

A Subthreshold PMOS Analog Cortex Decoder for the (8, 4, 4) Hamming Code

Jorge Pérez-Chamorro, Cyril Lahuec, Fabrice Seguin, Gérald Le Mestre, and Michel Jézéquel

This paper presents a method for decoding high minimal distance (d_{\min}) short codes, termed Cortex codes. These codes are systematic block codes of rate 1/2 and can have higher d_{\min} than turbo codes. Despite this characteristic, these codes have been impossible to decode with good performance because, to reach high d_{\min} , several encoding stages are connected through interleavers. This generates a large number of hidden variables and increases the complexity of the scheduling and initialization. However, the structure of the encoder is well suited for analog decoding. A proof-of-concept Cortex decoder for the (8, 4, 4) Hamming code is implemented in subthreshold 0.25- μm CMOS. It outperforms an equivalent LDPC-like decoder by 1 dB at BER= 10^{-5} and is 44 percent smaller and consumes 28 percent less energy per decoded bit.

Keywords: Cortex codes, analog decoding, sub-threshold CMOS circuit.

I. Introduction

Over the last fifteen years, many improvements have been made in error correcting techniques due to the invention of turbo codes by Berrou in 1993 and the rediscovery of low-density parity check (LDPC) codes by Mackay in 1995. Besides the usual digital implementation of turbo and LDPC decoders, some research has shown the feasibility of implementing them using analog networks to fully exploit the information available from the channel [1]-[4]. These theoretical works were followed by fully operational prototypes using either sub-threshold-biased CMOS [5]-[8] or forward-biased bipolar transistors [9], [10] to build the computing cells. The main advantage of such analog implementations is that the analog network does not require any scheduling, meaning that the network converges to a stable state corresponding to the code word. Moreover, the analog decoder power consumption is mostly static; therefore, it remains the same as the throughput increases. That is not the case in a digital decoder for which the power consumption is proportional to the clock frequency, that is, to the data throughput. Another characteristic of an analog decoder is the parallel data processing, which helps increasing the data rate. This is also a disadvantage as it implies a direct relationship between the frame length and the size of decoder. Analog decoding has been thus confined to a frame length of at most a few hundred bits, despite some architectural solutions to cope with longer frame lengths [4], [11]. However, for these frame lengths, it is well known that turbo and LDPC codes are not at their best [12]. There are many applications requiring such short codes and for which analog decoding can be of interest if codes other than turbo and LDPC are used, such as sensor networks and volatile

Manuscript received Apr. 5, 2009; revised June 3, 2009; accepted June 21, 2009.

This work was supported by the European Union and the Brittany region (France) in the context of the INTERACCES project.

Jorge Pérez-Chamorro (phone: +33 (0) 679088787, email: jorge.perezchamorro@ieee.org) was with the Electronics Department, TELECOM Bretagne, Brest, France, and is now with INVIA, Aix en Provence, France.

Cyril Lahuec (email: cyril.lahuec@telecom-bretagne.eu), Fabrice Seguin (email: fabrice.seguin@telecom-bretagne.eu), Gérald Le Mestre (email: gerald.lemestre@telecom-bretagne.eu), and Michel Jézéquel (email: michel.jezequel@telecom-bretagne.eu) are with the Electronics Department, TELECOM Bretagne, Brest, France.

