High performance IP core for HEVC quantization

Tiago Dias*†, Nuno Roma*‡ and Leonel Sousa*‡
Email: Tiago.Dias@inesc-id.pt Nuno.Roma@inesc-id.pt Leonel.Sousa@inesc-id.pt
*INESC-ID, Rua Alves Redol 9, 1000-029 Lisbon, Portugal
†ISEL – Instituto Politécnico de Lisboa, Rua Conselheiro Emídio Navarro 1, 1959-007 Lisbon, Portugal
‡IST – Universidade de Lisboa, Avenida Rovisco Pais 1, 1049-001 Lisbon, Portugal

***Abstract*—A new class of quantization architectures suitable for the realization of high performance and hardware efficient forward, inverse and unified quantizers for HEVC is presented. The proposed structures are based on a highly flexible and optimized integer datapath that can be configured to provide several pipelined and non-pipelined implementations, offering distinct trade-offs between performance and hardware cost, which makes them highly suitable for most video coding application domains. The experimental results obtained using a 90 nm CMOS process show that the proposed class of quantization architectures is able to process 4k UHDTV video sequences in real-time (3840 $\times$ 2160 @ 30fps), with a power consumption as low as 3.9 mW when the unified architecture is operated at 374 MHz.**

I. Introduction

High Efficiency Video Coding (HEVC) [1] is the latest standard developed by the ISO/IEC MPEG and the ITU-T VCEG, which aims at doubling the data compression ratio (for the same level of video quality) when compared with its predecessor, i.e. the H.264/AVC standard. Such higher coding efficiency mainly results from the adoption of several new or considerable improved coding methods, which include a highly optimized quantization scheme [2]. The set of operations that are involved in the HEVC quantization and de-quantization procedures are relatively complex, requiring the computation of several different multiplications, divisions and additions/subtractions, mostly due to the several optimizations that were exploited in the design of the core transforms [3].

In the encoder, these operations are tightly integrated in the processing chain, being performed in two distinct modules of the coding loop (quantizer and de-quantizer), while a single de-quantization module is present in the decoder. As a consequence, the computational performance of HEVC encoders and decoders greatly depends on the throughput (and latency) achieved with the adopted quantization and de-quantization circuits. This performance demand poses several difficult challenges in the design of HEVC systems, due to the huge data processing rates that are required for the real-time encoding and decoding of high definition video contents. As an example, the real-time encoding and decoding of video sequences in the 4k Ultra High Definition Television (UHDTV) format (i.e. the standard format that is emerging for new multimedia services), involves processing rates as high as 373 Mpixels/s. In practice, such processing requirements can only be met by using highly specialized hardware structures to realize the most critical operations of the video codec, which include the quantization and de-quantization procedures.

Accordingly, this paper presents a new class of quantization architectures suitable for the realization of high performance and hardware efficient HEVC codecs. The proposed processing structures, which are offered in the form of parameterizable IP cores, are based on a rather versatile architecture that can be used to implement quantizers and de-quantizers, as well as unified quantization/de-quantization hardware structures. In fact, to the best of the authors' knowledge, this is the first unified quantization architecture that is proposed for HEVC. The application scenarios of these circuits are ample and include the implementation of hardware accelerators in modern System-on-Chips (SoCs), as well as specialized functional units of Application Specific Instruction Set Processors (ASIPs), using both ASIC and FPGA technologies.

The rest of this paper is organized as follows. The HEVC quantization procedures are reviewed in section II, while the proposed class of quantization architectures is presented in Section III. Section IV discusses the ASIC implementation results that were obtained for several different quantization and de-quantization circuits. Section V concludes the presentation.

