

Summary

This thesis presents the research and academic achievements during the 2014-2019 period. Modern communication and storage standards require efficient Forward Error Correction (FEC). Due to their excellent error correction capability, Quasi-Cyclic Low-Density Parity-check codes (QC-LDPC) are a class of codes employed in wireless standards, digital video broadcasting, and non-volatile semiconductor memories. This fact prompted the research direction we have pursued during the last 5 years, mainly the study of QC-LDPC decoder architecture trade-offs and optimizations. More specifically, within the framework of the project DIAMOND - Message Passing Iterative Decoders based on Imprecise Arithmetic for Multi-Objective Power-Area-Delay Optimization -, in collaboration with researchers from CEA-LETI Grenoble (dr. Valentin Savin), and ENSEA Cergy-Pontoise (prof. David Declercq), we have tried to exploit the advantages of implementing imprecise operations in Low-Density Parity-Check (LDPC) decoder architectures, in order to optimize the cost/area/power consumption. The original project goals – to develop hardware architectures that use imprecise arithmetic – have been largely expanded due to the very favorable research results. The contributions presented in this thesis closely follow the DIAMOND project.

The first step was to develop a collection of LDPC decoder architecture baselines for which the correct operations would be substituted with imprecise arithmetic operations. Due to the significant effort required for developing and verifying such an architecture for a single LDPC code, we have proposed and implemented a template based approach in order to be able to automate the design process. The proposed automatic methodology allows to describe the variation points across different LDPC code matrices, as well as to be able to verify the hardware design automatically. The associated know-how is embedded in the template, with the verification frameworks in SystemVerilog being reused for increased productivity and overall process quality. Several hardware templates corresponding to different memory organizations, schedulings, as well as parallelization degrees have been designed. The primary target has been layered scheduling for QC-LDPC, since it provides the most efficient message storage.

After successfully developing and comparing different template baseline decoder architectures, we have moved to the next step of replacing correct fixed point arithmetic operations with imprecise computation units. An important issue in LDPC design is represented by message storage, due to the large size of codewords: thousands to tens of thousands of messages, with each message in the range of 2-8 bits. This fact motivated extending the research scope to imprecise storage as well. Furthermore, our investigations have shown that investigating imprecise operations for the decoding process provides a more efficient means to optimize design and reduce cost. Thus, DIAMOND project's scope has been widened to investigate imprecise LDPC decoder operations, as well as imprecise storage.

Layered scheduling QC-LDPC offer a unique message memory mapping that reduces to half memory requirements for the extrinsic messages. Furthermore, it has the advantage of increased convergence. The drawback of using layered scheduling is represented by data hazards due to the late update effect caused by memory access time and pipeline. Furthermore, if implementation-wise, the message memory uses banks made of Static Random Access Memory (SRAM) blocks, the access patterns according to the code graph also introduce data conflicts. Hence, two problems need to be solved

in case such architecture choices and scheduling are present. We approached this problem from two directions:

- A set of offline algorithms has been proposed such that an almost optimum message memory mapping and access scheduling that avoid RAW hazards is generated. The message memory mapping represents a hyper-graph coloring problem, while avoiding RAW pipeline hazards represents a traveling salesman problem. In addition to this, since the RAW hazard problem is very constrained, we have also proposed adequate architecture support by using residue message information for correct decoder operation.
- Architecture aware code design for application where the LDPC code is not fixed. The proposed algorithm builds on a well known construction algorithm - Progressive Edge Growth (PEG). The proposed architecture aware PEG (AL-PEG) extends the original PEG by adding new constraints related to pipeline and message memory mapping. It tries to find a successful solution based on a given choice of hardware architecture and code parameters.

The DIAMOND project yielded successful collaborations and high quality research output; at the same time, it also laid the foundations for new research directions such as probabilistic decoding, design and verification of families of hardware architectures, fault tolerant design, etc.