doi:10.4218/etrij.09.0109.0207

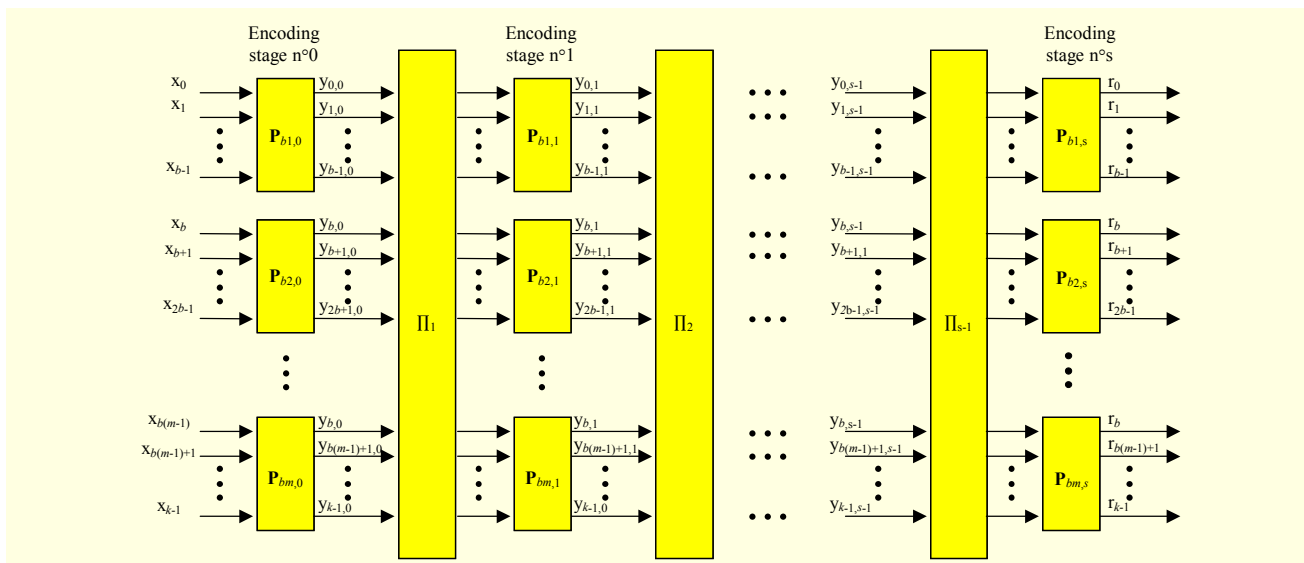


Fig. 1. General construction of Cortex codes: s encoding stages are used, each built with base codes having a length b . Different interleavers are used between two encoding stages.

memories. A potential candidate is the Cortex code family.

Cortex codes were invented by Carlach, from France Télécom, in 1999 [13] in an attempt to provide short codes with good minimal distances. The construction is multistage and uses a short base code to build the overall code. The base codes are interconnected through interleavers. Decoding has not been successful to date [14], [15], which has led to a drop in interest in such codes. This failure has been attributed partly to the fact that the resulting parity check matrix is not sparse and partly to the difficulty of correctly scheduling the complex digital network. Indeed, when decoding large frames, managing the large number of variables between decoding stages is not a simple matter.

An analog implementation seems a good choice because, once the input data is fed to the decoder, it converges naturally and continuously to a steady state corresponding to the most likely codeword.

This paper proposes a graph-based decoding structure which lends itself to an analog implementation. The background of the study is the (8, 4, 4) extended Hamming code because it is a simple and short code. A proof-of-concept Cortex analog decoder for this code is designed, tested, and compared to its equivalent LDPC-like decoder.

The remainder of this paper is organized as follows. Section II presents the construction of Cortex codes. Section III describes the decoder's structure along with the decoding algorithm used. Section IV deals with the design of the decoders using PMOS Gilbert multipliers. In section V, experimental results of a proof-of-concept Cortex decoder are shown and compared to an LDPC-like decoder. Finally, section VI concludes the paper.

II. Cortex Codes

1. General Construction

Cortex codes are built using the divide and conquer principle. The idea is to split the whole frame of length k into m sub-frames of length b , encode each sub-frame using a small base code, and then, after interleaving all of the resulting redundancy bits over the full length of the frame, to re-encode them using another encoding stage. This encoding process is repeated s times as shown in Fig. 1. The variables y_{ij} between each encoding stage are named hidden variables. The final encoding stage s outputs the redundancy bits r_{ij} which are concatenated to the initial frame to form the codeword. Hence, the code is of rate 1/2. The general form of the generator matrix is given as:

$$\mathbf{G}_{(n,k,d_{\min})} = [\mathbf{I}_k \mathbf{P}_0 \Pi_1 \mathbf{P}_1 \Pi_2 \cdots \Pi_{s-1} \mathbf{P}_s], \quad (1)$$

where \mathbf{I}_k is the identity matrix of length k , \mathbf{P}_i is the square matrix of size k representing the i -th encoding stage and is given by

$$\mathbf{P}_i = \begin{bmatrix} \mathbf{P}_{b1,i} & \mathbf{0}_{b \times b} & \cdots & \mathbf{0}_{b \times b} & \mathbf{0}_{b \times b} \\ \mathbf{0}_{b \times b} & \mathbf{P}_{b1,i} & \cdots & \mathbf{0}_{b \times b} & \mathbf{0}_{b \times b} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0}_{b \times b} & \mathbf{0}_{b \times b} & \cdots & \mathbf{P}_{b1,i} & \mathbf{0}_{b \times b} \\ \mathbf{0}_{b \times b} & \mathbf{0}_{b \times b} & \cdots & \mathbf{0}_{b \times b} & \mathbf{P}_{b1,i} \end{bmatrix}, \quad (2)$$

where each $\mathbf{P}_{b1,i}$ is the matrix representing a base code, and $\mathbf{0}_{b \times b}$ is a square null matrix of size b .

