

# Saturated Min-Sum Decoding: An “Afterburner” for LDPC Decoder Hardware

S. Scholl, P. Schläfer and N. Wehn  
Microelectronic Systems Design Research Group  
University of Kaiserslautern, Germany  
{scholl, schlaefer, wehn}@eit.uni-kl.de

**Abstract**—LDPC codes are usually decoded by iterative belief propagation. However especially for small block lengths conventional belief propagation exhibits significant losses in signal-to-noise ratio compared to maximum likelihood decoding. In this paper we propose the combination of a conventional min-sum decoder enhanced by an advanced decoding scheme, that acts as a kind of “afterburner” to improve the frame error rate. We present hardware architectures and implementation results for a 28 nm ASIC technology. The new decoder has a slightly higher complexity, but provides a gain of up to 1.6 dB signal-to-noise ratio over conventional belief propagation decoding for short block length. In addition, we show, that the new decoder implementation can decrease the amount of dark silicon.

## I. INTRODUCTION

Low-density parity-check (LDPC) codes are powerful channel codes, which are widely used in practical applications. For efficient hardware implementations of LDPC decoders usually belief propagation (BP) heuristics are used. A drawback of these heuristics is, that their frame error rate (FER) is considerably worse than optimum maximum likelihood (ML) decoding. This effect becomes more evident if the block length of the code is short, due to loops in the Tanner graph. It has been shown in literature, that gains of more than 1 dB are possible with Binary Phase Shift Keying (BPSK) modulation [1], [2] and even more when higher order modulation is used for ML decoding.

It is therefore worth to investigate new decoding methods and their hardware implementations, to improve the FER of currently used systems. Several algorithms have been published, e.g. using information set decoding [3], augmented BP [2] or saturated min-sum (SMS) decoding [4].

In practical applications, decoding algorithms must be implemented as dedicated hardware to meet throughput and power requirements. In [4] an SMS decoding algorithm and hardware implementation have been presented to improve the FER of LDPC decoders by up to 1.2 dB. However, the drawback of this advanced algorithm is its complexity, which leads to a poor area efficiency, i.e. large chip area or low decoder throughput.

In this paper, we propose a novel hybrid approach, which features a conventional LDPC decoder and an improved SMS decoder. It combines the superior FER of the SMS decoder with the throughput of a conventional decoder. SMS decoding serves as a kind of “afterburner”, which is only activated if the conventional decoder fails. The main objective of this approach is to increase the communications performance. We present a serial and parallel hardware architecture for this

new approach. These architectures have been implemented using a 28 nm Fully Depleted Silicon On Insulator (FD-SOI) ASIC technology. Detailed implementation results for area, throughput and power consumption are given. Compared to sole SMS decoding of [4] the new approach decreases the average complexity largely. This results in more efficient hardware designs or a further improvement in FER. Gains of up to 1.6 dB have been achieved over conventional min-sum decoding with only little impact on area efficiency and power consumption.

Moreover, it is shown in this paper, that the proposed decoding scheme lends itself very good to the dark silicon phenomenon. The term “dark silicon” stems from the fact that in emerging technologies large parts of a chip cannot be activated simultaneously due to thermal design power constraints. This can be explained by the discontinuation of Dennard scaling [5]. It states that as transistors get smaller, their power density stays constant. The discontinuation of Dennard scaling in future technologies leads to a rapidly increasing on-chip power density, that disallows to run the complete chip with full power, eventually leading to so-called dark silicon. Recent studies show that for 8 nm technologies the amount of dark silicon, i.e. the amount of transistors which cannot be turned on at the same time can be as much as 80% [6], [7], [8]. As illustrated in [9] an intelligent temperature management limiting the peak temperature and avoiding hot spots on the chip, so called dark silicon patterning [10] can reduce the amount of dark silicon. In this paper, we show how dark silicon can be exploited in LDPC decoding by activating costly SMS decoding only on demand and using dark silicon patterning for conventional decoding as will be described in Section V.

The structure of the paper is as follows: Section II introduces the system setup followed by the SMS decoding algorithm in Section III and the new approach in Section IV. Hardware architectures are presented in Section V and detailed results on FER and the ASIC implementation are given in Section VI.

