ADC Assisted Random Sampler Architecture for Efficient Sparse Signal Acquisition

Mehdi Safarpour, Reza Inanlou, Mostafa Charmi, Omid Shoaei, Olli Silvén

***Abstract*—A method for sampling Fourier sparse signals for efficient implementation of Analog to Information Convertors (AIC) is proposed. The solution reconstructs Nyquist rate high resolution signal from Nyquist rate low resolution and sub-Nyquist rate high resolution samples. For implementation an architecture based on customized reconfigurable successive approximation register (SAR) ADC is proposed, simulated, and demonstrated. The power consumption with a 90nm CMOS process is less than 26μW with 1Msample/s rate in reconfigurable 3/10-bit mode. The number of FLOPS needed for signal recovery is less than 2% required by the orthogonal matching pursuit algorithm. The functionality of the solution has been verified with an experimental system.**

***Index Terms*—analog-digital conversion; analog-digital integrated circuits.**

I. Introduction

Analog to Digital Convertors (ADC) are a bottleneck of high frequency electronic design for communication sensing systems. This is due to the high power consumption and design complexity of ADCs [1]. Even in many low sampling rate applications, e.g., battery powered Internet-of-Things (IoT) and mobile devices, it is important to minimize power consumption. On the other hand, there are applications, such as cognitive radio and radar [2-5], where the input signal is generally sparse in Fourier domain, meaning that, the occupied share of the total bandwidth is small [5].

Recently, Compressive Sensing (CS) has made it possible to leverage the sparsity structure of signals into reduced signal sampling rates [6-10]. The CS theory states that sparse signals can be acquired with much less measurements than the length of the signal, and the signal can be recovered from this incomplete set of observations [8]. Based on outcomes of the CS theory, Analog to Information Convertors (AIC) have been proposed as an alternative for conventional ADCs to efficiently obtain sparse signals with fewer number of quantized samples. In general, the AICs are composed of two sections: analog front-end and digital back-end. The analog section takes CS based non-adaptive measurements, while the digital back-end reconstructs the original signal from the measurements [9, 10].

The reported power consumptions of implemented AICs are much smaller than for equivalent ADCs. For example, the analog front-end of the AIC in [11] consumes 506.4mW, including the RF front-end, for Effective Instantaneous Band Width (EIBW) of 2GHz with 54dB dynamic range. A comparable ADC based solution is estimated to dissipate 6W [11, 12]. However, the digital back-end can still be power hungry. For example, in [13] and [14], 202mW and 24mW power dissipations are reported for recovery rates of only 11.1Msample/s and 400Ksample/s, respectively. In [13] the estimated power consumption for a 40Gsample/s recovery system is over 110W, with estimated chip area of 80mm×80mm. Clearly, the reconstruction algorithms demand lots of digital resources and power for real-time implementations [9], while energy efficiency is vital for mutually communicating battery powered devices. In some works, for instance [15], to circumvent the computationally costly complete CS recovery, the features of interest of a signal are extracted directly from compressed data.

In this contribution, a new sparse signal acquisition scheme is considered that utilizes interplay between sub-Nyquist rate high resolution and Nyquist rate low resolution sampling. We demonstrate the general approach with 3-bit Nyquist and 10-bit sub-Nyquist rate random samplings. In comparison to compressed sensing, the method is shown to provide for energy savings in the digital back-end, with minor sacrifice in the analog front-end.

II. Background

In this section, the CS theory is briefly reviewed against a typical, non-uniform sampler AIC approach.

