

RESEARCH ARTICLE



## A simple encoding method for Sigma-Delta ADC based biopotential acquisition systems

Federico N. Guerrero  and Enrique M. Spinelli 

Instituto de Investigaciones en Electrónica, Control, y Procesamiento de Señales LEICI (CONICET-UNLP), La Plata, Argentina

### ABSTRACT

Sigma Delta analogue-to-digital converters allow acquiring the full dynamic range of biomedical signals at the electrodes, resulting in less complex hardware and increased measurement robustness. However, the increased data size per sample (typically 24 bits) demands the transmission of extremely large volumes of data across the isolation barrier, thus increasing power consumption on the patient side. This problem is accentuated when a large number of channels is used as in current 128–256 electrodes biopotential acquisition systems, that usually opt for an optic fibre link to the computer. An analogous problem occurs for simpler low-power acquisition platforms that transmit data through a wireless link to a computing platform. In this paper, a low-complexity encoding method is presented to decrease sample data size without losses, while preserving the full DC-coupled signal. The method achieved a 2.3 average compression ratio evaluated over an ECG and EMG signal bank acquired with equipment based on Sigma-Delta converters. It demands a very low processing load: a C language implementation is presented that resulted in an 110 clock cycles average execution on an 8-bit microcontroller.

### ARTICLE HISTORY

Received 2 June 2017  
Revised 3 August 2017  
Accepted 8 August 2017  
Published online 26 September 2017

### KEYWORDS

Sigma Delta converter; encoding; lossless compression; biomedical signal; biopotential; EMG; ECG; dynamic range

## 1. Introduction

High-resolution Sigma-Delta analogue-to-digital converters ( $\Sigma$ - $\Delta$  ADCs) have enabled a paradigm shift in biopotential measurements due to the high dynamic range (DR) they achieve.

Electrophysiological phenomena produce potentials that can be generally represented with a DR from 10 to 16 bits. However, the electrodes needed to measure these biopotentials introduce a DC offset several orders of magnitude higher than the biomedical signals. Artefacts can also produce high amplitude baseline fluctuations. The traditional approach is to separate the high magnitude components in the frequency domain by means of high-pass filtering, also known as AC coupling. Alternatively, the use of  $\Sigma$ - $\Delta$  ADCs allows effectuating DC-coupled measurements by matching or exceeding the DR at the output of the electrodes.

There are several advantages to this approach. First, there is a significant hardware reduction because neither a high gain amplifier nor a filter with large time constants is needed. The first uses of  $\Sigma$ - $\Delta$  ADCs in biopotential acquisition demonstrated this benefit [1,2], and in the past decade the advances of complex

integrated mixed signal systems have drastically reduced cost and energy consumption per acquisition channel. Another advantage is that measurement robustness is increased because signal integrity is preserved even during large amplitude, fast artefacts. Furthermore, restoration of the baseline is trivial and does not require special fast-recovery circuitry. Finally, the possibility of a DC-coupled system allows to better preserve low-frequency components which are critical for some biomedical signals such as electrooculogram (EOG), slow cortical potentials (SCP) or full-band EEG (FbEEG) measurements [3].

Currently, application specific standard products (ASSPs) carrying high resolution  $\Sigma$ - $\Delta$  ADCs such as the ADS129x family from Texas Instruments (Dallas, TX), LTC244x from Linear Technology or proprietary  $\Sigma$ - $\Delta$  technology, replace traditional methods in a wide variety of biopotential acquisition systems ranging from high performance, high channel count equipment (such as g.Hlamp from g.tec or [4]) to simple wearable devices (e.g. [5,6]).

However, producing high DR signals has an inherent drawback: the increased data volume. Particularly in biopotential acquisition equipment, the digitised

signal must traverse an isolation barrier with an associated energy consumption cost. For example, a very robust acquisition set-up is a battery powered front-end with optical fibre isolation as presented in [7] and currently used in high-end commercial equipment such as Biosemi ActiveTwo, and almost mandatory in high EMI environments, such as magnetic resonance compatible equipment like Geodesic Net Amp 400 MR. Although the channel data transference capacity of optic fibre is very high, the increased data size reduces the isolated front end's battery life because the duty cycle increases when more bytes are needed. Another type of biopotential acquisition platforms are low-complexity battery-powered front-ends that transmit the acquired data via a wireless link to a more complex computing platform such as a smartphone or a PC (e.g. [8]). A higher data volume in these devices also increases power consumption because more transmission bursts are needed or a higher data rate must be selected. Finally, although modern permanent storage flash memories have a very high capacity, devices such as loop ECG recorders depend on limited volatile memory and an increase in sample size reduces the recording length.

