This is a repository copy of *A 65nm CMOS lossless bio-signal compression circuit with 250 femtoJoule performance per bit.*.

White Rose Research Online URL for this paper:
https://eprints.whiterose.ac.uk/150164/

Version: Accepted Version

Article:
Crispin-Bailey, Christopher orcid.org/0000-0003-0613-9698, Dai, Chengliang and Austin, James orcid.org/0000-0001-5762-8614 (2019) A 65nm CMOS lossless bio-signal compression circuit with 250 femtoJoule performance per bit. IEEE Transactions on Biomedical Circuits and Systems. pp. 1087-1100. ISSN 1932-4545

https://doi.org/10.1109/TBCAS.2019.2938672

Reuse
Items deposited in White Rose Research Online are protected by copyright, with all rights reserved unless indicated otherwise. They may be downloaded and/or printed for private study, or other acts as permitted by national copyright laws. The publisher or other rights holders may allow further reproduction and re-use of the full text version. This is indicated by the licence information on the White Rose Research Online record for the item.

Takedown
If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.

eprints@whiterose.ac.uk
https://eprints.whiterose.ac.uk/

A 65nm CMOS lossless bio-signal compression circuit with 250 femtoJoule performance per bit

Chris Bailey,Chenglaing Dai,Jim Austin

***Abstract*—A 65nm CMOS integrated circuit implementation of a bio-physiological signal compression device is presented, reporting exceptionally low power, and extremely low silicon area cost, relative to state-of-the-art. A novel 'xor-log2-sub-band' data compression scheme is evaluated, achieving modest compression, but with very low resource cost. With the intent to design the 'simplest useful compression algorithm', the outcome is demonstrated to be very favourable where power must be saved by trading off compression effort against data storage capacity, or data transmission power, even where more complex algorithms can deliver higher compression ratios. A VLSI design and fabricated Integrated Circuit implementation are presented, and estimated performance gains and efficiency measures for various bio-medical use-cases are given. Power costs as low as 1.2 pJ per sample-bit are suggested for a 10kSa/s data-rate, whilst utilizing a power-gating scenario, and dropping to 250fJ/bit at continuous conversion data-rates of 5MSa/sec. This is achieved with a diminutive circuit area of 155 um^2. Both power and area appear to be state-of-the-art in terms of compression versus resource cost, and this yields benefit for system optimization.**

***Index Terms*—Lossless Data Compression, VLSI Design, EEG, ECG, Wearable Sensors, Power Efficiency**

I. Introduction

DATA acquisition systems, in the domain of continuous time-varying signals, are increasingly operated under highly constrained resource limitations. This is especially so in the field of wearable devices, remote and self-powered sensors, Internet of Things (IOT), body-sensor networks, and in bio-medical applications. In such systems, power constraints may well demand reductions in data transfer and temporary local storage capacity, not the least because this can be a critical factor in extending battery-limited operating times. [1]–[3].

It is important to make a distinction between temporary local storage (in a non-volatile recording device memory, for instance), as compared to offline storage (for example, after transfer of data to some processing center or base station). In the case of offline storage, one may require data compression for reasons of practicality of data volume, and power is not necessarily a primary concern. But, where transmission requires a wired or wireless link, and/or data requires temporary storage in a device such as flash-memory, every bit of data has an associated power cost, and every bit that can be eliminated via compression will potentially save power, given appropriate conditions.

Therefore, where a power reduction can be gained, by virtue of compression of the data to be stored or transmitted, this can extend operating times, improve data-rates, and make possible systems that would otherwise not meet power targets. This is where data compression can offer significant opportunity: In principle, if a data compression element consumes less power than that saved by the reduction in storage or transmission power otherwise consumed, then there is potential to realize useful power optimizations for the overall system. This trade-off is therefore a function of compression ratio (CR) and power cost. Whilst it may seem counter-intuitive, it can be shown that the system with the highest CR is not necessarily the best overall solution: a point well-demonstrated by this paper.

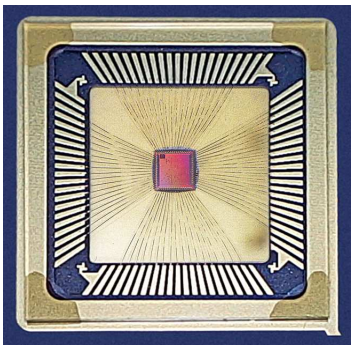

Fig. 1. The 'NOMAD' 84-pin CFPGA Test Chip. The IC was fabricated via IMEC/Europractice using UMC^{TM} 65nm CMOS process, and Faraday^{TM} Standard-Cell Libraries.

Although chip area appears to be an ever increasing resource in standard integrated circuit (IC) design, the silicon area cost of such a design, and therefore any sub-component such as a data-compressor, is still of significant concern. Where techniques such as printed organic semiconductor circuits are concerned, the area-cost constraint is even more demanding [4], [5], and such technologies will surely become more evident in wearable device implementations in the future. Either way, saving circuit area for other useful functionality is always desirable.

Whereas state-of-the-art design philosophy in this field often leads down the path of achieving maximum possible compression rates, with or without information loss, this typically comes at a high cost in terms of hardware area and power. The best compression ratio is not necessarily the best compression outcome as a function of area or power consumption, and this means that there is an interest also in very simple circuits that achieve modestly desirable compression ratios. Such circuits may even be near-optimal in terms of compression of sample bits per picoJoule, even if they are not superior in raw compression ratio terms. Being able to measure these factors using

E-mail: chrisb@cs.york.ac.uk
Manuscript received June, 2019; Accepted August, 2019.

suitable figures of merit (FOM) would also be a very useful tool of convenience with which to compare such candidate systems and their competitors.

This paper presents a compression circuit meeting those expectations of very low power and area cost. This circuit was fabricated as a 65nm CMOS test IC[1], a sample of which is shown in a packaged prototype glass-lid format, in Figure 1. Using a novel algorithm, 'Log2-Sub-Band' Encoding (L2SB), the authors are able to achieve modest compression ratios with meagre transistor counts, and ultra-low power per bit.

After briefly considering some common approaches to compression in this field of application, this paper will introduce 'Log2 Sub-band encoding', and describe it's fabrication on a 65nm CMOS test-chip, subsequent bench testing, and consideration of the implications for compression efficiency using this design. Following this, a performance comparison is made between some alternative compression algorithms, with suitable data-sets, including EEG, ECG, and MEG data-sets. Power savings and area trade-offs are estimated for each case, based on the core power and area data obtained from the initial evaluations presented. The authors introduce several FOM measures to support this comparison process. It is concluded that the power efficiency of the presented circuit is highly competitive for pJ/bit and area cost, both of which are extremely low. Projections are made for storage and transmission power trade-offs achievable by using various compression scenarios. In spite of the presented circuit's inherent simplicity, it is concluded that L2SB could be a favourable choice for future system designs.

II. A BRIEF OVERVIEW OF COMPRESSION OPTIONS

Compression algorithms in the domain of acquisition of continuous time-varying signals, particularly those associated with bio-physiological monitoring, seek to exploit the fact that most signal content consists of long-term low frequency variations and shorter term and smaller scale fluctuations on a more local scale. These signal transitions are rarely extreme in nature. Consequently, adjacent samples are often numerically close together with respect to the full scale signal range possible from one sample to the next. Whereas a signal range might occupy 12 bits, the typical state-change between samples may typically only cover a few of the least-significant bits for any two successive samples. To achieve some form of compression of this information, the simplest possible option might be to simply encode the difference between adjacent samples, also known as DPCM, or Differential Pulse Code Modulation, a technique which has already been in effective use for a very considerable time [6]. If some method is then employed to transmit only the necessary bits representative of the change, then compression would be achieved. There are a variety of ways in which this might be obtained.

In some systems, it is acceptable to use lossy compression to reduce real information content by approximating features of the signal of interest [7]–[9], or to utilize complex hardware to deliver highly domain-specific compression rates [10], [11]. In some cases, techniques exploit relationships between channels in multi-channel recording tasks [12], and any of these methods may be targeted specifically at real-time mobile data acquisition scenarios [13]. However, lossless compression is often an essential requirement, especially in safety-critical and bio-medical domains. Again, approaches vary in complexity. Simple Huffman code-table approaches are widely used, whereas some other techniques employ predictive techniques to reduce the data needing to be transferred whilst retaining lossless characteristics [14], [15]. This leaves the question of complexity to deal with - a complex circuit may well reduce data transmission power significantly, but if it consumes a lot of power in doing so, then this may optimize data volume more than power. In particular, where low data-rates are used, of the order of thousands of samples per second, and indeed sometimes hundreds of samples per second, it may well be observed that the dynamic power of the compressor is rather low, but in contrast, the static power (per sample) could become significant. Small, simple compression circuits, could in theory deliver very low static and dynamic power in these scenarios, and offer valuable gains, even without achieving state-of-the-art compression ratios in themselves.

III. LOG2 SUBBAND ENCODING

A. Design motivations.

The motivation of the authors in developing Log2-SubBand Encoding (L2SB), was to derive an effective, and configurable, compression algorithm for bio-physiological signals with a minimal hardware footprint. A further consideration was that the compression algorithm should be lossless, ruling out a number of algorithms which are capable of high compression ratios (which deliver CR of the order of 5-fold to 20-fold). In any case, such algorithms often require fairly demanding and complex mathematical approaches which immediately create questions in terms of hardware constraints, especially in low-power and real-time scenarios.

In terms of hardware footprint, the authors have three specific concerns: area, static power, and dynamic power. These factors are all considered in the design of L2SB encoding scheme, and this means that the compression ratio, in isolation, is not an overriding performance metric. Rather, the goal is to have moderate compression with minimal cost, and in doing so, demonstrate that a desirable gain in overall system power should be obtainable. Two Figure-of-Merit (FOM) performance metrics are also defined, which will be utilized later in the paper:

- **Data Reduction per pJ (DR/pJ)** which measures the reduction in data bits transmissible after compression, versus the pico-Joule cost of achieving that degree of compression.
- **Data Reduction per um^2 (DR/um^2)**, which measures the average circuit area utilized to achieve the reported data reduction in a given channel of interest.

