

MULTI-CHANNEL ANALOG-TO-DIGITAL CONVERSION USING A SINGLE-CHANNEL QUANTIZER

Youngchun Kim, Ahmed H. Tewfik

The University of Texas
Electrical and Computer Engineering
Austin, TX.

B. Vikram Gowreesunker

Texas Instruments Inc.
Applications R&D Center
Dallas, TX.

ABSTRACT

We present a novel analog-to-digital converter (ADC) architecture and conversion scheme to digitize multiple input channels using a single-channel quantizer. The proposed system consists of an analog front-end and a digital back-end. In the analog front-end, bandlimited M -channel sparse signals are discretized using sample-and-hold circuits, and then modulated using pseudo-random binary sequences. After modulation, the modulated signals are summed together before quantization. This mixture is converted to digital sequences with finite resolution followed by a single quantizer. The digital back-end separates the digitized mixture into M -channel digital sequences. For separation, we propose different classes of reconstruction methods that are used in sparse signal representation. The experimental results, with an ideal quantizer, show perfect recovery of the input signals if the input signals have sufficient sparseness. In the case of a realistic ADC with 16-bit quantization noise, the reconstruction is possible up to 108 dB in signal-to-reconstruction error ratio.

Index Terms— multi-channel ADC, source separation, sparse representation, spectrum spreading.

1. INTRODUCTION

Signal processing applications require digitization of multiple channels of analog signals, such as sensor arrays, monitoring physiological signals, and brain machine interfaces (BMI). In BMI, 32 to 128 channel data acquisition is usual to analyze and estimate brain activity. In cellphone and smart-phone applications, multi-microphone techniques are widely used for noise cancellation and speech enhancement techniques. Inspired from these examples, we propose a new ADC architecture to answer the question of whether or not we can use a fewer number of ADCs than channels. In this paper, we show that it is possible to use fewer ADCs than channels if we leverage sparse representation of signals and blind source separation (BSS) techniques [1, 2].

Two straightforward approaches to perform multi-channel A/D conversion with a single-channel quantizer are time multiplexing and frequency multiplexing. In time multiplexing,

every input channels are converted in turns based on a multiplexing scheme. Thus, the effective bandwidth per channel is linearly reduced by the total number of channels. In frequency multiplexing, analog inputs are modulated to occupy non-overlapping frequency bands, and then the sum of modulated signals is digitized. The main disadvantage of the frequency modulation is the high operation frequency, which increases linearly with respect to the number of channels, leading to extra power consumption. Compressed sensing (CS) approaches have been studied to sample and reconstruct a signal at sub-Nyquist rates [3, 4]. However, they focus on sampling single channel information rather than converting multiple channels. Recently, Slavinsky and Baraniuk [5] proposed the compressive multiplexer (CMUX) for multi-channel signals, but CMUX has an architecture based on a CS technique that leads to bandwidth expansion.

We describe here an alternative approach that does not require bandwidth expansion, assuming that all the input channels that will be digitized are represented in known dictionaries. The proposed ADC system has several advantages in circuit or semiconductor scale implementation. The system is realized with switched-capacitor (SC) based sample-and-hold (S/H) circuits. The reasons to use a SC circuit are its high linearity, and insensitivity to process, voltage, and temperature variations as well. Such systems perform A/D conversion using discrete-time operations implemented in analog domain. In our proposed solution, the input signals are sampled at the Nyquist rate or a slightly higher rate, which is determined by the maximum number of sinusoids in the input channels. The sampled signals are mixed after multiplication by properly selected binary sequences (± 1) that can easily be implemented via polarity reversal. The resulting output is then separated into digitized sequences corresponding to the input channels. For sufficiently sparse input signals, separation, up to the accuracy of a quantizer, can be obtained by using an extension of traditional sparse signal reconstruction methods.

This paper is organized as follows: in Section 2, we give an overview of the proposed system. In Section 3, we discuss sparse signal model and reconstruction algorithms. In Section 4, we present experimental results of the proposed

system. Finally, we discuss the outcome of the study in Section 5.

