Additive companding implementation to reduce ADC constraints for multiple signals digitization

Mathieu Vallérian, Florin Hutu, Benoît Miscopein, Guillaume Villemaud, Tanguy Risset

Additive Companding Implementation to Reduce ADC Constraints for Multiple signals Digitization

Mathieu Vallérian*†, Florin Hutu†, Benoît Miscopein*, Guillaume Villemaud† and Tanguy Risset†
*Orange-Labs, 38243, Meylan, France,
†Université de Lyon, INRIA, INSA-Lyon, CITI-INRIA, F-69621, Villeurbanne, France

***Abstract*—In urban sensor networks, the diversity of propagation conditions can lead to the simultaneous reception of signals having very different power levels. Given the diversity of wireless technologies used in this area, implementing gateways using a Software-Defined Radio (SDR) approach seems to be a very practical solution. Overcoming the large dynamic range may however require a very high resolution Analog-to-Digital Converter (ADC) to digitize the weakest signal with a satisfying precision. One possibility to relax this requirement is to use a companding technique before digitization. This paper describes how to use an additive companding approach to reduce ADC's complexity and proposes two implementations of the compressing law.**

I. Introduction

Urban sensor networks are generally composed of transmit-only sensors and sink nodes, also called gateways, that gather the data and send them to a datacenter. In many actual deployments, sensors are Short Range Devices (SRD) operating in the 868 MHz band, with an 8 MHz bandwidth [1]. These networks cover a broad set of applications, such as water and gas metering, parking management, etc. The diversity of these applications means that signals are emitted with various constraints and various propagation conditions. Thus the signals received on the gateways have different features, mainly in terms of amplitude and spectral occupancy.

A challenge in the design of gateways comes from the co-existence of different communication protocols that must be embedded. As a consequence, the gateway must be flexible enough to be reconfigured and updated in order to adapt to new communication protocols. A very relevant approach do this is to use a Software-Defined Radio (SDR) architecture [2]. To take full advantage of SDR technologies, a promising approach is to digitize the whole band at once in order that the signal processing can be performed as much as possible in the digital domain. In this approach, the gateway's Analog-to-Digital Converter (ADC) must be able to digitize signals having potentially different characteristics. Therefore, this ADC must have a fine enough resolution to simultaneously demodulate several signals potentially with a high dynamic range.

A previous study has shown [3] that, considering urban sensor network propagation conditions and technologies, signals can be received with a 100 dB power ratio, requiring to be digitized with a 21 bits resolution to be able to demodulate each of them. If the ADC resolution is lower, the weakest signal will be flooded in the quantization noise. As a 21 bits resolution cannot be reasonably achieved with an 8 MHz bandwidth with today's ADCs [4], hence another method to digitize the signals has to be found. One way to overcome this problem is the use of *companding* techniques.

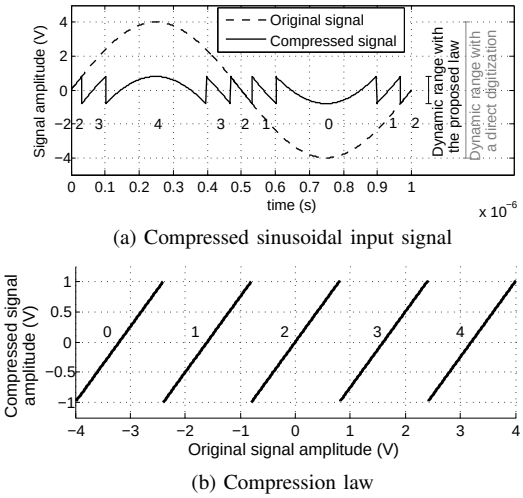
Companding is a signal processing technique aiming at reducing the dynamic range of the input signal before digitization [5]. This technique has been widely used in analog telephony and in transmission of speech messages in digital wireless systems like cellular telephony. Companding is clearly described in [6]. When applied to digitization, the analog signal is first compressed to reduce its dynamic range, the compressed signal is then digitized and eventually expanded to recover the original signal. The expansion is performed by means of the expanding function (or law), which is the compression reciprocal function. Different laws have been studied, for instance the μ-law [5] and Piecewise-linear (PL) law [7] companding approaches have been tested in [8] to reduce the dynamic range of signals. It appears that, whereas companding techniques are widely used for reducing the dynamic ranges of two signals occurring at different times, they appear to be inefficient to cope with the combination of two simultaneous signals. This paper studies another companding technique, proposed in [9]. The technique is based on adding offsets to the input signal, to make it able to cope with the peculiar situation of two incoming signals with unbalanced signal strengths.