Data reduction per pJ is calculated as the percentage of data bits saved by compression, divided by the energy consumed by each compression operation. Similarly, data reduction per

[1]The NOMAD IC was fabricated as part of the NOMAD project, funded by UK Gov. Innovate-UK Grant REF 26172- 182148.

um^2 is calculated as the percentage reduction in data bits saved by compression, divided by the area cost attributed to the compression operation. Note the term 'attributed' is used carefully here: where a single circuit can serve multiple data channels, then the area attributable per compression operation is divided by the number of channels supported. If a circuit supports eight channels successfully at a chosen data-rate, and latency, then one eighth of its area is equitably attributable to each channel in terms of compression versus area efficiency.

B. Important points to note.

It is important to note that data-rate, typically in units of kSa/sec, is not always equivalent to clock-rate, typically in the kHz range here. This is because some (though not all) implementations require multiple clock cycles per sample conversion. Also, whilst continuous operation can be envisaged for sample conversion at a given clock rate, a higher clock rate can allow interleaved periods of idle time. For example, a duty-cycle ratio of 10:1 would imply that the circuit spends 90% of its time idle (potentially in a sleep mode) and 10% of its time actively compressing sample words.

C. Log2 Amplitude Sub-Band Compression (L2SB).

The basic principle of L2SB is founded upon the idea of defining amplitude sub-bands, and then comparing the current sample word with the preceding sample, to detect changes between them. In theory, only the changes between samples need to be transmitted in order to convey the original information content, without loss of accuracy. However, in practice this is not easily achieved on a bit by bit basis. Instead, the L2SB encoder sub-divides a given sample word into multiple regions or 'sub-bands', each of which represents a part of the whole sample word. If changes are detected in one of these bands, it will imply that the new state of that particular band must be transmitted.

There are a large number of band permutations that are possible (as will be discussed later). The number of valid combinations increases as a function of sample bit width. In this paper, and for the case chosen for fabrication of the prototype integrated circuit, we choose a relatively straightforward case based upon a 12-bit sample word. This case is illustrated in Figure 2, where it can be seen that a 12-bit sample word is decomposed into three four-bit bands. In this case there is also a notional zero-width band, which represents the case where no bits change anywhere in the sample word. This results in four prefix-code and data-payload combinations, which are also shown, along with the total number of bits to be transmitted in each case.

Based on the relative frequencies of each band combination being utilized to encode data, one can see that compression ratio (CR) can vary from 0.85 to 6.00. Consider an example CR estimation as given in Table I. Here, the average contribution of bits per band combination sums to 8.4 bits in total, meaning that the CR in this case is 1.43 ($12 \div 8.4$), or a data reduction (DR) of approximately 30% of total bits representing successive samples in the original sample words.

TABLE I
COMPRESSION RATIO ESTIMATION EXAMPLE

Prefix code	Bits	Frequency	Effective Bits
00	2	10%	0.2
01	6	40%	2.4
10	10	30%	3.0
11	14	20%	2.8
Total Bits			8.4
CR	(12.0/8.4)		1.43

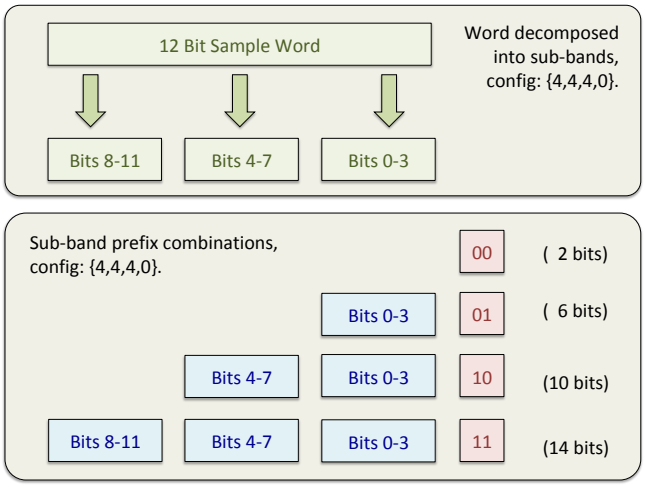

Fig. 2. Log2 Sub-Band Encoding example format (4,4,4,0) encoding.

An important point needs to be noted about how bands may be combined. This is not an arbitrary combination, but utilizes progressive aggregation of bands, such that it is assumed that if a band 'n' has changed, then band 'n-1' is also highly likely to have changed. Less significant bands are always transmitted when a more significant band is transmitted. This means that the actual band combinations that need to be transmitted are fewer, and the prefix code is kept short. Typically the prefix code might be a two-bit code. However where band frequencies make it advantageous, it is possible to have an alternative prefix code such as '0','10','110','111', where the shortest code is assigned to the most frequent band combination. In an extreme case, a zero-width band could be encoded with a single prefix bit, allowing a CR of 12.0 to be achieved, though this is unlikely to be observed happening over sustained periods in realistic data streams.

In this case we chose to make the fourth band a zero-width band, and use a simple two-bit prefix code. Using the author's proposed notation this can be represented as a $\{4,4,4,0\}$ configuration. However we could easily have chosen four three-bit bands with a configuration of $\{3,3,3,3\}$, or indeed any combination of band widths that accumulate to 12 source-bits in total.

Having explored the compression format itself, the next thing to observe is to note what it is that is actually being sub-divided. Taking the raw sample data is one option. Here, if any band is non-zero, then it is treated as requiring transmission. However it is known that this approach is overly sensitive to signal level drift due to low frequency and near-dc signal components in the source signal creating bias toward sam-

ple values with more significant bits being set persistently. To overcome this problem, a common solution is to apply DPCM pre-processing, and thus derive a normalized signal consisting of differences between samples, rather than absolute values. Accumulated differences at the receiver allow the original signal content to be restored. It might be noted however that this introduces more susceptibility to bit-errors, an area that is worthy of more investigation, but outside the scope of this paper. However, an equally effective, but often overlooked difference method, is to use XOR operation to determine locally significant changes. This has the advantage of a significantly simpler circuit design and no cascading of arithmetic stages, therefore potentially much faster. Consider that an arithmetic-differential DPCM circuit requires one full-subtractor circuit per bit, each consisting of 2 XOR-gates, 2 AND-gates, and two inverters, and one OR-gate. Meanwhile, the XOR differential method requires only one XOR gate per bit, a very substantial reduction in gate cost and area when a simple compression system is being designed (less so if the system is of greater complexity). This is an excellent optimization, since the encoder only needs to know that one or more bits have changed in each sample word band, rather than their numeric differences. With either method, the final step is to OR together all of the changed bits within each band to create a band indicator signal to determine if that band has any active changes. Such a scheme is presented in Figure 3, where the same {4,4,4,0} scheme is implemented, using a serial-in-serial-out (SISO) arrangement.

The algorithm is described as follows:-

- Let S_0 represent the previous sample, and S_1 represent the current Sample.
- Let X_1,Y_1 and Z_1 represent three n-bit sub bands of S_1 and X_0,Y_0 and Z_0 represent three correspondingly sized sub-bands of S_0.
- Let a, b, and c represent the True-False or 0-1 result of detecting a difference between corresponding bands, such that a compares X_1 and X_0, b compares Y_1 and Y_0, and c compares Z_1 and Z_0
- A prefix code is now generated via a look-up table/circuit as illustrated in Figure 3, where $\{H_1,H_0\} = f\{a,b,c\}$.
- The prefix is transmitted along with no-data, band Z_1, band Y_1,Z_1, or Bands X_1,Y_1,Z_1, according to the first,second, third, or fourth prefix code being selected.
- Untransmitted bands can be optionally set/cleared, depending upon circuit design needs. For example, in a serial implementation one can simply not shift-out the unused bits.
- Once the comparison process for a,b,and c is completed, the current Sample S_1 may be used to overwrite the previous sample S_0, and this becomes the new previous sample ready for the next encoding cycle.

Circuit functionality is as follows: After the input word is clocked into the input register, it is fed through the XOR array to generate band indicators. Then WE_2 is enabled, capturing the output word (bottom left of diagram), which is formed from the active bands, and the relevant header. Inactive bands are forced to logic '1' in this implementation as this

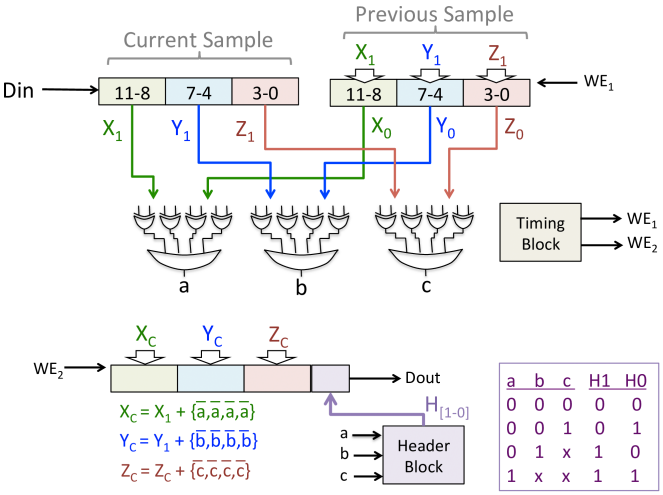

Fig. 3. Serial-In Serial-Out (SISO) L2SB block diagram 4,4,4,0 encoding.

Original Data	XOR	TRANSMITTED	Received
0000 1111 0000	1111 1111 1111	0000 1111 0000 [11]	0000 1111 0000
0000 1111 0001	0000 0000 0001	0001 [01]	0000 1111 0001
0000 1100 0001	0000 0011 0000	1100 0001 [10]	0000 1100 0001
0000 1100 1000	0000 0000 1001	1000 [01]	0000 1100 1000

Fig. 4. L2SB 4,4,4,0 Coder-Decoder Example.

automatically pads the serial output line high after the valid encoded bits have been shifted out. Finally, the circuit retains the current input word, by loading it into the relevant register (top right of diagram) when WE_1 is enabled, and which then becomes the 'previous' word for the next compression cycle.

The output word is prepended by a header prefix code, in this case a two-bit code $H_{[1-0]}$, generated from a very simple logic function, almost identical to a standard 4:2 priority encoder. The timing circuit is based upon a simple binary counter.