## II. PRELIMINARIES

We consider the transmission of  $N$  code bits  $\mathbf{x} = (x_0, x_1, \dots, x_{N-1})$ ,  $x_i \in GF(2)$  over an additive white Gaussian noise (AWGN) channel using BPSK and 64-QAM modulation. On the receiver side the received bits are described by a vector of log likelihood ratios (LLRs)  $\mathbf{y} = (y_0, y_1, \dots, y_{N-1})$ ,  $y_i \in \mathbb{R}$ , with  $y_i = \ln \left( \frac{p(x_i=0)}{p(x_i=1)} \right)$ .  $|y_i|$  can be seen as a measure of reliability of the received bits. A small  $|y_i|$  corresponds to

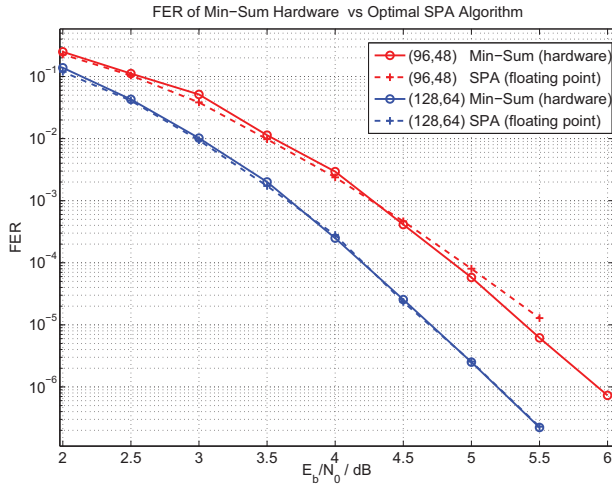


Fig. 1. Min-sum vs optimal SPA: For the considered codes the suboptimal min-sum approximation shows no negative impact on FER.

an unreliable bit that is more likely in error than others. The below described SMS algorithm will utilize this information.

Since the FER of BP in general gets worse for decreasing block length we focus on short LDPC codes. For this kind of codes used in today's or future applications the need for sophisticated algorithms is large. For analysis of the algorithm and the hardware implementation we consider a (96,48) quasi cyclic LDPC code and a (128,64) multi-edge type (MET) LDPC code [11] (public available from [12]).

MET codes have an excellent FER performance and are therefore interesting for systems with strict FER requirements. For the same reason, we have set the maximum number of decoding iterations for the LDPC decoders to a fairly large number of 30, above which only very little improvements in FER can be achieved.

In this paper we use hardware efficient min-sum LDPC decoding, instead of the optimal sum-product algorithm (SPA). However Figure 1 shows, that for the considered codes the min-sum decoder with fixed point values does not compromise the FER in comparison to the SPA. Besides that it should be mentioned, that the ideas in this paper are not restricted to the min-sum check node, but apply to other check node types as well.

### III. THE SMS DECODING ALGORITHM

The SMS decoding algorithm has been introduced in [4] and is briefly reviewed in the following. The algorithm is inspired by augmented BP [2]. This kind of algorithms perform multiple LDPC decoding runs. Each run gets different input LLRs, where some of the unreliable bits are saturated. Finally the best codeword among all runs is output according to the soft decision distance metric commonly used for list decoding [13].

In the SMS algorithm  $S$  bit positions are selected for saturation prior to starting the decoder. In a first step the least reliable bits are determined by partial sorting of  $|y_i|$ . After that,  $2^S$  input LLR vectors  $\mathbf{y}^l, l = 0, \dots, 2^S - 1$  are

generated by setting the least reliable positions to the minimum or maximum LLR value, i.e. they are saturated. In total  $2^S$  vectors are generated and for each vector a decoding run has to be executed.

#### Algorithm 1 SMS Decoding Algorithm

---

Input: LLR vector  $\mathbf{y}$ , number of saturated bits  $S$   
Output: decoded codeword  $\mathbf{x}_{best}$