$\Sigma$ - $\Delta$  based systems, thus, can have simpler hardware and improve measurement robustness, but at the expense of an increased data volume. It would be desirable to lower the sample size using a method that preserved the full DR of the DC-coupled signal while having a negligible impact on power consumption and system complexity, and maintaining real-time operation capabilities (i.e. operate on a sample-by-sample basis instead of block-wise). Moreover, as  $\Sigma$ - $\Delta$  systems can be multi-modal, the method should not be targeted to a specific biomedical signal but to the general case of biopotential acquisition.

A large number of methods for biomedical signal compression have been developed in the literature. Studies concerning lossless compression achieve compression ratios (CR, the quotient of the original signal size to its compressed size) between 2 and 3.5 for EEG and ECG as summarised in [9], and between 1.45 and 4 for EMG [10,11]. Reported algorithms suited for multi-modal systems [9,12] are targeted at powerful processors or VLSI, while simpler encoding methods leverage specific signal characteristics [13,14]. A reported method that achieves low cycles per sample in a microcontroller implementation uses lossy compression [15], similarly to alternatives that rely on compressive sensing [16].

In this work, we present a simple, real-time coding and decoding method that allows reducing the size of multi-modal biopotential signal samples acquired with

$\Sigma$ - $\Delta$  converters without loss, preserving the full DR of the signal but avoiding the cost of handling the excess bits. The method requires a minimal computation overhead that can be implemented in the simplest low-end microcontrollers. Although the method can be applied to samples acquired in any way with a  $\Sigma$ - $\Delta$  ADC, it achieves a better performance when the signal is properly conditioned to take advantage of the full available DR. Hence, next we discuss the DR requirements of biopotential signals and then proceed to present the encoding method.

## 2. Methods

### 2.1. Maximum dynamic range needed for biopotential signal acquisition

Current  $\Sigma$ - $\Delta$  converters for biopotential applications generally deliver samples in 24 bit (3 bytes) packets. However, only a portion of those bits contains useful information. A few of the least significant bits are masked by the internal noise of the ADC, while the most significant bits may exceed the necessary range. A trade-off is established between the obtainable resolution, measured in so-called "noise free bits" (usually between 19 and 22 bits) and the acquisition rate.

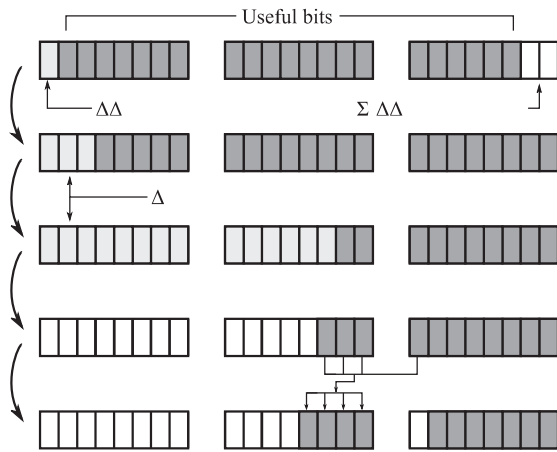
EEG signals require a very low input noise. A thorough evaluation of the input noise of EEG equipment is conducted in [17], where the authors conclude that a  $10 \text{ nV}/\sqrt{\text{Hz}}$  noise spectral amplitude is appropriate for the smallest signals and only in the case of skin abrasion or puncturing could a lower noise be necessary. If this spectral amplitude is considered in a 100 Hz bandwidth, an equivalent input noise voltage of  $100 \text{ nV}_{\text{rms}}$  is obtained. The amplitude of the least significant bit of an ADC in an acquisition system should be three times the RMS noise value [18], hence the lower bound for the DR calculation is 300 nV.