Figure 4 illustrates the coding-decoding mechanism with a simple example, to demonstrate that this coding method works correctly. As one can observe, the received data is identical to the original data. The XOR column represents the bit changes between the current sample and the preceding one. The first sample in the sequence (typically after reset) is always treated as if all bits have changed, since a full sample word is required at the start to establish the reference point. If this reference point is refreshed at regular intervals, as a frame start value, it becomes possible to detect transmission errors within frames of chosen length, since progressive reconstruction of erroneous data will eventually be found to disagree with the full-word value at these points. Indeed, full-range values are often incidentally transmitted within a frame too, thus allowing for potential earlier detection. This is a topic worthy of further investigation, but outside the scope of this paper. In the given example, 36 bits are used to convey four 12-bit samples, giving a CR of 1.33 (48/33).

D. L2SB Encoder Implementation and Verification

Now that the basic functionality of the Log2 Sub-Band Encoder has been established, and described, we turn attention to it's implementation. The authors were fortunate to have the opportunity to include an L2SB encoder test circuit on a more complex IC fabrication project, utilizing spare pins and chip area. The low pin-count available meant that a serial-in-serial-out (SISO) implementation was chosen for the fabrication and verification. This design had the advantage of being pipelined, such that new data words can be shifted inward to the compression stage, whilst the previous compressed bit pattern is shifted outward.

The circuit was modelled in VHDL, and then synthesized to a gate-level description. Cadence design tools were used throughout this phase of development. The 65nm Faraday™ standard cell library was used for HDL synthesis, and the design was then targeted at a 65nm CMOS fabrication technology: UMC™ 65N Logic/Mixed-Mode/RF CMOS process, with core and I/O voltages of 1.2v and 1.8v respectively. Automatic clock gating was enabled at the synthesis stage, and this resulted in approximately 2/3rds of flip-flops being gated, with 70% reduction in dynamic transitions. Figure 5 shows a trial layout for the synthesized circuit, where the dimensions of the module are 13um by 14um, giving a maximum circuit area of 182 um^2, if one ignores the unused space at top right, or around 162um^2 otherwise. For this particular tapeout, the layout relied upon standard cell abutment, but if a full custom layout methodology was used, with transistor folding and other layout optimizations, then this area cost could no-doubt be further reduced. Maximum frequency was 714MHz, giving a minimum input-output latency of 22ns for a raw sample to be compressed to an encoded state.

A PIPO implementation was also designed and tested at the layout level(but not fabricated) in order to give area cost, power data, etc, with identical compression behaviour. This design had an area cost of 5% less than the SISO model (approximately 155um^2), and used 38 flip-flops compared to 43 for SISO as a result of eliminating the cycle-state counter needed for the SISO model. This design is very similar to the design shown for the SISO model, but the input sample word and output sample word are written to and read from (respectively) in a bit-parallel fashion, rather than bit-serial as is the case for SISO.

The L2SB circuit was incorporated into the larger project chip layout, and fabricated via the European Europractice service to academia, via the IMEC centre. The chips were then packaged with an 84-pin CFPGA package, with some also supplied as glass-lid samples for display (one of which is shown earlier in Figure 1).

Validation of the fabricated L2SB encoder was performed with bench-test equipment, comprising of a Zynq™ FPGA board to generate test signals, and measurements taken on a LeCroy™ WaveSurfer-440 digital oscilloscope. Oscilloscope screen-shots, from the operational bench tests, are shown in Figure 6. Screen-colors are inverted for clarity in print. This shows three test cases, covering the single, double and triple band encoding cases. This implementation includes a start-bit

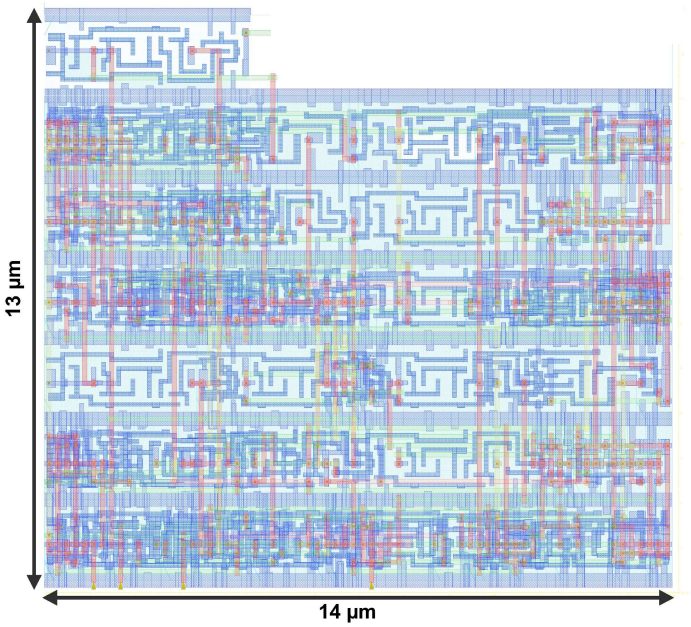

Fig. 5. L2SB 4,4,4,0 Coder-Decoder Example Layout, 65nm UMC CMOS.

(logic-low) and end-bit (logic-high) feature, shown in blue in the figure, which allowed data items to be framed for testing. More comprehensive tests were performed using automated test-pattern stimulus. The L2SB encoder passed all validation tests, and the fabricated data compressor module was fully operational. One limitation of incorporating the SISO L2SB test circuit into a more complex system design was that taking isolated power measurements from the chip was not possible. In this paper we use a data-driven post-synthesis power estimation methodology as described in the following section. This also means that SISO and PIPO can be compared on equal measures.

IV. L2SB Performance: Evaluation Test Cases

In order to evaluate L2SB encoder performance, test cases that relate to real-world application scenarios were chosen, hardware implementations of suitable L2SB compressor circuits were implemented, and power measurements were obtained from synthesis tools using Value Change Dump (VCD) stimulus files generated from simulations of each compression test-case using actual data streams. Two L2SB models were tested, the SISO (Serial In Serial Out) implementation as fabricated on the test chip, and an additional PIPO (Parallel in Parallel Out) model, whereby whole data words are clocked in on each successive clock cycle, with compressed data clocked out one cycle later.

A. Initial Power Analysis

For initial power tests, the Bonn University EEG Epilepsy data-set was utilized [22]. This data-set contains the following EEG test data-sets (Referred to by the originators as File-Sets O, F, and S, respectively).

- EEG1: Healthy subject [O.zip]
- EEG2: Epileptic subject, no seizure [F.zip]

- EEG3: Epileptic subject, with seizures [S.zip]

Each file set consisted of 100 segments of about 24 seconds of EEG data (more exactly, 4096 samples captured at 173.6Hz). These were mapped to a 12-bit positively ranged binary scale and compressed using L2SB VHDL implementations. Logic-level simulations of the compressors were used to generate value-change-dump (.vcd) files which were then used in the RTL synthesis tool to generate power estimates. The resulting data, normalized to a 1kHz continuous conversion rate, is summarized in Table II, where CR is tabulated alongside static and dynamic power estimates for both SISO and PIPO implementations, with maxima, minima, mean, and standard deviation to 95% limits. The scatter-graph of compression ratios is also given for the three data subsets within the whole data-set, as given in Figure 7, where it can be seen that CR is noticeably banded according to the three test cases. It can also be seen that compression ratios are significantly higher in the seizure patients during non-seizure EEG monitoring cases. This may well relate to the reported differences in EEG power spectra components for patients under similar conditions [16], which could account for reduced inter-sample differences and thus the higher compression rates.

It can be seen that CR is, on average, around 1.56 +/- 16% at 95% limit, with CR as high as 2.05. This translates into an average data reduction (DR) of over 35%. However, this figure uses an equally weighted average. In practice, a patient may have seizures infrequently (one would hope that even a 100:1 ratio of seizure versus non-seizure data is pessimistic), or a system may record only the seizure events (with assistance of a detection algorithm), though data volume in this case is relatively small anyway. This means that there are actually multiple CR scenarios to consider, some of which are postulated in Table III.

Examining the power data from post-synthesis simulations, with actual data-set stimulus, static power (at a 1kSa/sec continuous conversion mode) is found to be almost constant for both implementations, and significantly larger than dynamic power at this data-rate. Dynamic power is relatively small by comparison, and shows small variance of a few percent. It can be concluded that at these sample compression rates, power is highly consistent across a fairly significant range of signal behaviour. At 1kSa/sec data-rate, overall power consumption averages 234nW for PIPO. For easier comparison, it is potentially more convenient to measure power per-bit for each compression event. This measure equates to 19.6pJ/bit for PIPO, at this data-rate.

B. Further Optimization

At low device clock-rates, such that sample compression is a continuous back-to-back operation for successive samples, the device remains powered up 100% of the time. Alternatively, power-gating, with an on/off duty cycle, allows conversions to happen at higher clock-rates, with intervening power-down 'sleep' phases. Although we have not implemented a power-gated L2SB design in silicon, our initial evaluation suggests that static power could be reduced by 50% to 75% using simple on/off power gating, since almost all of the logic has

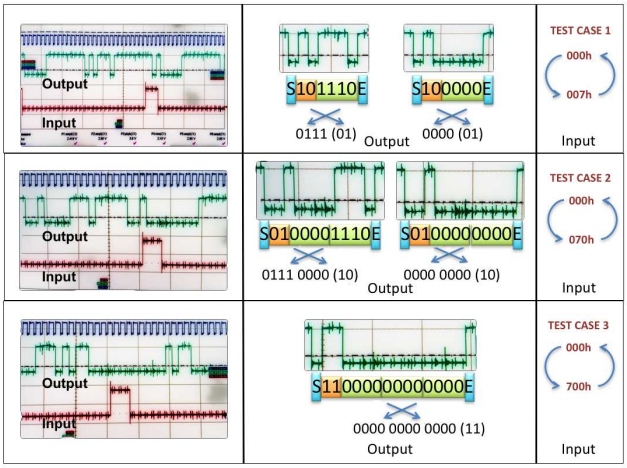

Fig. 6. LS2B 65nm CMOS ASIC Test Waveforms. (Images show Lecroy Wavesurfer 440 Screen-shots, colors inverted for print clarity.)