Although it is not mandatory, it is simpler to use the same encoding stage built from the same base code and the same

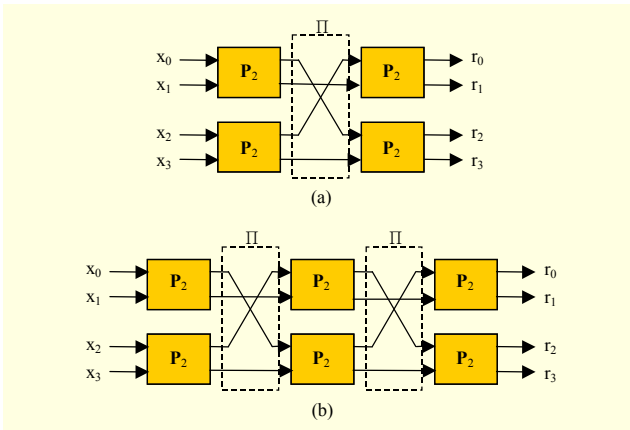


Fig. 2. Cortex (8, 4, x) Hamming code built using the (4, 2, 2) Hadamard code, denoted by P_2 , as a base code: (a) two encoding stages yield a $d_{\min}=3$ and (b) adding a third encoding stage and the same interleaver Π , increasing d_{\min} to 4.

interleaver throughout the encoding structure. Moreover, the base code is not necessarily a Cortex code itself. In [13], the same base code is used in the encoding stages, namely, the (8, 4, 4) Hamming code, but the interleavers are not all identical. The number of encoding stages used to build the code is directly related to the minimal distance. As shown in [16], when random interleavers are used and $s \rightarrow \infty$, the Cortex codes behave as random linear codes. Therefore, as s increases, d_{\min} increases too.

2. Building the (8, 4, x) Hamming Code from the (4, 2, 2) Hadamard Code

The (8, 4) Hamming code has a Cortex construction using the (4, 2, 2) Hadamard code as a base code, which is denoted by P_2 in Fig. 2. Its matrix P_2 is given by (3). Using two encoding stages made of two base Hadamard encoders separated by one interleaver as shown in Fig. 2(a) yields a minimal distance of three. This is easy to verify from the generator matrix $G_{(8,4,3)}$ given by (4), which is obtained from Fig. 2(a) and (1), (2), and (3). Adding an encoding stage identical to the first two and using the same interleaver for simplicity as shown in Fig. 2(b) increases d_{\min} to four. This is the maximum d_{\min} that can be obtained for this block length. The generator matrix $G_{(8,4,4)}$ is given by (5). Thus, the (8, 4, 4) Hamming code is obtained.

This illustrates, by a simple example, how adding encoding stages increases the minimal distance. Thus, the modular construction of Cortex codes allows larger codes to be built with good minimal distance, using either the Cortex Hamming codes or the (4, 2, 2) Hadamard code as base bricks. This paper focuses, however, only on the Cortex (8, 4, 4) Hamming code

to describe the proposed decoding method.

$$P_2 = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}, \quad (3)$$

$$G_{(8,4,3)} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}, \quad (4)$$

$$G_{(8,4,4)} = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \end{bmatrix}. \quad (5)$$

III. Cortex Decoder

The decoding structure resembles the encoding structure except that the edges are now bi-directional and the corresponding base code decoder replaces each base code encoder part. Due to the structure, digital decoding does not seem to be an appropriate method for this type of code. For instance, for the simple (8, 4, 4) Hamming digital decoder, considering that a decoding stage performs its task in a single clock cycle, three clock cycles are required to update the data on each side of the decoder. Because more than one iteration is required, usually six as in a digital turbo decoder, a total of eighteen (6×3) clock cycles are needed to decode one frame.

On the contrary, in an analog implementation, once the data has been fed to the decoder, it is able to output the codeword after a time period at most equal to the frame duration, and usually much less than that [8]-[10], that is, eight clock cycles. Thus, analog decoding avoids complex message-passing scheduling and should definitively offer a reduced latency compared to a digital solution as the size of the decoder increases. This further motivates us to use an analog solution to attempt to successfully decode Cortex codes.

1. Decoding Algorithm

Note that, from (5), the resulting parity check matrix is dense, in the sense that it contains many ones. This is usually considered to yield unsatisfactory decoding performance when a belief propagation algorithm, such as the sum-product algorithm (SPA) for which operands are probabilities, is used. A non-sparse parity-check matrix implies a large number of cycles, which renders probability computation intractable [17]. Nevertheless, the SPA algorithm is very simple to implement in analog circuits as the previous works showed, hence the motivation for using it.

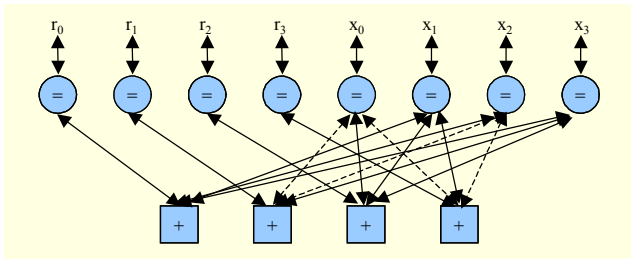


Fig. 3. (8, 4, 4) Hamming decoder using the SPA and based on the Tanner graph. One of the shortest cycles is shown with dashed lines.

