Unrolled Gande Papeline, de Decoders based on Look-Up Balatsoukas-Stimming, Maximilian Stark, and Gerhard Bauch) /Title (Unrolled and Pipelado Persler Ored Polar Codes)

Pascal Giard*, Syed Aizaz Ali Shah[†], Alexios Balatsoukas-Stimming[§], Maximilian Stark[†], and Gerhard Bauch[†]
*LaCIME, École de technologie supérieure, Montréal, Québec, Canada.

[†]Institute of Communications, Hamburg University of Technology, Germany.

§Department of Electrical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands. Email: pascal.giard@etsmtl.ca, aizaz.shah@tuhh.de, a.k.balatsoukas.stimming@tue.nl, {maximilian.stark, bauch}@tuhh.de

Abstract—Unrolling a decoding algorithm allows to achieve extremely high throughput at the cost of increased area. Lookup tables (LUTs) can be used to replace functions otherwise implemented as circuits. In this work, we show the impact of replacing blocks of logic by carefully crafted LUTs in unrolled decoders for polar codes. We show that using LUTs to improve key performance metrics (e.g., area, throughput, latency) may turn out more challenging than expected. We present three variants of LUT-based decoders and describe their inner workings as well as circuits in detail. The LUT-based decoders are compared against a regular unrolled decoder, employing fixedpoint representations for numbers, with a comparable errorcorrection performance. A short systematic polar code is used as an illustration. All resulting unrolled decoders are shown to be capable of an information throughput of little under 10 Gbps in a 28 nm FD-SOI technology clocked in the vicinity of 1.4 GHz to 1.5 GHz. The best variant of our LUT-based decoders is shown to reduce the area requirements by 23% compared to the regular unrolled decoder while retaining a comparable error-correction performance.

I. INTRODUCTION

Unrolled decoders are known for their extremely high throughput [?], [?], [?], [?]. In particular, they offer at least one order of magnitude improvement in throughput with respect to standard decoders at the cost of larger area requirements. While this unrolling technique has been applied to successive-cancellation (SC)-based polar decoders before, e.g., [?], [?], it has not yet been combined with look-up table (LUT)-based decoding that has the potential to reduce the required quantization bit-width and, hence, the area and power consumption of the decoder.

Contributions: In this paper, we describe the design and implementation of unrolled and pipelined LUT-based hardware decoders for polar codes. We present three different variants and provide results for all three, along with results for a regular fixed-point decoder, illustrating the challenges of realizing the LUT-based decoders in hardware. In the end, we show that, even for a short (128,64) polar code, a LUT-based decoder can reduce the area requirements by 23% while matching the error-correction performance and exceeding the throughput of a decoder using a standard fixed-point representation.

Outline: The remainder of this paper starts with ?? that provides the necessary background, consisting of a brief review of polar codes and an introduction to the SC and simplified successive-cancellation (SSC) decoding algorithms. Moreover, the concept of unrolled and pipelined hardware architectures is presented as well as that of using LUTs to implement functions. ?? describes our adaptation of the fully-unrolled and pipelined hardware architecture to LUT-based decoding. In particular, the generation of the LUTs, the architectures, and the decoders are discussed. ?? discusses implementation details and provides post-synthesis ASIC area and timing results using the 28 nm FD-SOI CMOS technology from ST Microelectronics. Finally, ?? concludes this paper.

II. BACKGROUND

A. Encoding of Polar Codes

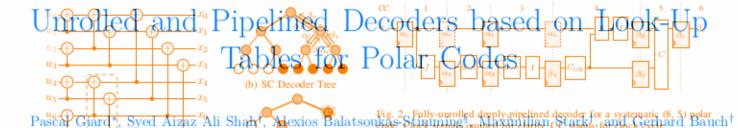
In matrix form, a polar code of length N can be obtained as $\boldsymbol{x} = \boldsymbol{u} \boldsymbol{F}^{\otimes n}$, where $\boldsymbol{F} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$, $n \triangleq \log_2 N$, \boldsymbol{u} is the vector of bits to be encoded, and $\boldsymbol{F}^{\otimes n}$ is the n^{th} Kronecker product of \boldsymbol{F} and $\boldsymbol{F}^{\otimes 1} = \boldsymbol{F}$. To obtain an (N, k) polar code of rate R = k/N, the k most-reliable bit locations in \boldsymbol{u} are used to hold the information bits while the other N-k bits, called frozen bits, are set to a predetermined value (usually 0). The bit-location reliabilities depend on the channel type and condition.

