

Alma Mater Studiorum Università di Bologna  
Archivio istituzionale della ricerca

A broadband multi-mode compressive sensing current sensor SoC in 0.16 $\mu$ m CMOS

This is the final peer-reviewed author's accepted manuscript (postprint) of the following publication:

*Published Version:*

Bellasi, D., Crescentini, M., Cristaudo, D., Romani, A., Tartagni, M., Benini, L. (2019). A broadband multi-mode compressive sensing current sensor SoC in 0.16 $\mu$ m CMOS. IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS. I, REGULAR PAPERS, 66(1), 105-118 [10.1109/TCSI.2018.2846573].

*Availability:*

This version is available at: <https://hdl.handle.net/11585/657220> since: 2019-05-17

*Published:*

DOI: <http://doi.org/10.1109/TCSI.2018.2846573>

*Terms of use:*

Some rights reserved. The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

This item was downloaded from IRIS Università di Bologna (<https://cris.unibo.it/>).  
When citing, please refer to the published version.

(Article begins on next page)

This is the post peer-review accepted manuscript of:

D. Bellasi, M. Crescentini, D. Cristaudo, A. Romani, M. Tartagni and L. Benini, "A Broadband Multi-Mode Compressive Sensing Current Sensor SoC in 0.16 $\mu$ m CMOS," in *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 66, no. 1, pp. 105-118, Jan. 2019.

The published version is available online at:

<https://doi.org/10.1109/TCSI.2018.2846573>

©2018 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

# A Broadband Multi-Mode Compressive Sensing Current Sensor SoC in 0.16 $\mu\text{m}$ CMOS

David Bellasi, *Student Member, IEEE*, Marco Crescentini, *Member, IEEE*, Domenico Cristaudo, Aldo Romani, Marco Tartagni, *Member, IEEE*, and Luca Benini, *Fellow, IEEE*

**Abstract**—Broadband current sensors are key components in numerous applications including power conversion, motor control, and smart-metering. We present a compressive sensing (CS) current sensor system-on-chip (SoC) designed and fabricated in STM 0.16  $\mu\text{m}$  BCD technology. The SoC is capable of measuring currents with amplitudes of up to 10 A peak with a sensing bandwidth of 1 MHz. Two broadband current sensing cores, each consisting of a Hall-effect probe, an AFE, and a 2 MS/s 9-bit ADC, are monolithically integrated together with a digital multi-mode compressive sensing encoder (DCSE) for data-rate reduction. We focus on the evaluation of CS as data compression codec for current sensing applications and on the details of the DCSE architecture designed for general purpose use. We introduce *multi-block decoding* that is a decoding modality to improve the reconstruction quality of “off-grid sparse” signals commonly occurring in practice. Moreover, we provide measurements of both the compression performance and power consumption of the SoC employed in two exemplary real-world applications; namely, sparse current sensing, and electrical appliance detection for non-intrusive load monitoring based on compressive measurements.

**Index Terms**—Hall sensor, current sensor, SoC, compressive sensing, non-uniform sampling, non-intrusive load monitoring

## I. INTRODUCTION

CURRENT-SENSING applications, such as power conversion, motor control, and smart-metering require small, cost- and energy-efficient current sensor devices offering a large sensing bandwidth up to the MHz range. As a consequence, data transfer rates of several Mbits per second need to be sustained, typically relying on serial data protocols.

In many practical applications, the effective signal bandwidth is considerably smaller than the sensing bandwidth [1], which allows the measured signal to be sparsely represented,

D. Bellasi, and L. Benini are with the Department of Information Technology and Electrical Engineering (D-ITET), ETH Zürich, 8092 Zürich, Switzerland (e-mail: {bellasi, benini}@iis.ee.ethz.ch).

L. Benini is also with the Department of Electrical, Electronic, and Information Engineering (DEI), University of Bologna, 40131 Bologna, Italy (e-mail: luca.benini@unibo.it).

M. Crescentini, A. Romani, and M. Tartagni are with the Advanced Research Centre on Electronic Systems (ARCES) and with the Department of Electrical, Electronic, and Information Engineering (DEI), University of Bologna, 47521 Cesena, Italy (e-mail: {marco.crescentini3, aldo.romani, marco.tartagni}@unibo.it).

D. Cristaudo is with STMicroelectronics, 95121 Catania CT, Italy (e-mail: domenico.cristaudo@st.com)

This work was financed in part by the Swiss National Science Foundation projects No. 162524 and No. 157048. Part of this work has received funding from the ECSEL Joint Undertaking (JU) under grant agreement No. 737434. This JU receives support from the European Union’s Horizon 2020 research and innovation programme and Germany, Slovakia, Netherlands, Spain, Italy. This work reflects only the authors’ view and the JU is not responsible for any use that may be made of the information it contains.

e.g., in the discrete Fourier or wavelet domain. In this case, data compression based on compressive sensing (CS) [2] can successfully be leveraged to reduce the interface data-rate between the sensor device and any down-stream data processing unit.

Data compression based on CS is generally lossy and both the recovery quality and the compression factor are signal-dependent. Yet, it has been noted that CS offers several advantages with respect to conventional data compression methods, such as the facts that CS is a highly asymmetric codec in which the encoding stage has a particularly low hardware complexity, and that no a-priori knowledge of the signal statistics is required to achieve data compression. These properties make CS an interesting data compression method for remote sensing applications, where the sensor data is primarily processed at a central data collection point with little constraints in computing power.

The application of CS was first considered in high-rate applications, such as radio-frequency communication, radar and high resolution imaging. The idea of applying CS in the context of digital data compression in low-rate sensors is relatively new, and only a handful of integrated CS encoder implementations have been reported in the literature so far. The majority of these encoders were conceived for single-mode encoding of biomedical signals [3]–[10].

In this paper, we elaborate on the practical aspects of realizing a CS-based data compression codec for current sensing applications and present a digital multi-mode compressive sensing encoder (DCSE) that was fabricated as integral part of a 2 MS/s, 10 A peak, current sensor system-on-chip (SoC). The SoC was realized in a 0.16  $\mu\text{m}$  BCD process using CMOS devices only, and combines two Hall-effect probes, two analog front-ends (AFEs), two 9-bit analog-to-digital converters (ADCs), and the DCSE on a single die [11]. To the best of our knowledge, this SoC is the first CS-enabled current sensor reported in the open literature.

The SoC was designed and implemented for general purpose broadband current sensing applications, including—but not limited to—smart metering for non-intrusive load monitoring and over-current protection in fast-switching power circuits requiring the detection of fast current spikes. In both cases, the detection of current signals with amplitudes up to 10 A peak with at least 6 bit of resolution and with bandwidths up to 1 MHz is required. Typical current sensors based on CMOS Hall sensors are limited to a few hundred kHz [12]–[14]. Our SoC distinguishes itself from prior current sensors by achieving a large bandwidth of 1 MHz, and from prior CS systems by supporting multi-mode encoding

including hardware sample skipping. The DCSE not only supports encoding based on random modulation (RM) [15], but also allows other modes to be realized, such as random filtering [16], modes based on sample-skipping like non-uniform sampling (NUS) [2], and more advanced modes aimed at compressive signal processing [17]. The multi-mode capability of the encoder provides the necessary flexibility to reduce the data-rate in the acquisition of a wide range of sparse current signals.

### A. Contributions

Our contributions can be summarized as follows:

- 1) We provide a detailed description of the digital multi-mode CS encoder and pertaining measurement results from our CS current sensor SoC prototype. This work complements [11], where our SoC has been unveiled already, but was only summarily presented.
- 2) We elaborate on CS as a data compression codec, and discuss practical aspects of different encoding modes. In particular, non-uniform sampling is confronted with random modulation in terms of quality/compression trade-off and power consumption, and hardware sample-skipping is discussed.
- 3) We address the problem of "off-grid sparsity" that is often encountered in practice and results in lower-than-expected reconstruction quality; specifically, we propose a multi-block decoding method that increases the resolution of the sparsifying basis without burdening the encoder hardware with a larger blocksize.
- 4) We present end-to-end tests and measurement results of our CS-enabled current sensor SoC prototype employed in two real-world application scenarios, one of which is the first implementation of a smart utility meter for non-intrusive load monitoring employing CS-based load-disambiguation.

### B. Paper Outline

The rest of the paper is organized as follows: We start with a review of the compressive sensing basics, followed by a discussion of the encoding modes most relevant for integrated CS encoders in Section II. A detailed description of the CS current sensor SoC is provided in Section III, while in Section IV the corresponding implementation results are reported. In Section V we report the performance results of the fabricated prototype evaluated in two exemplary application examples. Section VI concludes the discussion with a small summary.

## II. COMPRESSIVE SENSING-BASED DATA COMPRESSION

In this section, we briefly review the basics of CS and discuss its application as digital data compression codec. Then, we detail the CS encoding modes most suitable for the implementation in integrated hardware, and finally, introduce the multi-block encoding method to mitigate the effects of off-grid sparsity.

### A. The CS Codec

CS can achieve data compression by linearly projecting an  $N$ -dimensional signal vector  $\mathbf{x} \in \mathbb{R}^N$  into an  $M$ -dimensional subspace ( $M < N$ ) [18], [19]. The linear projections, called *CS measurements*, are obtained as follows:

$$\mathbf{y} = \Phi \mathbf{x} + \mathbf{n}, \quad (1)$$

where  $\Phi \in \mathbb{R}^{M \times N}$  is the *measurement matrix* and  $\mathbf{n} \in \mathbb{R}^M$  is additive measurement noise.<sup>1</sup>

CS can be understood as a data coder-decoder (codec) in which (1) describes the encoding stage. The decoding stage has the task of reconstructing the original signal vector  $\mathbf{x}$  from the measurements  $\mathbf{y}$ . Generally, (1) can not be inverted unambiguously, since the inverse problem—recovery of  $\mathbf{x}$  from  $\mathbf{y}$ —is ill-posed. To make reconstruction possible, CS leverages the fact that signal compression is possible if the signal is *sparsely* representable in some basis or frame  $\Psi \in \mathbb{C}^{N \times P}$  with  $P \geq N$ , such that

$$\mathbf{x} = \Psi \boldsymbol{\alpha}, \quad (2)$$

where the majority of coefficients in  $\boldsymbol{\alpha} \in \mathbb{C}^P$  is zero. Under this condition, by solving the convex optimization problem

$$\text{minimize } \|\boldsymbol{\alpha}\|_1 \quad \text{subject to } \|\mathbf{y} - \mathbf{D}\boldsymbol{\alpha}\|_2 < \epsilon, \quad (3)$$

$\mathbf{x}$  can be recovered from  $\mathbf{y}$  via (2), provided that the combined measurement matrix  $\mathbf{D} = \Phi \Psi$  fulfills certain mathematical properties related to the number of non-zero coefficients in  $\boldsymbol{\alpha}$  [19], [20]. The reconstruction tolerance  $\epsilon$  (an arbitrary non-negative number) renders the optimization problem feasible in the presence of measurement noise. It can be shown that (3) finds the sparsest  $\boldsymbol{\alpha}$  that is consistent with the measurements, and that this very  $\boldsymbol{\alpha}$  is the unique solution to the recovery problem [18].

The signal  $\mathbf{x}$  is said to be  $\kappa$ -sparse if  $\|\boldsymbol{\alpha}\|_0 \leq \kappa$ , i.e.,  $\boldsymbol{\alpha}$  has at most  $\kappa$  non-zero entries. Natural signals are typically not exactly sparse, but can often be approximated by an exactly  $\kappa$ -sparse vector  $\boldsymbol{\alpha}_k$ , such that  $\|\mathbf{x} - \Psi \boldsymbol{\alpha}_k\|_1 < \delta$  for an arbitrary non-negative  $\delta$ . In the latter case, the combination of (3) and (2) yields an approximation of  $\mathbf{x}$ .