2. SYSTEM OVERVIEW

The proposed system consists of two blocks, an analog front-end and a digital back-end. The analog front-end includes lowpass filters (LPFs), SC based S/H circuits and modulators, and a single-channel quantizer. The overall architecture of the proposed ADC system is shown in Fig. 1(a). In this paper, for the limitation of space, our exposition focuses on the discretization and reconstruction schemes rather than presenting a circuit realization of the proposed system.

2.1. Analog front-end

At the analog front-end, the bandlimited inputs are sampled and modulated with a set of pseudo-random sequences. The LPFs in Fig. 1(a) are traditional anti-aliasing filters, which limit each input signal bandwidth to the Nyquist rate of the quantizer. We can implement the modulators, S/H circuits, and the adder using SC circuits and an operational transconductance amplifier (OTA). We do not need to design a special single-channel quantizer, and any type of quantizers or ADCs, such as flash, folding and interpolating, pipelined, and successive approximation register (SAR), can be used to digitize the discrete-time mixture signal to digital sequences with limited resolution.

2.2. Digital back-end

The digital back-end recovers each input channel using a recovery algorithm. It also generates spreading sequences for each input channel and controls the S/H circuits and modulators. Once we select a pseudo-random sequence, the sequence will be pre-stored in a memory unit to control the modulators. The performance of the proposed ADC depends on the recovery algorithm operating on the digital back-end, and it could be a major power consuming unit. However, we expect that the proposed architecture will consume less energy than traditional multi-channel scheme with benefits from the progress speed and scale of digital circuit industry.

2.3. Sampling rate

Each input channel is assumed to have a sparse representation in a known dictionary, over any interval of time, and each signal can be reconstructed using a number of dictionary elements, which is less than the degrees of freedom associated with the time interval. The sampling rate is selected such that the number of samples in any time interval T is larger than the sum of the total number of complex exponentials in all input signals. For example, assuming that at the Nyquist sampling rate, the time interval over which the signals are quasi-stationary consists of N samples, and that each signal consists

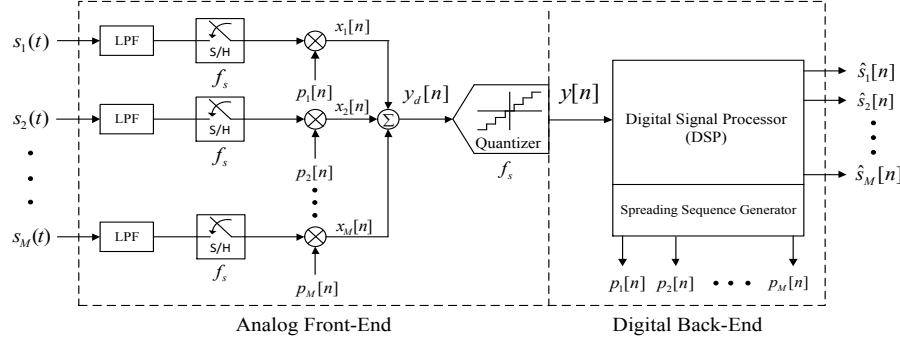
of a random superposition of a random number of columns of the N -point discrete Fourier transform (DFT) matrix. Then reconstruction from sampling the signals at the Nyquist rate is possible only if the sum of the numbers of complex exponentials in each signal is less than N . In a more general case, for signals that are quasi-stationary over intervals of length T , the sampling rate f_s must provide in any time interval T a number of samples $N = Tf_s$ such that N is larger than the sum of the total number of dictionary elements required to express each of the input signals.