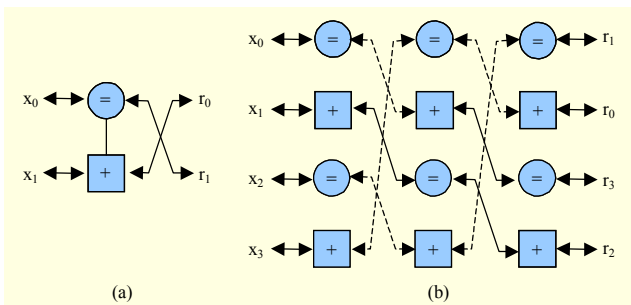


Fig. 4. Cortex decoder construction: (a) bipartite graph of (4, 2, 2) Hadamard code and (b) Cortex graph of the (8, 4, 4) Hamming code. One of the shortest cycles is shown with dashed lines.

2. Decoding Based on Tanner Graph

Codes are usually represented by means of graphs. Stemming directly from the generator matrix $G_{(8,4,4)}$, a decoder based on the Tanner graph when the SPA is used for the Cortex (8, 4, 4) Hamming code is represented in Fig. 3. First, note that the node's degree is not constant and is at most four. In fact, the nodes degree increases with the code length (if d_{\min} increases as well), which implies a higher complexity. Second, analyzing Fig. 3 shows that it contains 28 cycles and that the girth, that is, the length of the shortest cycle is four (one of them is represented with dashed lines in Fig. 3). Actually, there are four cycles at the girth. Because the girth is low and there are many short cycles, decoding is not optimal [17]. Because the parity check matrix is dense, decoding based on Tanner graph is termed LDPC-like, as it uses the same type of structure as that which is usually employed for LDPC decoders.

3. Decoding Based on Cortex Graph

Another representation is possible which takes into account the Cortex structure. Since the code is built based on the (4, 2, 2) Hadamard code, the (4, 2, 2) Hadamard decoder using the SPA is used as a base decoder. The bipartite graph representing the (4, 2, 2) Hadamard decoder is shown in Fig. 4(a). This version of the usual graph is simplified by noting

Table 1. Cycle and complexity comparison.

Decoder	Cortex	LDPC-like
Computing nodes	12	16
Maximum node degree	3	4
Bi-directional connections	8	12
Number of cycles	6	28
Girth	6	4
Number of girth cycles	2	4

that nodes of degree two perform no computation. Figure 4(b) shows the bipartite graph representing the Cortex (8, 4, 4) Hamming decoder built from the simplified (4, 2, 2) Hadamard graph. There are some interesting observations that can be made regarding the Cortex graph shown in that figure. First, all the nodes have the same degree. That is, the number of edges connecting them is always three. Any Cortex decoder built from the (4, 2, 2) Hadamard will have this property.

Intuitively, this is interesting in terms of complexity because longer codes do not imply a complexity increase of the computing nodes. Second, the number of cycles is low (only six) and the girth is six (one of them is represented as a dashed line in Fig. 4(b)). This relatively large value for such a small code is obtained because the base code graph is cycle-free. Having cycle-free base code graphs helps keep the large girth once such graphs are connected through interleavers. Otherwise, the girth of the overall graph would be limited to that of the base code's graph. Hence, decoding based on the proposed representation should yield better performance than decoding based on Tanner graph. Moreover, it should also have lower complexity. Both characteristics are summarized in Table 1.

4. Behavioral Simulation

Both (8, 4, 4) Hamming analog decoders are modeled using Simulink as described in [4]. The result of the behavioral simulation for an additive white Gaussian noise (AWGN) channel is shown in Fig. 5. For comparison, the maximum-likelihood decoding performance is also presented. The proposed Cortex decoding method is near optimal and outperforms LDPC-like decoding by 1 dB. This can be directly attributed to the comparatively high number of cycles and lower girth of the Tanner graph.

IV. Decoder Implementation

1. Design Guidelines

In [18], it was shown that an impaired transistor exponential

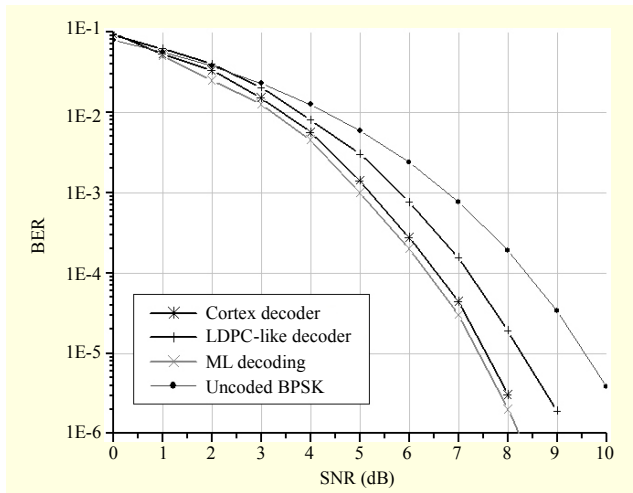


Fig. 5. BER performance comparison of analog decoders from behavioral simulation. Maximum-likelihood decoding and uncoded BPSK are shown for comparison.