II. Compression law using offsets

A. Motivation

The general idea of the additive companding law is to maximize the dynamic range occupied by the signal after compression (*i.e.* before digitization). A way to get this improvement is to add offsets on the input signal to center it around the zero level and to set the compressing gain in order to maximize the signal occupancy of the ADC's dynamic range. To do this, input signal voltage thresholds are identified, defining states, each state corresponding to a specific offset value added to the signal and to a specific gain. The principle of additive companding is shown on Fig. 1a with a sinusoidal input signal, the law used being illustrated in Fig. 1b. In the following, the additive companding law is referred to as a Piecewise-Linear, Constant-Gain compressing law with Offsets (PLCGO).

A more common approach is the PL law [7] which uses functioning states too, but in which the compressing gain is modified instead of adding offsets in each state. The Fig. 2 outlines the benefit of the PLCGO law (Fig. 2c) with regard to the PL law (Fig 2b) with a sinusoidal input (Fig. 2a): the quantization noise is dependent on the input signal amplitude with the PL law, leading to a performance loss because of

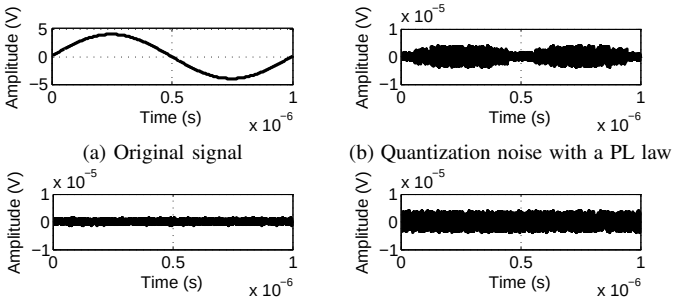


(a) Compressed sinusoidal input signal

(b) Compression law

Fig. 1. The PLCGO companding technique

the presence of the strong signal. This effect has disappeared with the PLCGO law. By comparison with the quantization noise with direct digitization (Fig. 2d), one can see that the quantization noise is reduced with the PLCGO law.



(a) Original signal

(b) Quantization noise with a PL law

(c) Quantization noise with a PLCGO law

(d) Quantization noise with a direct digitization

Fig. 2. Quantization noise with a PL law, PLCGO law and without companding

The PLCGO law has already been implemented in [9] for a different application (*i.e.* cochlear implants). However, because of a very different signal frequency, their technique cannot be transposed to our application.

B. Analytical impact of PLCGO companding

Assuming the input uncompressed signal varies between $-V_M$ and V_M, the i^{th} threshold will take the value:

$$V_{th,i} = -V_M + \frac{2 \cdot V_M}{N_{th}+1} \cdot i \tag{1}$$

with N_{th} being the total number of thresholds. In a given state, the gain g that should be applied to the signal depends on the ADC's dynamic range V_{ADC} and on the number of states $N_{th}+1$. In PLCGO, the thresholds are equidistributed and the gains do not depend on i. Hence, the gain can be expressed as the ratio of the ADC full-scale range and the difference of two voltage thresholds:

$$g = \frac{V_{ADC}}{V_{th,i+1} - V_{th,i}} = (N_{th}+1) \cdot \frac{V_{ADC}}{2V_M} \tag{2}$$

The impact of this law can be evaluated through the quantization noise. On Fig. 1a, it can be seen that after compressing, the total signal's dynamic range is reduced by a factor equal to the number of states. The quantization step Δ is reduced by the same factor (assuming the signal is scaled to the ADC input), since it is the ratio between the dynamic range and the number of quantization states. If Δ_{PLCGO} denotes the quantization step with the PLCGO compressing and Δ_{DD} denotes the quantization step with a direct digitization (*i.e.* without companding), we have:

$$\Delta_{PLCGO} = \frac{\Delta_{DD}}{N_{th}+1} \tag{3}$$

$N_{Q,PLCGO}$, $N_{Q,DD}$ are respectively the quantization noise with the PLCGO law and with a direct digitization. The quantization noise being $\Delta/\sqrt{12}$ [10], the benefit of the law in terms of quantization noise can be expressed from (3) as:

$$\frac{N_{Q,PLCGO}}{N_{Q,DD}} = \frac{\Delta_{PLCGO}}{\Delta_{DD}} = \frac{1}{N_{th}+1} \tag{4}$$

The quantization step is the ADC full scale voltage divided by the number of quantization states $2^n = 1/\Delta$ of the ADC (n being the ADC resolution). Here, n_{PLCGO} and n_{DD} are respectively the equivalent resolution when using the PLCGO compressing law and the direct digitization. From (4) we obtain:

$$n_{PLCGO} = n_{DD} + \log_2(N_{th}+1) \tag{5}$$

The great interest of this law is highlighted in (5) since PLCGO companding allows to save a number of bits equal to binary logarithm of the number of states. We remind here that 21 bits of resolution are required to digitize the set of received signals in the 8 MHz band. Implementing PLCGO companding with 32 states would require an ADC resolution of 16 bits. Such a resolution is still unaffordable today, but recent work [11] predicts such ADCs in a near future.