A pessimistic bound for the larger DC amplitude in a differential channel can be found in the AAMI standard for clinical ECG equipment [19], where it is required to accept a  $\pm 300 \text{ mV}$  electrode offset.

As a result, the measurement DR results in  $\log_2(600 \text{ mV}/300 \text{ nV}) = 20.9$  bits. Hence, a 21 bit DR is adequate, even under pessimistic considerations, for the most demanding biopotential measurement applications.

### 2.2. Proposed method

The proposed encoding method is based on a sequence of very simple operations, starting with the well-known delta coding [20], combined with



**Figure 1.** Schematic representation of the proposed method. Numbered transitions represent: (1) Sample conditioning; (2) Delta coding; (3) ZigZag coding; (4) VarInt coding.

a scheme used in data servers to store integers of variable size [21]. The method steps are depicted in Figure 1; the stages are:

1. Sample conditioning.
2. Delta coding.
3. ZigZag coding.
4. VarInt coding.

In Delta coding, two consecutive samples are subtracted and the difference is stored or transmitted. A first full sample must always be sent or stored as a starting point, and then the full signal can be exactly reconstructed by the addition of the difference values. Delta coding is useful when successive samples present small changes, which can occur when the higher amplitude components have low frequencies compared with the acquisition rate. The higher amplitude spurious components of signals as measured from the electrodes are, in general, slower than the smaller valued biopotentials. The biomedical signals themselves have low-frequency components that can benefit from Delta encoding, and therefore it is a usual first step in compression algorithms.

Delta coding can reduce a 3-byte sample to 1 or 2 bytes, but establishing a fixed reduced sample length would detriment the robustness of high DR acquisition. Ideally, the full DR must be available if necessary. In the case that a reduced byte count is not sufficient to represent a sample, the number of bytes should be allowed to increase. For this purpose, it is proposed to use variable length integer coding as used within the method to serialise data structures called *Protocol Buffers* from Google [21].

Variable length integers (called *VarInts*) are based on a very simple scheme: each value is divided into

$N$ -bit packets with an  $N-1$  bits payload and a 1-bit flag that indicates whether the represented value is composed of at least one additional packet. Thus, one integer value can occupy as little as one packet, but if necessary can grow to an unlimited number of packets to represent a value as high as needed, so no representation saturation can occur.

When VarInt values are received, an array of packets will be observed. One must look at the most significant bit of the current packet. If it is a zero, then the current value is composed of only this packet and no more are needed. If it is a one, the following packet must be also used and in turn, its MSB must be observed, and so on.

For simple implementation in microcontrollers, it is proposed to use VarInts with 1-byte packets, where 7 bits are used for a portion of the sample and 1 bit is the VarInt flag. Hence, because the maximum DR of biopotential signals is 21 bits as presented in section 2.1, a complete sample should not need more than 3 bytes. If the acquisition is not optimised and the 24 bits delivered by the  $\Sigma-\Delta$  ADC are used, the VarInt could take 4 bytes.

This is why step 1 from the proposed approach is optional. Its function is to discard the three excess bits that the 24-bit sample from the  $\Sigma-\Delta$  converter delivers. Those three discarded bits will be composed of  $n \leq 3$  least significant bits that represent the ADC noise, plus the  $3-n$  most significant bits that are excessive and the signal will not reach. Figure 1 presents an example with two noise bits and one excess bit. In that case, the sample conditioning would consist on two logic right shifts for each sample. The consequence of not implementing this step is simply that some values may be unnecessarily codified in 4 bytes.

Before VarInt encoding, an intermediate step must be taken because signed integers are usually represented using 2's complement. In 2's complement, small negative changes do not occupy a small number of bits but instead require the whole integer (because of the logical inversion). This is solved through the so-called ZigZag coding. In ZigZag coding, positive numbers are moved to even positions and negative numbers to odd positions. Hence, small variations are translated into small binary codes notwithstanding their sign. ZigZag coding can be implemented with a single bit check to determine if the sample is positive; if it is positive, a multiplication by two is required which can be implemented with a single left shift, if it is negative, the absolute value of the number must be recovered and then transferred to an odd position by a single left shift followed by a subtraction of 1.