II. Quantization in HEVC

HEVC improved the conventional block-based hybrid coding scheme that has been used since H.261, by adopting an enhanced procedure based on hierarchical quadtree-based techniques [2]. In this new approach, each frame is divided into a sequence of square units, called Largest Coding Units (LCUs), which hold the luma and chroma information. The LCUs are composed of one or more basic Coding Units (CUs) that can be recursively subdivided in four equally sized blocks, starting from the 64×64 samples LCU and going all the way down to a minimum of 8×8 samples. Such generic quadtree segmentation structure also considers the subdivision of the CUs into Prediction Units (PUs), used for Intra- and Inter-prediction, and Transform Units (TUs), which are the elementary units defined for transform and quantization processing. Accordingly, quantization is carried out over the TUs, which are also represented as a quadtree structure using square blocks. These blocks' dimensions can vary from 4×4 to 32×32 samples, depending on the adopted transform.

Similarly to H.264/AVC [4], quantization in HEVC is also based on an improved scalar quantizer that was designed not only to maximize the trade-off between the bit-rate and the resulting image quality but also to more accurately manage it. However, an extra attention was given to the design of the HEVC quantizer, in order to reduce the complexity of the quantization and de-quantization procedures [3], [5]. As a result, the following simplifications were introduced:

- all the transform coefficients in one TU are equally quantized and de-quantized, independently of their coordinates

978-1-4799-8391-9/15/$31.00 ©2015 IEEE

TABLE I. DEFINITION OF FUNCTIONS $f(.)$ AND $g(.)$ IN EQS. 1 AND 3.

QP%6	0	1	2	3	4	5
$f(.)$	26214	23301	20560	18396	16384	14564
$g(.)$	40	45	51	57	64	72

within the block;

- quantization and de-quantization can be computed without any divisions, which are replaced by integer multiplications involving the Quantization Step (Qstep) size and arithmetic shifts;
- an integer Quantization Parameter (QP) is used to determine the Qstep size, increasing it by approximately $\sqrt[6]{2}$ (i.e. 12%) for each increment of QP (in the range between 0 and 51);
- the same quantization and de-quantization procedure is applied for all the transform sizes, where the multiplier depends on the considered QP value and the shifts depend exclusively on the transform size.

By following the above considerations, Eq. 1 represents the quantization scheme as specified by HEVC, where *level* is the quantized transform coefficient, *coeff* is the corresponding transform coefficient, N is the size of the transform applied to the considered TU, and B is the bit depth of the input/output signal (e.g. 8 bits). In this definition, the *offset* term (see Eq. 2) is used to provide finer control over the quantization procedure near the origin (i.e. "the dead zone"), while function $f(.)$ implements the fixed-point approximation of the relationship between QP and Qstep (see Table I).

$$level = \left[(coeff \times f\,(\mathrm{QP}\%6) + offset) >> \frac{\mathrm{QP}}{6} \right] >> (29 - \log_2 N - B) \quad (1)$$

$$offset = 1 << (\log_2 N - 10 + B) \quad (2)$$

Equation 3 represents the de-quantization scheme as specified by HEVC, where $coeff_{DQ}$ is the de-quantized transform coefficient corresponding to the quantized transform coefficient *level*. In this definition, the value of $coeff_{DQ}$ is clipped to the range $[-32768, 32767]$, in order to guarantee that the quantization and de-quantization procedures can be computed with 16-bits arithmetic [3]. Function $g(.)$ implements the fixed-point approximation of the relationship between QP and Qstep for the de-quantization procedure (see Table I).

$$coeff_{DQ} = \left[(level \times g\,(\mathrm{QP}\%6)) << \frac{\mathrm{QP}}{6} + offset \right] >> (\log_2 N - 9 + B) \quad (3)$$

III. PROPOSED CLASS OF QUANTIZATION ARCHITECTURES

From a careful analysis of eqs. 1 and 3, it is clear that the HEVC quantization and de-quantization schemes follow quite similar procedures, involving operands with comparable characteristics and the same combination of operations. This fact can be observed by rewriting the two expressions as shown in eqs. 4 and 5, where $offset_{DQ}$ is given by Eq. 6.