2.4. Modulation and mixing

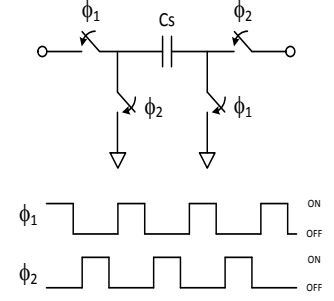
The main challenge in the proposed system is to separate each input channel after they have been summed together. To address this problem, we modulate all channel inputs by a properly designed sequence, and we select binary sequences of ± 1 to perform the modulation. Such a modulation scheme can be simply implemented by polarity reversal. In addition, it avoids amplifying or attenuating the signal amplitude. From all of the possible binary sequences, we seek a binary sequence that will simplify the signal reconstruction problem. Demodulation at the DSP block is very simple since it consists of multiplying the signal samples by the same sequence used in the modulation stage. After modulation by such a sequence, each modulated dictionary elements must have a non-sparse representation in terms of the original dictionary, but the sparsest representation of the mixture will consist of the columns of the modulated DFT matrix. In realization, we can combine a S/H and a modulator with an SC based block 1(b). The SC based implementation allows us to perform discrete-time operations in the analog domain. In particular, the modulation is a discrete-time modulation performed in the analog domain, and hence, leads to no bandwidth expansion.

2.5. Sequence selection

We seek a binary pseudo-random sequence for spreading each input signal over the bandwidth which is determined by the sampling rate f_s . A class of binary pseudo-random sequences with good correlation properties are reported in many communication studies. The maximum length, Gold, Kasami, and Hadamard sequences are popular sequences that belongs to this classes. Each binary sequence has different lengths of period, bounds, and auto- and cross-correlation characteristics. The first three, maximum length, Gold, and Kasami sequences have an odd integer lengths, while Hadamard sequence has an even integer length. These different periodicities can be exploited to design effective signal separation methods, including the general case where the input signals consist of unknown numbers of sinusoids of unknown frequencies. Although the sequences have different periodicities, they exhibit a good auto-correlation property, and the cross-correlations of Kasami sequences approach Welch's lower bound. The correlation properties of sequences are



(a) ADC system architecture



(b) S/H block realization using SCs.

Fig. 1. Proposed ADC structure with two input channels (a), and SC based S/H circuit (b). The S/H block is controlled by two non-overlapping ϕ_1 and ϕ_2 clocks. When ϕ_1 is high, the S/H circuit samples the input. When ϕ_2 is high, it holds the sampled signal to modulate before summing all channel inputs.

important to choose accurate dictionary atoms from the signal mixture, and we need to investigate these properties in selecting a sequence. Hadamard sequence is reported to offer a poor spectral estimation probability because it shows multiple peaks in its cross-correlation function. Kasami sequence is known to be the optimal choice in communication, but this sequence also yields many peaks at Welch's bound level [6]. The maximum length and Gold sequences yield a few peaks with low aggregate energy, thus we adopt the maximum length sequence for simplicity in generation.

3. SPARSE INPUT MODEL AND RECONSTRUCTION

In our ADC architecture, input channels are added all together after modulation, and the mixture is digitized at the quantizer. We try to find the sparsest basis vectors to represent the digitized mixture from a known dictionary via various sparse signal representation methods. The dictionary is a union of atoms that consist of real or complex spectral supports, and the dictionary and its atoms can be formed by commonly used basis for example, DFT, DCT, MDCT, and wavelet basis. We will use DFT basis in the rest of the paper, but we can choose one of the basis, which will lead to a sparse representation of input signals.

3.1. Sparse signal model

Without loss of generality and for simplicity of exposition, we assume a signal consists of a superposition of a random number of columns of the N -point DFT matrix. We can represent the m^{th} -channel $N \times 1$ sparse signal vector \vec{s}_m in terms of N -point DFT dictionary $\Omega = \{\vec{\omega}_n\}_{n=1}^N$, such that each atom $\vec{\omega}_n$ is an $N \times 1$ column vector,

$$\vec{s}_m = \Omega \vec{c}_m = \sum_{n=1}^N c_{m,n} \vec{\omega}_n, \quad (1)$$

where $m = 1, 2, \dots, M$, and $c_{m,n}$ is the n^{th} entry of a sparse coefficient vector \vec{c}_m corresponding to the dictionary atom $\vec{\omega}_n$. If the signal vector \vec{s}_m is K -sparse, then its coefficient vector \vec{c}_m has K nonzero entries. Each input signal is now modulated with a spreading sequence vector \vec{p}_m associated with the m^{th} input channel, which can be written as

$$\vec{x}_m = \vec{p}_m \circ \vec{s}_m, \quad (2)$$

where the operator \circ denotes entry-wise vector multiplication. Subsequently, we obtain a mixture of the modulated signal vectors which has the same bandwidth of the input signal,

