differences. A timing diagram illustrating the process of integration as well as the associated functional block are shown in Figure 3–17. If $\Delta T_{IN}(n)$ is the time-difference input to a time-mode integrator at sampling-instance (i.e. period) n, the time-difference equations for the

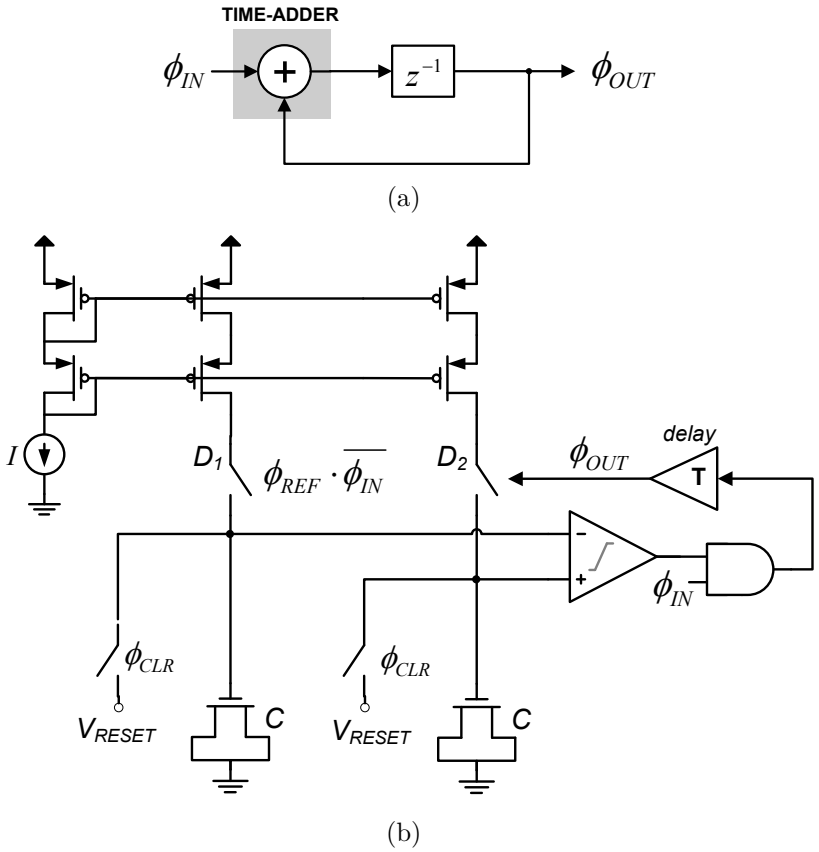


Figure 3–18: Time-mode integrator (a) Block diagram, (b) functional diagram.

time-mode integrator is given by

$$\Delta T_{OUT}(n) = \Delta T_{OUT}(n-1) + \Delta T_{IN}(n) \tag{3.28}$$

and its corresponding frequency domain equivalent is given by

$$\Delta T_{OUT}(z) = \frac{1}{1-z^{-1}} \Delta T_{IN}(z) \tag{3.29}$$

As shown in the block diagram of Figure 3–18(a), a time-mode integrator can be built from an adder, a unit delay, and some feedback. At the circuit level, a time-integrator may be realized by cascading the time adder described earlier in

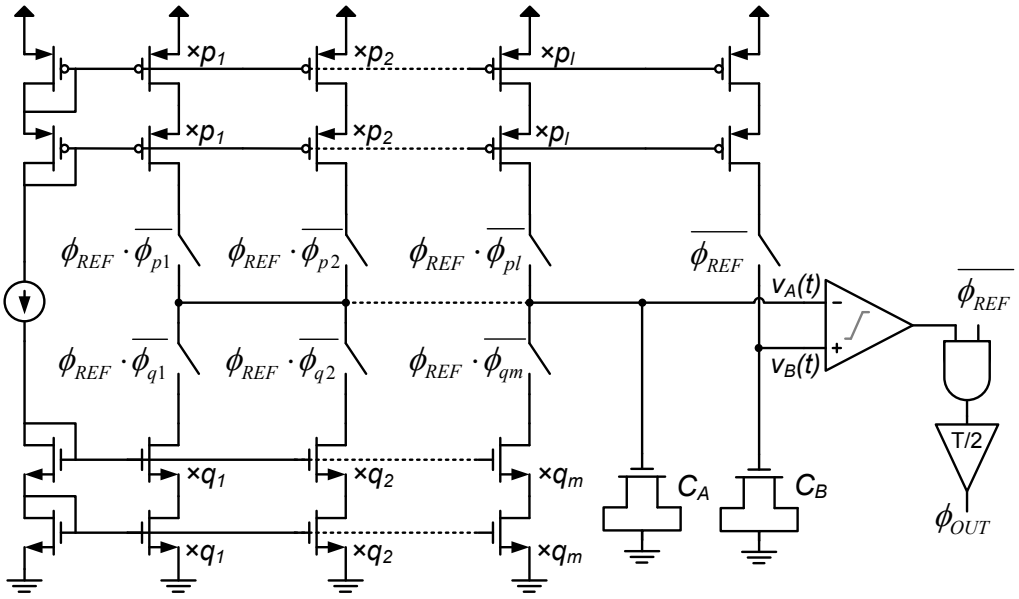


Figure 3–19: Functional diagram of a general multiple-input delayed time subtractor.

this chapter with a unit-delay circuit (whose design will be discussed later in this chapter), and feeding back the delayed output to the time adder's input ϕ_2, as shown in Figure 3–18(b).

Note that since this work only focuses on positive time-differences, the output of the time integrator will always be larger or equal to its previous value. In other words, the rising edge of the output ϕ_{OUT} will always occur some time after the rising edge of the input ϕ_{OUT}. As a result, the extra control logic used to determine the order of occurrence of the time adder inputs is not required.

3.2.8 Multiple-Input Time-Mode Building Blocks

Extending the presented methodology to design multiple-input building blocks is simple and requires few extra components to implement. Considering the WDTS presented in Section 3.2.4, it is noted that each input ϕ_i controlled the flow of one of

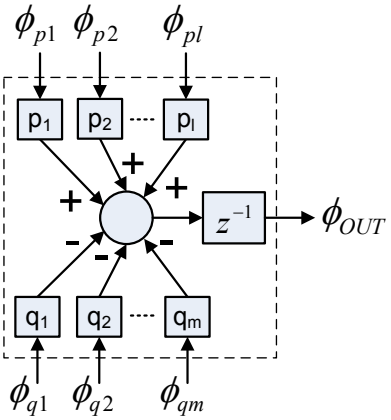


Figure 3–20: Block diagram for the general multiple-input time-mode WDTS.

the currents I_i which either charged or discharged the leftmost (reference) capacitor. Therefore, to achieve a multiple-input circuit, sourcing or sinking branches are simply added to the comparator reference node for each additional input signal ϕ_i. The circuit diagram which implements a general $(l + m)$-input WDTS, where l denotes the number of sourcing branches (positive operands), and m denotes the number of sinking branches (negative operands), is illustrated in Figure 3–19. The time-difference equation for of this circuit is

$$\Delta T_{OUT}(n) = \sum_{i=1}^{l} p_i \Delta T_{pi}(n-1) - \sum_{i=1}^{m} q_i \Delta T_{qi}(n-1) \tag{3.30}$$

with corresponding frequency domain equation

$$\Delta T_{OUT}(z) = \sum_{i=1}^{l} p_i z^{-1} \Delta T_{pi}(z) - \sum_{i=1}^{m} q_i z^{-1} \Delta T_{qi}(z) \tag{3.31}$$

The equivalent z-domain block diagram for the general multiple-input WDTS is shown in Figure 3–20. The theory presented for multiple-input blocks can also be easily applied to the time adder and the WDTA. Note that in the case of the

time adder, the multiple-input implementation would also require an increase in the digital logic used to properly route the input rising edges according to their order of occurrence.

3.2.9 Differential Time-Mode Building Blocks

For most of the blocks described in this section, the most straightforward differential implementation is obtained by designing two identical time-mode circuits, and stimulating each with the time-difference between one of the differential input time-signals and a common reference clock. The differential output time-difference is then taken between the two single-ended outputs of the two circuits. While this method does allow us to take advantage of some of the positive properties of differential signaling, such as noise immunity, the cost is double the effective area. In addition, the resultant mismatch and process variation that will exist between same elements of the two blocks will limit the effectiveness of this method.

Note that for some of the block described in this section, specifically, those that require an additional delay block for re-referencing purposes, computation (i.e. charging) is performed during both the positive and negative half-cycle of the reference clock period ϕ_{REF}. As a result, the issues described above apply. For circuits that are not delayed, such as the time adder, computation is performed during the positive half-cycle of the reference clock only, and the system remains in a hold state during the negative half-cycle. In this case, a second computation during the clock's negative half-cycle can be performed, and as a result, the same circuit components can be used to compute the two differential signals that otherwise required two separate time-mode circuits.

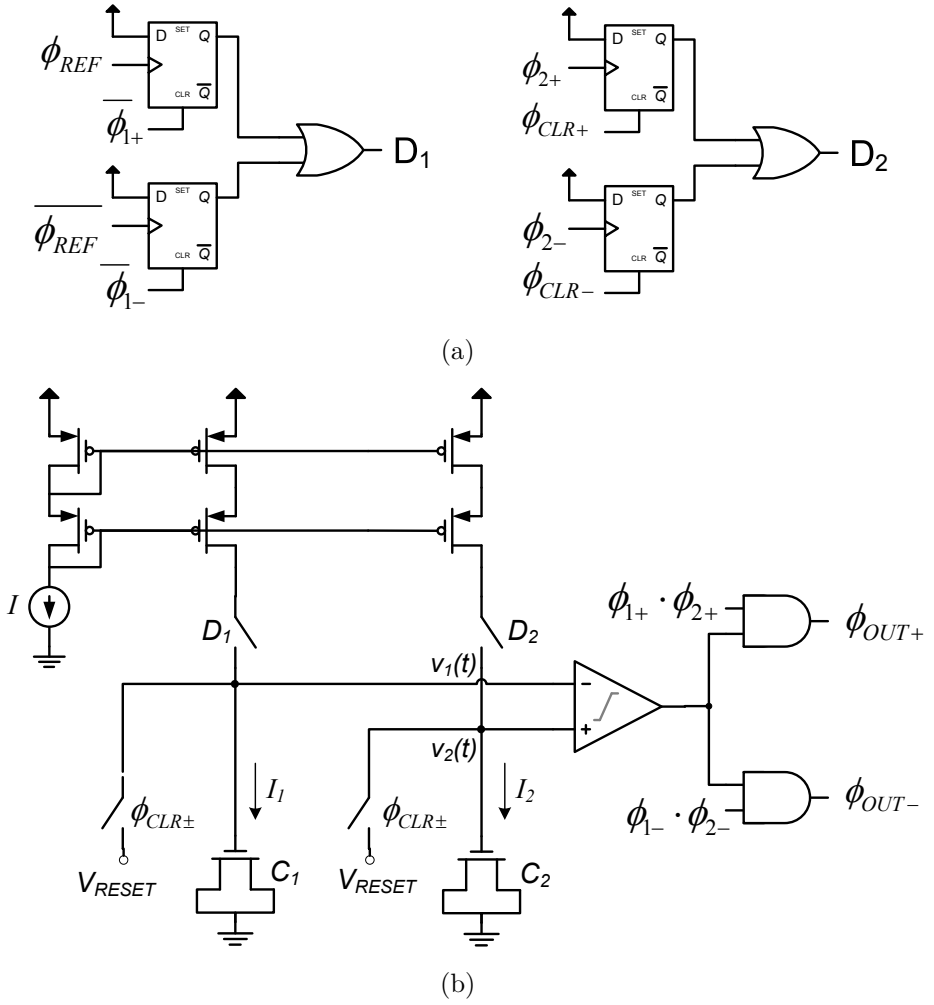


Figure 3–21: (a) Control logic for analog switches. (b) Functional diagram of a differential time adder.

To illustrate the use of a single time-mode block for differential TMSP, consider the time adder described earlier in this section. To convert the circuit into a differential time adder, the control signals for the switches D1 and D2 are modified, and a second AND gate is added to the output of the comparator, as illustrated in Figures 3–21(a) and 3–21(b).

The operation of the differential time adder is equivalent to that of two separate single-ended time adders operating on different half cycles of the reference clock. The first addition operation occurs during the positive half-cycle of ϕ_{REF}, during which the time-differences between the *rising* edge of the signals ϕ_{1+} and ϕ_{2+} and the *rising* edge of the reference ϕ_{REF} are summed. Note that during the positive half-cycle of the clock, the bottom flip flops in Figure 3–21(a) remain inactive, and thus the switch select signals for D1 and D2 are controlled by the top flip flops only. The resultant rising edge is then routed to the top branch of the comparator output by AND-ing it with the signals ϕ_{1+} and ϕ_{2+} resulting in ϕ_{OUT+}. This operation results in time-difference equation

$$\Delta T_{OUT+}(n) = \Delta T_{1+}(n) + \Delta T_{2+}(n) \tag{3.32}$$

where the time-difference $\Delta T_{OUT+}(n)$ is referenced with respect to the *rising* edge of $\phi_{REF}(n)$.

Similarly, the second addition operation occurs during the negative half-cycle of ϕ_{REF}. In this case, the input time-differences are defined as the time-difference between the *rising* edge of the signals ϕ_{1-} and ϕ_{2-} and the *falling* edge of the reference ϕ_{REF}. The resultant rising edge is then routed to the bottom edge of the comparator output by AND-ing it with the signals ϕ_{1-} and ϕ_{2-} resulting in ϕ_{OUT-}. The time-difference equation of this second addition is

$$\Delta T_{OUT-}(n) = \Delta T_{1-}(n) + \Delta T_{2-}(n) \tag{3.33}$$

where here the time-difference $\Delta T_{OUT-}(n)$ is referenced with respect to the *falling* edge of $\phi_{REF}(n)$.

The differential output of the system $\Delta T_{OUT-diff}(n)$ is taken between the rising edge of $\phi_{OUT-}(n)$ at time $t = t_{OUT-}(n)$ and the rising edge of $\phi_{OUT+}(n)$ at time $t = t_{OUT+}(n)$. The time-difference $\Delta T_{OUT-diff}(n)$ is proportional to the difference between $\Delta T_{OUT+}(n)$ and $\Delta T_{OUT-}(n)$, given by the equation

$$\Delta T_{OUT-diff}(n) = T/2 - (\Delta T_{OUT+}(n) - \Delta T_{OUT-}(n)) \tag{3.34}$$

where the extra $T/2$ factor accounts for the half-period shift between the references of $\phi_{OUT+}(n)$ and $\phi_{OUT-}(n)$. If $\phi_{OUT+}(n)$ is delayed by half a period of the reference clock T/2, it is re-referenced directly with $\phi_{OUT-}(n)$, and therefore the new output time-difference can be written as:

$$\Delta T'_{OUT-diff}(n) = \Delta T_{OUT+}(n) - \Delta T_{OUT-}(n) \tag{3.35}$$

Note that for the differential time adder, the voltages $v_1(t)$ and $v_2(t)$ must be reset before the occurrence of both the rising and falling edges of the reference clock $\phi_{REF}(n)$. Therefore, the reset switches are controlled by the digital signal $\phi_{CLR\pm}$, which is the logical AND of the clear signals ϕ_{CLR+} and ϕ_{CLR-} described in the Section on the WDTS.

3.3 Building Block Implementation

3.3.1 Voltage Comparator

As will be further discussed in the next Chapter, one of the most critical blocks in the design of the time-mode building blocks is the voltage comparator. The comparator needs to be able to make quick logic decision based on ramping voltages in continuous (i.e. asynchronous) time. More importantly, the comparator must

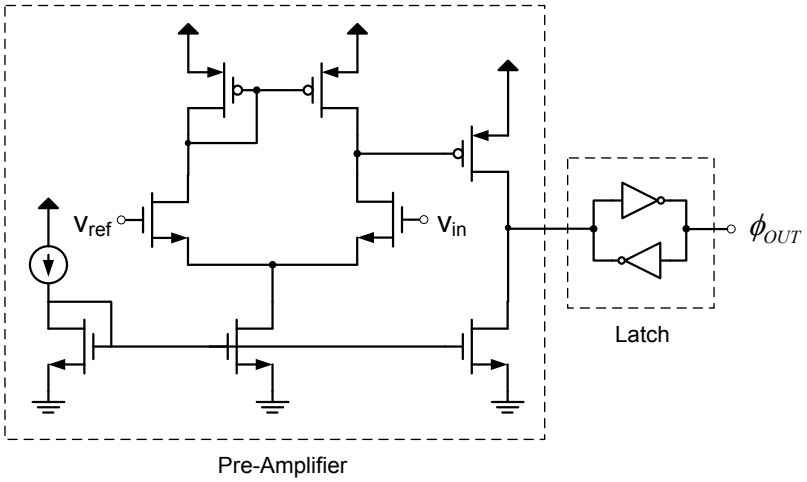


Figure 3–22: Transistor schematic of the asynchronous voltage comparator.

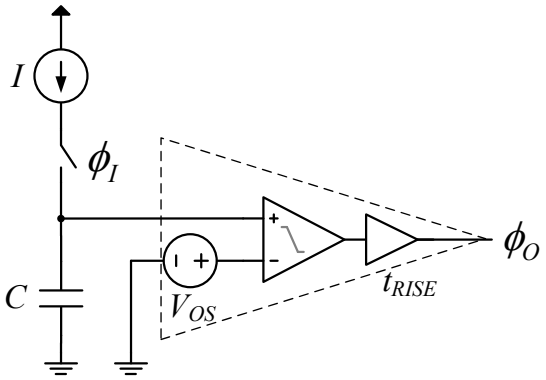


Figure 3–23: Time-offset of an asynchronous voltage comparator.

provide a large enough gain and bandwidth to ensure that changes in its output, specifically, low-to-high transitions, occur with minimal transient behaviour. In order to maximize the bandwidth of the comparator, a simple two stage pre-amplifier/latch combination, shown in Figure 3–22, was implemented. The latch used in the design was taken from a standard cell library.

The main issue in the implementation of the time-mode blocks is the non-zero time-offset generated by the comparator. To illustrate the sources of this offset,

consider the single branch time-mode circuit in Fig. 3–23. Here the offset of the voltage comparator is modeled with a voltage source V_{OS}. In addition, an additional delay block t_{RISE} is cascaded with the output of the voltage comparator to account for finite gain and bandwidth effects manifesting itself into a finite output rise time.

If the comparator in the figure were ideal, that is, V_{OS} and t_{RISE} are both zero, then as soon as the switch is closed, the output edge associated with ϕ_O would rise at exactly the same time as the input edge associated with ϕ_I. In the case where V_{OS} and t_{RISE} are nonzero, the output edge will be delayed with respect to the input edge time. We can quantified this delay based on the parameters of Fig. 3–23 as follows

$$t_{OFFSET} = C/I \cdot V_{OS} + t_{RISE} \tag{3.36}$$

Note that Equation (3.36) yields a delay time independent of the time-mode variables in the circuit. To confirm this analysis, the time-offset of the comparator was simulated and plotted in Figure 3–24 at different reference voltages. As expected, the comparator was found to have a relatively constant time-offset of 3 ns between 0.7 V and 1.4 V. To take advantage of this reference-independent time-offset, the time-mode building blocks were designed such that the comparator's input voltage range stayed within the constant region. Using the time adder of Figure 3–4 as an example, the component values are selected such that $v_1(t)$ and $v_2(t)$ are larger than $V_{REF-MIN}$ and smaller than $V_{REF-MAX}$. The lowest voltage that will occur at both $v_1(t)$ and $v_2(t)$ is during the reset stage, therefore we obtain the condition:

$$V_{RESET} \geq V_{REF-MIN} \tag{3.37}$$

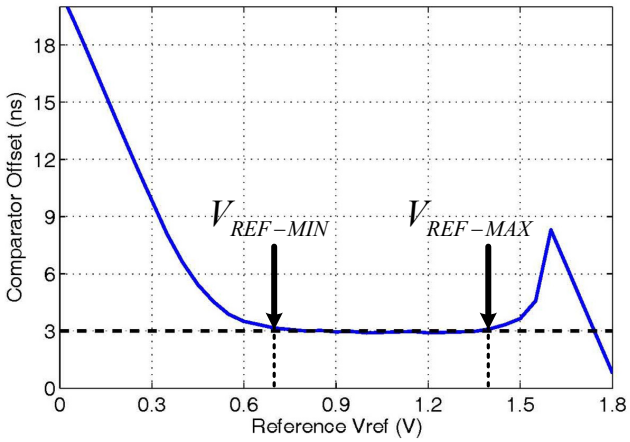


Figure 3–24: Time offset of asynchronous voltage comparator.

The maximum voltages that will occur at either $v_1(t)$ or $v_2(t)$ requires that the switches D1 or D2 remain open throughout the positive-half cycle of the clock, thus charging these voltages for a half period. The condition for the maximum voltage is therefore:

$$V_{RESET} + \frac{I}{C}(T/2) \leq V_{REF-MAX} \tag{3.38}$$

The effects of this offset on higher-order time-mode systems as well as methods for compensation will be further discussed in the next chapter.

3.3.2 Unit and Half-Unit Delay Circuit

In TMSP, a unit delay circuit is simply a cell which delays the rising edge of a time-mode signal by an amount equal to the period T of the reference clock ϕ_{REF}. Also required for the design of the time-mode building blocks is a circuit that delays by half a period, $t_{DELAY} = T/2$. In this section, we describe the design of a half-unit delay cell, $z^{-1/2}$, and cascade two of these cells to produce the full z^{-1} unit delay.

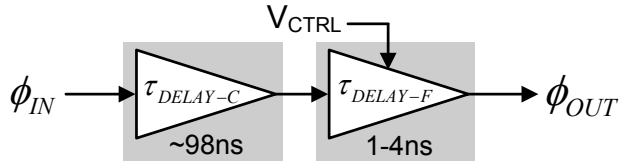


Figure 3–25: Half-unit delay circuit block diagram.

