An Architecture Study of Control-Bounded ADCs for Multi-Channel Receiver Systems

Fredrik Feyling

December 2020

Abstract

Control-bounded conversion is a conceptually new approach to A/D conversion, that extends the possibilities for what an A/D converter could be. Of special interest in this thesis, is how a control-bounded ADC could be used advantageously in receiver systems with multiple input channels. The underlying motivation is to utilize prior information about the input signal to achieve increased performance. We argue that, in a multi-channel receiver system, placing several independent, general purpose ADCs, side-by-side, is not likely the most efficient way of capturing the incoming information. The control-bounded conversion principle enables a new approach to the design of a multi-channel receiver system.

The main contribution of this thesis is a proposed ADC architecture, which allows for a more power efficient implementation of the previously introduced Hadamard ADC. This architecture is evaluated on an architectural level and potential implementation challenges are discussed. A simulation verifies that this architecture dramatically reduces harmonics in the spectrum under the presence of component mismatch. Another interesting feature of this architecture is the ability to utilize an uneven energy distribution between the different input channels. For a receiver with e.g. 1000 channels, this would give a direct SNR increasement of up to 30dB. The achievable SNR gain will depend on the application.

In addition to the proposed architecture, a simpler structure based on a chain of integrators is also considered. Some of the features that is presented together with the proposed architecture, could also be applied to this simpler system. In consequence, the thesis presents two very different architectures that both allow for new ways of utilizing the properties of a multi-channel receiver system. Some associated design challenges for each of them are also discussed. We hope that this thesis provides a useful background for choosing an appropriate control-bounded ADC architecture, considering the application at hand.

Contents

List of Symbols

Matrices and Vectors

a	a scalar value
$\boldsymbol{a}$	a column vector $(a_0 \ \cdots \ a_{N-1})^\mathsf{T} \in \mathbb{R}^N$
$\boldsymbol{A}$	a matrix $\begin{pmatrix} a_{00} & \cdots & a_{0(N-1)} \\ \vdots & \ddots & \vdots \\ a_{(M-1)0} & \cdots & a_{(M-1)(N-1)} \end{pmatrix} \in R^{M \times N}$
$\boldsymbol{A}_{ij}$	a sub-matrix, that is part of a block matrix, e.g. $\boldsymbol{A} = \begin{bmatrix} \boldsymbol{A}_{00} & \boldsymbol{A}_{01} \\ \boldsymbol{A}_{10} & \boldsymbol{A}_{11} \end{bmatrix}$
$\mathbf{0}_N$	an all-zero column vector of length N
$\mathbf{0}_{M \times N}$	an M-by-N all-zero matrix
$\mathbf{1}_N$	a column vector of length N with all elements 1
$\mathbf{1}_{M \times N}$	an M-by-N matrix with all elements 1
$\boldsymbol{I}_N$	an N-by-N identity matrix
$\boldsymbol{H}_2$	second order Hadamard matrix $\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$
$\boldsymbol{H}_N$	Hadamard matrix of order N defined by $\boldsymbol{H}_2 \otimes \boldsymbol{H}_{N/2}$
$\boldsymbol{H}'_N$	reduced Hadamard matrix $\begin{bmatrix} \boldsymbol{H}_{N/2} & \mathbf{0}_{N/2 \times N/2} \\ \mathbf{0}_{N/2 \times N/2} & \boldsymbol{H}_{N/2} \end{bmatrix}$
$\otimes$	Kronecker product
$()^\mathsf{T}$	transpose
$()^\mathsf{H}$	Hermetian transpose
$\lvert a \rvert$	absolute value
$\lvert\lvert \boldsymbol{b} \rvert\rvert_p$	p-norm $(\Sigma_i \lvert b_i \rvert^p)^{1/p}$
$\lvert\lvert \boldsymbol{c} \rvert\rvert_\infty$	max norm, equivalent to $\max(\lvert c_0 \rvert, \lvert c_1 \rvert, \cdots \lvert c_{N-1} \rvert)$

Sets

$\mathbb{R}$	the real numbers
$\mathbb{C}$	the complex numbers

Miscellaneous

$\dot{\boldsymbol{x}}$	elementwise time derivative $\frac{d}{dt}\boldsymbol{x}(t)$

Control-Bounded Conversion

L	input signal dimension
N_ℓ	system order corresponding to each channel
N	total system order LN_ℓ
κ	control gain
β	integrator gain
$\boldsymbol{A}$	system matrix
$\boldsymbol{B}$	input matrix
$\boldsymbol{C}$	signal observation (output) matrix
$\boldsymbol{\Gamma}$	control input matrix
$\tilde{\boldsymbol{\Gamma}}$	control observation matrix
$\boldsymbol{u}(t)$	input signal
$\hat{\boldsymbol{u}}(t)$	estimated/reconstructed input signal
$\tilde{\boldsymbol{u}}(t)$	normalized, Hadamard transformed input signal
$\boldsymbol{x}(t)$	state vector
$\boldsymbol{s}[k]$	control signal
$\boldsymbol{s}(t)$	control contribution
$\tilde{\boldsymbol{s}}(t)$	control observation
$\boldsymbol{y}(t)$	signal observation
$\breve{\boldsymbol{y}}(t)$	fictional, "open-loop", signal observation
$\boldsymbol{q}(t)$	the control signal, seen at the output of the analog system
$\boldsymbol{G}(\omega)$	analog transfer function (ATF) matrix
$\boldsymbol{H}(\omega)$	noise transfer function (NTF) matrix
$\boldsymbol{G}(\omega)\boldsymbol{H}(\omega)$	signal transfer function (STF) matrix

Acronyms

ADC	analog to digital converter
AS	analog system
ATF	analog transfer function
DAC	digital to analog converter
DC	digital control
DE	digital estimator
LNA	low noise amplifier
MIMO	multiple input, multiple output
NTF	noise transfer function
NRZ	non-return to zero
OTA	operational transconductance amplifier
PSD	power spectral density
SAP	sub-aperture processor
STF	signal transfer function

Chapter 1

Introduction

Receiver systems with multiple inputs channels are becoming increasingly popular in a variety of applications. Multiple-input, multiple-output (MIMO) systems are being utilized in both radio communication, radar and acoustics. Medical ultrasound systems, in particular those designed for 3D imaging, might contain several thousand transducers in the probe. Common to all multi-input applications is the need for digitalizing a huge amount of analog input signals.

In most of the mentioned multi-input applications, this is done using general purpose ADCs, converting a single analog signal into a single digital bit stream. Depending on the application and the number of input channels, having one dedicated ADC per channel might not be feasible. For instance, in medical ultrasonic 3D imaging, a technique called sub-aperture processor (SAP) beamforming is used to limit the number of ADCs necessary in the system.

A system with tens, hundreds or even thousands of input channels motivates a rethinking of receiver system architecture. A structure consisting of a large number of general purpose ADCs, operating independently of each other, might not be the most efficient way of capturing the information from several analog input channels. If there exist some kind of prior knowledge about the input signal, this information could potentially be utilized for increased performance. Relevant prior knowledge could be the correlation between different channels or some statistical properties of an individual channel. Any prior knowledge will in principle imply that a general purpose ADC is redundant.

Utilizing this information is however not trivial. Interestingly, a newly developed concept called *control-bounded conversion* enables novel ADC architectures that might address this issue in new and interesting ways.

1.1 Control-Bounded Conversion

Over the last few years, a group at the university ETH Zürich has developed a conceptually new approach to A/D conversion called control-bounded conversion [1]. The main advantage of this approach is that it opens a new and wider design space, enabling ADC architectures that have previously been unimaginable. The latest contribution to control-bounded conversion is the doctoral thesis of Hampus Malmberg [2]. This thesis explores the design space associated with these converters, and proposes several different architectures suited for various applications.

One particularly interesting architecture is the so called *Hadamard ADC*, which provides a new and interesting way of combining multiple input channels. Combining multiple input channels in one ADC enables the mentioned utilization of statistical information, yielding a potential performance increasement. The proposed architecture is however only evaluated on a theoretical level, and no optimized circuit implementation is reported so far.

1.2 The Scope of the Thesis

This thesis is an architecture study of control-bounded ADCs for multi-channel receiver applications. The presented work forms the background for a future circuit implementation, optimized for medical ultrasound imaging. The goal of the thesis is to study control-bounded ADCs for multi-channel systems in general, and the results will hopefully be relevant for other applications as well.

As the medical ultrasound application is of special interest, the specifications listed in table 1.1 is kept in mind during the process. The specifications has been considered when choosing the frequency range for simulations, but is otherwise not limiting the relevance of the work.

Table 1.1: System Specifications for Medical Ultrasound Applications

Parameter	Symbol	Value	Comment
Carrier Frequency	f_{ca}	5 MHz	
Bandwidth	$\mathcal{B}$	5 MHz	$2.5 - 7.5$MHz
Signal to noise ratio	SNR	> 68 dB	
Second harmonic distortion	HD2	< 50 dBc	
Technology		22 nm	FDSOI CMOS

The design space associated with control-bounded conversion is nearly infinite. Although the architectures evaluated in this thesis are rather simple, there are a lots of possible improvements to be considered. Some of them are pointed out, and many are probably not yet thought of. The challenge

when working with control-bounded ADCs is not to come up with the good ideas, but to choose between them. The focus for this work, and the future circuit implementation, is to choose an architecture that is simple enough to allow for a reasonable implementation time, while at the same demonstrating some of the advanced features allowed by this new approach to A/D conversion.

1.3 Main Contributions

The main contribution of this thesis is the proposed ADC architecture, which is presented together with an analytic transfer function analysis, theoretical simulations and a discussion of some possible design challenges. This architecture will hopefully enable a more power efficient implementation of the Hadamard ADC introduced in [2]. Furthermore, the discussion of various design challenges provides a useful background when considering a control-bounded ADC for a multi-input application.

In addition, an important contribution of this work is to develop an understanding of the control-bounded ADCs operating principle, together with the development of necessary simulation framework. As this is a new and different approach to A/D conversion, the available literature on the topic is limited. Extensive use of simulations, together with calculations and hypothesis testing has been necessary to properly understand the concept. Together with the associated simulation framework, this constitutes an important basis for further work.

1.4 Some Terminology

Throughout the thesis, certain terms with slightly overlapping reach are frequently used. To avoid any misunderstanding, their usage is stated in this section.

1.4.1 Architecture, Implementation, Topology and Design

Architecture is used to describe the internal structure of a system. We use the term architecture on different abstraction levels, e.g. to describe the building blocks of the overall ADC or the internal building blocks of the analog system.

Implementation is reserved for the lowest abstraction level, i.e. transistor level implementations of involved components.

Topology is slightly overlapping with architecture. In this thesis, the word topology is placed between implementation and architecture on the abstraction level scale. We will for instance refer to different integrator topologies,

which is the structure within an "interator-box", but without considering any transistors.

Design is used either to describe the *process* of creating something, or the complete result when this process is finished.

1.4.2 Signal, Input, Channel, Element and State

Signal is used to describe a time-varying, information-carrying quantity. A signal may be a scalar, e.g. $u(t) \in \mathbb{R}$, or a vector, e.g. $\boldsymbol{u}(t) \in \mathbb{R}^N$. We say that a multi-channel receiver system has one, multi-dimensional input signal.

An **Input** is a signal entering the system, or an internal subsystem of the receiver.

A **Channel** constitutes one dimension of the multi-dimensional interface to the analog world. Each channel carries one **element** of the input signal. An *input channel* is in this thesis considered a redundant expression, and means the same as a channel.

A **State** is an element of the multi-dimensional state vector of the analog system. The analog system of the ADC is described using a state-space notation, and a state could for instance represent a voltage on an internal node of the analog system.

1.5 Thesis Outline

The thesis is structured as follows.

Chapter 2 provides a short introduction to conventional oversampling A/D converters, and forms a basis for comparison with the presented control-bounded ADC.

Chapter 3 gives a general, theoretical introduction to the control-bounded conversion principle. The goal of the chapter is to provide an understanding of the operating principle and the fundamental building blocks of the ADC. When describing the control-bounded ADC we follow the language and conventions established by previous publications, and the content of this chapter is very close to that of [2], chapter 4. It is however less general and limited to what is considered relevant for the remaining part of the thesis.

Chapter 4 presents the chain-of-integrator ADC, which is the simplest and most straight-forward control-bounded ADC architecture. This simple structure is the basis for the proposed ADC, and its analysis reveals useful results that is also applicable to other architectures.

Chapter 5 covers the proposed ADC architecture. A discussion of possible hardware implementations and theoretical simulation results is presented. An analytic transfer function analysis is also provided.

Chapter 6 includes discussions of several topics that did not fit naturally in other parts of the thesis. Both potential for future improvement and limitations of the presented work is discussed.

Finally, **Chapter 7** concludes the thesis and present the current plan for future progress.

Chapter 2

Oversampling A/D Converters

It will become apparent that the control-bounded ADC shares some similarities with conventional oversampling converters, and in particular the continuous-time $\Sigma\Delta$ converter. In order to show where the control-bounded converter distinguishes from these architectures, a brief introduction to conventional oversampling ADCs is included in this section. The presented material is assumed well known to the reader, and is only included to establish a basis of comparison. For a proper introduction to the topic, the reader is referred to [3].

An oversampling ADC is based on sampling the input signal at a frequency much higher than the Nyquist rate. For an analog input signal that is bandlimited to f_0, we define the oversampling ratio as

$$\text{OSR} \triangleq \frac{f_s}{2f_0} \tag{2.1}$$

where f_s is the sampling frequency of the ADC. Sampling at a higher frequency generates redundant signal information, and a single estimate of the input signal is typically obtained through some kind of decimation filter. The redundancy is this way utilized to give a higher resolution, or equivalently reduced requirements on the involved circuit components.

Straight forward oversampling will itself give an improved signal-to-noise ratio (SNR) of 3dB per doubling of OSR [3]. The performance of the oversampling converter is further improved by noise shaping of the quantization noise, through a feedback loop with a loop filter. Such a system is known as a $\Sigma\Delta$ ADC and the part of the system that performs the noise shaping is called a $\Sigma\Delta$ modulator. Such a system is illustrated in figure 2.1. In figure 2.1, the box labeled "S/H" performs the sample-and-hold operation,

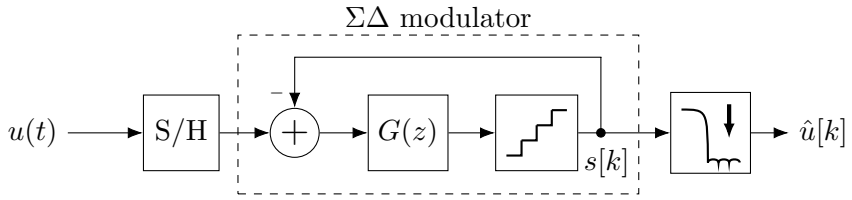

Figure 2.1: A discrete-time ΣΔ ADC.

and passes a discrete-time signal to the ΣΔ modulator. The ΣΔ modulator performs a quantization of the signal, together with a noise shaping of the quantization noise. It is common to include an anti-aliasing filter in front of the S/H-operation. The output of the ΣΔ modulator is passed on to a digital decimation filter. This filter usually performs a combination filtering and down-sampling.[1]

The system shown in figure 2.1 is called a discrete-time ΣΔ ADC because the ΣΔ modulator has a discrete-time input. A continuous-time ΣΔ converter is achieved by including the sampling in the feedback loop, as shown in figure 2.2. In this case, an eventual anti-aliasing filter is part of the loop filter $G(\omega)$.

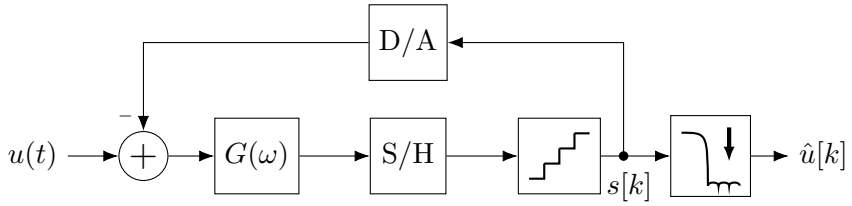

Figure 2.2: A continuous-time ΣΔ ADC.

Transfer Function Analysis

A transfer function analysis of the discrete-time ΣΔ modulator is obtained by evaluating the linearized model shown in figure 2.3. The analysis of the continuous-time modulator is similar. This model approximates the quantization error as a signal being independent of the input. This can of course not be strictly true, but is a useful approximation for the analysis.

Let $U(z), E(z)$ and $S(z)$ be the signals Z-transformed version of $u[k], e[k]$ and $s[k]$. The modulator output is then given by

$$S(z) = E(z) + G(z)[U(z) - S(z)], \tag{2.2}$$

[1]The figure indicates a low-pass response for the decimation filter, but it could in the general case have any pass band.

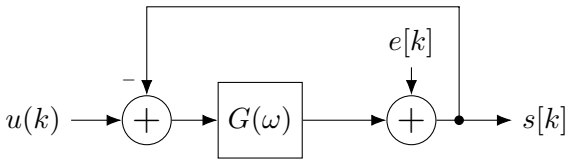

Figure 2.3: A simplified, linear model of the discrete-time ΣΔ ADC.

and the signal- and noise transfer function can be recognized by evaluating

$$S(z) = \underbrace{\frac{1}{1+G(z)}}_{\text{NTF}} E + \underbrace{\frac{G(z)}{1+G(z)}}_{\text{STF}} U(z). \tag{2.3}$$

as NTF $= \frac{1}{1+G(z)}$ and STF $= \frac{G(z)}{1+G(z)}$. Because the input signal and the quantization noise experience different transfer functions, it is possible to shape the noise such that most of the quantization noise appears outside the frequency band of interest, while simultaneously leaving the actual signal unchanged. This is the effect known as noise shaping. Note here that the necessary condition for noise shaping is that the signal and the quantization error enters the system at different points in the signal flow.

Chapter 3

Control-Bounded A/D Conversion

3.1 History and Background

Control-bounded A/D conversion is a conceptually new approach to the problem of creating a digital representation of an analog signal. The conversion technique has developed over the last years, and the progress is mainly pushed forward by prof. Hans-Andrea Loeliger et al., from the Signal and Information Processing Laboratory (ISI), ETH Zürich. The concept was first introduced at the IEEE Information Theory & Applications Workshop (ITA), february 2011 [4]. In this paper, the main building blocks of a control-bounded ADC was presented, but no explicit example of such an ADC was given, and no behavioural analysis presented. The approach was further developed in [5], which was published for the same conference in 2015. In this paper, the conversion algorithm is improved and a limited transfer function analysis is presented. The latest publication on control-bounded conversion is from 2020 [1]. This is a longer paper with the goal of providing the sufficient information for analog designers to experiment with control-bounded ADCs. The paper provides a more details on the operation of the building blocks, together with a full transfer function analysis. Measurements on a proof-of-concept hardware prototype is also presented.

In addition to the mentioned papers, Hampus Malmberg, co-author of the latest paper [1], has recently defended his Ph.D. on control-bounded converters. The thesis [2] is not yet published, but the author has been given access to a draft which serves as the main source of information on this topic. Malmberg also held a presentation at the 2020 IEEE International Symposium on Circuits and Systems (ISCAS) [6], where he presented the basic concept of control-bounded ADC, together with the Hadamard ADC

which is a basis for the architecture proposed in a later chapter.