I-V characteristic, used to translate log-likelihood ratios (LLRs) represented by voltages into probability represented by currents, degrades the decoder's performance. To avoid this, design guidelines were proposed in [19] for sub-threshold biased MOS. Only the main results are given here.

For weak-inversion, a better exponential MOS I-V characteristic is obtained if the inversion coefficient IC is much smaller than 1 and if the subthreshold slope factor m_s [20] is constant. A constant slope factor is obtained if $V_{BS} = 0$. This also reduces the dependence of the drain current on the drain-source voltage V_{DS} . The reduction of the threshold voltage V_T due to drain-induced barrier lowering and hence an increase of the drain current by electron injection from the source are limited if $V_{BS} = 0$. Finally, the exponential variations of the drain current are accentuated if either $V_{DS} < m_s \times U_T$ or $V_{BS} > 0$, where U_T is the thermal potential equal to 26 mV at 300K. Fixing by design $V_{DS} \geq 4U_T$ and $V_{BS} = 0$ avoids such undesired effects. However, $V_{BS} = 0$ can be only guaranteed if PMOS transistors are used because they all have an individual substrate formed by the N-well. Because the source can be physically tied to the bulk, $V_{BS} = 0$.

2. Computing Nodes

Unlike other CMOS analog decoders [5]-[8] which require a circuit to convert LLR from the channel into probability and a second circuit to multiply probabilities, the proposed decoder uses a PMOS-based Gilbert cell to perform both at the same time as in BJT-based decoders [9], [10]. This simplifies the overall circuit and thus reduces the power consumption. To further reduce the V_{DS} variations, the cells have a symmetrical design. Dummy transistors are added so that there is the same

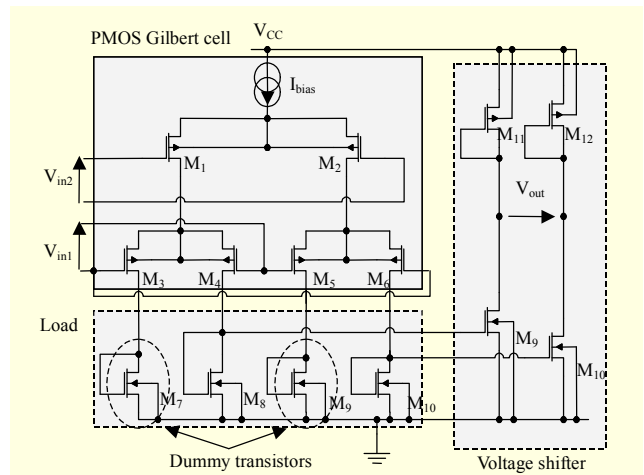


Fig. 6. PMOS-based Soft-EQU computing node with dummy load added. Note the physical connection between sources and bulk for each PMOS to insure $V_{BS} = 0$. The input differential voltages V_{in1} and V_{in2} and the output differential voltage V_{out} are proportional to LLRs ($V_{cc} = 1.4$ V, $I_{bias} = 1$ μ A).

number of MOSs between the power-rails in each branch. As an example, the soft-equality node is given in Fig. 6. Based on these design guidelines, the aspect ratio of the PMOS can be calculated as

$$\frac{W}{L} = \frac{I_{bias}}{2IC\mu_0 C_{ox} U_T^2}, \quad (6)$$

where I_{bias} is the bias current equal to 1 μ A, μ_0 is the carrier mobility, and C_{ox} is the gate capacitance per unit area. We set the IC to 0.05. At less than 0.1, good exponential behavior is obtained, and above 0.01, the transistor aspect ratio of the Gilbert cell is reduced. The latter directly affects the maximum speed at which the decoder runs. Therefore, we chose to use minimum length transistors, which yield $W/L=9/0.25$. The benefits of using PMOS rather than NMOS to build the computing cells come at the expense of a size increase. The aspect ratio of the transistors, 36, is larger than that used in other reported NMOS decoders, such as 10 in [8], due to a lower hole mobility (about three times) than that of an electron. The room taken by the N-well should be added, which further increases the area of the cell to about five times the area of an equivalent NMOS computing cell with the same IC and the same I_{bias} . The load NMOSs are $W/L=5/0.25$. There are three such Gilbert multipliers per node of degree three and six per node of degree four.

3. Decoders Block Diagram

The block diagram of both the Cortex and the LDPC-like decoder are shown in Fig. 7. The only part that changes is the

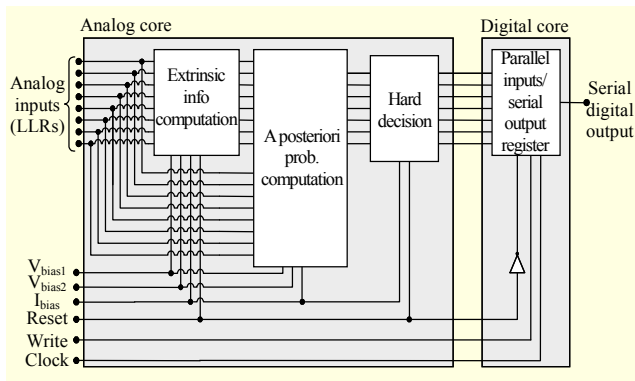


Fig. 7. Decoder block diagram. The difference between the LDPC-like and Cortex decoders is the extrinsic block, which corresponds to Fig. 3 and Fig. 4(b), respectively.