The encoding process can also be represented as a graph like that of Fig. $\ref{Fig. 27}$, where \oplus represents modulo-2 addition (XOR). In that representation, a codeword is generated by setting the frozen-bit locations (the u_0 to u_2 in light gray) to 0 and the information-bit locations (the u_3 to u_7 in black) to the message to be encoded, and by propagating the data through the graph, from left to right. As described in $\ref{Fig. 27}$, systematic encoding can be carried out by feeding the output values (x_0 to x_7) into the left-hand side, resetting the frozen-bit locations to 0, and propagating the data through the graph again.

B. Successive-Cancellation Decoding and Simplified Successive-Cancellation Decoding

The SC decoding algorithm was proposed in the seminal work that introduced polar codes [?]. Illustrating its execution using a decoder-tree representation, it proceeds by visiting the tree—e.g., Fig. ??—sequentially, from top to bottom, from left to right, successively estimating \hat{u} at the leaf nodes, from the noisy channel values. Visiting a left edge (blue) on this representation, the SC algorithm can calculate the soft-input log-likelihood ratios (LLRs) α_l to the child node with the minsum approximation [?]

$$\alpha_l[i] = \operatorname{sgn}(\alpha_v[i]\alpha_v[i+N_v/2]) \min(|\alpha_v[i]|, |\alpha_v[i+N_v/2]|), (1)$$



*LaČÍME, École Chechnologie supérieure, Montréal, Québec, Canada.

(a) Graph †Institute of Consideration is ellemburg University of Technology, Germany.

Fig. 1. Cheprantruent-of-Electrical Engineering) Eindheyen University Beb Technologie Eindheyen, The Northerlands.
Email: pascal.giard@etsmtl.ca, aizaz.shah@tuhh.de, a.k.balatsoukas.stimming@tue.nl, {maximilian.stark, bauch}@tuhh.de

where α_v and N_v are respectively the LLRs and node length from the other honolding and bending reliant by where to achieve seeshing columns of asgic by carefully crafted LUTs in unrolled decoders for polar codes. We show that using LUTs to improve key performance instriction of heart purchase the performance instriction of the performance in may turn out more challenging than expected. We presult three variants of LUT-bash and describe their inner workings as well as circuits in detail. The LUT-based decoders wherem paisethersietstoft fregenarbitaindices de Foden, systematic fioral-recide under SC decoding numbers invited bit occurred in rroscarreg^tian performance Acsbort systematic palar code contained at the efformance According processed the coders are sussed as an internation. All decoding processed the coders are hown or its equivalent an information throughput of little un Wisiting apsight edgen(red)DtBOLLRangloto the childrotte wightive calculated 1 pas 5 GHz. The best variant of our LUTbased decoders is shown to reduce the area requirements by 23% compared to the regular unrolled decoder while retaining a comparable error correction performance.