In this section, the operating principle of a control-bounded converter is described in detail, and we follow the notation established in [1]. The theoretical presentation given in this section will be very close to that of [2], but less general and limited to what is necessary for understanding the the rest of the thesis.

3.2 Overview

The control-bounded ADC approaches the A/D conversion problem differently compared to conventional A/D converters. The conceptual difference lies in the view on sampling. In a control-bounded converter, the analog input signal is never sampled in the traditional way. The circuit that constitutes a control-bounded ADC still contains quantizers, but the quantized signals are never treated as a sampled and quantized version of the input. Instead, they are intermediate digital signals that only indirectly relates to the input, and they are used by a digital estimation filter to perform the digital estimate of the input signal.

To clarify this, consider the general block diagram shown in figure 3.1.

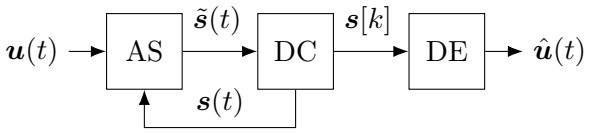

Figure 3.1: A block diagram of the control-bounded ADC. Figure from [2].

As the figure indicates, the control-bounded ADC consists of three main building blocks; an analog system (AS), a digital control (DC) and a digital estimator (DE). The signals $\boldsymbol{u}(t), \hat{\boldsymbol{u}}(t), \tilde{\boldsymbol{s}}(t), \boldsymbol{s}[k]$ and $\boldsymbol{s}(t)$ are in general vector-valued functions. The analog system amplifies the input signal $\boldsymbol{u}(t)$, preferably with very high gain within the frequency band of interest. The digital control stabilizes the analog system by forcing the internal states of the system to stay within its bounds. The internal states are observed through the *control observation* $\tilde{\boldsymbol{s}}(t)$ and controlled through the *control contribution* $\boldsymbol{s}(t)$. The digital estimator takes the *control signal* $\boldsymbol{s}[k]$ as an input and forms the digital estimate $\hat{\boldsymbol{u}}(t)$ of $\boldsymbol{u}(t)$.

Note that the output of the digital estimator is denoted as a continuous-time estimate $\hat{\boldsymbol{u}}(t)$ instead of a discrete-time estimate $\hat{\boldsymbol{u}}[k]$. The digital estimator models the continuous-time dynamics of the analog system, and is thereby capable of estimating $\boldsymbol{u}(t)$ at arbitrary time instances. The actual estimates will obviously be computed at discrete time steps, but because the

digital estimator itself imposes no criteria on this time interval, the output is denoted as a continuous time estimate.

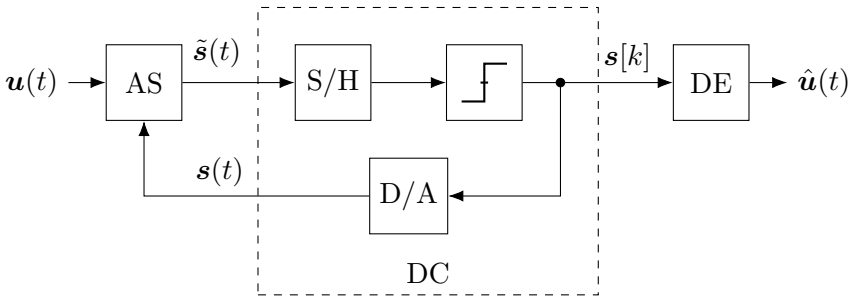

Figure 3.2: A control-bounded ADC with the DC box opened

Before going into detail on each of these building blocks, consider figure 3.2, which shows the same block diagram, but with the DC box opened. From this figure it is evident that the structure of the control-bounded ADC is very similar to that of the continuous-time $\Sigma\Delta$ modulator in figure 2.2. The control-bounded converter may in fact be viewed as a generalization of the continuous-time $\Sigma\Delta$ ADC. As mentioned, the main difference between these architectures arises from the different interpretation of the control signal $\boldsymbol{s}[k]$. In the $\Sigma\Delta$ ADC, this signal is viewed as a filtered, sampled and quantized version of the input signal and the digital output is obtained by averaging this signal through a decimation filter. In the control-bounded perspective, the direct relation between $\boldsymbol{s}[k]$ and $\boldsymbol{u}(t)$ is ignored completely. Instead, we focus solely on the fact that $\boldsymbol{s}(t)$ is the contribution needed to stabilize the internal states of the analog system. This view leads to a different estimation filter for the reconstruction of $\hat{\boldsymbol{u}}(t)$.

It should be noted that the contribution of the control-bounded ADC is not to provide an alternative decimation filter to already existing $\Sigma\Delta$ ADCs. Stepping away from the sampling oriented view of $\boldsymbol{s}[k]$, relaxes the requirements on the parts of the circuit that produces this signal. More importantly, the indirect relation between $\boldsymbol{s}[k]$ and $\boldsymbol{u}(t)$ enables a more general interaction between the analog and the digital parts of the system. There are for instance no need to require a one-to-one relation between the elements of these two signals, and they need not be of the same dimension. Each element of $\boldsymbol{s}[k]$ might contain information from all elements of $\boldsymbol{u}(t)$. These features are particularly interesting for multi-cannel systems which is of special interest in this thesis.

3.3 Analog System

The analog system, here assumed to be a continuous time filter, sets the frequency response of the overall ADC, and is designed to amplify the fre-

quency band of interest. As stability of the analog system is controlled digitally, the analog system itself need not be stable.

3.3.1 State Space Model

The dynamics of the analog system is described using a state space model notation, illustrated in figure 3.3. The multi-channel input signal $\boldsymbol{u}(t)$, the

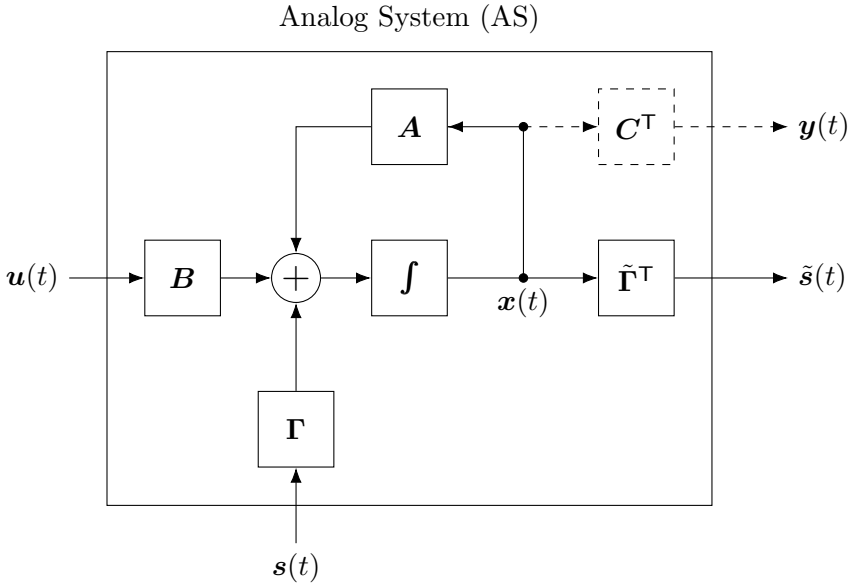

Figure 3.3: State space model of the AS. Figure from [2].

state-vector $\boldsymbol{x}(t)$ and the control contribution $\boldsymbol{s}(t)$ is related by the differential equation

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}(t) + \boldsymbol{\Gamma}\boldsymbol{s}(t). \tag{3.1}$$

where

$$\boldsymbol{u}(t) \triangleq (u_1(t), ..., u_L(t))^{\mathsf{T}} \in \mathbb{R}^L, \tag{3.2}$$

$$\boldsymbol{x}(t) \triangleq (x_1(t), ..., x_N(t))^{\mathsf{T}} \in \mathbb{R}^N \tag{3.3}$$

and

$$\boldsymbol{s}(t) \triangleq (s_1(t), ..., s_M(t))^{\mathsf{T}} \in \mathbb{R}^M. \tag{3.4}$$

This system is said to have L input channels, M controls and N states. We will refer to $\boldsymbol{A} \in \mathbb{R}^{N\times N}$, $\boldsymbol{B} \in \mathbb{R}^{N\times L}$ and $\boldsymbol{\Gamma} \in \mathbb{R}^{N\times M}$ as the *system matrix*, the *input matrix* and *the control input matrix* respectively. We will also refer to N as the *system order*.

The only physical output of the analog system is the control observation

$$\tilde{\boldsymbol{s}}(t) \triangleq \tilde{\boldsymbol{\Gamma}}^{\mathsf{T}}\boldsymbol{x}(t) \in \mathbb{R}^M, \tag{3.5}$$

which is used by the digital control to produce the control signal $\boldsymbol{s}[k]$. The control observation is a linear mapping of the internal state-vector, through the *control observation matrix* $\tilde{\boldsymbol{\Gamma}}^{\mathsf{T}} \in \mathbb{R}^{M\times N}$. The second output of the analog system is the purely conceptual signal

$$\boldsymbol{y}(t) \triangleq \boldsymbol{C}^{\mathsf{T}}\boldsymbol{x}(t) \in \mathbb{R}^{\tilde{N}}, \tag{3.6}$$

which is used by the digital estimator to produce the estimate $\hat{\boldsymbol{u}}(t)$. This signal has no physical meaning, and the *signal observation matrix* $\boldsymbol{C}^{\mathsf{T}} \in \mathbb{R}^{\tilde{N}\times N}$ is basically telling the digital estimation algorithm which of the internal analog states that could be treated as bounded. Thus $\boldsymbol{y}(t)$ and $\boldsymbol{C}^{\mathsf{T}}$ does only exist conceptually inside the digital estimator.

3.3.2 Transfer Function and Impulse Response Matrix

The transfer function of the analog system gives the frequency domain relation between the input $\boldsymbol{U}(\omega)$ and the output $\boldsymbol{Y}(\omega)$. Hence for the general case of L input channels and $\tilde{N}$ outputs, the analog transfer function (ATF) is a $\tilde{N}$-by-L matrix, defined by $\boldsymbol{Y}(\omega) = \boldsymbol{G}(\omega)\boldsymbol{U}(\omega)$. Each element $G_{i,j}(\omega)$ of $\boldsymbol{G}(\omega)$ is the transfer function from $U_j(\omega)$ to $Y_i(\omega)$. From (3.1) the ATF is obtained as

$$\boldsymbol{G}(\omega) = \boldsymbol{C}^{\mathsf{T}}\left(j\omega\boldsymbol{I}_N - \boldsymbol{A}\right)^{-1}\boldsymbol{B}, \tag{3.7}$$

and the derivation is given in appendix A. The analog impulse response matrix is then obtained from the inverse Laplace transform as

$$\boldsymbol{g}(t) = \boldsymbol{C}^{\mathsf{T}}\exp(\boldsymbol{A}t)\boldsymbol{B}, \tag{3.8}$$

where exp denotes the matrix exponential.

3.4 Digital Control

The digital control is a discrete time system which serves the purpose of stabilizing the analog system. It includes a sample-and-hold circuit, a one-bit quantizer and a D/A converter, as shown in figure 3.2. The control observation $\tilde{\boldsymbol{s}}(t)$ is sampled and quantized with a period T, resulting in the digital control signal $\boldsymbol{s}[k]$ which is passed on to the digital estimator. The D/A converter is in this thesis assumed to be a non-return to zero (NRZ) DAC generating the control contribution $\boldsymbol{s}(t)$.

The digital control is called effective if it manages to keep the state vector bounded, given a bounded input. The input vector $\boldsymbol{u}(t)$ is bounded if it satisfies

$$||\boldsymbol{u}(t)||_\infty \leq b_{\boldsymbol{u}} \ \forall t. \tag{3.9}$$

Equivalently, the state vector $\boldsymbol{x}(t)$ is bounded if it satisfies

$$||\boldsymbol{x}(t)||_\infty \leq b_{\boldsymbol{x}} \ \forall t. \tag{3.10}$$

In this thesis, the input signal will always be assumed bounded, and the boundary $b_{\boldsymbol{u}}$ is assumed to be determined by the application. The boundary for the state vector, $b_{\boldsymbol{x}}$, is a free variable and determines the magnitude of the state vector of the analog system.

A thorough analysis of the criteria for an effective control is found in [2]. The analysis is useful for the theoretical understanding of the system, but not necessary for the design process and is therefore beyond the scope of this paper. Intuitively however, there are three quantities affecting the stability of the analog system. The sampling period T of the digital control, the unity gain frequency of the analog system and the boundary $b_{\boldsymbol{x}}$. Increasing the speed of the analog system would require a shorter sampling period to counteract the faster growth of the system states. Reducing the boundary $b_{\boldsymbol{x}}$ would require either reducing the speed of the analog system or increasing the sampling frequency, in order to maintain a tighter bound.

It will become apparent in the next section that the performance of the overall ADC is related to the digital controls ability to bound the state vector. Designing the ADC for a stability guarantee means that it is theoretically impossible for the state vector to grow beyond $b_{\boldsymbol{x}}$ at any point in time, given any valid input signal. This will of course result in a large stability margin most of the time, which means that there is potential for increased performance not being utilized. The preferred way of tuning the stability of the system is therefore through simulations, and then to include the possibility of a full system reset if it happens to become unstable.

3.5 Digital Estimator

The digital estimator (DE) forms an estimate $\hat{\boldsymbol{u}}(t)$ of $\boldsymbol{u}(t)$ based on the control signals $\boldsymbol{s}[k]$ and the knowledge of the analog systems parameters. The purpose of this section is to describe the digital estimation problem, and to present the optimum linear estimation filter.

3.5.1 Statistical Estimation Problem and Transfer Functions

In the following analysis, the system described by (3.1) is assumed to be invariant and stable. This assumption is only needed to derive an analytic transfer function expression, and will not limit the actual estimation filter.

The objective of the digital estimator is to construct a digital estimate $\hat{\boldsymbol{u}}(t)$ of $\boldsymbol{u}(t)$, based on the control signals $\boldsymbol{s}[k]$. As highlighted previously in this chapter, the direct relation between $\boldsymbol{s}[k]$ and $\boldsymbol{u}(t)$ is ignored completely.

Instead, $\boldsymbol{s}[k]$ is only treated as the signal needed to stabilize the analog system, when triggered by an input signal $\boldsymbol{u}(t)$.

To formalize this approach, let $\breve{\boldsymbol{y}}(t) \triangleq (\boldsymbol{g} * \boldsymbol{u})(t) \in \mathbb{R}^{\tilde{N}}$ be the signal that would have occurred at the output of the analog system in the absence of any digital control. Futhermore, let $\boldsymbol{q}(t)$ be the control contribution signal seen at the output of the analog system. Because the control contribution enters the analog system in an additive way, their relation may be expressed as

$$\boldsymbol{y}(t) = \breve{\boldsymbol{y}}(t) - \boldsymbol{q}(t). \tag{3.11}$$

The situation is illustrated in figure 3.4. In this figure, solid lines represent the physical components of the ADC, while dashed lines represents conceptual quantities that only exist inside the digital estimator. It is illustrated how $\boldsymbol{q}(t)$ relates to the control contribution $\boldsymbol{s}(t)$. Because the digital estimator knows the parametrization of the analog system, as well as the waveform of the D/A converter, $\boldsymbol{q}(t)$ is (in principle) known from the observation of $\boldsymbol{s}[k]$. Note that this illustration is only meant to illustrate the estimation problem of the digital estimator, not to show how the actual estimate is computed. We denote the frequency response of the digital estimation filter by $\boldsymbol{H}(\omega)$ and the continuous time estimate $\hat{\boldsymbol{u}}(t)$ of $\boldsymbol{u}(t)$ is obtained by

$$\hat{\boldsymbol{u}}(t) = (\boldsymbol{h} * \boldsymbol{q})(t) \in \mathbb{R}^{L}. \tag{3.12}$$

Because the objective of the analog system is to greatly amplify the sought frequency content of $\boldsymbol{u}(t)$, both $||\breve{\boldsymbol{y}}(t)||_\infty$ and $||\boldsymbol{q}(t)||_\infty$ will be very large compared to $||\boldsymbol{y}(t)||_\infty$, which is bounded due to (3.10). We can therefore approximate $\breve{\boldsymbol{y}}(t) \approx \boldsymbol{q}(t)$ which is equivalent to the approximation $\boldsymbol{y}(t) \approx \boldsymbol{0}$. Hence the estimate may be written as

$$\begin{aligned}
\hat{\boldsymbol{u}}(t) &= (\boldsymbol{h} * \boldsymbol{q})(t) && (3.13)\\
&= (\boldsymbol{h} * \breve{\boldsymbol{y}})(t) - (\boldsymbol{h} * \boldsymbol{y})(t) && (3.14)\\
&\approx (\boldsymbol{h} * \breve{\boldsymbol{y}})(t) && (3.15)\\
&= (\boldsymbol{h} * \boldsymbol{g} * \boldsymbol{u})(t) && (3.16)
\end{aligned}$$

It is evident that $\hat{\boldsymbol{u}}(t)$ could have been computed with arbitrary accuracy, if the output $\boldsymbol{y}(t)$ was completely known to the digital estimator. This if statement is obvious, but it illustrates an important point. Instead of relying on inevitably inaccurate measurement of $\boldsymbol{y}(t)$ at fixed points in time, we choose to approximate this signal as being zero for all times t. The accuracy of the estimate then relies on the validity of the approximation $\boldsymbol{y}(t) \approx \boldsymbol{0}$, rather than the precision of a direct measurement of $\boldsymbol{y}(t)$ at some sampling points. This approximation is illustrated in figure 3.4, where it is indicated

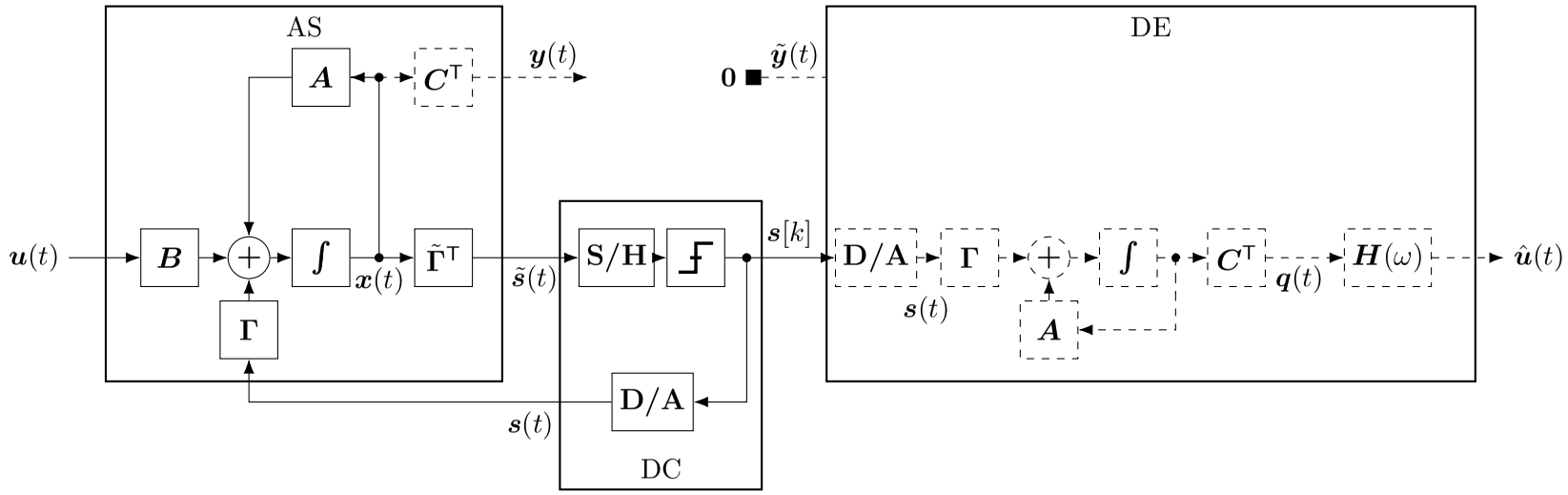

Figure 3.4: Block diagram of the complete control-bounded ADC, with the digital estimation problem visualized. The approximation $\boldsymbol{y}(t) \approx \mathbf{0}$ is indicated by the fixed observation of $\tilde{\boldsymbol{y}}(t) = \mathbf{0}$ outside the DE box.

that the actual analog system output is disregarded and substituted with the fictional observation $\tilde{\boldsymbol{y}}(t) = \mathbf{0}$.