A. Compressive Sensing

The CS theory provides a framework for acquisition and recovery of sparse signals with significantly reduced number of required samples. A length-N signal $X \in R^{N\times 1}$, is called K-sparse, if only K out of its N elements are non-zero in some basis [8]. An incomplete set of M measurements $Y_{M\times 1}$ are obtained by multiplying vector $X_{N\times 1}$ by $M \times N$ measurement matrix $\Phi_{M\times N}$

$$Y_{M\times 1} = \Phi_{M\times N} X_{N\times 1} \quad (1)$$

The number of measurements, M, can be much smaller than number of the signal elements N ($M<<N$). Since (1) is an underdetermined system, infinite number of solutions exist for X. Based on the sparsity assumption, an estimate $\hat{X}$ can be recovered by solving the l_1 norm minimization problem [8]:

Mehdi Safarpour and Olli Silvén are with the University of Oulu, Finland, (emails firstname.surname@oulu.fi). Reza Inanlou and Omid Shoaei (email: firstname.surname@ut.ac.ir) are with the University of Tehran, and Mostafa Charmi is with the University of Zanjan, Iran (email: charmi.mostafa@znu.ac.ir).

$$\hat{X} = \arg\min\left\|\hat{X}'\right\|_1 \text{ subject to } Y = \Phi\hat{X}' \quad (2)$$

If the input signal, X does not have a sparse representation in the sampling domain, while it has a sparse representation, α, in a transform domain ($\alpha = \Psi X$), the recovery problem can be modified to solve for $\hat{\alpha}$ instead, as below:

$$\hat{a} = \arg\min\|\hat{a}'\|_1 \text{ subject to } Y = \Phi\Psi^{-1}\hat{a}' \quad (3)$$

Once $\hat{a}$ is known, $\hat{X}$ can be recovered through $\hat{X} = \Psi^{-1}\hat{a}$. Incoherence of Φ and Ψ matrices is essential for exact recovery, requiring Φ to be random [8] to make each measurement a unique linear combination of elements of X.

B. Non-Uniform Sampler

Non-uniform sampler (NUS) is a typical AIC, first proposed in [8] with first implementation presented in [9]. Fig. 1 depicts a typical block diagram of the AIC. It consists of an ADC which is driven by a pseudo-random clock and a digital processor which performs CS recovery to reconstruct the original signal from the incomplete set of random samples. The NUS AIC can be assumed as a conventional Nyquist-rate ADC which randomly discards some of the uniformly taken samples. Using the NUS architecture, Fourier domain sparse signals can be sub-sampled and recovered.

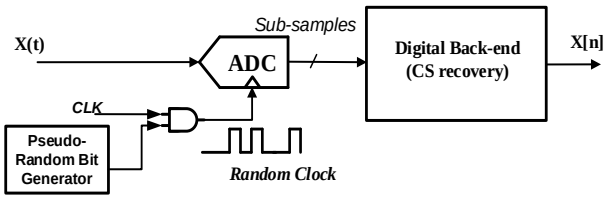

Fig. 1. Block diagram of the Non-Uniform Sampler (NUS) AIC

The major advantage of AICs is their capability of sub-Nyquist sampling. However, this is at the cost of power dissipation in the digital recovery process. To obviate this problem, we propose an approach that by sacrificing slightly the energy consumptions of the analog front-end, substantial savings in the digital back-end are achieved. In section IV, an experimental NUS AIC is used for verifying the simulation results.

III. PROPOSED ARCHITECTURE

Fig. 2 illustrates the proposed architecture, that augments the conventional AIC with a low-resolution ADC, a 3-bit version in our example. The high resolution sub-Nyquist rate path samples the Fourier domain sparse signal X through the measurement matrix Φ, and maps it to vector Y. The low-resolution Nyquist rate path detects most non-zero coefficients of the spectrum with the computationally simple Periodogram algorithm [16,17] for removal from the CS recovery process. As a result, the reconstruction problem is transformed into solving an over-determined system with a few unknowns. The non-zero coefficients and their corresponding columns are gathered into matrix A, which is $\Phi \times \Psi$. This problem is efficiently solved through l_2 norm minimization, providing the actual values of non-zero coefficients. Finally, the removed coefficients, that are known to be zero, are substituted into their places. Please note that, in section IV.C, for transistor level implementation, we merge the ADC into a NUS AIC.