(3) $\alpha_v[i + N_v/2] - \alpha_v[i]$, otherwise,

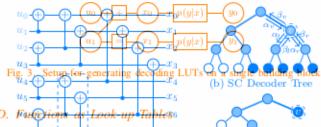
where β_l is the bit-estimate vector generated by the left sibling in the decoder-tree. Afothe left sibling is a leaf node. its restimated bit value we is used parthen at. Otherwise r the estimated-bitdyectormagaliannedembisscalculated asroughput with respect to standard decoders at the cost of larger area requirements. AW hile this when his technique has been applied to successive concellation is C)-based polar decoders before, e.g., [?], [?], it has not yet been combined witerel@landp3;tablethd.bitFestimate vectors from the left- and right whild to des ue spectively uired quantization bit-width and the house algorithm is an ariany trace appropriate that subtrees solely composed of either frozen (rate-0 codes) or information tindes [match is godes), doe not such de to chelchilly travlersed left le Frigt 22h shows no decoder l troip (bin the I(8,1/5)) postar badd war Figd 22 owherefule; SSGr algorithm We applied of .ct.hthe this con Figa? Paist spranted by indeed mixing that about no feits light hand-side [subtree, iseau tate flow bdo) and that the iff ght hands side slubilengiss and ateallization. The FigU??, bthe d'oriner daubtrée isarchiaced by alwhite-dodeeasii the thater with a black slodet. BOB, 64)esparendirecte applications a of d? de, circle rthe cestimated blite vector rioruareate-0t-code 28 al wayslether att-lzero vector and that of intraterforodanie composed of that decisions on the <u>AldRecaseit desegnotscantaindafixedduniancy</u>presentation.

Name Description

f Application of (??)

Outline: Tapplicetina in (22) of this paper starts with ?? that provides Application cess(32), where Cristandall consisting of a brief review of policies of the SC and share for the SC and simplified policeties of Cancellation (SSC) decoding algorithms. Moreover, the Concept of unitalities and pipelined hardware architectures is presented as well as that of using LUTs to implement functions. ?? describes our Clatinatited of the intelligent Halded area Architectures hardware architecture to LUT-based decoding. In particular, the Unrolled decoder architectures provide extremely high decoding speeds dused unrolled descoder architecture of adutant every operation required is instantiated in hardware so that data can flow through the decoder with minimal control ST Mi22 shows a fully rangotle drand deeply pipelined decoder for the (8,5) polar code illustrated in ??. Data flows from left to right. The α and β blocksgittusteated in light blue are registers storing PLRs or dis estimates, respectively. White blocks are the functions described in 22 and dotted registers are regular registers: they will be referred to when discussing partial pipelining no thed following As she contained to blocks do not contain logic, $\mathbf{r} \in \mathbf{r}$ they are equivalent to wires, they product of \mathbf{r} and \mathbf{r} and \mathbf{r} polar are inserted as preprocessing and postprocessing blocks respectively. Among the registers, three are needed to retain the channel LLRs denoted by a in the figure, during the 2nd N-k bits, called mozen bits, are set to a predetermined 3nd and 4nd look eycles (CC), Such renouled architectures yabie (usuallook). The bit-look atom renoulties depend on for polar decoders were described at length in [?].

Preply Pipelined Vs. Partially Sipelined: In a deeply a pinelined architecture such as that illustrated int ?? n adney frameria leader binto the degoder at severy islock cycle. Theres forcer a new Jest imated, code worde is 19150 Loutput 1, 81 (49Ch & Jock sycle. Attany point in timed there are as many frames being decoded as there are nipeline stages. This leads to a very high throughput at the roest of high memory requirements from left to Partial pipelining on the other hands allows to reduce the required careabat (the inest lot reducing the throughput; by irea moving redundant registors in parts of the pipeline where data remains unchanged joyet multiple clock cycles f.? LeRemoving the dotted registers in ?? results in an initiation interval of 2. Realing entite a Covery lation of Poer Civese, and we Simulified Successive Carcallation and adding codeword is estimated. WeThotSChatethbijgtatyalithdy vaffectsotheserbemohy, sootitlad computational relements in the decoder. [?]. Illustrating its



A"LUT is a type of memory that maps a set of input values to output values threan provide quick access to precomputed values that would otherwise require complex calculations. Fig. 1. Graph and decoder-tree representations of an (8, 5) polar and the same vein, LUTs can also be used to approximate mathematical functions. In this work, we use the expression LUT-based decoders to refer to decoders in which arithmetic exemption using a replaced with look-representations to integer value it in use a replaced with look-representations it is integer value it in the remaining investment of integer value at the replace with look and in the LUTs. We therefully upday the look in the look with look the look in the LUTs of replaces the look that the look is timinally left indome by an analytic limiting of integers and calculate the look in the look with the look of the look of the look in the look of the l