The task of the decoding stage consists of solving (3) which requires sophisticated reconstruction algorithms either based on  $L_1$ -minimization or greedy heuristics [21]. The computational complexity of the decoding stage is therefore considerably larger than that of the encoding stage, which makes the CS codec extremely asymmetrical in this sense. In remote sensing applications, there is typically no need to process or display the raw time-domain signal in the same place where the signal is acquired. The cost of a computationally intensive decoder stage is acceptable in such applications, because decoding is performed only at the receiving end of the remote link where the necessary computing power is readily available.

<sup>1</sup>We remark that we assume finite-dimensional discrete-time signals, since our aim is to provide a mathematical description relevant for our hardware implementation of CS-based data compression.

### B. Quality/Compression Trade-Off

The compression performance can be quantified by the compression factor (CF) defined as the ratio between the total number of bits that enter the encoder and the number of bits that are produced by the encoder in response to that input:

$$\text{CF} = (BN)/(QM), \quad (4)$$

where  $N$  and  $M$  are the number of signal samples and resulting CS measurements, respectively, while  $B$  and  $Q$  are the resolution in bits of a signal sample and of a CS measurement, respectively.

Naturally,  $M$  must be larger than  $\kappa$  for CS encoding to preserve the signal information (at least approximately). Yet,  $M$  not only depends on the sparsity but also on the degree of *coherence* between the measurement operator  $\Phi$  and the sparsifying basis  $\Psi$  [22]. The coherence is a measure for how sparsely the columns of  $\Psi$  can be represented with the rows of  $\Phi$  and vice versa. In general, the lower the coherence the lower the minimum number of measurements necessary to achieve a certain reconstruction quality.

Measurement matrices obtained from selecting random vectors independently from a probability distribution have an optimally low coherence with any fixed sparsifying basis with overwhelming probability [2]. In this sense such measurement matrices are considered a *universal* sampling modality. Under such conditions, it was shown that

$$M = \mathcal{O}(\kappa \log(N/\kappa)) \quad (5)$$

measurements suffice for stable recovery of  $\mathbf{x}$  from the noisy measurements  $\mathbf{y}$  [19], [20].

Since the CF is proportional to the ratio  $N/M$ , the choice of  $M$  is critical in determining the quality/compression trade-off of the CS codec. In practice, the number of measurements  $M$ —and thus the CF—necessary to achieve a certain reconstruction quality must be determined empirically.

### C. Encoding Modes

To extend a sensor system with CS-based data compression capabilities requires the addition of a CS encoder, i.e., a piece of hardware to calculate the matrix–vector–product described by (1). CS encoding can be realized at different points in the signal acquisition chain. The most frequently considered options are: Encoding between sensor and A/D conversion based on analog signal processing (analog CS), and encoding after A/D conversion using digital signal processing (digital CS). **Both the analyses in [9] and [15] showed that**—due to the large difference in power and area cost between analog and digital signal processing—digital CS is the more cost- and energy-efficient solution under the conditions typically encountered in remote sensing applications. Thus, the following discussion will be concerned with digital CS in which the signal  $\mathbf{x}$  consists of digital samples obtained from uniformly sampling, conditioning (i.e., amplifying and filtering), and digitizing the signal from the sensor.

Irrespective of the encoding domain, the hardware complexity and power consumption of a CS encoder are largely

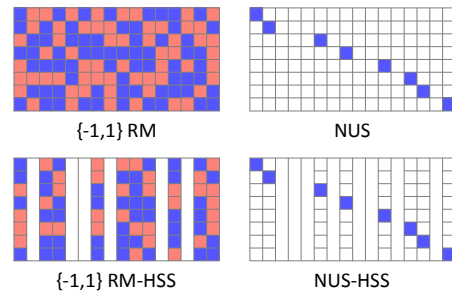


Fig. 1. Examples of  $\Phi$  matrices corresponding to different encoding modes. The blank squares represent the coefficient value zero.

influenced by the type of measurement matrix  $\Phi$ , its dimensions ( $M$  and  $N$ ), and the required degree of configurability. In addition, the choice of  $\Phi$  also affects the quality/compression trade-off of the codec and defines different encoding modalities. The most common encoding modalities are discussed in the following:

1) *Random-Modulation*: Measurement matrices with entries randomly chosen from a probability distribution work well with any kind of sparsifying basis. The corresponding mode is known as random-modulation (RM) since the CS measurements are inner-products between the input signal vector and the random vectors formed by the rows of  $\Phi$ .

The basic arithmetic operation required for RM is the multiply–accumulate (MAC) operation, whose hardware complexity and power consumption largely depend on the numeric values occurring in  $\Phi$ . Bounds on the reconstruction quality have been formally derived for real-valued and discrete (sub-)Gaussian distributions, and in particular also for binary-valued ( $\pm 1$  or  $\{0,1\}$ ) random matrices that allow the MAC operations to be realized without multipliers, based on adders alone. This circumstance and the observation that binary-valued RM yields virtually the same reconstruction quality as multi-valued RM (see, Section V-A) is the main reason for binary-valued RM being the most frequently considered CS mode in practice (e.g., in [4], [6]–[10]). An illustrative example of an RM-type binary-valued measurement matrix is shown in Figure 1 with the coefficient values displayed as different shades (colors).

In any case, RM measurement coefficients can either be provided by (on-chip) memory—such as ROM or RAM—or can be generated on-chip via a pseudo-random number generator, e.g., based on linear-feedback shift-registers (LFSR). RAM-based solutions offer user-configurability and allow encoding with structured measurement matrices, but have a higher area and power cost. Solutions based on LFSRs are most area- and power-efficient but produce fixed sequence(s) of coefficients. In any case, some form of digital memory is required to hold the intermediate and final CS measurements.

2) *Non-Uniform Sampling*: Non-uniform sampling (NUS) is a CS encoding mode in which  $\Phi$  is obtained by randomly selecting  $M$  rows from the  $N \times N$  identity matrix  $\mathbf{I}$ . Since  $\Phi$  is a  $\{0,1\}$ -coefficient matrix where each row only contains a single 1 in a random location (see Figure 1), it is highly coherent with the identity matrix and the reconstruction of (approximately) time-sparse signals is very unlikely to succeed. This may be the reason why NUS is only very

rarely considered in the CS literature. Yet, in many practical situations, the signals of interest are sparse in other domains that have a sufficiently low coherence with the identity matrix (e.g., Fourier-domain or wavelet-domain).

NUS-type encoding is equivalent to picking at random  $M$  of the possible  $N$  samples from the signal vector. Therefore no MAC operations are required in this mode, leading to a particularly low hardware complexity [15]. Same as in the RM mode, depending on the desired degree of programmability, the measurement coefficients for NUS can either be generated via an LFSR [23] or retrieved from RAM (the RAM size required in NUS is only a fraction of the one required for RM).

3) *Sample-Skipping*: If the measurement matrix contains all-zero columns, the signal samples at the corresponding position in the signal vector are not required in the encoding process, which allows these samples to be skipped. Sample-skipping (SS) can naturally be employed in the NUS encoding mode, but can be enforced in any other encoding mode as well; simply by zeroing complete columns of the measurement matrix. Examples of such measurement matrices are shown in the second row of Figure 1. Successful reconstruction of SS-encoded signals depends on the sparsity-domain being sufficiently incoherent with the time-domain (i.e., sufficiently low coherence between  $\Psi$  and  $\mathbf{I}$ ).

If the acquisition front-end is prevented from actually taking these samples, a significant reduction in the power consumption of a sensor is possible [15]. In the following, this hardware feature will be referred to as *hardware sample-skipping* (HSS) to emphasize the difference to simply multiplying the samples with zero during the encoding process. It is worth emphasizing that, HSS achieves both an A/D conversion rate and sampling rate reduction, whereas in standard RM-type encoding the signal is sampled and processed at the Nyquist rate in both digital and analog CS.

#### D. Multi-Block Decoding

Although CS acquisition and reconstruction can be designed with a certain degree of independence from each other, certain aspects of CS reconstruction must be taken into account in the design of the acquisition hardware and/or the design of the CS codec as a whole. One such aspect is related to CS reconstruction being performed on a per-block basis, i.e., consecutive blocks of  $N$  signal samples are reconstructed from their respective  $M$  CS measurements. Hardware design constraints such as power budget or hardware cost limit the maximum blocksize which is rarely larger than a thousand in practice [3]–[10]. A limited blocksize in combination with a large sensor bandwidth (i.e., a large sampling rate) can prevent a signal from being sparsely representable in  $\Psi$ , despite being characterized by only a small number of degrees of freedom. Following [24], such a signal may be described as *off-grid sparse*.

1) *Off-Grid Sparsity*: We illustrate the problem of off-grid sparsity with an example considering the DFT (discrete Fourier transform) as sparsifying transform. The top graph in Figure 2 shows the DFT-spectrum of  $N=500$  samples from a signal with a 300 kHz and a 400 kHz frequency component

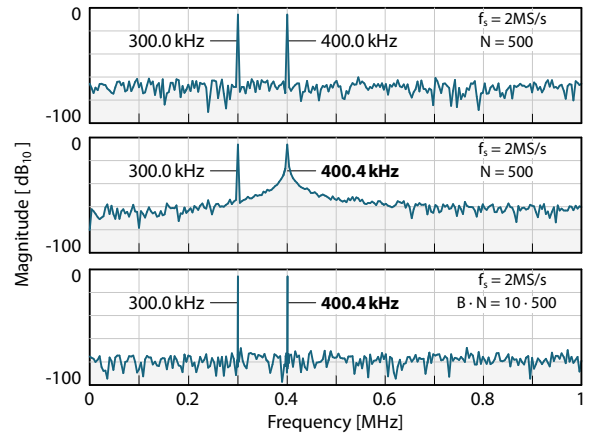


Fig. 2. The effect of off-grid tones on the single-sided DFT-spectrum: (top) on-grid tone at 400.0 kHz. (middle) off-grid tone at 400.4 kHz. (top) A 10-times larger blocksize brings the 400.4 kHz tone onto the grid again.

acquired at 2 MS/s. Both signal frequencies coincide with the frequency of atoms in the DFT matrix, resulting in a perfectly sparse spectrum corresponding to the exactly 2-sparse signal<sup>2</sup>.

The middle graph shows the spectrum of a signal with a 300 kHz and a 400.4 kHz component, acquired under the same conditions as in the first case. Although this signal is also exactly 2-sparse, the DFT-spectrum is not. The frequencies of the atoms in the DFT-matrix are integer multiples of  $f_s/N$  Hz (i.e., 4 kHz in the example), none of which coincides with the 400.4 kHz signal component. Therefore, the second signal qualifies as off-grid sparse.

Off-grid sparse signals have similar properties as approximately sparse signals, including a considerably reduced reconstruction quality [15], [25], [26]. Off-grid sparsity manifests itself as (spectral) leakage which is typical for representations of physical signals in finite discrete dictionaries. Therefore, it is an issue not only encountered in the DFT domain, but in other discrete transform domains as well [27]. Although extensions of the CS framework to handle continuous signal support were proposed [24], [26], [28], [29], they are based on parametrizing the sparsifying dictionary, which requires the decoding procedure to infer the sparsifying dictionary in the process of reconstructing the signal. The corresponding reconstruction procedure is naturally much more complex compared to standard CS where the dictionary is known.

In practical systems, some form of discretization has to be accepted. Therefore, leakage cannot be avoided entirely [30], but it can be brought to an acceptable level (e.g., below the quantization noise in digital signal processing systems). A straightforward solution is to increase the number of signal samples per block at a given sampling rate, which makes the frequency grid denser and therefore the resolution finer.