TABLE II
INITIAL POWER AND CR ESTIMATES FOR 1KSA/SEC CONTINUOUS SAMPLE-WORD COMPRESSION, AND EEG DATA-SET

Compression Ratio (CR)				
	healthy	SF	Seizure	ALL
CR min	1.45	1.1	1.57	1.38
CR max	1.62	1.57	2.05	1.75
CR ave	1.58	1.29	1.8	1.56
Std Dev 2	6%	25%	17%	16%
PIPO Power Consumption				
	healthy	SF	Seizure	ALL
Ps min nW	228.52	230.7	228.89	229.37
Ps max nW	234.57	233.01	232.79	233.46
Ps ave nW)	231.02	232.01	230.56	231.2
Std Dev 2	1%	1%	1%	1%
	healthy	SF	Seizure	ALL
Pd min nW	3	3.1	2.62	2.91
Pd max nW	3.19	3.21	2.78	3.06
Pd ave nW	3.12	3.14	2.71	2.99
Std Dev 2	4%	2%	5%	4%
SISO Power Consumption				
	healthy	SF	Seizure	ALL
Ps min nW	147.17	147.2	147.24	147.17
Ps max nW	147.31	147.68	147.4	147.68
Ps ave nW)	147.23	147.29	147.29	147.27
Std Dev 2	0%	0%	0%	0%
	healthy	SF	Seizure	ALL
Pd min nW	12.79	12.61	13.3	12.61
Pd max nW	13	12.74	13.62	13.62
Pd ave nW	12.91	12.68	13.37	12.99
Std Dev 2	1%	1%	10%	4%

TABLE III
HYPOTHETICAL EEG MONITORING SCENARIOS AND ESTIMATED COMPRESSION RATIOS

Mode	CR estimate	data reduction
Selective Seizure Recording	1.80[1]	44%
Continuous Monitoring (1% seizure)	1.29	22%
Pre-diagnostic Healthy/Epileptic	1.29-1.58	22-35%

[1] *compression for recorded portions. Actual data volume would be reduced much more, due to recording only the seizure events.*

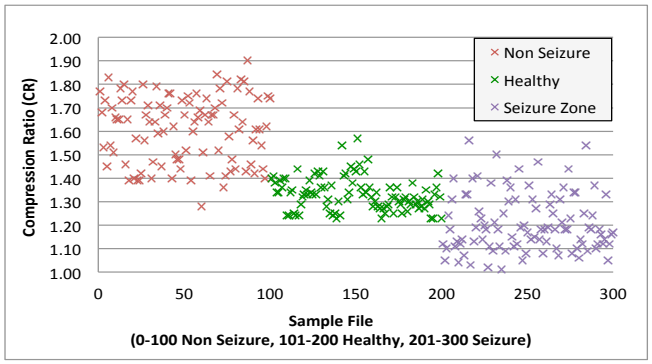

Fig. 7. Bonn EEG data sets, plotted in functional groups (Seizure Free, Healthy, and Seizure Event).

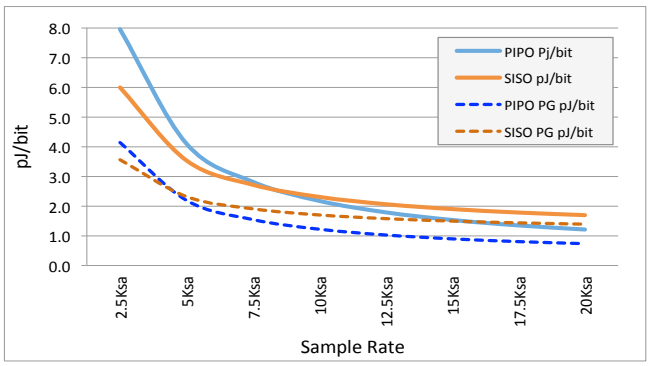

Fig. 8. Energy consumption Trade-off for PIPO vs SISO L2SB circuits. Showing energy consumed per bit, per sample conversion event.

TABLE IV
ENERGY ESTIMATES, BONN EEG DATA-SET, SHOWING PJ/BIT

Continuous Operation				
	1kSa/s	10kSa/s	100kSa/s	1MSa/s
PIPO Ps, pW/bit	19.27	1.93	0.19	0.02
PIPO Pd pW/bit	0.25	0.25	0.25	0.25
PIPO tot. pJ/bit	19.52	2.18	0.44	0.27
SISO Ps pW/bit	12.27	1.23	0.12	0.01
SISO Pd pW/bit	1.08	1.08	1.08	1.08
SISO tot pJ/bit	13.35	2.31	1.21	1.09
Power Gated 100:1				
	1kSa/s	10kSa/s	100kSa/s	1MSa/s
PIPO Ps, pW/bit	9.73	0.97	0.1	0.01
PIPO Pd pW/bit	0.25	0.25	0.25	0.25
PIPO tot. pJ/bit	9.98	1.22	0.35	0.26
SISO Ps pW/bit	6.2	0.62	0.06	0
SISO Pd pW/bit	1.08	1.08	1.08	1.08
SISO tot pJ/bit	7.28	1.7	1.14	1.08

non-persistent data content. The principle here is a partial power-gating strategy, to retain the previous sample in the relevant data-latches without any power-gating, whilst applying power-gating to most of the remaining circuitry, since only the previous sample represents persistent state information between successive sample encodings.

Evaluations are given of power consumption per bit in Table IV. With a 100:1 duty cycle, a moderate 50% reduction in leakage current by power-gating, and a 10kSa/sec data-rate (heading toward the upper end of usual per-channel bio-physiological signal sampling rates of 10's of kHz), it is estimated that the PIPO L2SB compression circuit would consume only 1.22pJ per bit, and about 1.7pJ for this SISO CMOS implementation at a 1.2v core voltage. It is also noted that at higher frequencies, PIPO and SISO move closer together for consumption per bit, regardless of power-gating usage. Above 1MSa/Sec power consumption reaches as low as 250 femtoJoules per bit for PIPO implementation, and this is roughly constant beyond this point, up to maximum operating data-rates. This diminishing benefit of power gating, at high frequencies, is due to static power becoming negligible with respect to increasingly dominant dynamic power consumption.

It is clear that SISO implementation is outperformed by PIPO L2SB in terms of power consumption, particularly for higher data-rates. For comparison, Figure 8, shows the picoJoule-per-bit energy consumption of the four cases for selected sample rates up to 20kSa, showing the comparative performances with and without power-gating.

C. Broader compression evaluations

Evaluation of direct power cost is one factor of interest for L2SB encoding. However, to gauge the overall benefit of L2SB encoding in a complete system, it is also necessary to evaluate its compression performance over a broader range of data-sets. Furthermore, whilst one particular configuration has been chosen for the implementation, there are many possible configurations of L2SB encoding. For an n-bit sample word, and an LS2b band configuration comprising of bands 'a','b','c', and 'd', every possible set of values of a,b,c and d, that sum to a total of 'n', are potentially valid permutations. So for example, a configuration {1,2,4,5} has a total of 12 bits, but each band has its own unique width, and each permutation will deliver a different compression ratio to the default {4,4,4,0} configuration. If some validity constraints are applied, for example, only allowing the least significant band to have the option of zero width, then a 12-bit sample word has around 300 valid permutations, out of over 1300 candidates.

Taking these valid permutations, and applying each of them in turn to a data-set, allows us to determine the best permutation(s) for a given data behaviour. In effect, we can tune the algorithm to suit the dynamics of the particular kind of data being compressed. For example, applying L2SB encoding, in XOR mode, to the whole Bonn EEG data-set (file groups O,F, and S), the compression ratio for the 4,4,4,0 configuration is found to be 1.31 (a saving of 24% of total sample data). However, when all valid band configurations are considered, it is apparent that there were better choices that could have been made. This can be seen in Figure 9, which plots all 299 permutations in terms of CR achieved in each case. Analysis of the individual results identifies configuration {-,4,3,5} as the best choice, with a CR of 1.39 and a bit storage/transmission saving of 28%. Further work may allow this band-tuning concept to be derived from the distribution of dynamic changes in the data (such as a Gaussian curve),

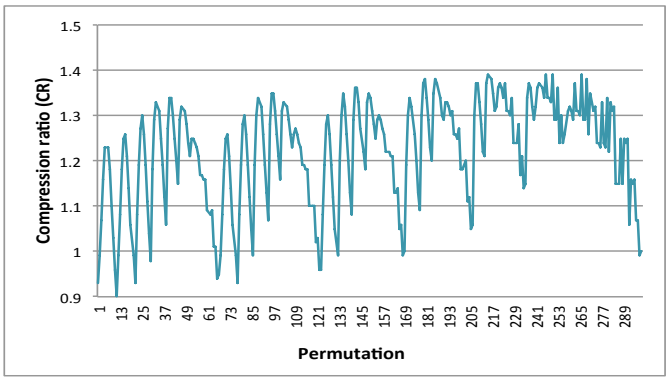

Fig. 9. L2SB with XOR (Bonn EEG Dataset). Where CR is plotted for each valid permutation 0 to 299. The pattern of variation results form the way the band permutations are iterated.

this is an interesting idea, but providing a mathematical basis and proof is beyond the scope of this paper.

A further observation may be made regarding band choices. In a typical system, we envisage the band configuration being static for a given application, or recording session, and either fixed in software or hardware or reset at power-on initialisation. No information about band settings need to be transmitted. A more complex solution might employ dynamically changing bands, and these might be indicated at the start of each new frame, however we do not investigate this idea here.

For the purposes of an initial 'reality check' comparison, a Huffman code-book, optimistically trained on 100% of the test data, and using DPCM precoding, delivers a CR of 1.98, a 49% reduction in data bits. However there are significantly higher costs associated with this compression approach, as will be highlighted later, whilst it is also true that compression rates may be lower if the Huffman code-book is trained using a reference data set and then used with new 'unseen' signal input. There is now a potential choice - save about 50% of data storage and/or transmission with high hardware cost, or save 30% with a much lower hardware footprint.

