Int. J. Electron. Commun. (AEÜ) 96 (2018) 267–272

Contents lists available at ScienceDirect

International Journal of Electronics and Communications (AEÜ)

journal homepage: www.elsevier.com/locate/aeue

Regular paper

A simplified radix-4 successive cancellation decoder with partial sum lookahead

Hussein G.H. Hassan *, Amr M.A. Hussien, Hossam A.H. Fahmy

Electronics and Communications Engineering Department, Cairo University, Giza 12613, Egypt

ARTICLE INFO

Article history:
Received 1 April 2018
Accepted 23 September 2018

Keywords:
Polar codes
Successive cancellation decoding
Radix-4
Partial sum
Lookahead
Simplified successive cancellation
Special subcodes

ABSTRACT

Polar codes recently received high attention by researchers as proven to approach channel capacity at higher codeword length. However, the decoding latency grows significantly with codeword length, rendering implementation for latency constrained applications impossible. To tackle this problem, this paper proposes a polar decoder architecture based on radix-4 processing units with a special last stage processing unit to decode up to 16 bits in the same clock. In addition, it proposes decoding extended special subcodes to reduce latency. Moreover, it uses partial sum look-ahead technique, resulting in a high throughput low latency decoding architecture.

1. Introduction

In [1] Shannon stated the maximum rate information which can reliably be transmitted over a channel. Since then, tremendous effort has been done aiming to achieve channel capacity [2–4] Arikan achieved a major milestone by proposing polar codes [5]. Polar codes provably achieves channel capacity for binary symmetric channels (BSCs). Several decoding techniques were defined for polar codes such as successive cancellation (SC) decoding [5], list decoding [6] and belief propagation decoding [7].

Although of its decoding simplicity, SC decoding suffers from high decoding latency because of its successive behaviour. A lot of work has been done recently to address the issue of high latency in SC decoding. This is classified into performance improvement of short block length, and minimizing the decoding latency. Similar problems have been addressed in other coding techniques like Low Density Parity Check (LDPC) codes. In [8], short block length rate compatible LDPC codes are presented. Additionally, Ref. [9] introduced a low-complexity construction of girth-8 LDPC with moderate length. For polar codes [10] introduced the usage of log likelihood ratios (LLRs) with minimum approximation, therefore eliminating the multiplications and division operations. In [11], a complex last stage decoding was introduced, where it decoded 2 bits simultaneously. Additionally, it used overlapped scheduling and precomputation decoder to further reduce the latency. In [12] a general log-likelihood-ratio successive cancellation list (SCL) decoder is introduced, where decision of 2^k bits can be determined simultaneously for arbitrary k. Ref. [13] proposed handling special subcodes of rate zero and rate one simply. Ref. [14] showed how polar codes can be used effectively on a binary erasure channel (BEC) with single deletion. The work in [15] proposes a construction for multi kernel polar codes based on the maximization of the minimum distance, Ref. [16] has compared polar decoders vs LDPC and Turbo decoders. In addition, it surveyed error-correction performance and hardware efficiency of different polar decoder techniques. The successive cancellation decoder of a certain bit depends on partial sums computed from previously decoded bits. A generation architecture is introduced in [17] which is mainly a matrix generation circuit and a shift-register-based partial sums computation unit. Ref. [18] transformed unrolled architecture to a multi-mode decoder along with a dedicated decoder for polar codes. A new matrix generation unit implementation was introduced in [19] where a multilevel structure is used.

Ref. [20] introduced a scalable architecture for SC decoding using a semi-parallel encoder-based partial-sum calculation module. Both the belief propagation decoder and the successive cancellation decoder are concatenated forming a hybrid decoder in [21]. Different approaches to improve successive cancellation decoding along with the analysis of the stochastic SC decoding performance were presented in [22]. Ref. [23] introduced a low power

* Corresponding author.
E-mail addresses: hussein.galal@ieee.org (H.G.H. Hassan), ahussien@ieee.org (A.M.A. Hussien), hfahmy@alumni.stanford.edu (H.A.H. Fahmy).

https://doi.org/10.1016/j.aeue.2018.09.027
1434-8411/

high-throughput combinational SC decoder. Ref. [24] studied the faulty SC decoding of polar codes and proves the fully reliable data transfer is not possible at any rate.