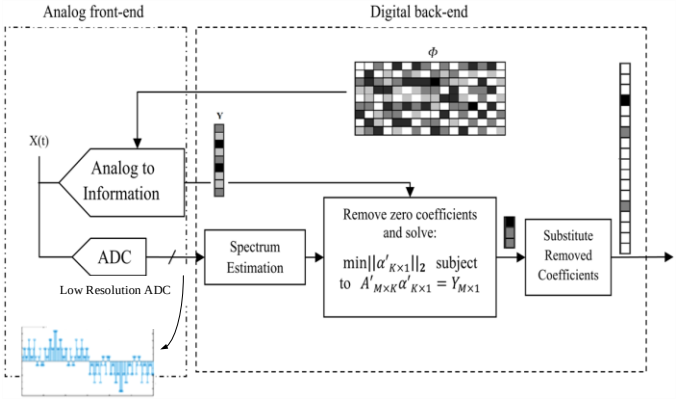

Fig. 2. Architecture of proposed acquisition system

A. Recovery Algorithm for the Proposed Architecture

In the algorithm below X_{ADC} refers to the signal acquired by the low-resolution ADC, and A to $\Phi \times \Psi$ matrix. The proposed algorithm recovers the signal in four steps. First, the locations of the possible non-zero coefficients in the spectrum are detected. Then, the detected zero coefficients are removed from the equation of the recovery system, with the recovery problem becoming an over-determined one. Finally, the solution is obtained through l_2 norm minimization [18].

Algorithm 1. The proposed recovery procedure

Step 1. Find K largest($S<K<<M$) FFT coefficients of the X_{ADC} (S is sparsity degree)

Step 2. Except for columns corresponding to the locations of K largest coefficients, remove all columns of the $A_{M\times N}$ matrix and assume the resultant matrix as $A'_{M\times K}$

Step 3. Find the a' through solving the following via the $l_2$norm minimization

$$\min||\alpha'_{K\times 1}||_2 \text{ subject to } A'_{M\times K}\alpha'_{K\times 1} = Y_{M\times 1}$$

Step 4. Substitute the removed zero coefficients, expanding $\alpha'_{K\times 1}$ to full-length recovered signal, $\hat{\alpha}_{N\times 1}$

IV. RESULTS AND DISCUSSION

The proposed architecture was simulated in MATLAB with the above recovery algorithm. Then, transistor level simulation of SAR ADC/NUS was carried out. Finally, a demonstration system was set-up using off-the-shelf components.

A. Computational Complexity of Recovery Algorithms

The computational complexity of the proposed recovery method was compared against two CS algorithms, Basis Pursuit (BP) [18] and Orthogonal Matching Pursuit (OMP) [7]. The complexity of the FFT based Periodogram is $O(N\log N)$ [17], for sorting[2] it is $O(KN)$ [19, 20], and for Step 4 $O(MK^2 + K^3)$, where ($K << M << N$). The overall computational complexity is:

$$\text{CC} = O(N\log N) + O(MK^2 + K^3) + O(N) = O(N\log N) \quad (4)$$

Table 1 outlines the complexity of the different algorithms.

[2] The most complex sorting algorithm is of order $O(N^2)$. Here, sorting only K largest, would result in computational cost of $O(KN)$.

Table 1.Computational cost of recovery algorithms

Algorithms	BP [6]	OMP [7]	Proposed
Complexity	$O(M^2N^{1.5})$	$O(sMN)$	$O(N\log N)$

*Notes. The notation s and K refer to the sparsity and number of non-zero coefficients; M denotes the number of measurements; N denotes the signal length.

For complexity comparison of algorithm implementations, their MATLAB execution FLOP counts were used [21]. Fig. 3 shows the FLOP counts by BP, OMP and the proposed method. The data for recovering is a sparse signal of length N=1024 for different number of measurements M. We chose the number of potential non-zero coefficients K, to be equal to 10% of the number of measurements M, $(K = 0.1M)$. For the proposed method Fig. 3 demonstrates a significantly lower FLOP counts than for BP and OMP.