$$level = (coeff \times f\,(\mathrm{QP}\%6) + offset) >> \left(29 - \log_2 N - B + \frac{\mathrm{QP}}{6} \right) \quad (4)$$

$$coeff_{DQ} = \left(level \times g\,(\mathrm{QP}\%6) + offset_{DQ} \right) >> \left(\log_2 N - 9 + B - \frac{\mathrm{QP}}{6} \right) \quad (5)$$

$$offset_{DQ} = 1 << \left(\log_2 N - 10 + B - \frac{\mathrm{QP}}{6} \right) \quad (6)$$

Accordingly, these two procedures can be represented by an unique and more generic formulation, as it is shown in Eq. 7. In this alternative formulation, *output* can either be the quantized transform coefficient or the de-quantized transform coefficient of the TU block that is being processed, while *input* is the corresponding transform coefficient or the quantized transform coefficient, respectively. The terms φ and ε represent the offset parameter that controls the accuracy of the quantization scheme near the origin (i.e., *offset* or $offset_{DQ}$) and the scaling factor, respectively, while function $\sigma(.)$ represents the correlated Qstep function (i.e., $f(.)$ or $g(.)$).

$$output = (input \times \sigma\,(QP\%6) + \varphi) >> \varepsilon \quad (7)$$

The resulting combined formulation consist of four operations involving the same arithmetic circuits, i.e. an integer multiplier, some integer adders and a couple of arithmetic and logical shifters. Therefore, this definition can be used not only to develop highly specialized architectures for the realization of the HEVC quantization and de-quantization procedures but also resource-shared architectures, capable of efficiently implementing both procedures. In terms of hardware cost, these unified structures offer some important advantages in the design of video encoders, since they enable the use of time-multiplexed schemes using the same computational circuit to implement both the quantization and the de-quantization modules of the codecs. In addition, they are also very convenient for the combined implementation of the transform and quantization modules, because they allow improving the efficiency and hardware cost of such structures by using a single circuit to support all the necessary operations.

To fully address these implementation scenarios, the class of architectures herein proposed comprehends three distinct processing structures: a Quantization Architecture (QA), a De-Quantization Architecture (DQA) and a Unified Quantization Architecture (UQA). The three architectures are based on a very efficient integer datapath offering the aimed multi-functionality (i.e., capable of realizing the quantization and the de-quantization procedures), as defined by the formulation presented in Eq. 7. Nonetheless, QA and DQA present several specific optimizations to the base design, in order to fine tune their hardware requirements and offered performance levels to their more constrained functionality.

The block diagram of the most generic UQA is presented in Fig. 1. As it can be seen, the proposed class of quantization architectures is based on three major computational circuits (i.e. the 16-bits signed multiplier, the rounding adder and the arithmetic barrel-shifter used in the (re)scaling operations), albeit including other less complex logical elements to provide all the required constant values for the computation of some intermediate values (e.g., smaller adders and shifters to determine the shift amounts). The quantization and de-quantization procedures are realized in four pipelined processing phases (as defined in the bottom of Fig. 1), in order to keep the pipeline as balanced as possible.

Phase `A` is used to generate all the constant values depending on QP, including the Qstep values. Then, the multiplication is performed in phase `B`, while the rounding operation is realized in phase `C`, by using the control parameter value (φ) computed in phase `B`. In phase `D`, the final values of the quantized/de-quantized transform coefficients are adjusted using a barrel shifter, according to the amount and direction