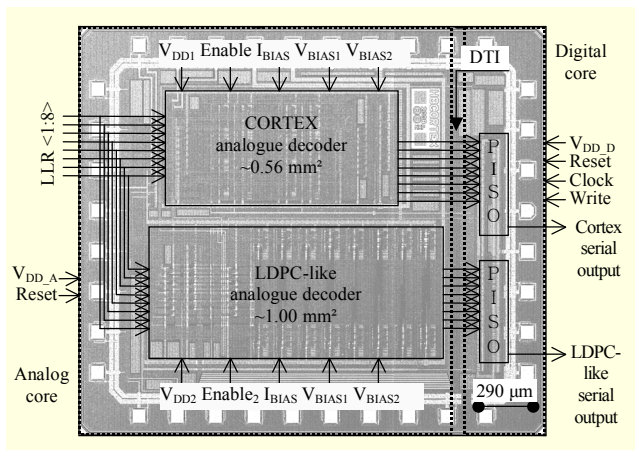


Fig. 8. Die microphotograph of the integrated circuit and floor plan.

extrinsic information block, which corresponds to the graphs shown in Fig. 3 and in Fig. 4(b) for the LDPC-like decoder and Cortex decoder, respectively. Considering the low number of analog inputs, we fed them in parallel to the decoders. Besides the clock, reset and write signals were added to synchronize the circuit with the outside test bench.

V. Experimental Results and Discussion

Both decoders were implemented on a chip. A microphotograph of the integrated circuit is given in Fig. 8 along with the floor plan. The total silicon area is $2.17 \text{ mm} \times 2.6 \text{ mm}$. Although they use the same computing cell, due to the higher node degree and larger number of interconnections, the LDPC-like decoder is almost twice as large as the Cortex decoder. To enable repeatable and reliable tests, random frames were generated, encoded, modulated, and altered by an AWGN channel and then demodulated on an FPGA. The resulting digital signals corresponding to the noisy LLRs were

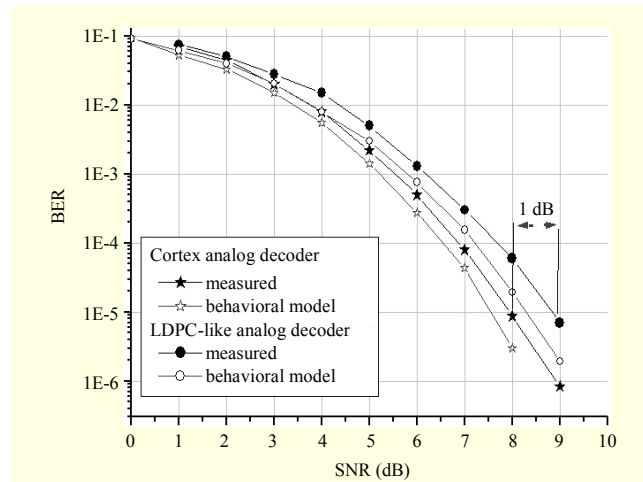


Fig. 9. Measured BER for the Cortex (black stars) and the LDPC-like (black circles) analog decoders. Behavioral simulation results are given for comparison.

transferred onto a set of synchronized DACs to provide the eight analog inputs to the decoders.

The BER results are shown in Fig. 9 for the Cortex and LDPC-like decoders. As expected from the behavioral simulations, the Cortex decoder outperformed its LDPC-like counterpart by 1 dB at $\text{BER}=10^{-5}$. Table 2 summarizes the main features of some existing analog decoders and compares them to the two proposed decoders. Due to the lower complexity of the proposed decoders, their energy per decoded bit is two orders of magnitude lower than that of the sub-threshold analog decoders [6]-[8] using the SPA. The Tanner decoder for the (8, 4, 4) Hamming code (an LDPC-like decoder) presented in [5] performs better than the Cortex decoder in terms of energy consumption but is 1 dB worse than the Cortex decoder at a $\text{BER} = 10^{-6}$. Besides a smaller minimum size feature, $0.18\text{-}\mu\text{m}$ instead of $0.25\text{-}\mu\text{m}$, the difference in terms of size is explained by the fact that a PMOS computing cell is five times larger than an NMOS one.

Unlike all the other decoders in Table 2, the decoder implemented in [21] does not use the SPA but the min-sum (MS) algorithm. NMOS and PMOS current mirrors are used to implement the algorithm instead of multipliers in the other decoders. Although it works in strong inversion, it is still slower than BJT decoders. Compared to the Cortex decoder, the MS LDPC is larger, given that it uses a smaller minimum length technology ($0.18\text{-}\mu\text{m}$) than the one used for the Cortex decoder, while consuming five times more energy per decoded bit. However, its data throughput is three times higher thanks to the MOS operating in the saturation region.