Any deviation of $\boldsymbol{y}(t)$ from $\mathbf{0}$ will result in a conversion error, meaning that $\boldsymbol{y}(t)$ is the conversion error signal seen at the output of the analog system. This conversion error does not enter the estimate directly, but is filtered by $\boldsymbol{h}(t)$. From the Fourier transform of (3.16),

$$\hat{\boldsymbol{U}}(\omega) = \underbrace{\boldsymbol{H}(\omega)\boldsymbol{G}(\omega)}_{\text{STF}}\boldsymbol{U}(\omega) - \underbrace{\boldsymbol{H}(\omega)}_{\text{NTF}}\boldsymbol{Y}(\omega), \tag{3.17}$$

we recognize the signal and noise transfer functions as STF $= \boldsymbol{H}(\omega)\boldsymbol{G}(\omega)$ and NTF $= \boldsymbol{H}(\omega)$ respectively.

The Statistical Estimation Problem

Up to this point, the focus has been to describe the context and the operating principle of the digital estimator. To derive an expression for the actual estimation filter, the problem is first described as a statistical estimation problem. The derivation is carried out in detail in [2] and only the main results is presented in this section.

The error introduces by the approximation in (3.15) is treated by using statistical methods. In the following, both $\boldsymbol{y}(t)$ and $\boldsymbol{u}(t)$ is assumed to be independent, centered, multivariate and wide-sense stationary stochastic processes. The estimation filter is then determined by

$$\boldsymbol{h}(t) = \underset{\bar{\boldsymbol{h}}}{\operatorname{argmin}}\, \mathrm{E}[(\hat{\boldsymbol{u}}(t) - \boldsymbol{u}(t))^2] \tag{3.18}$$

$$= \underset{\bar{\boldsymbol{h}}}{\operatorname{argmin}}\, \mathrm{E}[((\bar{\boldsymbol{h}} * \boldsymbol{q})(t) - \boldsymbol{u}(t))^2]. \tag{3.19}$$

This minimization problem is exactly the objective of the Wiener-filter [7] and the impulse response matrix is given by the solution to the well-known Wiener-Hopf equations:

$$(\boldsymbol{h} * \boldsymbol{R}_{\boldsymbol{q}\boldsymbol{q}^\mathsf{T}})(\tau) = \boldsymbol{R}_{\boldsymbol{u}\boldsymbol{q}^\mathsf{T}}(-\tau) \tag{3.20}$$

where

$$\boldsymbol{R}_{\boldsymbol{q}\boldsymbol{q}^\mathsf{T}} \triangleq \mathrm{E}[\boldsymbol{q}(t)\boldsymbol{q}(t+\tau)^\mathsf{T}] \tag{3.21}$$

$$\boldsymbol{R}_{\boldsymbol{u}\boldsymbol{q}^\mathsf{T}} \triangleq \mathrm{E}[\boldsymbol{u}(t)\boldsymbol{q}(t+\tau)^\mathsf{T}] \tag{3.22}$$

are the autocovariance and cross-covariance matrices respectively. By taking the Fourier transform of (3.20) we obtain the frequency response matrix $\boldsymbol{H}(\omega)$ as

$$\boldsymbol{H}(\omega) = \boldsymbol{G}^\mathsf{H}(\omega)\left(\boldsymbol{G}(\omega)\boldsymbol{G}^\mathsf{H}(\omega) + \eta^2\boldsymbol{I}_N\right)^{-1}, \tag{3.23}$$

and the reader is referred to [2] for computational details. The parameter η is defined as

$$\eta \triangleq \frac{\sigma_{\boldsymbol{y}}^2}{\sigma_{\boldsymbol{u}}^2}, \tag{3.24}$$

where $\sigma_{\boldsymbol{y}}^2$ and $\sigma_{\boldsymbol{u}}^2$ are the power spectral densities of $\boldsymbol{y}(t)$ and $\boldsymbol{u}(t)$ respectively.

3.5.2 Estimation filter implementation

With the digital estimation filter described by (3.23), the estimation could in principle be carried out by computing $\hat{\boldsymbol{u}}(t)$ as in (3.12). This computation is however not straight forward. First of all, the elements of $\boldsymbol{q}(t)$ will necessarily be very large in magnitude, as this was the condition for the approximation (3.15). Carrying out a continuous time convolution with this unbounded signal would obviously lead to numerical problems. In addition the computation of $\boldsymbol{q}(t)$ from $\boldsymbol{s}[k]$, as illustrated in figure 3.4, might be computationally expensive.

In [4] it was shown that the estimate $\hat{\boldsymbol{u}}(t)$ can be computed in an alternative way, using a non-standard version of the Kalman smoothing algorithm. This algorithm converges to the the estimate (3.12) as the considered time window extends towards infinity. The algorithm is also indifferent to the stability assumptions made in the previous section.

As the algorithm is nothing more than an efficient way of computing (3.12), a description of the implementation is not needed for understanding the behaviour of the digital estimator. It is however important for simulations and a concise description of the filter algorithm is provided in appendix B.

3.5.3 Practical Remarks

We conclude this section with some practical considerations.

Controlling the Filter Bandwidth

In (3.24) the parameter η was defined in terms of the power spectral densities of $\boldsymbol{y}(t)$ and $\boldsymbol{u}(t)$, when these signals are modeled as independent stochastic processes. In practice however, η is a free variable and is used by the designer to control the bandwidth of the estimation filter. To see this, consider the scalar input case where both $\boldsymbol{G}(\omega)$ and $\boldsymbol{H}(\omega)$ are column vectors. In this case, the noise transfer function (3.23) reduces to

$$\boldsymbol{H}(\omega) = \text{NTF} = \frac{\boldsymbol{G}^{\mathsf{H}}(\omega)}{||\boldsymbol{G}(\omega)||_2^2 + \eta^2} \in \mathbb{C}^{1\times\tilde{N}}, \tag{3.25}$$

and the signal transfer function becomes

$$\text{STF} = \frac{||\boldsymbol{G}(\omega)||_2^2}{||\boldsymbol{G}(\omega)||_2^2 + \eta^2} \in \mathbb{R}. \tag{3.26}$$

Assuming $||\boldsymbol{G}(\omega)||_\infty$ is monotonically decreasing in ω, the bandwidth of the digital estimator may be defined in terms of the critical frequency, ω_c, as

$$||\boldsymbol{G}(\omega_c)||_2^2 = \eta^2. \tag{3.27}$$

As the parameter η appeared with a precise definition from the optimum filter derivation, it might sound strange that this parameter is now treated as a free variable. This may be understood as follows. In the derivation that lead to (3.24), no assumptions where made on the bandwidth of the input signal $\boldsymbol{u}(t)$. If $||\boldsymbol{G}(\omega)||_\infty$ is monotonically decreasing in ω, then above a certain frequency, the magnitude of the error signal, $\boldsymbol{y}(t)$, and the input signal, $\boldsymbol{u}(t)$, will become comparable. Beyond this frequency, $\boldsymbol{q}(t)$ contains more error than information, and the quality of the estimate is improved by reducing the influence of these higher frequency components. Therefore, with no prior knowledge of $\boldsymbol{u}(t)$, the optimum "cut-off" frequency of the estimation filter is given by $||\boldsymbol{G}(\omega_c)||_2 = \sigma_{\boldsymbol{y}}^2/\sigma_{\boldsymbol{u}}^2$.

In a practical application however, we usually know which frequency components of $\boldsymbol{u}(t)$ that contains the sought information. In this case the quality of the estimate would obviously be improved by choosing the cut-off frequency based on this prior knowledge.

Signal-to-Noise Ratio

An analytic derivation of the SNR of the control-bounded ADC is given in [1]. The analysis models the output of the analog system, $\boldsymbol{y}(t)$, as white noise, i.e. assuming the power spectral density is given by $\boldsymbol{S}_{\boldsymbol{y}\boldsymbol{y}^\intercal}(\omega) \approx \sigma_{\boldsymbol{y}|\mathcal{B}}^2 \boldsymbol{I}_{\tilde{N}}$. In this expression, $\mathcal{B}$ denotes the frequency band of interest and $\sigma_{\boldsymbol{y}|\mathcal{B}}^2$ is the variance of $\boldsymbol{y}(t)$ within this frequency band. From this assumption an approximated expression for the SNR is obtained as

$$\text{SNR} \approx \frac{\sigma_{\boldsymbol{y}|\mathcal{B}}^2}{2\pi} \int_{\omega \in \mathcal{B}} \frac{1}{||\boldsymbol{G}(\omega)||_2^2} d\omega. \tag{3.28}$$

Even though this is an approximation it reveals a useful intuition of how the quantities affect the performance of the ADC. $\sigma_{\boldsymbol{y}|\mathcal{B}}^2$ relates to the magnitude of $\boldsymbol{y}(t)$, and is minimized by tightening the control bound $b_{\boldsymbol{x}}$. $||\boldsymbol{G}(\omega)||_2^2$ is maximized by increasing the gain of the analog system. Therefore, a tight control bound together with a high analog system gain result in large SNR.

The SNR is also related to the bandwidth parameter η, as seen by considering the ratio between the STF and NTF

$$\frac{\mathrm{STF}(\omega_c)}{||\boldsymbol{H}(\omega_c)||_2} = \frac{||\boldsymbol{G}(\omega_c)||_2^2}{||\boldsymbol{G}(\omega_c)||_2^2 + \eta^2} \left(\frac{||\boldsymbol{G}(\omega_c)||_2}{||\boldsymbol{G}(\omega_c)||_2^2 + \eta^2} \right)^{-1} \tag{3.29}$$

$$= ||\boldsymbol{G}(\omega_c)||_2 \tag{3.30}$$

$$= \eta. \tag{3.31}$$

Therefore a trade-off has to be made between the bandwidth of the ADC and the suppression of the conversion error. This is similar to the trade-off in a $\Sigma\Delta$ ADC when considering the cut-off frequency of the decimation filter. This trade-off will be exemplified when particular ADC implementations is considered in the following chapters.

Chapter 4

The Chain-of-Integrators ADC

The simplest control-bounded ADC is the Chain-of-integrators ADC, as presented in [1]. This architecture consists of an integrator chain, where each integrator is stabilized by a local, independent digital control loop. A reader that is familiar with $\Sigma\Delta$ converters, might recognize this structure as very similar to a MASH $\Sigma\Delta$ modulator, and it is shown in [2] that the performance is also very similar. A block diagram of the full chain-of-integrators ADC is shown in figure 4.1.

This rather simple structure does not not utilize much of the mentioned design flexibility associated with control-bounded converters. However, the chain-of-integrators serves as an important starting point, and a thorough understand of its operating principle is helpful before considering more advanced architectures. The theoretical analysis is also rather straight forward and several important results from the analysis of the chain-of-integrators are directly applicable to other architectures. In addition, the simulations on the chain-of-integrators demonstrates the developed simulation framework which is an essential tool for further work on the topic.

This chapter is organized as follows. The first section presents a description of the analog system together with a transfer function analysis, based on the results from section 3.3.2. The digital control is then briefly described and the conditions for an effective control is discussed. Finally we present and discuss the simulation results.

4.1 Analog System

The analog system of the chain-of-integrators ADC is part of the block diagram shown in figure 4.1. The (ideal) integrators has a transfer function β/s, and the parameter β is referred to as the *integrator gain*. The input signal $u(t)$ is passed through N such integrators, each being controlled by a local, independent control loop. Note that the input signal and output estimates of this ADC are both scalars, as a multi-input chain-of-integrators would be nothing more than L equal systems in parallel.

The system dynamics is described by the equation

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}_{CI}\boldsymbol{x}(t) + \boldsymbol{B}_{CI}u(t) + \boldsymbol{\Gamma}_{CI}\boldsymbol{s}(t). \tag{4.1}$$

where the system matrix and the input matrix are given by

$$\boldsymbol{A}_{CI} = \begin{pmatrix} 0 & & & \\ \beta & 0 & & \\ & \ddots & \ddots & \\ & & \beta & 0 \end{pmatrix} \in \mathbb{R}^{N\times N} \tag{4.2}$$

and

$$\boldsymbol{B}_{CI} = \begin{pmatrix} \beta & 0 & \cdots & 0 \end{pmatrix}^{\mathsf{T}} \in \mathbb{R}^{N\times 1} \tag{4.3}$$

respectively.

The N dimensional state-vector $\boldsymbol{x}(t)$ is observed directly by the local digital control, and the control observation matrix is given by

$$\tilde{\boldsymbol{\Gamma}}_{CI}^{\mathsf{T}} = \boldsymbol{I}_N. \tag{4.4}$$

The output of the 1 bit D/A converter is given by

$$s_i(t) = \begin{cases} \kappa, & \text{if } s_i[k] = 1 \\ -\kappa, & \text{if } s_i[k] = 0, \end{cases} \tag{4.5}$$

and κ is referred to as the *control gain*. The control matrix is given by

$$\boldsymbol{\Gamma}_{CI} = \begin{pmatrix} \kappa\beta & & \\ & \ddots & \\ & & \kappa\beta \end{pmatrix}. \tag{4.6}$$

As mentioned in chapter 3, the signal observation matrix $\boldsymbol{C}^{\mathsf{T}}$ maps the state vector $\boldsymbol{x}(t)$ to the output vector $\boldsymbol{y}(t)$. As this matrix is purely conceptual, it has no part in the physical implementation and may be chosen independent

of the analog system. Typically one would choose to map either all or only the last state to the output, by choosing either

$$\boldsymbol{C}_{CI_s}^{\mathsf{T}} = \begin{pmatrix} 0 & \cdots & 0 & 1 \end{pmatrix} \in \mathbb{R}^{1\times N} \tag{4.7}$$

or

$$\boldsymbol{C}_{CI_m}^{\mathsf{T}} = \boldsymbol{I}_N. \tag{4.8}$$

These different choices of $\boldsymbol{C}^{\mathsf{T}}$ results in what is referred to as single and multiple output reconstruction, respectively. Intuitively one would think that considering all internal states of the analog system in the estimation filter would give increased performance, and this is indeed the case. The computational complexity of the filter is also indifferent to the choice of $\boldsymbol{C}^{\mathsf{T}}$, so $\boldsymbol{C}_{CI_m}^{\mathsf{T}}$ is the natural choice for this matrix. The single output matrix is still considered in this thesis for the sake of a tractable analysis.

4.1.1 Transfer Function Analysis

For this scalar input ADC, the analog transfer function of 3.7 reduces to a column vector. Each element of the transfer function vector is given by

$$G_k(\omega) = \prod_{\ell=0}^{N-1} \frac{\beta}{j\omega}. \tag{4.9}$$

Hence for the single output reconstruction,

$$G_s(\omega) = \left(\frac{\beta}{j\omega}\right)^N \tag{4.10}$$

and

$$||G_s(\omega)||_2^2 = |G_{N-1}(\omega)|^2 = \left(\frac{\beta}{\omega}\right)^{2N}. \tag{4.11}$$

For multiple output,

$$||G_m(\omega)||_2^2 = \frac{1 - \left(\frac{\omega}{\beta}\right)^{2N}}{\left(\frac{\omega}{\beta}\right)^{2N}\left(1 - \frac{\omega^2}{\beta^2}\right)}. \tag{4.12}$$

A comparison of the analog transfer function obtained from single and multiple output is shown in figure 4.2, for $\beta = 2\pi \cdot 20\,\text{MHz}$ and $N = 5$. As the figure shows, the difference is mainly visible for frequencies above the unity gain of the integrators.

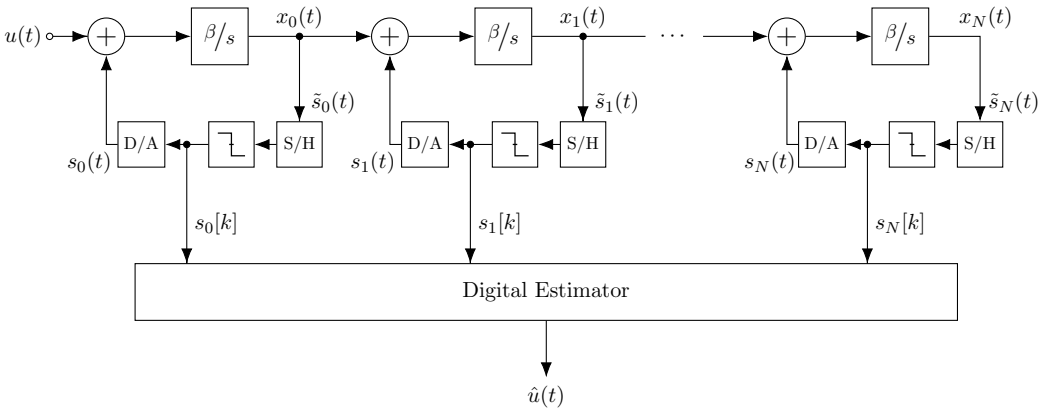

Figure 4.1: A block diagram of an Nth order chain-of-integrators ADC

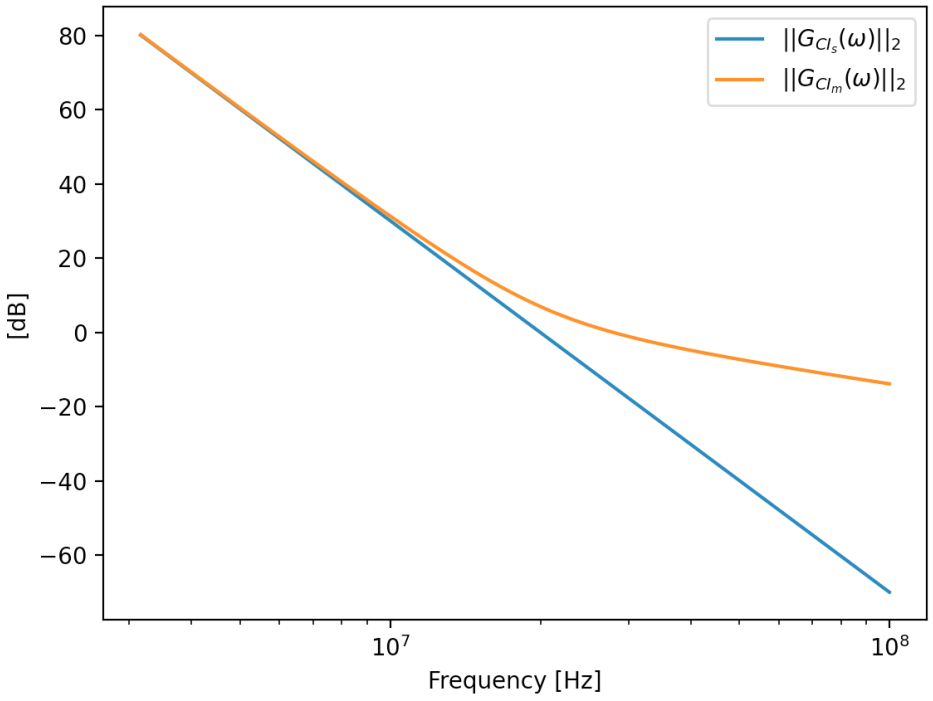

Figure 4.2: Comparison of analog transfer function obtained from single and multiple output, for $\beta = 2\pi \cdot 20\,\text{MHz}$ and $N = 5$.

