



A low-complexity soft-decision decoding architecture for the binary extended Golay code

Patrick Adde, Raphaël Le Bidan

► To cite this version:

Patrick Adde, Raphaël Le Bidan. A low-complexity soft-decision decoding architecture for the binary extended Golay code. ICECS 2012: IEEE International Conference on Electronics, Circuits, and Systems, Dec 2012, Sevilla, Espagne. pp.705 - 708. hal-00797565

HAL Id: hal-00797565

<https://hal.science/hal-00797565v1>

Submitted on 6 Mar 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A low-complexity soft-decision decoding architecture for the binary extended Golay code

Patrick ADDE et Raphael LE BIDAN

* Institut Mines-Télécom/Télécom-Bretagne; CNRS Lab-STICC UMR 3192
Technopôle Brest Iroise, CS 83818 29238 Brest Cedex 3
Université Européenne de Bretagne, France
email: firstname.surname@telecom-bretagne.eu

Abstract — The (24, 12, 8) extended binary Golay code is a well-known rate-1/2 short block-length linear error-correcting code with remarkable properties. This paper investigates the design of an efficient low-complexity soft-decision decoding architecture for this code. A dedicated algorithm is introduced that takes advantage of the code's properties to simplify the decoding process. Simulation results show that the proposed algorithm achieves close to maximum-likelihood performance with low computational cost. The decoder architecture is described, and VLSI synthesis results are presented.

I. INTRODUCTION

Forward Error Correction has become an important practical mean for improving the bit error rate (BER) performance of digital communication and storage systems. The (23,12,7) binary Golay code is a perfect binary triple-error-correcting code introduced in 1949 [1] with remarkable mathematical properties. The addition of an overall parity-check bit yields the rate-1/2, self-dual (24,12,8) extended binary Golay code which has found numerous practical applications either as a standalone code (for example on the 1977 Voyager spacecraft mission [2]) or as an inner code in concatenated coding systems [3].

A number of hard-decision algebraic decoding algorithms have been investigated over the years (see e.g. [2],[4],[5]). In contrast to hard-decision decoders which operate on binary values, soft-decision decoders directly process unquantized (or quantized on more than two levels in practice) samples at the output of the matched filter, thereby avoiding the loss of information. Over the Additive White Gaussian Noise (AWGN) channel, soft-decision decoding may offer up to 3 dB coding gain over hard-decision decoding, but at the cost of increased computational complexity. Soft-decision decoding algorithms for the extended Golay code have also received a lot of attention (see [6] and references therein, or [7][8] for more recent results). Yet very few decoder architectures have been published ([6] is a notable exception). This paper addresses the challenging issue of designing a low-complexity (less than 5,000 gates) soft-decision decoder architecture with near-optimal performance for the (24,12,8) code.

The proposed approach is based on the decoding

algorithm introduced in [9][10]. This algorithm is dedicated to rate-1/2 linear codes having a generator matrix composed of an invertible sub-matrix for the redundancy part. A double re-encoding process inspired by Chase's algorithm [11] is used to create a list of candidate codewords among which the most likely is retained as the decoder decision. A few number of error patterns were shown to be usually sufficient to achieve close to maximum-likelihood (ML) performance for short block codes.

The remainder of the paper is organized as follows. Section II discusses different approaches to encode the (24,12) Golay code. Section III introduces the principle and performance of the proposed soft-decision decoding algorithm. The design of an hardware-efficient decoder architecture with near-ML performance is developed in Section IV. Conclusions follow in Section V.

II. ENCODING THE (24,12,8) GOLAY CODE

Since the proposed soft-decision decoder performs several re-encoding of the received data sequence, we first review different ways to encode the Golay code.

The binary (23,12,7) Golay code can be described in cyclic form as a quadratic residue code with generator polynomial $g(x)=x^{11}+x^9+x^7+x^6+x^5+x+1$ [4]. Thus a 11-stage shift-register followed by an accumulator can be used to perform systematic encoding of the (24,12,8) extended Golay code in 24 clock periods.