It is striking that the decoder with the second lowest energy consumption per bit is the BJT-based decoder from [9]. This is due to a high throughput as explained in the introduction. This

Table 2. Comparison with other reported analog decoders (measured data).

Published	Decoder type	Technology	Topology	Frame size (inf. bits)	Throughput (Mbps)	Core area (mm ²)	Energy per bit (nJ)
[5]	LPDC-like	CMOS 0.18 μm	NMOS	4	3.7	0.027	0.022
[6]	Turbo	CMOS 0.35 μm	NMOS	16	13	1.4	13.9
[7]	Tailbiting BCJR	CMOS 0.5 μm	NMOS	4	0.4	0.85	9.5
[8]	3GPP Turbo	CMOS 0.35 μm	NMOS	40	2	4.1	12.6
[9]	Tailbiting BCJR	BiCMOS 0.25 μm	BJT	8	320	1.19	0.063
[10]	Tailbiting BCJR	BiCMOS 0.25 μm	BJT	48	120	4.3	4
[21]	LDPC (MS)	CMOS 0.18 μm	CMOS	8	6	0.57	0.83
This work	LDPC-like	CMOS 0.25 μm	PMOS	4	2	1	0.222
This work	Cortex	CMOS 0.25 μm	PMOS	4	2	0.56	0.160

shows that the BJT-based analog decoder should not be discarded when low energy consumption is required. In that respect, the decoder presented in [10] also achieves good performance when compared to most of the subthreshold CMOS decoders.

Comparing the two proposed decoders shows that, as expected, the Cortex decoder consumes 28 percent less energy per decoded bit than the LDPC-like decoder. The discrepancies between behavioral simulations and actual measurements are about 0.3 dB, which is consistent with the mismatch described in [8].

VI. Conclusion

A modular method to easily decode Cortex codes has been proposed. The decoding algorithm applied is the sum-product algorithm. An LDPC-like decoding method based on the Tanner graph, is outperformed by the Cortex decoder in terms of complexity (44 percent smaller), power consumption (28 percent less), and decoding performance (1 dB at BER=10⁻⁵). The first working prototype confirmed all these. Fabricated in a CMOS 0.25 μm technology, these decoders are also the first reported decoders that use PMOS-based computing cells. Further works are required to build useful codes using the proposed method.

References

- [1] J. Hagenauer and M. Winklhofer, "The Analog Decoder," *Proc. IEEE Int. Symp. Information Theory*, Cambridge, USA, 16-21 Aug. 1998, p. 145.
- [2] H.-A. Loeliger et al., "Probability Propagation and Decoding in Analog VLSI," *Proc. IEEE Int. Symp. Information Theory*, Cambridge, USA, 16-21 Aug. 1998, p. 146.
- [3] F. Lustenberger, "On the Design of Analog Iterative VLSI Decoders," doctoral dissertation, ETH Zurich, No 13879, Hartung-Gorre, Konstanz, Series in Signal and Information Processing, vol. 2, ISBN 3-89649-622-0, ISSN 1616-671X, Nov. 2000.
- [4] M. Arzel et al., "Semi-iterative Analog Turbo Decoding," *IEEE Trans. Circuits and Systems I: Regular Papers*, vol. 54, no. 6, June 2007, pp. 1305-1316.
- [5] C. Winstead et al., "A 0.8 V CMOS Analog Decoder for an (8, 4, 4) Extended Hamming Code," *IEEE Trans. Circuits and Systems I: Regular Papers*, vol. 53, no. 4, Apr. 2006, pp. 829-841.
- [6] V. Gaudet and G. Gulak, "A 13.3 Mbps 0.35 μm CMOS Analog Turbo Decoder IC with a Configurable Interleaver," *IEEE J. Solid-State Circuits*, vol. 38, no. 11, Nov. 2003, pp. 2010-2015.
- [7] C. Winstead et al., "CMOS Analog MAP Decoder for (8, 4) Hamming Code," *IEEE J. Solid-State Circuits*, vol. 39, no. 1, Jan. 2004, pp. 122-131.
- [8] D. Vogrig et al., "A 0.35- μm CMOS Analog Turbo Decoder for the 40-Bit Rate 1/3 UMTS Channel Code," *IEEE J. Solid-State Circuits*, vol. 40, no. 3, Mar. 2005, pp. 753-762.
- [9] M. Moerz et al., "An Analog 0.25 μm BiCMOS Tailbiting MAP Decoder," *Proc. IEEE Int. Solid-State Circuits Conference*, Feb. 2000, pp. 356-357.
- [10] C. Lahuec et al., "Design and Test of a 0.25- μm BiCMOS Double-Binary Analogue APP Decoder," *Proc. 5th Analog Decoding Workshop*, June 2006, pp. 35-38.
- [11] A. Schaefer et al., "Analog Rotating Ring Decoder for an LDPC Convolutional Code," *Proc. IEEE Information Theory Workshop*, Paris, France, 31 Mar.-4 Apr. 2003, pp. 226-229.
- [12] D. MacKay, "Good Error-Correcting Codes Based on Very Sparse Matrices," *IEEE Trans. Information Theory*, vol. 45, no. 2, Mar. 1999, pp. 399-431.
- [13] J. Carlach, A. Otmani, and C. Vervoux, "A New Scheme for Building Good Self-Dual Block Codes," *Proc. IEEE Int. Symp. Information Theory*, Sorrento, Italy, 25-30 June 2000, p. 476.