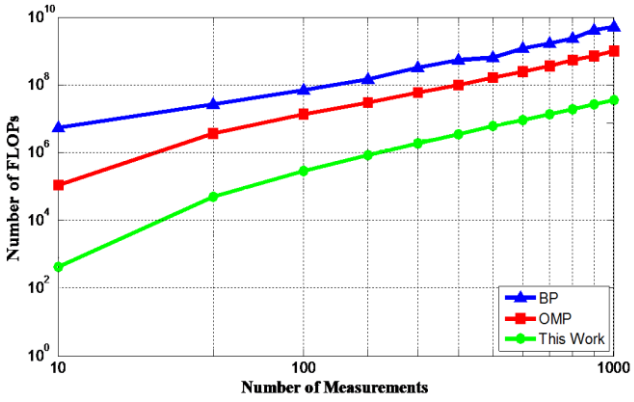

Fig. 3. Measurement of computational complexity

In MATLAB simulation, as well as HSPICE hardware simulation, and the final practical verification, a sum of three sinusoids with frequencies 100, 250 and 450 KHz was fed to the ADC as a sparse signal.

To evaluate the recovery algorithms for communication sensing, uniformly distributed wideband noise was injected into the input. Fig. 4 shows the SNRs of reconstructed results as a function of input SNRs. Notice that the bit rate of the proposed approach is twice the rate fed to BP and OMP algorithms.

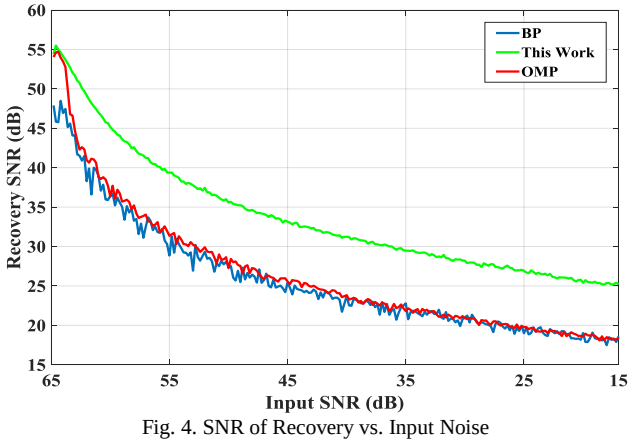

Fig. 4. SNR of Recovery vs. Input Noise

Furthermore, for transmission purposes in sensor network applications, smaller average rates of bit per sample can be captured, through reducing number of high resolution samples, while the SNR of recovery, according to simulations, is maintained approximately at the same level. Fig. 5 depicts simulation results for recovery of a sparse signal in different bit per sample rates through proposed method and CS method.

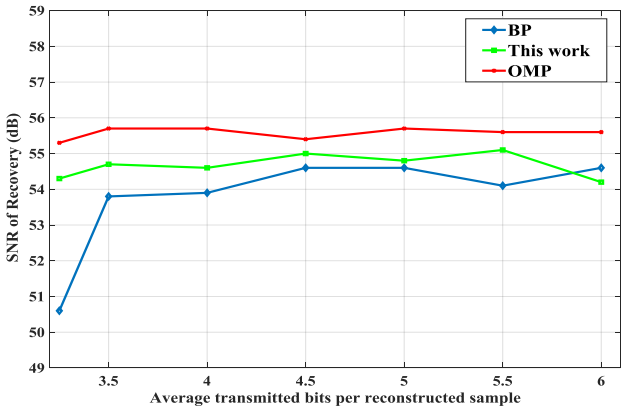

Fig. 5. Recovery comparison for equivalent bit rates

We have demonstrated the performance for 3- and 10-bit sampling precisions, selected as examples of low and high precision ADCs. The choice of *M/N* ratio depends on the characteristics of the application signals. With higher *M/N* ratios both the CS and our scheme reach higher SNRs, but the cost is increased power consumption.