This paper shows improvement over the 2-bit SC decoder in [11]. This work proposes a radix-4 architecture for computing LLRs at intermediate stages, and a minimum of simultaneous 4-bit decoding at last stage. It also decodes extra special subcodes of higher length in the same cycle. Additionally, this paper proposes a partial-sum lookahead (PSL) architecture to further reduce the latency. The proposed PSL starts the computation of subsequent nodes without waiting for decoding of current information bits, and corresponding partial sums. This paper extends the work in [25].

This paper is organized as follows; Section 2 presents the proposed architecture. Section 3 shows the results of comparison to state of the art implementation. Section 4 concludes the paper.

2. Proposed architecture

2.1. Overview

In SC decoding, the decoded bits are estimated sequentially. For a codeword size of n, there are n successive computations to decode one complete codeword. The i^{th} decoded bit $\hat{u}_i$ is estimated based on the previously decoded bits $\hat{u}_0, \hat{u}_1 \ldots \hat{u}_{i-1}$. The basic decoder unit consists of two basic computational units; namely f and g. The ordinary successive cancellation decoding depends on using multiple processing units (PUs). Each PU computes f or g functions defined in (1) and (2) respectively. Fig. 2 shows a typical 4-bit polar encoder and decoder. The decoder consists of two decoding levels. In general, an n-bit decoder consists of $log_2(n)$ decoding levels. The first level 0 computes f and g for input LLRs, while the last decoding level $log_2(n)-1$ computes the decoded bits. To reduce the decoding latency, radix-4 based PUs are used to compute ff, fg, gf, or gg allowing latency reduction compared to a conventional radix-2 architecture, and saving the memory required to store intermediate LLRs. Fig. 1 shows the decoding of 16-bit codeword using the proposed architecture. In the last stage, the decoder decodes 4 bits through a special processing unit, defined as Last Stage Processing Unit (LSPU). Section 2.3 presents the proposed unit, as depicted in Fig. 3.

2.2. Radix-4 processing unit

In [10], f, g functions were calculated and simplified to

$$f(L_0, L_1) \approx sgn(L_0)sgn(L_1)min\{|L_0|, |L_1|\} \quad (1)$$

$$g(L_0, L_1, \hat{s}) = L_0(-1)^{\hat{s}} + L_1 \quad (2)$$

where L_0, L_1 are log likelihood ratios and $\hat{s}$ is the partial modulo-2 sum of previously estimated bits. Using the Eqs. (1) and (2), the new radix-4 PU functions can be easily proven to be

$$\begin{aligned} ff(L_0, L_1, L_2, L_3) \approx\ & sgn(L_0)sgn(L_1)sgn(L_2)sgn(L_3). \\ & \min\{|L_0|, |L_1|, |L_2|, |L_3|\} \end{aligned} \quad (3)$$

$$\begin{aligned} fg(L_0, L_1, L_2, L_3, \hat{s}_{00}) \approx\ & sgn(L_1)sgn(L_3).\min\{|L_1|, |L_3|\} \\ & +(-1)^{\hat{s}_{00}} sgn(L_0)sgn(L_2).\min\{|L_0|, |L_2|\} \end{aligned} \quad (4)$$

$$\begin{aligned} gf(L_0, L_1, L_2, L_3, \hat{s}_{01}, \hat{s}_{11}) \approx\ & sgn(L_0(-1)^{\hat{s}_{01}} + L_2). \\ & sgn(L_1(-1)^{\hat{s}_{11}} + L_3). \\ \min\Big\{|sgn(L_0(-1)^{\hat{s}_{01}} + L_2)|, & |L_1(-1)^{\hat{s}_{11}} + L_3|\Big\} \end{aligned} \quad (5)$$

$$\begin{aligned} gg(L_0, L_1, L_2, L_3, \hat{s}_{10}, \hat{s}_{01}, \hat{s}_{11}) =\ & (-1)^{\hat{s}_{10}+\hat{s}_{01}} L_0 \\ & +(-1)^{\hat{s}_{11}} L_1 + (-1)^{\hat{s}_{10}} L_2 + L_3 \end{aligned} \quad (6)$$

where $\hat{s}_{ij}$ is the i^{th} partial sum located in the j^{th} decoding level.