- [14] A. Otmani, "Codes Cortex et construction de codes auto-duaux optimaux," doctoral dissertation, Université de Limoges, France, Oct. 2002.
- [15] E. Cadic, "Construction de turbo codes cours possédant de bonnes propriétés de distances minimales," doctoral dissertation, Université de Limoges, France, Oct. 2003.
- [16] G. Olocco and J. Tillich, "A Family of Self-Dual Codes which Behave in Many Respects Like Random Linear Codes of Rate 1/2," *Proc. IEEE Int. Symp. Information Theory*, Washington DC, USA, 24-29, June 2001, p. 15.
- [17] D.J.C. MacKay, "Good Error-Correcting Codes Based on Very Sparse Matrices," *IEEE Trans. Information Theory*, vol. 45, no. 2, Mar. 1999, pp. 399-431.
- [18] N. Duchaux et al., "Effect of BJT's Parasitics on Computing Cells for Analog Decoders," *Proc. Joint Int. IEEE Northeast Workshop on Circuits and Systems and TAISA Conference*, Montréal, Canada, June 2008, pp. 301-304.
- [19] J. Pérez-Chamorro et al., "Design Rules for Subthreshold MOS Circuits," *Proc. 5th Analogue Decoding Workshop*, Torino, Italy, June 2006, pp. 23-26.
- [20] R. van Langevelde, A. Sholten, and D. Klaasen, "Physical Model of MOS Model 11," Koninklijke Philips Electronics N.V., unclassified report 2003/00239, Tech. Rep., 2003.
- [21] S. Hemati, A.H. Banihashemi, and C. Plett, "A 0.18 μm MOS Analog Min-Sum Iterative Decoder for a (32, 8) Low-Density Parity-Check (LDPC) Code," *IEEE J. Solid-State Circuits*, vol. 41, no. 11, Nov. 2006, pp. 2531-2540.



Jorge Pérez-Chamorro received the degree of electronics engineer from the Universidad de los Andes, Bogota, Colombia, in 2005. During his doctoral studies, he focused on channel coding and decoding and on analog circuits design. He was an invited researcher at the High Capacity Digital Communications Laboratory of the ECE Department of the University of Alberta, Edmonton, Canada, in 2007. He received his PhD (with highest honours) from TELECOM Bretagne, Brest, France, in 2009. Afterwards, he joined INVIA, a start-up company based in Aix en Provence, France, specializing in the design of IPs for secure platforms. His research interests are error correcting codes, bipartite graphs, decoding algorithms, and code compression.



Cyril Lahuec received the BSc (Hons) degree from the University of Central Lancashire, UK, in 1993, and his MEng and PhD degrees from Cork Institute of Technology, Ireland, in 1999 and 2002, respectively. He was with Parthus Technologies (now Ceva), Cork for his PhD work and then as a consultant. He joined the Electronics Department of TELECOM Bretagne as a full-time lecturer in 2002. His research interests are in frequency synthesis, analog IC design, channel decoding, and biomedical applications.



Fabrice Seguin received the PhD from the Université Bordeaux 1, France, in 2001. His doctoral research concerned the current mode design of high-speed current conveyors and applications in RF circuits. In 2002, he joined the Electronic Engineering Department of TELECOM Bretagne as a full-time lecturer. He is currently involved with design issues of analog channel decoders and related topics.



Gérald Le Mestre received the Master's Degree from ENSSAT Lannion, France. During his Master's degree study, he spent his final year at the University Laval, Quebec, Canada, in 2003. He joined the Electronics Department of TELECOM Bretagne as a full-time research engineer in 2005, where he is in charge of prototyping of FPGAs, testing analog, and digital systems.



Michel Jézéquel received the degree of "Ingénieur" in electronics from the École Nationale Supérieure de l'Électronique et de ses Applications, Paris, France, in 1982. From 1983 to 1986 he was a design engineer at CIT ALCATEL in Lannion, France. Then, after experience with a small company, he took a one-year course on software design. In 1988, he joined the École Nationale Supérieure des Télécommunications de Bretagne where he is currently a professor and head of the Electronics Department. His main research interest is circuit design for digital communications. He focuses his activities in the fields of Turbo codes, adaptation of the turbo principle to iterative correction of intersymbol interference, the design of interleavers, and the interaction between modulation and error correcting codes.