The half-unit delay block is implemented using two delay stages, as depicted in the block diagram of Figure 3–25. Designed to produce a total delay of $T/2 = 100ns$, this circuit is composed of a coarse delay unit which produces the majority of the required delay $\tau_{DELAY-C}$, and the fine delay unit which is tunable (i.e. voltage controlled) and produces a much smaller delay $\tau_{DELAY-F}$. The tune-ability of the final stage's delay allows the designer to calibrate the total delay by adjusting the voltage V_{CTRL}.

The coarse delay unit, depicted in Figure 3–26(a), is composed of a D-flip flop and four inverters (M_{1-6}), and three bypass transistors (M_{7-9}). The delay range of the cell is controlled by the length L of the inverter transistors M_{1-6}. Since this circuit will be creating large delays on the order of half a period, the slow rise times of the inverter can cause a degradation or a loss of either the logic '1' or '0' regions. Therefore the circuit is preceded by a D-flip flop which acts as a pulse restorer, that is, it extends the length of the logic '1' pulse of ϕ_{IN}. To ensure that the logic '0' region of the digital signal is not lost, and since we are only interested in the rising edge of the input clock signal ϕ_{IN} and not the falling edge, each inverter stage includes a bypass transistor (M_{7-9}) which ensure the input's falling edge is conserved rather than delayed.

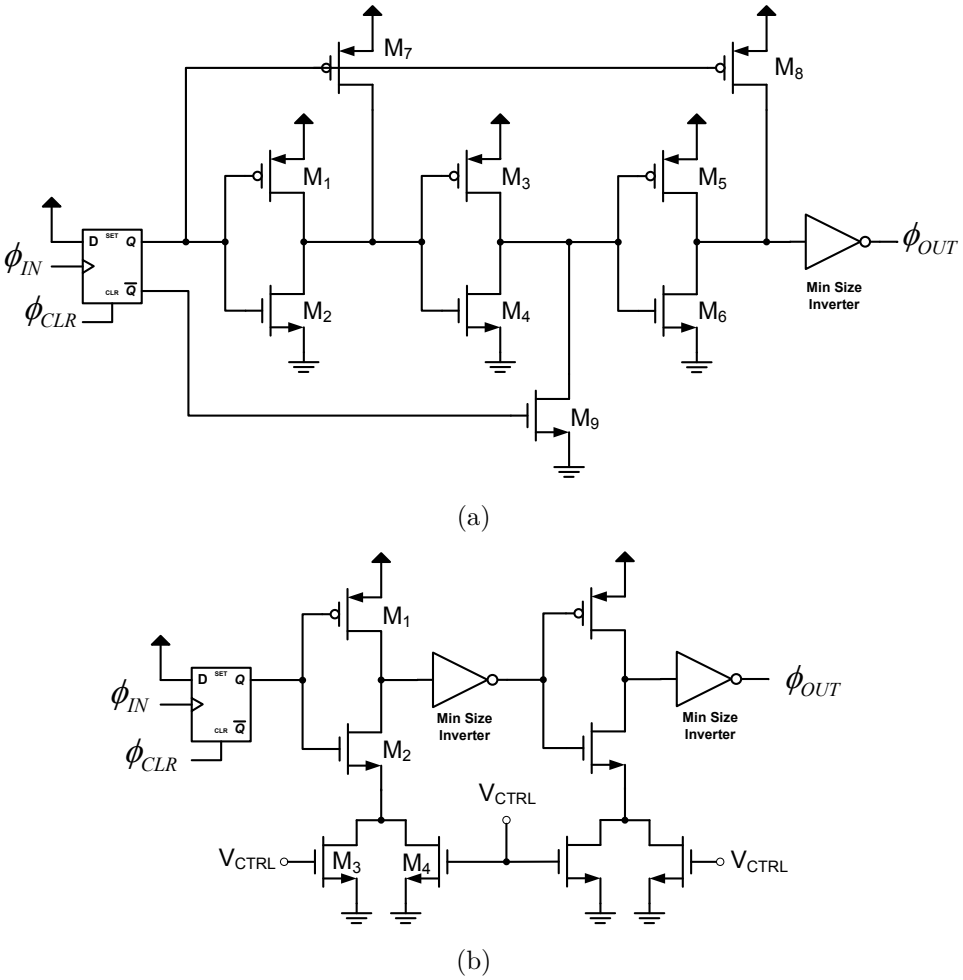


Figure 3–26: Functional schematic for (a) coarse delay block, (b) fine delay block.

To construct the fine delay unit, the D-flip flop used in the coarse unit is cascaded with two VCDUs and two minimum size inverters. The VCDUs are implemented using a current starved topology (M_{1-4}) with control voltage V_{CTRL}. Since we are only interested in the delay of the rising edge of the input, the VCDUs are designed to delay the rising edge of the input signal, while the falling edge is simply fed through the subsequent minimum size inverter.

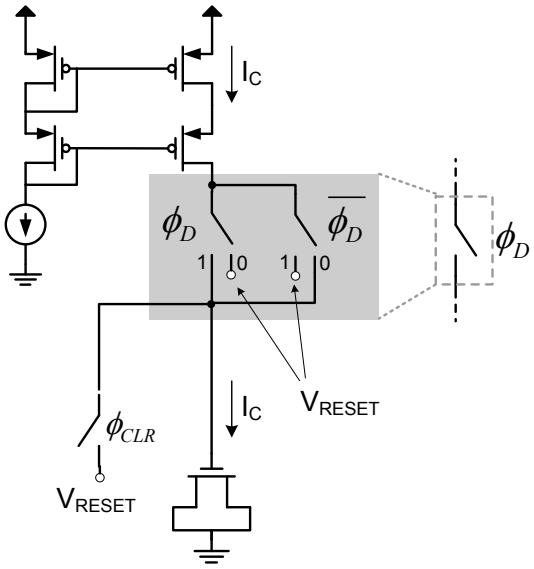


Figure 3–27: Complementary switching.

3.3.3 Analog Switches

The analog switches used in the time-mode building blocks were implemented using a complementary differential configuration to remove any common-mode noise. An example of the use of complementary differential switching in the I-C branches is shown in Figure 3–27. In this situation, when the rising edge of ϕ_D occurs, both switches short-circuit and the current I_C begins charging the capacitor. When the falling edge of ϕ_D occurs, the switches both open-circuit, and the output of the current-source is short-circuited to V_{RESET}. Thus this switch topology is equivalent to the single switch case.

The differential switches were implemented using two transmission gates and a static inverter, as illustrated in Figure 3–28.

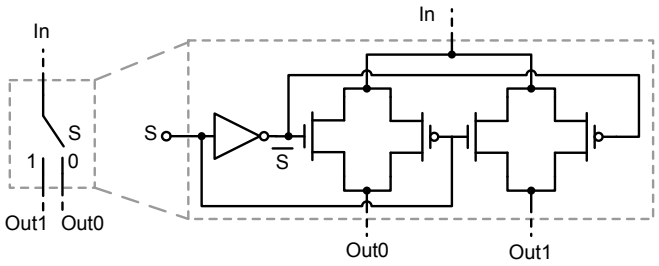


Figure 3–28: Differential switching.

Table 3–1: Components used for implementation of the time-mode building blocks.

Component	**Part Used**	**Specifications**
N-Channel MOSFET	2N7000	$0.8V < V_{GS-thn} < 3V$
P-Channel MOSFET	ZVP3360	$-3.5V < V_{GS-thp} < -1.5V$
Analog Switches	ADG412	$t_{ON} = 110ns$, $R_{ON} = 25\Omega$
Voltage Comparator	AD8561	$t_P = 7ns$, $V_{OS} = 1mV$
D-type Flip Flop	HEF4013	$t_P = 110ns$, $f_{max} = 14MHz$
Inverter	HEF4069	$t_P = 45ns$

3.4 Discrete Component Implementation

To verify the operation of the circuits presented in this chapter, the time adder and time-mode WDTS were constructed using discrete components and tested using bench top test equipment. The current mirror circuits were implemented using discrete NMOS and PMOS devices, as well as biasing resistors. As for the remainder of the components (i.e. analog switches, comparator, digital logic), industry manufactured ICs were utilized. A summary of of the components used to implement the building blocks is given in Table 3–1.

The input digital clocks and edges were produced using two single-channel Agilent 33220A 20-MHz arbitrary waveform generators (AWGs) and one HP 165-MHz

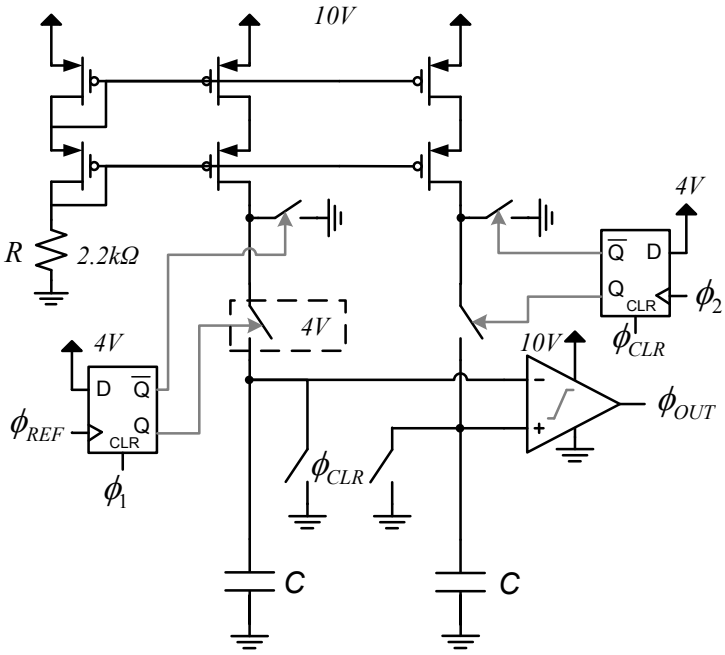
81170A dual-channel pulse-pattern generator. The output digital edges were captured using a 600-MHz, 4GSa/s Agilent Infiniium 54830D oscilloscope.

Finally note that for the discrete implementation, two voltage values were used to power the devices. The voltage requirement resulting from the stacking of transistors in the current mirror required the analog circuity to be run off a 10V supply (A_{VDD}), while the maximum output voltage swing of the signal sources required that the digital circuits utilize a 4V supply (D_{VDD}). Both supply voltages were generated using an HP E3630A triple output DC power supply.

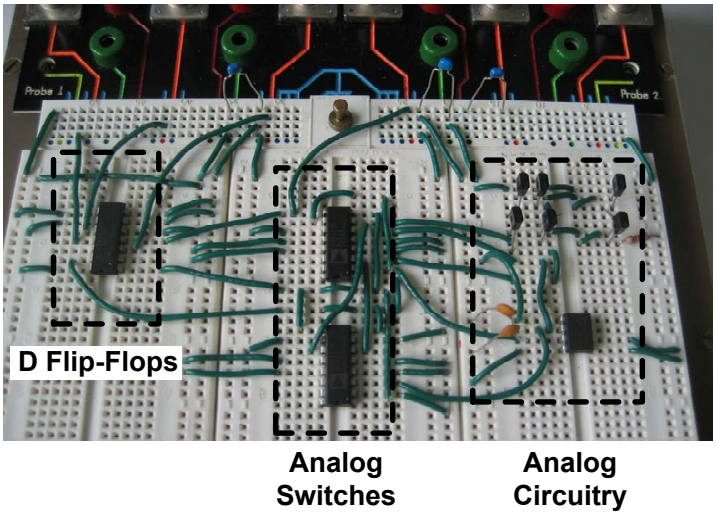
3.4.1 Time-Mode Adder

In order to implement the time adder using discrete components, some modifications were made to the original circuit schematic presented earlier in this chapter (Figure 3–4). First of all, the ideal current source used to set the reference current was implemented using a bias resistor R. The charging capacitors, implemented on chip using drain-source grounded transistors, were replaced with discrete capacitors. Since we were unable to tune the sizing of the available transistors, this modification resulted in a greater range of capacitance values being available for test. Finally, the output AND gate in the original design was omitted since the offset voltage was found to be negligible. The final schematic of the time adder used in the discrete component implementation is illustrated in Figure 3–29(a). A photograph of the physical circuit is shown in Figure 3–29(b).

The circuit of Figure 3–29(b) was tested using the setup described above. The frequency of the reference clock was set to both 20-kHz and 200-kHz which required charging capacitances of C of 10nF and 1nF, respectively. The input and output time-mode signals, as well as the input voltages of the comparator were captured

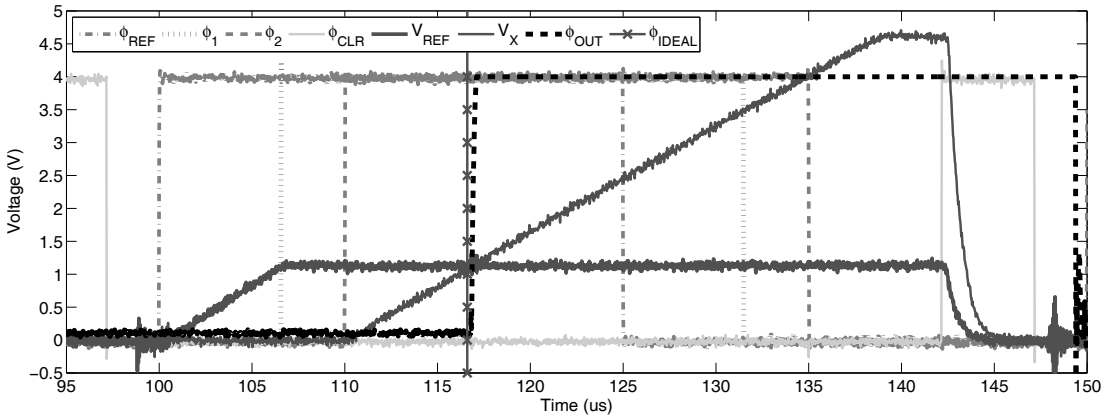


(a)

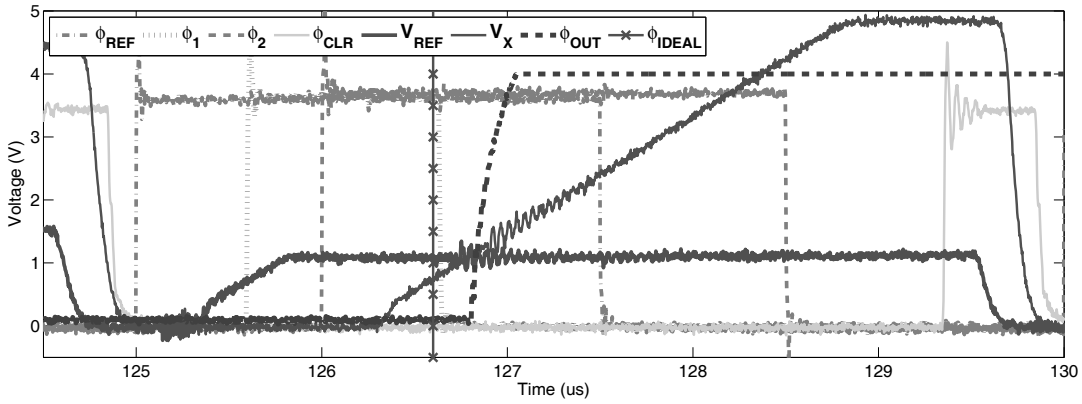


(b)

Figure 3–29: Discrete implementation of the time adder: (a) circuit schematic (b) photograph of physical circuit.



(a)



(b)

Figure 3–30: Experimental results of the discrete component implementation of a time adder at (a) 20-kHz, (b) 200-kHz.

using the oscilloscope at a rate of 2GSa/s. The output of the comparator was then scaled using MATLAB from 10V down to 4V (the same range of the input edges) for easier viewing of the results.

A comparison between the experimental results and the ideal behaviour of the time adder at 20-kHz and 200-kHz are shown in Figures 3–30(a) and 3–30(b), respectively. Comparing these to the voltage-time graph of Figure 3–5, we note that

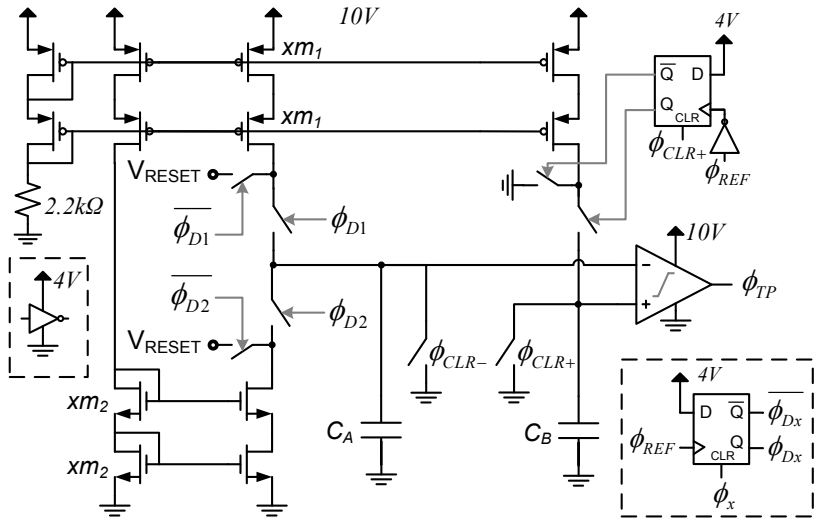


Figure 3–31: Discrete implementation of the time-mode WDTS.

the circuit agrees well with the theory presented. The noise on the input nodes of the comparator is expected due to the fact that discrete component implementations are very susceptible to thermal and switching effects. By looking at Figure 3–30(b), the limitations of the discrete circuit at high frequencies can be identified. Specifically, the figure illustrates how the non-zero prorogation time of the digital ICs and turn-on time of the analog switches result in a delay between the rising edge of ϕ_{REF} and the start of the increase of V_{REF} around $t = 125\mu s$. The effects of the limited bandwidth and non-zero rise time of the voltage comparator are also apparent at approximately $t = 127\mu s$. Both these effects result in an error between the expected and the actual time of the output edge.

3.4.2 Weighted Delayed Time-Mode Subtractor

The modified schematic for the discrete implementation of the time WDTS in which $k_1 = 2$ and $k_2 = 0.5$ is illustrated in Figure 3–31. As with the time adder,

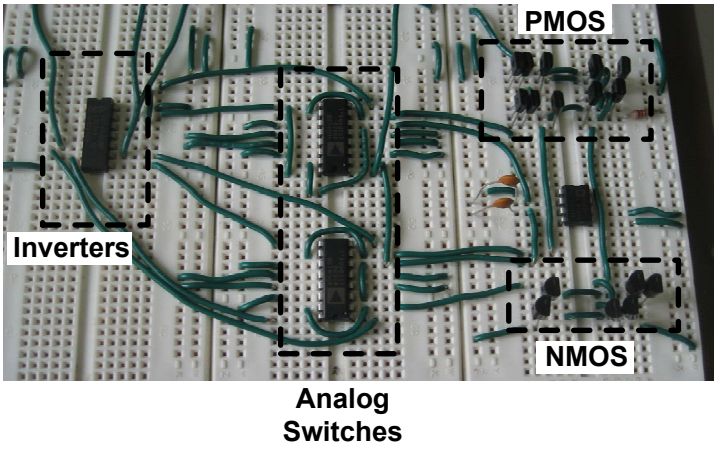


Figure 3–32: Photograph of discrete implementation of the time WDTS.

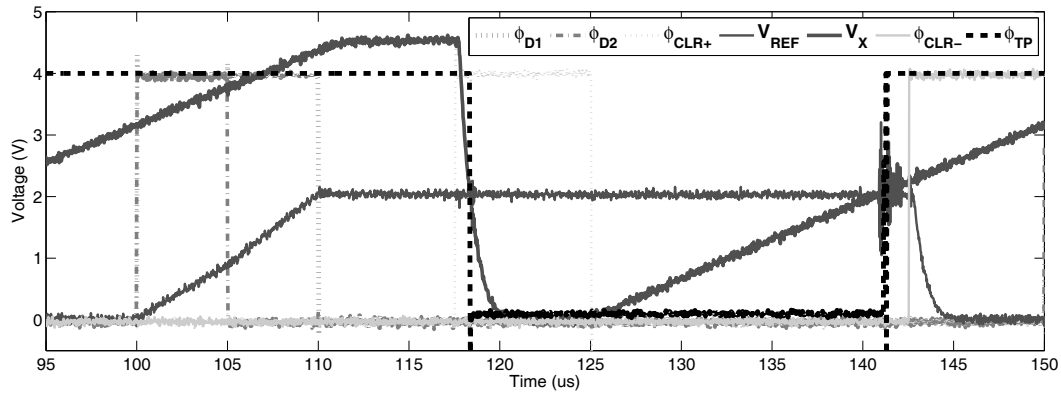
the current mirror is biased with a discrete resistor, while the charging capacitances are implemented using discrete capacitors. The half-delay cell that is included at the output of the comparator in the original design (Figure 3–9) is not included in this design. This element is omitted due to the inability to obtain components which could produce delays large enough to satisfy the slow speeds (20 kHz) of the reference clock. The frequency domain equation for the circuit shown in Figure 3–31 is therefore only delayed by half a sample:

$$\Delta T_{OUT}(z) = k_1 z^{-1/2} \Delta T_1(z) - k_2 z^{-1/2} \Delta T_2(z) \tag{3.39}$$

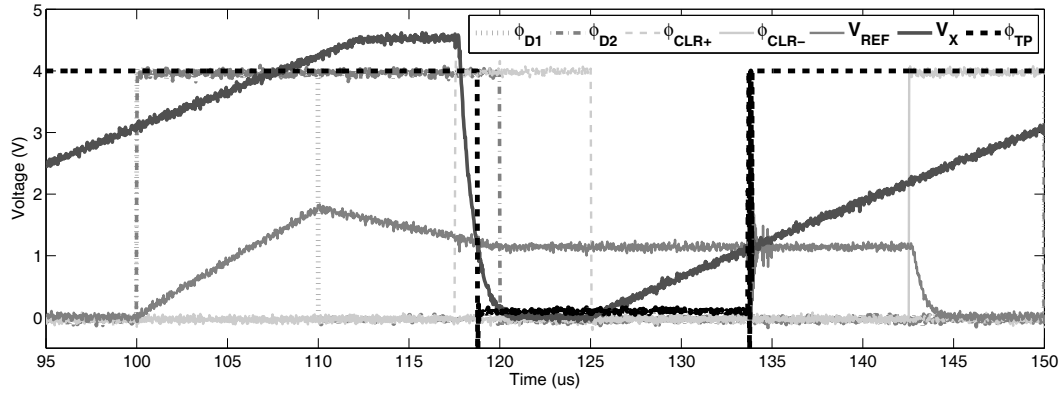
where in this implementation, $k_1 = 2$ and $k_2 = 0.5$. Note that in this case, k_1 and k_2 are determined by the ratio of the capacitors C_A, C_B and the number of discrete transistors m_1 and m_2:

$$k_1 = \frac{C_B}{C_A} m_1, k_2 = \frac{C_B}{C_A} \frac{1}{m_2} \tag{3.40}$$

To obtain $k_1 = 2$ and $k_2 = 0.5$, C_A and $C_B = 10nF$ were both set to $10nF$ with $m_1 = m_2 = 2$.