B. Hardware design and simulation

Since both the ADC and NUS AIC observe the same signal and only differ by resolutions, they were merged into a reconfigurable ADC. For the ADC successive approximation register (SAR) approach was selected due to its simplicity [22]. Binary-weighted capacitive [23] array DAC with attenuation capacitor is employed to convert the digital numbers of the SAR logic to analog signal, with minor modification for dual mode operation. The details and considerations of the unit capacitor in the DAC capacitor array as well as the SAR logic are similar to [24]. Fig. 6 presents the block diagram of the fully differentially dual mode SAR ADC which consists of a comparator, Digital Control Logic (DCL) and capacitive Digital to Analog Converter (DAC). The sampling of the amplified input signal is done by DAC itself. The NUS mode sampling clock (SMP_NUS) is generated from Nyquist mode sampling clock (SMP_Nyq.). The introduced comparator or in [25] is utilized for dual mode operation.

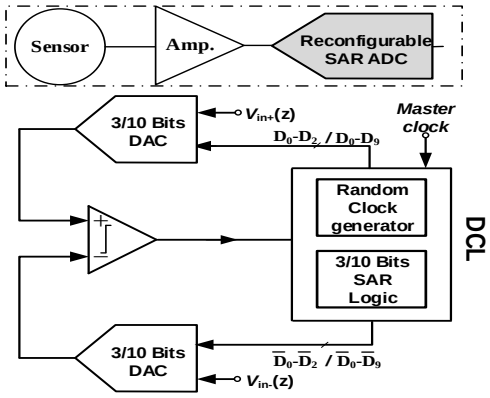

Fig. 6. Dual mode SAR ADC structure

The ADC operates in two modes based on the Random Clock Generator (RCG) [26] output. Here, 25% $(M = N/4)$ of the input samples are converted in 10-bit sub-Nyquist rate NUS

mode and the rest 75% in the 3-bit Nyquist rate mode. That is, high resolution non-uniform samples are collected on the Nyquist sampling grid skipping randomly $(N-M)$ time instants, and the rest of the samples are collected in the 3-bit mode, to reduce the average sampling-quantization rate. The10-bit and 3-bit samples are complementary and cover the whole Nyquist sampling grid. For generating the random clock, a Linear Feedback Shift Register (LFSR) consisting of *R* flip-flops [27] is used (*R* must at least be 10 to ensure that the repeated pattern of sampling clock over the sampling window of 1024 sample is random [27]). To set the ratio between the NUS and the Nyquist rate sampling, LFSR has been designed to generate random bit stream with adjustable, unequal distribution of ones and zeros (Fig.7).

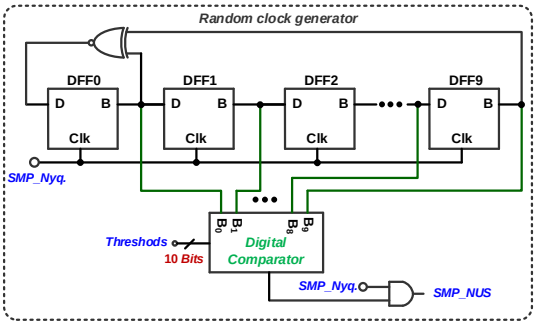

Fig.7. Random clock generator

The output of the LFSR is an R-bit integer, that is compared to a threshold, determining the distribution of 1s and 0s. The hardware for the proposed scheme was designed using a 90nm CMOS process. The simulations with HSPICE were carried out with 1MS/s sampling rate in Nyquist rate mode. The SNR of the recovered signal was above 55dB, essentially the same as in behavioral simulations. The HSPICE simulation results showed average power consumption of less than 26µW with 1V power supply. This is somewhat higher than the 18µW dissipated by the 10-bit NUS alone. However, due to the noise and linearity requirements relaxation in the 3-bit operation mode, this difference can be reduced, too. Supply voltage scaling [28] can also be used to further reduce the energy-per-conversion.