Proceeding with more comprehensive tests, Table V presents a number of data-sets, evaluated for achievable compression ratios. This tabulates the compression ratio achieved by the best L2SB configuration in each case, alongside Huffman code-book. In all cases the Huffman code-book is trained on the whole data-set, and code-books are fully populated with all possible input symbols. This latter point is essential: it is possible to create a code-book that partially populates a 12-bit symbol table (for example only 800 symbols out of a possible 4096 may be used by the training set), but this is not a valid representation for a real system. Otherwise, what would the hardware do when presented with a previously unseen symbol? A valid Huffman code-book must assign a code-word to every possible lookup value if it is used in a real-time compression application. The data-sets have the following details:

- Bonn University EEG Database, 12-bit 173Hz, 300 traces, no rescaling required.
- MIT CHB EEG scalp electrode data, 256Hz [24], [26].

TABLE V
COMPRESSION RATIOS FOR SELECTED DATA-SETS

DATA-SET	Type	L2SB.std	L2SB.best	Huffman
a. BONN EEG	EEG 173Hz,12bit	1.31 (x-y)	1.39 (x-y)	1.98 (x-y)
b. MIT CHB	EEG 256Hz,11bit	1.87 (1.67-2.25)	1.94 (1.72-2.28)	2.02 (1.87-2.32)
c. MIT BIH	ECG 360Hz,11bit	1.57 (1.34-1.79)	1.63 (1.41-1.83)	2.6 (2.28-2.95)
d. YI MEG	MEG 678Hz,12bit	1.36 (1.12-1.39)	1.54 (1.49-1.59)	1.86 (1.70-1.94)

- MIT BIH ECG arrhythmia database, 360Hz, 11bit, re-scaled to 12bit full scale. [25], [26].
- York Instruments nanoTesla MEG data, 24-bit 678 Hz, re-scaled to occupy 2/3rds of 12 bit scale.

For the MIT BIH Arrhythmia ECG data-set, 48 separate data files were analyzed. Each data file is relatively short and contains differing aspects of ECG observation, therefore variations are likely. In this case, the standard deviation at 95% limits is 13%, centered around an average CR of 1.57.

For the MIT-CHB Scalp-electrode database, due to the size of the complete database, fourteen files were analyzed, each containing twenty-eight channels of data (40Mbyte each file), four of which contained seizures within the data. Although this data is stated as being acquired as 16-bit, all of the files used were able to map onto a 12-bit range such that, on average, sample range covers 90% or more of the full signal range without rescaling. L2SB Compression performs well in this data-set, achieving results quite close to the Huffman compression model, where CR is found to be 2.02 for Huffman vs 1.94 for the optimal L2SB band configuration. This is much better than the Bonn EEG dataset, perhaps due to the higher sample rate.

For the MEG data-set, provided by York Instruments ltd (UK), the data was analyzed as a whole, yielding a CR of 1.36 for the default choice of configuration {4,4,4,0}. However the best case configuration, {-,5,2,5} with three bands and prefix codes '0','10' and '11', achieved a CR of 1.54, highlighting the importance of identifying the best band configuration for a given data behaviour. Note that these permutation analyses included every possible variation of 1,2,3 and 4 band encoding, with all possible prefix options. Further comparisons, with other work in the field with the same or broadly comparable data sets, are given in Table-VI.

D. Performance Comparisons

To assess the usefulness of any compression method, and implementation, there are three major concerns. The first relates to the compression ratio achieved, since this determines how much data transmission or storage effort is saved. The second concern is the power consumed whilst achieving the reduction in data needing to be managed. The third consideration is circuit area, since large circuits may be unwelcome additions to an SOC or FPGA design.

TABLE VI
CR COMPARISONS WITH SELECTED REPORTED COMPARISON CASES

REF	TEST CASE	CR	L2SB	DPCM-HUF	Data Sets (12bit)
[32]	ECG[1]	2.53	1.63	2.6	MIT-BIH (c)
[21]	ECG[2]	2.38	1.63	2.6	MIT-BIH (c)
[34]	ECG[3]	2.67	1.63	2.6	MIT-BIH (c)
[36]	ECG[4]	3.07	1.63	2.6	MIT-BIH (c)
[21]	EEG[5]	1.37	1.67	2.0	EEG (a,b)
[9]	EEG[6]	2.51	1.67	2.0	EEG (a,b)
[33]	EEG[7]	2.77	1.67	2.0	EEG (a,b)
[33]	EEG[8]	1.34	1.67	2.0	EEG (a,b)
[33]	EEG[9]	1.97	1.67	2.0	EEG (a,b)

note-1: 12-bit ECG, Adaptive predictor, 2-stage Huffman encoder , note-2: 10-bit ECG, 256 Sa/s, DPCM plus Golumb-Rice encoding , note-3: Adaptive Linear Predictor, MIT-BIH Dataset , note-4: 12-bit ECG, Template matching, subset of MIT-BIH data set. . note-5: 10-bit EEG, 128 Sa/s, DPCM plus Golumb-Rice encoding note-6: 12-bit EEG, FLO/FIX2 Multichannel predictor, Golumb-Rice Encoding note-7: 16-bit EEG, DPCM+Karhunen-Loeve-transform+Temporal-Decorreleation, Huffman Encoder note-8: 16-bit EEG, GZIP note-9: 16-bit , jpeg2000

A good starting point, for the basis of comparative performance analysis, is Huffman Encoding, since this is very well defined, and widely used by researchers to measure their own compression algorithms against. It is effectively a 'standard measurement' and allows those novel techniques to be compared with those proposed here, using Huffman performance as a common reference point.

Techniques that employ code-books, such as variations of Huffman encoding, can potentially imply rather large circuit area cost, due to the need for look-up ROMs or RAMs with thousands of locations and tens of bits per location. However, one advantage of a code-book is that it can be accessed multiple times per sample epoch to permit one code-book to generate compressed encodings for multiple simultaneously acquired channels. Thus, in the case of ECG, where there are only a handful of channels, a code-book may be quite costly in terms of circuit area, whilst in a 300-channel MEG system, it might potentially be more desirable. In contrast, for the simple implementations of L2SB discussed in this paper, a multi-channel system would typically require a separate circuit for each channel, albeit each being an instance of a very small circuit.

1) A Basis for Huffman Power Estimation.: In order to evaluate Huffman compression circuitry, suitable reported 65nm CMOS memory costs have been considered [27]–[29], from which it was established that 200pw for static power per bit and 3.3 um^2 area cost per bit-cell is a reasonable assumption. Huffman code-book area and static power cost can thus easily be estimated. For example, a 4 Kword by 39 bit code-book has a total of 159,744 bits, total leakage power of 31.9uW, and an area cost of 527,155 um^2. This area cost is highly significant even for a low-nm silicon CMOS design, but even more so for emerging technologies such as thin-film organic semiconductor sensor circuits. An important point to note here is that the Huffman circuit methodology is a 'plain' Huffman baseline approach. Whilst techniques exist to truncate and augment Huffman code books to reduce hardware cost [32], these are often intimately interlinked with particular predictor algorithms, and are therefore not easy to generalise. There are also many potential ways to condense a Huffman code book, and it therefore makes sense here to use a single well-defined case as a baseline.

Dynamic read-power per code-book symbol look-up can also be estimated. Based upon work in the field, a typical value was found to be 5pW/bit per read. Implying 195pW for a 39-bit read. Finally, therefore, at 1kSa/Sec, total power is 32.15uW, power per sample is 32.15nw, and energy per bit is 2.7nJ per bit. Huffman coding is more power efficient per-sample at higher data rates, which may include interleaving multiple channels. For example, supporting 10 channels at 10kSa/sec gives a code-book throughput of 100kSa/Sec and power consumption of 428pJ per sample bit.

2) Huffman Power and Area Projections: Taking relevant scenarios implemented using L2SB and the described Huffman code-books, a comparison is presented in Table VII, where data is provided for power and area cost for Huffman and L2SB compression circuits operating at a 1KSa/sec data throughput. Power and area calculations are given for systems with 1, 4, and 8 channels. For L2SB compression, the power and area per channel is constant, whilst total power and total area increase proportional to channel count. For Huffman compression, total power rises as a function of channel count, but power cost per channel reduces, since a single circuit with associated leakage is serving, 1, 4, 8, or more channels.

Note that the Huffman code-book size is calculated as follows: suppose a 12-bit sample word is compressed by Huffman encoding and the maximum resulting code-word size in the code-book element width is 18 bits. Now, although the code-word 'w' occupies 18 bits, an associated code-word size indicator 's' is also needed to inform the next stage of the system as to the code-word size being provided in the current compression cycle. Since 's' must be a binary number, it must be 5 bits in this case, since $2^5 = 32$, which is greater than or equal to 18, whilst $2^4 = 16$, which is too small. Therefore, the total code-book is the binary lengths of 'w' and 's' combined (in this example case 23 bits).

Examining again Table VII, it can be seen, then, that a single channel Huffman compression circuit might consume around 32.2nW (2.69pW per bit) at 1kSa/Sec. A PIPO L2SB encoder using a 100:1 duty cycle to support power gating, and assuming a 50% static power saving in sleep mode, would consume around 120pW (10pW/bit), representing an approximate 270:1 power advantage for L2SB. However, the L2SB circuit (in its simplest embodiment) must be duplicated in area for each additional channel concurrently supported, whereas the single Huffman code-book can be interleaved between channels, effectively sharing the static power burden. It is seen that for eight channels, Huffman consumes only 350pw/bit, a much improved figure, but still a 35:1 ratio of power consumption per channel compared to L2SB. If a wide range of channel counts are considered, as given in Table VIII, a single-stage huffman-DPCM model tends eventually toward a figure of around 16pw/bit for very high channel counts (of the order of 1000's). However, few bio-physiological sensing applications have such demands.

For circuit area overhead, L2SB and Huffman are also compared in Table VII. Here it is possible to see that for a single channel, Huffman is vastly larger than L2SB in terms

of circuit area, averaging 529.14um^2 for one-channel DPCM-Huffman, versus 155um^2 for L2SB. For multiple channels, L2SB area cost is a linear product of channel count, whereas it is a fixed cost for Huffman. Consequently, area cost per channel is always 155um^2 for L2SB, but reduces for Huffman as channel counts increase.