Another approach directly implements with logic gates the product of the binary data vector $\mathbf{d}$ with the generator matrix $\mathbf{G}$ of the code. Since the extended Golay code is a self-dual code, the generator matrix $\mathbf{G}_d$ in canonical form can be written as

$$\mathbf{G}_d = [\mathbf{I}_{12}, \mathbf{P}]$$

where $\mathbf{I}_{12}$ is the 12×12 identity matrix corresponding to the 12 information bits $\mathbf{d}$, and where $\mathbf{P}$ is a 12×12 invertible binary matrix that generates the 12 parity-check bits $\mathbf{p}$. The corresponding codeword $\mathbf{c}$ then reads $\mathbf{c} = (\mathbf{d}, \mathbf{p})$. From the self-dual property of the extended Golay code, $\mathbf{P}$ satisfies the property $\mathbf{P}^{-1} = \mathbf{P}^t$. Thus, in just the same way that the 12 information coordinate $\mathbf{d}$ are used to compute the parity bits $\mathbf{p}$ using the generator matrix $\mathbf{G}_s=[\mathbf{I}_{12}, \mathbf{P}]$, the 12 parity

coordinates $\mathbf{p}$ may also be encoded using the alternative generator matrix $\mathbf{G}_p = \mathbf{P}^{-1} \times \mathbf{G}_d = [\mathbf{P}^t, \mathbf{I}_{12}]$ to obtain the information vector $\mathbf{d}$.

A third method uses the Cortex construction. Cortex codes are a family of rate-1/2 self-dual linear block codes first introduced in [12]. As shown in Fig. 1, they combine a very short mother code E with a sequence of permutations to produce the parity bits. If the mother code is self-dual, the resulting Cortex code inherits from the self-dual property [13].

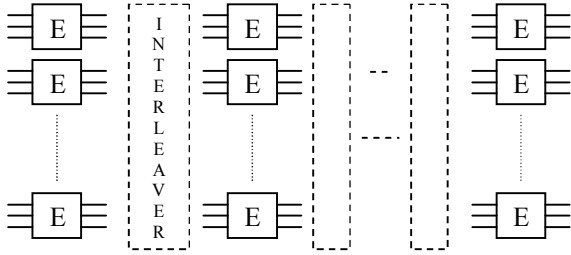


Fig. 1: General Cortex encoding scheme build from elementary code E

The Cortex structure corresponding to the extended Golay code is shown in Fig. 2. It is based on the (8,4,4) extended Hamming code (denoted H). The 12 information bits $\mathbf{d}$ are divided into 3 blocks of 4 bits. Each block is encoded by the (8,4,4) code to produce 4 parity bits (systematic bits are discarded). The sequence of 12 parity bits is then shuffled by a suitable permutation function, and the whole process is repeated 3 times in order to generate the parity bits $\mathbf{p}$. The codeword is finally obtained by concatenating the 12 systematic bits $\mathbf{d}$ with the 12 final parity bits $\mathbf{p}$.

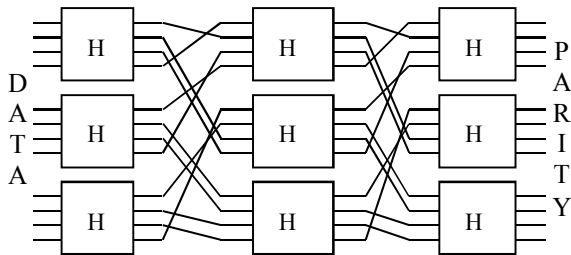


Fig. 2: Cortex Architecture for the extended Golay Code

Interestingly, the (8,4,4) extended Hamming code used as a building block for obtaining the extended Golay code in Cortex form may also be described as a Cortex code. The corresponding Cortex architecture is shown in Fig. 3 [12]. It is based on the (4,2,2) Hadamard code (Had).

In this paper, we choosed to implement the Cortex architecture for encoding the (24,12,8) Golay code.