Although a radix-4 PU is more complicated than a regular radix-2 PU computing only f and g, they can be implemented easily with less than double the area and less than double the delay required, a smaller number of PU is used, so the latency decreases with an overall smaller area.

2.3. Last stage processing unit

The LSPU has four LLR inputs denoted by L_0, L_1, L_2, and L_3 respectively, and four binary inputs denoted by $frozen_0, frozen_1, frozen_2$, and $frozen_3$. Each binary input $frozen_i$ indicates whether the corresponding bit u_i is frozen or not. The LSPU computes four decoded bits $\hat{u}_0, \hat{u}_1, \hat{u}_2$, and $\hat{u}_3$ simultaneously. This unit performs the hard decision decoding of Eqs. (3)–(6). Fig. 4 indicates the function of the LSPU. It considers the 6 possible cases labeled as case 1 through case 6 depending on the set of frozen and non-frozen bits. Although there are 2^4 combinations, the algorithm considers only the valid sets of frozen bits based on polar code construction. The notations s_i, s_{ij}, and s_{ijkl} represent the sign bits of L_i, $\{L_i + L_j\}$, and $\{L_i + L_j + L_k + L_l\}$ respectively where the arguments are represented in 2's complement format. The corresponding hardware architecture of LSPU is depicted in Fig. 3.

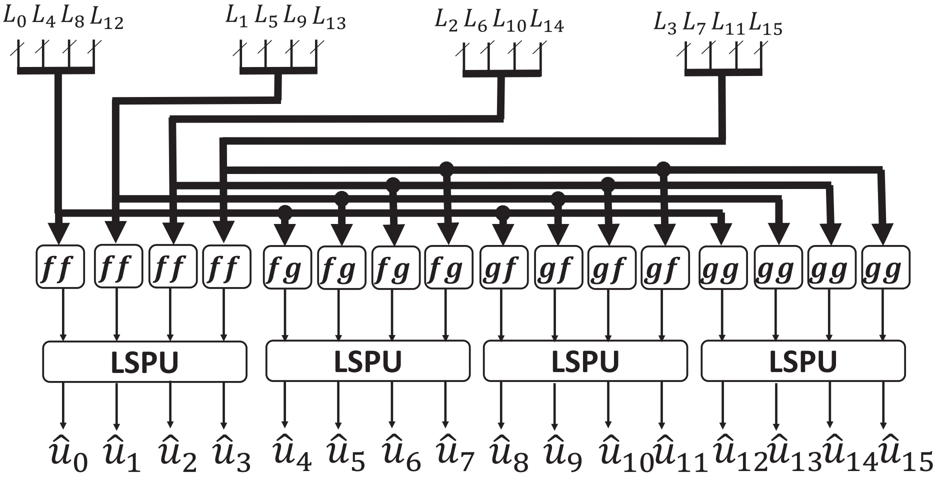
$L_0\ L_4\ L_8\ L_{12}$
$L_1\ L_5\ L_9\ L_{13}$
$L_2\ L_6\ L_{10}\ L_{14}$
$L_3\ L_7\ L_{11}\ L_{15}$
ff
ff
ff
ff
fg
fg
fg
fg
gf
gf
gf
gf
gg
gg
gg
gg
LSPU
LSPU
LSPU
LSPU
$\hat{u}_0\ \hat{u}_1\ \hat{u}_2\ \hat{u}_3\ \hat{u}_4\ \hat{u}_5\ \hat{u}_6\ \hat{u}_7\ \hat{u}_8\ \hat{u}_9\ \hat{u}_{10}\hat{u}_{11}\hat{u}_{12}\hat{u}_{13}\hat{u}_{14}\hat{u}_{15}$

Fig. 1. A radix-4 architecture to 16-bit SC decoder.