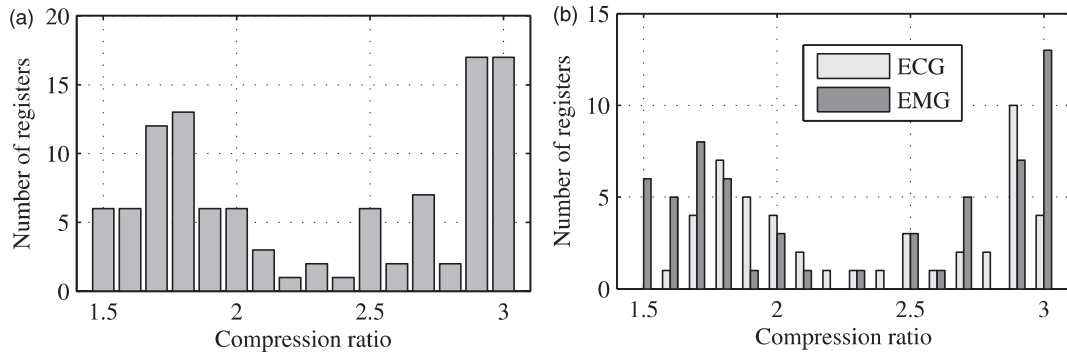


Figure 2. Compression ratios achieved for a database of ECG and EMG registers.

The last step is the VarInt coding itself. Its implementation is highly dependent on the low-level characteristics of the microcontroller or computing platform where it is programmed. For reference, a sample C language implementation of the complete method is shown in [Appendix A](#).

### 2.3. Evaluation

The method performance was evaluated by applying it to a set of EMG and ECG signals acquired with  $\Sigma-\Delta$  ADCs based equipment. The systems are described in [22,23] and are based around the ADS1299 front-end from Texas Instruments (Dallas, TX).

The signals correspond to measurements carried out by the authors with various research purposes using both wet and dry contact electrodes, with sampling rates ranging from 250 to 4000 samples per second and gains between 1 and 8. All records were truncated to a 20,000 sample length. No conditioning of the signals was effectuated in order to obtain an evaluation that represented a generic use of the algorithm for different biomedical signals under diverse acquisition conditions. Forty-eight ECG registers and 59 EMG registers were processed. A large part of the EMG recordings had been conducted with double differential electrodes, a technique that allows to increase spatial selectivity in sEMG measurements.

The compression ratio was selected as a performance metric, calculated as:

$$CR = \frac{N_o}{N_c} \tag{1}$$

where  $N_o$  is the original register length in bytes and  $N_c$  is the coded register length.

### 3. Results and discussion

The results of the complete signal bank analysis are presented in the histogram of [Figure 2\(a\)](#). The average

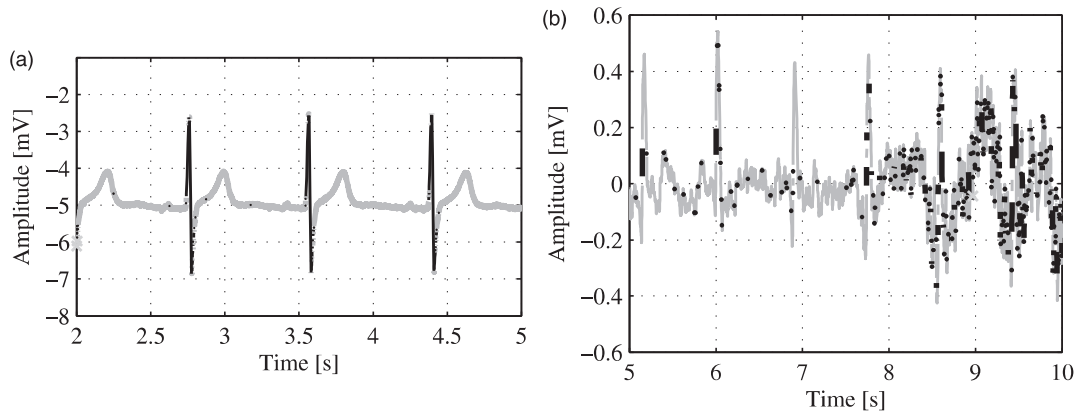
result was a CR of 2.3 with a standard deviation of 0.5. It must be noted that a CR comparison with similar works in the literature is not direct because the uncompressed signals have an excessive DR.