3) Figure of Merit (FOM) measurements: The power and area data, quoted in the preceding section, is a raw power and area cost estimate, and does not take into account the differing compression ratios delivered by each case. Yet, compression ratio is an important element. Therefore, a further perspective on relative performance can be obtained by taking the calculated power and area data for the L2SB/Huffman comparisons, and utilizing these to generate figures of merit. There are two figures of merit used here, as introduced earlier: Data Reduction per pJ (DR/pJ), and Data Reduction per um^2 (DR/um^2).

This FOM data is also calculated in Table VII, and it can be seen that the relative efficiency of L2SB and Huffman algorithms in achieving a given compression goal can be evaluated using these measures. In power terms, Huffman is almost 27 times less power efficient per bit of data reduction achieved through compression than L2SB, for an 8-channel system, and 192 times less efficient for a single-channel mode. For area efficiency, L2SB is several thousand times more efficient in delivering data bit reduction via compression for a once-channel system, and still over 300 times better for eight channels, compared to Huffman. A further area comparison (albeit less precise) can be made with other work in the field, provided that area-costs are scaled to the same process node (65nm). Such a comparison is given in Table X, where a variety of loss-less compression schemes are reported. For these cases, it can be observed that L2SB delivers considerably more data compression per um^2 of silicon than the cited cases.

Additional area comparisons are given in Table X, where a variety of reported compression scheme VLSI implementations are compared for estimated area cost at 65nm, with data scaled from process nodes, as reported, where necessary.

It is important to remember, here, that these figures relate to data compression as a function of power or area, not absolute compression ratio. L2SB is not compressing data volume 30 times more than Huffman, indeed Huffman delivers better absolute compression than L2SB. What the data shows is that L2SB achieves 30 times more compression per pico-Joule. Huffman consumes disproportionately more power and area to achieve its superiority, whilst L2SB is moderately inferior in compression ratio, but with much lower resource cost. At this point, a question must arise - is more compression at much higher cost better than less compression at extremely low cost? This can be answered when the goal of compression is finally dropped into place - we wish to reduce one or more system overheads, and primarily memory storage requirements, memory storage power, and data transmission power. This is examined in the next section.

V. System Level Performance Trade-offs

At the system level, there are two major areas of concern in the context of this study. The first concern relates to systems

TABLE VII
HUFFMAN VS. L2SB FIGURE OF MERIT ESTIMATIONS

HUFFMAN Compression Versus Power (1ksa/s)

	Ch	CR	CWL (bits)	Ps/Sa (pW)	Pd/Sa (pW)	pW/Sa /ch (pW)	DR/pJ (1 x 10-4)
MIT.CHB	1	2.02	42	34286	210	34496	0.6
MIT.BIH	1	2.6	34	27755	170	27925	0.9
YIMEG	1	1.86	44	35919	220	36139	0.5
BONN	1	1.98	37	30204	185	30389	0.7
AVE.1	1	2.12	39	32041	196	32237	0.7
AVE.4	4	2.12	39	32041	785	8207	2.6
AVE.8	8	2.12	39	32041	1570	4201	5

PIPO L2SB Compression Versus Power (1kSa/s, 100:1 Duty Cycle)

	Ch	CR	CWL (bits)	Ps/Sa (pW)	Pd/Sa (pW)	pW/Sa /ch (pW)	DR/pJ (1 x 10-4)
MIT.CHB	1	1.39	n/a	117	3	120	116.1
MIT.BIH	1	1.94	n/a	117	3	120	162
YIMEG	1	1.63	n/a	117	3	120	136.1
BONN	1	1.54	n/a	117	3	120	128.6
AVE.1	1	1.63	n/a	117	3	120	135.7
AVE.4	4	1.63	n/a	467	12	120	135.7
AVE.8	8	1.63	n/a	934	24	120	135.7

HUFFMAN Area Versus Compression

	Ch	CR	CWL (bits)	Total Area (um^2)	Area/Ch (um^2)	DR/um2 (1x10-6)
MIT.CHB	1	2.02	42	566,503	566,500	0.9
MIT.BIH	1	2.60	34	458,598	458,600	1.3
YIMEG	1	1.86	44	593,480	593,480	0.8
BONN	1	1.98	37	499,062	499,060	1
AVE.1	1	2.12	39	529,411	529,410	1
AVE.4	4	2.12	39	529,411	132,353	4
AVE.8	8	2.12	39	529,411	66,176	8

PIPO L2SB Area Versus Compression (1kSa/s, 100:1 Duty Cycle)

	Ch	CR	CWL (bits)	Total Area (um^2)	Area/Ch (um^2)	DR/um2 (1x10-6)
MIT.CHB	1	1.39	n/a	155	155	1810
MIT.BIH	1	1.94	n/a	155	155	3126
YIMEG	1	1.63	n/a	155	155	2494
BONN	1	1.54	n/a	155	155	2262
AVE.1	1	1.63	n/a	155	155	2481
AVE.4	4	1.63	n/a	620	155	2481
AVE.8	8	1.63	n/a	1240	155	2481

Where CR = Compression Ratio, CWL= Huffman Code Word Length, Data Reduction per pJ (DR/pJ) for Power, Data Reduction per um2 (DR/um^2) for Area.

TABLE VIII
POWER AND AREA COSTS: HUFFMAN VS L2SB (MULTI-CHANNEL).

HUFFMAN AT 1kSa/s

Channels	1	4	16	256	512
Pd nW	195	780	3120	49920	99840
Ps nW	31,949	31,949	31,949	31,949	31,949
Tot nW	32,144	32,729	35,069	81,869	131,789
PJ/Ch/Sa/bit	2679	682	183	27	21
Tot area (um^2)	529,410	529,410	529,410	529,410	529,410

PIPO L2SB at 1kSa/s, 100:1 duty cycle

Channels	1	4	16	256	512
Pd nW	3.0	12.0	47.9	765.9	1,531.7
o Ps nW	116.8	467.0	1,868.1	29,889.4	59,778.7
Tot nW	119.7	479.0	1,916.0	30,655.2	61,310.4
PJ/Ch/Sa/bit	10.0	10.0	10.0	10.0	10.0
Tot area (um^2)	155.0	620.0	2,480.0	39,680.0	79,360.0

RELATIVE COST (LS2B=1)

Huff Power	268.4	68.3	18.3	2.7	2.1
Huff Area	3,415.5	853.9	213.5	13.3	6.7

TABLE IX
SYSTEM POWER TRADEOFF - FLASH POWER VERSUS COMPRESSION.

HUFFMAN AT 1kSa/s					
Channels	1	4	16	256	512
pJ/Ch/Sa/bit	2,679	682	183	27	21
Data Reduction	47 %	47 %	47 %	47 %	47 %
Write pW saved/ch	2.4	2.4	2.4	2.4	2.4
pW Saved/ch	-2,676	-679	-180	-24	-19
Tot uW saved	-2.7	-2.7	-2.9	-6.2	-9.8
PIPO L2SB at 1kSa/s, 100:1 duty cycle					
Channels	1	4	16	256	512
pJ/Ch/Sa/bit	10.0	10.0	10.0	10.0	10
Data Reduction	38 %	38 %	38 %	38 %	38 %
Write pW saved/ch	1.9	1.9	1.9	1.9	1.9
pW Saved/ch	-8.1	-8.1	-8.1	-8.1	-8.1
Tot uW saved	-0.01	-0.03	-0.13	-2.07	-4.15

in which data is stored in non-volatile on-board memory, for later access. This is typically achieved via an on-board flash memory components, and often of the order of several gigabits capacity. Examples of this use-case include wire-free miniature data-recorders [17], [18]. Such systems are designed to operate in a wearable ambulatory mode of operation, with neither a wired (umbilical) connection to a master unit, or a wireless (radio tethered) connection to a base-station. Here, the major power costs associated with data are the storage costs in terms of (a) storage capacity, and (b) data write power.

A. Compression-Storage Trade-offs

Considering non-volatile memory storage first, it is self evident that reducing data by 30% via compression will increase potential storage capacity of the system by the same amount. There are no major insights here, other than to say a higher compression rate is better if storage capacity is the only concern. However, for power consumption, the picture is somewhat different. Typical flash memory chips have a write power consumption of around 3-4pW per bit [19], [20], though write operations are typically performed in blocks after accumulating enough bytes to fill a write page. It is often overlooked that bytes must first be written to the flash page buffer, before the actual write is completed internally, and this consumes more power. Therefore, whilst the typical page write time, averaged on a per-byte basis, may be of the order of 120-150ns per byte, the data-transfer to internal buffer may require an additional 20-40ns of time spent in active power mode per byte written. On this basis, a figure of 5pW per bit appears to be a reasonable benchmark for comparing flash write power against data compression power, on a bit for bit basis.

Taking the figure of 5pW per bit for flash-memory write-power, and employing this in a trade-off between compression-power consumed, versus flash-memory write-power saved (by reduced volume of writes), the analysis presented in Table IX is derived. This analysis utilizes the same data-sets as those used for Tables VII and VIII. For both Huffman code-book compression, and L2SB, it is clear that there is no direct benefit between compression ratios achieved versus flash storage power cost. However, for a 16-channel system L2SB has relatively small power cost to achieve worthwhile storage compression, whilst Huffman has relatively high power penalty.

TABLE X
AREA COST FIGURE-OF-MERIT (FOM), L2SB VS. BROADER COMPARISON SCHEMES.

REF	Details	CH	Area est. um^2	CR	DR	DR/um^2 (65nm)
[21]	ECG/EEG[1]	19	7563	1.875	0.47	1172
[35]	ECG[2]	1	9910	2.38	0.58	59
[15]	ECG[3]	1	5998	2.43	0.59	98
[32]	ECG[4]	1	4486	2.53	0.6	135
[34]	ECG[5]	1	3390	2.67	0.63	184
L2SB	EEG/ECG.[6]	1	155	1.63	0.39	2494

note-1: 10-bit ECG+EEG, DPCM plus Golumb-Rice encoding, note-2: Slope Predictor+Huffman, note-3: ECG Mit BIH Data-set, predictive coder, plus 2-stage Huffman, note-4: 12-bit ECG, Adaptive predictor, 2-stage Huffman encoder, note-5: MIT BIH + Other Data-sets, Adaptive region-specific preditcion, plus Modified Variable-Length Coding, note-6: MIT BIH, BONN Data-set, XOR-Differential, PIPO Log2 Sub-band Variable Length Coding.