(a)



(b)

Figure 3–33: Experimental results of the discrete component implementation of a time WDTS (a) $t_1 > t_2$, (a) $t_1 < t_2$.

A photograph of the final WDTS circuit is shown in Figure 3–32. The frequency of the reference clock (T) was set to 20-kHz. The circuit was tested under two conditions. In the first, the input time-differences were selected such that $\Delta T_1 > \Delta T_2$, that is, the inputs were chosen such that the sinking switch D2 would open before D1. In the second test, $\Delta T_1 < \Delta T_2$ such that D1 would be the first switch to open.

Table 3–2: Comparison between the actual and expected WDTS output time-differences.

Circuit	ΔT_1	ΔT_2	$\Delta T_{TP-IDEAL}$	$\Delta T_{TP-ACTUAL}$	% Diff.
(a) $t_1 > t_2$	$10\mu s$	$5\mu s$	$17.5\mu s$	$16.3\mu s$	6.86%
(b) $t_1 < t_2$	$10\mu s$	$20\mu s$	$10\mu s$	$8.9\mu s$	11%

In Figure 3–33(a), the switch D2 opens at $t = 105\mu s$ and the voltage V_{REF} continues to rise (with a larger slope) until D1 closes at $t = 110\mu s$. On the other hand, the voltage at V_{REF} in Figure 3–33(b) starts to discharge through D2 after D1 opens at $t = 110\mu s$. A comparison between the actual output time-difference and the expect output time-differences is given in Table 3–2. Note that there is a small discrepancy between the expected output edge and the actual edge, which can be attributed to the mismatch between the transistors and capacitors which leads to a shift in the coefficients of the WDTS.

3.5 Building Blocks as Mixed-Signal Circuits

One of the main objectives in TMSP is to design systems where the information is carried by time-mode signals only. By achieving this, the circuit would be able to take full advantage of the properties of time-mode systems described in Chapter 1. Unfortunately, the building blocks presented in this chapter are not pure time-mode systems, rather they require both time-mode and voltage-domain signals. To obtain the desired operation, the input time-differences are first converted into analog voltages through the charging of capacitors. These voltage-domain signal are later transformed back into the time-mode by the zero-to-one transitioning of the voltage comparator. We can therefore classify the time-mode building as circuits which are

mixed-signal in nature. A discussion of the effects of the conversion between domains on the performance of these circuits, as well as measures that can be taken to minimize them will be presented in the next chapter.

3.6 Summary

In this chapter, a broad set of signal processing building blocks in the time-mode were introduced. To implement the proposed circuits, the design methodology of the general VCDU, that is, the combination of a constant current sources, charging capacitors and an analog voltage comparator, is applied and modified such that the desired circuit operation is obtained. Specifically, the mathematical operations of addition, weighted addition and subtraction, scaling and integration were presented in this methodology. Discrete implementations of the time adder and time WDTS were realized, confirming that the design methodology presented in this chapter results in correct time-mode functionality.

While both the inputs and outputs of these designs are time-difference signals, the designs themselves are not pure time-mode systems. Rather, these designs are classified as mixed-signal systems since they require a conversion to and from the voltage-domain for the desired operation. The implications of the conversion is discussed further later in this work.

In the remainder of this work, we will show how these simple building blocks can be arranged to construct higher-order time-mode systems. In particular, the implementation of second and higher-order sampled-data filters will be presented.

Chapter 4

Time-Mode Filtering

One application of a time-mode filter for sampled-data IIR filtering is illustrated in Figure 4–1(a). To perform the required filtering operation, the input voltage information is first converted into a time-difference signal using a voltage-to-time converter (VTC). The resultant time-difference signal is then processed by a time-mode IIR filter with transfer function $F(z)$. Finally, the filtered time-difference signal output is converted back into the voltage domain via a TVC.

A second potential application of a time-mode filter is shown in Figure 4–1(b). In this case, rather that performing analog computation, there is no transformation between voltage and time-difference and the system processes information in the time-mode only. Examples of potential applications of IIR filters for time-domain computation include jitter reduction in digital clocks, as well as for performing anti-aliasing when sampling time domain signals.

In the following sections, the design methodology of time-mode sampled-data IIR filters will be presented. Specifically, it will be shown how the basic set of TMSP building blocks described in Chapter 3 can be organized in order to realize IIR filter transfer functions of second and higher-order. These systems differ from

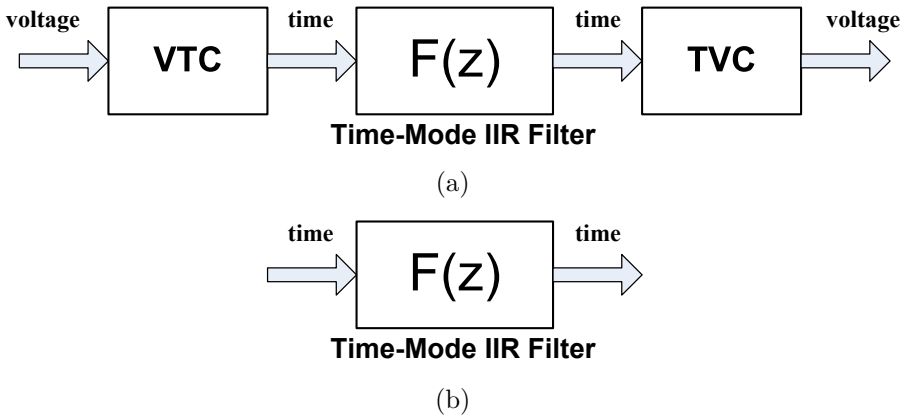


Figure 4–1: Application of a time-mode filter for (a) analog and (b) time-mode computation.

the FIR structures described in [20] in that IIR filters have system poles, as well as an inherent feedback structure, which is designed to place the poles according to a desired response. In terms of hardware, IIR systems require both the delay and summation operations of the FIR structure as well as an additional integrator element with localized feedback.

4.1 Low-Pass IIR Filter Implementation

4.1.1 Design Methodology

The direct form block diagram of a second-order low-pass sampled-data IIR filter is shown in Figure 4–2(a). As illustrated in the diagram, four basic signal processing operations are required to implement the time-mode filter: addition, subtraction, scaling, and unit delay. While time-mode addition and unit delay are easily realized in the time-mode (see relevant sections in the previous chapter), direct time subtraction and time attenuation are not causal operations, and thus, the direct form block diagram of Figure 4–2(a) is unrealizable. To implement the time-mode IIR filter of

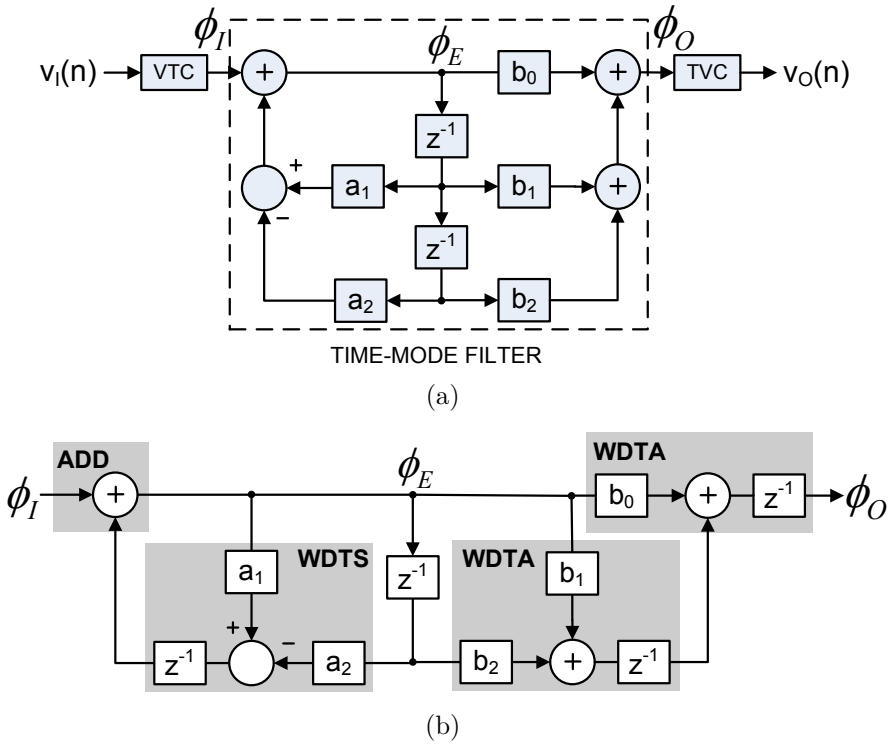


Figure 4–2: (a) Direct form, and (b) rearranged block diagram of a second-order low-pass time-mode filter.

Figure 4–2(a), the block diagram of the filter is first re-arranged in order to facilitate the substitution of the blocks described in Chapter 3.

The final topology of the IIR filter is shown in Figure 4–2(b). If $\Delta T_I(z)$, $\Delta T_E(z)$ and $\Delta T_O(z)$ define the z-transforms of the time-differences between the reference signal ϕ_{REF} and digital signals ϕ_I, ϕ_E, and ϕ_O respectively, then the following feedback and feedforward difference equations for the modified block diagram are obtained

$$\Delta T_E(z) = \Delta T_I(z) + z^{-1}(a_1 \Delta T_E(z) - a_2 z^{-1} \Delta T_E(z)) \tag{4.1}$$

$$\Delta T_O(z) = z^{-1}(b_0 \Delta T_E(z) + z^{-1}(b_1 \Delta T_E(z) + b_2 z^{-1} \Delta T_E(z))) \tag{4.2}$$

Combining (4.1) and (4.2) yields the system transfer function of the circuit

$$\frac{\Delta T_O(z)}{\Delta T_I(z)} = z^{-1}\frac{b_0 + b_1 z^{-1} + b_2 z^{-2}}{1 - a_1 z^{-1} + a_2 z^{-2}} = F_{LP}(z) \tag{4.3}$$

which is equal to the transfer function of the original block diagram, a second-order low-pass filter, delayed by one sample. The additional z^{-1} factor in Equation (4.3) is equivalent to adding a zero to the system at $z = \infty$, which will only alter the phase and not the magnitude response of the system. The low-pass response of the system is thus conserved. For the remainder of this work, we will refer to the time-mode low-pass sampled-data IIR filter as a low-pass time-mode filter (LPTF).

In the modified topology of Figure 4–2(b), the signal processing operations required are addition, weighted delayed addition, weighted delayed subtraction, and unit delay, all of which can be implemented using the methodology presented in the last chapter. By replacing the corresponding operations by the blocks described in Chapter 3, the time-mode filter can be realized by arranging a time-adder, a WDTS, two WDTAs and a unit delay element.

Note that the low-pass time-mode filter may also be expressed in terms of its sampled-data time-difference equation

$$\Delta T_O(n) - a_1\Delta T_O(n-1) + a_2\Delta T_O(n-2) = b_0\Delta T_I(n-1) + b_1\Delta T_I(n-2) + b_2\Delta T_I(n-3) \tag{4.4}$$

where here $\Delta T_I(n)$ and $\Delta T_O(n)$ refer to the time-differences between the rising edges of the digital signals ϕ_I and ϕ_O with respect to the rising edge of the common reference clock signal ϕ_{REF}.

Note that for this work, the VTC and TVC components used to transform the voltage and time-mode signals in Figure 4–2(a), as well as for all other filter

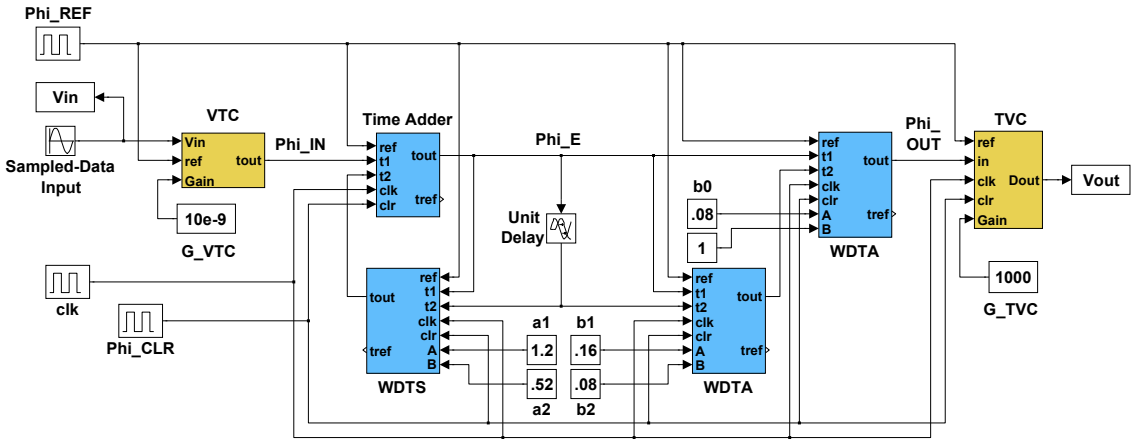


Figure 4–3: Simulink model of a second-order LPTF

structures that will be presented in this work, are implemented off chip in order to minimize their impact on the system.

In the remainder of this section, the implementation of a second-order low-pass time-mode filter (LPTF) with magnitude response

$$\frac{\Delta T_O(z)}{\Delta T_I(z)} = z^{-1}\frac{0.08 + 0.16z^{-1} + 0.08z^{-2}}{1 - 1.2z^{-1} + 0.52z^{-2}} \tag{4.5}$$

will be presented. The filter will first be modeled and simulated using MATLAB, followed by a space-level spectra extracted transistor level simulation of the circuit schematic in a 0.18-μm process. For this study, the sampling rate, that is, the frequency of the reference clock ϕ_{REF} was set to 5 MHz.

4.1.2 Modeling with MATLAB and Simulink

A second-order LPTF with transfer function described by Equation (4.5) was simulated in MATLAB using the block diagram shown in Figure 4–3.

To implement the various time-mode building blocks, a digital counter and very fast clock *clk* were used to simulate the action of the current source and capacitor,

while the voltage comparator was replaced in Simulink by a digital comparator (i.e. relational operator in MATLAB). A comparison between the operation of the analog current source-capacitor and the digital counter-clock combinations is illustrated in Figure 4–4. In the Simulink model, the action of charging a capacitor with a constant current to obtain a linear voltage (illustrated on the left) is mimicked by counting the number of periods of the fast clock *clk* that pass during the the input time-difference $\Delta T_I = t_I - t_{REF}$ using a digital counter (illustrated on the right). The equation for the final count value is

$$Cnt = \frac{1}{T_{clk}}(t_I - t_{REF}) = f_{clk}(t_I - t_{REF}) \tag{4.6}$$

where T_{clk} is the period of the fast clock *clk*. The MATLAB model can be made equivalent to that of the original analog system by setting $T_{clk} = C/I$. One problem with this method is that the clock-counter solution can only measure discrete values of time-difference, and thus, will inject quantization noise into the system. To effects of this noise can be minimized by decreasing the period of the fast clock T_{clk} to increase the resolution of the system. The period of the fast clock was set to 10 ps, resulting in a maximum resolution of approximately 14.3 bits for a 5 MHz reference clock.

For the Simulink model of Figure 4–3, the input voltage was implemented using a sampled sinusoid block, which was fed to the VTC. The VTC was implemented using a preamplifier stage, where the input voltage was first multiplied by a voltage-to-time constant G_{VTC}, and a transport delay block, which converted the voltage value into a time-delay on ϕ_{REF}. A diagram of the VTC Simulink model is shown in Figure 4–5. The TVC was designed using the same methodology as the time-mode

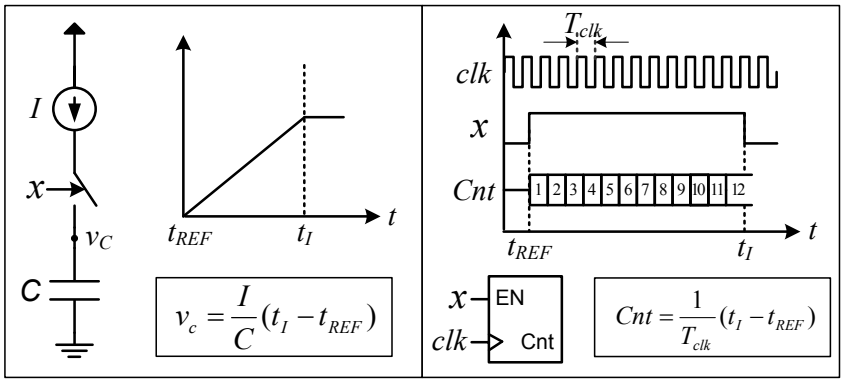


Figure 4–4: Operation of the counter and fast clock for MATLAB model

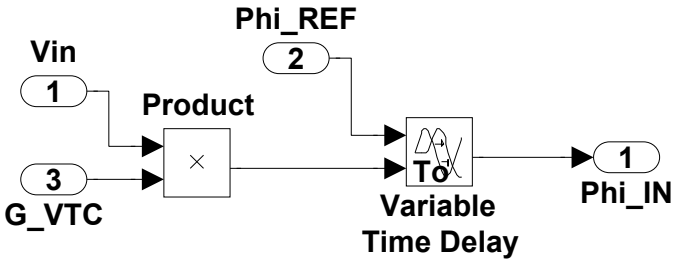
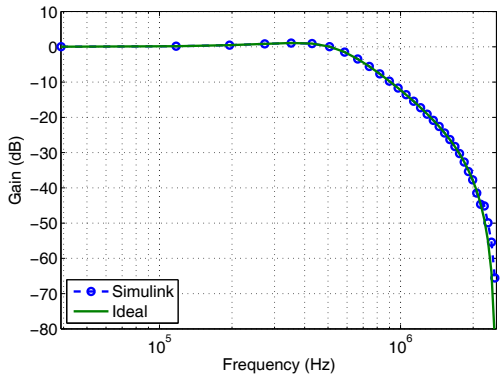


Figure 4–5: Simulink model of a VTC

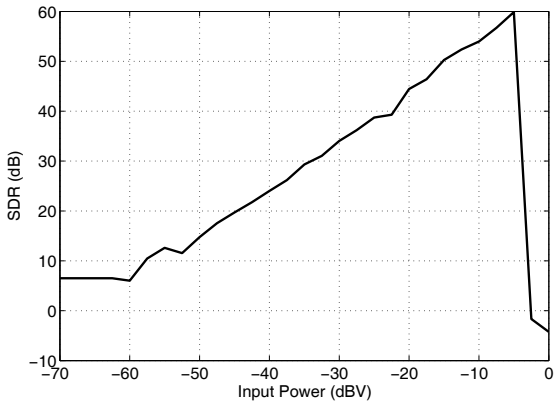
building blocks, that is, the time-difference between the reference and output edges was converted into the number of periods of *clk* that passed during this time. The final output voltage was obtained by dividing the count of the TVC by a constant G_{TVC} to compensate for the initial conversion factor of the VTC. In this case, the relationship between the constants G_{VTC} and G_{TVC} is

$$G_{VTC} = G_{TVC} T_{clk} \tag{4.7}$$

Similarly to the TVC, the scaling of the time-differences was achieved by multiplying the count value by a scale factor (implemented in MATLAB using constants and



(a)



(b)

Figure 4–6: (a) Frequency response and (b) dynamic range of the LPTF (273 kHz) modeled in Simulink.

a multiplier). Finally, the unit and half-unit delay cells were implemented using transport delay blocks.

A simulation with 128 points was performed and the frequency response, as well as the filter's signal-to-distortion ratio (SDR) was calculated. A comparison between the Simulink model's frequency response and the ideal frequency response of the filter described in Equation (4.5) is shown in Figure 4–6(a). The filter was found to have a maximum SDR of 59.9 dB, corresponding to a resolution of 9.65 bits. The

dynamic range of the MATLAB model, limited by the quantization noise from the finite resolution of MATLAB, is shown in Figure 4–6(b) at a frequency of 273 kHz.

4.1.3 Transistor Design and Simulation

The second-order LPTF was designed in a standard 0.18-μm CMOS process by arranging the transistor schematics of the building blocks presented in Chapter 3. The schematic showing the substitution of the time-mode buidling blocks into the block diagram is presented in Figure 4–7. For this schematic, all logic gates were taken from standard CMOS cell libraries. The full unit was built by cascading two half-unit delay cells, the transistor schematics of which were shown in Figures 3–25 and 3–26.