- 1: Determine  $S$  bit positions  $s_0, s_1, \dots, s_{S-1}$  with smallest reliability  $|y_i|$
- 2: Generate  $2^S$  input LLR vectors  $\mathbf{y}^l, l = 0, \dots, 2^S - 1$ : enumerate positions  $s_0, s_1, \dots, s_{S-1}$  to max or min value
- 3: **for**  $l = 0, \dots, 2^S - 1$  **do**
- 4:   run BP for 30 iterations with input  $\mathbf{y}^l$
- 5:   **if** codeword found **then**
- 6:     save output in  $\mathbf{x}^l$
- 7:   **end if**
- 8:   **if** number of found codewords  $>$  threshold **then**
- 9:     stop loop and **goto** Line 12
- 10:   **end if**
- 11: **end for**
- 12: output  $\mathbf{x}_{best} = \underset{\mathbf{x} \in \{\mathbf{x}^l | l=0,1,\dots,2^S-1\}}{\operatorname{argmin}} \sum_{i=0}^{N-1} y_i x_i$

---

The main difference between augmented BP in [2] and SMS decoding is in its potential of an efficient hardware implementation without compromising FER. Advantages over augmented BP are:

- simplified node selection, i.e. reliability based only
- simplified scheduling of decoding runs
- advanced stopping criterion, which does not impact FER
- simplified check node with min-sum decoding

Since the SMS decoding process is time consuming for large values of  $S$  a stopping criterion has been introduced in [4]. This stopping criterion is based on the observation, that in  $2^S$  decoding runs only a small number of decoders converge to a codeword (often only two or three even for  $2^S = 16$  or  $2^S = 64$ ). Instead of evaluating the results of all decoding runs, the decoder stops after it has found a previously specified number of codewords (regardless if they are different codewords or not). This number is reflected by the threshold given in Algorithm 1 Line 8. The number of codewords that have to be considered is dependent on the used code and the parallelism of the hardware architecture. In [4] it has been shown, that typical values range between 2 to 4 for  $2^S = 16$  or 2 to 5 for  $2^S = 64$ .

### IV. SMS DECODING AS AFTERBURNER

SMS as a standalone decoder is quite complex since many decoding runs have to be performed. Thus its hardware implementation has a poor area efficiency. The main idea of this paper is to overcome this problem by using a hybrid approach of a min-sum decoder operating with the original LLR channel values, in the following termed "conventional decoding", and a sophisticated SMS decoder.

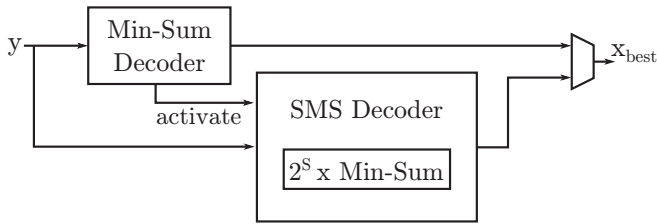


Fig. 2. Proposed Decoding Scheme with Afterburner

Decoding is first tentatively performed with the conventional min-sum decoder. If it fails to find a valid codeword, and only then, SMS decoding is executed. In this configuration SMS decoding acts just like an afterburner in a jet plane: most of the time “normal” and economic operation of the engine is sufficient. But if the pilot requires maximum thrust in critical situations, he can activate the afterburner. It injects additional fuel burned after the turbine of the engine and provides maximum performance, however at a very high cost of fuel. Following this principle: in most cases conventional (and economic) min-sum decoding is sufficient. But if it fails, and only then, a large effort is spend by activating SMS decoding as an afterburner. It “burns” down the intractable LLR vector and thus often eventually outputs the correct code word, achieving very good FER. For planes and decoding the same holds true: since the afterburner is only activated, when required, the overall system operates very economically. Algorithm 2 and Figure 2 outline the procedure.

**Algorithm 2** Decoding Algorithm with an SMS Afterburner

Input: LLR vector  $y$ , number of saturated bits  $S$   
Output: decoded codeword  $x_{best}$

- 1: Perform 30 iterations of standard min-sum decoding
- 2: **if** Codeword is found (and optional: CRC is fulfilled) **then**
- 3:     Output this codeword
- 4: **else**
- 5:     Perform SMS decoding (Algorithm 1)
- 6:     Output SMS result
- 7: **end if**

Figure 3 shows how often the afterburner SMS is activated in comparison to the total number of decoded frames. Note, that different values of  $S$  lead to different plots, because the Signal-to-Noise Ratio (SNR) to achieve a specific FER is dependent on the FER for the afterburner decoding and therefore on  $S$ . Even for an FER of  $10^{-3}$  as it is typically used in communication systems, the SMS afterburner is activated only rarely in approximately 2% of the cases and less.