III. Architecture proposal and implementation

A. Performance evaluation

The PLCGO law has been first cosimulated to assess the relevance of the proposed approach. Using the Agilent ADS software, a strong (of power S_s) and a weak (of power S_w) Binary Phase Shift Keying (BPSK)-modulated signals are generated together with an Additive White Gaussian Noise (AWGN). The resulting signal is compressed, digitized and expanded. The compression and expansion blocks are implemented using the Matlab software and digitization is performed with a sampling frequency of 16 MHz. After demodulation, a Bit Error Rate (BER) test is performed through the Monte-Carlo method to evaluate the companding impact. The noise level is defined with respect to the weak signal energy per bit on noise density ratio (E_b/N_0). The E_b/N_0 of the weak signal has been arbitrarily set to 7 dB in all the simulations.

The power ratio between the strong and the weak signal (S_s/S_w) is not of prime importance since the ADC resolution improvement only depends on the number of states. Indeed the required resolution depends on S_s/S_w, but it can be lessened by the PLCGO law by a number of bits independent of S_s/S_w. The S_s/S_w value is then arbitrary set to 50 dB. Simulations are done by varying the ADC resolution and the number of states.

The BER test is performed only on the weak signal since it would take too much simulation time if done on the strongest one. Assumption is made that there is no delay between the time the signal cross a threshold and the time the appropriate offset is added to the signal.

For a given number of states, the minimum resolution is obtained when the BER value is close to the expected value of 10^{-3} (which is the theoretical BER when using a BPSK modulation with $E_b/N_0 = 7$ dB). It can be seen on Fig. 3 that increasing the number of states allows to reach the expected BER floor with a smaller resolution. The use of 40 states reduces the required resolution by 5 bits, showing the interest of the method.

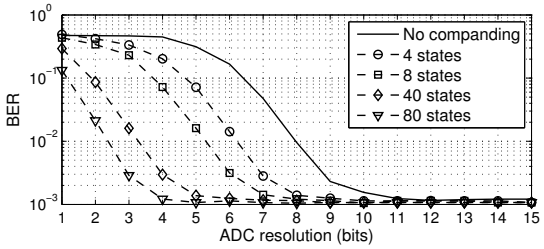


Fig. 3. BER of the weak signal vs the ADC resolution for $S_s/S_w = 50$ dB, depending on the number of thresholds used with the PLCGO law

B. Impact of the delay of updating of the state variable

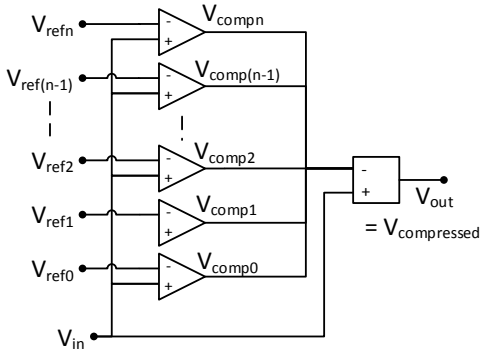
With the PLCGO law as well as with the PL law, the state variable must be updated when the signal reaches a threshold level to apply respectively the proper offset and gain. The threshold crossing can be detected both in analog or digital domains.

It has been previously assumed that the state variable update was instantaneous, but this assumption does not stand if the threshold crossing detection is performed after digitization. In that case, the delay would be at least one sampling period T_s. The existence of this delay of updating is a problem, as during this delay a wrong offset is added to the signal. This can lead to an ADC clipping and thus to the loss of the signal's information during this time interval. This problem cannot be addressed by using a scaling factor between compression and digitization steps, since the number of thresholds that the signal can cross during the delay of updating cannot be predicted on a real signal.