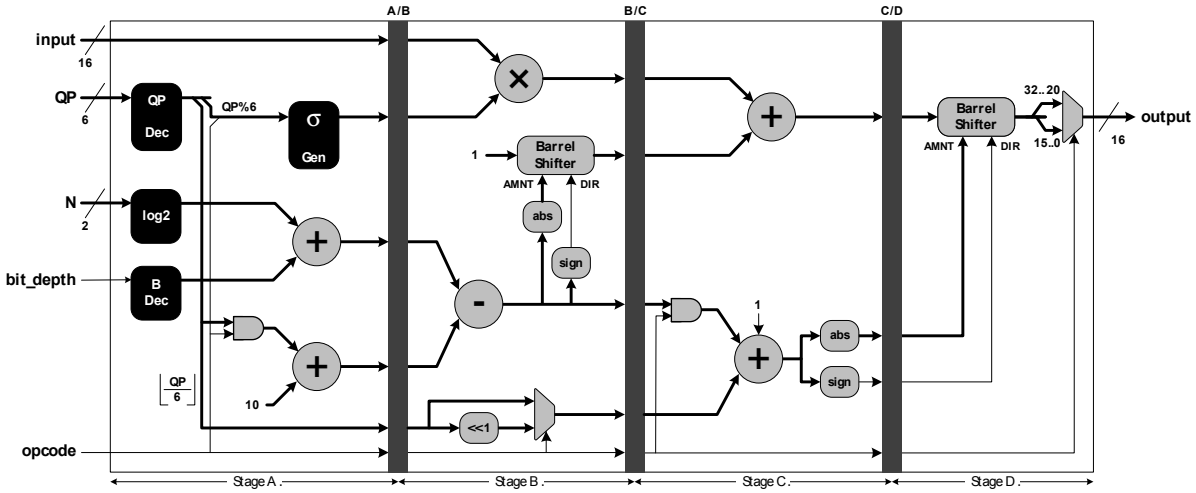

Fig. 1. Block diagram of the proposed Unified Quantization Architecture (UQA); the operation is specified by the `opcode` signal (quantization when `opcode`=0 or de-quantization when `opcode`=1).

TABLE II. POSSIBLE CONFIGURATIONS OF THE PROPOSED CLASS OF ARCHITECTURES.

Configuration	Pipeline registers		
	A/B	B/C	C/D
Non-pipelined	-	-	-
2 pipeline stages	-	√	-
3 pipeline stages	√	√	-
4 pipeline stages	√	√	√

of the shifting operation that are computed in phase `C`. Nevertheless, the preliminary steps in the computation of such values are realized in phases `A` and `B`, in order to optimize the performance of the architecture. The resulting 16-bits value of the quantized (or de-quantized) transform coefficient is provided at the output port of the architecture, in phase `D`.

The configurable nature of the proposed class of architectures allows it to be used according to the four distinct setups listed in Table II. Such configurations consist both of pipelined and non-pipelined versions of the unified datapath, thus offering distinct performance *vs* latency *vs* hardware cost characteristics that allow the system designer to optimally address the requirements of the target application. Still, the throughput that is offered by all the configurations is always one transform coefficient per clock cycle.

IV. EXPERIMENTAL RESULTS

The advantages offered by the proposed class of quantization architectures, in terms of performance and hardware cost, were experimentally assessed with ASIC implementations based on a 90 nm CMOS process from UMC [6]. To achieve such goal, the presented QA, DQA and UQA were described using three distinct IEEE-VHDL parameterizable descriptions. Such descriptions were carried out by considering a generic VHDL coding style, so that efficient realizations can also be obtained for other processes and technologies (e.g. FPGA), and by extensively employing `generic` type parametrization inputs to allow an ample configuration of the aimed IP core (see Table II). Nonetheless, a special attention was given to the description of the most performance-critical modules (e.g. multipliers, barrel-shifters and adders), in order to better assist the synthesis tool in inferring the most efficient primitives for its implementation, according to the chosen synthesis strategy and implementation technology.

In particular, two alternative VHDL descriptions were considered for the multipliers included in the proposed QA, DQA and UQA: *i*) a behavioral description, aimed at the instantiation of embedded multipliers in FPGAs, as well as the realization of generic binary multiplication circuits in ASIC; *ii*) a structural description of a time-multiplexed Multiple Constant Multiplication (mux-MCM) circuit [7] using directed acyclic graphs generated by the SPIRAL project framework (`www.spiral.net`). This alternative description provides a multiplier structure containing several adders and multiplexers that are switched by convenient control logic, in order to compute the product of a transform coefficient by a given constant that can take one out of a few different values. As a result, it allows improving the performance and the hardware efficiency of the multiplication circuits in applications that only compute one product at a time, just like the HEVC quantization and de-quantization procedures.