The graph at the bottom of Figure 2 again shows a spectrum of the same signal with the 300 kHz and the 400.4 kHz components, sampled at the same rate as before. However,

<sup>2</sup>According to the definition of  $\kappa$ -sparsity given in Section II-A a real-valued two-tone signal would qualify as 4-sparse signal since its DFT has four non-zero complex coefficients. However, due to the redundancy of the DFT two of these coefficients are enough to fully characterize such a signal.

the blocksize was increased 10-times, leading to the DFT having 10 times more atoms with a 10-times finer frequency spacing. In this case, each of the two signal frequencies coincides with one of these atoms, resulting again in an exactly sparse spectrum, as in the first example.

Increasing the blocksize is not desirable in the encoder, where the hardware complexity and the power consumption depend on it. Instead, we propose *multi-block decoding* (MBD) that increases the blocksize in the decoding stage, avoids the need to deal with an infinite signal support, and allows the blocksize in the encoder to remain within practical limits.

2) *Multi-Block Decoding*: MBD can be formally stated as follows:  $B$  consecutive measurement vectors obtained from the encoder are stacked into the column vector  $\mathbf{v}_B = [\mathbf{y}_1^\top \mathbf{y}_2^\top \cdots \mathbf{y}_B^\top]^\top$ . Mathematically,  $\mathbf{v}_B$  is obtained via

$$\mathbf{v}_B = \Phi_B \mathbf{u}_B, \quad (6)$$

where  $\mathbf{u}_B = [\mathbf{x}_1^\top \mathbf{x}_2^\top \cdots \mathbf{x}_B^\top]^\top$  is the column vector stacking the samples from  $B$  compression blocks, and  $\Phi_B = \mathbf{I}_B \otimes \Phi$  is the multi-block measurement matrix that is the Kronecker product between the identity matrix of size  $B \times B$  and the  $M \times N$  measurement matrix used in the encoder. Extending (3) correspondingly, a sequence of  $B \cdot N$  consecutive samples can be reconstructed solving

$$\text{minimize } \|\alpha\|_1 \quad \text{subject to } \|\mathbf{v}_B - \Phi_B \Psi_B \alpha\|_2 < \epsilon, \quad (7)$$

with  $\Psi_B \in \mathbb{C}^{B \cdot N \times P}$ , and  $\alpha \in \mathbb{C}^{P \times 1}$ .

Clearly, the compression factor remains the same for any choice of  $B$ , but considering—e.g.—the  $(B \cdot N) \times (B \cdot N)$  DFT matrix as sparsifying basis  $\Psi_B$ , it is easy to see that the frequency spectrum  $\alpha$  has  $B \times$  more entries with respect to single-block decoding, which results in a correspondingly increased spectral resolution.

It is worth noting that MBD has no additional cost in the encoder, but the size of the reconstruction problem increases with increasing number of blocks. The reconstruction problems occurring in this work are solved employing the orthogonal matching pursuit (OMP) algorithm [31]. Based on [31], one can show that the asymptotic computational complexity of MBD solved in  $M$  iterations of OMP is  $\mathcal{O}(B^2 MN + BM^4)$ , which means that decoding  $B$  blocks together using MBD can take up to  $B \times$  longer than decoding them individually when  $M$  is small. However, the run-times rapidly become equal with increasing  $M$  and up to a number of multiblocks exponentially growing with  $M$ . Indeed, in evaluations based on our reconstruction setup, consisting of the OMP algorithm implemented in Matlab software running on a general purpose PC, we found that the run-time was roughly the same in both cases for the problem sizes that occurred in the applications discussed in Section V.

In terms of related work, [26] should be mentioned, where the use of a **redundant DFT frame** was proposed to improve the spectral resolution in the single-block decoding context. In contrast to MBD where orthogonal bases are used, this approach increases the problem size only in one dimension, but suffers from coherency problems resulting in degrading performance with increasing number of blocks.

Considering the modest blocksizes encountered in practical sensing applications and that the overall reconstruction time is not necessarily affected by the use of MBD, it appears reasonable to avoid such issues by using MBD instead.

CS formulations that aim at reconstructing a block-wise encoded signal vector and share some similarity with MBD are *block CS* (B-CS) [32] and the *multi measurement vector* (MMV) approach [21]:

B-CS is aimed at speeding up the recovery by post-processing a large signal vector that was encoded and decoded sub-block-wise. In both MBD and B-CS the effective measurement operator is the Kronecker-product between an identity matrix and the measurement matrix used to encode the blocks, which leads to the same starting point for the recovery. However, in B-CS, recovery is performed in two stages of which the first consists of block-wise reconstruction of the signal vector, which not only introduces blocking artifacts, but also suffers from the effects of off-grid sparsity. Such artifacts do not occur in MBD and the effects of off-grid sparsity are mitigated by full-length signal reconstruction.

In MMV a number of measurement vectors—obtained from individually encoding consecutive blocks of a signal vector—is stacked in parallel to form a matrix. In contrast to MBD, the reconstruction problem has the form of a matrix equation, based on which the common support of the associated signal vectors is recovered in a first step. The knowledge of this support simplifies the reconstruction of the signal vector, that is performed in a block-wise manner in a second step. While MMV can improve the speed and robustness of the reconstruction, it leaves the granularity of the sparsifying basis unchanged, and therefore suffers from the effect of off-grid sparsity to the same extent as single-block decoding.

### III. HARDWARE ARCHITECTURE

The current sensor SoC monolithically integrates two independent current sensing cores, two analog-to-digital converters (ADCs), the digital multi-mode compressive sensing encoder (DCSE), and a standard 4-wire serial peripheral interface (SPI) for configuration and data transfer. A schematic overview of the SoC is shown in Figure 3.

#### A. Current Sensing Cores

The current to be sensed is directly fed through the chip via a copper strip realized in the top metal layer. The copper strip has a minimum width of  $180 \mu\text{m}$ , while eight  $30 \mu\text{m}$  thick copper bonding wires are used, enabling sensing of currents up to  $10 A_{\text{peak}}$ . The current is measured contactlessly by sensing its magnetic field based on two sensing cores, each consisting of one octagonal Hall-effect probe, a 4-phase current spinning bias generator, and a discrete-time analog front-end (AFE).

The Hall-effect probe is a magnetic sensor that—in contrast to other sensor solutions, such as, e.g., resistive shunts [33] or current transformers [34]—offers both low insertion losses and ease of integration in standard CMOS technology. The probes are placed symmetrically with respect to the copper strip, such that the magnetic field lines

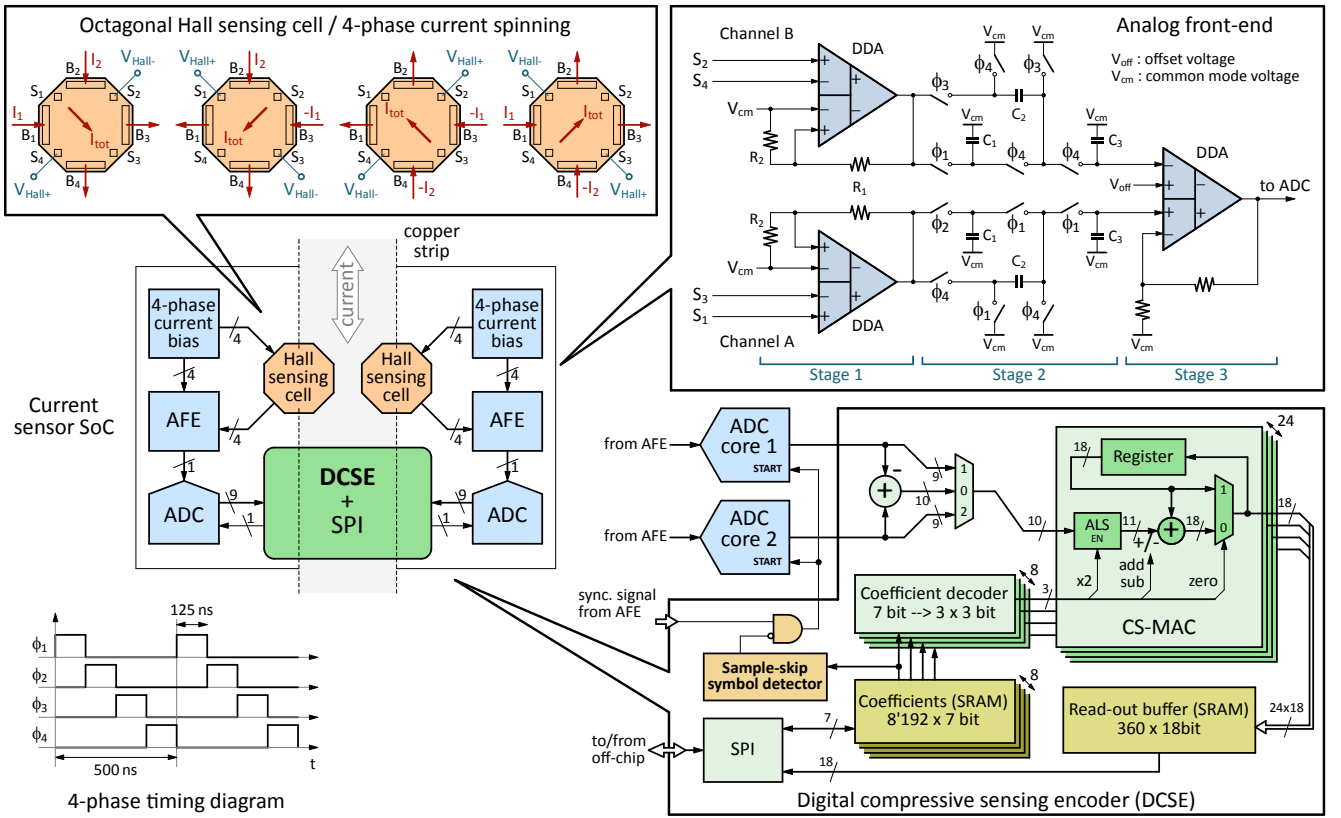


Fig. 3. System overview of the multi-mode compressive sensing current sensor SoC. Detailed views of the octagonal Hall sensing cell, the AFE and its timing diagram, and the digital compressive sensing encoder.

act on the sensing cores with opposite signs; hence, differential measurement can be realized. In this way, common-mode interference caused by the earth magnetic field or any other magnetic field not symmetric with the copper strip is canceled out. Temperature dependent sensitivity and offset impairing the accuracy are disadvantages of the Hall-effect probe. Although not implemented in the present version of the SoC, different temperature compensation methods were proposed to mitigate this drawback [35], [36]. Yet, in order to dynamically cancel out the high intrinsic offset of the Hall-effect probe, a fast 4-phase current-spinning technique was implemented [37]. The application of current-spinning to the octagonal Hall-effect probe is illustrated in Figure 3. It requires the concurrent injection of two bias currents  $I_1$  and  $I_2$  into adjacent bias contacts in each phase (e.g.,  $B_1 - B_2$ ), so that the overall effective current ( $I_{tot} = I_1 + I_2$ ) flows always along a diagonal axis of the probe. The magnetic field lines traverse the probe perpendicular to the plane of the probe, which results in a measurable potential difference (i.e., the Hall voltage) in the plane and in the direction perpendicular to the effective current, i.e., between two opposite sense contacts (e.g.,  $S_2 - S_4$ ). In each of the 4 phases, the direction of one of the bias currents is inverted such that the total effective current  $I_{tot}$  is rotated in the plane by 90 degree. Due to the discrete-time nature of this offset-reduction technique, the spinning frequency  $f_{spin}$ —i.e. the frequency at which the bias current is rotated—limits the Nyquist bandwidth to  $B_W \leq f_{spin}/8$  [13]; thus, high

spinning-frequency is mandatory to achieve broadband sensing. Since  $f_{spin}$  is linked via the 4 phases to the main clock frequency  $f_{clk}$  by  $f_{spin} = f_{clk}/4$ , the Nyquist acquisition bandwidth can be adapted to the application by changing the main clock frequency. To achieve the target Nyquist bandwidth  $B_W = 1$  MHz, a main clock speed of 32 MHz was selected. This relationship between clock frequency and Nyquist bandwidth can be exploited to trade-off bandwidth for resolution [38] but does not affect the power consumption of the sensing cores.