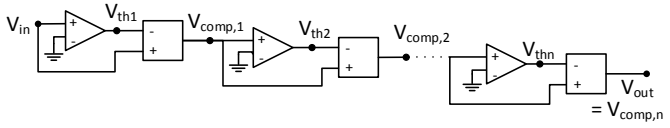
The impact on a single signal's BER of the state variable delay of updating has been evaluated by performing the compression with a delay set to T_S. If ideal companding (*i.e.* without delay) would allow to digitize properly the signal, the BER is heavily degraded (it is equal to 0.5) when a delay equal to T_S is added, regardless of the ADC resolution: the delay is so big that the expanding law is almost never applied using the proper state variable. Thus the law must be implemented analogically, to guarantee the delay of updating to be negligible compared to T_S.

C. Implementation proposal

One can use comparators to generate the offsets. A parallelized and a cascaded implementations are proposed, whose diagrams are shown on Fig. 4. In the parallelized implementation (Fig. 4a), the comparators compare the signal with pre-defined thresholds that can be generated through bandgap voltage references. Their outputs are summed, giving a proportional value to the offset that must be added to the signal. This value is scaled to generate the real offset and is subtracted from the signal. The resulting signal is then scaled to the ADC input. On Fig. 4a, a differential amplifier performs all these operations. Another digitization (not shown on Fig. 4a) is still required to get the offset in a digital form. To do this, the comparators outputs are summed, giving the state variable and digitized with a resolution equal to the binary logarithm of the number of thresholds. As we found through (5) that 32 thresholds should be enough to properly digitize the signal with a 16 bits resolution (with $S_s/S_w = 100$ dB), the offsets can be digitized with a 5 bits ADC. So this solution could allow in theory (*i.e.* without considering the implementation imperfections) to replace one ADC with a 21 bits resolution with two ADCs with resolutions of 16 and 5 bits.



(a) Parallelized implementation



(b) Cascaded implementation

Fig. 4. Proposed implementations of the PLCGO law

With the cascaded implementation (Fig. 4b), the offset value is found iteratively. The first comparator gives the most significant bit of the offset value and this value (V_{thi}) is scaled to the input signal and subtracted from it through a differential amplifier. The resulting signal is the input signal compressed with one threshold. Then the second comparator gives the next bit of the offset value, and the corresponding value is subtracted from the signal. This operation is repeated until the total offset has been subtracted from the signal.

Given that the application of this digitization method targets sensors network gateways the main constraint to design a suitable implementation of the proposed companding technique is the demodulation performance. The power consumption is not a key point, as it is assumed that the gateway is powered by the mains supply.

These architectures are compared through simulations with a S_s/S_w ratio of 50 dB. E_b/N_0 is still 7 dB in all the simulations and only the BER considering the weak signal is simulated. With both architectures, 8 states are used in order to compress the signal, allowing in theory (from (5)) to save

3 bits. Simulations have been made using ADC resolutions set to 25, 8 and 5 bits. It should be noted that a minimum resolution of 11 bits is required to digitize the signal without companding (see Fig. 3), meaning that any resolution higher than 11 bits will provide the same BER performance. A 25-bit resolution therefore represents the expected satisfying level of performance that solutions implementing companding should be compared with.

Results reported in Tab. I are given with a relative variance of 0.01. They show that both architectures provide good results: the expected gain of 3 bits provided by the PLCGO law is confirmed. Indeed both architectures with 8-bit resolution provide the same BER value as the one provided by the minimum required 11-bit resolution (represented by the 25-bit resolution row). Along the same line, the BER levels obtained with companding with a 5-bit resolution are comparable (even if slightly above) to the BER value of a 8-bit resolution without companding.

TABLE I. SIMULATED BER DEPENDING ON THE PLCGO COMPRESSING ARCHITECTURE

ADC resolution	Simulated BER		
	parallelized architecture	cascaded architecture	without companding
5	$1.7 \cdot 10^{-2}$	$1.4 \cdot 10^{-2}$	$3.7 \cdot 10^{-1}$
8	$1.2 \cdot 10^{-3}$	$1.3 \cdot 10^{-3}$	$8.6 \cdot 10^{-3}$
25 ($11 \rightarrow \infty$)	$1.3 \cdot 10^{-3}$	$1.1 \cdot 10^{-3}$	$1.2 \cdot 10^{-3}$

D. Discussion on the offset generation proposal

From the BER performance perspective, one cannot prefer the cascaded or the parallel implementation to generate offsets. But both of them has some pros and cons: on one hand the parallelized architecture ensures that an error on one offset does not propagate to the other offsets, contrary to the cascaded architecture. On the other hand, a problem exists on the state transition: when the signal crosses a threshold, the comparators output does not switch from the low to the high state immediately, as the comparator amplifier's gain is not high enough. During this transition time, the offset takes a voltage value ranging between two predefined values. If this happened at an ADC sampling time (assuming the clocks of the offset's ADC and of the compressed signal's ADC are aligned), the signal would be compressed and expanded with a different offset value. The probability of this to happen is smaller with the cascaded architecture, as the amplifiers' gains cumulate from one stage to the other.

