

# Power Optimization in Advanced Channel Decoding

Norbert Wehn

Microelectronic System Design Research Group  
University of Kaiserslautern  
Erwin-Schrodingerstrasse, D67663 kaiserslautern, Germany  
[wehn@eit.uni-kl.de](mailto:wehn@eit.uni-kl.de)

**Abstract.** Channel Coding is an important building block in the outer modem of baseband processing of wireless communication systems. Turbo-Codes and LDPC Codes are the most efficient coding techniques known today. They are already in use of many standards (e.g. UMTS and DVB) and in discussion for emerging standards e.g. WLAN. Due to their computational complexity the implementation of these coding techniques implies many challenges. In this summary we give an overview on techniques to reduce energy consumption in such decoders.

**Keywords.** Channel Coding, Turbo-Codes, LDPC-Codes, Energy Consumption

## 1 Extended Summary

Today's information society demands access to huge amounts of data anywhere and at any time. Hence wireless communications is a key technology. In such systems bandwidth and transmission power are critical resources. Thus advanced communications systems have to rely on sophisticated forward error correction (FEC) schemes. FEC allows to reduce the transmission power by maintaining the Quality of Service (QoS). Since the transmission power is one of the main energy consumers, the use of FEC techniques is an energy optimization technique from a system point of view. Shannon proved 1948 in his pioneering work the noisy channel coding theorem which predicts the minimum bit energy to noise spectral density to achieve reliable communication i.e. the existence of codes for reliable communication. However he gave no hint on the structure of the codes. During the following more than four decades, researches have been trying to find such codes. In 1993 a giant leap was made towards reaching these goal when Berrou published *Turbo-Codes*. The important innovation of Berrou was the introduction of iterative decoding schemes by means of soft information exchange. In the context of Turbo-Codes, *Low-Density-Parity-Check codes* were re-discovered in 1996. They were already invented by Gallager in 1963. However at this time it was impossible to implement these codes due to their implementation complexity. The iterative nature of the decoding algorithms of these codes

implies big implementation challenges with respect to throughput, low latency and low energy consumption.

Turbo- and LDPC codes are already used in some standards with moderate throughput requirements and are in discussion for emerging standards with higher throughput requirements. Especially in handheld devices energy consumption is of great importance. Thus energy minimization in these decoders is an important issue. It is well known that the system level implies the highest optimization potential. Thus, most efficient techniques to reduce power are transformations on the system level. However these optimizations are non bit-true optimizations i.e. the algorithmic behaviour is changed. Thus a careful trade-off between communications performance and implementation performance has to be carried out. In the following we give some examples of system level optimizations based on a UMTS compliant turbo-decoder i.e. we assume a maximum blocksize of 5114 bits and 2 Mbit/s throughput.

- Use of suboptimal decoding algorithms by reducing the operation strength in the algorithm e.g. Max-Log-MAP versus Log-MAP. The Max-Log-MAP costs about 0.1-0.3 dB communications degradation but it saves a factor of 2-3 in energy on DSP implementations. The energy saving in pure hardware implementation is less than 10%. The reason for this difference is that the Log-MAP algorithm has to carry out an exponential function which is implemented as a table-lookup. This table-lookup implies a small hardware overhead, but it takes about 10 instructions on a DSP.
- Use of so called windowing techniques: windowing allows to split up the sequential processing of a data block into smaller subblocks which can be independently processed. This increases slightly the computational complexity and memory accesses but reduces the memory size by a factor of four which yields an energy reduction by a factor of two.
- Efficient quantization and renormalization schemes can save up to 20% energy.
- Use of iteration control: due to the iterative nature of the decoding algorithms, efficient iteration control which stops the decoding process as soon as possible - for decodable and undecodable blocks - saves up to 80% of energy. Combining iteration control with voltage scheduling yields another 10-15% energy reduction.

Parallel architectures are key for high throughput. Moreover they increase power efficiency due to locality and use of voltage scaling techniques. A simple architectural approach is to put several instances of a decoder in parallel. There is no communication between the individual decoders. Unfortunately this architecture has a lower efficiency and a large latency which is critical in many applications. Thus, parallelizing on algorithmic level is a better solution.

The bottleneck in parallel architectures is the data exchange in the iterative loop of the decoding algorithm. This exchange is "randomly" carried out. The quality of the randomness strongly influences the communications performance i.e. the error floor. In the case of LDPC codes the randomness is determined by the Tanner graph, in the case of Turbo-Codes by the interleaver.

LDPC decoders have an inherent algorithmic parallelism. Check- and variable nodes can work independently from each other. Thus, a full parallel implementation can be derived by instantiating each node in hardware and the information exchange between the nodes is implemented by wires. However such a solution is only feasible for small data blocks e.g. 1024 bits. But the block size in the DVB-S2 standard is 64800 bits. Moreover a full parallel implementation provides no flexibility with respect to the "random" data exchange.