The averaged result is not representative of the behaviour of the method. The presented histogram shows that the achieved CR is coarsely divided into two regions, one between 1.5 and 2 and another between 2.5 and 3. The reason is that the compression is strongly dependent on the acquisition conditions. The same behaviour is observed when results are decomposed by type of signal as shown in [Figure 2\(b\)](#).

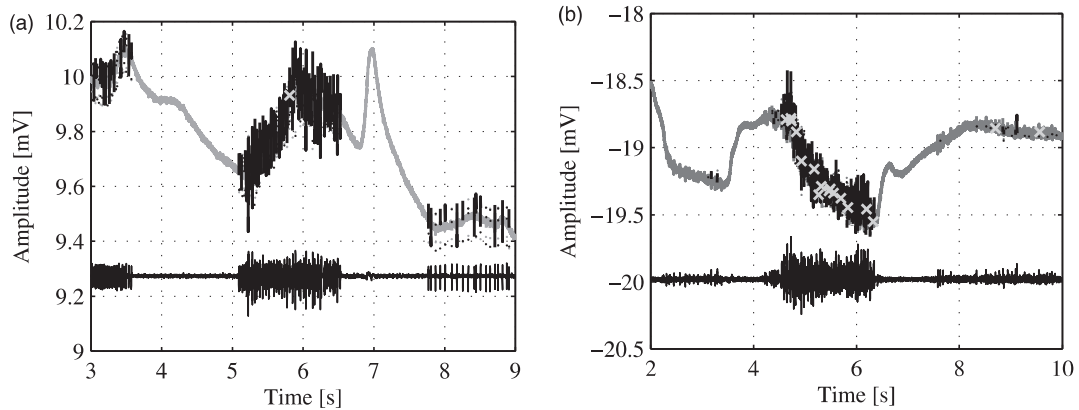
An analysis of individual signals within the register bank provides insight into the achieved CR. Sample ECG and EMG signals are presented in [Figures 3 and 4](#), respectively. In these figures, sample signals are shown with portions highlighted in different colours according to the resulting number of bytes that each sample occupied after compression. Samples that were encoded in 1 byte are coloured in a grey tone while those encoded in 2 bytes are coloured black. Samples that required 3 bytes are marked with light colour crosses.

The figures reveal that a significant portion of the baseline signal noise floor (i.e. when there is no biopotential activity) can be represented by the first 7 bits, hence samples corresponding to the baseline noise will be encoded in a single byte. Thanks to Delta encoding, slow variations do not require an increased byte count per sample. These “slow” variations encompass artefacts and smooth portions of the biomedical signals.

This can be directly observed for ECG registers. In [Figure 3\(a\)](#), the ECG waveform is encoded with 1-byte samples for a large portion of time and it is only necessary to increase byte count per sample during the short RQ or RS segments. This results in a high compression factor. In contrast, when the signal presents interference, either electromyographic or electromagnetic, the number of bytes required for



**Figure 3.** ECG sample signals. The original waveforms before compression are shown. Light grey traces correspond to segments that were encoded using 1 byte while black traces correspond to segments encoded using 2 bytes.



**Figure 4.** EMG sample signals. Upper traces are the raw waveforms while lower traces correspond to the same signal band-pass filtered in the usual EMG bandwidth of 10–450 Hz. The colour scheme is similar to Figure 3, with crosses representing samples coded with 3 bytes.

representation is increased. In particular, Figure 3(b) shows how an increase of electromyographic artefacts makes a larger number of 2-byte encoded samples necessary.

A higher number of bytes per sample are mainly demanded by fast components of the biopotential signals. This happens in cases of relatively rapid and large-amplitude variations such as ECG signal peaks or in general with EMG signals as seen in Figure 4(b). On the other hand, artefacts that produce baseline variations are in general slower phenomena and are therefore effectively compressed by delta encoding, as seen in both Figures 4(a,b).

Figure 4(b) is an example that showcases the ability of the method to robustly preserve signal integrity. The  $\Sigma$ - $\Delta$  acquisition system allows following the signal even in the presence of strong movement artefacts. The signal is encoded with 1 byte per sample for extended time periods. When an EMG event arises,

2 bytes are used to represent each sample. If only delta coding with saturation at 2 bytes was used, the integrity of the signal would be lost at some points marked in the figure with white crosses, where 3 bytes were necessary. The proposed method, however, successfully maintains the signal representation by momentarily extending the bytes per sample.