4.2 Effective digital control

In the chain-of-integrators ADC, each integrator is under local digital control. This simple control structure is easy to implement and allows for an analytic derivation of the effective control criteria. This derivation is carried out in detail in [2] and the main results is presented here.

In the analysis, the input is assumed to be bounded, i.e. $|u(t)| \leq b_u \ \forall t$, and the conditions for effective control ensure $||\boldsymbol{x}(t)||_\infty \leq b_{\boldsymbol{x}} \ \forall t$, see section 3.4. For the chain-of-integrators this is guaranteed If

$$|\kappa| \geq b_{\boldsymbol{x}} \tag{4.13}$$

and

$$T|\beta|(|\kappa| + b_{\boldsymbol{x}}) \leq b_{\boldsymbol{x}}. \tag{4.14}$$

4.2.1 Implications on the Sampling Rate

A natural choice of κ and $b_{\boldsymbol{x}}$ is to let both equal the positive supply voltage. With $\kappa = b_{\boldsymbol{x}}$, (4.14) reduces to

$$T|\beta| \leq \frac{1}{2}. \tag{4.15}$$

To see how this condition influences the sampling rate, let $f_s = \frac{1}{T}$ be the sampling frequency of the digital control and let $f_u = \frac{\beta}{2\pi}$ be the unity gain frequency of the integrators. Equation (4.15) may then be written as

$$f_s \geq 4\pi f_u, \tag{4.16}$$

i.e. given $\kappa = b_{\boldsymbol{x}}$, the sampling rate must be approximately 12.6 times the unity gain frequency of the integrators in order to guarantee an effective control.

To place this requirement in the context of oversampling, the unity gain frequency must be related to the frequency band of interest. For simplicity, only the single output transfer function is now considered. Assume that the frequency content of the input signal is upper bounded by a frequency f_0 and the critical frequency of the estimation filter is set as $f_c \geq f_0$. From (3.27) and (4.11), the parameter η may be expressed as

$$\eta = \left(\frac{\beta}{\omega_c}\right)^N = \left(\frac{f_u}{f_c}\right)^N. \tag{4.17}$$

Furthermore, from (3.29)-(3.31), η is also the relation between the magnitude of the signal and noise transfer function at the critical frequency. From

(4.17) it is seen that η grows with system order as long as $\frac{f_u}{f_c} \geq 1$. We therefore define a practical limit

$$f_u \geq 2f_c \tag{4.18}$$

for this relation, in order to take advantage of the higher system order.

Based on this discussion the relation between the signal bandwidth f_0 and the sampling rate may ultimately be expressed as

$$f_s \geq 8\pi f_0, \tag{4.19}$$

or equivalently

$$\text{OSR} \geq 4\pi. \tag{4.20}$$

Thus, for a reasonable chain-of-integrators ADC with guaranteed stability and $\kappa = b_{\boldsymbol{x}}$, the minimum oversampling rate is approximately 12.6.

As mentioned in section 3.4, designing for stability guarantee is not attractive from a performance point of view. In section 3.5.3 it was shown that T, β and $b_{\boldsymbol{x}}$ are the only parameters affecting the SNR of a control-bounded ADC and it is therefore an inevitable trade-off between stability and performance. The preferred way of determining this trade-off is by simulations, and the results obtained in this section serves as a useful starting point.

4.3 Simulations

The process of simulating a control-bounded ADC is divided into two separate steps. First, the interaction between the analog system and the digital control is simulated in time domain to generate the control signal $\boldsymbol{s}[k]$. In the following, this part of the simulation process is referred to as the *system simulation*. These control signals are then applied as input to the digital estimation filter, which reconstructs the output estimate $\hat{\boldsymbol{u}}(t)$. For the chain-of-integrators ADC, the system simulations are done using the Spectre simulation platform [8] and the estimation filter is implemented offline in Python.

4.3.1 System Simulation

The circuit used for the system simulation is a 5th order system, derived from the block diagram of figure 4.1. The integrators are implemented using a first order opamp-RC filter as shown in figure 4.3, and the opamp is modelled by an ideal voltage controlled voltage source. The Sample-and-hold, 1 bit quantizer and 1 bit dac, are all incorporated in an ideal, clocked comparator written in VerilogA. The digital control signal $\boldsymbol{s}[k]$ does therefore not exist in the system simulation and is extracted from the control contribution, $\boldsymbol{s}(t)$, in a post-processing step.

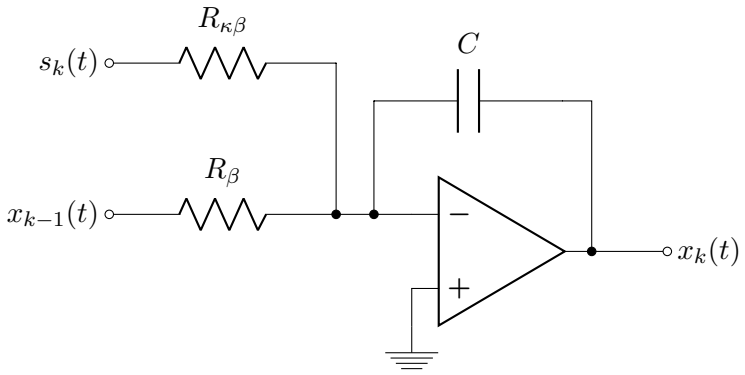

Figure 4.3: An Opamp-RC integrator used for the chain-of-integrators simulation.

4.3.2 Estimation Filter Implementation

The digital estimation filter is implemented in Python and the source code is available at GitHub [9]. The filter is first initialized by computing the offline matrices, defined by equations (B.4)-(B.10). The actual estimate is then carried out recursively as described by (B.1)-(B.3). As the recursive estimate takes place after the system simulation is complete, both the forward and the backward recursion uses all available control signals. In a production environment, this is obviously not possible and only a limited amount of "future" samples are available. The filter algorithm is quite sensitive to the boundary conditions, but usually a few hundred samples is enough for the filter to settle properly.

4.3.3 Simulation Results

The parameters used for the simulation is summarized in table 4.1. The integrator gain is achieved by choosing $R_\beta = 320\,\mathrm{k\Omega}$ and $C = 10\,\mathrm{fF}$. The comparator output is $\pm 1\,\mathrm{V}$, and the control gain is set to $\kappa = 1$ by having $R_{\kappa\beta} = R_\beta$.

A plot of the estimated power spectral density is provided in figure 4.4, for both the single and multiple output case. Note that the difference is mainly notable for frequencies above the critical frequency of the estimation filter.

Table 4.1: Simulation parameters, Chain-of-Integrators

Parameter	Symbol	Value
Critical frequency	f_c	10 MHz
Unity gain frequency	f_u	50 MHz
Integrator gain	β	315 MHz
Sampling frequency	f_s	650 MHz
Oversampling ratio	OSR $\left(\frac{f_s}{2f_c}\right)$	32.5
System order	N	5
Control gain	κ	1 V
State boundary (input boundary)	$b_{\boldsymbol{x}}$ ($b_{\boldsymbol{u}}$)	1 V (1 V)
Input signal frequency	f_{in}	650 kHz
Input signal amplitude	a_{in}	0.9 V

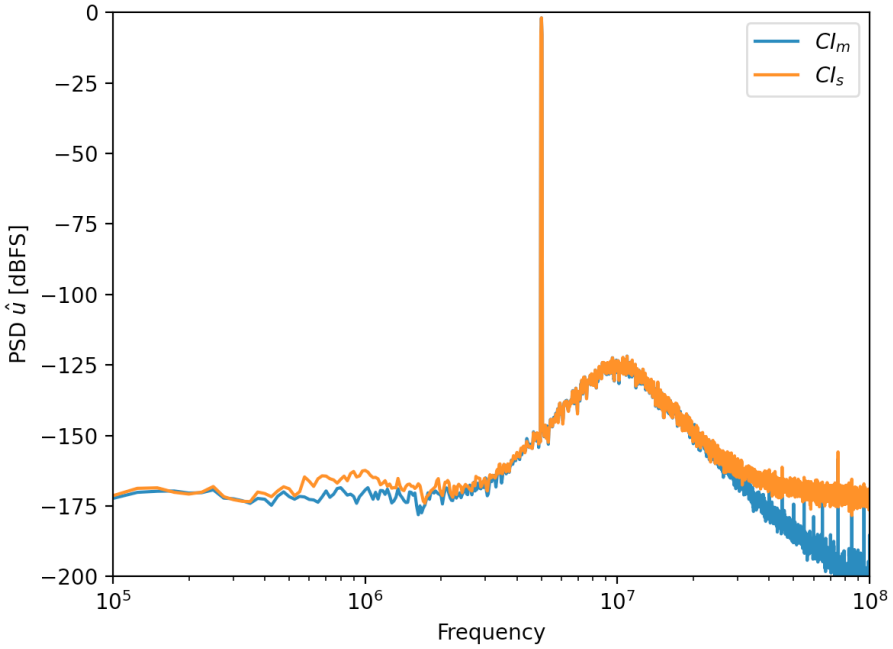

Figure 4.4: Estimated PSD for a 5th order chain-of-integrator ADC. CI_m and CI_s refers to multiple and single output reconstruction respectively.

Chapter 5

Proposed ADC Architecture

The proposed architecture is based on the Hadamard ADC that was proposed by Malmberg in [2] and [6]. The main objective of the Hadamard ADC is to distribute the sensitivity to component mismatch equally among all involved circuit components. In addition, the Hadamard ADC enables a beneficial way of combining multiple input channels. The properties of the Hadamard ADC will be further explored later in this chapter.

The key idea of the Hadamard ADC is to separate the logical states from the physical ones, which is achieved through a Hadamard transform of the state vector. By *logical states*, we mean the signals that are considered by the digital estimation filter to be a state of the analog system. These states may or may not coincide with the *physical states* of the analog system. Separating the two means that each element of the state vector, $\boldsymbol{x}(t)$, does not have a one-to-one relationship with the voltages on the different nodes of the analog system. This is a fundamental difference from the chain-of-integrator system where the physical and logical states coincides completely.

A proposed architecture for the Hadamard ADC is presented in [2] and [6]. This architecture has good mismatch properties and performs very well in combination with overcomplete control and multiple input channels. A disadvantage of this architecture is however the use of buffers in the analog system. These are active components that will consume a considerable amount of power, without contributing with any gain to the analog transfer function.

In this chapter, an alternative architecture for the Hadamard ADC is presented. The proposed architecture has no buffers, thereby providing more gain for the same number of active components. The architecture also show a similar tolerance to component mismatch as the original architecture. The proposed architecture is presented together with an analytical transfer function analysis and simulation results.

5.1 Analog System

5.1.1 Parametrization

The Hadamard transform is obtained by performing an analog multiplication of the input signal with a Hadamard matrix. For N being powers of two, the Hadamard matrix is defined recursively as

$$\boldsymbol{H}_N = \boldsymbol{H}_2 \otimes \boldsymbol{H}_{N/2} \tag{5.1}$$

where

$$\boldsymbol{H}_2 = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \tag{5.2}$$

The Hadamard matrix is an orthogonal matrix with the useful properties

$$\boldsymbol{H}_N = \boldsymbol{H}_N^{\mathsf{T}} \tag{5.3}$$

and

$$\boldsymbol{H}_N^{\mathsf{T}} \boldsymbol{H}_N = N \boldsymbol{I}_N. \tag{5.4}$$

The Hadamard analog system is described by the equations

$$\dot{\boldsymbol{x}}(t) = \boldsymbol{A}_H \boldsymbol{x}(t) + \boldsymbol{B}_H \boldsymbol{u}(t) + \boldsymbol{\Gamma}_H \boldsymbol{s}(t) \tag{5.5}$$

$$\boldsymbol{y}(t) = \boldsymbol{C}_H^{\mathsf{T}} \boldsymbol{x}(t) \tag{5.6}$$

and

$$\tilde{\boldsymbol{s}}(t) = \tilde{\boldsymbol{\Gamma}}_H^{\mathsf{T}} \boldsymbol{x}(t). \tag{5.7}$$

The $\boldsymbol{\Gamma}$-matrices will be covered together with the digital control in the next section. The proposed architecture is derived from the parametrization introduced in [2], where

$$\boldsymbol{A}_H = \frac{1}{N} \boldsymbol{H}_N \boldsymbol{A}_{CI} \boldsymbol{H}_N^{\mathsf{T}}, \tag{5.8}$$

$$\boldsymbol{B}_H = \frac{1}{N} \boldsymbol{H}_N \boldsymbol{B}_{CI} \tag{5.9}$$

and

$$\boldsymbol{C}_H^{\mathsf{T}} = \boldsymbol{C}_{CI}^{\mathsf{T}} \boldsymbol{H}_N^{\mathsf{T}} \tag{5.10}$$

$\boldsymbol{A}_{CI}, \boldsymbol{B}_{CI}$ and $\boldsymbol{C}_{CI}^{\mathsf{T}}$ refers to the chain-of-integrator matrices defined in equation (4.2)-(4.8). The architecture associated with this parametrization uses N integrators and N buffers, and is presented in [2] and [6]. As mentioned in the introduction, the motivation for the proposed architecture is to use all active components for integration. By simply replacing all buffers with integrators, a different parametrization is obtained, with a very similar transfer function and performance characteristics.

For the scalar input case, the proposed system is described by[1]

$$\boldsymbol{A}_H = \boldsymbol{H}'_N \boldsymbol{A}' \in \mathbb{R}^{N\times N}, \tag{5.11}$$

$$\boldsymbol{B}_H = \boldsymbol{H}'_N \boldsymbol{B}_{CI} \in \mathbb{R}^{N\times 1} \tag{5.12}$$

and

$$\boldsymbol{C}_H = \boldsymbol{C}_{CI_s} \in \mathbb{R}^{N\times 1} \tag{5.13}$$

or

$$\boldsymbol{C}_H = \boldsymbol{C}_{CI_m} \in \mathbb{R}^{N\times N}, \tag{5.14}$$

where $\boldsymbol{C}_{CI_s}$ and $\boldsymbol{C}_{CI_m}$ yields single and multiple output reconstruction respectively. Furthermore,

$$\boldsymbol{H}'_N = \begin{bmatrix} \boldsymbol{H}_{N/2} & \boldsymbol{0}_{N/2} \\ \boldsymbol{0}_{N/2} & \boldsymbol{H}_{N/2} \end{bmatrix} \in \mathbb{R}^{N\times N} \tag{5.15}$$

and

$$\boldsymbol{A}' = \begin{bmatrix} \boldsymbol{0}_{N/2} & \beta\boldsymbol{L}_{N/2} \\ \beta\boldsymbol{I}_{N/2} & \boldsymbol{0}_{N/2} \end{bmatrix} \in \mathbb{R}^{N\times N}. \tag{5.16}$$

The matrix $\boldsymbol{H}'_N$ is referred to as *the reduced Hadamard matrix*. The matrix $\boldsymbol{A}'$ is described as a block matrix and the sub-matrix $\boldsymbol{A}'_{01} = \beta\boldsymbol{L}_{N/2}$ is a strictly lower triangular matrix. $\boldsymbol{L}_{N/2}$ is given by

$$\boldsymbol{L}_{N/2} = \begin{pmatrix} 0 & & & & \\ 1 & 0 & & & \\ 0 & 1 & 0 & & \\ \vdots & \ddots & \ddots & \ddots & \\ 0 & \cdots & 0 & 1 & 0 \end{pmatrix} \in \mathbb{R}^{\frac{N}{2}\times\frac{N}{2}}. \tag{5.17}$$

As for the chain-of-integrators, both single and the multiple output reconstruction is possible. However, for the sake of a tractable analysis, only the single output (i.e. $\boldsymbol{C}_H^{\mathsf{T}} = \boldsymbol{C}_{CI_s}^{\mathsf{T}}$) is considered in this thesis. Note that the term single output refers to *output per channel*, i.e. for an ADC with L input channels, a single output configuration will give L outputs to the estimation filter.

For the case of multiple input channels, i.e. $L > 1$, we define $N = LN_\ell$, where N_ℓ is the order of the corresponding scalar input system. Due to the shape of the state-space matrices, we restrict both N_ℓ and L to be powers of 2. For $L > 1$, the state-space matrices generalizes as

$$\boldsymbol{L}_{N/2} = \begin{bmatrix} \boldsymbol{L}_{N_\ell/2} & & \\ & \ddots & \\ & & \boldsymbol{L}_{N_\ell/2} \end{bmatrix} \in \mathbb{R}^{\frac{N}{2}\times\frac{N}{2}}, \tag{5.18}$$

[1]It could be mentioned that the road to this parametrization was to start with the architecture, and subsequently extract the parametrization that describes the system.

$$\boldsymbol{B}_H = \begin{bmatrix} \boldsymbol{B}_{H_\ell} & & \\ & \ddots & \\ & & \boldsymbol{B}_{H_\ell} \end{bmatrix} \in \mathbb{R}^{N \times L} \tag{5.19}$$

and

$$\boldsymbol{C}_H = \begin{bmatrix} \boldsymbol{C}_{H_\ell} & & \\ & \ddots & \\ & & \boldsymbol{C}_{H_\ell} \end{bmatrix} \in \mathbb{R}^{N \times L}, \tag{5.20}$$

where subscript ℓ refers to the scalar input equivalent, defined by (5.12) and (5.13). With $N = LN_\ell$ and $\boldsymbol{L}_{N/2}$ as above, $\boldsymbol{A}_H$ is still given by (5.11).

For single output reconstruction, the transfer function is a column vector given by

$$\boldsymbol{G}(\omega) = \boldsymbol{C}_H^{\mathsf{T}} \left(j\omega \boldsymbol{I}_N - \boldsymbol{A}_H\right)^{-1} \boldsymbol{B}_H \tag{5.21}$$

where each element gives the transfer function of the corresponding input channel. It is shown in appendix C that all channels will experience the same transfer function, given by

$$G(\omega) = \left(\sqrt{\frac{N}{2}} \frac{\beta}{j\omega}\right)^{N_\ell}. \tag{5.22}$$

5.1.2 Proposed Hardware Architecture

The proposed hardware architecture for this system is shown in figure 5.1 with $N = 8$ and $L = 2$. In this figure, the integrators are abstracted out, and the choice of integrator topology is discussed subsequently. For this example, $\boldsymbol{A}_H$ is given by

$$\boldsymbol{A}_H = \boldsymbol{H}_8' \boldsymbol{A}' \tag{5.23}$$

$$= \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 \\ 1 & 1 & -1 & -1 & 0 & 0 & 0 & 0 \\ 1 & -1 & -1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 & 1 & -1 \\ 0 & 0 & 0 & 0 & 1 & 1 & -1 & -1 \\ 0 & 0 & 0 & 0 & 1 & -1 & -1 & 1 \end{pmatrix} \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \beta & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \beta & 0 \\ \beta & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \beta & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \beta & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \beta & 0 & 0 & 0 & 0 \end{pmatrix}. \tag{5.24}$$

In figure 5.1 color coding is used to distinguish between the input channels (green), the *physical* states (orange) and the control contributions (red).