The message here is that L2SB can deliver useful data storage capacity compression outcomes at minimal power cost in some cases. However, neither of the algorithms examined can reduce flash memory data-write power enough to compensate for the additional cost of the associated data compression.

B. Compression-Transmission Trade-offs

A brief survey of data transmission approaches at data-rates up to the order of 1Mb/sec, and used in relevant work in the field, yields the data presented in Table XI, and showing a wide range of power consumptions.

Power tradeoffs for compression versus transmission power can be estimated on the following simplified basis: Given a particular sample rate, such as 10kSa/sec, with perhaps 12 bits per sample, then the minimum data-rate required to support the data transmission without compression must be 120 kbits/sec in this case. Assuming an average of 111nj/bit, then the data transmission power cost must be at least 13,200,000nJ, continuous power/energy cost. If compression reduces the bit rate by 30%, then the power consumption attributable to transmission would reduce by 30% also. In this case, 3,996,000nJ would be saved. If the compression cost was, for example, 1.3pJ per bit, then total compression power cost is found to be 156,000pJ, and the total power saving is actually 3,840,000pJ, or 3.8uW continuous operating power reduction.

Clearly, the critical factors here are (a) the data reduction/compression ratio, (b) the power consumed per bit compressed, and (c) the transmission power saved for each bit no longer needing to be transmitted. Taking several compression scenarios, including L2SB and Huffman, it is possible to compare systems in terms of power saved.

For this comparison, we assume several test-case scenarios, including those already introduced (notably DPCM-Huffman and XOR-L2SB), but also including other reported work where power data is readily interpreted for known data-rates and repeatable configurations. To give representative transmission cases, power figures are based upon literature in the field, and several chosen aggregate cases, as defined in Table XI.

For the the primary comparison with the low-power lossless encoder [21], separate static and dynamic power data is not

TABLE XI
SYSTEM POWER TRADEOFF - TRANSMISSION POWER VERSUS COMPRESSION.

Description	Data-Rate	nj/bit	Ref
Zigbee CC340	0.25Mb/s	296	[21]
Bluetooth BlueCore2	0.72Mb/s	143	[21]
Bluetooth T1 I BRF6150	1.00 Mb/s	75	[30]
Nordic RF transciever GFSK	1.00 Mb/s	21	[30]
ZL70250 ISM	0.10 Mb/s	20	[30]
PicoRadio, ultralow Data rates	1-10 Hz typ	1.25	[31]
Average-1 (excl Picoradio)		111	
Average-2 UWB/WIFI		21	

reported, and so cannot easily be translated to a nominal operating sample rate. However, this work reports a test case of a 24MHz operating frequency, and 170uW power consumption at an equivalent data-rate of 1226 x 12 bits and one channel. It is possible, therefore to align L2SB and Huffman operating conditions to this model. For this scenario, L2SB is configured to operate as a single channel PIPO compressor, clocked at 24MHz, with a sample data-rate of 1226 Sa/sec, thus allowing a power-gating duty cycle of approximately 19500:1 to be assumed. Huffman encoding also operates in a 1226 sample per second, single-channel look-up mode. This comparison results in the data presented in Table XII

The best-case band configuration is chosen for L2SB, with power as measured in the earlier described CMOS implementation.

Although Huffman produces a higher power saving for both Bluetooth and Zigbee test cases, correlating to its higher compression ratio, the L2SB model delivers power savings very similar to chua's low-power lossless compressor, in spite of having a lower compression rate. A lower compression rate acquired by much lower power cost, allows overall power saving to be comparable to a system with higher CR. Even Huffman encoding does not deliver a very substantial overall power saving as compared to L2SB, and when area cost is also considered, it is clear that L2SB offers a very desirable combination of overall power saving versus area cost invested at the chip layout level.

For more power-efficient transmission cases, L2SB becomes even more attractive, apparently outperforming Chua, and almost matching Huffman, and of course with much lower chip area. This is particularly clear for the 21nj/bit transmission cost scenario, where Huffman starts to enter a region where it begins to lose its power benefit, and the 'Chua' test case actually results in a negative power optimization (i.e. it consumes more power than it saves).

This last point is a very important observation: as transmission power cost is gradually improved by better design of transmitters, the proposed XOR-L2SB methodology has significant advantages to offer here. If future improvements in compression can be made, then L2SB may well be envisaged as a preferred method of achieving valuable system power gains in radio-linked systems.

TABLE XII
CHUA ET AL VS HUFFMAN AND L2SB, 24MHZ TEST CASE.

GENERAL DATA	Chua	Huffman	L2SB
Area um2	58000	529410	155
Compression Ratio	1.82	2.2	1.65
ZIGBEE 296 nj/bit			
Raw TX power nj/bit	296.00	296	296.00
TX power With CR, nJ/bit	162.94	134.55	179.03
Compression Cost nJ/bit	12.00	2.22	0.0081
Total Power with CR, nj/bit	174.94	136.76	179.04
Saving	40.9%	53.8%	39.5%
BLUETOOTH 143 njJbit			
Raw TX power nj/bit	143.00	143.00	143.00
TX power With CR, nJ/bit	78.72	65.00	86.49
Compression Cost nJ/bit	12.00	2.22	0.008
Total Power with CR, nJ/bit	90.72	67.22	86.50
Saving	36.6%	53.8%	39.5%
TxAVE-1, 110nJ/bit			
Raw TX power nJ/bit	111.00	111.00	111.00
TX power With CR, nJ/bit	61.10	50.45	67.14
Compression Cost nJ/bit	12.00	2.22	0.0081
Total Power with CR, nJ/bit	73.10	52.67	67.15
Saving	34.1%	52.5%	39.5%
TxAVE-2, 20nj/bit			
Raw TX power nj/bit	21.00	21.00	21.00
TX power With CR, nj/bit	11.56	9.55	12.70
Compression Cost nj/bit	12.00	2.22	0.0081
Total Power with CR, nj/bit	23.56	11.76	12.71
Saving	-12.2%	44.0%	39.5%

VI. SUMMARY AND CONCLUSIONS

A 65nm CMOS circuit was designed, fabricated and validated, comprising of a novel bio-physiological signal compression circuit with state-of-the art power-per-bit and gate-area cost. Employing an XOR bit-change detection scheme, and a 'log2-subband' data encoding scheme, allows an exceptionally simple circuit design. The specific configuration of Log2-Subband encoding is capable of being tuned to the characteristics of the data stream in question, and thus able to maximize available compression ratios under this algorithm. This paper demonstrates that careful choice of configuration can boost CR significantly for a given type of data-set. EEG and ECG, for instance, are very different in their dynamic content. Designs for dynamically configurable L2SB encoders have been envisaged, and could offer further compression improvements and flexibility. For example, bi-modal compression configurations might boost overall CR for an ECG where there are two distinct signal behaviours, or where an EEG records both seizure and non-seizure data. This would be achieved with relatively small power and area penalty, especially given that the design is already extremely lightweight in these terms.

Power comparisons with alternative schemes have been presented, though this proves difficult to do comprehensively, as most work in the field reports overall power rather than static and dynamic power, or reports power with compression as an indivisible part of a system. This makes extrapolation to relevant sample rates and normalized operating conditions difficult. Nonetheless, comparisons are made where possible, and L2SB is found to have favourable capabilities.

Where power-gating techniques are employed, it has been demonstrated that power consumption per sample bit could be of the order of pico-joules, with a 100:1 power-gating scheme delivering 1.2pj per bit power consumption for a 10kSa/s data-rate. Again, this is believed to be state-of-the art, and at the extreme low-end of what is envisaged to be possible at 65nm, or indeed where scaled to other process technologies. There is every reason to believe that as lower process nodes are targeted, data compression with sub-picojoule per-bit power cost would be readily achievable for bio-physiological signal measurement systems.

Using the proposed figure of merit (FOM) efficiency measures, comparisons can easily be made between L2SB and other systems. A limited survey and comparison was indeed provided in this paper (see Table X), and illustrates the value of a technology-neutral figure of merit.

The authors consider the presented circuit to be potentially one of the simplest possible encoders available, yet it delivers useful compression ratios in the context of flash-memory storage compression, and data transmission power drain. Although L2SB delivers modest compression ratios compared to a wide variety of other algorithms, those algorithms come at the cost of higher complexity, large gate-area costs, and higher power consumption. Because of this, their ability to trade-off compression versus transmission power is hindered in spite of their high CR values. Consequently, even though L2SB is inferior in compression ratio, it can be shown to deliver similar or superior performance for transmission power trade-offs in conjunction with extremely small silicon area. Indeed, in extreme resource-limited situations, such as printed organic thin film semiconductors, L2SB may even enable compression where other alternatives are simply not viable. This is a very interesting conclusion, in this case the concept of 'less is more' appears to be well observed.

ACKNOWLEDGMENT

The NOMAD 65nm CMOS Integrated Circuit was fabricated as part of the NOMAD project, funded by UK Gov. Innovate-UK Grant REF 26172-182148, and utilized the 65nm Faraday™ Standard Cell Library, fabricated on a UMC™ 65nm CMOS process.

The authors would like to thank the IMEC Europractice support team for their assistance in finalizing the 65nm tapeout submission and manufacture, and also to acknowledge the assistance of IC Mask Design Ltd, www.icmaskdesign.com, for their provision of additional tape-out support.

The MEG Data-sets were provided by York Instruments Ltd, www.york-instruments.com

REFERENCES

[1] K. Nair, J. Kulkarni, M. Warde, Z. Dave, V. Rawalgaonkar, G. Gore, and J. Joshi, "Optimizing power consumption in iot based wireless sensor networks using Bluetooth Low Energy", In IEEE Green Computing and Internet of Things (ICGCIoT), International Conference on (pp. 589-593), 2015.

[2] B. Kellogg, V. Talla, S. Gollakota, and J. R. Smith, "Passive Wi-Fi: Bringing Low Power to Wi-Fi Transmissions", In NSDI (Vol. 16, pp. 151-164), 2016.