The system in Figure 4–7 was simulated using Spectre. The input signal was biased at a DC voltage of 200-mV and carried a sinusoid with a peak amplitude of 50-mV. One hundred and twenty-eight points were collected and the extracted frequency response, as well as the filter's SDR was measured. The measured filter response as well as the ideal response are shown in Figure 4–8(a). The mismatch between the simulation results and ideal filter response at high frequencies is attributed to a shift in the filter coefficients resulting from the layout and extraction process. The maximum SDR of the simulated circuit was found to be 44.1 dB, corresponding to a resolution of 7.1 bits. The power consumption of the circuit was 0.76 mW. The dynamic range of the simulated circuit at 273 kHz is shown in Figure 4–8(b). Note that the steep drop in SDR occurs because at high input powers, the time-mode difference falls outside the clock's period length and the circuit enters an undefined state.

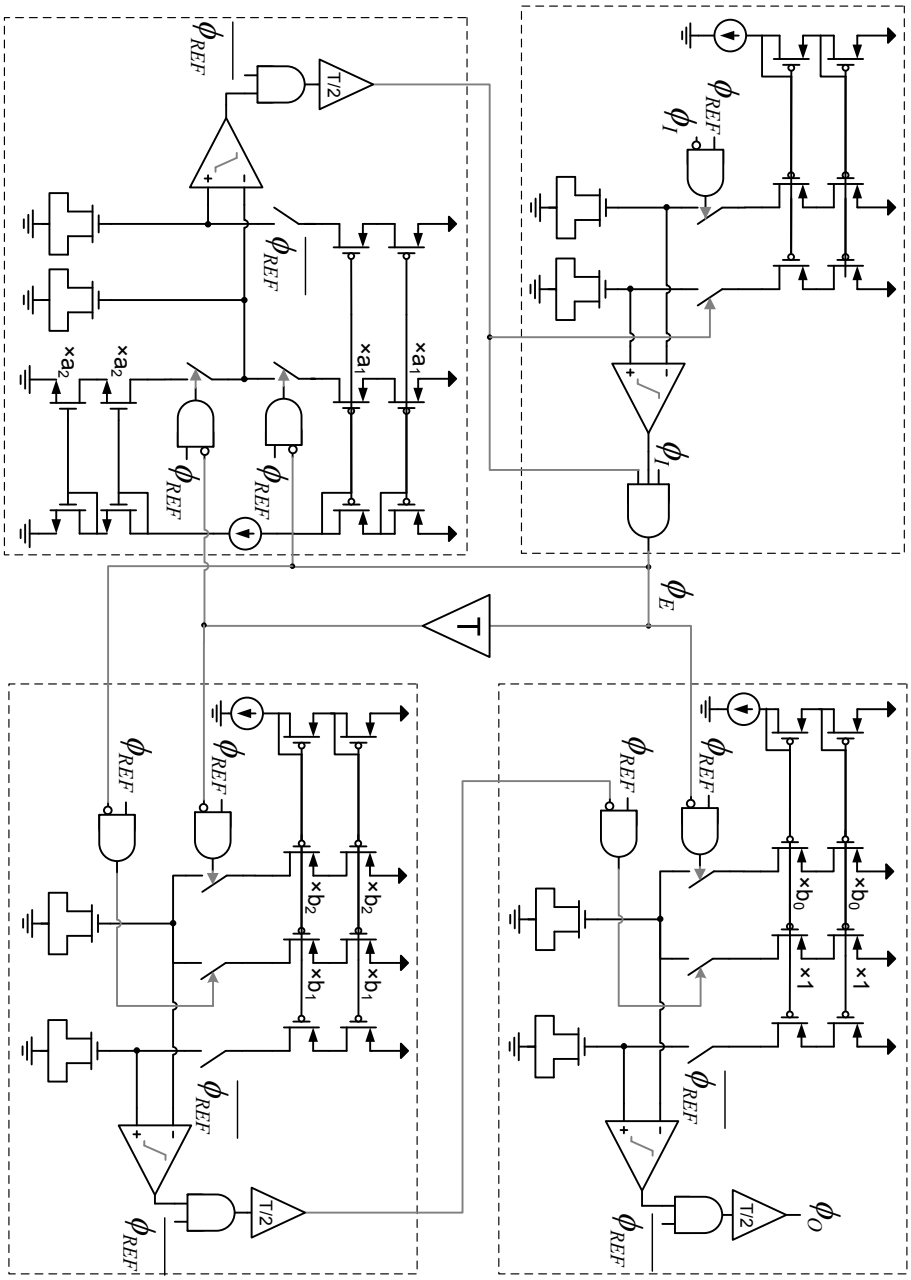
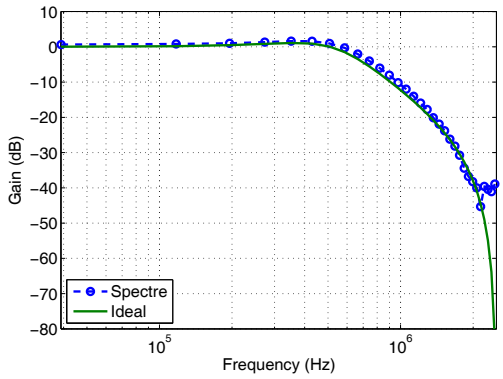
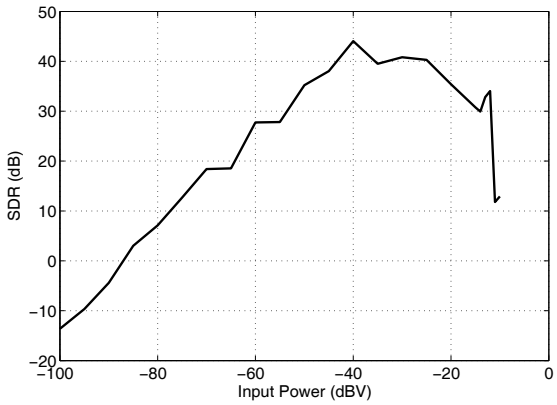


Figure 4–7: Transistor schematic of a second-order LPTF. (Note $b_0 = b_2 = 0.08$, $b_1 = 0.16$, $a_1 = 1.2$ and $a_2 = 0.52$)



(a)



(b)

Figure 4–8: (a) Frequency response and (b) Dynamic range of the transistor-level simulated LPTF.

4.2 Band-Reject IIR Filter Implementation

4.2.1 Design Methodology

The direct form block diagram of a second-order band-reject sampled-data IIR filter is shown in Figure 4–9(a). This topology differs from the low-pass design in that the operator acting on the b_1 and b_2 branches of the feedforward path is changed from a sum to a difference. The simplest way to implement a time-mode band-reject

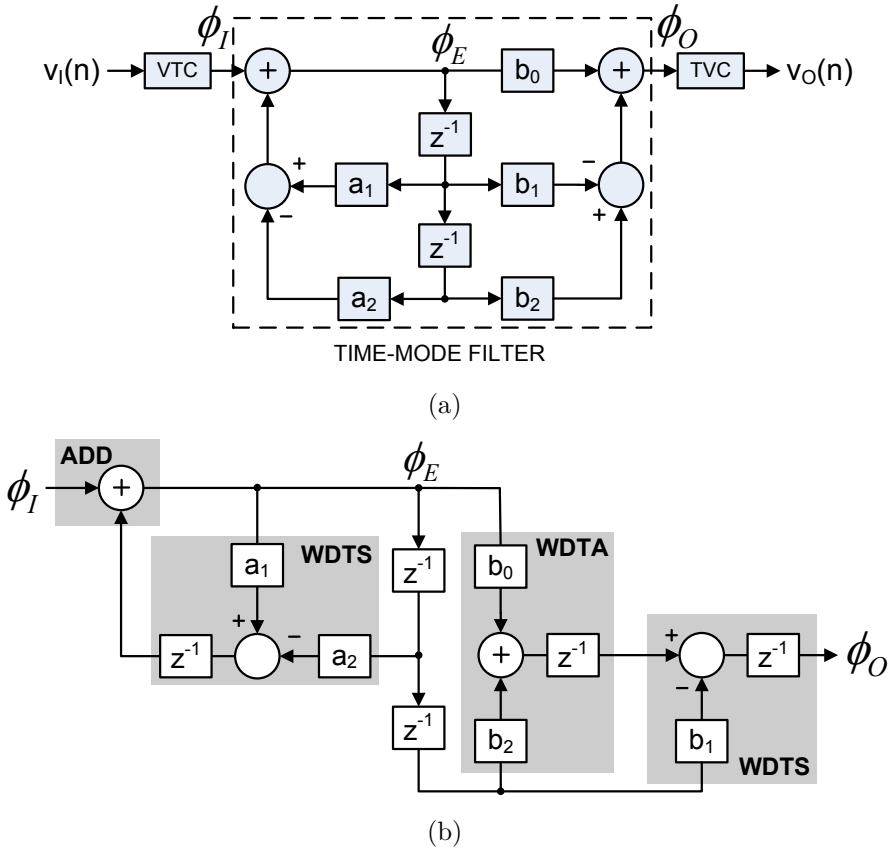


Figure 4–9: (a) Direct form, and (b) rearranged block diagram of a second-order band-reject time-mode filter.

IIR filter (BRTF) is to therefore replace the bottom WDTA in the low-pass structure of Figure 4–2(b) with a WDTS. The problem with this topology arises from the fact that for most sampled-data filters, the coefficients $b_1 > b_2$. In this case, if ΔT_E is defined as the time-difference between the rising edge of the reference ϕ_{REF} and the signal ϕ_E shown in Figure 4–9(a), then the difference at the output of the feedforward (rightmost) subtractor element in the block diagram becomes

$$b_2 \Delta T_E(n-2) - b_1 \Delta T_E(n-1) \tag{4.8}$$

which, given $b_1 > b_2$, can be a negative quantity. Since our definition of time-mode signals only deals with positive time-differences, the circuits presented in this work would be unable to process these negative time-mode outputs. To solve this issue, the feedforward portion of the filter is rearranged as shown in Figure 4–9(b) such that the addition of the b_0 and b_2 terms of the error signal ΔT_E is performed before the subtraction of the b_1 term. An analysis of the block diagram of Figure 4–9(b) results in the following system difference equation

$$\frac{\Delta T_O(z)}{\Delta T_I(z)} = z^{-1}\frac{b_0 - b_1 z^{-1} + b_2 z^{-2}}{1 - a_1 z^{-1} + a_2 z^{-2}} = F_{BR}(z) \tag{4.9}$$

which is a delayed version of the ideal frequency response of a band-reject filter.

A single-ended second-order time-mode IIR filter with a band-reject frequency response of

$$\frac{\Delta T_O(z)}{\Delta T_I(z)} = z^{-1}\frac{0.81 - 1.31 z^{-1} + 0.81 z^{-2}}{1 - 1.31 z^{-1} + 0.62 z^{-2}} \tag{4.10}$$

was implemented for this design. The frequency of reference clock ϕ_{REF}, was again set to 5 MHz. The BRTF described by Equation (4.10) was modeled using Simulink and MATLAB, as well as simulated at the transistor level using Spectre. The results of these implementations are presented in the remainder of this section.

4.2.2 Modeling with MATLAB and Simulink

The second-order BRTF described in Equation (4.10) was simulated in MATLAB using the Simulink model shown in Figure 4–10.

As with the LPTF implemented in MATLAB, the building blocks were implemented using a very fast clock *clk* and digital counters. The values of the voltage-to-time conversion factor G_{VTC} and time-to-voltage constant G_{TVC} were modified (see

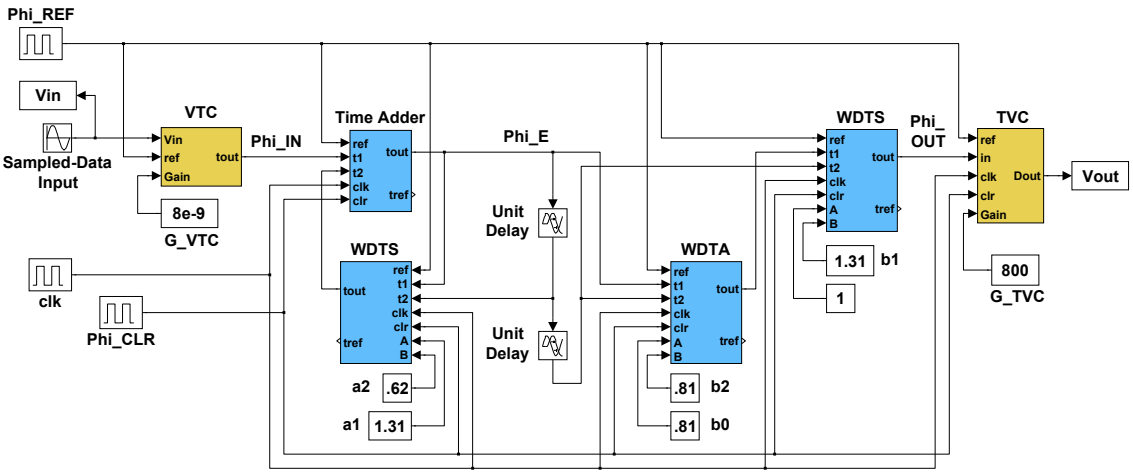


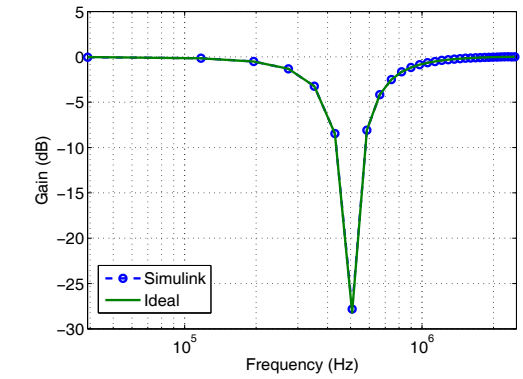
Figure 4–10: Simulink model of a second-order BRTF

Figure 4–10 for values) such that the maximum swing at the error-signal ϕ_E was realized.

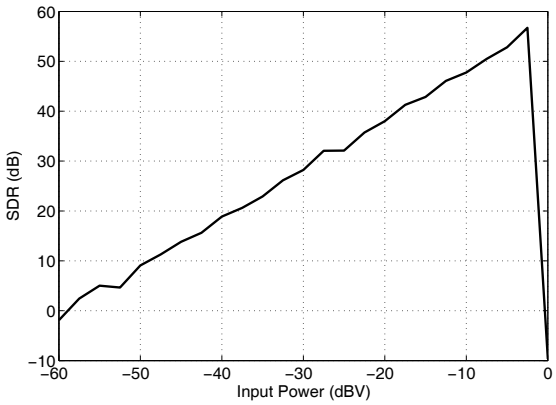
A simulation with 128 points was collected and the frequency response, as well as the filter's SDR was calculated. A comparison between the Simulink model's frequency response and the ideal frequency response of the filter described in Equation (4.10) is shown in Figure 4–11(a). The filter was found to have a maximum SDR of 56.7 dB with a full scale input of 1V, corresponding to a resolution of 9.1 bits. The dynamic range of the MATLAB model, limited by quantization noise, is shown in Figure 4–6(b) at a frequency of 117 kHz.

4.2.3 Transistor Design and Simulation

The second-order BRTF was designed in a standard CMOS process by connecting the transistor schematics of the building blocks presented in Chapter 3. The schematic showing the substitution of the time-mode building blocks into the block diagram is presented in Figure 4–12.



(a)



(b)

Figure 4–11: (a) Frequency response and (b) dynamic range of the BRTF (117 kHz) modeled in Simulink.

The system in Figure 4–12 was design in a 0.18-μm CMOS process and simulated using Spectre. The input signal was biased at a DC voltage of 325-mV and carried a sinusoid with a peak amplitude of 50-mV. One hundred and twenty-eight points were collected and the schematic frequency response, as well as the filter's SDR were measured. The measured filter response as well as the ideal response are shown in Figure 4–13(a). The maximum SDR of simulated circuit was found to be 41.6 dB,

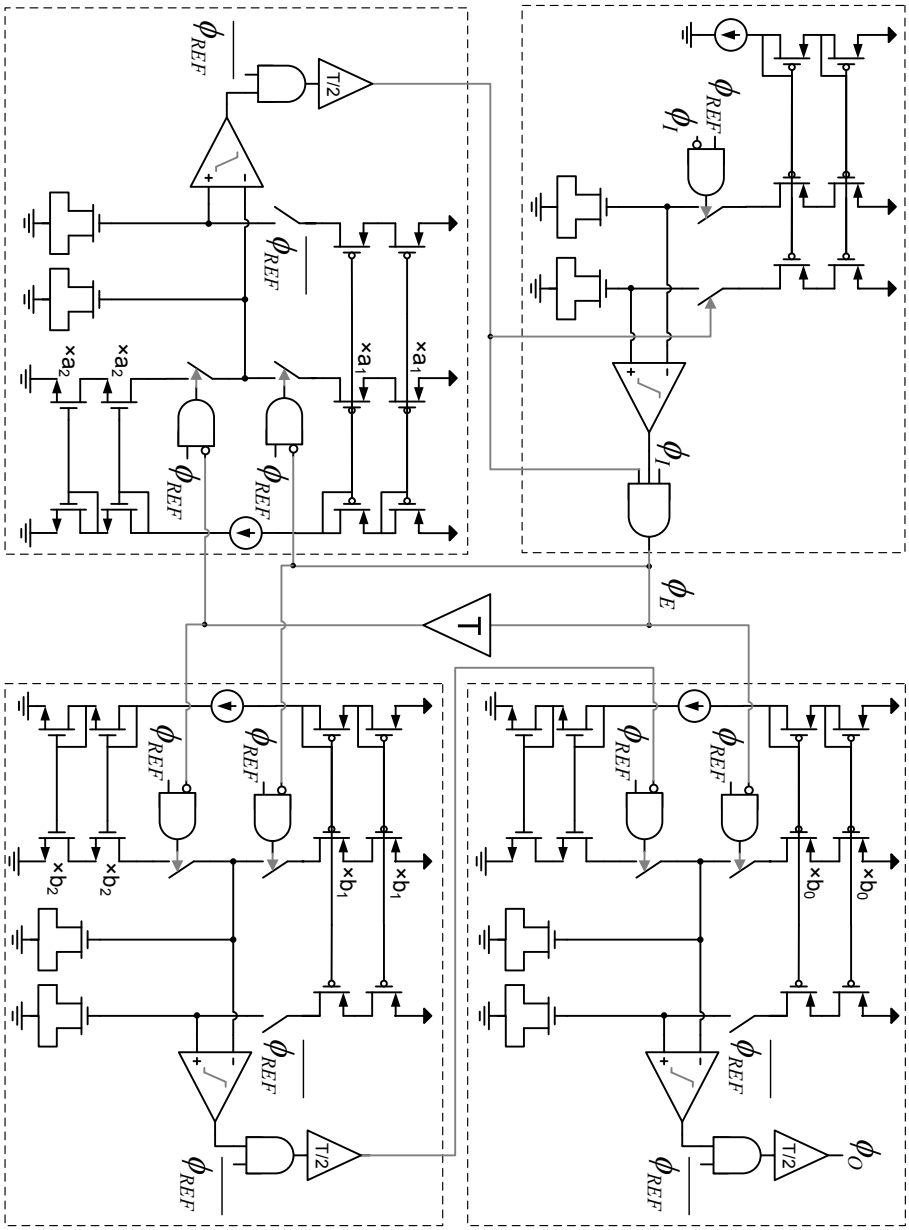
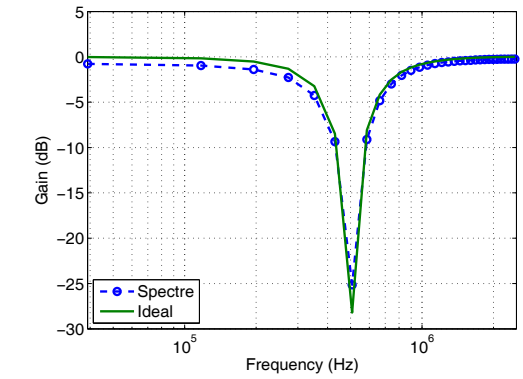
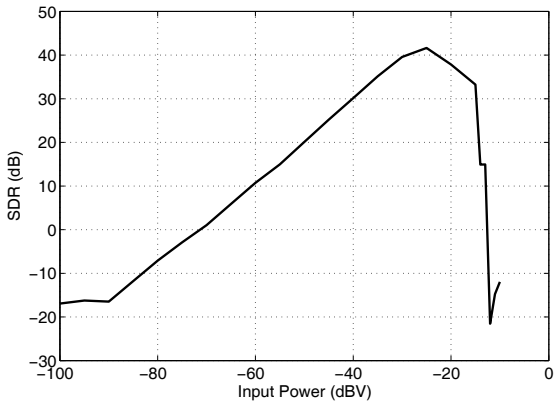


Figure 4–12: Transistor schematic of a second-order BRTF. (Note $b_0 = b_2 = 0.81$, $a_1 = b_1 = 1.31$ and $a_2 = 0.62$)



(a)



(b)

Figure 4–13: (a) Frequency response and (b) Dynamic range of the transistor-level simulated BRTF.

corresponding to a resolution of 6.6 bits. The dynamic range of the simulated circuit is shown in Figure 4–13(b).

4.3 High-Pass/Band-Pass IIR Filter Implementation

4.3.1 Design Methodology

Since the ideal high-pass and band-pass operation removes the low frequency (i.e. DC) component of the input signal, negative time-difference values $\Delta T_X(n) < 0$

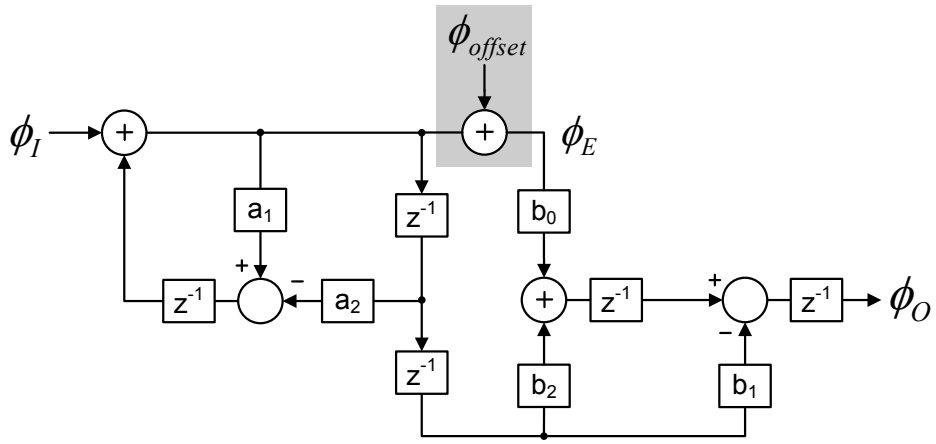


Figure 4–14: Block diagram of a second-order high-pass IIR time-mode filter.

would be required to properly represent the output signal as a time-difference. Since negative time-differences are not defined in the methodology presented in this work, ideal high-pass and band-pass time-mode filters cannot be directly realized.

