Gbit/s Non-Binary LDPC Decoders:
High-Throughput using High-Level Specifications

Oscar Ferraz, Srinivasan Subramaniyan*, Guohui Wang†, Joseph R. Cavallaro†, Gabriel Falcao, and
Madhura Purnaprajna*

Department of Electrical and Computer Engineering, University of Coimbra, and Instituto de Telecomunicações, Portugal
*Department of Computer Science and Engineering, Amrita Vishwa Vidyapeetham, Bangalore, India
†Department of Electrical and Computer Engineering, Rice University, US
{oscar.ferraz, gff}@co.it.pt, srinivasansubramaniam74@gmail.com, robertwgh@gmail.com, cavallar@rice.edu,
p_madhura@blr.amrita.edu

Abstract—It is commonly perceived that an HLS specification targeted for FPGAs cannot provide throughput performance in par with equivalent RTL descriptions. In this work we developed a complex design of a non-binary LDPC decoder, that although hard to generalise, shows that HLS provides sufficient architectural refinement options. They allow attaining performance above CPU- and GPU-based ones and excel at providing a faster design cycle when compared to RTL development.

I. INTRODUCTION

Exploiting the powerful low-density parity-check (LDPC) codes in their non-binary form [1] requires significant processing and memory access capabilities in order to operate under constrained bit error rate (BER) performance. Compared to binary LDPC codes, the code length can be relaxed but all the arithmetic is performed on the more complex Galois Field domain. The many degrees of freedom of a high performing non-binary LDPC code require a vast set of experiments to be tested successfully. Targeting very small BER involves simulations that can take from months to years to complete [2].

Computer architects who design these systems have to first simulate and test many different codes, and then optimize hardware deployment for achieving high throughput under constrained power limits. In this work we exploit parallelism, pipelining and loop unrolling [2], in order to achieve a high throughput and low-power performance design, a strategy that suits developers with limited hardware skills.

II. CPU, GPU AND FPGA HIGH-LEVEL SPECIFICATIONS

The proposed non-binary LDPC decoder is based on the Min-Max Algorithm, mainly composed of two dominant processing blocks: check node processors that are row-parallel, and variable node processors that are column-parallel.

HLS optimisations and code refactoring implemented consist of (1) using array partitioning on BRAM (instead of using the external DRAM) to expose more R/W ports and increase bandwidth; (2) unrolling of the innermost loops for increasing efficiency of the base core; (3) pipelining of the outermost loops to maximize the occupancy of all the hardware on each clock cycle; and (4) reducing data precision by converting doubles to 8-bit unsigned chars.

TABLE I
THROUGHPUT PERFORMANCE OF THE MIN-MAX ALGORITHM ON CPU, GPU AND FPGA (THE FPGA RESULTS INDICATE THE PERFORMANCE OF A SINGLE CORE).

	Frequency (MHz)			Throughput (Mbps)		
	CPU	GPU	FPGA	CPU	GPU	FPGA (1 CU)
GF(2^2)	2000	1300	**476.2**	0.741	1.856	22.7
GF(2^3)	2000	1300	476.2	0.352	2.335	31.9
GF(2^4)	2000	1300	434.8	0.132	1.013	**38.7**

III. EVALUATION

In Table I we compare LDPC decoder throughput and frequency of operation for CPU, GPU and FPGA devices. The matrix characteristics are $M = 256$, $N = 384$, $d_c = 3$ and $d_v = 2$, with 2, 3 and 4 bits used in codeword representation for GF(4), GF(8) and GF(16), respectively. In the experiments we used a dual-core Denver 2 CPU and an Nvidia GP10B GPU (Jetson TX2) running CUDA. The FPGA adopted is a XCZU15EG-FFVC-900-3-e, using Vivado HLS 2018.3. In the FPGA, although the frequency of operation is higher for GF(4), throughput is superior for GF(16) since more bits are decoded per symbol. A single compute unit (CU) of the GF(4) circuit requires less than 0.5% of the available hardware resources, which implies that more than 200 CUs of LDPC decoder cores can be replicated, producing a theoretical aggregate throughput above 4 Gbps.

REFERENCES

[1] G. Wang, H. Shen, B. Yin, M. Wu, Y. Sun, and J. R. Cavallaro, "Parallel non-binary LDPC decoding on GPU," in *ASILOMAR*, November 2012.
[2] J. Andrade, N. George, K. Karras, D. Novo, F. Pratas, L. Sousa, P. Ienne, G. Falcao, and V. Silva, "Design space exploration of LDPC decoders using high-level synthesis," *IEEE Access*, vol. 5, pp. 14 600–14 615, 2017.

IEEE computer society