*Optional CRC aided Decoding:* It is worth to mention, that Algorithm 2 fails, if the conventional decoder finds an incorrect, but valid, codeword. The occurrence of such cases may be dependent on the operating SNR region and the used channel code. Although this problem does not occur with the considered codes for this paper, we propose a solution to this potential problem: the adoption of CRC checks. The CRC can determine with a very high reliability, if the found codeword is the correct one. Checking the correctness of the conventional decoder output avoids accepting a (wrong) codeword. If the CRC fails SMS is activated to raise the probability to find

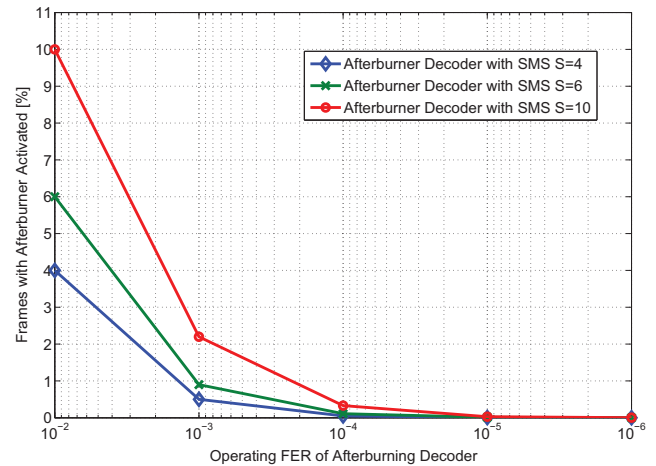


Fig. 3. Afterburner Activation Analysis for LDPC(96,48)

the correct codeword. For many communication standards, that utilize CRC checks, this technique can be easily adopted.

## V. ARCHITECTURE

In the following section the hardware architectures for the decoder are presented. Figure 2 shows the basic decoder setup with SMS afterburning. Since the min-sum decoder and the SMS decoder both require an architecture for min-sum decoding, they can share the same min-sum decoder kernel.

Two approaches are possible: A parallel or a serial architecture, see Figure 4 and Figure 5. In both cases, the min-sum decoder in Figure 2 is moved into the SMS decoder architecture. General min-sum decoding is performed by the “Inner Min-Sum Decoder”. It can be implemented by any state-of-the-art architecture used for LDPC decoding today. For our case we used a well-known partially parallel LDPC decoder architecture implementing min-sum check nodes. It is based on the slot-layered concept [14] processing three submatrices of the parity check matrix in parallel and applying layered decoding. The architecture has a good trade-off between implementation complexity and throughput. However, the presented concept of an afterburner for min-sum decoding can be adapted for any possible inner decoder architecture.

Choosing a serial or parallel architecture for the SMS processing has no impact on the communications performance of the final decoder, but affects area efficiency and latency.

### A. Serial Decoding Architecture

In the serial architecture only one inner min-sum decoder is instantiated. Embedded in the serial decoding architecture, see Figure 4, this inner decoder first performs conventional decoding using the channel values as input. If conventional decoding fails, the additional  $2^S$  decoding runs of SMS decoding are successively carried out by the same inner decoder instance. In this case, it uses saturated input LLR vectors.

The number of iterations for a block, and therefore latency depends largely on the success of the first (conventional) min-sum decoding run. Very often it will converge after only a few