Consider a second-order high-pass time-mode Chebyshev filter, whose transfer function has the same form as a band-reject filter depicted in Figure 4–9(b), i.e.,

$$\frac{\Delta T_O(z)}{\Delta T_I(z)} = z^{-1}\frac{b_0 - b_1 z^{-1} + b_2 z^{-2}}{1 - a_1 z^{-1} + a_2 z^{-2}} = F_{HP}(z) \tag{4.11}$$

To compensate for the lost DC component removed by the filter function, a DC restorer function is incorporated in the existing filter structure. This is achieved by injecting a DC signal into the positive input terminal of the final feed forward WDTS as shown in Figure 4–14. In practice, this DC signal is simply a delayed version of the reference clock. Through direct signal-flow graph analysis, we can write the signal that appears at the output of the filter as

$$\Delta T_O(z) = F_{HP}(z)\Delta T_I(z) + b_0 z^{-2}\Delta T_{offset} \tag{4.12}$$

Next, consider ΔT_{offset} having a constant delayed value of τ_{offset}, then we can write (4.12) as

$$\Delta T_O(z) = F_{HP}(z)\Delta T_I(z) + b_0 z^{-2}\tau_{offset} \tag{4.13}$$

By the appropriate choice of τ_{offset} we will establish the DC reference point for the filter and in turn maximize the time-difference output swing (i.e., SNDR of the filter).

It can also be shown, using a similar argument as presented above, how an delayed version of the clock reference can be used as a DC restorer for the design of band-pass time-mode filters.

4.3.2 Modelling with MATLAB and Simulink

To illustrate the design of high-pass and band-pass time-mode filters, a second-order time-mode IIR filter with high-pass Chebyshev frequency response

$$\frac{\Delta T_O(z)}{\Delta T_I(z)} = z^{-1}\frac{0.51 - 1.02z^{-1} + 0.51z^{-2}}{1 - 0.88z^{-1} + 0.42z^{-2}} \tag{4.14}$$

was implemented in Simulink. Since the transfer function above has the same form as the band-reject case, a similar model was used for the MATLAB simulations. The compensation delay unit was modeled using a constant transport delay block at the input of the second feedforward WDTS. The full Simulink model is illustrated in Figure 4–15.

A comparison of the MATLAB frequency response of the filter compared with the ideal filter response of Equation (4.14) is shown in Figure 4–16(a). The high-pass filter was found to have a maximum SDR of 60.8 dB, corresponding to a resolution of 9.8 bits. The dynamic range of the Simulink model at 977 kHz is shown in Figure 4–16(b).

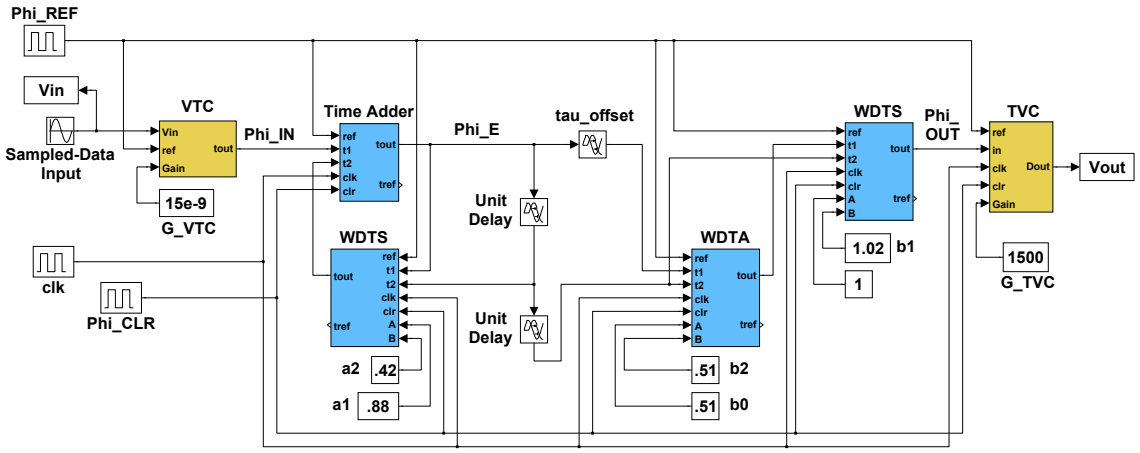


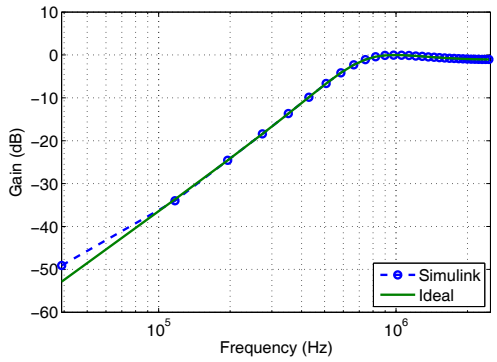
Figure 4–15: Simulink model of a second-order HPTF

4.3.3 Transistor Design and Simulation

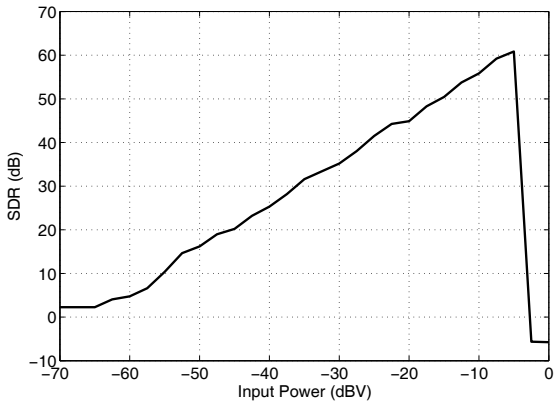
As with the Simulink the model, the transistor level schematic of the second-order HPTF is identical to that of the band-reject filter, except that in this case, an extra delay unit is required in the feedforward path. For the Spectre simulation, this delay unit was simulated using an ideal delay block provided in the Cadence libraries. To implemented the extra delay block using actual transistors, a string of inverters, similar to those used to implemented the half and full unit-delays blocks, may be utilized. The added biasing delay may be also obtained from the inherent delay offset of the voltage comparators, the theory of which will be further discussed later in this chapter. In this case, no extra circuit components would be required.

By substituting the time-mode building blocks into the modified block diagram for the high-pass filter circuit, the more complete schematic of Figure 4–17 is obtained.

The system in Figure 4–17 was design in a 0.18-μm CMOS process and simulated using Spectre. The input signal was biased at a DC voltage of 325-mV and carried a



(a)



(b)

Figure 4–16: (a) Frequency response and (b) dynamic range of the HPTF (997 kHz) modeled in Simulink.

sinusoid with a peak amplitude of 50-mV. The DC compensation delay τ_{OFFSET} was set to 40-ns. One hundred and twenty-eight points were collected and the schematic frequency response, as well as the filter's SNR and SNDR were measured. The measured high-pass filter response as well as the ideal behaviour of the system are shown in Figure 4–18(a). The maximum SDR of the simulated circuit was found to be 46.9 dB, respectively, corresponding to a resolution of 7.5 bits. The dynamic range of the simulated circuit is shown in Figure 4–18(b).

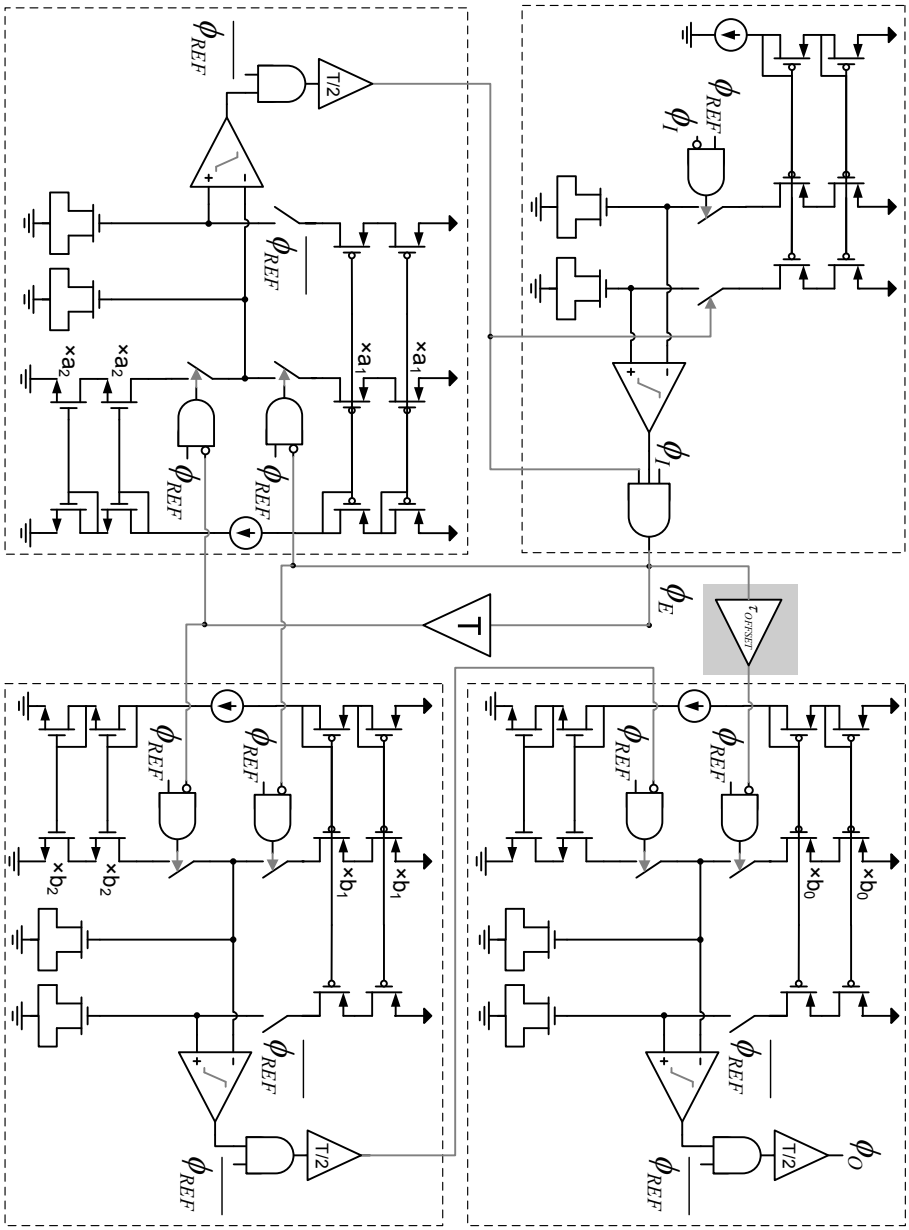
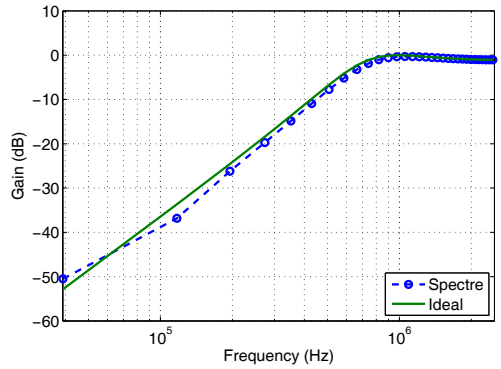
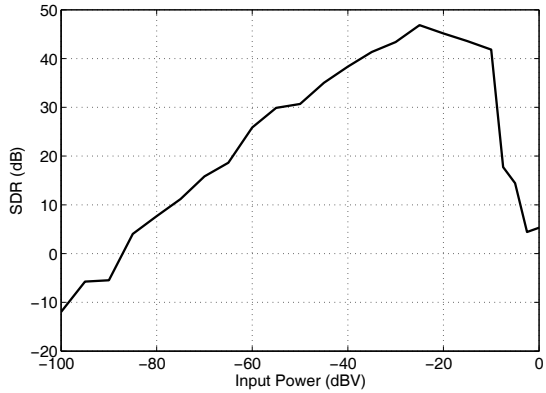


Figure 4–17: Transistor schematic of a second-order HPTF. (Note $b_0 = b_2 = 0.51$, $b_1 = 1.02$, $b_1 = 0.88$ and $a_2 = 0.42$)



(a)



(b)

Figure 4–18: (a) Frequency response and (b) Dynamic range of the transistor-level simulated BRTF.

4.4 Implementation Issues

There are several implementation issues in the design of time-mode IIR filters which are inherent to the architecture of the filter, as well as the design of the time-mode building blocks. Some of these issues manifest themselves directly in the time-mode, while some, due to the fact that the time-mode building blocks presented are mixed analog/digital designs, are inherent to the analog voltage domain but still

can affect the time-mode operation of the circuit. These issues are discussed in this section.

4.4.1 Effect of Positive-Only Delay on Dynamic Range

One of the most significant challenges with respect to the design of time-mode filters is the effect of supporting only positive time-delays has on the dynamic range of the signals in the filter. To demonstrate this issue, consider the transfer function between the input time-difference ΔT_I and the error signal ΔT_E in the block diagram of a low-pass filter shown in Figure 4–2(b):

$$\frac{\Delta T_E(z)}{\Delta T_I(z)} = \frac{1}{1 - a_1 z^{-1} + a_2 z^{-2}} \tag{4.15}$$

Since the implemented filter only allows positive time-difference values, the DC quantities of the signals will be non-zero values. The DC transfer function can be obtained by replacing $z = \exp^{j(0)} = 1$ into the equation above:

$$DC_{gain} = \frac{1}{1 - a_1 + a_2} \tag{4.16}$$

Note that as the difference $1 - a_1 + a_2$ in the equation above becomes smaller, the DC gain of the error signal will increase, decreasing the available range for the signal component. In fact, it can be shown that reducing the pole frequency by a factor k results in an amplification of the DC gain by the same factor. Since the dynamic range of the time-mode signal is limited by the period of the reference clock T, this means that as the pole frequency becomes lower, so does the maximum allowable input signal.

Note that the above analysis pertains to all of the nodes in the filter circuit of Figure 4–2(b). While this issues can cause significant limitations to the available

dynamic range in the circuit, this information is easily obtained from the filter coefficients, and thus, the input signal can be adjusted to fit within the predetermined limits.

4.4.2 Comparator Time-Offset and Compensation

A second design issue which is directly related to the limitations of positive-only delay is the time-offset generated by the comparator. As previously described in Section 3.3.1, this output delay results from a combination of the mapping of the input-referred voltage offset V_{OS} to a time-offset through the charging of a capacitor C via a current I, and the finite rise time t_{RISE} of the output associated with the finite bandwidth and gain of the pre-amplifier. Determined to be relatively constant over the linear range of the pre-amplifier, the comparator time-delay t_{offset} was given by the equation:

$$t_{offset} = C/I \cdot V_{OS} + t_{RISE} \tag{4.17}$$

To illustrate the effect of the comparator offset of each time-mode building block on the operation of a time-mode IIR filter, consider the modified block diagram of a second-order low-pass IIR filter presented earlier in this Chapter. A block diagram of the low-pass filter that includes the comparator added offsets of the time adder, the WDTS, and the two WDTAs is shown in Figure 4–19(a). By lumping together the time-offset components of the time-adder and feedback WDTS, as well as the two feedforward WDTAs, the modified block diagram of Figure 4–19(b) is obtained, where now

$$\Delta T_{offset,FB} = \tau_{offset,1} + \tau_{offset,2} \tag{4.18}$$

$$\Delta T_{offset,FF} = z^{-1}\tau_{offset,3} + \tau_{offset,4} \tag{4.19}$$

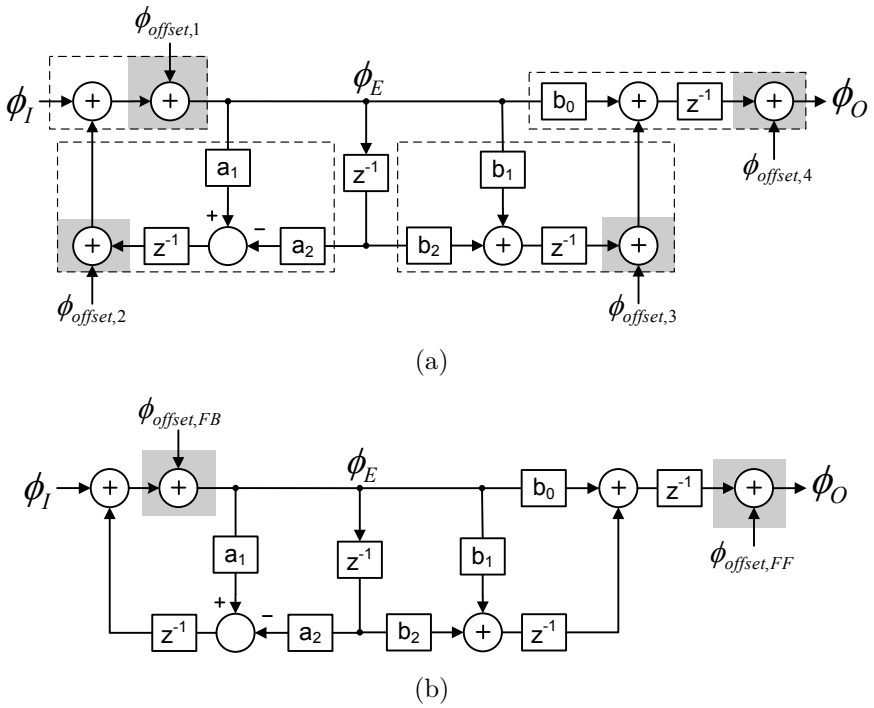


Figure 4–19: Block diagram of a second-order low-pass time-mode filter with (a) comparator time-offsets, (b) lumped feedback and feedforward time-offsets.

are the expressions for the feedback delay $\Delta T_{offset,FB}$ and the feedforward delay $\Delta T_{offset,FF}$ respectively.

Since $\Delta T_{offset,FF}$ is injected into the circuit at the output, it does not have a significant effect on the signal range. $\Delta T_{offset,FB}$, on the other hand, is injected in the feedback loop, and thus will be amplified by the DC gain of the error signal given by Equation 4.16 as described in Section 4.4.1. The same tradeoff between the reduction of the pole frequency vs maximum allowable input signal applies in this case. In order to ensure that the range of the input signal is sufficient, the time-mode building blocks must be modified to compensate for the time-offset of the comparator.

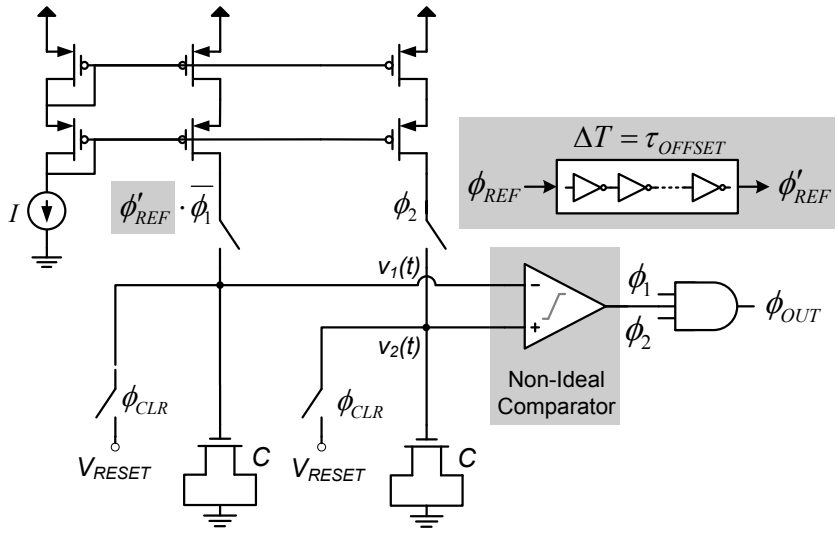


Figure 4–20: Compensating for the comparator offset in the time adder.

To illustrate how this is performed, consider the time adder circuit presented in Chapter 3. The time-difference equation of the adder can be modified to include the offset of the comparator, τ_{OFFSET}, as follows:

$$\Delta T_{OUT}(n) = \Delta T_1(n) + \Delta T_2(n) + \tau_{OFFSET} \tag{4.20}$$

To compensate for the comparator offset delay in the time-adder, the reference clock ϕ_{REF} is delayed by an amount τ_{OFFSET} using a string of inverters, as illustrated in Figure 4–20. This results in the following relationship between the time at which the rising edges of ϕ_{REF} and ϕ'_{REF} occur:

$$t'_{REF} = t_{REF} + \tau_{OFFSET} \tag{4.21}$$

The analysis of the operation of the circuit of Figure 4–20 is equivalent to that of the original time-adder. In this case, the final equation relating the time of occurrence

of the rising edges of the control signals becomes:

$$t_{OUT} - t_{REF} = (t_1 - t'_{REF}) + (t_2 - t_{REF}) + \tau_{OFFSET} \tag{4.22}$$

If (4.21) is replaced in (4.22), the τ_{OFFSET} terms cancel, and we are left with the ideal equation for time-addition. Note that this method also has an effect on the input time-difference range of $\Delta T_1(n)$ and $\Delta T_2(n)$. Specifically, it must be ensured that the rising edges of ϕ_1 and ϕ_2 both occur after that of ϕ'_{REF}, thus setting a lower limit of $\Delta T_{MIN} = \tau_{OFFSET}$ on the input time-differences $\Delta T_1(n)$ and $\Delta T_2(n)$.