On the other hand, the analog bandwidth of the Hall-effect probe is defined by the capacitive load seen by the probe, which is primarily given by the input load of the AFE [39]–[42]. To achieve the target bandwidth, an AFE with reduced input capacitance was implemented [38]. The architecture of the AFE is shown in Figure 3. It is subdivided into two acquisition channels—channel A and channel B—that work in time-interleaved fashion, and are controlled by the signals  $\phi_1, \phi_2, \phi_3, \phi_4$  that also control current-spinning.

The first stage of each channel is a non-inverting differential difference amplifier (DDA) [43] with gain  $G_1 = (1 + R_1/R_2)$ . The high input impedance and low input capacitance of this stage ( $<1$  pF) ensure broadband operation of the Hall-effect probe. The second stage is a switched capacitor circuit and serves the purposes of storing the measured voltage during each spinning phase, and of cancelling both the  $1/f$  noise and the offset of the DDAs via auto-zeroing. In the third stage, the output voltages of the two



channels are summed-up, which cancels the intrinsic offset of the Hall-effect probe. The resulting voltage  $V_{OUT}$  is fed to the ADC, and is proportional to the Hall voltage  $V_H$  as follows  $V_{OUT} = 2G_1G_2V_H$ , where  $G_2$  is the gain of the third stage. A more detailed description of the Hall-effect probe and the AFE can be found in [11], [38].

The conditioned Hall voltages from the two sensor cores are digitized by two ADCs, one for each core. The ADCs are instances of a 9-bit SAR ADC from a proprietary library, and are operated at  $f_{clk}$ . A single conversion requires 16 clock cycles, resulting in a maximum sampling rate of 2MS/s per core.

### B. Digital Multi-Mode CS Encoder

The digital multi-mode CS encoder (DCSE) implements the CS encoding stage discussed in Section II-A. To allow the current sensor SoC to be employed in the widest variety of applications, the CS encoder is required to handle different types of signals, possibly having varying degree of sparsity and requiring different sparsifying bases. In order to provide the necessary flexibility, the multi-mode encoder was designed as matrix-vector multiplier employing parallel multiply-accumulate (MAC) units and programmable memory to store a user-defined measurement matrix.

The architecture of the DCSE is shown in detail in Figure 3. The encoder compresses  $N$  consecutive 9-bit samples from the ADCs into  $M$   $Q$ -bit wide CS measurements in a streaming fashion at 2MS/s. The digital logic operates at the SoC main clock speed of 32 MHz, making 16 clock cycles available for the encoding of a new sample. Only 15 cycles are actually used to calculate the product of a new sample with the corresponding column of the measurement matrix and accumulate the result vector with the intermediate CS measurements from the previous encoding round (see (1) in Section II-A). The remaining cycle is reserved for a coefficient pre-fetch required to realize HSS. The completed CS measurements are available after  $N$  sampling instants, and are transferred into a read-out buffer that allows the compressed data to be read-out via the SPI interface without interrupting the encoding process.

The MAC unit supports the reduced CS measurement coefficient set  $\{-2, -1, 0, 1, 2\}$ , which allows all necessary multiplications to be implemented with hardware friendly bit-shift and add/subtract operations; the implementation of full four-quadrant multipliers can be avoided. This particular set allows us to realize all encoding modes discussed in Section II-C and, in addition, offers support for advanced compressive signal processing methods, such as punctured estimation proposed in [17]. The digital representation of the 5-valued coefficients would require 3 bits per coefficient, if the coefficients were to be stored individually. Instead, encoding the 125 possible values of three coefficients together into a single 7 bit digital word, results in 22% memory saving compared to individual storage.

The DCSE implements hardware sample skipping (HSS) by masking the start signal to the ADCs whenever a sample is not used. Although not supported by the AFE and ADC in this SoC, the same masking signal could be used to power

duty-cycle the AFE and the ADC to save power in the HSS mode. The sample-skip symbol detector (see Figure 3) impedes the ADC from taking and converting the next signal sample, when it detects a special HSS symbol in the measurement matrix during the pre-fetch phase. The HSS symbol is a reserved 7-bit coefficient value that is not used for encoding.

The coefficient memory size is directly proportional to the maximum blocksize  $N_{max}$  and maximum number of measurements  $M_{max}$  supported by the DCSE. The blocksize and the number of measurements required to capture a signal of a certain sparsity  $\kappa$  are linked via (5). In practice,  $\kappa$  is unknown and the  $N_{max}/M_{max}$ -ratio that yields the optimal compression/quality trade-off must be evaluated empirically for a specific type of signal. Instead, when designing a general purpose CS encoder, the  $N_{max}/M_{max}$ -ratio must be designed for the worst case which is characterized by a compression factor of 1. Beyond that, the design rules for  $N_{max}$  and  $M_{max}$  in absolute terms are highly case specific, and may depend on factors such as power budget, fabrication cost, and other application constraints. In our DCSE,  $N_{max}$  and  $M_{max}$  were selected as follows: Since there are 15 cycles for encoding and the measurement coefficients are read from memory in batches of three,  $M_{max}$  was selected to be an integer-multiple of 45. For efficient addressing the multiple was chosen to be 8, resulting in  $M_{max}=360$ . Consequently, 24 parallel multiply-accumulate (MAC) units are required for encoding  $M_{max}$  measurements in 15 cycles.

The effective compression factor (CF) according to (4) is  $CF = (9N)/(QM)$ , where  $Q$  and 9 are the wordwidths of the measurements and of the raw signal samples, respectively. The DCSE allows  $Q$  to be configured between 9 bits and 18 bits, which enables the compression factor to be optimized in each CS mode; more specifically, in the NUS mode,  $Q = 9$  bit can be set safely because—in contrast to the RM mode where  $N$  samples are being accumulated—there is no risk of measurement saturation, whereas in the RM mode the risk of measurement saturation can be traded against a better compression factor by selecting  $9 \text{ bit} \leq Q \leq 18 \text{ bit}$ . In [15],  $Q = 9 + \log_2(N)/2 + 1$  bit was found to provide a good trade-off. The DCSE was designed with a predefined and fixed silicon area budget that allowed  $N_{max}=546$  to be realized. The effective blocksize  $N$  and number of measurements  $M$  used during encoding can be selected accordingly in the range from 1 to 546 and in the range from 1 and 360, respectively.

The SRAM storing the measurement matrix is 56 KiB in size, while 0.8 KiB of SRAM are used for the read-out buffer that ensures continuous encoding operation.

DCSE configuration, coefficient loading, and measurement read-out is realized via a standard 4-wire serial peripheral interface (SPI). The same interface alternatively allows the uncompressed raw data to be read from the ADCs directly.

## IV. IMPLEMENTATION RESULTS

The current sensor SoC was implemented and fabricated in a 0.16- $\mu\text{m}$  BCD (Bipolar-CMOS-DMOS) technology, using CMOS devices only. The micrograph of the SoC is shown

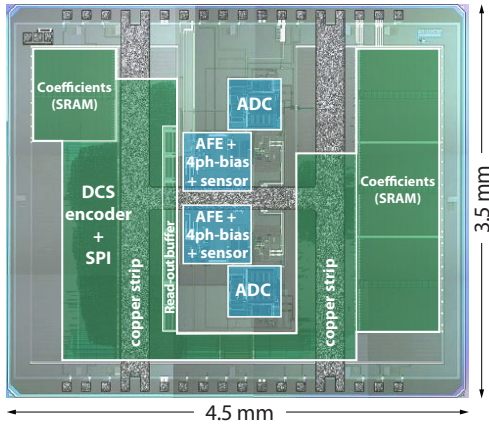


Fig. 4. Annotated chip micrograph of the Hall sensor SoC.

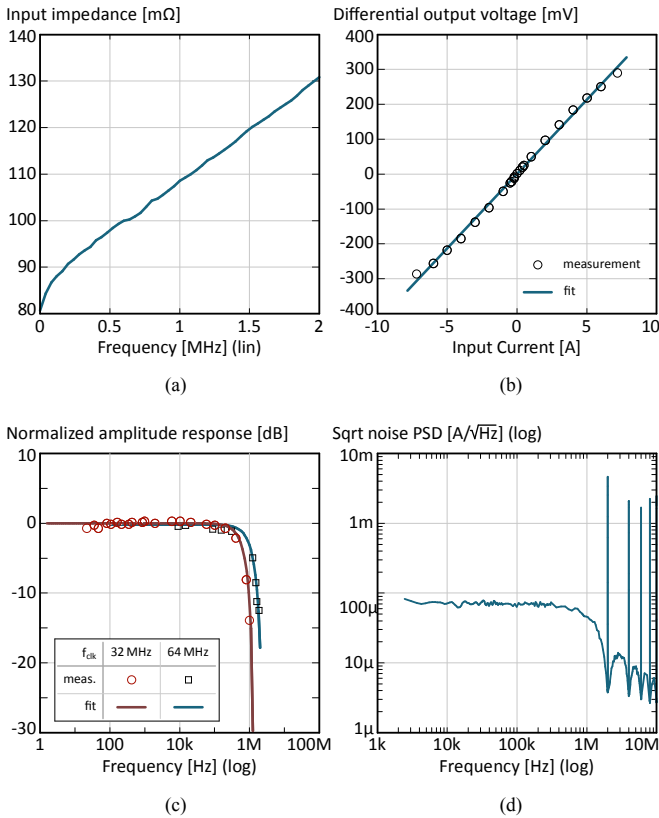


Fig. 5. AFE measurement results: (a) input impedance vs. input frequency, (b) input current to differential output voltage transfer function, (c) normalized amplitude response, and (d) input-referred noise power spectral density.

in Figure 4 with the various functional blocks highlighted and labeled. The chip was packaged into a plastic power small outline (PwSO) package that allows sufficient heat dissipation when measuring large currents beyond 4 A.