If $I^{[i]}$ Un red (ref) and $I^{[i]}$ Precident UT-Baser Simplified Successive-Cancellation Decoding (1) where α_v and N_v are respectively the LLRs and node $I^{[i]}$ and $I^{[i]}$ and $I^{[i]}$ are respectively the LLRs and node length from the parent node, and $\operatorname{sgn}(x)$ returns -1 when x. The LUT claused signed has ingotheoinformation for the red (1) method $I^{[i]}$ decreased by method $I^{[i]}$ decreased down $I^{[i]}$ such that $\operatorname{max}_{p(t|y)}I(X;T)$. The random variable X=x is the designated quantity of primary relevance, e.g., a bit value while the random variable $I^{[i]}$ and $I^{[i]}$ is the compressed or quantized observation. The deterministic mapping $I^{[i]}$ can where $I^{[i]}$ is the set of irozen-bit indices. For a systematic be, treated as a LLC with $I^{[i]}$ and $I^{[i]}$ as input and object vector decoding, the estimated-bit vector

polar code under SC decoding, the estimated-bit vector respectively, can be obtained at the end of the decoding process by Each message t is associated with a LUI $L_x(t)$ which can calculating t = t or its equivalent. be computed from the distribution p(x,t) that is provided by Visiting a right edge (red.), the LLRs α_r to the child the lB solution. Further the finite alphabet T for the quantized messages is chosen such that

 $\alpha_r[I]_x(t) = \alpha_r[I]_x - N_r[2]_y \alpha_x[I]_{L_x} \text{ when } \beta_r[I]_{xy}(t) = 0; \qquad (3)$ where $j = 0, \cdots, \frac{1}{2}, -1$. Moreover, $L_x(t) > L_x(t')$ if t > t', where β_r is the ablustinast parameter at the shared decisions of the big in the distribution of the process of generalities decoding LUTs for a polar code, of length $N_{\text{He}n} = I_x$. In the figure, the uncoded bits in are transformed into coded bits x and received as y after transmission through a quantized additive white Ghussian noises (AWGN) channel much transition probabilities p(y|x) glibe LAWGNO channel parameter is designed for a certain noise ablifice $\log x$ itsing the Burnethoch with the ploits. The fact the Within the quantized champosed upputs if y = y of order and a dear (general ted by a quantizing the interpolation of the plants of the polar desiration of the plants of the polar decided of the polar desiration of the plants of the polar desiration of the polar decided of the polar d

 $p(t_0|y_0)$ can serve as a EUT that replaces the f^5 function. It map y_0 to t_0 which, in turn can be used to decode v_0 as v_0 and v_0 as v_0 as v

with i=0. Similarly, the output $y_1=[y_0,y_1,u_0]$ of the bit Februariologic depis-recompressed on for a systematicing in a clar code $I(b_1 \cup F_1)$ in the bit $F_1 \cap f_2 \cap f_3$ can then replaced the grant on, where it is used to map y_1 to t_1 and decode u_1 using (??).