C. *Power Analysis of Analog Front-end*

We analyzed the power of the proposed architecture based on power estimations for NUS. Briefly, the power consumption of the ADC, can be expressed as [13, 29]:

$$P_{ADC,sys} = 2(B.W_f)\left[\underbrace{FOM.2^{ENOB}}_{ADC} + \underbrace{3C_1.G_A^2.2^{2ENOB}}_{Amplifier}\right] \quad (5)$$

Since, a NUS is basically an ADC with non-uniform sampling behavior, its power consumption equation is almost the same as with ADC. The difference is multiplication of the first term by factor of M/N, as the average NUS sample rate is M/N (M random samples in each N sample window). So, for the NUS AIC we have:

$$P_{NUS,sys} = 2(B.W_f)\left[\underbrace{FOM.2^{ENOB}.\frac{M}{N}}_{ADC} + \underbrace{3C_1.G_A^2.2^{2ENOB}}_{Amplifier}\right] \quad (6)$$

The power of the analog front-end of the proposed structure, is the sum of power consumption for an AIC and a low resolution (3-Bits) ADC. Hence, based on the above, we get:

$$P_{proposed} = P_{AIC,sys} + P_{ADC,sys} = P_{AIC,sys} + (2(B.W_f)[FOM.2^{ENOB} + 3C_1.G_A^2.2^{2ENOB}]) \quad (7)$$

Depending on the AIC type, the power estimate of the respective AIC is substituted in the above equation. In case the AIC is a NUS, the result becomes:

$$P_{Front-end,sys} = P_{AIC,sys} + P_{ADC,sys} = 2(B.W_f)\left[\frac{M}{N}.FOM.2^{ENOB_{NUS}} + FOM.2^{ENOB_{3Bits}} + 3C_1.G_A^2.2^{2ENOB_{NUS}}\right] \quad (8)$$

Since the implementation here is in fact a dual mode reconfigurable ADC (ADC and NUS share samples), the power is further saved and is:

$$P_{proposed,sys} = 2(B.W_f)\left[\frac{M}{N}.FOM.2^{ENOB_{10Bits}} + \left(\frac{N-M}{N}\right)FOM.2^{ENOB_{3Bits}} + 3C_1.G_A^2.2^{2ENOB_{10Bits}}\right] \quad (9)$$

The power consumption compared to a NUS is higher, but still lower than the power consumption of an equivalent ADC.

D. *Experimental Results*

An implementation for a verification of the proposed architecture was carried out based on STM ARM Cortex®-M4 microcontroller. This chip was selected due to its three independent programmable 12-Bit, 1MS/s SAR ADCs and a Random Number Generator (RNG) Unit [30]. Firmware was written to emulate a low-resolution Nyquist-rate ADC and a NUS AIC. Fig. 8 shows the setup.

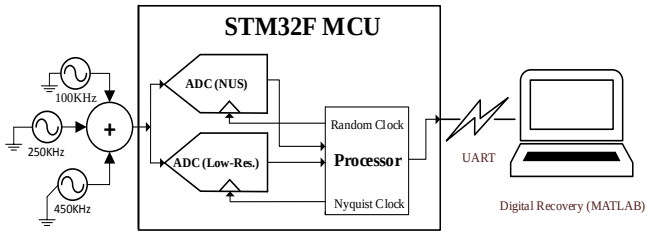

Fig.8. Test bench setup

A sparse signal same as the previous section was generated through summing outputs of three signal generators. An Op-Amp used as an analog adder and buffer. The same lengths of 1MS/s rate of the signal was collected as random non-uniform 10-Bits and uniform Nyquist rate 3-Bits samples and recovered. The experimental implementation, subjected to the non-idealities of the real world, resulted in the reconstructed signal with slightly above 50dB SNR.

E. *Discussion*

Table 2 compares results of HSPICE simulations of the proposed architecture to other AIC solutions. Since computational cost of our method grows linearly, we can capture longer windows, therefore, the solution is more robust against spectral leakage. However, the downside of the method is that the minimum amplitude of each frequency bin should be high enough to be detectable through the spectrum sensing algorithms. Admittedly, the power consumption of our AIC solution is not the lowest one in Table 2, but the primary intention has been in showing the efficient reconstruction with the proposed scheme. That is achieved at a moderate overhead in the analog part. Moreover, the proposed method can be integrated to other AICs such as the Random Demodulator AIC architecture proposed by Guo et al [33], to push their efficiency

further. The applications of the proposed methods are in two scenarios. First, it can be used for wide-band sparse spectrum sensing and recovery. In this case, there is no communication cost, and getting the low-resolution version of data only adds slightly to power consumption of analog front-end. The second application, is in sensor networks where commonly the signal needs to be compressed to be transmitted from a battery powered sensor node to another, and the received signal needs to be reconstructed there. We have shown that the proposed approach is capable of recovery with the same average bit per sample rate as CS methods, with tolerable loss in reconstruction quality.