To compensate for the delay offset of the comparator in the WDTA and WDTS, there is no need to add extra circuitry since the output edge already passes through a delay element (half-unit delay) before the output. Therefore, the time-offset can be compensated for directly by modifying the half-unit delay element so that it produces a delay of

$$\Delta T = T/2 - \tau_{OFFSET} \tag{4.23}$$

This is achieved by simply changing the size of the inverter transistors in the unit-delay block. Also note, that as with the unit and half-unit delay elements described in Chapter 3, the compensation delay units for both the adder and WDTS/WDTA can be voltage controlled to allow for tuning.

4.4.3 Jitter Noise

Jitter noise on the reference clock ϕ_{REF} may arise from thermal noise, poor power or ground supplies, or EMI from external lines or devices. This jitter affects the system by moving the rising edge of the clock at each sampling period n by a random amount $T_j(n)$. Depending on the type of filter designed and the filter

coefficients, we can model the jitter as an additive time-difference noise source (i.e. delay) at the output of each time-mode building block. For example, if the time-difference equation of a jitter-free building block is written as

$$\Delta T_{OUT}(n) = f(\Delta T_1(n), \Delta T_2(n)) \tag{4.24}$$

then the corresponding time-difference equation after the inclusion of clock jitter can be written as

$$\Delta T_{OUT}(n) = f(\Delta T_1(n), \Delta T_2(n)) + f_j(T_j(n)) \tag{4.25}$$

where $f_j(x)$ is a linear function.

4.4.4 Component Mismatch

For the time-mode building blocks to produce the desired operation, good matching is required between the capacitors and current mirror transistors. Any mismatch between these component will change the rate of increase of one of the comparator voltages. Consider the time adder circuit, where mismatch in the current mirror causes the current charging C_1 to change from I to $I + \Delta I$. In this case, Equation (3.7) would become

$$\frac{I + \Delta(I)}{C_1}(t_1 - t_{REF}) + V_{RESET} = \frac{I}{C_2}((t_{OUT} - t_{REF}) - (t_2 - t_{REF})) + V_{RESET} \tag{4.26}$$

Simplifying the above equation and writing it as a sampled-data system with discrete index n yields the time-difference equation

$$\Delta T_{OUT}(n) = \frac{I + \Delta(I)}{I}\Delta T_1(n) + \Delta T_2(n) \tag{4.27}$$

Therefore, any mismatch in the charging currents of the time adder, or any of the time-mode building blocks, due to mismatch in the current mirror transistors will manifest itself as a change in the scaling factor for one (or both) of the input time-differences. Similarly, if there is mismatch in the MOS capacitors, causing the capacitor C_1 to change from C to $C + \Delta C$, then the output time-difference equation would become

$$\Delta T_{OUT}(n) = \frac{C}{C + \Delta C}\Delta T_1(n) + \Delta T_2(n) \tag{4.28}$$

resulting in the same effect as the current mirror transistor mismatch. The effects of component mismatch can be minimized using common centroid layout techniques for the capacitors and current mirror transistors.

4.4.5 Effect of Circuit Noise on Decision Time

The decision (i.e. switching) time for the time-mode circuits used in this work can be described by the equation

$$\Delta T_O(n) = \frac{C}{I}V_T(n) \tag{4.29}$$

where $V_T(n)$ is the reference voltage of a comparator, and $T_O(n)$ is the charging time of a capacitor C on its positive terminal by the current I. Noise present in the circuit will cause the variations in the current level I, as well as changes to the threshold voltage $V_T(n)$. We can express these noises with two random variable terms inserted into the above expression as follows

$$\Delta T_O(n) = \frac{C}{I + i_n}(V_T(n) + v_n) \tag{4.30}$$

Assuming the two error terms are zero mean with variances, $\sigma_{i_n}^2$ and $\sigma_{v_n}^2$, together with the assumption that the mean value of the threshold level is midway between the supply level, i.e. $V_{DD}/2$, then the variance in the decision time following the method of Taylor series approximation [22] can be written as

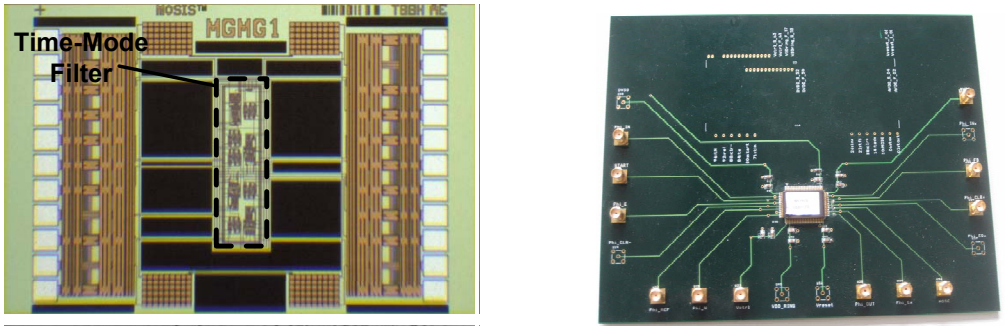
$$\sigma_{\Delta T_O(n)}^2 = \left(\frac{C}{I}\right)^2 \sigma_{v_n}^2 + \left(\frac{CV_{DD}}{2I^2}\right)^2 \sigma_{i_n}^2 \tag{4.31}$$

The effect of noise in the circuit can therefore be minimized through a large current bias I or a small capacitance C. While the manufacturing limits impose the smallest C that be used in the design, power requirements will limit the current level that is chosen.

4.4.6 Building Block Non-Linearities

When designing the building blocks of Chapter 3, the range of the comparator input voltage must be chosen such that the MOS equivalent capacitance and the current being produced by the mirror transistors are constant with respect to the ramping voltage. If either C or I become voltage dependent, then the increase of the voltage will no longer be linear, rather it will have a rate of change which is dependent on the current voltage at the node v_C. This is further complicated by the fact that generated current $I_1(t)$ and MOS capacitance $C_1(t)$ are actually a function of the node voltage $v_1(t)$.

Ensuring that the slope of the voltage is constant is crucial to the operation of the time-mode building blocks, and therefore, the operation of the filter. Moreover, attempting to implement the proposed time-mode circuits in fine-line CMOS processes will be susceptible to greater non-linearity in the charging process on account



(a) (b)

Figure 4–21: (a) Microphotograph of test die, (b) Chip mounted on PCB.

of the lower transistor output resistance. This will therefore limit the application of such circuits to low-resolution applications.

4.5 Low-Pass Filter IC

The second-order LPTF was implemented in a 1.8V, 0.18 μm CMOS process. A microphotograph of the manufactured test die is shown in Figure 4–21(a). The 5-MHz design occupies dimensions on-chip of 550 μm by 180 μm, for an area of 0.099 cm^2. The chip was mounted on a custom six-layer PCB, as shown in Figure 4–21(b) such that it could be interfaced with a Teradyne Automated Test Equipment.

Unfortunately, no experimental results were obtained from the test die. The authors attribute this to issues with the pins and packaging of the die. Instead, a summary of the simulated results associated with the test die is given in Table 4–1.

Table 4–1: Summary of simulated second-order low-pass filter performance.

Technology	0.18-μm CMOS
Supply Voltage	1.8 V
Dimensions	550-μm x 180-μm
Clock Frequency	5 MHz
Power Consumption	0.76 mW
Full Scale Input	250 mV
Peak SDR(@275 kHZ)	44.1-dB

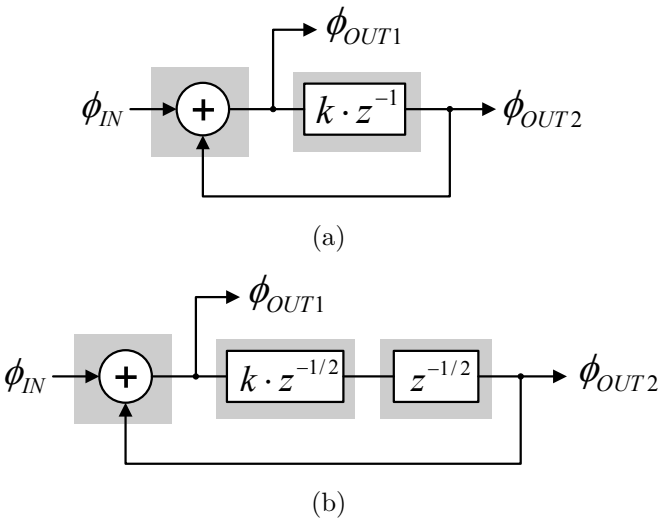


Figure 4–22: Discrete implementation of a damped time-mode integrator: (a) original block diagram, (b) modified block diagram.

4.6 Discrete Component Implementation

4.6.1 Damped Time-Mode Integrator

To illustrate how the time-mode building blocks may be combined to form larger time-mode systems, a damped time-mode integrator was constructed using discrete components and bench top testing equipment. The block diagram of a damped

integrator is equivalent to that of the time-mode integrator presented in Section 3.2.7 with a damping coefficient k added to the feedback path, as depicted in Figure 4–22(a). The z-domain transfer functions between the two output signals ϕ_{OUT1} and ϕ_{OUT2} to the input ϕ_{IN} can be obtained:

$$\frac{\Delta T_{OUT1}(z)}{\Delta T_{IN}(z)} = \frac{1}{1 - kz^{-1}} \tag{4.32}$$

$$\frac{\Delta T_{OUT2}(z)}{\Delta T_{IN}(z)} = \frac{kz^{-1}}{1 - kz^{-1}} \tag{4.33}$$

As discussed in the discrete implementation of a WDTS in Section 3.4.2, the authors were unable to obtain a half-delay element that could produce the required low-speed delay for the kz^{-1} block, and therefore, a modification to the block diagram was required. In the final block diagram of Figure 4–22(b), the feedback time-mode attenuator block was divided into two distinct time-mode functions: the first being a weighted half-delay time-mode attenuator, and a second which has a gain of unity and produces the second half-unit delay.

A photograph of the full time-mode damped integrator with $k = 1/2$ and $T = 50\mu s$ is shown in Figure 4–23. The same discrete implementation of the time adder described in Chapter 3 was used for this design. The half-delay attenuator circuit was implemented by removing the sinking (negative) branch of the discrete WDTS described in Section 3.4.2, as shown in Figure 4–24(a). The final half-unit delay was implemented using the same topology as the half-delay attenuator, except that in this case the reference ϕ_{REF} was replaced with $\overline{\phi_{REF}}$ and the two clearing signals ϕ_{CLR+} and ϕ_{CLR-} were swapped to account for the fact that the input time-mode information occurred during the negative half-cycle of the reference. Finally, k was set to unity for the final block. The final circuit schematic is shown in Figure 4–24(b).

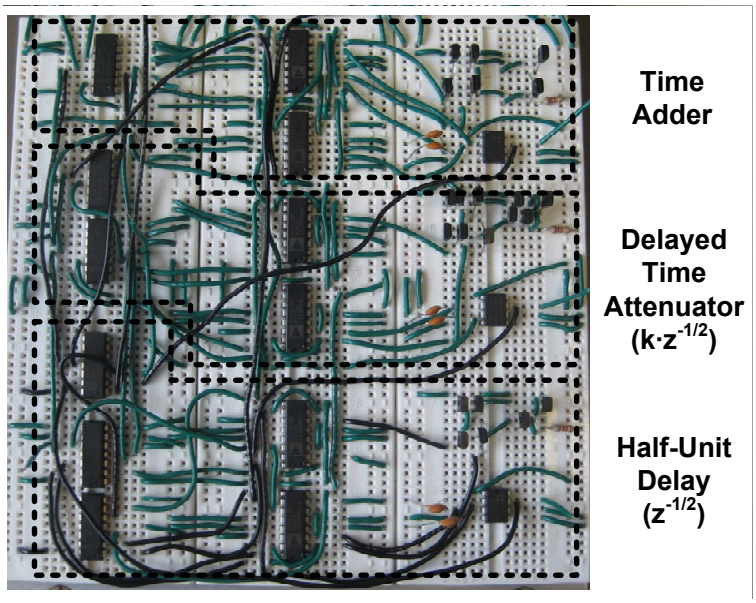


Figure 4–23: Photograph of discrete implementation of the time-mode damped integrator.

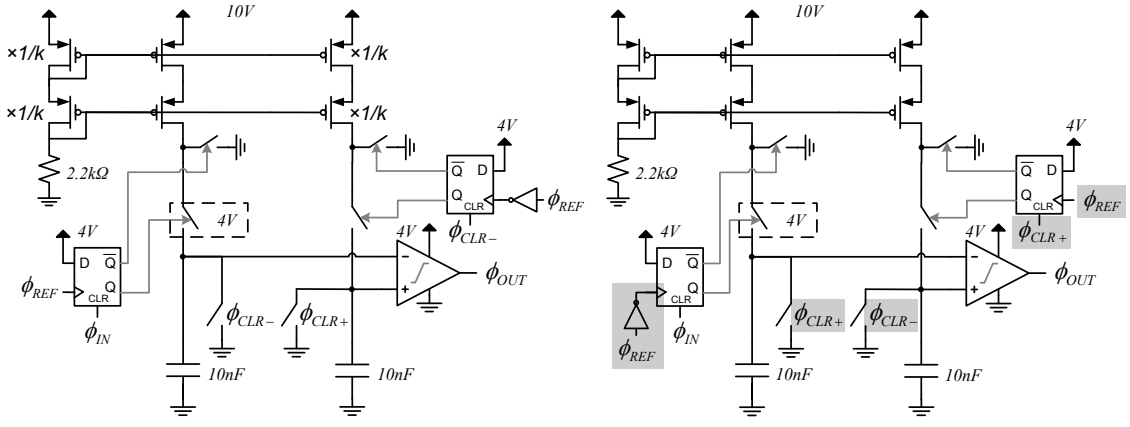
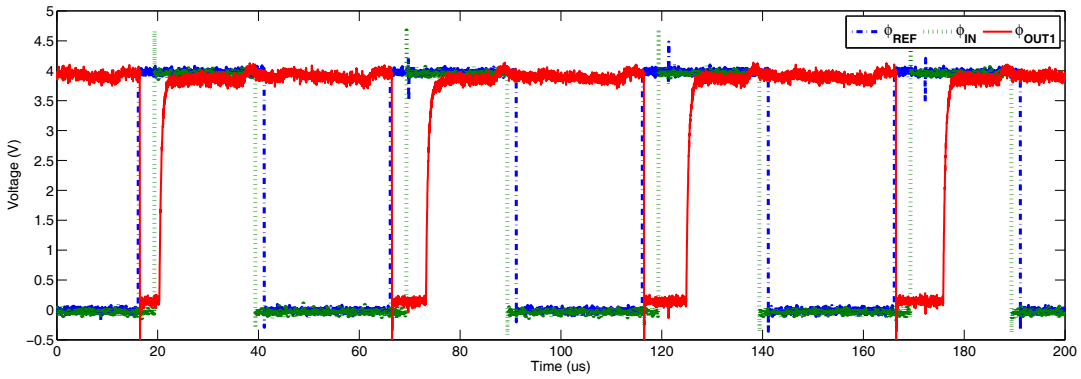
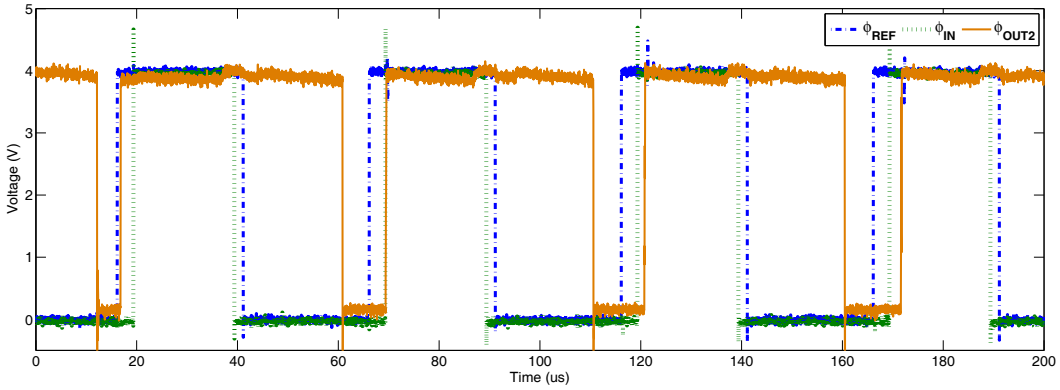


Figure 4–24: Discrete implementation of a (a) half-delay attenuator (b) half-unit delay.



(a)



(b)

Figure 4–25: Experimental results of the discrete implementation of the damped integrator: (a) ϕ_{OUT1} (b) ϕ_{OUT2}.

The circuit of Figure 4–23 was tested by applying a constant time-mode difference signal to the input. A section of the time domain experimental results are shown in Figures 4–25(a) and 4–25(b) for ϕ_{OUT1} and ϕ_{OUT2}, respectively. A graph showing the progression of the time-differences ΔT_{IN}, ΔT_{OUT1} and ΔT_{OUT2} as a function of the instance (or period number) n is given in Figure 4–26. As expected, the output time-differences ΔT_{OUT1} and ΔT_{OUT2} in the graph both experience a rise in the early portion, only to level off as the number of periods n increases.

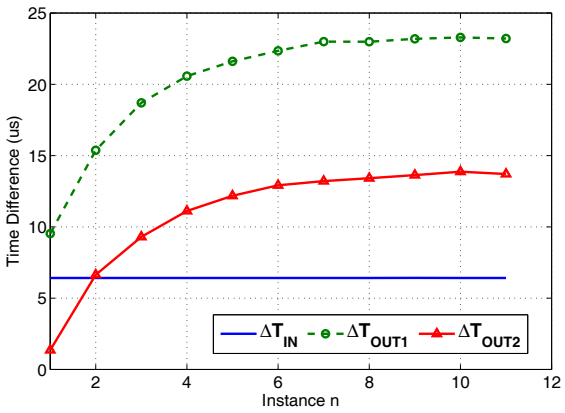


Figure 4–26: Progression of the damped integrator time-differences.

4.6.2 Second-Order Low-Pass Time-Mode IIR Filter

In order to implement a second order low-pass time-mode IIR filter using discrete components, some modification were required to the original system (reproduced in Figure 4–27(a)) and its parameters. For example, the period of the discrete system needed to be lowered to 20 kHz to account for the increased parasitics of the discrete components. The final block diagram shown in Figure 4–27(b) is composed of time-mode blocks that are achievable from discrete components, that is, a time adder, a WDTS, a three-input WDTA and two unit delay blocks.

As with the time-mode damped integrator, the WDTS is divided into two blocks. The first of these blocks produces the weighted difference with an added half-unit delay (Figure 3–31), followed by a second half-unit delay to re-reference the time-mode information with the clock's rising edge (Figure 4–24(b)). For the WDTS, the transistor scaling was set to $m_1 = 1$, $m_2 = 2$ and the capacitors values chosen were $C_A = 28nF$, $C_B = 12.5nF$ resulting in the weighting coefficients $k_1 = 0.446$ and $k_2 = 0.223$. The discrete implementation of a unit-delay block was built by cascading

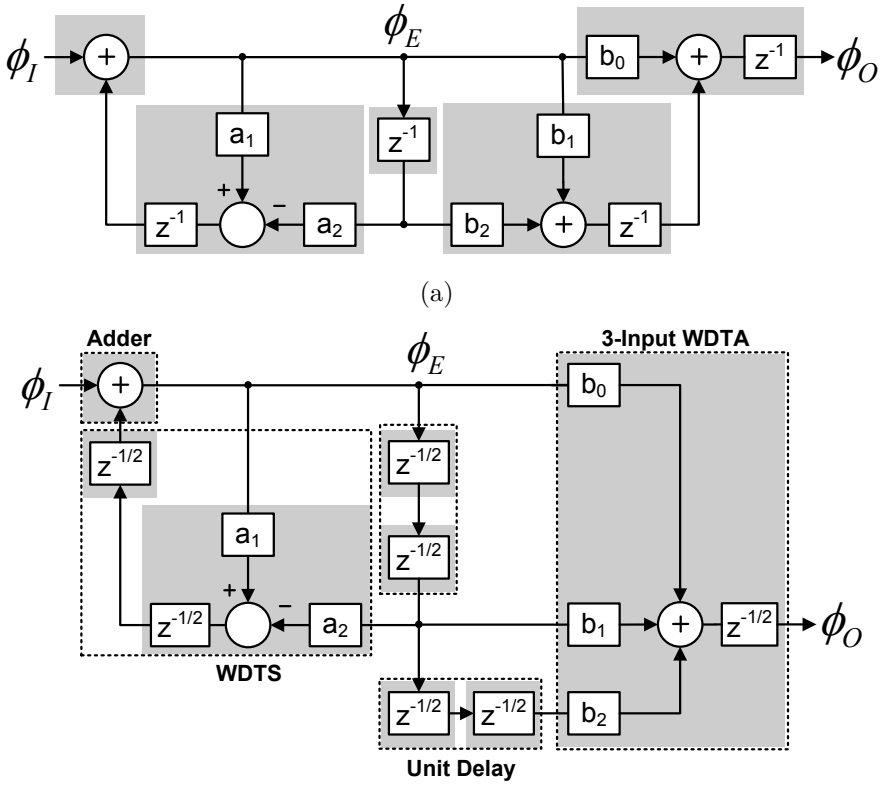


Figure 4–27: Discrete implementation of a second-order LPTF: (a) original block diagram, (b) modified block diagram.

a time-mode attenuator circuit (Figure 4–22(a)) with $k = 1$ and a half-unit delay block. The discrete implementation of a time adder circuit was presented earlier in Section 3.4.