For a polar code of length the unrother decoding LUTs are obtained by integrating the aforementioned mutual information preserving quantization into its density evolution [?]. These decoding LUTs then replace, (??) and (??) in the SC decoding. This is illustrated for the root node in the decoder tree of Fig. $^{22}_{90R}$ and $^{12}_{4}$ $^{12}_{4}$ $^{12}_{4}$ $^{12}_{4}$ $^{12}_{4}$ $^{12}_{4}$ $^{12}_{4}$ abtained in this section. In this gase in the flater carrying tye ctors α get replaced with \mathcal{O} ectorApplicanteger (Valued messages t. With $N_v=8$, t_v will carry pthis atjuantized) channel eta outputs d_{eta} requesting to very eta . The LUT $p(t_i|y_i, y_{i+N_v/2})$, i.e., the mapping $p(t_0|y_0)$ with labels adjusted for N=8, will be used to generate the updates t_l for the defit did be the following the control of t that other bight nebild ewith the isencrated cusing Ith Fig. UT phe, former, subtreit) is explaced pring pybite, howith adjusted labels. Write thatch separate of ecoding aLA/Tiriscuscul fon teach edge? In the Scheleeddingterbebit.eve240r-forLaUffst in totale As the deaf thodes. (22) can be used to estimate rater dread dank tomFhesedecodebarthatlersesothe obUtFre deki@nedsuisingothenHB framework is the needorth referred to as the IB decoder.

B. Hardblade-EfficienteLinokluHallablance Architectures

The bleffs de Sebtion 22 itsecobtained frode quantized dehsity dvolution; of cedolal modern Thill ethde Noeldraist in it tegt-operation Let II svancy designed one conding doistic scheck anode implated water, i.e., the socialled box-plus operation. The op day Tsiate designed according to the variable-node update rule. From a hardware point of oview, fifficient rollplementation by populing l, declader foighten(8, 5è yalid corlea idifférented) iiLUT Because visefitovo lwfll thavegldiff@bnt ctruthd tablelocklowbystraiteds ibeligficiall to reduceglise mansboriofed Islinet bUTs to esthautite same hardware-Whitent breaks attent berfibectiseds for smiltiple in UTsnd dotted redist(2); LtET-basedapolaristecoders evere idobstructed rwhere thhemidisum-sapproximation itselutilized nfoth designing the As the TC whiled likelocks the not Letrataire idesigned, using the EBuinalhott dtwivas, thewarinin 2 that the effects of guing the tapproximate blinksumesple: forelyUA design the their troschrrection performance to firth dedudenels Integligible to Seed to y decoderhis freferred-troisse ath ANSVAB AECoderl 4th clock cycles (Clfis)arSuMS-tBrodleebder;hiald-threesffol.bjdlsarhaveodhes sarne input/output relation. The number of distinct f LUTs, w.r.t. an IBDleepderPthusineduces frontially-PipelIneMostrinapoleaptly, thic chine-sum chaised: fir E.U. Tichanashet healided susaa echine suffi, of thevirftegere valued message the the oTerRecalle (cf. Section 22) That the integer mes sages embled delevindors at lon, of item to st signifidant bit (MSR) afribedildTiinputscandre treated asabe sign of the agsociated LlaR with MSB pipelineaning megative

IdaRs and vicey vegla.t libeugemainingt bits cean fleighemens the input message magnitude associated to the LLR. The min-sum operation of the imput integer messages rean ablowances sed as: the required area, at the cost of reducing the throughput, by removing redundant registers in parts of the pipeline where data wemains unchanged over multiple clock cycles ?] In Removing of he dested by the egistiment of in early is in used initiation interval of cesnife virgit hat a survey season felock ax complementation view pointed this tast begulvared to the remy encleward time estimated bits or other that the interval went affects the memory point he camputational telements, the Hittel appliabet T. This extra logic can be discarded if a partially flipped finite alphabet is used. More precisely, all the integer messages are relabeled such that they belong A LUT is a type of memory that maps, a set of input to the relabeled finite alphabet $f_{re} = \{3,2,1,0,4,5,6,7\}$ values to ontput values at can provide crick access to instead of $f = \{0,1,2,3,4,5,6,7\}$ for $f = \{8,0,0,4,5,6,7\}$ values to ontput values at can provide crick access to instead of $f = \{0,1,2,3,4,5,6,7\}$ for $f = \{8,0,0,4,5,6,7\}$ values to ontput values that would otherwise require complex the relabeled alphabet, the min-sum expression of $\{7,7\}$ can calculations. In the same vein, LUTs can also be used to directly be applied to integer-valued messages f_{re} to f_{re} . A sprackimate mathematical functions. In this work, we use LUT-based MS-IB decoder that uses f_{re} is referred to as rethe expression LUT-based decoders to refer to decoders. MS-IB decoder. Relabeling has no effect on error-correction in which arithmetic computations are replaced with look-performance. up operations of integer-valued messages [?], [?]. The @liabilittleit Architectore's Gembediand in the integer valued mergaggastward implientellulation we are generally arraft using sour serlace the logic that would normally en into the fieadly fillipstived over the years the extend its runctionality, including Regailer are marking at hunderien usefolders q 21. FBs this work, we fuffile an order to add support for substituting the