Table 2. Summary and comparison with different approaches

	Architecture	AIC					This Work
		[9] Chip	[11] Chip	[31] Chip	[32] Sim.	[33] Chip	Sim.
Analog	Power	5.8W	506mW	1.8μW	69.5nW	5μW	26μW
	ENOB	9 Bits	9 Bits	6.5 Bits	9.5 Bits	9.8 Bits	9.1 Bits
	Bandwidth	2GHz	2GHz	1KHz	500Hz	500KHz	500KHz
	Technology	Custom Design[2]	90nm	0.13μm	65nm	0.13μm	90nm
	FoM [/step]	2.7pJ	0.24pJ	9.6pJ	187fJ	5fJ	46fJ
Digital [1]	Algorithm	BP		OMP			Proposed
	Number of FLOPs	~100M		~10M			~0.1M
FoM		$\frac{Power}{2 \times B.W \times 2^{ENOB}}$					

[1] *Number of FLOPs for recovery of windows 1024 samples and 120 measurements*

[2] *in-house S/H and off-the-shelf ADC*

V. SUMMARY

The computational cost of the proposed architecture for sparse signal acquisition is small in comparison to compressed sensing recovery algorithms. It can be an alternative for applications with energy dissipation of computing as a crucial factor. Nevertheless, the power consumption in the analog front-end increases slightly due to adding a low-resolution ADC. The proposed reconfigurable implementation is flexible, combining a random sampler SAR ADC/NUS with an adjustable output distribution random clock generator. The functionality of the approach has been demonstrated with HSPICE together with a practical implementation.

REFERENCES

[1] H. Chung and G.-Y. Wei, "ADC-based backplane receiver design-space exploration," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 22, pp. 1539-1547, 2014.

[2] Baraniuk, Richard, and Philippe Steeghs. "Compressive radar imaging." In Radar Conference, 2007 IEEE, pp. 128-133. IEEE, 2007.

[3] A. Gruget, M. Roger, C. Lelandais-Perrault, P. Bénabès, and P. Loumeau, "Wide-band multipath A to D converter for cognitive radio applications," in *RF Front-ends for Software Defined and Cognitive Radio Solutions (IMWS), 2010 IEEE International Microwave Workshop Series on*, 2010, pp. 1-4.

[4] R. Pribic and I. Kyriakides, "Design of sparse-signal processing in radar systems," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, 2014, pp. 5008-5011.

[5] B. Razavi, "Cognitive radio design challenges and techniques," *IEEE Journal of Solid-State Circuits*, vol. 45, pp. 1542-1553, 2010.

[6] D. Needell and J. A. Tropp, "Cosamp: iterative signal recovery from incomplete and inaccurate samples," *Communications of the ACM*, vol. 53, pp. 93-100, 2010.

[7] J. A. Tropp and A. C. Gilbert, "Signal recovery from random measurements via orthogonal matching pursuit," *IEEE Transactions on information theory*, vol. 53, pp. 4655-4666, 2007.

[8] E. J. Candès and M. B. Wakin, "An introduction to compressive sampling," *IEEE signal processing magazine*, vol. 25, pp. 21-30, 2008.

[9] M. Wakin, *et al.*, "A nonuniform sampler for wideband spectrally-sparse environments," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 2, pp. 516-529, 2012.

[10] T. Murray, P. Pouliquen, A. G. Andreou, and K. Lauritzen, "Design of a CMOS A2I data converter: Theory, architecture and implementation," in *Information Sciences and Systems (CISS), 2011 45th Annual Conference on*, 2011, pp. 1-6.

[11] J. Yoo, S. Becker, M. Loh, M. Monge, E. Candes, and A. Emami-Neyestanak, "A 100MHz–2GHz 12.5 x sub-Nyquist rate receiver in 90nm CMOS," in *2012 IEEE Radio Frequency Integrated Circuits Symposium*, 2012, pp. 31-34.