The boxes labeled $\boldsymbol{H}_4(Z)$ contains the analog implementation of a 4-by-4 Hadamard matrix. The differential implementation enables an efficient realization of the ± 1 operation of the Hadamard matrix, by crossing/not crossing the wires. The implementation of the $H_4(Z)$ matrix is taken from [2], and shown in figure 5.2. The impedance Z will be resistive or capacitive depending on the choice of integrator topology.[2]

The two input channels u_0 and u_1 is connected to the first Hadamard matrix together with the states x_4 and x_6. The input to the first column of integrators will therefore be an orthogonal mixture of each of these signals. The outputs of these integrators are denoted x_0-x_3. Because of the state-space rotation achieved by $\boldsymbol{H}_4(Z)$, these physical states are separated from the logical ones. The control contributions s_0-s_3 is applied at the input of these integrators to bound their outputs.

These four states are then applied as an input to the second Hadamard matrix. As the Hadamard matrix is its own inverse up to a constant, cf. (5.3) and (5.4), this second Hadamard matrix rotates the state-space back to the origin. In consequence, the states x_4-x_7 lies in the "normal" state-space, and the logical states coincides with the physical ones at this point. This means that e.g. x_4 is the result of u_0 passing through two integrators, each being controlled by a digital feedback loop. To increase the system order, x_4 is fed back to the first Hadamard matrix, and the relation between x_5 and x_4 is the same as the relation between x_4 and u_0. The same holds for u_1, x_6 and x_7.

The system may therefore be understood as 4th order, multi-channel chain-of-integrators, where the signal from all input channels are distributed over the first column of integrators. As the Hadamard matrix is only defined for N being powers of two, both the number of channels, L, and the system order per channel, N_ℓ, is also restricted to being powers of two. Increasing N_ℓ from 4 to 6 would require a $\boldsymbol{H}_6$ matrix which is undefined. Increasing the number of channels to e.g. $L = 3$ could however be achieved by leaving the 4th input to the Hadamard matrix shortened.

5.1.3 Choice of integrators

For the implementation of the fully differential integrators, both opamp-RC and Gm-C integrators are considered. A schematic of both topologies are shown in figure 5.3 and 5.4 respectively. The Hadamard ADC is only simulated by solving the state-space equations (5.5) directly using an ODE solver

[2]Note here that the elements of the impedance matrix will be part of the integrator gain β. It is not the case that $\boldsymbol{H}_4(Z)$ implements $\boldsymbol{H}_4$ directly, while $\boldsymbol{A}'$ is implemented by the integrators alone. $\boldsymbol{A}_H = \boldsymbol{H}'_N\boldsymbol{A}'$ is implemented through a combination of the integrators and impedance matrices.

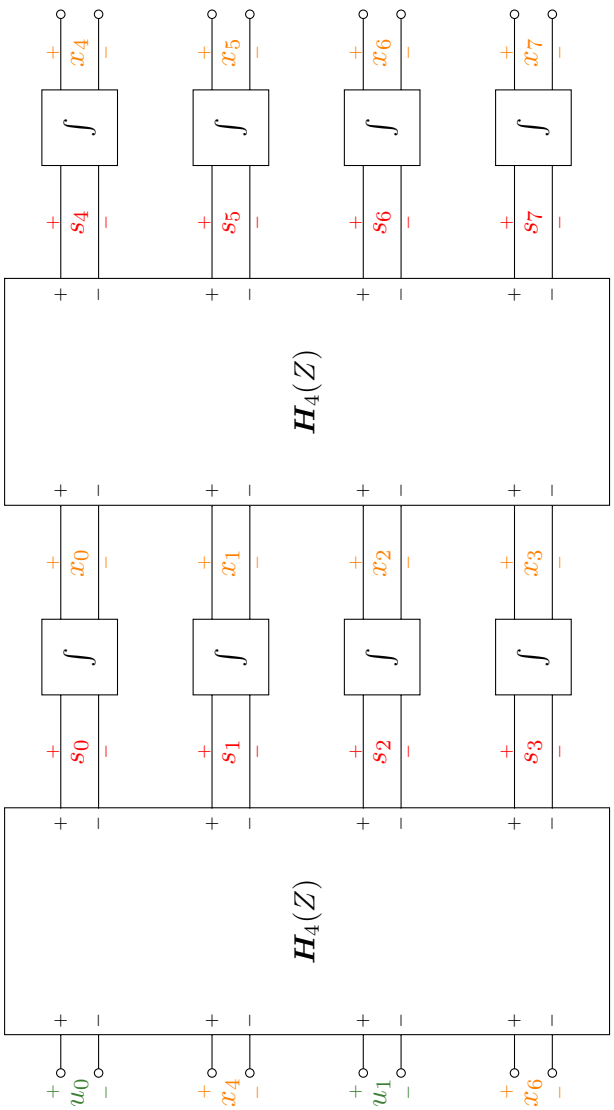

Figure 5.1: Proposed hardware architecture of the Hadamard analog system for $N = 8, L = 2$

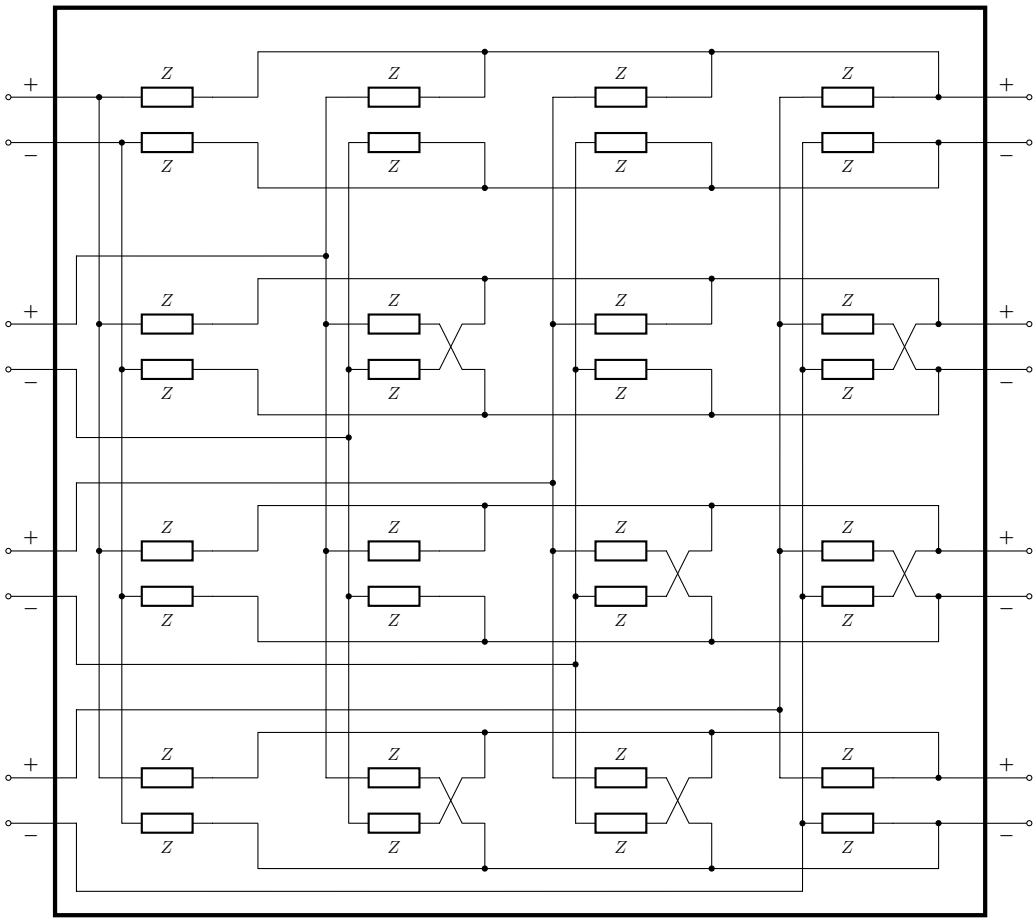

Figure 5.2: A 4th order Hadamard matrix implemented with impedance Z. Straight wires correspond to a multiplication of 1 and crossing wires to a multiplication of -1

library in Python, and the system is not simulated on circuit level. A discussion of the different integrator topologies is still included as background for future work.

The Opamp-RC Integrator

The fully differential opamp-RC integrator, shown in figure 5.3, integrates the inputs through an operational amplifier with capacitive feedback. This topology requires the Hadamard matrices to be implemented with resistors. The Hadamard matrices takes a voltage signal as input and deliver a current signal at the output. Referring to figure 5.2, each element of the matrix will perform a voltage to current conversion, with a potential sign change. The output currents from each element of the matrix is then summed at the integrators virtual ground.

From figure 5.3, the integrator gain is recognized as

$$\beta = \frac{1}{R_\beta C}. \tag{5.25}$$

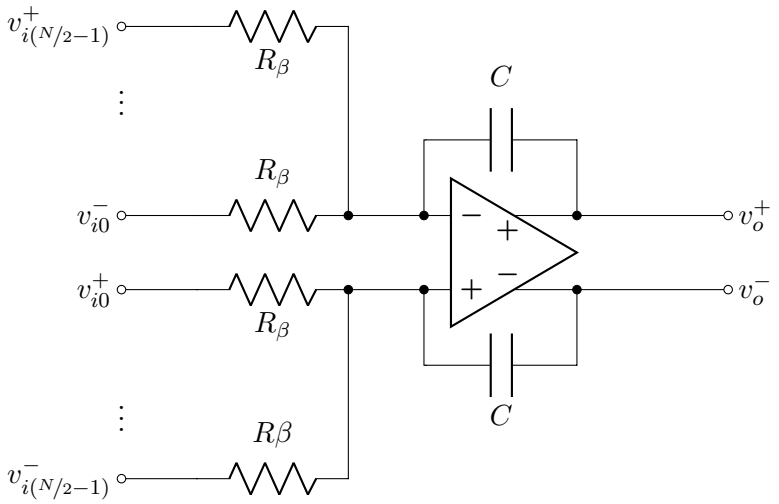

Figure 5.3: A fully differential, opamp-RC integrator with current summation at the input.

The Gm-C Integrator

The fully differential Gm-C integrator is shown in figure 5.4. This integrator is based on an open loop, operational transconductance amplifier (OTA). There is no virtual ground at the input, and hence no current summation is possible at this node. The proposed solution is to do voltage summation through a floating gate configuration. Hence the Gm-C integrator requires

a capacitive implementation of the Hadamard matrices. In this case, the Hadamard matrices operates with voltage signals at both the input and the output.

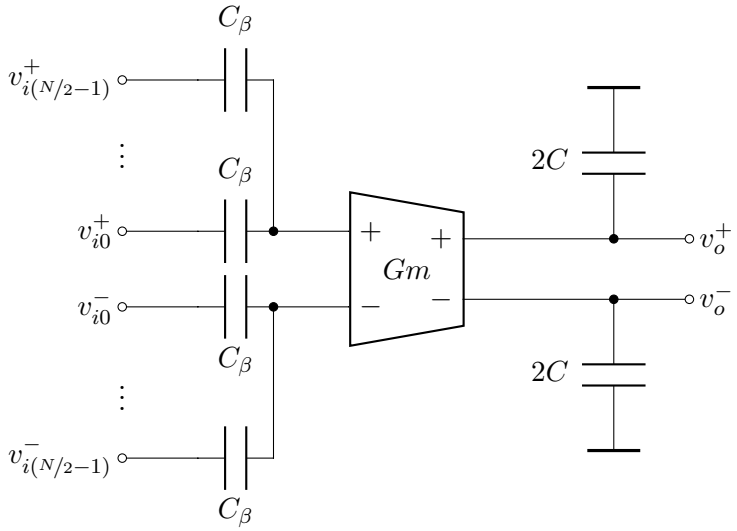

Figure 5.4: A fully differential, floating gate, Gm-C integrator with voltage summation on the input.

The transfer function analysis for this integrator is slightly more involved, as the additive operation is performed by a capacitive voltage division at the input of the OTA. For simplicity, the transfer function is analysed for the single ended equivalent of figure 5.4 and the result is extended to the fully differential case subsequently.

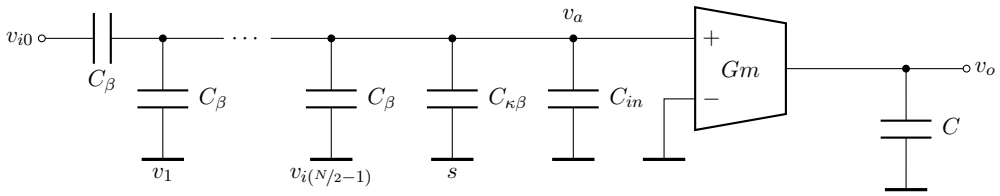

Figure 5.5: A single ended equivalent of the fully differential Gm-C integrator of figure 5.4 used for transfer function analysis.

The single ended equivalent of figure 5.4 is shown in figure 5.5. The capacitors C_β is part of the Hadamard matrix and $C_{\kappa\beta}$ is connected to the digital control contribution. The capacitor C_{in} is modelling the input capacitance of the OTA. In the following analysis, the transfer function from v_{i0} to v_o is found using the superposition principle, and both the control signal s and the other Hadamard inputs are grounded. Furthermore, the input v_{i0} is

assumed to behave as an ideal voltage source. From these assumptions, the OTA input, v_a, is given by

$$v_a = v_{i0} \left(\frac{C_\beta}{\frac{N}{2} C_\beta + C_{\kappa\beta} + C_{in}} \right). \tag{5.26}$$

Referring to figure 5.1, most of the integrators will be connected to another Hadamard matrix. Hence, the output voltage v_o will depend on the total capacitance on this node. To make the integrator gain independent of the subsequent Hadamard matrix, we choose $C \gg C_\beta$ and assume that the total capacitance on the output node is approximately equal to C. From this assumption, the integrator gain is recognized as

$$\beta = \frac{Gm}{C} \frac{C_\beta}{\frac{N}{2} C_\beta + C_{\kappa\beta} + C_{in}}. \tag{5.27}$$

5.1.4 Discussion of the Different Integrator Realizations

The two mentioned integrator topologies has very different properties and challenges. As neither of them are evaluated using an analog circuit simulator, e.g. SPICE, the following discussion is based general knowledge about these circuits, and is intended to provide a useful background for future work.

Load Impedance

A major disadvantage of the opamp-RC integrator is the need for resistors in the Hadamard matrices. The need to drive resistive loads, means that the amplifiers most likely will need two stages, and therefore consume more bias current. In addition, when determining the resistor values, there will be an unpleasant trade-off between current consumption and area.

For the Gm-C integrator the situation is the opposite. The transconductors are only driving capacitive loads, and it should be possible to use a power efficient, one-stage architecture for the implementation. The necessary transconductance, Gm, depends on the capacitive loading seen by the OTA. The OTAs used in the Gm-C integrators could therefore presumably be designed for a much lower current consumption than the opamps of the opamp-RC realization.

Linearity and Mismatch Sensitivity

For the opamp-RC integrator, the accuracy of β is depending directly on the values of the involved resistors and capacitors, which is difficult to control accurately in integrated CMOS processes. In order to set the integrator gain

with sufficient accuracy it might be necessary to trim the component values. The linearity of the integrator will however be good, due to to the negative feedback configuration.

The accuracy of the Gm-C time-constant depends on the matching of the transconductance to the capacitance seen at the output of the OTA. This ratio has an expected accuracy of about 30% over process variations [3] and a tuning circuitry would be required in order to control the integrator gain accurately. In addition, internal linearization techniques might be necessary as the open-loop OTA has no feedback linearization.

Input Signal Summation and Gate Leakage

The opamp-RC integrator performs current summation at the virtual ground while the Gm-C integrator uses floating gate voltage summation. With the considerable gate leakage in modern CMOS processes, the input node of the Gm-C integrator will need a periodic reset. The associated design challenges as well as the impact on the overall system performance needs further investigation.

The Effect of Multiple Inputs

From the analog transfer function of the proposed architecture, we see that the gain of the analog system is scaled up by a factor $\sqrt{\frac{N}{2}}$ relative to the chain-of-integrator system, cf. (5.22). This means that in order to achieve the same analog system gain, each integrator should be designed to be a factor $\sqrt{\frac{N}{2}}$ slower. This scaling is a natural consequence of applying the Hadamard transform, which combines multiple signals at the input of each integrator. As this increases the input signal to each integrator, each integrator should have reduced gain in order to maintain the same analog transfer function. This scaling effect has quite different implications on the two integrator topologies.

For the opamp-RC integrator, the integrator gain is given by $\frac{1}{R_\beta C}$ and could be reduced by increasing the resistor sizes. Driving larger resistors would enable a more energy efficient amplifier at the cost of increased area. At some point the required area will become too big, and it might not be possible to use this integrator in a system with a large number of input channels.

Reducing the gain of the Gm-C integrator could be done by reducing the transconductance of the OTA, which would reduce its power consumption directly. Assuming the Gm of the OTA is proportional to the bias current, this would imply a reduction in power consumption proportional to $\sqrt{\frac{N}{2}} = \sqrt{\frac{LN_\ell}{2}}$, which would give a considerable effect for large numbers of input

channels. This immediately seem to be a great benefit of having multiple input channels with the Gm-C integrator. However, from (5.27) we notice that β is proportional to $\frac{1}{N/2}$ due the capacitive voltage division at the input node of the integrator. This $\frac{1}{N/2}$ reduction from voltage division cancels the $\sqrt{\frac{N}{2}}$ gain from the Hadamard matrix. The net effect is, unfortunately, that the transconductance of the OTA must be increased with an increasing number of input channels.

In summary, from a system architecture point of view, the Gm-C realization seems the most promising in terms of power consumption and area. The design challenges associated with the floating gate operation under gate leakage needs to be investigated for the given technology. Both integrators have issues associated with the scaling to a large number of input channels and the optimum number of inputs will have to be determined by considering the overall system performance.

5.2 Digital Control

The previous section was concerned with the matrices $\boldsymbol{A}, \boldsymbol{B}$ and $\boldsymbol{C}^\mathsf{T}$, but $\boldsymbol{\Gamma}$ and $\tilde{\boldsymbol{\Gamma}}^\mathsf{T}$ still need to be determined. It is highlighted here that even though the control matrices determines the performance of the digital control, they are part of the *analog* system, cf. figure 3.3. In particular, $\tilde{\boldsymbol{\Gamma}}^\mathsf{T}$ determines how the digital control *observes* the internal states of the analog system, and could in general be any linear mapping. The same applies for $\boldsymbol{\Gamma}$, which determines how the control contribution $\boldsymbol{s}(t)$ *enters* the analog system.

It was pointed out in section 3.2 that the indirect relation between $\boldsymbol{s}[k]$ and $\boldsymbol{u}(t)$ enables more general interactions between the analog system and the digital control, compared to that of a $\Sigma\Delta$ modulator. The structure of the Hadamard ADC gives an example such an interaction that would not be possible in a $\Sigma\Delta$ ADC.