f. 9.19.07Ufficers of Physiked EUT-based Simplified

Our toolchain enotably Ctakes natpolar code is onstruction as well as a configuration file as input, and from that, it optimizes the decoder tree, e.g., going from the decoder tree of Fig. ?? to that of 472 me and then generates the inferment among the essing attable mostlons are thereode remain water the stopes of Abaes vera quantity aseat ante from the sort decouer, such that Brayodes dictaics the becoming one or this work, the designat fragware tour of red decoders in verbe, what this realist whiles the random variable $T = t \in \mathcal{T} = \{0, 1, \dots, |\mathcal{T}| - 1\}$ is the compressed or quantized observation. The deterministic may pilm presentation to the Result SUT with yanth this seption we optown insperientation results for variousaclurussasse univonesosiecostervitla afized point tunvones devoder is usual for reference hour idecidate is target a systematic presided bytahoods solutioned Forthety, the 3 in its alphabute incthod of Turnide Vardys Rearid thave unsufficion interval of 10 and a fixed latency of 86 clock cycles. Without loss of generality, systematic coding is used as it offers better bit-error rahe(BER) performance than nonesystematic coding althe cost of a negligible complex hydrogenses The quantization used for the dixelegiont decode was idetermined by fively of a simulation whith ditetrate anothers We denote qualification last QtQ swhere Callist the stotal sumbers of i bets Lised to store a channel LLR and Q_i ? Is the soft bill number off bits assess to stone ant interhal delugg AUTsLRs: usep@lsacomplementerepresentation. Ail fldTfibased decodersowhere designed for fractisf or n3 d1 dB and collect b0, si.c. Arbitimesolutions y after transmission through a quantized



additive white Gaussian noise (AWGN) channel with transition probabilities p(y|x). The AWGN channel quantizer is designed for a certain noise variance σ_N^2 using the IB method with $y \neq 0$ in the MS-IB decoder.

With the quantized channel outputs $[y_0, y_1]$ at hand, the LUTs for decoding v_0 and u_1 are generated by quantizing their bit channels using the IB method. For decoding u_0 , the output $y_0 = [y_0, y_1]$ of its bit channel are quantized into $t_0 \in \mathcal{T}$ such that $\max_{p(t_0 \nmid y_0)} I(U_0; T_0)$. The deterministic mapping $p(t_0|y_0)$ can say we as a LUT that replaces the formula of the property of the

Fig. 5. min-sum for
$$\tau_{re}$$
 in the re-MS-IB decoder. (5)