$$\vec{y} = \sum_{m=1}^M \vec{x}_m = \sum_{m=1}^M \sum_{n=1}^N c_{m,n} (\vec{p}_m \circ \vec{\omega}_n), \quad (3)$$

where \vec{y} is a $N \times 1$ mixture vector. We rewrite equation (3) with an augmented matrix $A = [A_1 \ A_2 \ \dots \ A_M]$ which consists of modulated DFT dictionaries $A_m = \{\vec{p}_m \circ \vec{\omega}_n\}_{n=1}^N$ such that,

$$\vec{y} = A \vec{c}, \quad (4)$$

where $\vec{c}^T = [\vec{c}_1^T \ \vec{c}_2^T \ \dots \ \vec{c}_M^T]$ is the concatenated coefficient vector of each \vec{c}_m . The equation (4) shows an under-determined system configuration, and thus we can recover each channel input \vec{s}_m by accurate estimation of the coefficient vector \hat{c}_m and the original DFT dictionary Ω as

$$\hat{s}_m = \Omega \hat{c}_m. \quad (5)$$

We introduce three major families of algorithms to find a sparse solution of the under-determined system in the following subsection.

3.2. Reconstruction algorithms

There are several classes of algorithms that provide a sparse solution of under-determined system setup, which include

convex optimization and greedy methods. We propose a third method that exploits advantages from both approaches and shows the best performance in our experiments.

Linear/convex optimization: The first approach to obtain a sparse solution is the traditional ℓ_1 sparse signal recovery method [7] that is widely used in CS community. LASSO-styled ℓ_1 minimization of [8] is a popular method in this category regularizing ℓ_1 minimization problems. Finding ℓ_1 and LASSO solutions are obtained through disciplined convex programming with CVX, a MATLAB package for specifying and solving convex programs [9].

Greedy methods: Another class of sparse signal representation methods is greedy methods, the orthogonal matching pursuit (OMP), and the orthogonal least square (OLS). We tailor the OLS sparse signal representation method [2] fit to our recovery system. The OLS approach is similar to the OMP but uses a different directional update scheme. The modification consists of drawing, at each step, simultaneously elements from each of the dictionaries in the union of spread dictionaries that provide a sparse representation of the mixture. That is, in our illustration, in each step we select one column from the spread DFT matrix A . In contrast, the traditional OLS method selects one entry from the signal dictionary Ω at each step. We also compare the OLS method with CoSamp [10].

Combined $\ell_{1,2}$ method: By combining advantages from convex and greedy approaches, we can achieve better reconstruction performance than each independent method. First, a collection of frequency index to represent the m^{th} channel input \vec{s}_m , is estimated from a ℓ_1 solution such that $I_m = \{1 \leq i \leq N | c_{m,i} > \epsilon\}$, where ϵ is a positive thresholding value slightly greater than zero, and $c_{m,i}$ is the i^{th} entry of the coefficient vector \vec{c}_m . Secondly, we compute a least square solution \tilde{c} using a new dictionary formed by column vectors that belongs to I_m from the modulated matrix A_m , minimize $\|\vec{y} - \sum_{i \in I_m} \vec{a}_{m,i} \tilde{c}_{m,i}\|_2$, where $\vec{a}_{m,i}$ is the i^{th} column vector of A_m . The recovery is performed as, $\vec{s}_m = \sum_{i \in I_m} \tilde{c}_{m,i} \vec{w}_i$.

In the general case, the signal can be represented with an overcomplete dictionary with columns of the form $\vec{d}_n = \{e^{-jw_n k}\}$, where $w_n = n\Delta w$ and Δw is determined to provide the decreased frequency resolution. A better approach to the general case is using **Combined $\ell_{1,2}$ method** to yield an estimate of the frequencies in the mixture. These frequencies $\{w_n\}_{n=1}^N$ are used to generate the columns of the dictionary $\vec{d}_n = \{e^{-jw_n k}\}$. The augmented measurement matrix will then consist of the modulated dictionary elements such that $A_m = \{\vec{p}_m \vec{d}_n\}$.