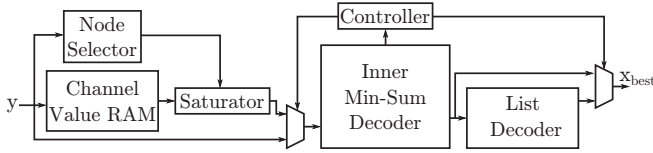


Fig. 4. Serial Decoder Architecture

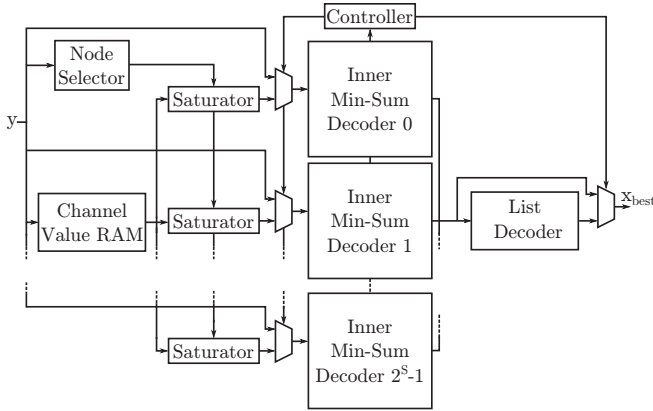


Fig. 5. Parallel Decoder Architecture

iterations. Only if it does not converge, serial execution of SMS is performed which largely increases the latency for this block. The advantage of this serial architecture is its small area and high flexibility. The overhead in means of chip area compared to a conventional min-sum decoder is small (between 10% and 20%, as will be shown later in Section VI, Table I and Table II). Different values for  $S$  can easily be implemented or even changed at runtime, dependent on the current system load. This architecture can be used as a drop-in replacement for state-of-the-art decoders if worst case latency is not critical.

### B. Parallel Decoding Architecture

In the parallel architecture we use  $2^S$  inner min-sum decoders instead of a single one, see Figure 5. Thus the decoding runs for SMS decoding are completely parallelized. As in the serial architecture, the parallel one first executes conventional min-sum decoding on one of the inner decoders. All other  $2^S - 1$  LDPC decoders are power gated at this time and thus require no power at all. Only if min-sum decoding fails, all  $2^S$  SMS decoders are activated and run in parallel with the accordingly saturated input vectors. The latency of the parallel architecture is two times the latency of conventional min-sum decoding in the worst case. However, in most cases, min-sum decoding is successful and no SMS decoding is performed. Therefore the parallel architecture features a high throughput and in most cases with the same energy efficiency as a simple min-sum decoder. The biggest drawback of this architecture is the relatively large area.

### C. Application to Dark Silicon

Considering the dark silicon phenomenon, area is no more the limiting factor but power consumption and heat dissipation. Therefore the parallel architecture is especially interesting for future ASIC technologies, as will be shown in the following.

As already stated, in most cases only one of the  $2^S$  inner decoders of the parallel architecture is active at a time, performing conventional decoding. We can switch off the other decoders by power gating, resulting in a low average power consumption of the decoder. Furthermore, to avoid hotspots, we do not always use the same inner decoder for conventional decoding. Instead we switch after every decoded frame to the next inner decoder in a round robin fashion. Then a single inner decoder is active only for approximately  $1/2^S$ th of the time. Thus the inner decoders can cool down before the next activation and the temperature distribution is averaged. This “heat dissipation aware” decoding largely reduces hot spots on the chip, a major concern in future technologies. This can be seen as a dark silicon patterning [10] for LDPC decoding. SMS decoding is applied only in the rare event of a min-sum decoding failure. In this case all inner decoders are activated for a short time. However, the largest part of the decoder is switched on only rarely (for SMS decoding) and keeps the average power consumption and thus heat dissipation low.

## VI. RESULTS

In the following section, we first introduce the setup used for the hardware implementation, then present the resulting FER performance and ASIC synthesis results.

### A. Implementation Setup

All designs are implemented on a 28 nm low power FD-SOI CMOS library. We considered the following PVT parameters: Worst Case (WC, 0.9 V, 125°C), Nominal Case (NOM, 1.0 V, 25°C) and Best Case (BC, 1.1 V, -40°C). Synthesis as well as place & route (P&R) were performed with the Worst Case PVT settings of the 28 nm library. The designs were synthesized for a target frequency of 500 MHz which is achieved under Worst Case conditions after P&R. The power consumption was calculated using Synopsys PrimeTime-PX. The average power consumption at a clock frequency of 500 MHz is given for the Nominal Case in Table I and Table II. We show results on area, throughput and energy. Area efficiency normalizes throughput to area and allows for a fair comparison of different architectures with different parallelism. Results are shown for a FER of  $10^{-5}$  which is realistic for systems with high demands on the error correction performance. The decoders operate on fixed-point values, which were optimized to not impacting the FER. The chosen quantisation is 7 bits for the (96,48) LDPC code and 8 bits for the MET code.