III. NEAR MAXIMUM-LIKELIHOOD SOFT-DECISION DECODING OF THE GOLAY CODE

ML soft-decision decoding is known to offer the best decoding performance but is usually computa-

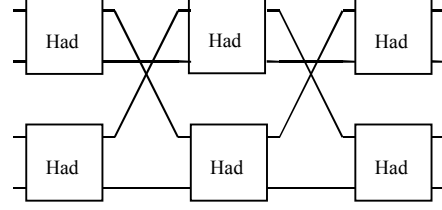


Fig. 3: Cortex encoding architecture for the (8,4,4) extended Hamming code

-tionally intractable for most codes of practical interest. Brute-force ML decoding of the extended Golay code requires correlating the received word with each of the $2^{12} = 4096$ candidate codewords, which is computationally feasible yet intensive. A smarter approach would be to apply a variant of the Viterbi algorithm to the 12-sections 16-state tail-biting trellis representation of the Golay code introduced in [7]. However, in spite of its apparent simplicity, this approach is not the most attractive one when low-complexity (low gate count) and very high data rate are sought. For these reasons, we have chosen to focus rather on the simple and efficient algorithm introduced in [10]. This algorithm can be applied to any self-dual codes, and is particularly attractive for short codes for which it offers near-ML performance at low decoding complexity.

The algorithm operates as follows. A list of VT_s candidate codewords is obtained by applying binary test patterns to the k message bits obtained by taking a hard-decision on the received information sequence, and then re-encoding the resulting candidate sequence. As suggested by Chase in [11], the VT_s test patterns attempt to correct the most likely errors patterns confined in the least reliable positions in the received information sequence. The same procedure is applied in parallel to the k parity bits, by inverting the encoding equations in order to re-compute the k message bits from the k parity bits. This produces a second list of VT_p candidate codewords. The decoder finally selects the candidate codeword at minimum Euclidean distance (maximum correlation metric) from the received word.

Bit Error Rate (BER) performance vs. Signal to Noise Ratio (SNR) of this algorithm for the (24,12,8) extended Golay code using 8-bit quantization is presented in Fig. 4 for different numbers $VT = VT_s + VT_p$ of test patterns. Binary Phase-Shift-Keying (BPSK) transmission over AWGN is assumed. For comparison purpose, ML performance of this code is also shown. We observe that 32 (16+16) test patterns are sufficient to obtain close to ML performance (within 0.1 to 0.2 dB) in the simulated BER range. The optimized architecture introduced in the next section uses a total of 48 (24+24) test patterns, thereby virtually achieving ML performance. These

results demonstrate the attractive trade-off between performance and complexity provided by this algorithm.

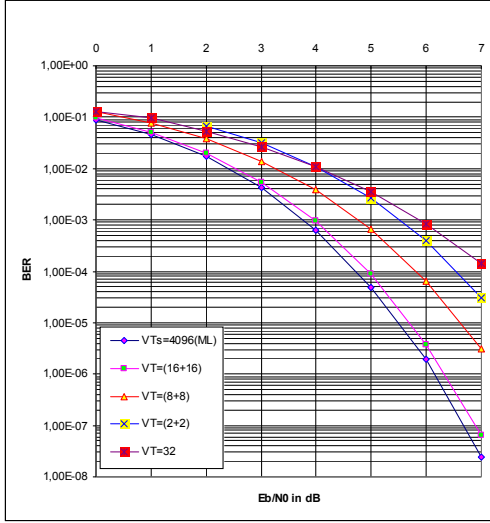


Fig. 4: BER performance of the considered soft-decision decoder for the (24,12,8) code

The influence of input quantization on the BER performance is studied in Fig. 5. We observe that $q=3$ bits (sign-bit excluded) are sufficient to obtain near-ML performance (within 0.3 dB at most) with the low-complexity algorithm based on 48 error patterns.

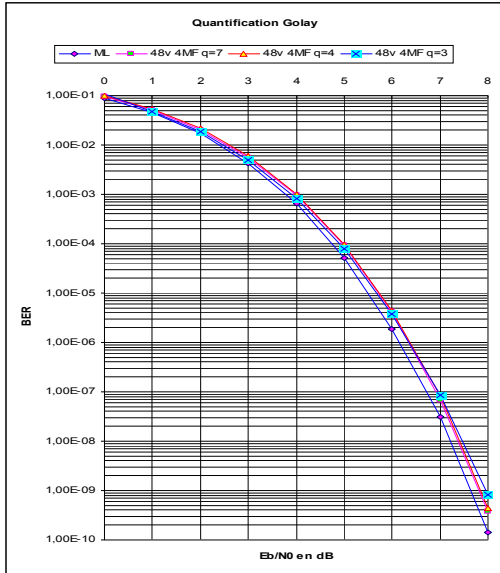


Fig. 5: Performance of the considered soft-decision decoding algorithm vs. the number q of bits used for quantization (sign bit excluded)