Figure 5 reports main measurement results of the analog section of the SoC. The resistance across the current input terminals of the SoC is 80 mΩ, including thick top metal copper strip, bonding wires and package contacts. The low input resistance makes the parasitic inductance of the bonding wires significant at high frequency, which was taken into account in the evaluation of the sensor accuracy. Figure 5(b) shows the static characteristic of the analog section in the  $\pm 7$ -A range, which roughly corresponds to 10  $A_{\text{peak}}$ . The

TABLE I  
COMPARISON

|                                   | This work          | [44]               | [33]             | [45]      |
|-----------------------------------|--------------------|--------------------|------------------|-----------|
| CMOS technology [ $\mu\text{m}$ ] | 0.16               | 0.18               | 0.13             | 0.18      |
| Supply voltage [V]                | 1.8                | 1.8                | 1.5              | 1.8       |
| Probe type                        | Hall only          | Hall+coil          | R shunt          | Hall only |
| Measured peak current [A]         | 10                 | n.a.               | 5                | n.a.      |
| Nyquist Bandwidth [kHz]           | 1000               | 3000 <sup>1)</sup> | 50               | 62.5      |
| Effective resolution [bit]        | 6.4                | 5                  | 14 <sup>2)</sup> | n.a.      |
| Non-linearity error [%]           | 3                  | n.a.               | n.a.             | < 0.2     |
| AFE power cons. [mW]              | 4.5 <sup>3)</sup>  | 13.9               | 0.053            | 0.12      |
| Area of analog [ $\text{mm}^2$ ]  | 0.60 <sup>2)</sup> | 8.75               | 1.15             | 1.16      |
| ADC integration                   | yes                | no                 | yes              | no        |
| ADC resolution                    | 9                  | n.a.               | 14               | n.a.      |
| ADC power cons. [mW]              | 5                  | n.a.               | 0.03             | n.a.      |
| Digital data compression          | CS, on-chip        | no                 | no               | no        |

1) by using coils; 2) ADC-limited; 3) per sensing core

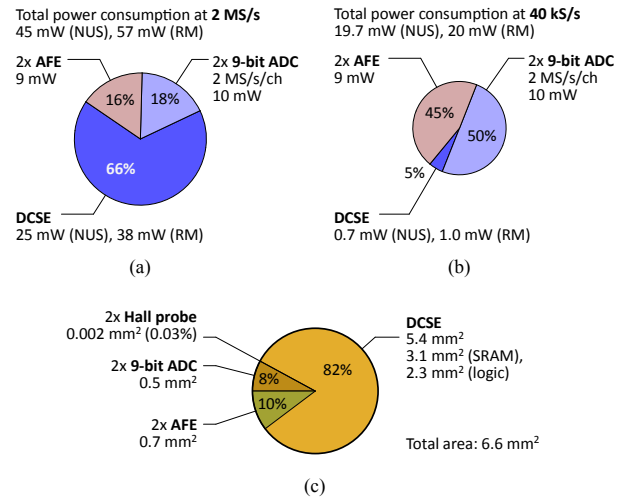


Fig. 6. Implementation summary: (a) power breakdown at 2 MS/s, (b) power breakdown at 40 kS/s, and (c) area breakdown.

sensor and the AFE demonstrate a good linearity with a maximum non-linearity error of less than 3% after linear calibration. The normalized amplitude response is reported in Figure 5(c), and shows the  $-3$  dB bandwidth at roughly 600 kHz and at 1 MHz corresponding to the system being operated at 32 MHz and 64 MHz, respectively. The Nyquist bandwidth of the SoC is 1 MHz at 32 MHz clock speed, which allows us to compensate the amplitude response by digital post-processing [46], [47] to achieve an effective 1 MHz bandwidth at 32 MHz (i.e., the nominal operating speed of the SoC). Figure 5(d) shows the input-referred noise spectrum which is flat up to 1 MHz with a root mean square value of 70  $\text{mA}_{\text{RMS}}$ , leading to an effective resolution of 6.4 bit. The main parameters characterizing the AFE are summarized in Table I for comparison with state-of-the-art current sensors, while a more detailed characterization of the AFE can be found in [38].

Detailed area and power breakdowns are shown as pie charts in Figure 6. The complete SoC occupies a total area of 16  $\text{mm}^2$  of which 6.6  $\text{mm}^2$  is active. A single current sensing core (i.e., Hall probe, spinning-current generator, and AFE) occupies 0.6  $\text{mm}^2$  and consumes 4.6 mW from a 1.8 V supply. Each ADC occupies 0.24  $\text{mm}^2$ , and consumes

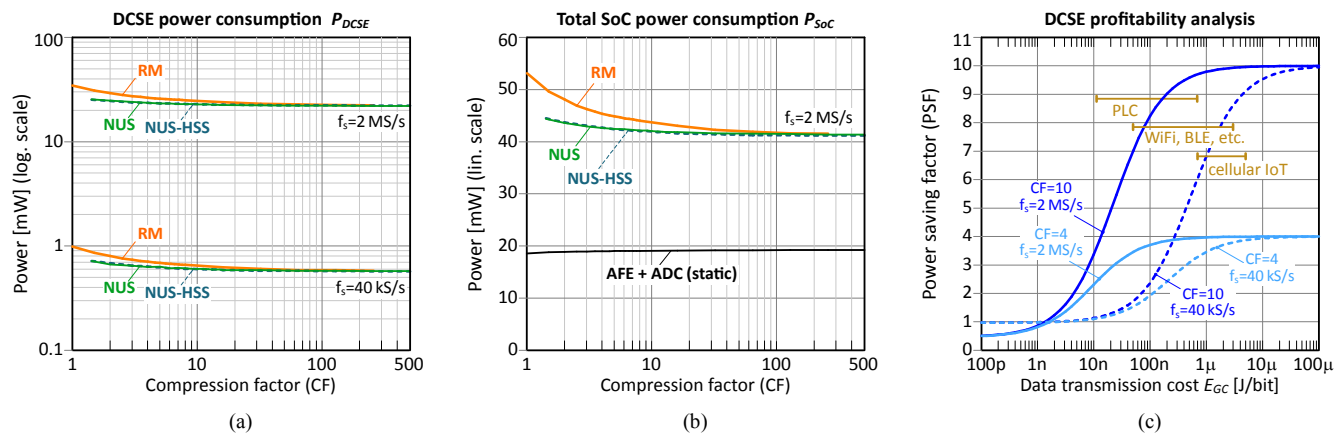


Fig. 7. Power consumption vs. compression factor for binary RM and NUS; (a) SoC total consumption at 2 MS/s, and (b) DCSE consumption at 2 MS/s and at 40 kS/s. (c) PSF vs. energy cost of data transmission (PLC=power-line communication, BLE=bluetooth low energy, IoT=Internet of Things).

5 mW from a 1.8 V supply. The DCSE occupies 5.4 mm<sup>2</sup> with the SRAM memory accounting for 3.1 mm<sup>2</sup> or 57% of the DCSE. The power consumption of the DCSE depends on the encoding mode and on the selected compression factor.

Figure 7(a) reports the consumption of the DCSE, at the nominal sampling rate of 2 MS/s and at 40 kS/s which is the rate used in the application example discussed in Section V-B while the total consumption of the SoC at the nominal sampling rate is shown in Figure 7(b). Under nominal conditions, with the SoC running at 32 MHz, the DCSE power consumption scales with increasing compression factor from 38 mW down to 22 mW in RM-encoding mode (RM), and from 25 mW down to 22 mW in NUS-mode (NUS). The DCSE calculates the maximum number of CS measurements irrespective of the actual  $M$  setting. Without this design flaw, the decrease in power consumption of the DCSE with increasing CF would be stronger. The current version of the SoC does not support power-duty-cycling of the AFEs and ADCs the consumption of which is predominantly static (cf. Figure 6(a,b)). This shortcoming prevents us from taking full advantage of HSS that would allow the static consumption of the analog section to be reduced via duty-cycling whenever encoding modes based on sample-skipping are being used. As a consequence, Figure 7(a) shows practically identical total power consumption of the SoC in the NUS and NUS-HSS modes, whereas with the ability to power-duty-cycle the AFEs and ADCs the total SoC power consumption in the HSS mode would decrease approximately linearly with increasing CF.

A reduction of the data-rate achieved by the DCSE can have several benefits, e.g., it can relax the requirements of the down-stream electronics interfacing the SoC, resulting in lower system design complexity and lower cost, or it can reduce the cost of data transmission if the SoC is employed in a remote sensor context. To assess the profitability of the DCSE with regard to the second case we define and evaluate the *power saving factor* (PSF):

$$\text{PSF} = \frac{P_{SoC} - P_{DCSE} + E_{GC} f_s B}{P_{SoC} + E_{GC} f_s B / CF}, \quad (8)$$

where  $P_{DCSE}$  and  $P_{SoC}$  are the power figures reported in Figure 7(a) and (b), respectively, while  $f_s$  is the sampling

frequency,  $B = 9 \text{ bit}$  is the wordwidth of the raw signal samples, and  $E_{GC}$  is the energy-per-bit-cost of a generic communication circuit (GC). The denominator is the power consumption of a system consisting of our SoC and a GC transmitting the compressed data produced by the DCSE, whereas the numerator is the consumption of the same system with the DCSE switched off and the GC transmitting uncompressed data. The PSF quantifies the factor by which the power consumptions of the system including SoC and GC is reduced by operating the DCSE. Thus, a PSF larger than unity means that the power cost of the DCSE is outweighed by the savings in transmission power due to the data-rate reduction achieved by the DCSE, indicating that the DCSE is profitable. Figure 7(c) plots the PSF at exemplary sampling rates and CFs against  $E_{GC}$  in a range including the approximate cost of typical data transmission methods used in smart-metering applications ([48] and references therein). We note that at a given  $E_{GC}$  the PSF increases both with the sampling rate and the CF, and that the maximum power reduction achievable is a factor of CF. The results show that the DCSE proposed here is profitable throughout the range of parameter values to be expected in remote sensing.

## V. APPLICATION EXAMPLES

In this section, we demonstrate the multi-mode compressive sensing current sensor SoC in two exemplary applications: First, acquisition of sparse current waveforms, and second, household appliance state detection for non-intrusive load monitoring based on compressive measurements.

### A. Acquisition of Sparse Current Waveforms

To characterize the trade-off between quality and data-rate reduction achievable with our CS codec applied to generic current waveforms, we performed a series of experiments with different encoding modes, varying the CF. The experiments were carried out using a hardware testbed built around the SoC prototype for performance and power measurements, and for end-to-end tests. The OMP algorithm was used for CS reconstruction. The quality is reported in terms of *signal-to-reconstruction-error-ratio* (SRER) that measures the ratio

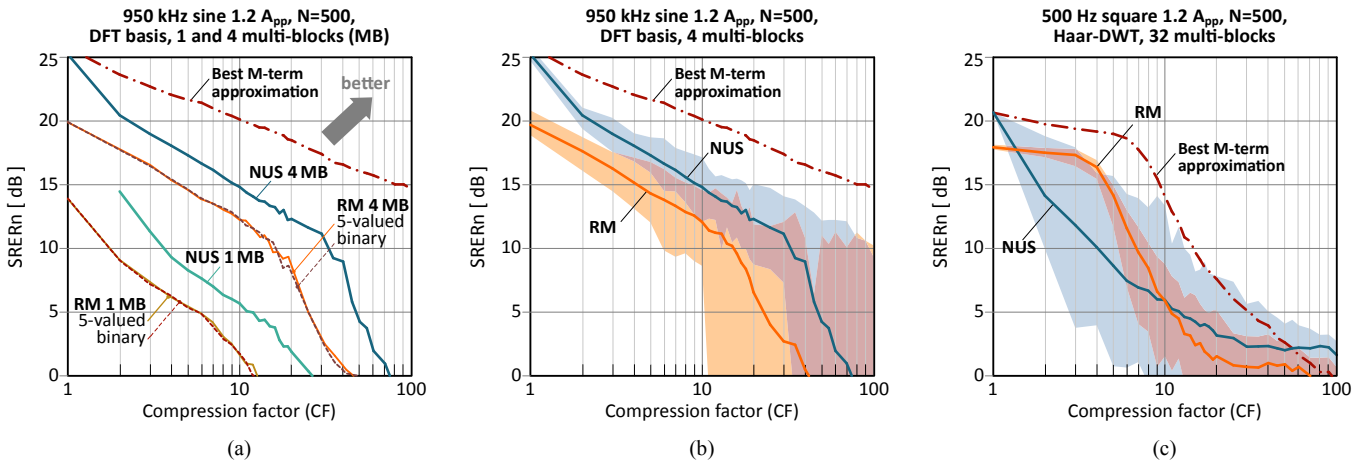


Fig. 8. Reconstruction quality vs. compression factor: (a) the effect of multi-block decoding; (b) and (c) comparison of RM and NUS encoding for a DFT-sparse and the Haar-DWT-sparse signal, respectively.

between the signal energies of the mean-free test signal and of the reconstruction error:

$$\text{SRER} = 20 \log_{10}(\|s - \bar{s}\|_2 / \|s - \hat{s}\|_2), \quad (9)$$

where  $\bar{s}$  is the mean-value of the test signal  $s$ , and  $\hat{s}$  is the reconstructed signal. Qualitatively, the higher the SRER value, the lower the reconstruction error.