The selection of a given architecture can also be led from an implementation complexity perspective. The parallelized architecture uses more amplifier stages: a k states compression requires k amplifier stages if they are in parallel, whereas $2 \cdot \log_2(k)$ amplifier stages are enough if cascaded. To digitize the state variable, the parallelized implementation is more convenient as it sums the comparators outputs. So it can be digitized using a summing amplifier. In the cascaded implementation case, each comparator output is one bit of the final offset value. The all comparators outputs should be digitized separately, with a one-bit ADC.

IV. CONCLUSION AND PROSPECTS

The PLCGO law has been studied for reducing the required resolution when digitizing simultaneous signals with a high dynamic range. It allows to reduce the quantization noise in the particular case of the decoding of two simultaneous signals having a high difference of amplitude present on different frequencies. This quantization noise reduction allows us to reduce the ADC complexity for a given BER obtained on the weak signal.

Theoretical performance results are given and two implementations based on cascaded or parallelized comparators are proposed. Both architectures have been simulated using Agilent ADS software and confirm their expected theoretical performances. The main advantage of the cascaded architecture is its low complexity, whereas the parallelized architecture allows more precision when offsets are integrated since an error on one comparator does not propagate to the other ones. However, the cascaded architecture cumulative gain is an advantage, as the comparators saturation is faster, leading to a smaller probability of error when digitizing the state variable.

Both architectures have limits, since the offset must be generated with the same precision than the target equivalent resolution, and its AC component must be less than 1 LSB. Future work will focus on a physical implementation of these architectures and tests on real systems in order to evaluate this method's feasibility in practice.

REFERENCES

[1] ETSI, "Electromagnetic compatibility and Radio spectrum Matters (ERM); System Reference document (SRdoc): Spectrum Requirements for Short Range Device, Metropolitan Mesh Machine Networks (M3N) and Smart Metering (SM) applications," Sept. 2011, TR 103 055.

[2] M. Dardaillon, K. Marquet, J. Martin, T. Risset, and H.-P. Charles, "Cognitive radio programming: Existing solutions and open issues," Inria, Tech. Rep. 8358, Sep. 2013. [Online]. Available: http://hal.inria.fr/hal-00859467/

[3] M. Vallerian, G. Villemaud, B. Miscopein, T. Risset, and F. Hutu, "SDR for SRD: ADC specifications for reconfigurable gateways in urban sensor networks," in *Radio and Wireless Symposium (RWS), 2014 IEEE*, Jan 2014, pp. 178–180.

[4] B. Jonsson, "A survey of A/D-Converter performance evolution," in *Electronics, Circuits, and Systems (ICECS), 2010 17th IEEE International Conference on*, 2010, pp. 766–769.

[5] D. Van Ess, *Algorithm - Logarithmic Signal Companding - Not just a good idea - it is μ-Law*, Cypress, december 2013, AN2095.

[6] Y. Tsividis, "Externally linear, time-invariant systems and their application to companding signal processors," *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on*, vol. 44, no. 2, pp. 65–85, 1997.

[7] V. Maheshwari, W. Serdijn, and J. Long, "Companding Baseband Switched Capacitor Filters and ADCs for WLAN Applications," in *Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on*, 2007, pp. 749–752.

[8] Vallerian, M. and Villemaud, G. and Hutu, F. and Miscopein, B. and Risset, T., "Rapport de recherche inria," INRIA, Tech. Rep., 2015.

[9] C. Bes, C. Sawigun, and W. Serdijn, "An additive instantaneously companding readout system for cochlear implants," in *Biomedical Circuits and Systems Conference (BioCAS), 2010 IEEE*, Nov 2010, pp. 126–129.

[10] S. Gupta and A. Phatak, *ADC Guide, Part 1 – The Ideal ADC*, Cypress Semiconductor, Janvier 2012.

[11] C. Hurrell, C. Lyden, D. Laing, D. Hummerston, and M. Vickery, "An 18 b 12.5 MS/s ADC With 93 dB SNR," *Solid-State Circuits, IEEE Journal of*, vol. 45, no. 12, pp. 2647–2654, 2010.