For the chain-of-integrator ADC in figure 4.1, each integrator has its own dedicated control loop. Hence, for this architecture, the relation between $\boldsymbol{s}[k]$ and $\boldsymbol{u}(t)$ is not very different from that of a $\Sigma\Delta$ modulator. This local digital control is described by $\tilde{\boldsymbol{\Gamma}}_{CI}^\mathsf{T} = \boldsymbol{I}_N$ and $\boldsymbol{\Gamma}_{CI} = \kappa\beta\tilde{\boldsymbol{\Gamma}}_{CI}^\mathsf{T}$, cf. equations (4.4) and (4.6). In general, the control matrices require an analog implementation, in the same way as $\boldsymbol{H}_4(Z)$ implements the Hadamard matrix in the analog system of figure 5.1.

5.2.1 A Peek Into the Design Space

There are an infinite number of possible choices for the control matrices. In the proposed analog system, some states lie in the physical space (x_4-x_7

in fig. 5.1) while other states are separated from the physical ones (x_0-x_3). The digital control may observe the states in one of these orientations, a mixture of the two or in a completely different orientation, depending on the choice of $\tilde{\mathbf{\Gamma}}^{\mathsf{T}}$. When there is a misalignment between the control and signal dimensions, each control signal has contribution to each signal dimension.

Furthermore, the $\mathbf{\Gamma}$-matrices need not be square. From section 3.3 $\mathbf{\Gamma}$ and $\tilde{\mathbf{\Gamma}}^{\mathsf{T}}$ are N-by-M and M-by-N matrices respectively. The number of controls, M, could be both larger and smaller than the number of states, N. Having more controls than states results in an *overcomplete control* as described in [2]. The $\tilde{\mathbf{\Gamma}}^{\mathsf{T}}$ matrix will in this case map the N-dimensional state signal to a higher dimension where the control decisions are being made. The M control contributions, $\boldsymbol{s}(t)$, are then applied to the analog system through the N-by-M $\mathbf{\Gamma}$-matrix. This solution was presented in [2] as an alternative to multi-bit quantizers, and is an effective way of tightening the bound while avoiding the challenges associated with multi-bit quantizers.

The opposite approach is to have $M < N$, resulting in an *undercomplete control*. This solution could be particularly interesting when the number of input channels, L, is huge. For a system with many input channels, having one control per state might give a tighter bound than what is necessary to reach the SNR requirements of the overall ADC. Hence reducing the number of controls, i.e. reducing the number of comparators in the circuit, might be a good way of reducing the overall power consumption.

In addition, the digital control need not be a static, time-independent system designed for general purpose A/D conversion. If the $\mathbf{\Gamma}$-matrices are implemented in a reconfigurable way, one could implement a digital control system that adapts to the statistical properties of the input signal, during operation. If the digital control has knowledge about the statistical properties of the input signal, this knowledge might be used to focus more of the control power on the dimensions that contains the most of the signal energy. This might enable a significantly tighter bound than what is possible in a general purpose system with no statistical information.

These ideas illustrates some of the possibilities within the design space of a control-bounded converter. Although there are most likely numerous implementation challenges to be discovered, these advanced features are all enabled by the control-bounded framework. These ideas are mentioned here to indicate possibilities for future development.

5.2.2 Proposed Control Implementation

In this work, three different choices of control matrices are considered. In all cases $\mathbf{\Gamma}_H = \kappa\beta\tilde{\mathbf{\Gamma}}_H^{\mathsf{T}}$ and the number of controls equals the number of states,

i.e. $M = N$. The different parametrizations differs in how the control dimensions align with the signal dimensions.

The simplest solution from an implementation point of view is to use local control for each integrator, i.e.

$$\boldsymbol{\Gamma}_H^l = \boldsymbol{I}_N. \tag{5.28}$$

In this case, the digital control observes the physical signal states and controls each of them independently. In the example of figure 5.1, this would mean that the control dimensions align with the signal dimension for x_4-x_7, while x_0-x_3 are observed in a rotated space, as these does not align with the physical dimensions.

In addition to local control, complete alignment and complete misalignment between control and signal dimensions are also considered. The former is achieved by Hadamard transforming the rotate states x_0-x_3, while leaving the remaining states unchanged. This operation is described by

$$\boldsymbol{\Gamma}_H^a = \begin{bmatrix} \sqrt{\frac{2}{N}}\boldsymbol{H}_{N/2} & \boldsymbol{0}_{N/2\times N/2} \\ \boldsymbol{0}_{N/2\times N/2} & \boldsymbol{I}_{N/2} \end{bmatrix}. \tag{5.29}$$

The scaling of $\sqrt{\frac{2}{N}}$ is included to normalize the $\boldsymbol{H}_{N/2}$ matrix.

The latter is achieved by transforming the unrotated states x_4-x_7, thereby observing all states in a misaligned dimension. In this case,

$$\boldsymbol{\Gamma}_H^m = \begin{bmatrix} \boldsymbol{I}_{N/2} & \boldsymbol{0}_{N/2\times N/2} \\ \boldsymbol{0}_{N/2\times N/2} & \sqrt{\frac{2}{N}}\boldsymbol{H}_{N/2} \end{bmatrix}. \tag{5.30}$$

5.3 Simulation Results

In contrast to the chain-of-integrators ADC, the proposed Hadamard ADC is not simulated on a circuit level. The system simulation is done by solving the state space equations (5.5) using an ODE solver library in Python. However, the obtained result should not be significantly different from a circuit simulation using ideal components. As the main goal of this work is to establish a starting point for a future transistor level implementation, simulating the state-space parametrization directly was considered a more effective way of exploring the properties of the different parametrizations. The reconstruction of the estimate $\hat{\boldsymbol{u}}(t)$ from $\boldsymbol{s}[k]$ is done using the same Python library as for the chain-of-integrators simulation. The only change to the estimation filter is that a different parametrization is specified.

The simulations presented in this section is based on the parameters summarized in table 5.1. The values for amplitude and state boundary are unitless because the mentioned system simulation is run without the concept

of impedance. In the presented simulations, the chain-of-integrator ADC is included for reference. The integrator gain for the chain-of-integrator is scaled up by $\sqrt{\frac{N}{2}}$, cf. 5.22, to make a fair comparison.

Table 5.1: Simulation parameters, proposed Hadamard ADC

Parameter	Symbol	Value	Comment
Critical frequency	f_c	10 MHz	
Unity gain frequency	f_u	20 MHz	
Sampling frequency	f_s	650 MHz	
Oversampling ratio	OSR $\left(\frac{f_s}{2f_c}\right)$	32.5	
System order	N	4	
Integrator gain	β	126 MHz	$\beta = 2\pi f_u$
Control gain	κ	1	
State boundary (input boundary)	$b_{\boldsymbol{x}}$ ($b_{\boldsymbol{u}}$)	1 (1)	
Input signal frequency	f_{in}	650 kHz	
Input signal amplitude	a_{in}	0.9FS	FS = $b_{\boldsymbol{u}}$

A simulation of the proposed architecture with the three different choices of control matrices is shown in figure 5.6. The figure shows the power spectral density (PSD) of the output estimate $\hat{\boldsymbol{u}}(t)$ for the different architectures. The simulation shows that the proposed ADC performs very similar to the chain-of-integrators from chapter 4. Especially the aligned control configuration is almost indistinguishable from the chain-of-integrators. As the proposed Hadamard ADC is essentially a chain-of-integrators with an intermediate state rotation, completely aligning the control and signal dimensions makes this configuration essentially equal to the chain-of-integrators in the ideal case. For the two other configurations, the performance is slightly different.

As the transfer function of the all four systems are identical, the only performance difference is (in this ideal case) caused by the digital controls ability to bound the state vector. We observe from this simulation that the misaligned control seem to perform slightly better than the other configurations.

As mentioned in the introduction to this chapter, one of the main motivations for the Hadamard ADC was to distribute the mismatch sensitivity of the involved components. A mismatch simulation is performed to see how the proposed ADC performs under component mismatch, compared to the chain-of-integrator. The simulation is done by using the same parameters as in the previous, ideal simulation. Before running the system simulation, i.e. solving (5.5), all elements of the matrices $\boldsymbol{A}, \boldsymbol{B}$ and $\boldsymbol{\Gamma}$ are offset from their ideal value. The deviation is drawn randomly from a normal distribution with a standard deviation of 1%. The reconstruction of $\hat{\boldsymbol{u}}(t)$ is ultimately

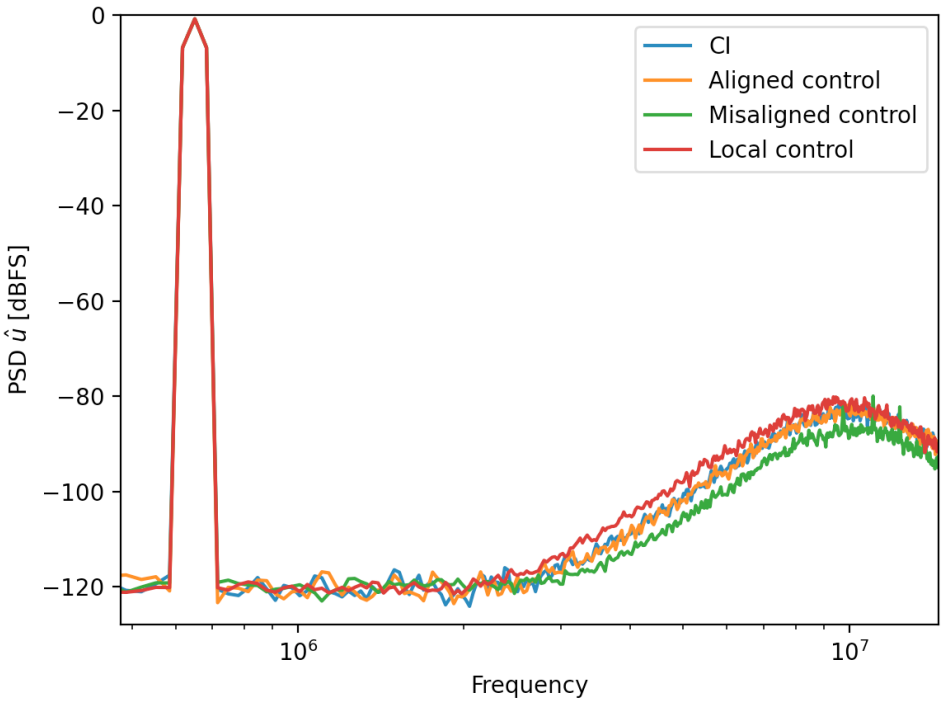

Figure 5.6: Simulation results of the proposed ADC architecture for $N = 4$. The three considered choices of control matrix are compared together with the chain-of-integrators (CI) system of the same order. The input is a single sinusoidal signal with a 90% full scale amplitude.

done using the nominal values. The presented results is obtained by averaging the output estimate of 10 simulation runs. Although this simple simulation does not completely resemble a realistic mismatch situation, it gives an indication of how the system is affected by non-ideal component values.

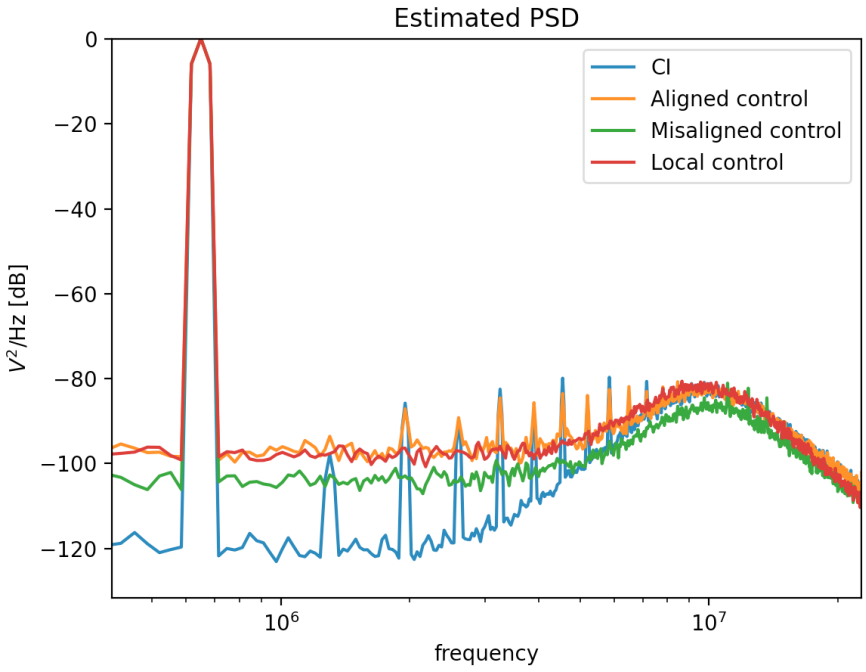

Figure 5.7: The PSD from a mismatch simulation, where all parameters drawn from normal distribution with std=1%. The different control configurations are compared together with the chain-of-integrator ADC for reference.

The estimated PSD from this simulation is shown in figure 5.7. First of all, we note that the power spectrum of the chain-of-integrators suffers from significant harmonics due to the introduced mismatch. Furthermore, the different Hadamard configurations all show an increased noise floor, and the amount of harmonics differs between the control configurations. The increased noise floor might be understood from the averaging effect of the Hadamard transform. As the logical signal is offset from the physical states, the mismatch causes a leakage between the signal dimensions which appears as an increased noise floor in the spectrum.

The magnitude of this noise floor as well as the amount of harmonics differ considerably between the three control configurations. The aligned control has almost the same amount of harmonics as the chain-of-integrator. The

alignment of control and signal dimensions causes mismatch in the control input matrix to affect the signal dimensions separately, thereby introducing harmonics in the spectrum. On the other hand, both the local and misaligned control show essentially no harmonics in the spectrum, which might be understood from the completely or partial misalignment between control and signal dimensions. Observe that the misaligned control produces a noise floor that is a few dB lower than that of the local control configuration. Whether this effect of turning harmonics into an increased noise floor is favorable will depend on the application.

Chapter 6

Final Discussions

This chapter provides some final discussions based on the results and experiences obtained from working with control-bounded ADC in general and the proposed Hadamard ADC architecture in particular. In this chapter, some discussions that did not fit naturally in the previous chapters are presented. There are also a lot of relevant topics that is beyond the scope of this thesis, and some of the limitations of the presented work is mentioned.

6.1 Additional Features of the control-bounded ADCs

6.1.1 Shared Analog State Space

It has been pointed out in section 5.2.1 that there is a significant potential in utilizing the shared control system of the Hadamard ADC. A shared control enables a more effective distribution of the available control resources, and hence a tighter state bound and increased performance.

In the multi-channel Hadamard ADC, the different input channels also shares the analog state-space. If there are an unequal distribution of the signal energy between the different input channels, combining them through the Hadamard matrix will have an averaging effect. More precisely, if the input signal is more concentrated in one of the physical dimensions than in one of the Hadamard dimensions, the inputs to the first column of integrators (i.e. x_0-x_3 of fig 5.1) will have a more even distribution of energy than the input channels of the ADC. This effect could be utilized in terms of lowering the state bound or alternatively increasing the gain of the analog system, compared to the situation where each input channel is converted independently. Formally, the analog system could be scaled more towards $\mathrm{E}\left[||\boldsymbol{u}(t)||_2\right]$, rather than $\mathrm{E}\left[||\boldsymbol{u}(t)||_\infty\right]$, which would be required when each channel lies in a separate state-space.

This effect is illustrated in figure 6.1. The illustration is meant to give a simplified picture of the situation when a pulse is coming towards a receiver array, as a plane wave with a steep angle. The receiver array has two elements, and due to the angle of incident, the pulse reaches the two elements at different times. The signal from the two elements are denoted $u_0(t)$ and $u_1(t)$ respectively. Through the Hadamard matrix, the input signals are combined in an orthogonal way. The resulting normalized outputs signals, denoted $\tilde{u}_0(t)$ and $\tilde{u}_1(t)$, are illustrated at the right side of the Hadamard matrix.

The pictured situation is the ideal application for a Hadamard ADC. At any time instance, the signal energy lies almost entirely in one of the physical dimensions, i.e. $u_0(t)$ or $u_1(t)$. Hence, the energy becomes evenly distributed between the outputs $\tilde{u}_0(t)$ and $\tilde{u}_1(t)$. The magnitude of each output is reduced by a factor $\sqrt{2}$ relative to the input, but the effect is barely visible for the illustrated situation of two channels. For a larger system with hundreds or thousands of input channels, a scaling of $\sqrt{L}$ would be significant. As mentioned, this reduced signal strength could be utilized in terms of increased gain or tighter control bound. From equation (3.28) we see that increasing the integrator gain, or lowering the state boundary, by a factor $\sqrt{L}$ would both give an SNR increasement proportional to L.

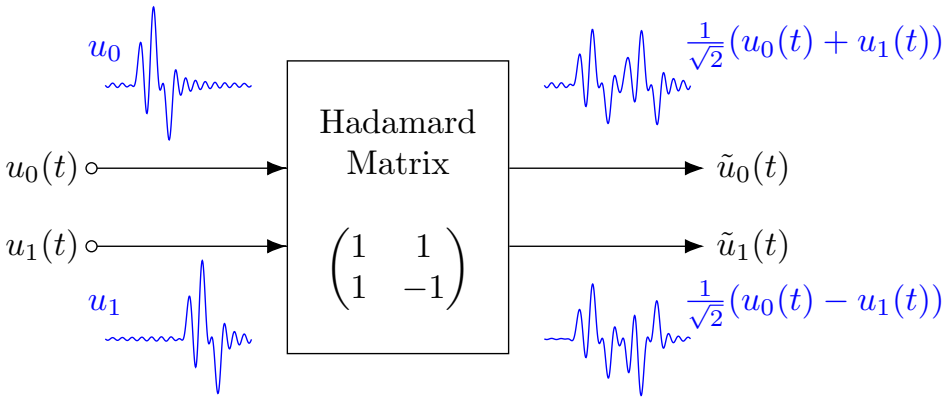

Figure 6.1: An illustration of a pulse arriving at two different receiver elements at different times. $\boldsymbol{u}(t) = (u_0(t), u_1(t))^\mathsf{T}$ is the input signal and $\tilde{\boldsymbol{u}}(t) = (\tilde{u}_0(t), \tilde{u}_1(t))^\mathsf{T}$ is the output signal of the Hadamard matrix.

Figure 6.2 pictures the opposite situation. In this case the pulse arrives at the two inputs almost simultaneously, and the signal energy lies almost entirely in the first Hadamard dimension. In consequence, almost all energy is concentrated at the first output of the Hadamard matrix, $\tilde{u}_0(t)$. The magnitude of $\tilde{u}_0(t)$ is therefore increased by $\sqrt{L}$ relative to the input, which would require a reduction in gain or state bound, and hence decreased SNR.

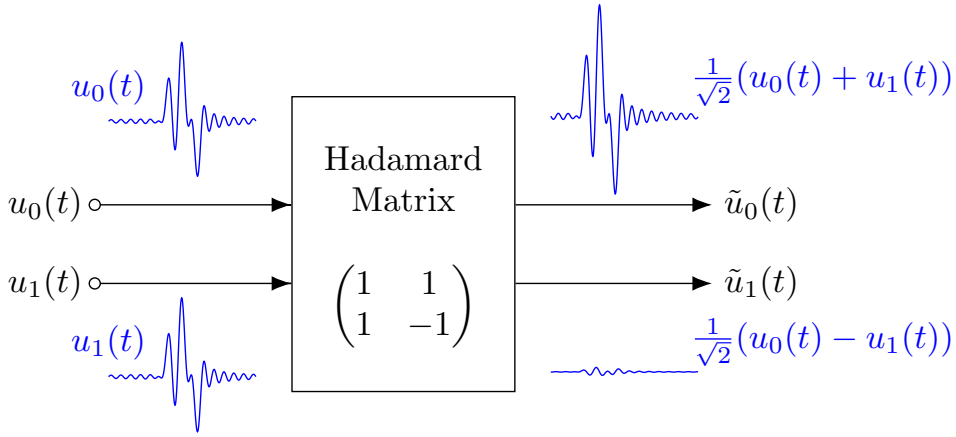

Figure 6.2: An illustration of a pulse arriving at two different receiver elements at the same time. $\boldsymbol{u}(t) = (u_0(t), u_1(t))^{\mathsf{T}}$ is the input signal and $\tilde{\boldsymbol{u}}(t) = (\tilde{u}_0(t), \tilde{u}_1(t))^{\mathsf{T}}$ is the output signal of the Hadamard matrix.