 X^{it}_{p} unc \overline{ti} on at \overline{Sivcks} \overline{u} , LUT based $\overline{Deebders}$ $[y_0, y_1, u_0]$ of the bit channel of u_1 is compressed to $t_1 \in \mathcal{T}$ aiming this section presents how the polock types to replace adaptated to the LUT-based decoders y_1 to t_1 and decode y_1 Then given is implementing the C and C_{0R} blocks are identical to the fixed-point decoder. The L blocks are similar to those of the fixed point decoder, with an added inverter ner bit. It corresponds to a series of inverters that take the MSB milormation preserving quantization into its density evoof the soft messages of the integer messages tas input and (?The grands to the blocks are realized by the synthesis tools as logic circuits according to the truth table of corresponding LUTS in the three LUT thas ed decoders. In the last ecoder, the fablocks are realized in the same way the vectors of integer va??edlustrates the Wiblock circuit for the MS-iB decoder The interpretation y_0 , of a y_0 -bit horses y_0 , $(t_1, y_1, y_2, y_3, y_4, y_5)$ bit horses y_0 , (t_2, y_3, y_4, y_5) bit horses y_0 , (t_3, y_4, y_5) bits y_0 . By slightsabuse of symbolism, the invertors swith heators as imput and pulpul caury s n t h t t t t t t for the right child $_{\rm wi}$ ll? $_{\rm bill}$ ustrates the $_{\rm us}$ i $_{\rm bl}$ locke circuit, for the re-MS-IB [variant. Comparing Figs (2) and with it control secultarithe critical path for the real MS-IB opright of df for much shorter than that ef-cMS-IB+-tt-consists of a Isingle-multiplexer as approsed to 3?multiplexerseando 2sinyertersa Furthermoret, the namount def resources required is much less using will be preflected in the results of Section 22 to as the IB decoder.

B. Encoheaveelifini Performance and Impact of Quantization

The how state in the connection performance of the systematic (le28) (34) polaricode for puriod leddecoders support, modulated with district pulses which shall be shall be supported by the distribution of the constraint of the

implementation of one, e.g., f, LUT might not be valid for a different f LUT because the two will have different truth tables. However, it is beneficial to reduce the number of distinct LUTs osd that the same hardware-efficient realization can be used for multiple LUTs.

In [?], LUT pased polar decoders were constructed where the min-sum approximation is rtilized for designing the f LUTs while, like [?], the g LUTs are designed using the IB method. In was shown in [2] that the effect of using the approximate min-survival for LUT design on the error-correction performance of the decorer is negligible. Such a decoder is referred to as an MS-IB decoder.

In an MS-IB decoder, all the # LUTs5have the same input/output relation. The myndeb of distinct f LUTs. w.r.t. an IB decoder, thus reduces from N-1 to 1. Most importantly, the min-sum based f LUT can be realized as a min-sum of the integer valued message $t_a, t_b \in T$. Recall (cf. Section ??) that the integer messages embed LLR information. OThe most-significant but (MSB) of the LUT inputs can be treated as the sign of the associated LLR with MSE 10-10 meaning negative LLR and vice versa. The remaining bits can be seen as the input-message magnitude associated to the LR. The grin-sum operation

$$t_o = f(t_a - \Delta \Delta \Delta) + \Delta$$
 (6)

of the input integer messages can be estagessed as: $t_o = f_0(t_a - \Delta - \Delta) + \Delta = f_0(t_b - \Delta) + \Delta = f_0(t_$

In comparison to (??), the argument Δ in (??) is used for pre- and post-processing of the min-sum operation. Fig. 61 Error correction performance of the insternation (\$28,64) polaricode. inverting the magnitude carrying bits of any input message as well as the output mestage if it falls in the first half of COMPARISON OF IUNE GELFE DECODERS FOR A SYSTEMATIC (128-64) the filling alphabet 7. This extra logic can be discarded if a polar code. All decoders have an initiation interval of 10. partially flipped finite alphabet is used. More precisely, all the integer messages accordanceled such retracted belong to the relabeled finite alphabet $\mathcal{T}_{r_{IB}} = \{3_{MS-IB}, 0_{15-MS}, 6_{IB}, 7\}$ in the admin $\mathcal{T}_{r_{IB}} = \{0, 1, 2, 30.495, 6, 7\}$ is $\mathcal{T}_{r_{IB}} = \{0, 1, 2, 30.495, 6, 7\}$. Est $|\mathcal{T}_{0.248}\rangle$. Uncless the relativistic and the holder of the holder of the can directly of applied to integer-values iness ages $t'_a, t_{0.68}^{56.8} \mathcal{T}_{re}$. AAnd Eft a Good Mins IB decorder that 4 uses 74,2 is referred to as re-MS-IB decoder. Relabeling has no effect on errorcorrection performance.