The main difference between the two diagrams is the feedforward path. In the modified block diagram, the two WDTAs are replaced with a single three-input WDTA. The advantage of using a single cell is that is provides better matching between the feedforward coefficients (b_0,b_1,b_2) than the two-cell design. This improvement in performance becomes even more important in this implementation given the significantly larger distances between the discrete components versus the transistors

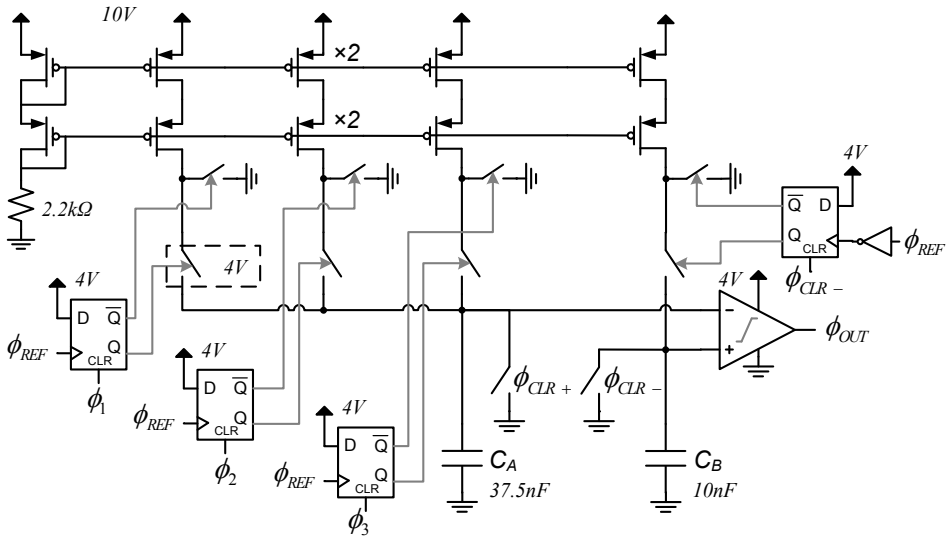


Figure 4–28: Discrete implementation of the three-input WDTA.

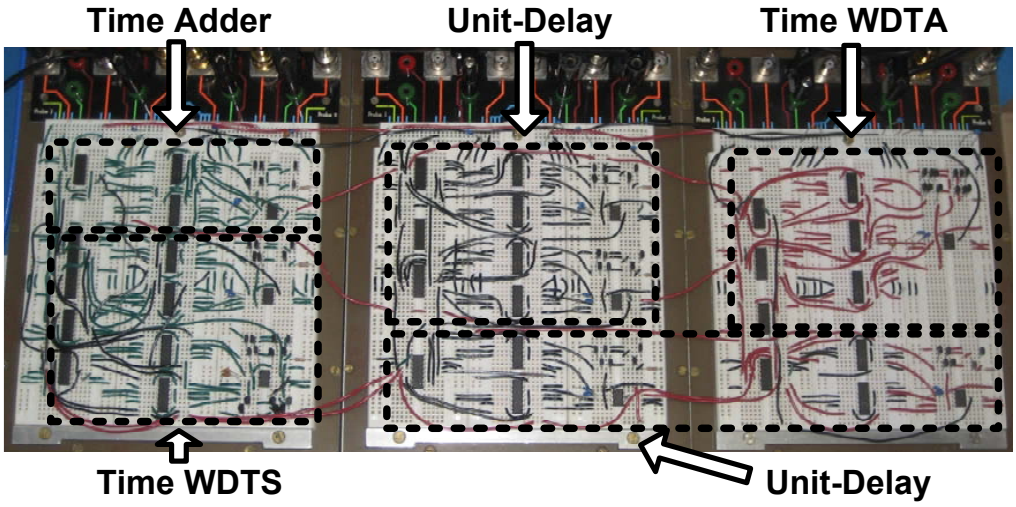


Figure 4–29: Photograph of discrete implementation of the second-order LPTF.

on the fabricated IC. Note also that this modification also requires a second unit delay element to produce the z^{-2} feedforward term. The schematic of the discrete implementation of a three-input WDTA is shown in Figure 4–28. A photograph of

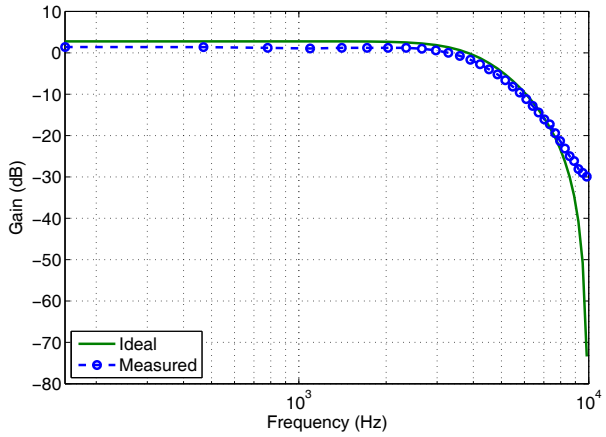
the full second-order LPTF is shown in Figure 4–29. Given the values of the weighting coefficients of the WDTS and three-input WDTA, the final expected transfer function for the system of Figure 4–27(b) is:

$$\frac{\Delta T_O(z)}{\Delta T_I(z)} = z^{-1/2}\frac{0.267 + 0.533z^{-1} + 0.267z^{-2}}{1 - 0.446z^{-1} + 0.223z^{-2}} \tag{4.34}$$

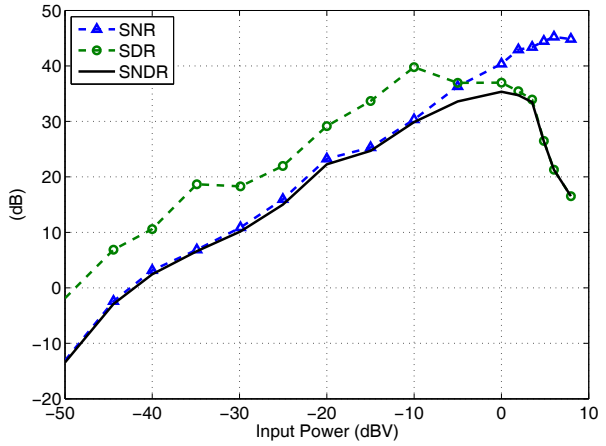
Note that the transfer function of Equation 4.34 differs from the original system (Figure 4–27(a)) by one half-delay. This is due to the fact that the final half-delay unit required for the three-input WDTA was omitted for the discrete implementation due to the lack of space on the breadboards. As was mentioned earlier in this Chapter, this modification affects only the phase and not the magnitude response of the system. Therefore in this case, the output time-mode signal ϕ_{OUT} is taken with respect to the falling edge of the reference ϕ_{REF}.

Rather than build a TVC at the front end of the filter, the input time-mode variable ϕ_I was applied directly to the time-mode filter to remove any non-idealities that would result from the voltage-to-time conversion process. To simulate the process of the VTC, where the rising edges of an input digital signal are delayed by an amount proportional to the input control voltage, the rising edges of a digital clock signal were phase modulated by a sampled-data sinusoid and the result was fed to the input ϕ_I in Figure 4–27(b). This input signal was created using two Agilent 33220A arbitrary waveform generators (AWGs), one of which generated the sine wave, and the other was used to generate the digital clock signal.

A third Agilent AWG and a HP 165-MHz 81170A dual-channel pulse-pattern generator were used to generate the reference ϕ_{REF} as well as the two clear signals ϕ_{CLR+} and ϕ_{CLR-}. The reference, input and output digital signals were sampled



(a)



(b)

Figure 4–30: (a) Frequency response and (b) dynamic range (1.09 kHz) of a LPTF implemented using discrete components.

every $t = 2ns$ (500 MHz) and captured using a LeCroy SDA 6000 oscilloscope. One hundred and twenty eight periods of the digital signals were captured. The information was then processed in MATLAB to determine the time of occurrence of the rising edges of the signals. Finally, the frequency response, SNR and SNDR of the circuit were calculated. A summary of the measured results is given in Table 4–2.

Table 4–2: Summary of performance of the second-order low-pass filter implemented with discrete components.

Analog Supply Voltage	10 V
Digital Supply Voltage	4 V
Clock Frequency	20 kHz
LeCroy Sampling Rate	500 MHz
Power Consumption	130 mW
Full Scale Input	1 V
Peak SNR(@1.09 kHZ)	45.2-dB
Peak SNDR(@1.09 kHZ)	35.3-dB

A comparison between the measured filter response and the ideal response calculated in MATLAB are shown in Fig.4–30(a). The mismatch between the measured results and the ideal filter response with respect to the DC gain and behaviour at high frequency is attributed the fact that the actual filter coefficients obtained are not exactly those that were designed for. One of the big consequence of implementing the circuit using discrete components on a breadboard is the inherent mismatch between the actual devices themselves, as well as the additional resistive and capacitive parasitics added by the positioning and routing on the board.

The measured dynamic range is is shown in Figure 4–30(b). The maximum SNR and maximum SNDR of the simulated circuit was found to be 45.2 and 35.3 dB respectively, which is significantly lower than the performance obtained through the extracted simulations earlier in this Chapter. While part of this decrease in performance is attributed to the expected increase in noise and distortion in measured versus simulated results, there is also an added noise-distortion component which is a consequence of building the system on multiple breadboards.

Some final observations can be noted for the discrete implementations of time-mode circuits. Firstly, due to the inability to easily obtain discrete delay cells, many more circuit elements are required to produce the additional half-unit blocks. The physical size of the system thus increases dramatically with the filter order N. One solution to this problem would be to decrease the period of the reference clock so that it coincides with the available delay unit values. A problem with this solution is that there is a limitation to how fast the discrete implementation can run before the system's parasitics begin to disrupt the operation of the circuit. Another important issue with discrete realizations result from the fact that there are only a finite number of devices (capacitors, transistors) with different component values available in discrete packages. Since the filter coefficients are determined by the capacitance ratio and the number of devices, this limits the number of filter transfer functions that can be implemented in this technology.

4.7 Higher-Order Filter Implementation

Extending the methodology presented in this chapter to the design of higher-order time-mode IIR filters is straightforward. As in the case of the second-order systems, one method for implementing higher-order designs is achieved by first re-organizing the direct form block diagram of the sampled-data filter, followed by a substitution of the time-mode building blocks into the structure. To illustrate how this can be realized for sampled-data IIR filters of any order N, consider the design of both a general third-order as well as a general fourth-order low-pass filter with

impulse responses:

$$\frac{\Delta T_O(z)}{\Delta T_I(z)} = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3}}{1 - a_1 z^{-1} + a_2 z^{-2} - a_3 z^{-3}} \tag{4.35}$$

$$\frac{\Delta T_O(z)}{\Delta T_I(z)} = \frac{b_0 + b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3} + b_4 z^{-4}}{1 - a_1 z^{-1} + a_2 z^{-2} - a_3 z^{-3} + a_4 z^{-4}} \tag{4.36}$$

To construct the higher-order LPTFs, the corresponding direct form block diagrams of the filters are first generated, shown in Figures 4–31(a) and 4–31(b) for the third and fourth-order circuits respectively. Similarly to the second-order case, these systems can not be implemented directly due to the laws of causality, and thus a reorganization of the block diagrams is first required.

The final topologies for the third and fourth-order LPTFs are shown in Figures 4–31(c) and 4–31(d) respectively. According to the first diagram, a third-order LPTF may be implemented by taking the second-order design (blocks are indicated by dotted boxes) and connecting two time adders, two TATNs and a unit-delay element (the new block ares highlighted in the diagram) to account for the extra branch in the block diagram. In the case of the fourth-order design, an additional two time adders, two WDTSs and two unit-delay elements are required. The implementations of the third-order and fourth-order block diagrams is easily generalized to block diagrams of any odd and even order. Essentially, this amounts to vertically cascading the required blocks (equivalent first or second-order sections) to obtain the desired order.

As discussed in the literature of filter design, the direct-form realization depicted in Figures 4–31(a) and 4–31(b) is extremely sensitivity to parameter quantization effects, and thus it is not used for practical applications. One alternative for obtaining

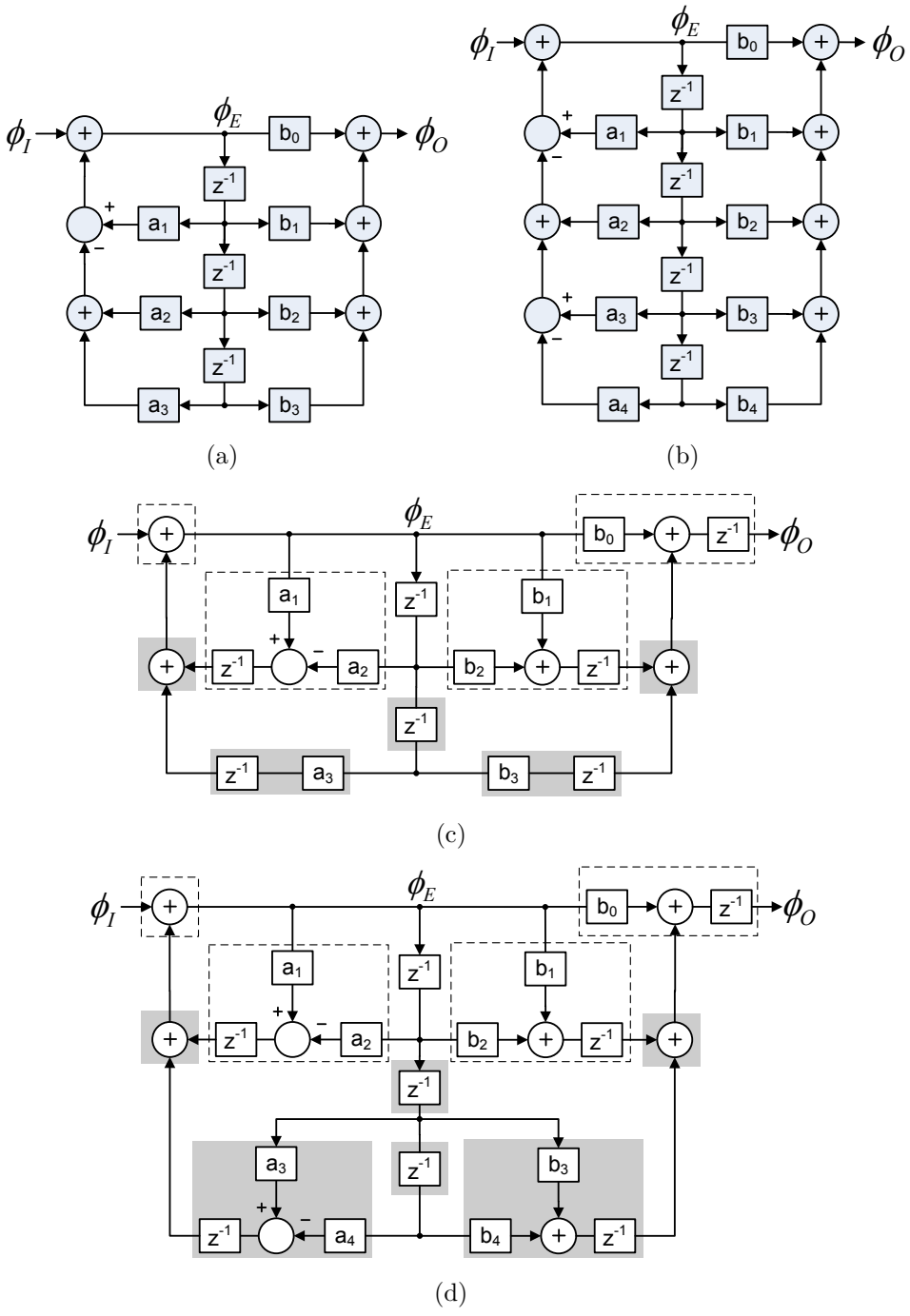


Figure 4–31: Direct form block diagram for (a) third-order and (b) fourth-order low-pass filters. The diagrams are rearranged to implement LPTFs or order three in (c) and four in (d).

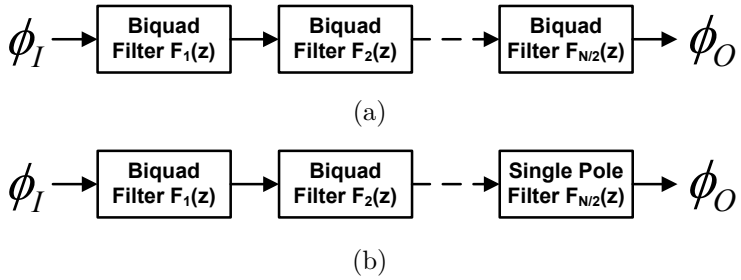


Figure 4–32: Cascade-of-biquad method for the design of time-mode filters: (a) even order and (b) odd order.

better sensitivity properties is to design a high-order filter as a cascade of second-order (i.e. biquad) sections [23]. To apply this method to TMSP, a number of second-order time-mode filter sections are cascaded (the design of which has been covered in detail) to obtain the desired even order N. To obtain an odd filter order, an extra single zero/pole filter is required. A single-pole system is easily realized using the same methodology described in the earlier sections. The design of even and odd time-mode filters using the cascade-of-biquads is illustrated in Figure 4–32.

A very popular methodology used for the design of higher-order sampled-data filters is the operational simulation of LC ladders [24]. The main advantage of these designs is they benefit from the lowest component sensitivity [25]. Unfortunately, the authors were unable to synthesize the time-mode filters described in this chapter using the LC ladder methodology due to the limited number of operations (described in Chapter 3) that can be implemented using our method. The study of LC ladder synthesis will thus be left for a topic of future research.

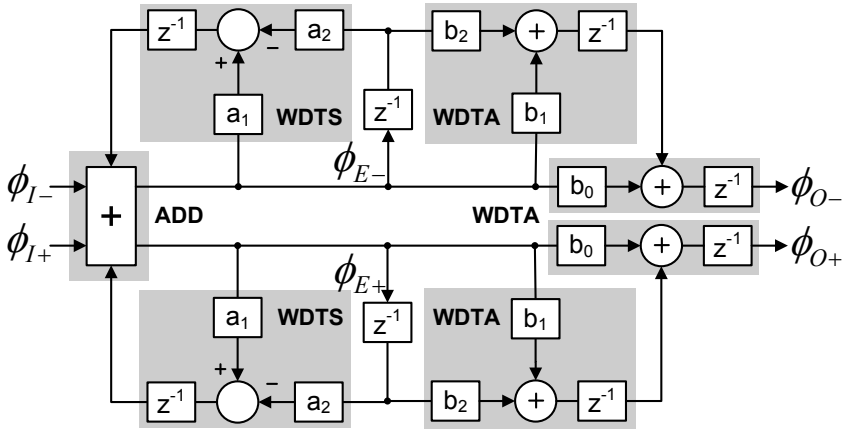


Figure 4–33: Differential implementation of a second-order LPTF.

4.8 Differential-Input Implementation

Ideally, a differential time-mode system would take two input phases ϕ_{I+} and ϕ_{I-} and produce two output signals ϕ_{O+} and ϕ_{O-} without the need for a reference ϕ_{REF}. In the context of this work, ideal differential structures are difficult, and to the author's knowledge, unrealizable using the methodology presented here due to the limitations resulting from the feedback action of these circuits. Instead, a quasi-differential design method is presented in which the reference ϕ_{REF} is used, but where the input and output time-difference are defined by the rising edges of ϕ_{I+} and ϕ_{I-}, and ϕ_{O+} and ϕ_{O-}, respectively.

To illustrate how the methodology presented in this work can be utilized to implement a quasi-differential time-mode systems, consider the single-ended second-order low-pass design presented earlier in Figure 4–9(b). To convert the system into a differential design, a second version of the same low-pass structure is included, and the two separate time-adders are replaced with the differential time-adder described in Section 3.2.9, as depicted in Figure 4–33. In this case, the time-mode blocks in the

path of the positive input signal ϕ_{I+} utilize the *rising* edge of ϕ_{REF} as the reference edge while the blocks in the path of the negative input signal ϕ_{I-} utilize the *falling* edge of ϕ_{REF}. The differential output of the system $\Delta T_{O-diff}(n)$ is taken between the rising edge of $\phi_{O-}(n)$ and the rising edge of $\phi_{O+}(n)$.

The potential advantages of implementing differential time-mode systems are similar to those of traditional sampled-data systems. Firstly, and most importantly, the differential structure will reject any common-mode noise in the circuit. Secondly, the even-order harmonics components of the output caused by non-linear distortion in the circuit are suppressed, resulting in a larger SDR and thus larger SNDR.

The disadvantages associated with the circuit of Figure 4–9(b) is the increased silicon area required for the design, as well as the additional power that is consumed. Also, as mentioned earlier, the quasi-differential implementation still requires a reference clock, unlike traditional differential structures, where the reference is omitted.

4.9 Summary

In this Chapter, the methodology for the design of sampled-data filters using time-mode signal processing was presented. This was achieved by modifying the direct form block diagrams of the filters and substituting in the basic mathematical building blocks described earlier in this work. Second-order low-pass, band-reject, and high-pass filter structures were implemented in theory, then modelled in MATLAB and simulated using Spectre. A second order low-pass time-mode IIR filter was also implemented using discrete components.

A second-order LPTF was fabricated in a 0.18-μm CMOS process, but unfortunately, the authors were unable to obtain experimental results due to problems

in the chip. Nonetheless, the theory presented in this Chapter has the potential for high speed sampled-data filtering due to the fact that the area and voltage range required for the operation of these circuits is inversely proportional to the frequency of operation.

Chapter 5

Conclusion

5.1 Thesis Summary

This thesis investigated the use of time-mode signal processing in the design of sampled-data infinite impulse response filters. This was achieved by first converting the sampled-data voltage into a time-difference variable and then processing it using a time-mode filter before finally translating the output time-difference back into the voltage domain.

To implement a filter in the time-mode, the basic mathematical operations required for TMSP were identified. A set of basic building blocks were designed, based on the Direct Voltage-Controlled VCDU, that implemented the operations of addition, weighted delayed subtraction, weighted delayed addition, amplification, attenuation, and integration in the time-mode. Finally, by modifying the direct form block diagrams of sampled-data structures, the basic time-mode building blocks were arranged to form second-order IIR time-mode filters. It was shown that the TMSP methodology could be used to implement the four basic filter functions: low-pass,

band-reject, high-pass and band-pass. Furthermore, a method to generalize the design of these circuits to filters of higher-order as well as an extension to differential implementations was described.