Any realistic situation would most likely be somewhere in between these two extremes. Whether combining the inputs in a Hadamard ADC gives a direct SNR improvement or not, will depend on which of the two extremes that is closest to the situation at hand. If we denote the output vector of the Hadamard matrix as $\tilde{\boldsymbol{u}}(t)$, the analog system will always have to be scaled towards $\mathrm{E}\left[||\tilde{\boldsymbol{u}}(t)||_\infty\right]$. The question of is wether or not $\mathrm{E}\left[||\tilde{\boldsymbol{u}}(t)||_\infty\right] < \mathrm{E}\left[||\boldsymbol{u}(t)||_\infty\right]$.

6.1.2 Shared Control Using a Chain-of-Integrators

In this thesis, the chain of integrators was presented with a local control only, as shown in figure 4.1. It is possible to misalign control and signal dimensions, also for an analog system that integrates the input in a chain structure. The $\boldsymbol{\Gamma}$-matrices could for instance be implemented as Hadamard matrices, hence rotating the control space, even though the states never share the same analog state space. In line with the discussion of section 5.2.1, it might also very well be that the Hadamard matrix is not the best choice for these matrices. Due to the orthogonality of the Hadamard transform, it could be considered a general purpose solution. If statistical properties of the input channels is known, this knowledge should be used to find the optimum control mapping.

Applications with signal properties closer to the situation pictured in figure 6.2 than that of figure 6.1 are not suited for using the Hadamard analog system. For these applications, combining an integrator chain with a shared digital control would be particularly attractive.

6.1.3 Absorbing the LNA Into the ADC

Systems receiving weak analog information signals, often use a low noise amplifier (LNA) to amplify the signal before A/D conversion. When using a control-bounded converter, there should ideally not be any components before the ADC. The overall system performance achievable by including an LNA into the ADC, versus keeping it on the outside, is not evaluated. However, it would be a surprising result if separating the two components showed to be the most attractive. We propose as a general rule of thumb, that one should aim to include as much of the necessary functionality into the ADC as possible, as this will most likely reduce the requirements on the involved components. However, the associated design complexity and necessary implementation time must also be considered in the trade-off.

In the presented work, the control gain κ is not given much attention. In an application that normally uses an LNA, absorbing it into the ADC would give the ADC a much weaker input signal. This is prior knowledge of $b_{\boldsymbol{u}}$ that should be utilized. In this case, the control gain κ should probably not be equal for all control signals. In an integrator chain for instance, the first integrator will have a weak input signal. Hence, this integrator could have been stabilized with less control gain, which in turn allows a higher β. This scaling would give a direct performance increasement, cf. (3.28).

6.1.4 Comparator Offset Voltage

Another advantage of control-bounded ADCs that should be mentioned, is the sensitivity to offset voltage in the comparators. As highlighted several times, the only information about the control signals that is used by the digital estimator is that this was the signal needed to stabilize the internal states of the analog system. The digital estimator do not care about how these control signals relate to the input or the states of the analog system. Note that $\tilde{\boldsymbol{\Gamma}}$ is not used by the DE in figure 3.4. This means that the output estimate is completely invariant to the offset voltage of the comparators, as long as the offset voltage is not disabling the digital control from bounding the state vector. Is this interesting?

6.1.5 Noise in the Hadamard System

An important non-ideality that is not treated in this thesis is the noise generated by the components of the analog system. When the signal is integrated in a chain structure, the first integrator will be the most critical and most of the power budget will be consumed by this integrator in order to keep its noise contribution according to the requirement. In the Hadamard ADC however, all integrators in the first column will contribute equally to the total output noise. It was shown in [2] that it is possible tune the situation

by amplifying the different signal dimensions unequally. By increasing the gain of one/some signal dimensions, and reducing it on others, the noise contribution is distributed unequally between the integrators. This however comes at an expense of reduced nominal performance.

The input node of the system is critical, as this is where the information carrying input signal is at its weakest. The vulnerable input signal should be handled with care, and connecting more active components to this node does not intuitively sound beneficial. However, when designing the active components, trade-offs will have to be made between noise, speed, linearity, etc., and other design specification might influence the cost of reaching a certain noise requirement. A thorough noise analysis of the different architectures remains for future work.

6.2 Limitations of the Presented Work

Some limitations of the presented work is discussed in this section.

6.2.1 Filter Complexity

The work of this thesis is mainly concerned with the analog system and the digital control of the proposed architecture. The simulations are performed using an offline estimation filter implemented in Python. An efficient, online filter algorithm is presented in [2], but is not tested in this work. The power consumed by the estimation filter will of course count on the total power budget of the ADC, and should therefore be taken into account when evaluating the overall system performance.

For larger systems with several input channels, it might be more efficient to run the algorithm on a micro-controller, rather than using a dedicated HDL implementation of the filter algorithm. These are interesting questions that remains for future research.

6.2.2 Complex Poles and Optimized Transfer Function

In this thesis, the analog transfer function of both the chain-of-integrators and the Hadamard system constitutes a chain of first order integrations. However, in a practical realization it would be favorable to have more advanced transfer functions in the analog system. Adding zeros in the transfer function enables band-pass and notch filter realizations, and complex pole pairs enables sharper transitions between pass-band and stop-band. This way, more of the analog system gain may be concentrated in the frequency band of interest, resulting in an overall performance increasement. Optimizing the noise transfer function is an important part of the design of $\Sigma\Delta$ modulators and should be done in control-bounded ADCs as well.

It was shown in [2] how a chain-of-integrator-like structure can be modified to allow for any general N-th order transfer function polynomial. Poles are introduces to the transfer function by connecting the different states through additional feedback loops, and zeros by feed-forward paths from the input. The same general transfer function is also achievable for the Hadamard system by using a similar structure.

Evaluating the proposed architecture with an optimized transfer function is unfortunately beyond the scope of this thesis. The main focus has been to find and evaluate an efficient realization of the Hadamard analog system, and optimized transfer functions is kept in mind as an important part of future development.

6.2.3 Quantitative Measures

The simulations presented in this thesis are only meant to give a qualitative indication of the performance of the different architectures. When designing for a specific application one would want to find the optimum filter order, OSR, integrator gain, state boundary etc., in order to reach the performance requirements with the lowest possible power consumption.

The simulations performed in this thesis are highly ideal, and no attempt has been made to do such optimizations. These optimizations will be considered together with transistor level implementations in the given technology and the goal of this thesis is to provide a useful background for the next part of the design process.

Chapter 7

Conclusion and future progress

This thesis have explored how control-bounded A/D converters could be used advantageously for receiver systems with multiple input channels. A thorough introduction to the control-bounded conversion concept was given, which provided the necessary background for understanding the considered architectures. The simple chain-of-integrator ADC was presented as a soft introduction to what a control-bounded ADC could be. In addition, several useful results obtained from its analysis also applies to other architectures.

The main focus of the thesis has been the proposed Hadamard architecture, which was based on the architecture presented in [2]. The proposed architecture is very similar to the existing one, but allows for a more power efficient implementation due to the absence of buffers in the analog system. Some design challenges associated with two different integrator topologies are also considered.

In contrast to the chain-of-integrator, The Hadamard ADC utilizes some of the flexibility within the control-bounded conversion framework. Important features of this architecture include the tolerance to component mismatch and the potential SNR gain by "averaging" the input channels through a Hadamard matrix. In addition, the ability to share the control resources between the different states has a great potential for future development.

These benefits does however not come without limitations. First of all, realizing an analog implementation of the Hadamard network takes area and has a negative impact on the design of integrators. When considering mismatch sensitivity, harmonics in the spectrum are turned into an increased noise floor. Whether or not this is beneficial will depend on the application. The averaging effect obtained from the Hadamard transform has a huge poten-

tial for SNR increasement for systems with large number of input channels. This requires however, that the signal energy has a sufficiently uneven distribution among the input channels. The potential SNR gain will therefore depend on the application at hand.

The shared control could be implemented with any analog system, and might provide a significant performance increasement for applications where the Hadamard analog system is not suited. The potential for implementing an adaptive control system that learns statistical properties of the input signal during operation, is highlighted as a particularly interesting feature.

In summary, we have seen that the control-bounded conversion framework enables interesting ways of tailoring the ADC design for a specific application. In a multi-channel receiver system, placing several independent, general purpose ADCs side-by-side, is not likely the best way of capturing the incoming information content. With the control-bounded converter, the tools for utilizing prior knowledge of the input signal does now exist. We therefore argue, that when designing a receiver system architecture, stepping away from the conventional view on A/D conversion should at least be considered.

Future progress

As mentioned in the introduction, the next step of this project is to realize a circuit implementation of a control-bounded ADC, optimized for medical ultrasound imaging. Some key specifications for this application was listed in table 1.1. Based on the discussions in this thesis, we therefore have to choose which architecture to go for in the next stage.

As seen in table 1.1, the only distortion requirement listed for this application is the magnitude of the second harmonic, and the other harmonics are considered less significant. This points towards that trading reduced harmonics for increased noise floor in the Hadamard ADC, might not be beneficial for this application.

The medical ultrasound probes used for 3D imaging has several thousand receivers. In the best case scenario of figure 6.1, the averaging effect of the Hadamard transform would give a direct SNR improvement of about 30dB. However the statistical properties of the input channels are not well understood by the author. A simple analysis of experimental raw data has been performed, but as the 3D ultrasound systems uses the mentioned SAP beamforming, the true raw data from the individual receivers are somewhat hidden from the output of the ADCs. The required knowledge to decide on this topic is therefore not present at the moment.

Also considering the increased design challenges associated with the Hadamard

system, the current plan is to implement a chain-based structure in the next step. This will be combined with some kind of shared control system, but exactly what features to include in this first realization is not yet decided. One potentially low-hanging fruit, is to pick only a subset of the available control dimensions, resulting in an undercomplete control. If the number of active comparators could be reduced significantly below the number of states, this might give a considerable reduction in overall power consumption. Furthermore, we will look for an efficient way of including the LNA into the ADC system.

It will be exiting to see how the total power consumption of this implementation compares to already existing solutions.

Acknowledgement

I would particularly like to thank Hampus Malmberg for the contributions to this work. He has provided invaluable assistance on debugging and implementation of the required simulation framework. I would also like to thank him for interesting and inspiring discussions. Most of the ideas presented in this thesis does somehow originate from a discussion with Hampus.

I would also like to thank my supervisors Trond Ytterdal and Carsten Wulff for great support and feedback along the way.

Appendix A

General Transfer Function of the Analog System

In this appendix presents the derivation of the general transfer function of the analog system. From (3.1) the frequency domain relation between the input and the state vector is obtained as

$$j\omega \boldsymbol{X}(\omega) = \boldsymbol{A}\boldsymbol{X}(\omega) + \boldsymbol{B}\boldsymbol{U}(\omega) \tag{A.1}$$

$$(j\omega \boldsymbol{I}_N - \boldsymbol{A})\, \boldsymbol{X}(\omega) = \boldsymbol{B}\boldsymbol{U}(\omega) \tag{A.2}$$

$$\boldsymbol{X}(\omega) = (j\omega \boldsymbol{I}_N - \boldsymbol{A})^{-1}\, \boldsymbol{B}\boldsymbol{U}(\omega). \tag{A.3}$$

The output vector is obtained by multiplying the state vector with the signal signal observation matrix, $\boldsymbol{C}^{\mathsf{T}}$. Hence

$$\boldsymbol{Y}(j\omega) = \boldsymbol{C}^{\mathsf{T}}\, (j\omega \boldsymbol{I}_N - \boldsymbol{A})^{-1}\, \boldsymbol{B}\boldsymbol{U}(j\omega) \tag{A.4}$$

and we recognize the ATF as

$$\boldsymbol{G}(\omega) = \boldsymbol{C}^{\mathsf{T}}\, (j\omega \boldsymbol{I}_N - \boldsymbol{A})^{-1}\, \boldsymbol{B}. \tag{A.5}$$

Appendix B

Description of the Estimation Filter Algorithm

This appendix provides a short and concise description of the estimation filter algorithm.

The algorithm consist of a forward recursion

$$\overrightarrow{\boldsymbol{m}}_{k+1} \triangleq \boldsymbol{A}_f \overrightarrow{\boldsymbol{m}}_k + \boldsymbol{B}_f \boldsymbol{s}[k], \tag{B.1}$$

a backward recursion

$$\overleftarrow{\boldsymbol{m}}_{k-1} \triangleq \boldsymbol{A}_b \overleftarrow{\boldsymbol{m}}_k + \boldsymbol{B}_b \boldsymbol{s}[k-1], \tag{B.2}$$

and finally the estimate

$$\hat{\boldsymbol{u}}(t_k) \triangleq \boldsymbol{W}^{\mathsf{T}} (\overleftarrow{\boldsymbol{m}}_k - \overrightarrow{\boldsymbol{m}}_k). \tag{B.3}$$

The matrices $\boldsymbol{A}_f, \boldsymbol{A}_b, \boldsymbol{B}_f, \boldsymbol{B}_b$ and $\boldsymbol{W}$ is computed offline, and is given by the following equations.

$$\boldsymbol{A}_f \triangleq \exp\left(\left(\boldsymbol{A} - \frac{1}{\eta^2}\overrightarrow{\boldsymbol{V}}\right) T\right) \tag{B.4}$$

$$\boldsymbol{A}_b \triangleq \exp\left(-\left(\boldsymbol{A} + \frac{1}{\eta^2}\overleftarrow{\boldsymbol{V}}\right) T\right) \tag{B.5}$$

$$\boldsymbol{B}_f \triangleq \int_0^T \exp\left(\left(\boldsymbol{A} - \frac{1}{\eta^2}\overrightarrow{\boldsymbol{V}}\right)(T-\tau)\right) \boldsymbol{\Gamma} d\tau \tag{B.6}$$

$$\boldsymbol{B}_b \triangleq -\int_0^T \exp\left(-\left(\boldsymbol{A} + \frac{1}{\eta^2}\overleftarrow{\boldsymbol{V}}\right)(T-\tau)\right) \boldsymbol{\Gamma} d\tau \tag{B.7}$$

In equations (B.4 - B.7), exp(.) denotes the matrix exponential, which is not to be confused with the element-wise exponential operation.

The matrices $\overrightarrow{\boldsymbol{V}}$ and $\overleftarrow{\boldsymbol{V}}$ used in (B.4 - B.7) is obtained by solving the continuous-time algebraic Riccati (CARE) equations

$$\boldsymbol{A}\overrightarrow{\boldsymbol{V}} + \left(\boldsymbol{A}\overrightarrow{\boldsymbol{V}}\right)^{\mathsf{T}} + \boldsymbol{B}\boldsymbol{B}^{\mathsf{T}} - \frac{1}{\eta^2}\overrightarrow{\boldsymbol{V}}\boldsymbol{C}^{\mathsf{T}}\boldsymbol{C}\overrightarrow{\boldsymbol{V}} = \boldsymbol{0}_{N\times N} \tag{B.8}$$

and

$$\boldsymbol{A}\overleftarrow{\boldsymbol{V}} + \left(\boldsymbol{A}\overleftarrow{\boldsymbol{V}}\right)^{\mathsf{T}} - \boldsymbol{B}\boldsymbol{B}^{\mathsf{T}} + \frac{1}{\eta^2}\overleftarrow{\boldsymbol{V}}\boldsymbol{C}^{\mathsf{T}}\boldsymbol{C}\overleftarrow{\boldsymbol{V}} = \boldsymbol{0}_{N\times N}. \tag{B.9}$$

The matrix $\boldsymbol{W}$ is finally obtained by solving the linear equation system

$$\left(\overrightarrow{\boldsymbol{V}} + \overleftarrow{\boldsymbol{V}}\right)\boldsymbol{W} = \boldsymbol{B}. \tag{B.10}$$

Appendix C

Transfer Function Analysis of Proposed Architecture

This appendix contains the derivation of the transfer function of the AS described by equations 5.11, 5.12 and 5.13. It will become apparent that for multiple inputs, i.e. $L > 1$, the transfer function decouples into a set of identical expressions. For this analysis, we first consider the case of single input and then show how the obtained results generalizes for the multiple input case. For a single input, the transfer function is a scalar given by

$$G(\omega) = \boldsymbol{C}_H^{\mathsf{T}} \left(j\omega \boldsymbol{I}_N - \boldsymbol{A}_H\right)^{-1} \boldsymbol{B}_H \tag{C.1}$$

$$= \boldsymbol{C}_{CI_s}^{\mathsf{T}} \left(j\omega \boldsymbol{I}_N - \boldsymbol{H}_N' \boldsymbol{A}'\right)^{-1} \boldsymbol{H}_N' \boldsymbol{B}_{CI} \tag{C.2}$$

$$= \boldsymbol{C}_{CI_s}^{\mathsf{T}} \boldsymbol{M}^{-1} \boldsymbol{H}_N' \boldsymbol{B}_{CI}, \tag{C.3}$$

where we have defined $\boldsymbol{M} \triangleq (j\omega \boldsymbol{I}_N - \boldsymbol{H}_N' \boldsymbol{A}') \in \mathbb{R}^{N \times N}$. To obtain an analytic expression for the transfer function, we first need a closed form expression for the inverse of this matrix. Referring to equations 5.15 and 5.16, the matrix $\boldsymbol{M}$ can be expressed in block form as

$$\boldsymbol{M} = \begin{bmatrix} \boldsymbol{M}_{11} & \boldsymbol{M}_{21} \\ \boldsymbol{M}_{12} & \boldsymbol{M}_{22} \end{bmatrix} = \begin{bmatrix} j\omega \boldsymbol{I}_{N/2} & -\beta \boldsymbol{H}_{N/2} \boldsymbol{L}_{N/2} \\ -\beta \boldsymbol{H}_{N/2} & j\omega \boldsymbol{I}_{N/2} \end{bmatrix}. \tag{C.4}$$

The general inverse of a block matrix can be expressed using

$$\boldsymbol{D}_1 = \boldsymbol{M}_{11} - \boldsymbol{M}_{12} \boldsymbol{M}_{22}^{-1} \boldsymbol{M}_{21} \tag{C.5}$$

and

$$\boldsymbol{D}_2 = \boldsymbol{M}_{22} - \boldsymbol{M}_{21} \boldsymbol{M}_{11}^{-1} \boldsymbol{M}_{12} \tag{C.6}$$

as

$$\boldsymbol{M}^{-1} = \begin{bmatrix} \boldsymbol{D}_1^{-1} & -\boldsymbol{M}_{11}^{-1} \boldsymbol{M}_{12} \boldsymbol{D}_2^{-1} \\ -\boldsymbol{D}_2^{-1} \boldsymbol{M}_{21} \boldsymbol{M}_{11}^{-1} & \boldsymbol{D}_2^{-1} \end{bmatrix}. \tag{C.7}$$