The $Q_i,Q_i=5.4$ fixed-point decoder and the MS-IB and re-MS-IB LUI-based variants are shown to have a coding loss of Thack and view a implementation are represented the ingrania BERVare 1000 claimaireat constituación - Polibuterias directions de la propertie de la propert Meanwarde orac chringers stor rate not its towastionalityerins childing der generate at hand ware unerlied deerdevaues. The ELUT vbaska viecbiteks minarlýfinal ét a ismarter é odadel isop man the constitutions three point recollections eventually come to maker asophelanian notablye taker a golar code construction as well as a configuration file as input, and from that, it Sptfinizes riner as the Unralled Decaders from the decoder tre2? othews. 39nthesis results justing an 28 timer FQ-SQL GMQS tbehnblogyAfrong STe MicroglecthonicsptiAlls decoders were

synthesized to target a clock frequency of 1.5 GHz. The first column is for the Q_i , $Q_c = 5$ fixed point decoder and the last three for the 4-bit LUT-based decoders.

We observe that our first and relatively naive LUT-based implementation (IB) unfortunately has 182% higher area and 6% lower throughput than the baseline fixed point decoder. The situation improves when using the f block of ??. The MS-IB decoder requires 142% higher area for a 4% lower throughput compared to the fixed-point decoder. Finally, significant gains with respect to the baseline decoder are observed when applying the relabeling described in Section ??, where the resulting decoder, re-MS-IB, is 23% smaller and 3% faster than the fixed-point decoder, leading to a 35% better area feligific vate, the types of nodes that can be used, and the type of decoder. The type of podes dictates the decoding algorithm. For this work, we generate hardware unrolled in this work, we showed that replacing blocks of logic by decoders in VHIIL that implement the SSC algorithm. LUTs in an unrolled decoder for polar codes may not necessarily lead to gains in terms of key performance metrics. We presented three variants of LUT-based decoders and compared them againspectiogular/fixed point decoder: We tised acsilder fystemainuspetal Todaesa inusalatel beyorlar earefulityset arting therellers, denedrey is neglection for reference cool operformance hassbeen gysteremine (LOTG dealizahons geleving tiruik etables ffrat Nead 360 efficient entalementation of his was dachieved by deploying the mittigationapproximate/LIOT design together with approphate retabetin Worthautingars and output for Expremasia resting the third variant of the LDT based decoders rie-MSFIB) wasfehawnee bawertormsneterastine ndeesant deceder of all makinik len ombry offering 35% better area effreteney with symitale of wester eighter despectation was relative and in eight of the second of simulation with bit-true models. We denote quantization as $Q_i.Q_c$, where Q_c is the well-presented by the store a cFiberralthdrR would; like the total khMarzieberHashernipour Nazárir@Bindhotvem:UnideRsitAlbfLTeRanosog2/s formprovidéné the restitation thesis Liestilbas Etheleworkers (webscals i Giard fois Supported BydNSERC T Disdovery, Grant #650824on.

A. Functional Blocks in LUT-based Decoders

This section presents how the block types of ?? were adapated to the LUT-based decoders.

The circuits implementing the C and C_{0R} blocks are identical to the fixed-point decoder. The I blocks are similar to those of the fixed-point decoder, with an added inverter per bit. It corresponds to a series of inverters that take the MSB of the soft messages α or the integer messages t as input.

The g and g_{0R} blocks are realized by the synthesis tools as logic circuits according to the truth table of corresponding LUTs in the three LUT-based decoders. In the IB decoder, the f blocks are realized in the same way.

?? illustrates the f block circuit for the MS-IB decoder. The binary form of a 4-bit message t_x is denoted as $[t_{x,3}, t_{x,2}, t_{x,1}, t_{x,0}]$. Thick lines indicate vectors of bits. By slight abuse of symbolism, the inverters with vectors as input and output carry out bitwise inversions.