IV. SOFT-DECISION DECODER ARCHITECTURE

In spite of its short block length, designing low-complexity soft-decision decoding architectures for the Golay code that are amenable to very high data

rate remains a challenging issue. Here, we describe a digital implementation tailored to the soft-decision decoding algorithm investigated in the previous Section. The decoder operates on soft inputs quantized on $q=3$ bits (+ sign bit). A total of $VT = VT_s + VT_p = 24 + 24 = 48$ candidate codewords is used to generate the decoder decision. The 2×24 error patterns are chosen so as to correct the most likely errors located in the $L_{rs} = L_{rp} = 5$ least reliable positions in both the information and parity parts of the received vector. The corresponding architecture is inspired by [14] and is shown in Fig. 6. It consists of four main blocks: reception, processing, transmission and control.

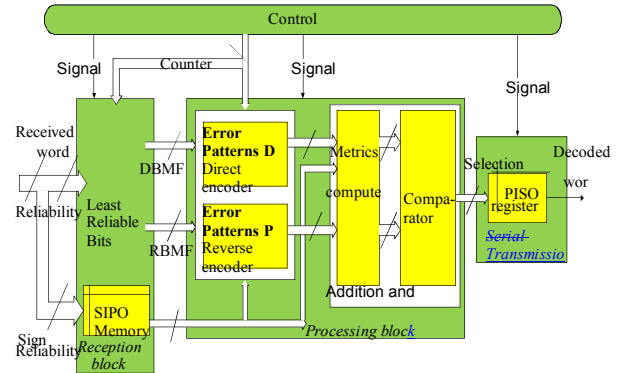


Fig. 6: Proposed soft-decision decoding architecture

Inside the reception block, the $n=24$ soft symbols of the received word are processed sequentially. This block first identifies successively the L_{rs} and L_{rp} least reliable positions within the systematic and parity parts of the received word, respectively. In parallel, a serial-in parallel-out (SIPO) shift-register memorizes sequentially the 24 soft samples of the received word.

The processing block comprises three main tasks. First, error patterns are generated from the sign bits of the received word by testing different combinations of 0s and 1s in the least reliable bit positions. Then, these error patterns are added (modulo-2) to the information/parity sequence and the resulting sequence is re-encoded to produce a codeword which is scored (correlation metric). Finally, a selection function (comparator) identifies the most likely codeword within the input list of 48 candidate codewords. Note that this process is realized in parallel for the information and the parity parts of the received word. Moreover, the metric of each candidate codeword is computed on-the-fly from the 24 soft symbols provided by the SIPO shift register,

Finally, the transmission block is composed solely of a parallel-in serial-out (PISO) shift register, used to deliver sequentially the decoded message (systematic bits of the decoder decision) at the decoder output.

The three previous blocks are supervised by a

control block. In our design, this task is realized by a 5-bit counter that generates the required control signals.

As shown in Fig. 6, the soft decoder architecture is structured in two pipelined stages: reception and processing transmission. The first stage sequentially processes the 24 received soft symbols in 24 clock periods. In the second stage, the 24+24 candidate codewords are generated, scored and compared in a total of 24 clock periods, thanks to parallel processing. Finally, the 12 decoded information bits are sequentially delivered in 12 clock periods. The decoder latency L depends on the number of pipeline stages and also on the codeword length n . For the proposed decoder architecture, the resulting latency is $L=2n=48$ clock symbols.

Logic synthesis has been performed using the Synopsys tool in order to estimate the hardware complexity of the proposed decoder architecture. A STMicroelectronics 0.09 μm CMOS process ASIC target was considered. The soft decoder is clocked at $f=432\text{MHz}$. The results are presented in Table 1, which gives the complexity (measured in terms of equivalent 2-input NAND gate) for each function of the soft decoder. We observe that the proposed soft-decision decoding architecture requires about 4K equivalent gates. In this implementation the maximum data rate is 432 Mb/s. Thus the proposed soft decoder is less complex than the fast parallel hard-decision permutation decoder at the core of the soft-decision decoder architecture described in [6], which achieved a data rate of 500 Mb/s, but with 50% additional complexity (about 6k gates) in 1.2 μm standard cell CMOS technology.