[12] J. Yoo, "Compressed sensing receivers: theory, design, and performance limits," California Institute of Technology, 2012.

[13] O. Salehi-Abari, "Building compressed sensing systems: sensors and analog-to-information converters," Massachusetts Institute of Technology, 2012.

[14] P. Maechler, et al., "VLSI design of approximate message passing for signal restoration and compressive sensing," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 2, pp. 579-590, 2012.

[15] P. V. Rajesh et al., "22.4 A 172µW compressive sampling photoplethysmographic readout with embedded direct heart-rate and variability extraction from compressively sampled data," 2016 IEEE International Solid-State Circuits Conference (ISSCC), San Francisco, CA, 2016, pp. 386-387.

[16] Parhi, Keshab K., and Manohar Ayinala. "Low-complexity welch power spectral density computation." IEEE Transactions on Circuits and Systems I: Regular Papers 61, no. 1 (2014): 172-182.

[17] J. G. Proakis and D. K. Manolakis, *Digital Signal Processing (4th Edition)*: Prentice-Hall, Inc., 2006.

[18] S. Boyd and L. Vandenberghe, *Convex optimization*: Cambridge university press, 2004.

[19] C. H. Papadimitriou, *Computational complexity*: John Wiley and Sons Ltd., 2003.

[20] S. Abdel-Hafeez and A. Gordon-Ross, "An Efficient O (N) Comparison-Free Sorting Algorithm," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 2017.

[21] T. Minka. The lightspeed matlab toolbox, [Online] Available: https://github.com/tminka/lightspeed.

[22] B. Razavi, "A Tale of Two ADCs: Pipelined Versus SAR," *IEEE Solid-State Circuits Magazine*, vol. 7, pp. 38-46, 2015.

[23] R. J. Baker, *CMOS: circuit design, layout, and simulation* vol. 1: John Wiley & Sons, 2008.

[24] R. Inanlou, M. Shahghasemi, and M. Yavari, "A noise-shaping SAR ADC for energy limited applications in 90 nm CMOS technology," *Analog Integrated Circuits and Signal Processing*, vol. 77, pp. 257-269, 2013.

[25] V. Chaturvedi, T. Anand, and B. Amrutur, "An 8-to-1 bit 1-MS/s SAR ADC with VGA and integrated data compression for neural recording," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 21, pp. 2034-2044, 2013.

[26] H. Seo, J. Choi, H. Kim, T. Park, and H. Kim, "Pseudo random number generator and hash function for embedded microprocessors," in *Internet of Things (WF-IoT), 2014 IEEE World Forum on*, 2014, pp. 37-40.

[27] M. Stipčević and Ç. K. Koç, "True random number generators," in *Open Problems in Mathematics and Computational Science*, ed: Springer, 2014, pp. 275-315.

[28] Bohr, Mark T., and Ian A. Young. "CMOS Scaling Trends and Beyond." IEEE Micro 37, no. 6 (2017): 20-29.

[29] R. H. Walden, "Analog-to-digital converter survey and analysis," *IEEE Journal on selected areas in communications*, vol. 17, pp. 539-550, 1999.

[30] S. Microelectronics, "RM0090 Reference Manual," in *DocID018909 Rev* vol. 8, ed, 2014.

[31] 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," in IEEE Journal of Solid-State Circuits, vol. 49, no. 2, pp. 426-438, Feb. 2014.

[32] S.Wazir, A. Shukla, S. Deb, and A. Majumdar. "Energy efficient EEG acquisition and reconstruction for a Wireless Body Area Network." INTEGRATION, the VLSI journal vol. 58, pp.295-302, 2017.

[33] W. Guo, Y. Kim, A. H. Tewfik and N. Sun, "A Fully Passive Compressive Sensing SAR ADC for Low-Power Wireless Sensors," in IEEE Journal of Solid-State Circuits, vol. 52, no. 8, pp. 2154-2167, Aug. 2017.