Table III presents the synthesis results that were obtained for each configuration of the three considered architectures using the devised mux-MCMs, by adopting the `G10K` wire load model and the typical operating conditions of the considered CMOS process (Vdd=1.2V, T=25ºC). Naturally, more conservative results (in terms of the attained maximum operating frequency and the amount of hardware resources) would be obtained after a complete place and route phase. However, given the IP core nature of the proposed architecture, the presented results were obtained by considering only the synthesis phase with the imposition of specific timing constraints to guide the synthesis tool, in order to obtain designs offering the highest performance possible, and wholly capable of reaching the real-time encoding of the 4k UHDTV format (3840×2160 @ 30fps).

In what concerns the hardware cost, the presented results emphasize the reduced and similar amount of resources used by the four configurations of each architecture. In fact, the minor differences observed in the pipelined implementations mostly concern the extra resources that are required to implement the several pipeline registers. This is a direct consequence of the timing constraints that were imposed to the synthesis tool, which resulted in the instantiation of very similar addition, multiplication and shifting circuits for all the considered configurations. The greater silicon areas of the non-pipelined

TABLE III. IMPLEMENTATION RESULTS OF THE QA, DQA AND UQA.

Architecture	Configuration	Area	Gate Count	Max. Freq.
QA	Non-pipelined	0.018 mm^2	3254	512.8 MHz
	2 pipeline stages	0.015 mm^2	2670	571.4 MHz
	3 pipeline stages	0.016 mm^2	2960	684.9 MHz
	4 pipeline stages	0.017 mm^2	3197	694.4 MHz
DQA	Non-pipelined	0.015 mm^2	2662	657.9 MHz
	2 pipeline stages	0.011 mm^2	2073	826.5 MHz
	3 pipeline stages	0.012 mm^2	2212	1000.0 MHz
	4 pipeline stages	0.014 mm^2	2502	1041.7 MHz
UQA	Non-pipelined	0.024 mm^2	4413	465.1 MHz
	2 pipeline stages	0.021 mm^2	3753	546.5 MHz
	3 pipeline stages	0.022 mm^2	3919	632.9 MHz
	4 pipeline stages	0.023 mm^2	4123	632.9 MHz

TABLE IV. COMPARISON OF THE IMPLEMENTED MULTIPLICATION CIRCUITS.

Architecture	Area	Gate Count	Max. Freq.
16-bits signed binary multiplier	0.013 mm^2	2442	645.2 MHz
mux-MCM of the QA	0.012 mm^2	2217	694.4 MHz
mux-MCM of the DQA	0.006 mm^2	1178	1041.7 MHz
mux-MCM of the UQA	0.015 mm^2	2776	666.7 MHz

configurations are also a direct result of the considered timing constraints, which forced the synthesis tool to use more and faster hardware resources in order to meet them.

By comparing these synthesis results with those presented in Table IV, it can be concluded that the adopted mux-MCMs occupy a significant part of the total implementation area (between 50% and 75%). The same can be observed in terms of the maximum clock frequency, which is mainly constrained by the mux-MCMs. Nevertheless, such requirements fully justify the use of mux-MCMs in ASIC implementations, since these circuits can offer significantly higher performances, when compared to the alternative binary multiplier solution. In fact, the data presented in tables III and IV clearly show that the maximum clock frequency of the implemented circuits is constrained by the devised mux-MCMs.

By comparing the results presented in Table III with the processing requirements of the 4k UHDTV format for real-time operation (i.e. 373 Mpixels/s), it can be concluded that the proposed architectures are fully compliant with the requirements of the targeted video format. This is illustrated in Fig. 2, which shows the upper bound limits for the performance that is offered by each configuration of the three architectures, when operated using their maximum clock frequencies. As it can be observed, the non-pipelined configurations of the proposed architectures allow the processing of 657×10^6 transform coefficients per second, while the throughput offered by the fastest pipelined configuration is over 1×10^9 transform coefficients per second. Hence, it can be concluded that the multiple configurations of the proposed architectures effectively allow to trade-off hardware cost for throughput and latency in ASIC implementations.