[3] P. J. Soh, G. A. Vandenbosch, M. Mercuri, and D. M.-P. Schreurs, Wearable Wireless Health Monitoring: Current Developments, Challenges, and Future Trends, IEEE Microwave Magazine, vol. 16, no. 4, pp. 5570, 2015.

[4] Y. Takeda, K. Hayasaka, R. Shiwaku, K. Yokosawa, T. Shiba, M. Mamada, D. Kumaki, K. Fukuda and S. Tokito, "Fabrication of Ultra-Thin Printed Organic TFT CMOS Logic Circuits Optimized for Low-Voltage Wearable Sensor Applications", Scientific Reports, vol. 6, no. 1, 2016.

[5] M. Elsobky, Y. Mahsereci, Z. Yu, H. Richter, J. Burghartz, J. Keck, H. Klauk and U. Zschieschang, "Ultra-thin smart electronic skin based on hybrid system-in-foil concept combining three flexible electronics technologies", Electronics Letters, vol. 54, no. 6, pp. 338-340, 2018.

[6] N. Jayant, "Digital coding of speech waveforms: PCM, DPCM, and DM quantizers", Proceedings of the IEEE, vol. 62, no. 5, pp. 611-632, 1974.

[7] G. Higgins, S. Faul, R. P. McEvoy, B. McGinley, M. Glavin, W. P. Marnane, W. P., and E. Jones, "EEG compression using JPEG2000: How much loss is too much?", In IEEE Engineering in Medicine and Biology Society (EMBC), 2010 Annual Intl. Conference of the IEEE (pp. 614-617).

[8] P. T. Dao, X. J. Li and H. N. Do, "Lossy compression techniques for EEG signals", 2015 International Conference on Advanced Technologies for Communications (ATC), Ho Chi Minh City, pp. 154-159.

[9] G. Dufort, F. Favaro, F. Lecumberry, A. Martin, J. P. Oliver, J. Oreggioni, I. Ramirez, G. Seroussi, L. Steinfeld, "Wearable EEG via lossless compression", 2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), pp. 1995-1998.

[10] C. Deepu, X. Zhang, C. Heng and Y. Lian, "A 3-Lead ECG-on-Chip with QRS Detection and Lossless Compression for Wireless Sensors", IEEE Transactions on Circuits and Systems II: Express Briefs, vol. 63, no. 12, pp. 1151-1155, 2016.

[11] S. Chen and J. Wang, "VLSI implementation of low-power cost-efficient lossless ECG encoder design for wireless healthcare monitoring application", Electronics Letters, vol. 49, no. 2, pp. 91-93, 2013.

[12] L. Shaw, D. Rahman and A. Routray, "Highly Efficient Compression Algorithms for Multichannel EEG", IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 26, no. 5, pp. 957-968, 2018.

[13] M. A. Serhani, M. El Menshawy, A. Benharref and A. N. Navaz, "Real time EEG compression for energy-aware continous mobile monitoring", 27th International Conference on Microelectronics (ICM), Casablanca, 2015, pp. 291-294.

[14] G. Antoniol and P. Tonella, "EEG data compression techniques", IEEE Trans. on Biomedical Engineering, vol. 44, no. 2, pp. 105-114, 1997.

[15] Chen, S. L., and Wang, J. G. (2013). VLSI implementation of low-power cost-efficient lossless ECG encoder design for wireless healthcare monitoring application. Electronics Letters, 49(2), 91-93.

[16] A. Adebimpe, A. Aarabi, E. Bourel-Ponchel, M. Mahmoudzadeh and F. Wallois, "EEG resting state analysis of cortical sources in patients with benign epilepsy with centrotemporal spikes", NeuroImage: Clinical, vol. 9, pp. 275-282, 2015.

[17] A. Vyssotski, A. Serkov, P. Itskov, G. Dell'Omo, A. Latanov, D. Wolfer and H. Lipp, "Miniature Neurologgers for Flying Pigeons: Multichannel EEG and Action and Field Potentials in Combination With GPS Recording", Journal of Neurophysiology, vol. 95, no. 2, pp. 1263-1273, 2006.

[18] C. Bailey, J. Austin, G. Hollier, A. Moulds, M. Freeman, A. Fargus, and T. Lampert, "Evaluating a Miniature Multisensor Biosignal Recorder for Unsupervised Parkinson's Disease Monitoring", Sensors and Transducers Journal, 184(1), pp 66-76, 2015.

[19] Micron Technology Inc., "2, 4, 8Gb: x8/x16 Multiplexed NAND Flash Memory Features" datasheet, 2gb_nand_m29b__1.fm, Revision I 1/06 EN, 2004.

[20] Cypress Semiconductors Inc., "1-bit ECC, x8 and x16 I/O, 1.8V VCC SLC NAND Flash for Embedded", Document Number: 002-00330 Rev. *K, Revised April 20, 2017.

[21] E. Chua and Wai-Chi Fang, "Mixed bio-signal lossless data compressor for portable brain-heart monitoring systems", IEEE Transactions on Consumer Electronics, vol. 57, no. 1, pp. 267-273, 2011.

[22] R. Andrzejak, K. Lehnertz, F. Mormann, C. Rieke, P. David and C. Elger, "Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state", Physical Review E, vol. 64, no. 6, 2001.

[23] S. Wilton and N. Jouppi, "CACTI: an enhanced cache access and cycle time model", IEEE Journal of Solid-State Circuits, vol. 31, no. 5, pp. 677-688, 1996.

[24] A. H. Shoeb, "Application of machine learning to epileptic seizure onset detection and treatment", (Doctoral dissertation), Massachusetts Institute of Technology, 2009.

[25] G. Moody and R. Mark, "The impact of the MIT-BIH Arrhythmia Database", IEEE Engineering in Medicine and Biology Magazine, vol. 20, no. 3, pp. 45-50, 2001.
[26] A. Goldberger, L. Amaral, L. Glass, J. Hausdorff, P. Ivanov, R. Mark, J. Mietus, G. Moody, C. Peng and H. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet : Components of a New Research Resource for Complex Physiologic Signals", Circulation, vol. 101, no. 23, pp. e215-e220, 2000.
[27] M. F. Farid, A. A. Ghonem, and M. Dessouky, "An 8kb SRAM macro in 65nm for ultra-low voltage applications operating from 1.2 V to 0.5 V", In Electronics, Circuits, and Systems (ICECS), IEEE International Conference on (pp. 458-461). IEEE, 2015.
[28] T. Fukuda, K. Kohara, T. Dozaka, Y. Takeyama, T. Midorikawa, K. Hashimoto, and T. Hojo, "A 7ns-access-time 25W/MHz 128kb SRAM for low-power fast wake-up MCU in 65nm CMOS with 27fA/b retention current", In Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2014 IEEE International (pp. 236-237).
[29] J. Shah, D. Nairn and M. Sachdev, "A 32 kb Macro with 8T Soft Error Robust, SRAM Cell in 65-nm CMOS", IEEE Transactions on Nuclear Science, vol. 62, no. 3, pp. 1367-1374, 2015.
[30] W. Marnane, S. Faul, C. Bleakley, R. Conway, E. Jones, E. Popovici, and K. Patel, "Energy efficient on-sensor processing in body sensor networks", In Engineering in Medicine and Biology Society (EMBC), Annual Intl. Conference of the IEEE (pp. 2025-2029). IEEE, 2010.
[31] J. Rabaey, M. Ammer, J. da Silva, D. Patel and S. Roundy, "PicoRodio supports ad hoc ultra-low power wireless networking", Computer, vol. 33, no. 7, pp. 42-48, 2000.
[32] G.A. Luo, S. L. Chen, and T. L. Lin, "VLSI implementation of a lossless ECG encoder design with fuzzy decision and two-stage Huffman coding for wireless body sensor network", In Information, Communications and Signal Processing (ICICS), 9th Intl. Conference on (pp. 1-4). IEEE, 2013.
[33] Y. Wongsawat, S. Oraintara, T. Tanaka, and K. R. Rao, "Lossless multi-channel EEG compression", In IEEE International Symposium on Circuits and Systems (pp. 4-pp). IEEE, 2006.
[34] K. Li, F. Chen, Y. Pan, R. Huan and K. Cheng, "Real-time lossless ECG compression for low-power wearable medical devices based on adaptive region prediction", Electronics Letters, vol. 50, no. 25, pp. 1904-1906, 2014.
[35] C. Chen, S. Chen, H. Huang and C. Luo, "An Efficient Micro Control Unit with a Reconfigurable Filter Design for Wireless Body Sensor Networks (WBSNs)", Sensors, vol. 12, no. 12, pp. 16211-16227, 2012.
[36] X. C. Qu, and Y. Zhang, "A Compression Algorithm for ECG Data Using Variable-Length Classified Template Sets", In Computer, Consumer and Control (IS3C), Intl. Symposium on (pp. 856-859). IEEE, 2014.

Chris Bailey Dr. Chris Bailey, is a Senior Lecturer in Microelectronics and Computer Systems. He received a PhD in 1996, and joined the Advanced Computer Architectures Group, at the University of York in 1999, studying low power microelectronics and novel processor architectures. He is currently a member of The Healthcare and Well-being Research Theme, Department of Computer Science. He recently undertook an industrial secondment with Cybula Ltd. working on a 65nm CMOS ASIC Design.

Chengliang Dai Dr. Chengliang Dai is currently a post-doctoral research assistant in Data Science, with Imperial College, London. Previously, he studied for a PhD in the Advanced Computer Architectures Group, at The University of York, Awarded 2016, investigating novel low-cost bio-physiological signal compression algorithms. He subsequently held several post-doctoral positions in data analytics, including pollution sensing, and latterly Human brain MEG data analysis.

Jim Austin Prof. Jim Austin, is Professor of Neural Computing and has interests in massively parallel array processing, scalability, pattern matching architectures, and data analytics. In 2012, Professor Austin Received The Times Higher Education Award for the Outstanding Research Team of the Year for work on distributed air-craft maintenance data-analytics. He is also CEO of Cybula Ltd, a successful UK technology research and development consultancy.