ST Microelectronics 90 nm CMOS	soft decoder functions	equivalent gates
reception block	least reliable bits identification	744
	serial-in, parallel-out shift register 24x4	1005
processing block	error pattern construction	936
	error pattern encoding	
	metric computation and selection	1155
transmission block	parallel-in, serial-out shift register 12x1	146
control block	5-bit counter	47
soft decoder complexity	4033	

Table 1: hardware complexity of the proposed soft decoder architecture

VI. CONCLUSION

Soft-decision decoding of Golay codes has been investigated and a decoder architecture has been described. The proposed approach relies on a dedicated decoding algorithm which exploits the code properties to achieve near-ML performance using a small number of error patterns. The simulation results and the hardware complexity of the prototype demonstrate the practicality and the benefits of the proposed decoding algorithm.

REFERENCES

- [1] M. J. E Golay, "Notes on digital coding," *Proc. IRE*, vol. 37, p. 657, 1949.
- [2] S. B. Wicker, *Error control systems for digital communication and storage*, Prentice Hall, 1995.
- [3] E. R. Berlekamp, R. E. Peile and S. P. Pope, "The application of error control to communications", *IEEE Commun. Mag.*, vol. 25, no. 4, Apr. 1987, pp. 44-57.
- [4] T.-K. Truong, I.S Reed and X. Yin, "Decoding of $\frac{1}{2}$ -Rate (24,12) Golay Codes," *TDA Progress Report*, vol. 42-97, Jan.-Mar 1989, pp.202-204.
- [5] M. Elia, "Algebraic decoding of the (23,12,7) Golay codes," *IEEE Trans. Inform. Theory*, vol. 33, pp. 150-151, Jan. 1987.
- [6] W.Cao, "High-speed parallel hard and soft-decision Golay decoder: algorithm and VLSI-architecture", in *Proc. IEEE Int. Conf. on Acoustics, Speech and Signal Proc. ICASSP'96*, Atlanta, Georgia, 7-10 May 1996, pp 3295-3297
- [7] A. R. Calderbank, G. D. Forney, and A. Vardy, "Minimal tail-biting trellises: The Golay code and more," *IEEE Trans. Inform Theory*, vol. 45, no. 5, Jul. 1999, pp. 1435-1455
- [8] H. P. Lee and H. C. Chang, "An improvement on Soft-Decoding of the (24,12,8) Golay Code," in *Proc 11th IEEE International conference on Communications Systems, ICCS 2008*, Singapore, 19-21 Nov. 2008, pp. 246-249.
- [9] P. Adde, D. Gomez Toro and C. Jégo, "Maximun likelihood soft decoding of systematic block codes: from algorithm to architecture," to appear in *IEEE Trans. Signal Proc.*, 2012.
- [10] P.Adde, C. Jégo and R. Le Bidan, "Near maximum likelihood soft-decision decoding of a particular class of rate $-1/2$ systematic linear block codes", *Electron. Lett.*, vol. 47, n° 4, Feb. 2011, pp. 259-260.
- [11] D. Chase, "A class of algorithms for decoding block codes with channel measurement information," *IEEE Trans. Inform. Theory*, vol IT-18, Jan. 1972, pp 170-182
- [12] J.-C. Carlach and C. Vervoux, "A new family of block turbo-codes," in *Proc. 13th Int Symp on Applied Algebra, Algebraic Alg. and Error-Correcting Codes*, Honolulu, USA, Nov. 1999.
- [13] J.-C. Carlach and A Otmani, "A systematic construction of self-dual codes, " *IEEE Transactions on Information Theory*, vol. 49, no. 11, Nov. 2003, pp. 3005 – 3009.
- [14] P. Adde, C. Jégo, R. Le Bidan, J.E. Perez Chamorro "Design and implementation of a soft-decision decoder for cortex codes", in *Pr. 31th Int. Conf. on Electronics, Circuits and Systems ICECS 2010*, Athens, Greece, 12-16 Dec. 2010, pp. 663-666