### B. Simulation Results for FER

Figures 6 to 8 show the FER for the implemented decoders for the (96,48) LDPC and the (128,64) multi-edge type code. All provided SNR gains in dB of the proposed designs are relative to the FER of a conventional min-sum decoder also shown in the figures. Under BPSK modulation the gain ranges between 0.5 and 1.0 dB and under 64-QAM between 0.7 and 1.6 dB for four, six or ten saturated bits. The ML decoding plots show the maximum theoretically achievable gain for the code. This plots have been obtained by the methods described in [1]. With ten saturated bits the FER can be pushed close to ML decoding performance.

TABLE I. COMPARISON OF CONVENTIONAL MIN-SUM AND SMS AFTERBURNER DECODER ARCHITECTURES FOR THE (96,48) CODE

Architecture	min-sum	min-sum + serial SMS	min-sum + parallel SMS	min-sum + serial SMS	min-sum + parallel	min-sum + serial SMS
Code	(96,48)					
# saturated bits $S$	-	<b>4</b>	<b>4</b>	<b>6</b>	<b>6</b>	<b>10</b>
Post P&R Area [mm <sup>2</sup> ]	0.06	0.07	0.89	0.07	3.52	0.07
TP @ FER=10 <sup>-5</sup> [Mbit/s]	797	713	737	570	712	86
Area Eff. [Gbit/s/mm <sup>2</sup> ]	<b>14.6</b>	<b>10.4</b>	<b>0.8</b>	<b>8.3</b>	<b>0.2</b>	<b>1.3</b>
Worst Case Latency [iterations]	30	510	60	1950	60	30750
Energy Efficiency [pJ/Codeword]	394	551	533	690	552	4574
<b>SNR gain BPSK [dB]</b>	<b>0.0</b>	<b>0.5</b>	<b>0.5</b>	<b>0.7</b>	<b>0.7</b>	<b>1.0</b>
SNR gain 64-QAM [dB]	0.0	0.7	0.7	1.2	1.2	1.6

TABLE II. COMPARISON OF CONVENTIONAL MIN-SUM AND SMS AFTERBURNER DECODER ARCHITECTURES FOR THE (128,64) MULTI-EDGE TYPE LDPC

Architecture	min-sum	min-sum + serial SMS	min-sum + parallel SMS	min-sum + serial SMS	min-sum + parallel SMS
Code	(128,64)				
# saturated bits $S$	-	<b>4</b>	<b>4</b>	<b>6</b>	<b>6</b>
Post P&R Area [mm <sup>2</sup> ]	0.11	0.13	1.82	0.13	7.24
TP @ FER=10 <sup>-5</sup> [Mbit/s]	420	388	391	364	382
Area Eff. [Gbit/s/mm <sup>2</sup> ]	<b>3.7</b>	<b>3.1</b>	<b>0.2</b>	<b>2.9</b>	<b>0.05</b>
Worst Case Latency [iterations]	30	510	60	1950	60
Energy Efficiency [pJ/Codeword]	2020	2449	2431	2614	2492
<b>SNR gain BPSK [dB]</b>	<b>0.0</b>	<b>0.4</b>	<b>0.4</b>	<b>0.5</b>	<b>0.5</b>
SNR gain 64-QAM [dB]	0.0	0.7	0.7	1.0	1.0