For this particular matrix, $\boldsymbol{D}_1$ and $\boldsymbol{D}_2$ coincide as

$$\boldsymbol{D} = j\omega \boldsymbol{I}_{N/2} - \frac{\beta^2}{j\omega}\frac{N}{2}\boldsymbol{L}_{N/2} \tag{C.8}$$

and the inverse of $\boldsymbol{M}$ can then be written as

$$\boldsymbol{M}^{-1} = \begin{bmatrix} \boldsymbol{D}^{-1} & \frac{\beta}{j\omega}\boldsymbol{H}_{N/2}\boldsymbol{L}_{N/2}\boldsymbol{D}^{-1} \\ \frac{\beta}{j\omega}\boldsymbol{D}^{-1}\boldsymbol{H}_{N/2} & \boldsymbol{D}^{-1} \end{bmatrix}. \tag{C.9}$$

It remains to find an expression for $\boldsymbol{D}^{-1}$. We first define the parameter $\psi \triangleq \frac{\beta^2}{(j\omega)^2}\frac{N}{2}$ and write

$$\boldsymbol{D} = j\omega\left(\boldsymbol{I}_{N/2} - \psi\boldsymbol{L}_{N/2}\right). \tag{C.10}$$

By (5.17), the matrix $\boldsymbol{L}_{N/2}$ is strictly lower triangular which implies that $\boldsymbol{L}_{N/2}^k = \boldsymbol{0}$ for $k \geq \frac{N}{2}$. We can therefore express $\boldsymbol{D}^{-1}$ by the Neumann series (generalized geometric series) of $\boldsymbol{L}_{N/2}$ as

$$\boldsymbol{D}^{-1} = \frac{1}{j\omega}\left(\boldsymbol{I}_{N/2} - \psi\boldsymbol{L}_{N/2}\right)^{-1} \tag{C.11}$$

$$= \frac{1}{j\omega}\sum_{k=0}^{\infty}\psi^k\boldsymbol{L}_{N/2}^k \tag{C.12}$$

$$= \frac{1}{j\omega}\left(\boldsymbol{I}_{N/2} + \sum_{k=1}^{\frac{N}{2}-1}\psi^k\boldsymbol{L}_{N/2}^k\right). \tag{C.13}$$

Because of the shape of $\boldsymbol{L}_{N/2}$, the matrix $\left(\boldsymbol{I}_{N/2} + \sum_{k=1}^{\frac{N}{2}-1}\psi^k\boldsymbol{L}_{N/2}^k\right)$ will have the form

$$\left(\boldsymbol{I}_{N/2} + \sum_{k=1}^{\frac{N}{2}-1}\psi^k\boldsymbol{L}_{N/2}^k\right) = \begin{pmatrix} 1 & 0 & & & \\ \psi & 1 & 0 & & \\ \psi^2 & \psi & 1 & 0 & \\ \vdots & & \ddots & \ddots & 0 \\ \psi^{\frac{N}{2}-1} & \cdots & & \psi & 1 \end{pmatrix} \tag{C.14}$$

To obtain a compact expression, we introduce the vector

$$\boldsymbol{\psi}_i^M \triangleq \begin{pmatrix} \boldsymbol{0}_i \\ \psi^0 \\ \psi \\ \vdots \\ \psi^{M-1-i} \end{pmatrix}, \tag{C.15}$$

where $\mathbf{0}_i \in \mathbb{R}^i$ is an all-zero column vector of length i. Using this vector, we write

$$\left(\boldsymbol{I}_{N/2} + \sum_{k=1}^{\frac{N}{2}-1} \psi^k \boldsymbol{L}_{N/2}^k \right) = \boldsymbol{\Psi}_{N/2} \tag{C.16}$$

where

$$\boldsymbol{\Psi}_{N/2} \triangleq \begin{bmatrix} \boldsymbol{\psi}_0^{N/2} & \cdots & \boldsymbol{\psi}_{N/2-1}^{N/2} \end{bmatrix}. \tag{C.17}$$

After the introduction of this helper matrix, we can write the expression for the transfer function as

$$G(\omega) = \boldsymbol{C}_{CI_s}^{\mathsf{T}} \begin{bmatrix} \boldsymbol{D}^{-1} & \frac{\beta}{j\omega} \boldsymbol{H}_{N/2} \boldsymbol{L}_{N/2} \boldsymbol{D}^{-1} \\ \frac{\beta}{j\omega} \boldsymbol{D}^{-1} \boldsymbol{H}_{N/2} & \boldsymbol{D}^{-1} \end{bmatrix} \boldsymbol{H}_N' \boldsymbol{B}_H \tag{C.18}$$

$$= \boldsymbol{C}_{CI_s}^{\mathsf{T}} \begin{bmatrix} \frac{1}{j\omega} \boldsymbol{\Psi}_{N/2} & \frac{\beta}{(j\omega)^2} \boldsymbol{H}_{N/2} \boldsymbol{L}_{N/2} \boldsymbol{\Psi}_{N/2} \\ \frac{\beta}{(j\omega)^2} \boldsymbol{\Psi}_{N/2} \boldsymbol{H}_{N/2} & \frac{1}{j\omega} \boldsymbol{\Psi}_{N/2} \end{bmatrix} \boldsymbol{H}_N' \boldsymbol{B}_H \tag{C.19}$$

Before proceeding, we recognize the following. As

$$\boldsymbol{B}_{CI} = \begin{pmatrix} \beta & 0 & \cdots & 0 \end{pmatrix}^{\mathsf{T}} \in \mathbb{R}^{N \times 1} \tag{C.20}$$

we get

$$\boldsymbol{H}_N' \boldsymbol{B}_{CI} = \beta \begin{bmatrix} \mathbf{1}_{N/2} \\ \mathbf{0}_{N/2} \end{bmatrix}. \tag{C.21}$$

Together with $\boldsymbol{C}^{\mathsf{T}} = \begin{pmatrix} 0 & \cdots & 1 \end{pmatrix}$ we see that only $(\boldsymbol{M}^{-1})_{11} = \frac{\beta}{(j\omega)^2} \boldsymbol{\Psi}_{N/2} \boldsymbol{H}_{N/2}$ will influence the transfer function, and we can write

$$G(\omega) = \begin{pmatrix} 0 & \cdots & 1 \end{pmatrix} \frac{\beta}{(j\omega)^2} \boldsymbol{\Psi}_{N/2} \boldsymbol{H}_{N/2} \beta \mathbf{1}_{N/2} \tag{C.22}$$

$$= \frac{\beta^2}{(j\omega)^2} \begin{pmatrix} 0 & \cdots & 1 \end{pmatrix} \boldsymbol{\Psi}_{N/2} \boldsymbol{H}_{N/2} \mathbf{1}_{N/2} \tag{C.23}$$

The matrix product $\boldsymbol{\Psi}_{N/2} \boldsymbol{H}_{N/2} \mathbf{1}_{N/2}$ can be analyzed recursively as

$$\boldsymbol{\Psi}_{N/2} \boldsymbol{H}_{N/2} \mathbf{1}_{N/2} = \begin{bmatrix} \boldsymbol{\Psi}_{N/4} & \mathbf{0}_{N/4 \times N/4} \\ \hat{\boldsymbol{\Psi}}_{N/4} & \boldsymbol{\Psi}_{N/4} \end{bmatrix} \begin{bmatrix} \boldsymbol{H}_{N/4} & \boldsymbol{H}_{N/4} \\ \boldsymbol{H}_{N/4} & -\boldsymbol{H}_{N/4} \end{bmatrix} \mathbf{1}_{N/2} \tag{C.24}$$

$$= 2 \begin{bmatrix} \boldsymbol{\Psi}_{N/4} \boldsymbol{H}_{N/4} \mathbf{1}_{N/4} \\ \hat{\boldsymbol{\Psi}}_{N/4} \boldsymbol{H}_{N/4} \mathbf{1}_{N/4} \end{bmatrix}, \tag{C.25}$$

where

$$\hat{\boldsymbol{\Psi}}_{N/4} \triangleq \begin{bmatrix} \psi^{N/4} \boldsymbol{\psi}_0 & \cdots & \psi \boldsymbol{\psi}_0 \end{bmatrix}. \tag{C.26}$$

Furthermore,

$$\hat{\boldsymbol{\Psi}}_{N/4}\boldsymbol{H}_{N/4}\mathbf{1}_{N/4} = \begin{bmatrix} \psi^{N/8}\hat{\boldsymbol{\Psi}}_{N/8} & \hat{\boldsymbol{\Psi}}_{N/8} \\ \psi^{N/4}\hat{\boldsymbol{\Psi}}_{N/8} & \psi^{N/8}\hat{\boldsymbol{\Psi}}_{N/8} \end{bmatrix} \begin{bmatrix} \boldsymbol{H}_{N/8} & \boldsymbol{H}_{N/8} \\ \boldsymbol{H}_{N/8} & -\boldsymbol{H}_{N/8} \end{bmatrix} \mathbf{1}_{N/2} \tag{C.27}$$

$$= 2\begin{bmatrix} \psi^{N/8}\hat{\boldsymbol{\Psi}}_{N/8}\boldsymbol{H}_{N/8}\mathbf{1}_{N/8} \\ \psi^{N/4}\hat{\boldsymbol{\Psi}}_{N/8}\boldsymbol{H}_{N/8}\mathbf{1}_{N/8} \end{bmatrix}. \tag{C.28}$$

Starting at

$$\hat{\boldsymbol{\Psi}}_1\boldsymbol{H}_1\mathbf{1}_1 = \psi, \tag{C.29}$$

these recursive expressions may be combined to give

$$\boldsymbol{\Psi}_{N/2}\boldsymbol{H}_{N/2}\mathbf{1}_{N/2} = \frac{N}{2}\boldsymbol{\psi}_0^{N/2}. \tag{C.30}$$

Finally, the transfer function is given by

$$G(\omega) = (0 \ \cdots \ 1)\,\frac{\beta}{(j\omega)^2}\boldsymbol{\Psi}_{N/2}\boldsymbol{H}_{N/2}\beta\mathbf{1}_{N/2} \tag{C.31}$$

$$= \frac{\beta^2}{(j\omega)^2}\frac{N}{2}(0 \ \cdots \ 1)\,\boldsymbol{\psi}_0^{N/2} \tag{C.32}$$

$$= \frac{\beta^2}{(j\omega)^2}\frac{N}{2}\psi^{N/2-1} \tag{C.33}$$

$$= \left(\frac{\beta^2}{(j\omega)^2}\frac{N}{2}\right)^{N/2} \tag{C.34}$$

$$= \left(\sqrt{\frac{N}{2}}\frac{\beta}{j\omega}\right)^{N} \tag{C.35}$$

Multiple inputs

The transfer function expression (C.35) was derived assuming a single input only. We now show how this results generalizes for multiple inputs. We consider the case of $L = 2$ inputs and the extension to arbitrary L is straightforward.

For $L > 1$, we let $N \triangleq N_\ell L$, where N_ℓ is the system order for a single channel. For $L = 2$ we have

$$\boldsymbol{C}_{CI_s}^{\mathsf{T}} = \begin{pmatrix} \mathbf{0}_{N/2}^{\mathsf{T}} & 0 \ \cdots \ 1 & 0 \ \cdots \ 0 \\ \mathbf{0}_{N/2}^{\mathsf{T}} & 0 \ \cdots \ 0 & 0 \ \cdots \ 1 \end{pmatrix} \in \mathbb{R}^{2\times N}, \tag{C.36}$$

$$\boldsymbol{B}_H = \begin{pmatrix} \beta \ \cdots \ 0 & 0 \ \cdots \ 0 & \mathbf{0}_{N/2}^{\mathsf{T}} \\ 0 \ \cdots \ 0 & \beta \ \cdots \ 0 & \mathbf{0}_{N/2}^{\mathsf{T}} \end{pmatrix}^{\mathsf{T}} \in \mathbb{R}^{N\times 2}, \tag{C.37}$$

and

$$\boldsymbol{H}_N'\boldsymbol{B}_H = \beta \begin{bmatrix} \mathbf{1}_{N_\ell/2} & \mathbf{1}_{N_\ell/2} \\ \mathbf{1}_{N_\ell/2} & -\mathbf{1}_{N_\ell/2} \\ \mathbf{0}_{N/2} & \mathbf{0}_{N/2} \end{bmatrix} \in \mathbb{R}^{N\times 2}, \tag{C.38}$$

$\boldsymbol{A}'$ is as given by (5.16), but with

$$\boldsymbol{L}_{N/2} \triangleq \begin{bmatrix} \boldsymbol{L}_{N_\ell/2} & \mathbf{0}_{N_\ell/2\times N_\ell/2} \\ \mathbf{0}_{N_\ell/2\times N_\ell/2} & \boldsymbol{L}_{N_\ell/2} \end{bmatrix}. \tag{C.39}$$

In (C.12) we used the power series of $\boldsymbol{L}_{N/2}$. For $L = 2$ we have

$$\boldsymbol{L}_{N/2}^k = \begin{bmatrix} \boldsymbol{L}_{N_\ell/2} & \mathbf{0}_{N_\ell/2\times N_\ell/2} \\ \mathbf{0}_{N_\ell/2\times N_\ell/2} & \boldsymbol{L}_{N_\ell/2} \end{bmatrix}^k \tag{C.40}$$

$$= \begin{bmatrix} \boldsymbol{L}_{N_\ell/2}^k & \mathbf{0}_{N_\ell/2\times N_\ell/2} \\ \mathbf{0}_{N_\ell/2\times N_\ell/2} & \boldsymbol{L}_{N_\ell/2}^k \end{bmatrix}, \tag{C.41}$$

and in consequence

$$\boldsymbol{\Psi}_{N/2} = \begin{bmatrix} \boldsymbol{\Psi}_{N_\ell/2} & \mathbf{0}_{N_\ell/2\times N_\ell/2} \\ \mathbf{0}_{N_\ell/2\times N_\ell/2} & \boldsymbol{\Psi}_{N_\ell/2} \end{bmatrix}. \tag{C.42}$$

Furthermore

$$\boldsymbol{\Psi}_{N/2}\boldsymbol{H}_{N/2} = \begin{bmatrix} \boldsymbol{\Psi}_{N_\ell/2} & \mathbf{0}_{N_\ell/2\times N_\ell/2} \\ \mathbf{0}_{N_\ell/2\times N_\ell/2} & \boldsymbol{\Psi}_{N_\ell/2} \end{bmatrix} \begin{bmatrix} \boldsymbol{H}_{N_\ell/2} & \boldsymbol{H}_{N_\ell/2} \\ \boldsymbol{H}_{N_\ell/2} & -\boldsymbol{H}_{N_\ell/2} \end{bmatrix} \tag{C.43}$$

$$= \begin{bmatrix} \boldsymbol{\Psi}_{N_\ell/2}\boldsymbol{H}_{N_\ell/2} & \boldsymbol{\Psi}_{N_\ell/2}\boldsymbol{H}_{N_\ell/2} \\ \boldsymbol{\Psi}_{N_\ell/2}\boldsymbol{H}_{N_\ell/2} & -\boldsymbol{\Psi}_{N_\ell/2}\boldsymbol{H}_{N_\ell/2} \end{bmatrix}, \tag{C.44}$$

and

$$\boldsymbol{\Psi}_{N/2}\boldsymbol{H}_{N/2} \begin{bmatrix} \mathbf{1}_{N_\ell/2} & \mathbf{1}_{N_\ell/2} \\ \mathbf{1}_{N_\ell/2} & -\mathbf{1}_{N_\ell/2} \end{bmatrix} = \begin{bmatrix} 2\boldsymbol{\Psi}_{N_\ell/2}\boldsymbol{H}_{N_\ell/2} & \mathbf{0}_{N_\ell/2} \\ \mathbf{0}_{N_\ell/2} & 2\boldsymbol{\Psi}_{N_\ell/2}\boldsymbol{H}_{N_\ell/2} \end{bmatrix}, \tag{C.45}$$

The expression for the transfer function then becomes

$$\boldsymbol{G}(\omega) = \frac{\beta^2}{(j\omega)^2} \begin{pmatrix} 0 & \cdots & 1 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 0 & \cdots & 1 \end{pmatrix} \boldsymbol{\Psi}_{N/2}\boldsymbol{H}_{N/2} \begin{bmatrix} \mathbf{1}_{N_\ell/2} & \mathbf{1}_{N_\ell/2} \\ \mathbf{1}_{N_\ell/2} & -\mathbf{1}_{N_\ell/2} \end{bmatrix} \tag{C.46}$$

$$= \frac{2\beta^2}{(j\omega)^2} \begin{pmatrix} 0 & \cdots & 1 & 0 & \cdots & 0 \\ 0 & \cdots & 0 & 0 & \cdots & 1 \end{pmatrix} \begin{bmatrix} \boldsymbol{\Psi}_{N_\ell/2}\boldsymbol{H}_{N_\ell/2} & \mathbf{0}_{N_\ell/2} \\ \mathbf{0}_{N_\ell/2} & \boldsymbol{\Psi}_{N_\ell/2}\boldsymbol{H}_{N_\ell/2} \end{bmatrix} \tag{C.47}$$

$$= \begin{pmatrix} \left(\sqrt{\frac{2N_\ell}{2}}\frac{\beta}{j\omega}\right)^{N_\ell} \\ \left(\sqrt{\frac{2N_\ell}{2}}\frac{\beta}{j\omega}\right)^{N_\ell} \end{pmatrix} = \begin{pmatrix} \left(\sqrt{\frac{N}{2}}\frac{\beta}{j\omega}\right)^{N_\ell} \\ \left(\sqrt{\frac{N}{2}}\frac{\beta}{j\omega}\right)^{N_\ell} \end{pmatrix}. \tag{C.48}$$

From this derivation we see that for an ADC with an arbitrary number of input channels, the transfer function is

$$G(\omega) = \left(\sqrt{\frac{N}{2}} \frac{\beta}{j\omega} \right)^{N_\ell} \tag{C.49}$$

for all channels.

Bibliography

[1] H.-A. Loeliger, H. Malmberg, and G. Wilckens, "Control-bounded analog-to-digital conversion: Transfer function analysis, proof of concept, and digital filter implementation," 2020.

[2] H. Malmberg, "Control-bounded converters." 2020.

[3] T. C. Carusone, D. Johns, and K. Martin, *Analog Integrated Circuit Design, second edition.* John Wiley and Sons Inc., 2013.

[4] H. Loeliger, L. Bolliger, G. Wilckens, and J. Biveroni, "Analog-to-digital conversion using unstable filters," in *2011 Information Theory and Applications Workshop*, pp. 1–4, 2011.

[5] H. Loeliger and G. Wilckens, "Control-based analog-to-digital conversion without sampling and quantization," in *2015 Information Theory and Applications Workshop (ITA)*, pp. 119–122, 2015.

[6] H. Malmberg and H. A. Loeliger, "Analog-to-digital conversion using self-averaging analog hadamard networks," in *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 1–1, 2020.

[7] B. D. O. Anderson and J. B. Moore, *Optimal Filtering.* Prentice Hall, 1979.

[8] I. Cadence Design Systems, "Spectre simulation platform." `https://www.cadence.com/ko_KR/home/tools/custom-ic-analog-rf-design/circuit-simulation/spectre-simulation-platform.html`, 2020.

[9] F. Feyling, "cbadcsim2." `https://github.com/fredrief/cbadcsim2`, 2020.