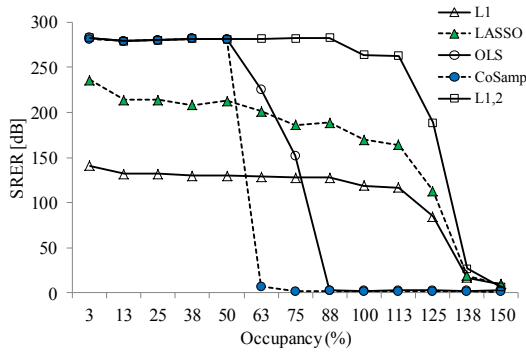
3.3. Comparison with prior works

Here we emphasize several differences between the proposed ADC system and conventional CS systems including a recent

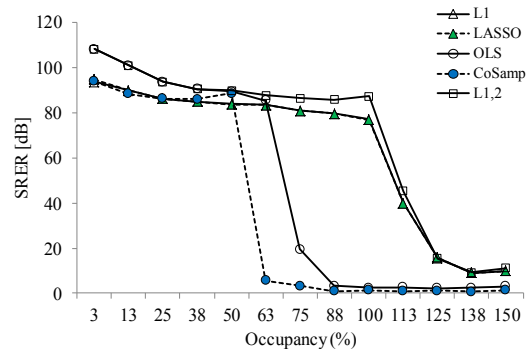
study [5]. Like CS systems, the proposed system relies on the use of ± 1 sequences. But, in CS, the binary sequences are used to compute random projections of the input signal in the analog domain, leading to bandwidth expansion, and multiplication is followed by integration to compute a projection. For the random projection of input signals and integration of that, an integrator [3] is required before or after modulating input signals. However, we do not compute projections and do not use an integrator after modulation. In our system, the modulation and summing processes do not expand the input bandwidth which is limited by the LPFs and the SC based S/H circuit. The modulation in our system is a discrete-time operation, even though implemented in the analog domain using SC circuits. The purpose of modulation in the analog front-end is to maximally decorrelate the dictionaries used to represent different input channels. The proposed system is implemented using SC based S/H with modulators and an adder. Thus, the proposed system has more implementable than CS systems. For example, in CS system, integrators need to be reset every sample frame, but the proposed system does not require the reset stage. In addition, the proposed system is easier to calibrate than CS systems because all operations are performed in discrete-time domain, and we can implement the analog front-end with high linearity using SC based S/H circuits.

4. EXPERIMENTAL RESULTS

In this section, we demonstrate experimental results obtained by digitizing two-channel sparse input signals. To evaluate reconstruction fidelity, we define signal-to-reconstruction error ratio (SRER) between each input and the corresponding reconstructed signal as $\text{SRER}(\vec{s}_m, \hat{s}_m) = 20 \log_{10} (\|\vec{s}_m\|_2 / \|\vec{s}_m - \hat{s}_m\|_2)$, where \vec{s}_m is the m^{th} channel input and its recovery signal \hat{s}_m . To look at the relationship between input signal sparsity versus reconstruction performance, we define a metric for sparseness measure. We compute a percentage of underlying frequency components in the mixture signal in terms of *occupancy* (%) = $\sum_{m=1}^M K_m / (B T) \times 100$, where K_m is the number of frequency components of the m^{th} channel input \vec{s}_m , B is the bandwidth of the mixture signal, and T is the time interval of sampling. First, we investigate reconstruction fidelity according to the occupancy of input signals. We generate multi-tone input signals that consist of K number of random integer frequency components. The frequency components and amplitudes of sinusoids are randomly selected within the available bandwidth. The mixture of all modulated inputs is sampled with the Nyquist rate, and we evaluated the reconstruction SRERs after reconstruction. Fig. 2 illustrates the reconstruction SRERs of the recovered signals using different recovery methods that we introduced in Subsection 3.2. We find empirically greedy methods outperformed over convex optimization methods for sufficiently ($\leq 60\%$) sparse input signals in Fig. 2(a). Yet, as the occupancy of input signals increases ($> 60\%$), con-