A second-order low-pass filter was fabricated in a 0.18-μm CMOS process, but the author's were unsuccessful in extracting any experimental results. Instead, this work presents the simulation results of three time-mode filter circuits, as well as the experimental results of the discrete component implementation of a second-order low-pass time-mode IIR filter and damped integrator. The measured results showed the ability to perform low-pass filtering, offering maximum SNR and SNDR of 45.2 and 35.3 dB, respectively. While limited to a discrete implementation, the results demonstrate that time-mode signal processing can be used to implement arbitrary sampled-data transfer functions. Nonetheless, more research is required to verify the feasibility of time-mode circuits for fully monolithic implementation.

5.2 Future Works

The design of analog signal processing circuits, specifically sampled-data filters, has advanced over time in an effort to design circuits which are smaller, provide better performance, and are more efficient (i.e. consume less power). Since this work represents the first attempt into the design of time-mode IIR filters, there is still plenty of development that can follow. Some examples of studies that will advance and extend this work are:

1) Fabricating a new test chip. By fabricating a test chip and obtaining experimental results, the methodology and simulation results for time-mode filters can be verified.

2) Designing a full sampled-data time-mode filter. As mentioned earlier, this work assumed perfect and ideal voltage-to-time and time-to voltage conversion. By designing and implementing the input voltage-to-time and time-to-voltage converters, the performance of the full system can be measured.

3) Extending the method of the operational simulation of LC ladders to time-mode filtering. The LC ladder method is very popular in the design of higher-order sampled-data filters due to the fact that the synthesized designs benefit from the lowest component sensitivity. By developing an LC ladder method based on the time-mode building blocks described above would result in better designs that could potentially be manufactured in an industrial setting.

4) Developing purely time-based building blocks. The basic mathematical circuits developed in this work require the input time-difference to be converted into a voltage through the charging of a capacitor by a constant current. As a result, the time-difference variable is susceptible to all the non-idealities (i.e. voltage domain noise and non-linearities) associated with the voltage and sampled-data domains. One of the main goals in TMSP is to design systems that are purely digital in nature so that they may take full advantage of digital technologies.

Appendix A

Transistor Dimensions and Component Values

In this appendix, all the transistor sizes, component values and bias conditions required to reproduce the second order low-pass time-mode IIR filter are presented. The values given were those simulated using Spectre. Note that all logic gates and flip-flops were taken from a 0.18-μm CMOS standard cell library.

A.1 Time-Mode Adder

The time-mode adder that was described in Section 3.2.2 is reproduced in Figure A–1. The transistor dimensions, component values and bias conditions are summarized in Table A–1. The comparator schematic information will be presented later in this appendix.

A.2 Time-Mode Weighted Delayed Subtractor

The time-mode weighted delayed subtractor (WDTS) that was described in Section 3.2.4 is reproduced in Figure A–2. The transistor dimensions, component values and bias conditions used in design of the the second-order low-pass time-mode filter (LPTF) IC are summarized in Table A–2.

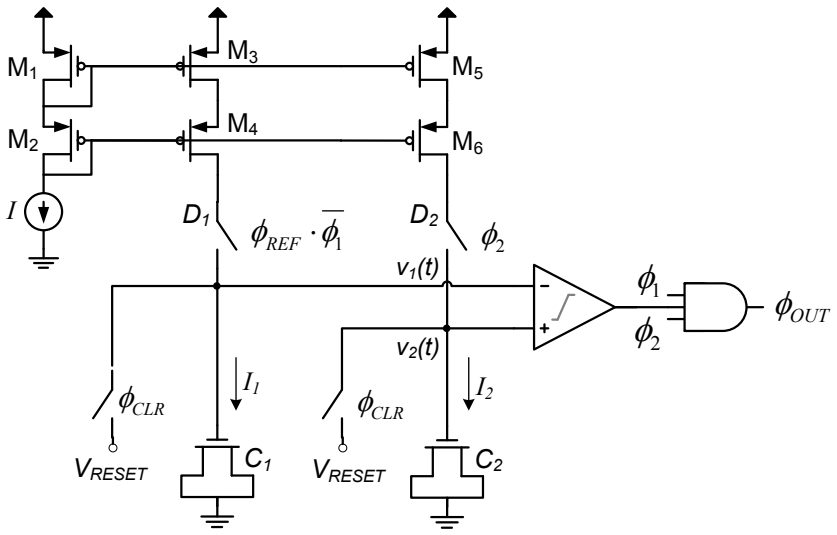


Figure A–1: Transistor schematic of the time-mode adder.

Table A–1: Component values and bias conditions for time-mode adder.

Component	**Value**
M_1, M_2	W/L = 1 μm/0.25 μm
M_3, M_4, M_5, M_6	W/L = 2 μm/0.25 μm
C_1, C_2	W/L = 50 μm/1 μm
I	3 μA
V_{DD}	1.8 V
V_{RESET}	0.6 V

A.3 Time-Mode Weighted Delayed Adder

The time-mode weighted delayed adder (WDTA) that was described in Section 3.2.5 is reproduced in Figure A–3. Two of such circuits were required to implement the second-order low-pass time-mode filter IC. The transistor dimensions, component values and bias conditions used in the first WDTA ($k_1 = 0.08, k_2 = 0.16$) and second WDTA ($k_1 = 0.08, k_2 = 1$) are summarized in Table A–3.

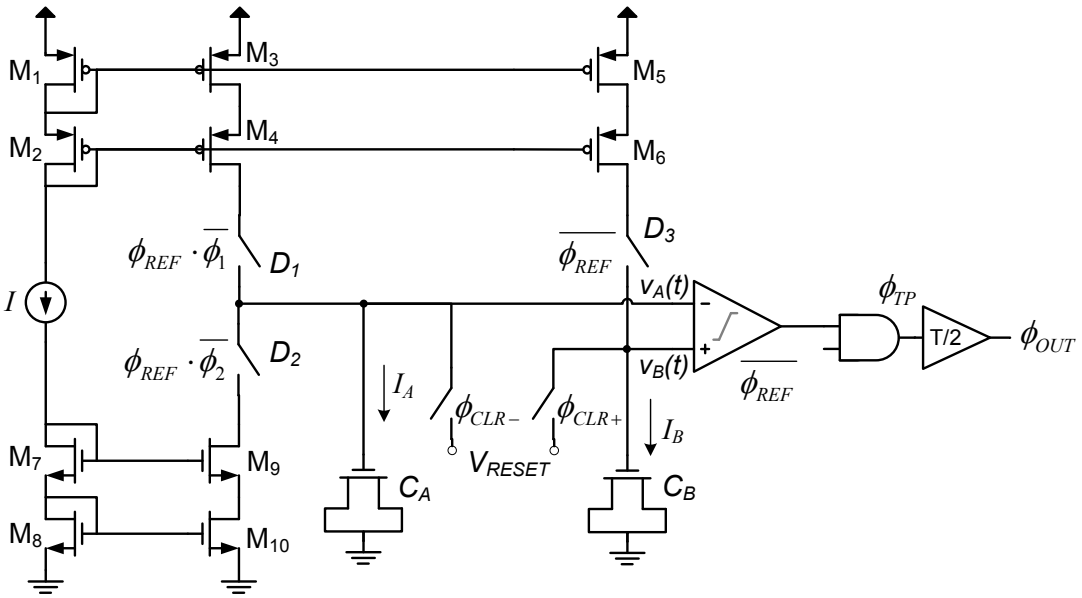


Figure A–2: Transistor schematic of the time-mode WDTS.

Table A–2: Component values and bias conditions for the time-mode WDTS used in design of the the second-order LPTF IC.

Component	**Value**
M_1, M_2, M_9, M_10	W/L = 0.6 μm/0.25 μm
M_3, M_4	W/L = 1.4 μm/0.25 μm
M_5, M_6, M_7, M_8	W/L = 1.2 μm/0.25 μm
C_A, C_B	W/L = 48 μm/2 μm
I	2.1 μA
V_{DD}	1.8 V
V_{RESET}	0.6 V

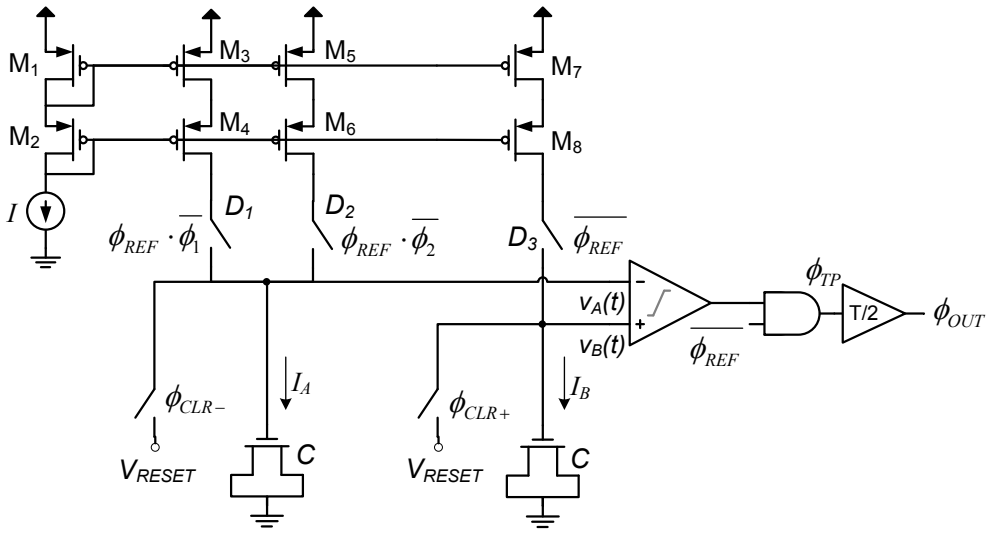


Figure A–3: Transistor schematic of the time-mode WDTA.

Table A–3: Component values and bias conditions for the two time-mode WDTAs used in design of the the second-order LPTF IC.

Component	**Value** ($k_1 = 0.08, k_2 = 0.16$)	**Value** ($k_1 = 0.08, k_2 = 1$)
M_1, M_2	W/L = 12 μm/0.25 μm	
M_3, M_4	W/L = 2 μm/0.25 μm	W/L = 1.2 μm/0.25 μm
M_5, M_6	W/L = 1.1 μm/0.25 μm	W/L = 11 μm/0.25 μm
M_7, M_8	W/L = 12 μm/0.25 μm	W/L = 11 μm/0.25 μm
C	W/L = 48 μm/2 μm	
I	11.7 μA	
V_{DD}	1.8 V	
V_{RESET}	0.6 V	

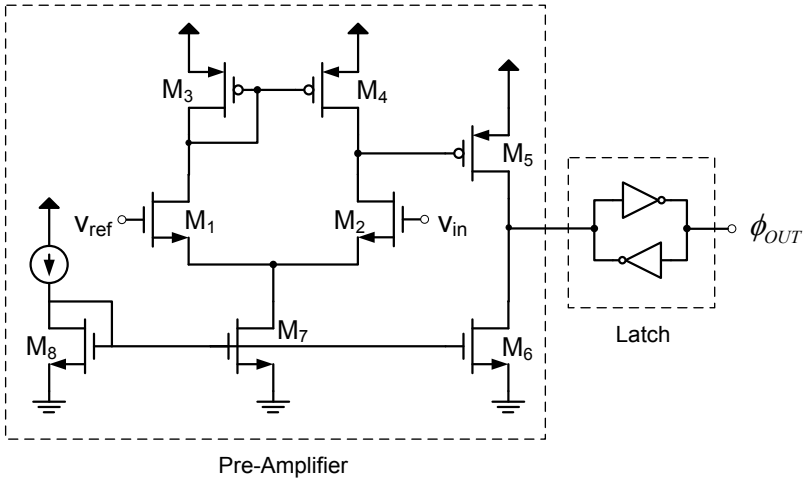


Figure A–4: Transistor schematic of the asynchronous voltage comparator.

Table A–4: Component values and bias conditions for the voltage comparator.

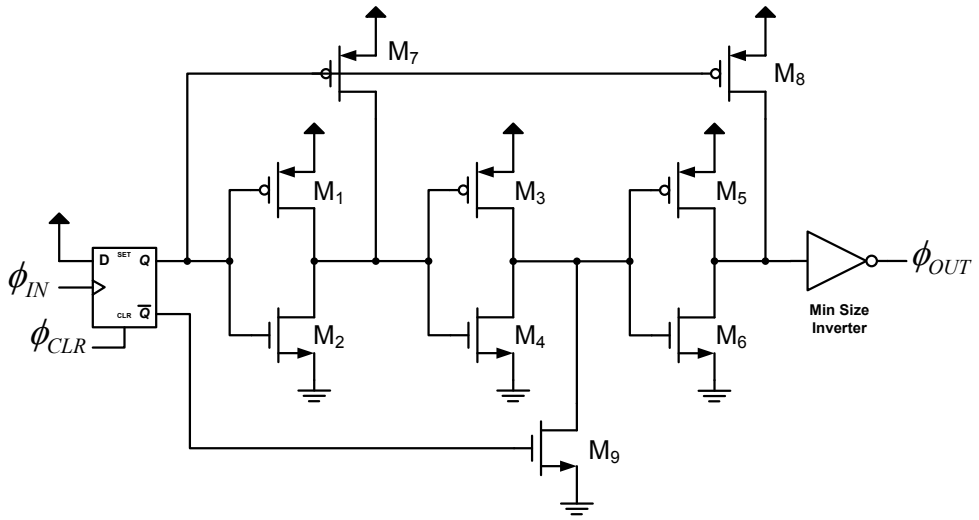
Component	**Value**
M_1, M_2, M_3, M_4	W/L = 0.6 μm/0.5 μm
M_5, M_6, M_7, M_8	W/L = 1 μm/0.5 μm
I	23.4 μA
V_{DD}	1.8 V

A.4 Voltage Comparator

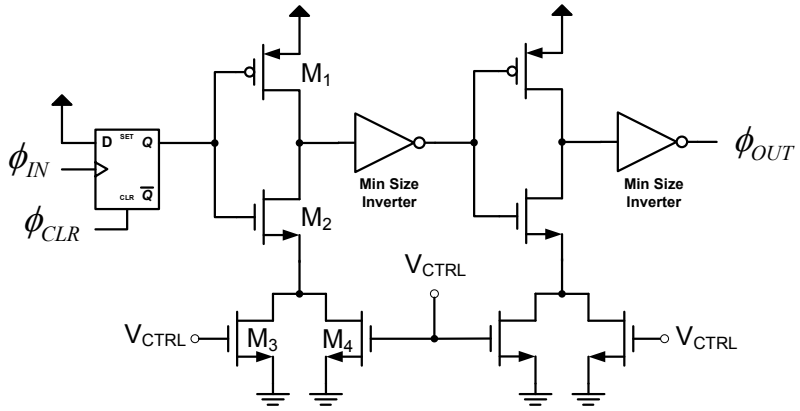
The voltage comparator that was described in Section 3.3.1 is reproduced in Figure A–4. The transistor dimensions, component values and bias conditions are summarized in Table A–4.

A.5 Unit and Half-Unit Delay Circuit

As mentioned at the beginning of the appendix, the unit delay cell is implemented by cascading two half-unit delay cells. This section will thus present the values associated with the half-unit delay cell, specifically, the coarse and the fine



(a)



(b)

Figure A–5: Transistors schematic for (a) coarse delay block, (b) fine delay block.

delay units that were presented in Section 3.3.2. The schematic for these two circuits are reproduced in Figure A–5. The transistor dimensions, component values and bias conditions for the coarse and fine delay units are summarized in Tables A–5 and A–6, respectively.

Table A–5: Component values and bias conditions for the coarse delay unit.

Component	**Value**
M_1, M_6	W/L = 1.8 μm/7 μm
M_2	W/L = 0.6 μm/7 μm
M_3	W/L = 2.7 μm/7 μm
M_4	W/L = 1.2 μm/7 μm
M_5	W/L = 3.6 μm/7 μm
M_7, M_8	W/L = 1.8 μm/0.18 μm
M_9	W/L = 0.6 μm/0.18 μm
V_{DD}	1.8 V

Table A–6: Component values and bias conditions for the fine delay unit.

Component	**Value**
M_1	W/L = 4 μm/1 μm
M_2	W/L = 2 μm/1 μm
M_3, M_4	W/L = 1 μm/1 μm
V_{DD}	1.8 V

References

[1] A. J. Annema, B. Nauta, R. V. Langevelde, and H. Tuinhout, "Analog circuits in ultra-deep-submicron cmos," *IEEE Journal of Solid-State Circuits*, vol. 40, pp. 132–143, January 2005.

[2] C. N. Chuang and S. I. Liu, "A 20-mhz to 3-ghz wide-range multiphase delay-locked loop," in *IEEE Trans. Circuits and Systems Part II: Express Briefs*, pp. 850–854, IEEE, November 2009.

[3] M. Safi-Harb and G. W. Roberts, "70-ghz effective sampling time-base on-chip oscilloscope in cmos," *IEEE Journal of Solid-State Circuits*, vol. 42, pp. 1743–1757, August 2007.

[4] N. R. Mahapatra, S. V. Garimella, and A. Tareen, "An empirical and analytical comparison of delay elements and a new delay element design," in *Proc. IEEE Computer Society Workshop on VLSI*, pp. 81–86, IEEE, April 2000.

[5] K. Gyudong, M. Kim, B. Chang, and W. Kim, "A low-voltage, low-power cmos delay element," *IEEE Journal of Solid-State Circuits*, vol. 31, pp. 966–971, July 1996.

[6] H. Pekau, A. Yousif, and J. W. Haslett, "A cmos integrated linear voltage-to-pulse-delay-time converter for time based analog-to-digital converters," in *Proc. IEEE Symposium on Circuits and Systems*, pp. 2373–2376, IEEE, May 2006.

[7] M. Maymandi-Nejad and M. Sachdev, "A monotonic digitally controlled delay element," *IEEE Journal of Solid-State Circuits*, vol. 40, pp. 2212–2219, November 2005.

[8] J. Yang, C. Chao, and S. Lin, "Tunable delay element for low power vlsi circuit design," in *TENCON 2006, IEEE Regional 10 Conference*, pp. 1–4, IEEE, November 2006.

[9] A. M. Abas, A. Bystrov, D. J. Kinniment, O. V. Maevsky, G. Russell, and A. V. Yakolev, "Time difference amplifier," *Electronic Letters*, vol. 38, pp. 1437–1438, November 2002.

[10] M. Oulmane and G. W. Roberts, "A cmos time amplifier for femto-second resolution timing," in *Proc. IEEE International Symposium on Circuits and Systems*, pp. 509–512, IEEE, May 2004.

[11] C. S. Taillefer and G. W. Roberts, "Process-insensitive modulated-clock voltage comparator," in *Proc. IEEE International Symposium on Circuits and Systems*, pp. 3910–3913, IEEE, May 2006.

[12] A. E. Stevens, R. P. V. Berg, J. V. D. Spiegel, and H. H. Williams, "A time-to-voltage converter and analog memory for colliding beam detectors," *IEEE Journal of Solid-State Circuits*, vol. 24, pp. 1748–1752, December 1989.

[13] E. Risanen-Ruotsalainen, T. Rahkonen, and J. Kostamovaara, "A time digitizer with interpolation based on time-to-voltage conversion," in *Proc. IEEE 40th Midwest Symposium on Circuits and Systems*, pp. 197–200, IEEE, August 1997.

[14] R. L. Sumner, "Apparatus and method for measuring time intervals with very high resolution," October 24 2000. U.S. Patent 6 137 749.

[15] C. S. Taillefer and G. W. Roberts, "Delta-sigma a/d conversion via time-mode signal processing," *IEEE Transactions on Circuits and Systems*, vol. 56, pp. 1908–1920, September 2009.

[16] P. Dudek, S. Szczepanski, and J. V. Hatfield, "A high-resoltion cmos time-to-digital converter utilizing a vernier delay line," *IEEE Journal of Solid-State Circuits*, vol. 35, pp. 240–247, February 2000.

[17] P. Chen, C. C. Chen, and Y. S. Chen, "A low-cost low-power cmos time-to-digital converter based on pulse stretching," *IEEE Transactions on Nuclear Science*, vol. 53, pp. 2215–2220, August 2006.

[18] V. Gutnik and A., "On-chip picosecond time measurement," in *Symposium on VLSI*, pp. 52–53, IEEE, June 2000.

[19] P. Chen and S. L. Liu, "A cyclic cmos time-to-digital converter with deep sub-nanosecond resolution," in *Proc. IEEE Custom Integrated Circuits Conference*, pp. 605–608, IEEE, May 1999.

[20] V. Ravinuthula, V. Garg, J. G. Harris, and J. A. B. Fortes, "Time-mode circuits for analog computation," *International Journal of Circuit Theory and Applications*, vol. 37, pp. 631–659, June 2009.

[21] V. Ravinuthula and J. G. Harris, "Time-base arithmetic using step functions," in *Proc. IEEE Symposium on Circuits and Systems*, pp. 305–308, IEEE, May 2004.

[22] W. Hines and D. Montgomery, *Probability and Statistics in Engineering and Management Science*. New York: John Wiley & Sons, 3 ed., 1990.

[23] J. G. Proakis and D. G. Manolakis, *Digital Signal Processing Principles, Algorithms, and Applications*. Upper Saddle River, NJ: Prentice-Hall International, 4 ed., 2007.

[24] P. O. Brackett and A. S. Sedra, *Filter Theory and Design: Active and Passive*. Forest Grove, OR: Matrix Publishers, 4 ed., 1978.

[25] M. L. Blostein, "Sensitivity analysis of parasitic effects in resistance terminated lc two-ports," *IEEE Transactions on Circuits Theory*, vol. 14, pp. 21–25, March 1967.