The method was implemented with simple operations (requiring no multiplication or more than a 1-bit logic shift) and therefore constituted a low processing load suitable for very low-cost microcontrollers or scaling to large channel count systems. The C language implementation from Appendix A was compiled with XC8 v1.31 compiler from Microchip for a PIC18f2550 target, without optimisation, and resulted in a 110 clock cycle average execution (minimum 77 cycles for positive deltas resulting in 1 byte compressed size and maximum 140 cycles for negative deltas resulting in 3 bytes size).



## 4. Conclusions

In this paper, a low-complexity encoding method to reduce the sample size of biomedical signals acquired with  $\Sigma$ - $\Delta$  ADCs while preserving their full DR was presented. The method can be used to lower the energy cost of transmitting data through an isolation barrier or wireless link when the full DC-coupled DR of the acquired signal must be preserved. Samples can be encoded and decoded in real time.

An average compression ratio of 2.3 was achieved over several ECG and EMG signals. When interference is low and fast biopotential activity is sporadic, the method achieves its best performance with a CR between 2.5 and 3, unaffected by movement artefacts. A minimum 1.5 CR results when signals present a high noise or interference level. However, signal integrity is fully preserved under all circumstances. The algorithm was successfully implemented on a simple 8-bit microcontroller.

## Disclosure statement


The authors report no conflicts of interest.

## Funding

This work was funded in part by projects CONICET PIP-112-2015-0100558, ANPCyT PICT-2015/2257, and UNLP I-219.

## ORCID

Federico N. Guerrero  <http://orcid.org/0000-0002-7323-7872>

Enrique M. Spinelli  <http://orcid.org/0000-0002-4164-9151>

## References

- [1] Curtin M. Sigma-delta techniques reduce hardware count and power consumption in biomedical analog front end. *Analog Dialog*. 1994;28:6–7.
- [2] McKee J, Evans N, Wallace D. Sigma-delta analogue-to-digital converters for ECG signal acquisition. *Proceedings of 18th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, vol. 1; 1996. p. 19–20.
- [3] Vanhatalo S, Voipio J, Kaila K. Full-band EEG (FbEEG): an emerging standard in electroencephalography. *Clin Neurophysiol*. 2005;116:1–8.
- [4] Usakli AB, Gencer NG. USB-based 256-channel electroencephalographic data acquisition system for electrical source imaging of the human brain. *Instrument Sci Technol*. 2007;35:255–273.
- [5] Weyer S, Menden T, Leicht L, et al. Development of a wearable multi-frequency impedance cardiography device. *J Med Eng Technol*. 2015;39:131–137.
- [6] Antonio J, Gnecci G, Téllez C. Following the path towards intelligently connected devices for on-line, real-time cardiac arrhythmia detection and classification. *IEEE International Autumn Meeting on Power, Electronics and Computing*; 2016. p. 0–4.
- [7] MettingVanRijn AC, Kuiper AP, Linnenbank AC, et al. Patient isolation in multichannel bioelectric recordings by digital transmission through a single optical fiber. *IEEE Trans Biomed Eng*. 1993;40:302–308.
- [8] Neuroelectrics. *Neuroelectrics User Manual V2.0*; 2017. Available from: [http://www.neuroelectrics.com/download/NE\\_UM\\_Part01\\_Enobio.pdf](http://www.neuroelectrics.com/download/NE_UM_Part01_Enobio.pdf)
- [9] Chua E, Fang WC. Mixed bio-signal lossless data compressor for portable brain-heart monitoring systems. *IEEE Trans Consumer Electron*. 2011;57:267–273.
- [10] Chanasabaeng P, Charoen B, Paphangkorakit J. Lossless compression of electromyographic signal. *5th 2012 Biomedical Engineering International Conference, BMEiCON*; 2012.
- [11] Cho GY, Lee GY, Lee TR. Efficient real-time lossless EMG data transmission to monitor pre-term delivery in a medical information system. *Appl Sci*. 2017;7:366.
- [12] Brinkmann BH, Bower MR, Stengel KA, et al. Large-scale electrophysiology: acquisition, compression, encryption, and storage of big data. *J Neurosci Methods*. 2009;180:185–192.
- [13] Mukhopadhyay SK, Mitra S, Mitra M. A lossless ECG data compression technique using ASCII character encoding. *Comput Electr Eng*. 2011;37:486–497.
- [14] Mukhopadhyay SK, Mitra S, Mitra M. A combined application of lossless and lossy compression in ECG processing and transmission via GSM-based SMS. *J Med Eng Technol*. 2015;39:105–122.
- [15] Chang CT, Nien CM, Rieger R. Microcontroller implementation of low-power compression for wearable biosignal transmitter. *2016 International Symposium on VLSI Design, Automation and Test, VLSI-DAT*; 2016. p. 2–5.
- [16] Chidean MI, Barquero-Perez O, Zhang Q, et al. High diagnostic quality ECG compression and CS signal reconstruction in body sensor networks. *2016 IEEE International Conference on Acoustics, Speech, and Signal Processing*; Mar. IEEE; 2016. p. 6255–6259.
- [17] Scheer HJ, Sander T, Trahms L. The influence of amplifier, interface and biological noise on signal quality in high-resolution EEG recordings. *Physiol Meas*. 2006;27:109–117.
- [18] Linnenbank A, Metting Van Rijn AC, Grimbergen C, et al. Choosing the resolution in AD conversion of biomedical signals. *Proceedings of the XXII<sup>nd</sup> International Congress on Electrocardiology*; Nijmegen, The Netherlands; 1995. p. 198–199.
- [19] AAMI. *Diagnostic electrocardiographic devices*. ANSI/AAMI EC11:1991/(R)2001/(R)2007; 2007.
- [20] Chang GC, Lin YD. An efficient lossless ECG compression method using Delta coding and optimal selective Huffman coding. *6th World Congress of Biomechanics (WCB 2010)*; 2010 Aug 1–6; Singapore. In *Conjunction with 14th International Conference on Biomedical Engineering (ICBME) and 5th Asia Pacific Conference on Biomechanics (APBiomech)*. Berlin, Heidelberg: Springer; 2010. p. 1327–1330.