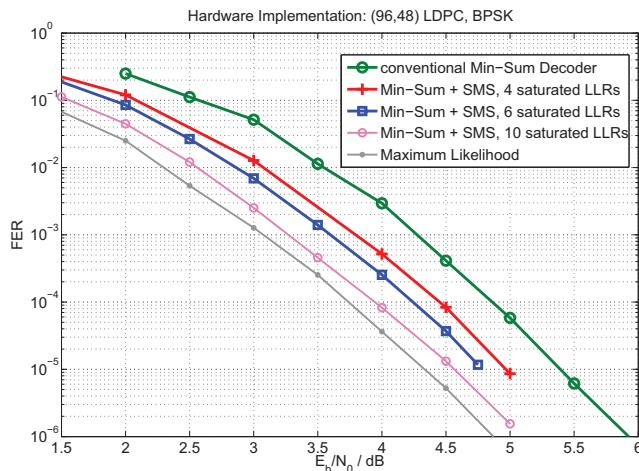


Fig. 6. (96,48) LDPC Decoder Results

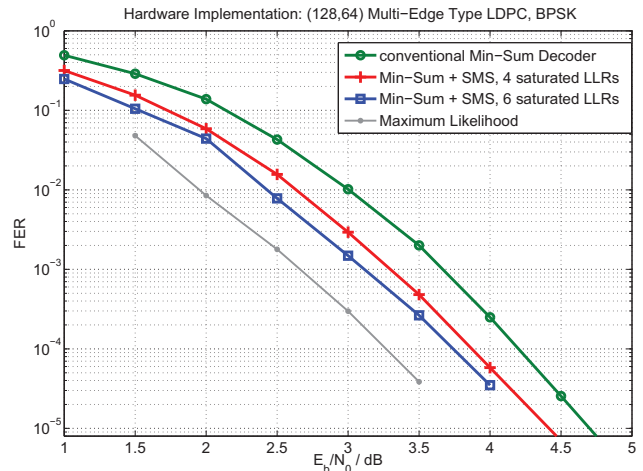


Fig. 7. Multi-Edge Type LDPC Decoder Results

### C. Synthesis Results

Synthesis results for different SMS decoder architectures and the two considered codes are shown in Tables I and II.

For the serial architectures three scenarios are considered: flipping four, six or ten bits. One can observe that the decoder throughput and area efficiency is very good. Only when flipping ten bits, there is a significant impact on these parameters. This shows that the serial architecture is an excellent candidate to replace a conventional min-sum decoder. The only drawback of this architecture is the worst case latency which is significantly increased. In average the latency is however similar to the one of the min-sum decoder.

Parallel architectures solve this issue and exhibit only a small increase in worst case latency. Due to the fully parallel processing of the SMS decoding, the worst latency is doubled compared to min-sum decoding. However they suffer from reduced area efficiency.

Summarizing, serial architectures perform best unless latency requirements are strict. In this case, parallel architectures should be chosen. If very good FER is required and therefore a large number of saturated bits is used, a fully parallel architecture is no more feasible because of the huge area requirements, while a serial architecture may be prohibitive due to the very high latency. Thus a partial parallel architecture

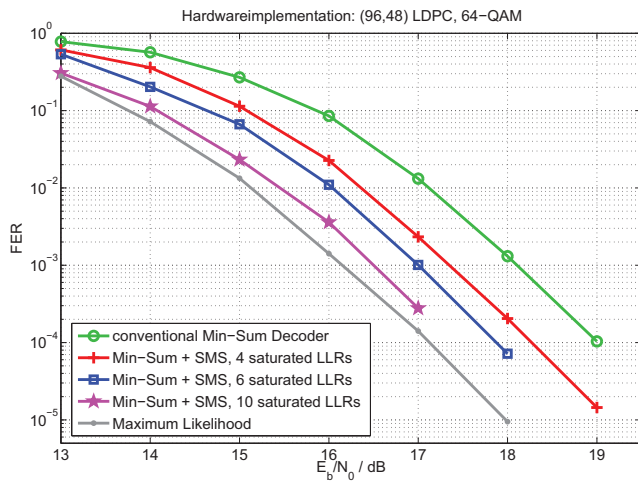


Fig. 8. With 64-QAM modulation SNR gains of 1.6 dB can be achieved

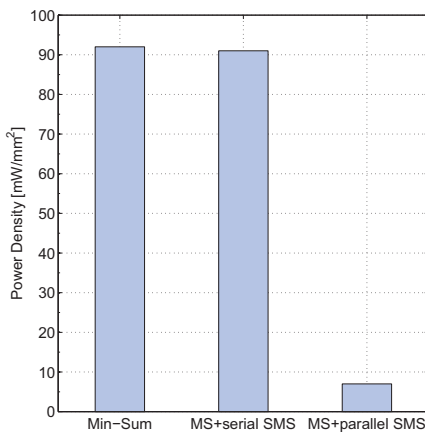


Fig. 9. Power density for the different architectures shows a large advantage for the parallel architecture.

should be considered in future works, which keeps area and latency at a moderate level.

For applications making use of advanced technologies and suffer from the dark silicon phenomenon, the proposed decoder with parallel SMS afterburner is interesting. As shown in Figure 9 the power density of the parallel decoder architecture is significantly lower compared to the conventional min-sum and serial SMS decoder due to power gating. The power density is reduced by approximately 90% from 92 to 7 mW/mm<sup>2</sup>. The serial architecture cannot gain an advantage in terms of power density.