In Figure 8, we report the reconstruction quality with respect to the CF for different types and bandwidths of input current signals, for both NUS and RM encoding. Due to the signal-dependent performance of the recovery algorithm, the reconstruction quality was measured for each CF setting, performing 100 separate encoding-decoding runs. For each CF setting a new measurement matrix was loaded into the SoC, and remained unchanged for all 100 codec runs. As a quality reference we show the SRER of the *best-M-term approximation* that is an approximation of the original signal based on the  $M$  largest coefficients in the sparsifying domain, which is the best approximation in the  $L_2$ -sense. We note that the achieved quality/compression trade-off is close to the optimum that can be expected from CS-based data compression for the test signals at hand.

Figure 8(a,b) show the reconstruction results of a sinusoidal test current of  $1.2 A_{pp}$  at 950 kHz, while Figure 8(c) shows the results of compressively measuring a  $1.2 A_{pp}$  square-wave current with a 2 ms period. As sparsifying bases the DFT was used for the sinusoid and the Haar discrete wavelet transform (Haar-DWT) for the square-wave. In Figure 8(a) the effect of multi-block decoding is shown. When decoded using standard block-wise CS reconstruction using a blocksize of  $N = 500$ , the signal is off-grid sparse in the DFT basis, resulting in a poor quality/compression trade-off in both RM and NUS encoding modes. In contrast, when multi-block decoding (MBD) with four blocks is used, the quality/compression trade-off is improved by 8 dB to 10 dB at a given CF in both encoding modes, which demonstrates the effectiveness of MBD.

Figure 8(a) also reveals that the reconstruction quality based on multi-valued and binary-valued RM encoded data is virtually identical, confirming earlier studies [3], [49]. Thus,

for the sole purpose of CS-based data compression binary-valued coefficient support is sufficient. The overhead in terms of memory size for supporting 5-valued coefficients instead of 3-valued and binary coefficients is 40% and 130%, respectively. Yet, [17] and the references therein have investigated into the fusion of classical estimation and filtering with CS encoding relying on multi-valued coefficient sets. The multi-valued coefficient support and the programmability of the measurement matrix enable the DCSE to be employed for CS-based signal processing beyond data compression. The exploration of such methods is not covered here.

Referring to Figure 8(b,c), the solid line labeled with RM and NUS show the average reconstruction quality, while the shaded area around it corresponds to the range of qualities obtained in the codec runs. In Figure 8(b) we observe that, in the best case, high reconstruction quality can be achieved up to a CF in excess of 100 in both modes, while the minimum quality drops at a CF of around 10 and 30 in the RM and NUS mode, respectively. Although in practice the worst-case performance maybe more relevant—e.g., to assess the robustness of the CS codec—it is easier to identify trends looking at the average performance. Indeed, the almost identical shape of the average quality curve of NUS and RM indicates that both modes are equally well suited to encode DFT-sparse signals. However, NUS achieves roughly twice the CF at the same average quality as RM, which is due to the CS measurement being 18 bit in RM but only 9 bit in NUS (see Section III-B for details).

The situation is different for the square-wave test signal; Figure 8(c) shows that the quality of RM drops earlier, at around a CF of 4, while NUS has a much larger spread and a steady drop in quality already at low CFs. The reason is that the NUS encoding matrix is more coherent with the Haar-DWT basis than RM, making it less suitable for signals that are sparse in this basis. Intuitively, the sample-skipping of NUS can lead to missing important signal information such as a sudden signal level change. The difference in quality/compression trade-off between Figure 8(b) and Figure 8(c) is evidence of the signal-dependent performance of the CS codec. However, in both cases robust reconstruction

**Algorithm 1** Modified OMP (MOMP)

---

```

1: input  $\mathbf{v}, \Phi_B, \Psi_B$ 
2:  $\mathbf{P} = \Phi_B \Psi_B, \mathbf{r}_0 = \mathbf{v}, \alpha_0 = \mathbf{0}_{B \cdot N \times 1}, \Gamma_0 = \emptyset$ 
3:  $\mathbf{Q} \leftarrow \mathbf{P}$  column-wise normalized
4: while  $k = 1, \dots, M$  do
5:    $c \leftarrow \arg \max_i |\mathbf{r}_{k-1}^\top \mathbf{q}_i|$ 
6:    $\Gamma_k \leftarrow \Gamma_{k-1} \cup \{c\}$ 
7:    $\alpha_{k,\Gamma} \leftarrow \arg \min_z \|\mathbf{P}_{\Gamma_k} \mathbf{z} - \mathbf{v}\|_2$  {least-squares approx.}
8:    $\mathbf{r}_k \leftarrow \mathbf{v} - \mathbf{P} \alpha_k$ 
9: end while
10: return  $\alpha_k$ 

```

---

is achieved with SRERs between 10 dB and 25 dB.

In summary, the results show that the NUS encoding mode—if applicable—achieves a better quality/compression trade-off than RM. Yet, for a practical CS codec it is useful to support RM encoding to be able to capture signals that are characterized by sudden or rare changes.

### B. Household Appliance Detection

To demonstrate the versatility of our SoC and of the CS codec, we study its application in a smart utility meter for non-intrusive load monitoring (NILM) [50]. Identifying the type and the state of connected electrical appliances in a building or household enables inhabitants to have better control over energy usage, and allows utility providers to optimize their services [51].

In contrast to monitoring each appliance individually, in NILM, each appliance and its state are detected by observing the current and voltage waveforms at the mains entry point of the building only. This approach greatly reduces the cost for installation and maintenance of the monitoring system, but requires load disambiguation which represents a non-trivial signal processing problem. A large variety of methods leveraging different types of distinguishing features exist [52], [53], among which the method based on compressive sensing—proposed in [54] and further refined in [55]—is of particular interest in this application example. The experiments reported here are inspired by these prior efforts, and extend this line of work with a compressive domain mains cancellation method and a hardware demonstrator based on the current sensor SoC prototype.

1) *Compressive Sensing Appliance Detection*: The approach of [54] and [55] assumes that only a small number of connected devices is active at any given time. Thus, the NILM information is sparse in the “appliance domain”, i.e., the domain of appliance signatures represented by a dictionary containing characteristic current waveforms uniquely identifying a specific appliance.

Same as [54] and [55], we use averaged steady-state current waveform of each appliance observed during one mains period as signatures. The bandwidth of these signatures was found to be no larger than 20 kHz, which allows us to run the SoC at a sampling rate of 40 kS/s (i.e., with a 640 kHz clock). To obtain CS measurements that encode the 800 signal samples spanning an entire mains period, we employ

**Algorithm 2** CS-based Appliance Detection

---

```

1: input  $\mathbf{v}, \Phi_B, \Psi_B, f_s$ 
2:  $\mathbf{t} \leftarrow [0, \dots, (NB - 1)]^\top / (NB)$  {the time base}
3:  $N_m \leftarrow 50 NB / f_s$  {number of encoded mains periods}
4:  $\mathbf{e} \leftarrow \exp(i 2\pi N_m \mathbf{t})$  {complex exponential at 50Hz}
5:  $c \leftarrow \mathbf{v}^\top (\Phi_B \cdot \mathbf{e}) / \|\Phi_B \cdot \mathbf{e}\|_2^2$  {est. offset and amplitude}
6:  $\mathbf{w} \leftarrow 2 \Re\{c^* \cdot \mathbf{e}\}$  {reconstructed mains osc.}
7:  $\hat{\mathbf{v}} \leftarrow \mathbf{v} - \Phi_B \cdot \mathbf{w}$  {measurements without mains osc.}
8:  $s \leftarrow \arg \max_{2 \leq i \leq NB} \{w_i - w_{i-1}\}$  {find pos. zero-crossing}
9:  $\hat{\Psi}_B \leftarrow \Psi_B$  column-wise circularly-shifted by  $s$ 
10:  $\alpha \leftarrow \text{MOMP}(\hat{\mathbf{v}}, \Phi_B, \hat{\Psi}_B)$ 
11: return  $\alpha$ 

```

---

MBD using  $N = 200$  and  $B = 4$ . Referring to the MBD formulation from Section II-D2, we define  $\Psi_B \in \mathbb{R}^{(B \cdot N) \times P}$  as the dictionary containing signatures of length  $B \cdot N = 800$  from  $P$  different appliance types as columns. We specify  $\alpha$  as a vector of non-negative integers with each of its  $P$  entries being the number of active devices of the corresponding type of appliance. To reconstruct  $\alpha$  we use the modified OMP algorithm detailed in Algorithm 1, with the only modification being the additional rounding operation in line 8 to enforce integer-valued results.

In [54] it was shown that canceling the 50 Hz mains oscillation in both the appliance signatures and the measured current before encoding improves the detection performance significantly. Since in our DCSE no such pre-filtering is available, we instead propose the procedure described in Algorithm 2 to cancel the mains oscillation directly in the compressive domain and based on the compressive measurements only. More specifically, a complex exponential at 50 Hz is encoded (line 3) using the same CS measurement matrix used in the encoder. Correlating the result with the CS measurements obtained from the encoder yields a complex coefficient  $c$  representing an estimate of the mains phase offset and amplitude (line 4). The mains oscillation can then be reconstructed (line 5), encoded again, and subtracted from the CS measurements (line 6). The result is an estimate of the CS measurements that would have been obtained from encoding the signal after mains cancellation.

Knowing the mains phase offset, the dictionary does not have to include all possible phase-shifted versions of each appliance signature, which massively reduces the size of  $\Psi$ . It is enough to include a single signature per appliance and then circularly shift each column of  $\Psi$  by the estimated phase offset (line 8) to obtain a dictionary of signatures with the correct phase offset. The phase offset  $s$  is calculated from the reconstructed mains oscillation by calculating the index of its maximum slope (line 7) which corresponds to the positive zero-crossing point.

The extraction of an appliance signature from a recorded current waveform can be done following steps 1 to 7 of Algorithm 2, with  $\mathbf{v}$  containing the unencoded signal samples and with an identity matrix of appropriate size instead of  $\Phi_B$ . Setting  $B = N_R + 1$  and following this procedure,  $N_R$  complete signatures can be extracted from  $\hat{\mathbf{v}}$  starting at the

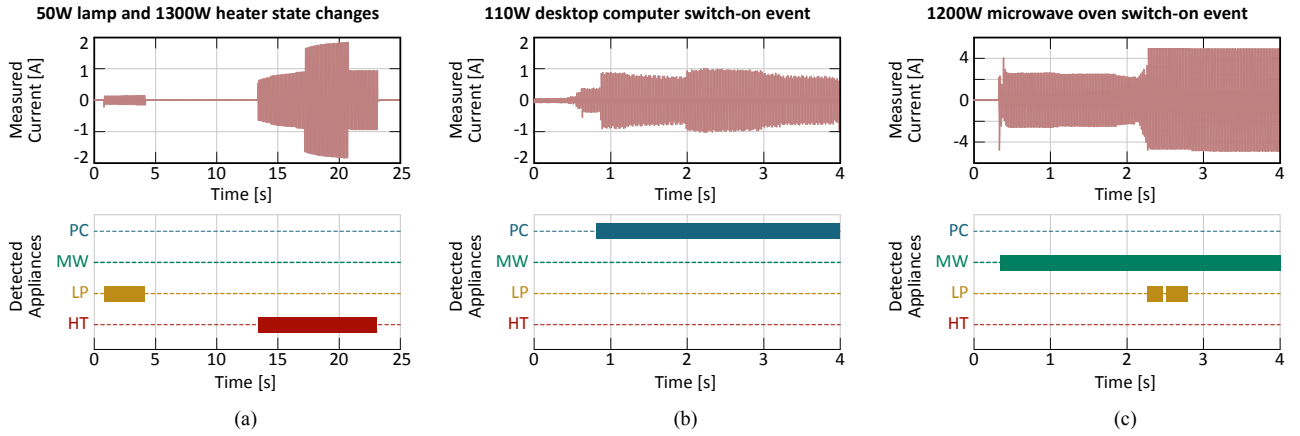


Fig. 9. CS-based appliance detection examples. Setup: NUS encoding,  $N=200$ ,  $M=10$ ,  $CF=20$ ,  $B=4$  multi-block decoding.