- [21] Google. Protocol buffer. Developers guide: encoding; 2014. Available from: <https://developers.google.com/protocol-buffers/docs/encoding#varints>
- [22] Guerrero FN, Spinelli EM, Haberman MA. Analysis and simple circuit design of double differential EMG active electrode. *IEEE Trans Biomed Circuits Syst.* 2016;10:787–795.
- [23] Guerrero FN, Spinelli E. High gain driven right leg circuit for dry electrode systems. *Med Eng Phys.* 2017;39:117–122.

## Appendix A. C language implementation

C language implementation of the proposed method using Microchip XC8 compiler data types. The presented code extract includes only the declaration of intervening variables and the segment that implements the compression algorithm. Pointers are assumed to have been set to respective arrays of incoming and outgoing samples.

```
typedef union {
    unsigned short long integer;
    struct{
        unsigned char B0;
        unsigned char B1;
        unsigned char B2;
    } bytes;
} sample;
sample m_aux,*m_cod,m_pre,*m_orig;
char*m_cod_next;

/***** Delta coding *****/
m_orig++;
m_aux.integer=m_orig->integer - m_pre.integer;
m_pre.integer=m_orig->integer;

/***** Zig-Zag coding *****/
if(m_aux.bytes.B2 & 0x80)
    m_aux.integer = ((~m_aux.integer + 1) << 1) - 1;
else
    m_aux.integer <<= 1;

/***** VarInt coding *****/
m_cod = (sample*)m_cod_next;
m_cod->bytes.B0 = m_aux.bytes.B0 & 0x7F;
m_aux.integer <<= 1;
m_cod_next++;
if(m_aux.bytes.B2 || m_aux.bytes.B1){
    m_cod->bytes.B0 |= 0x80;
    m_cod->bytes.B1 = m_aux.bytes.B1 & 0x7F;
    m_aux.integer <<= 1;
    m_cod_next++;
    if(m_aux.bytes.B2)
    {
        m_cod->bytes.B1 |= 0x80;
        m_cod->bytes.B2 = m_aux.bytes.B2 & 0x7F;
        m_cod_next++;
    }
}
```