As highlighted in [9] lowering the power density for the parallel architecture leads to a reduction of dark silicon and makes the proposed architecture an excellent candidate for future platforms. Moreover, the proposed “heat dissipation aware” property of the decoder (achieved by continuously switching between the inner decoders) leads to a more even distributed heat dissipation. This avoids hot spots and thus decreases the amount of dark silicon and increases the reliability.

## VII. CONCLUSION

In this paper we have introduced a decoder hardware, that uses SMS decoding as an afterburner, if conventional min-sum decoding fails. The approach has been shown to decrease the average energy and area efficiency for practical FER only slightly compared to a standard min-sum decoder, while providing a large gain in SNR of up to 1.6 dB. Furthermore, it has been shown how the proposed architecture can use dark silicon in future ASIC technologies as an advantage by sophisticated scheduling and power gating.

## ACKNOWLEDGMENT

We gratefully acknowledge financial support by the DFG (project-ID: KI 1754/1-1 and WE 2442/8-1).

## REFERENCES

- [1] S. Scholl, F. Kienle, M. Helmling, and S. Ruzika, “Integer Programming as a Tool for Code Analysis,” in *9th International ITG Conference on Systems, Communications and Coding*, Jan 2013.
- [2] N. Varnica, M. P. C. Fossorier, and A. Kavcic, “Augmented Belief Propagation Decoding of Low-Density Parity Check Codes,” *IEEE Transactions on Communications*, vol. 55, no. 7, pp. 1308–1317, 2007.
- [3] M. Fossorier, “Iterative Reliability-based Decoding of Low-Density Parity Check Codes,” *IEEE Journal on selected Areas in Communications*, vol. 19, no. 5, pp. 908–917, May 2001.
- [4] P. Schläfer, S. Scholl, E. Leonardi, and N. Wehn, “A New LDPC Decoder Hardware Implementation with Improved Error Rates,” in *2015 IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AECT)*, The Dead Sea, Jordan, Nov. 2015.
- [5] R. Dennard, V. Rideout, E. Bassous, and A. LeBlanc, “Design of ion-implanted MOSFET’s with very small physical dimensions,” *IEEE Journal of Solid-State Circuits*, vol. 9, no. 5, pp. 256–268, 1974. [Online]. Available: <http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=1050511>
- [6] H. Esmailzadeh, E. Blem, R. St.Amant, K. Sankaralingam, and D. Burger, “Dark silicon and the end of multicore scaling,” in *Computer Architecture (ISCA), 2011 38th Annual International Symposium on*, 2011, pp. 365–376.
- [7] M. Shafique, S. Garg, J. Henkel, and D. Marculescu, “The EDA challenges in the dark silicon era,” in *Design Automation Conference (DAC), 2014 51st ACM/EDAC/IEEE*, 2014, pp. 1–6. [Online]. Available: <http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6881512>
- [8] M. Shafique, S. Garg, T. Mitra, S. Parameswaran, and J. Henkel, “Dark silicon as a challenge for hardware/software co-design,” in *Hardware/Software Codesign and System Synthesis (CODES+ISSS), 2014 International Conference on*, 2014, pp. 1–10. [Online]. Available: <http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6971829>
- [9] J. Henkel, H. Khdr, S. Pagani, and M. Shafique, “New trends in dark silicon,” in *Design Automation Conference (DAC), 2015 52nd ACM/EDAC/IEEE*, 2015, pp. 1–6. [Online]. Available: <http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=7167304>
- [10] M. Shafique, D. Gnad, S. Garg, and J. Henkel, “Variability-aware dark silicon management in on-chip many-core systems,” in *Design, Automation & Test in Europe Conference & Exhibition (DATE), 2015*, 2015, pp. 387–392. [Online]. Available: <http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=7092419>
- [11] T. Richardson and R. Urbanke, *Modern Coding Theory*. Cambridge University Press, 2008.
- [12] S. Scholl and M. Helmling. (2015) Database of Channel Codes and ML Simulation Results. [www.uni-kl.de/channel-codes](http://www.uni-kl.de/channel-codes).
- [13] T. K. Moon, *Error Correction Coding: Mathematical Methods and Algorithms*. Wiley-Interscience, 2005.
- [14] P. Schläfer, M. Alles, C. Weis, and N. Wehn, “Design Space of Flexible Multi-Gigabit LDPC Decoders,” *Hindawi VLSI Design Journal*, vol. 2012, 2012.