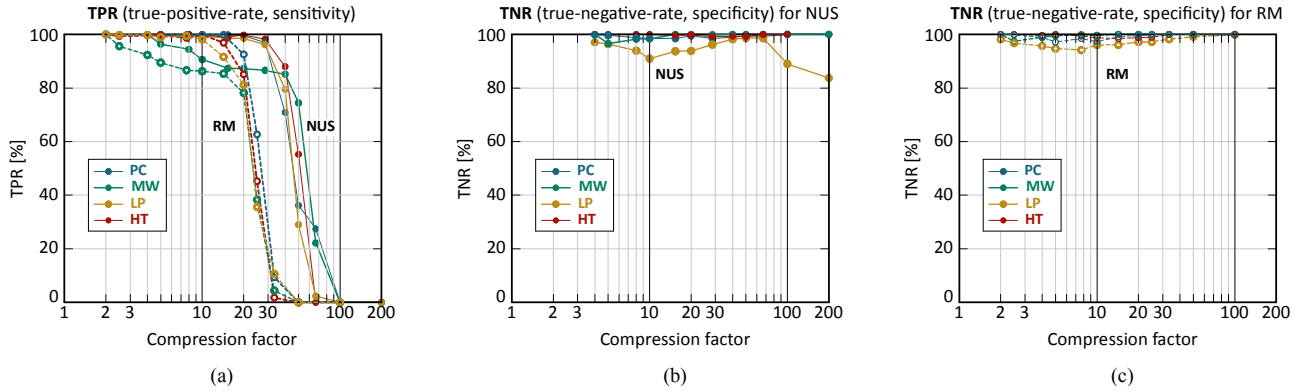


Fig. 10. CS-based appliance detection performance vs. CF: (a) sensitivity of both RM and NUS; (b) and (c) specificity of NUS and RM, respectively.

estimated phase offset  $s$ . Finally, the extracted signatures can be averaged to obtain the final signature in the column of  $\Psi_B$  corresponding to that specific appliance.

2) *End-to-End Tests*: The CS-based appliance detection method was implemented **on the same hardware testbed used in the previous application example**. The appliance dictionary was constructed from 18 signatures from 4 different common household appliances; namely, a 50 W lamp, a 1300 W electric heater, a 110 W desktop computer, and a 1200 W microwave oven. The CS acquisition was run continuously over the duration of several minutes, while driving the pre-recorded current waveforms through the current sensing SoC, **using an arbitrary waveform generator**.

The detection resolution of Algorithm 2 is one mains period which is much shorter than the minimum time a household appliance remains in a specific state. Therefore, short-lived false positives and false negatives in the output from Algorithm 2 were eliminated using a median filter with length 25, corresponding to half a second in real-time.

Figure 9 shows exemplary current waveforms (at the top) with the corresponding detection results using NUS encoding with a CF of 20 (at the bottom). In all three cases the correct appliance was detected reliably with a short occurrence of falsely detected lamp activity in Figure 9(c).

To quantify the performance of the compressive appliance detection scheme as a function of CF, the true-positive-rate (TPR) and the true-negative-rate (TNR) were evaluated over

several minutes (i.e., several thousand mains cycles). **The TPR quantifies the sensitivity in percent and is defined as**

$$\text{TPR} = \frac{\#(\text{correctly-detected-active})}{\#(\text{effectively-active})} \cdot 100\%, \quad (10)$$

**whereas the TNR quantifies the specificity in percent and is defined as**

$$\text{TNR} = \frac{\#(\text{correctly-detected-inactive})}{\#(\text{effectively-inactive})} \cdot 100\% \quad (11)$$

Reliable detection is characterized by both a high TPR and a high TNR value. The results of the reliability analysis are reported in Figure 10, and demonstrate robust detection performance with both sensitivity and specificity of over 95% up to CFs around 16, and up to CFs around 30 for NUS. The somewhat lower TPR of the microwave oven can be explained by the fact that this appliance goes through a sequence of different internal states not all of which were captured in the manual signature generation phase. The detection performance could be improved by completing the set of signatures.

For this experiment, the SoC was operating at 640 kHz (40 kS/s), consuming between 20 mW and 22 mW, 95% of which is due to the consumption of the AFEs and ADCs (see Figure 6(b)). At this speed, the interface raw data-rate is reduced by a factor of 25 from 18 Mbit/s at nominal speed to 360 kbit/s. Using the DCSE in NUS mode, the data-rate is further reduced by a factor of 30 to 12 kbit/s.

## VI. CONCLUSION

We presented a compressive sensing (CS) current sensor system-on-chip (SoC) designed and fabricated in STM 0.16  $\mu\text{m}$  BCD technology. The SoC includes two broadband, Hall-effect based current sensing cores, two ADCs, and a digital multi-mode CS encoder (DCSE) for data-rate reduction. The system offers a sensing bandwidth of 1 MHz and allows currents with amplitudes of up to  $10 A_{\text{peak}}$  to be measured with over 6 bit effective resolution. It occupies  $6.6 \text{ mm}^2$  active area while consuming less than 57 mW from a 1.8 V supply when operated at 32 MHz.

In this paper, we focused on the details of the DCSE design and the evaluation of CS as data compression codec for current sensing applications. Different encoding modes were compared in terms of reconstruction-quality/compression trade-off and power consumption. **On the decoding side, the problem of off-grid sparsity was pointed out and multi-block decoding was introduced as a method to mitigate it.**

The SoC and CS codec were tested and measured in two application case studies; first, compressive acquisition of sparse current waveforms, and second, household appliance detection for non-intrusive load-monitoring using CS measurements as features and sparse-reconstruction as classifier. The first case study has demonstrated the effectiveness of multi-block decoding to improve the reconstruction quality of off-grid-sparse signals. It was shown that the CS codec can achieve robust data compression with SRERs between 10 dB and 25 dB and a signal-dependent data-rate reduction reaching up to factors of several tens. It was found that the NUS encoding mode—if applicable—achieves the better quality/compression trade-off than RM, but for a practical CS codec it is useful to support RM encoding to be able to capture signals that are characterized by sudden or rare changes. In the second case study it was shown that the type and state of different appliances can be reliably identified at high compression factors of up to 30 using the proposed CS-based appliance detection strategy.

## REFERENCES

- [1] S. Gupta, M. S. Reynolds, and S. N. Patel, "ElectriSense: single-point sensing using EMI for electrical event detection and classification in the home," in *Proc. ACM Int. Conf. Ubiquitous Comput.*, 2010, pp. 139–148.
- [2] E. Candès and M. Wakin, "An introduction to compressive sampling," *IEEE Signal Process. Mag.*, vol. 25, no. 2, pp. 21–30, Mar. 2008.
- [3] D. Gangopadhyay, E. G. Allstot, A. M. R. Dixon, K. Natarajan, S. Gupta, and D. J. Allstot, "Compressed sensing analog front-end for bio-sensor applications," *IEEE J. Solid-State Circuits*, vol. 49, pp. 426–438, Feb. 2014.
- [4] M. Shoaran, M. H. Kamal, C. Pollo, P. Vanderghyest, and A. Schmid, "Compact low-power cortical recording architecture for compressive multichannel data acquisition," *IEEE Trans. Biomed. Circuits Syst.*, vol. 8, no. 6, pp. 857–870, Dec. 2014.
- [5] X. Liu, M. Zhang, T. Xiong, A. G. Richardson, T. H. Lucas *et al.*, "A fully integrated wireless compressed sensing neural signal acquisition system for chronic recording and brain machine interface," *IEEE Trans. Biomed. Circuits Syst.*, vol. 10, no. 4, pp. 874–883, Aug. 2016.
- [6] J. Zhang, S. Mitra, Y. Suo, A. Cheng, T. Xiong *et al.*, "A closed-loop compressive-sensing-based neural recording system," *J. Neural Eng.*, vol. 12, pp. 1741–2552, Jun. 2015.
- [7] F. Pareschi, P. Albertini, G. Frattini, M. Mangia, R. Rovatti, and G. Setti, "Hardware-algorithms co-design and implementation of an analog-to-information converter for biosignals based on compressed sensing," *IEEE Trans. Biomed. Circuits Syst.*, vol. 10, no. 1, pp. 149–162, Feb. 2016.
- [8] J. Zhang, Y. Suo, S. Mitra, S. . Chin, S. Hsiao *et al.*, "An efficient and compact compressed sensing microsystem for implantable neural recordings," *IEEE Trans. Biomed. Circuits Syst.*, vol. 8, no. 4, pp. 485–496, Aug. 2014.
- [9] F. Chen, A. P. Chandrakasan, and V. M. Stojanovic, "Design and analysis of a hardware-efficient compressed sensing architecture for data compression in wireless sensors," *IEEE J. Solid-State Circuits*, vol. 47, no. 3, pp. 744–756, Mar. 2012.
- [10] M. Shoaib, K. H. Lee, N. K. Jha, and N. Verma, "A 0.6–107  $\mu\text{W}$  energy-scalable processor for directly analyzing compressively-sensed EEG," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 61, no. 4, pp. 1105–1118, Apr. 2014.
- [11] M. Crescentini, D. Bellasi, M. Marchesi, D. Cristaudo, D. Rossi *et al.*, "A 2 MS/s 10 A Hall current sensor SoC with digital compressive sensing encoder in 0.16  $\mu\text{m}$  BCD," in *Proc. European Solid-State Circuits Conf.*, Sep. 2016, pp. 393–396.
- [12] T. Chang and K. C. Juang, "CMOS SC-spinning, current-feedback hall sensor for high speed and low cost applications," in *Proc. IEEE Sensors*, Nov. 2014, pp. 527–530.
- [13] J. Jiang, W. J. Kindt, and K. A. A. Makinwa, "A continuous-time ripple reduction technique for spinning-current hall sensors," *IEEE J. Solid-State Circuits*, vol. 49, pp. 1525–1534, Jul. 2014.
- [14] H. Heidari, E. Bonizzoni, U. Gatti, and F. Maloberti, "A CMOS current-mode magnetic hall sensor with integrated front-end," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 62, pp. 1270–1278, May 2015.
- [15] D. E. Bellasi and L. Benini, "Energy-efficiency analysis of analog and digital compressive sensing in wireless sensors," *IEEE Trans. Circuits and Syst. I, Reg. Papers*, vol. 62, pp. 2718–2729, Nov. 2015.
- [16] J. A. Tropp, M. B. Wakin, M. F. Duarte, D. Baron, and R. G. Baraniuk, "Random filters for compressive sampling and reconstruction," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, vol. 3, May 2006, pp. III–III.
- [17] D. Bellasi, R. Rovatti, L. Benini, and G. Setti, "A low-power architecture for punctured compressed sensing and estimation in wireless sensor-nodes," *IEEE Trans. Circuits and Syst. I, Reg. Papers*, vol. 62, pp. 1296–1305, May 2015.
- [18] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, pp. 1289–1306, Apr. 2006.
- [19] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, pp. 489–509, Feb. 2006.
- [20] E. J. Candès and Y. Plan, "A probabilistic and RIPless theory of compressed sensing," *IEEE Trans. Inf. Theory*, vol. 57, pp. 7235–7254, Nov. 2011.
- [21] Y. C. Eldar and G. Kutyniok, *Compressed sensing: theory and applications*. Cambridge University Press, 2012.
- [22] E. Candès and J. Romberg, "Sparsity and incoherence in compressive sampling," *Inverse problems*, vol. 23, no. 3, p. 969, 2007.
- [23] D. Bellasi, L. Bettini, C. Benkeser, T. Burger, Q. Huang, and C. Studer, "VLSI design of a monolithic compressive-sensing wideband analog-to-information converter," *IEEE J. Emerg. Sel. Topics. Circuits Syst.*, vol. 3, no. 4, pp. 552–565, Dec. 2013.
- [24] G. Tang, B. N. Bhaskar, P. Shah, and B. Recht, "Compressed sensing off the grid," *IEEE Trans. Inf. Theory*, vol. 59, no. 11, pp. 7465–7490, Nov. 2013.
- [25] Y. Chi, L. L. Scharf, A. Pezeshki, and A. R. Calderbank, "Sensitivity to basis mismatch in compressed sensing," *IEEE Trans. Signal Process.*, vol. 59, no. 5, pp. 2182–2195, 2011.
- [26] M. F. Duarte and R. G. Baraniuk, "Spectral compressive sensing," *Appl. Comput. Harmonic Anal.*, vol. 35, no. 1, pp. 111–129, 2013.
- [27] Z. Peng, M. R. Jackson, J. Rongong, F. Chu, and R. M. Parkin, "On the energy leakage of discrete wavelet transform," *Mech. Syst. Signal Process.*, vol. 23, no. 2, pp. 330–343, 2009.
- [28] M. Mishali and Y. C. Eldar, "Reduce and boost: Recovering arbitrary sets of jointly sparse vectors," *IEEE Trans. Signal Process.*, vol. 56, no. 10, pp. 4692–4702, 2008.
- [29] P. Stoica and P. Babu, "Sparse estimation of spectral lines: Grid selection problems and their solutions," *IEEE Trans. Signal Process.*, vol. 60, no. 2, pp. 962–967, 2012.
- [30] J. A. Hogan, *Time-Frequency and Time-Scale Methods: Adaptive*