Turbo decoders can be also parallelized by using the already mentioned windowing technique. However again the data exchange is the bottleneck.

Hence parallel decoder architectures are *interconnect centric architectures* due to the "random" information exchange. This exchange can be considered as a crossbar functionality with blocking conflict. Especially the blocking behaviour can cause problems. There are different solutions to tackle this problem.

- Conflict avoidance by code design: this is a trend in emerging standards. The interconnect problem is solved on the system level i.e. the interleaver/Tanner graph is designed according to a fixed architectural template with a regular interconnect topology e.g. a shuffling network. However the architectural template imposes constraints on the code which influences the communications performance. This is the most efficient approach with respect to power and throughput but requires a code/architecture codesign.
- Run time conflict resolution: the conflicts are solved on the implementation level. It provides the largest flexibility and has no impact on the communications performance. Packet-based NoC is the most promising method for run time conflict resolution and yields scalable architectures.

We implemented parallel and scalable decoder architectures using the mentioned NoC approach as parametrizable and synthesizable VHDL models and applied architecture driven voltage scaling using a state-of-the art 0.18 $\mu$ m CMOS technology. This technology is characterized for two different supply voltages, 1.8 V and 1.3 V respectively. The throughput decrease when reducing the voltage from 1.8 V to 1.3 V was counterbalanced by increasing the parallelism degree of the architecture. An energy saving of up to 35% per decodable block resulted. Even from an architectural efficiency (=throughput/(area\*energy)) point of view the low voltage decoder was superior to the high voltage architecture i.e. the energy saving was larger than the increase in the area.

The interested reader is referred to the following literature which describes the techniques presented in this extended summary in more detail.

## References

1. Thul, M.J., Gilbert, F., Vogt, T., Kreiselmaier, G., Wehn, N.: A Scalable System Architecture for High-Throughput Turbo-Decoders. In: Proc. 2002 Workshop on Signal Processing Systems (SiPS '02), San Diego, California, USA (2002) 152–158
2. Kienle, F., Brack, T., Wehn, N.: A Synthesizable IP Core for DVB-S2 LDPC Code Decoding. In: Proc. 2005 Design, Automation and Test in Europe (DATE '05), Munich, Germany (2005)

3. Wellig, A., Zory, J., Wehn, N.: Energy- and Area-Efficient Deinterleaving Architecture for High-Throughput Wireless Applications. In: Proc. 2004 International Workshop on Power and Timing Modeling, Optimization and Simulation (PATMOS '04), Santorini, Greece (2004)
4. Kienle, F., Wehn, N.: Joint Graph-Decoder Design of IRA-Codes on Scalable Architectures. In: Proc. 2004 Conference on Acoustics, Speech, and Signal Processing (ICASSP '04), Montreal, Canada (2004) IV-673-676
5. Thul, M.J., Kienle, F., Wehn, N.: A Survey on LDPC- and Turbo-Decoder Implementations. In: International Conference on Software, Telecommunications and Computer Networks (SoftCOM 2003), Venice, Italy (2003) 122-126
6. Gilbert, F., Vogt, T., Wehn, N.: Architecture-Driven Voltage Scaling for High-Throughput Turbo-Decoders. *Journal of Embedded Computing* (2004) accepted for publication.
7. Kienle, F., Thul, M.J., Wehn, N.: Implementation Issues of Scalable LDPC Decoders. In: Proc. 3rd International Symposium on Turbo Codes & Related Topics, Brest, France (2003) 291-294
8. Gilbert, F., Kienle, F., Wehn, N.: Low Complexity Stopping Criteria for UMTS Turbo-Decoders. In: Proc. 2003-Spring Vehicular Technology Conference (VTC Spring '03), Jeju, Korea (2003) 2376-2380
9. Thul, M.J., Vogt, T., Gilbert, F., Wehn, N.: Evaluation of Algorithm Optimizations for Low-Power Turbo-Decoder Implementations. In: Proc. 2002 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '02), Orlando, Florida, USA (2002) 3101-3104
10. Worm, A., Lamm, H., Wehn, N.: Design of Low-Power High-Speed Maximum a Posteriori Decoder Architectures. In: Proc. 2001 Design, Automation and Test in Europe (DATE '01), Munich, Germany (2001) 258-265
11. Gilbert, F., Worm, A., Wehn, N.: Low Power Implementation of a Turbo-Decoder on Programmable Architectures. In: Proc. 2001 Asia South Pacific Design Automation Conference (ASP-DAC '01), Yokohama, Japan (2001) 400-403
12. Worm, A., Michel, H., Gilbert, F., Kreiselmaier, G., Thul, M.J., Wehn, N.: Advanced Implementation Issues of Turbo-Decoders. In: Proc. 2nd International Symposium on Turbo Codes & Related Topics, Brest, France (2000) 351-354