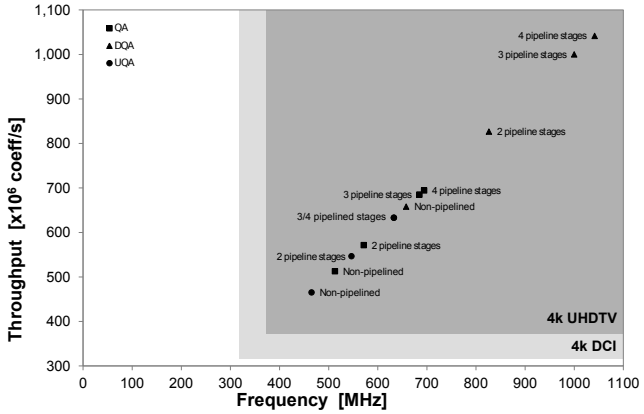

Fig. 2. Performance comparison of the several configurations of the proposed class of architectures when operated with the maximum clock frequency.

Finally, by taking into account the low-power requirements that are usually imposed to video encoding/decoding systems, the power dissipation of the proposed configurations was evaluated when they are executed at the minimum operating frequency that still ensures the real-time encoding of the 4k UHDTV format (i.e., 374 MHz). According to the results that were obtained with the Synopsys Power Compiler, the proposed UQA presents a total power dissipation of 3.9 mW. Such reduced power budget, allied with the presented characteristics in terms of hardware resources and encoding throughput emphasize the advantages that are offered by the proposed HEVC quantization IP core.

V. CONCLUSION

A new class of high performance and hardware efficient architectures for the realization of the HEVC quantization procedures is presented. The proposed structures can be used not only to realize HEVC quantizers and de-quantizers but also unified quantization circuits for the computation of both procedures with reduced hardware cost. In addition, they can be easily configured to provide implementations offering different trade-offs between performance, latency and hardware cost, making them highly suitable for multiple application domains with distinct requisites. The experimental results obtained using ASIC implementations have proved the above observations and showed that the proposed class of architectures can be used to process video sequences up to the 4k UHDTV format in real-time, demanding a power budget as low as 3.9 mW when the unified architecture is used.

ACKNOWLEDGMENT

This work was supported by national funds through Fundação para a Ciência e a Tecnologia (FCT) with reference UID/CEC/50021/2013.

REFERENCES

[1] ITU-T and ISO/IEC, *ITU-T Recommendation H.265 and ISO/IEC 23008-2 'High efficiency coding and media delivery in heterogeneous environments – Part 2: High efficiency video coding'*, ITU-T and ISO/IEC JVT, Apr. 2013.

[2] K. Ugur *et al.*, "High performance, low complexity video coding and the emerging HEVC standard," *IEEE Trans. Circuits Syst.*, vol. 20, no. 12, pp. 1688–1697, Dec. 2010.

[3] M. Budagavi, A. Fuldseth, and G. Bjontegaard, "HEVC transform and quantization," in *High Efficiency Video Coding (HEVC): Algorithms and Architectures*, V. Sze, M. Budagavi, and G. Sullivan, Eds. Springer, 2014.

[4] T. Dias *et al.*, "High performance unified architecture for forward and inverse quantization in H.264/AVC," in *15th Euromicro Conference on Digital System Design (DSD 2012)*, Sep. 2012, pp. 632–639.

[5] M. Budagavi, V. Sze, and M. Sadafale, *JCTVC-G132: Hardware analysis of transform and quantization*, ITU-T and ISO/IEC, Geneva, Switzerland, Nov. 2011, iTU-T and ISO/IEC JCT-VC meeting.

[6] *Faraday ASIC Cell Library FSD0A_A 90nm Standard Cell*, Faraday Technology Corporation, Feb. 2009.

[7] P. Tummeltshammer, J. C. Hoe, and M. Puschel, "Time-multiplexed multiple-constant multiplication," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 26, no. 9, pp. 1551–1563, Sep. 2007.