*Decompositions, Uncertainty Principles, and Sampling.* Springer Science & Business Media, 2007.

- [31] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Trans. Inf. Theory*, vol. 53, no. 12, pp. 4655–4666, 2007.
- [32] L. Gan, "Block compressed sensing of natural images," in *Proc. Int. Conf. Digit. Signal Process.*, 2007, pp. 403–406.
- [33] S. H. Shalmany, D. Draxelmayer, and K. A. Makinwa, "A  $\pm 5$  A integrated current-sensing system with  $\pm 0.3\%$  gain error and  $16 \mu\text{A}$  offset from  $-55^\circ\text{C}$  to  $+85^\circ\text{C}$ ," *IEEE J. Solid-State Circuits*, vol. 51, pp. 800–808, Apr. 2016.
- [34] L. Dalessandro, N. Karrer, and J. W. Kolar, "High-performance planar isolated current sensor for power electronics applications," *IEEE Trans. Power Electron.*, vol. 22, pp. 1682–1692, Sept. 2007.
- [35] A. Ajbl, M. Pastre, and M. Kayal, "A fully integrated hall sensor microsystem for contactless current measurement," in *Proc. IEEE Sensors*, 2012, pp. 1–4.
- [36] H. Blanchard, C. de Raad Iseli, and R. S. Popovic, "Compensation of the temperature-dependent offset drift of a hall sensor," *Sensors and Actuators A: Physical*, vol. 60, no. 1, pp. 10–13, 1997.
- [37] A. Bilotti, G. Monreal, and R. Vig, "Monolithic magnetic hall sensor using dynamic quadrature offset cancellation," *IEEE J. Solid-State Circuits*, vol. 32, no. 6, pp. 829–836, June 1997.
- [38] M. Crescentini, M. Marchesi, A. Romani, M. Tartagni, and P. A. Traverso, "A broadband, on-chip sensor based on Hall effect for current measurements in smart power circuits," *IEEE Trans. Instrum. Meas.*, vol. 67, pp. 1470–1485, June 2018.
- [39] M. Crescentini, A. Romani, and E. Sangiorgi, "Physical simulations of response time in hall sensor devices," in *Proc. Int. Conf. Ultimate Integr. Silicon (ULIS)*, Apr. 2014, pp. 89–92.
- [40] M. Crescentini, P. Traverso, P. Alberti, A. Romani, M. Marchesi *et al.*, "Experimental characterization of bandwidth limits in hall sensors," in *Proc. IMEKO TC-4 Int. Symp. Understanding World through Elect. Electronic Meas.*, 2016, pp. 1–6.
- [41] M. Crescentini, M. Biondi, A. Romani, M. Tartagni, and E. Sangiorgi, "Optimum design rules for cmos hall sensors," *Sensors*, vol. 17, no. 4, p. 765, 2017.
- [42] M. Crescentini, M. Marchesi, A. Romani, M. Tartagni, and P. A. Traverso, "Bandwidth limits in hall effect-based current sensors," *ACTA IMEKO*, vol. 6, no. 4, pp. 17–24, 2017.
- [43] E. Sackinger and W. Guggenbuhl, "A versatile building block: the CMOS differential difference amplifier," *IEEE J. Solid-State Circuits*, vol. 22, pp. 287–294, Feb. 1987.
- [44] J. Jiang and K. A. Makinwa, "Multipath wide-bandwidth cmos magnetic sensors," *IEEE J. Solid-State Circuits*, vol. 52, pp. 198–209, Jan. 2017.
- [45] H. Heidari, E. Bonizzoni, U. Gatti, and F. Maloberti, "A CMOS current-mode magnetic Hall sensor with integrated front-end," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 62, pp. 1270–1278, May 2015.
- [46] G. Dantona, A. Ferrero, and R. Ottoboni, "Bandwidth enhancement of an insulated voltage to voltage transducer via digital compensation for industrial applications," in *Proc. IEEE Instrumentation and Measurement Technology Conference*, 1998, pp. 380–384.
- [47] L. Hars, "Frequency response compensation with dsp," *IEEE Signal Processing Magazine*, vol. 20, no. 4, pp. 91–95, 2003.
- [48] A. Piti, G. Verticale, C. Rottondi, A. Capone, and L. Lo Schiavo, "The role of smart meters in enabling real-time energy services for households: The Italian case," *Energies*, vol. 10, p. 199, Feb. 2017.
- [49] J. Haboba, M. Mangia, F. Pareschi, R. Rovatti, and G. Setti, "A pragmatic look at some compressive sensing architectures with saturation and quantization," *IEEE J. Emerg. Sel. Topics. Circuits Syst.*, vol. 2, no. 3, pp. 443–459, Sep. 2012.
- [50] G. W. Hart, "Nonintrusive appliance load monitoring," *Proceedings of the IEEE*, vol. 80, no. 12, pp. 1870–1891, 1992.
- [51] K. C. Armel, A. Gupta, G. Shrimali, and A. Albert, "Is disaggregation the holy grail of energy efficiency? the case of electricity," *Energy Policy*, vol. 52, pp. 213–234, 2013.
- [52] Y. Du, L. Du, B. Lu, R. Harley, and T. Habetler, "A review of identification and monitoring methods for electric loads in commercial and residential buildings," in *Proc. IEEE Energy Convers. Congress Exposition*, Sept. 2010, pp. 4527–4533.
- [53] J. A. Hoyo-Montaño, C. A. Pereyda-Pierre, J. M. Tarín-Fontes, and J. N. Leon-Ortega, "Overview of non-intrusive load monitoring: A way to energy wise consumption," in *Proc. Int. Conf. Power Electron.*, June 2016, pp. 221–226.
- [54] Y. Wang, A. Filippi, R. Rietman, and G. Leus, "Compressive sampling for non-intrusive appliance load monitoring (nalm) using current

waveforms," in *Proc. Int. Conf. Signal Process. Pattern Recognition Applicat.*, 2012, p. 8.

- [55] M. Clark and L. Lampe, "Single-channel compressive sampling of electrical data for non-intrusive load monitoring," in *Proc. IEEE Global Conf. Signal Inf. Process.*, Dec. 2015, pp. 790–794.



**David Bellasi** received his M.A. degree in architecture from Accademia di architettura, Mendrisio, Switzerland, in 2007 and was awarded the Premio Boni for his M.A. Thesis. In 2013, he received his M.Sc. degree (with distinction) in electrical engineering and information technology from ETH Zurich, Switzerland, where he is currently pursuing his Ph.D. degree. He was awarded with the ETH Medal in 2014 for his M.Sc. Thesis and is the winner of the Swisscom/ICTnet Innovation Award 2013.



**Marco Crescentini** received the Dr.Eng degree (cum laude) in electronics engineering and the Ph.D. degree in ICT from the University of Bologna, Bologna, Italy, in 2008 and 2012, respectively. He was an intern at Infineon Technologies, Villach, Austria, in 2009, working on DC-DC converters. He serves as Research Fellow at the University of Bologna, Cesena campus, Italy, since 2016. His research interests are in high-accuracy, low-noise and broadband measurement systems.



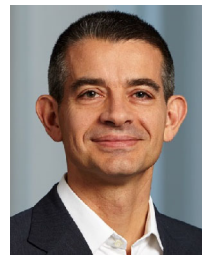
**Aldo Romani** received the Dr.Eng. degree in electronics engineering and Ph.D. degree in electronics engineering, computer science, and telecommunications from the University of Bologna, Bologna, Italy, in 2001 and 2005, respectively. In 2008, he joined the University of Bologna, Cesena, Italy, where he currently serves as an Associate Professor. He has been involved with different types of sensor interfaces and systems, applications of piezoelectric materials, and circuits for energy harvesting. He has authored or coauthored more

than 60 international scientific publications. Dr. Romani was a co-recipient of the 2004 Jan Van Vessem Award of the IEEE International Solid-State Circuits Conference.



**Marco Tartagni** is Associate Professor at the Department of Electronics, University of Bologna, Italy. He is one of the founders of Silicon Biosystems, and a member of the Center of Excellence for Electronic Design ARCES, University of Bologna. He was co-recipient of the Van Vessem Outstanding Paper Award, received at the 2004 IEEE ISSCC conference, for presenting a biosensor platform based on dielectrophoresis. He was European coordinator of FP6 Receptronics in the Nanotechnology thematic area, coordinator of the

Energy Autonomous Systems. Since 2014 he is member of the scientific board and team leader of the joint STMicroelectronics and Bologna University lab for sensor design.



**Luca Benini** is professor of Digital Circuits and Systems at ETH Zurich, Switzerland, and is also professor at University of Bologna, Italy. His research interests are in energy-efficient system design and multicore SoC design. He is also active in the area of energy-efficient smart sensors and sensor networks. He has published more than 800 papers in peer-reviewed international journals and conferences, four books and several book chapters. He is a fellow of the ACM and Member of the Accademia Europea.

He received the IEEE CAS Mac Van Valkenburg Award.