(a) ADC without quantization noise



(b) ADC with 16-bit quantization noise

Fig. 2. Experiments with multi-tone inputs. The horizontal axis indicates the occupancy and the vertical axis shows SRERs.

vex minimization methods show better SRERs than greedy methods. The other experiments are also performed with a practical ADC model with a 16-bit quantizer and the results are presented in Fig. 2(b). Since the ADC model has a 16-bit quantizer, 98 dB signal-to-quantization-noise ratio (SQNR) is expected under perfect recovery condition. The results with both ideal and realistic quantizers show the same trend on the SRER according to the occupancy of input frequency components, and they reach to maximum SRERs using the combined $\ell_{1,2}$ method. In practice, we need to use framing and windowing techniques for continuous conversion because the quasi-stationary property of input signals are not guaranteed. The recovery process is performed block-wise using a window function with 50% overlap among neighboring frames. There are many options to improve the recovery performance in the continuous conversion, and we list the expected improvements by applying different selections for the block-wise processing in Table 1.

5. CONCLUSION

In this paper, we proposed a new A/D conversion method for multi-channel inputs using a single-channel quantizer. We discussed the design of the system, including the selection of modulating sequences, sampling rate, dictionary selection to represent the mixture of spread input signals, and reconstruction algorithms. For this, we proposed different recovery methods and evaluated their experimental performances. The proposed system can recover each input signal with high SRERs when the input signals are sufficiently sparse. The sampling rate can be slightly higher than Nyquist rate, which is determined by the maximum expected bandwidth occupancy of the signals. In practice, the reconstruction performance depends on the accuracy of quantizers. From the empirical results, we conclude that sparse signals can share limited bandwidth with other sparse signals. Currently, we are working to answer questions, such as the maximum effective number of channels, the ratio of bandwidth expansion, and hardware re-

alization in semiconductor scale to make the proposed system applicable to devices requiring multi-channel ADCs.

Table 1. Expected Improvements

Category	Selections	Improvements
Window Functions	Hanning, Hamming, Sine, Ogg-Vorbis	≤ 5 dB
Spreading Sequences	Maximum, Gold, Kasami, Hadamard	≤ 5 dB
Dictionary	DFT, DCT, MDCT	≥ 10 dB
Recovery Methods	ℓ_1 , LASSO, OLS, CoSamp, $\ell_{1,2}$	≥ 10 dB

6. REFERENCES

- [1] S. Lesage, S. Krstulović, and R. Gribonval, "Under-determined source separation: comparison of two approaches based on sparse decompositions," in *ICA 2006*. 2006, pp. 633–640, Springer.
- [2] B.V. Gowreesunker and A.H. Tewfik, "Learning sparse representation using iterative subspace identification," *IEEE Transactions on Signal Processing*, vol. 58, no. 6, pp. 3055–3065, 2010.
- [3] W. Skones, B. Oyama, S. Stearns, J. Romberg, and E. Candes, "Analog to information (a-to-i), technical and management proposal," in *response to DARPA BAA 05-35.*, 2005.
- [4] M. Mishali and Y.C. Eldar, "From theory to practice: Sub-nyquist sampling of sparse wideband analog signals," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 375–381, 22 Feb. 2010.
- [5] J.P. Slavinsky, J.N. Laska, M.A. Davenport, and R.G. Baraniuk, "The compressive multiplexer for multi-channel compressive sensing," in *Proc. of IEEE Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*, May 2011.
- [6] Moshe Mishali and Yonina C. Eldar, "Expected rip: Conditioning of the modulated wideband converter," in *IEEE Information Theory Workshop*, May 2009.
- [7] Emmanuel J. Candes, Michael B. Wakin, and Stephen P. Boyd, "Enhancing sparsity by reweighted ℓ_1 minimization," *Journal of Fourier Analysis and Applications*, vol. 14, pp. 877–905, 2008.
- [8] R. Tibshirani, "Regression shrinkage and selection via the lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.
- [9] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 1.21," <http://cvxr.com/cvx>, April 2011.
- [10] D. Needell and J. A. Tropp, "Cosamp: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, pp. 301–321, May 2009.