H.264/SVC deblocking filter prototyping on FPGA platform

Omar Espino Santana, Antonio Núñez Ordóñez, Pedro Pérez Carballo
SICAD Division
Institute for Applied Microelectronics
Las Palmas de Gran Canaria, Spain
oespino@iuma.ulpgc.es

Abstract—**This work describes key concepts in the implementation of a deblocking filter (DF) for a H.264/SVC video decoder. The DF supports QCIF and CIF video formats with temporal, quality and spatial scalability. The design flow starts from a SystemC functional model and has been refined using high-level synthesis methodology to RTL microarchitecture. The process is guided with performance measurements (latency, cycle time, power, resource utilization) with the objective of assuring the quality of results of the final system. The functional model of the DF is created using Open SVC Decoder source code as reference. The design flow continues with the logic synthesis and the implementation on the FPGA.**

The FPGA implementation is capable to run at 100 MHz, and macroblocks are processed in 6,500 clock cycles for a throughput of 130 fps for QCIF format and 37 fps for CIF format.

A validation platform has been developed using the embedded PowerPC processor in the FPGA, composing a SoC that integrates the tasks for frame generation and visualization on a TFT screen. The FPGA implements the DF core. This core is connected to the PowerPC440 embedded processor using a DMA with LocalLink interfaces. The FPGA also contains a local memory capable of storing information necessary to filter a complete frame and to store a decoded picture frame. The complete system is implemented in a Virtex5 FX70T device.

High level synthesis, H.264/SVC, FPGA, validation, platform, resources, frequency.

I. INTRODUCTION

Currently there are many studies describing hardware implementations of the deblocking filter for H.264/AVC video codec standard. Some of these are aimed to reduce the power consumption for embedded designs, in which the battery duration is a key factor. Some implementations are presented in the related work [1, 2, 3].

This paper addresses this issue for the more recent SVC standard, which features improvements to the battery life of devices using different scalability conditions: temporal, spatial or quality, depending on consumption profiles or battery level. The impact of the use of each of the scalability conditions on energy consumption is explained in more detail next.

Temporal scalability. The objective is to change in real time the frame decoding rate, leading to a reduced number of frames to be decoded, thus, reducing the functional use of the device and its power consumption. Several studies look for an optimal way to establish dependencies between frames, so that it can be easier to decide which of them will not be decoded to reduce the decoding rate [4].

Spatial scalability. It enables the reduction of the image resolution, thereby reducing and processing a smaller number of pixels, which also entails reduced consumption [5].

Quality scalability. The variation in the quantization parameter allows selecting different quality levels. As shown in [5], the reduction of this parameter also produces a decrease in the numbers of block edges to be filtered.

II. DF REFERENCE ARCHITECTURE

Figure 1 shows the deblocking filter inner architecture. The deblocking filter is composed of five parts:

- The first block corresponds to the Interface responsible for providing I/O data as well as performing control functions for other blocks.
- Param_bs block, calculates Boundary Strength for each filtering axis for both luminance and chrominance blocks.
- Param_clip block is responsible for calculating the threshold parameters, α and β, and C parameter in case of weak filtering.
- Luma_core block checks whether the conditions for filtering are met and, if so, performs filtering for each luminance block axis.
- Chroma_core block has the same functionality as the luma_core block but for chrominance samples.

III. SYNTHESIS RESULTS

Table 1 presents the summary of the resources used by the system once the design has been implemented in the FPGA.

Deblocking filter memories used for communication between the internal blocks are not included in hierarchical results. That is the reason why the sum of DF block resources does not match the total resource consumption.

The deblocking filter can decode 130 fps of a QCIF image (176x144 pels) and 37 fps of a CIF image (352x288 pels) with a mean latency of 6,500 clock cycles by macroblock (Table 2) working at a frequency of 100 MHz.

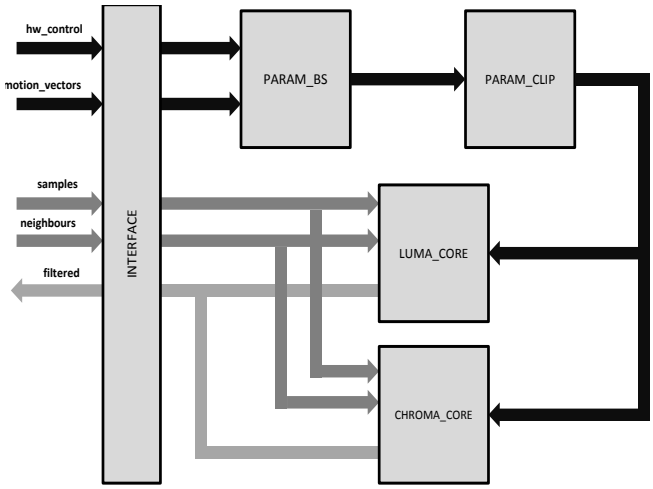

Figure 1. Deblocking filter architecture.

	LUTs	Registers	DSPs	BRAM
d_filter	27,192	18,359	1	10
Interface	637	446	0	0
param_BS	9,552	9,523	0	0
param_clip	1,911	877	1	0
luma_core	9,832	4,170	0	1
chroma_core	4,470	3,099	0	1
ll_interface	336	414	0	0
inout_handler	6,599	3,413	2	0
Memory	2	0	0	96

Table 1 Resources used by FPGA implementation.

Technology	FPGA Virtex5 (CMOS 65 nm)
Frequency	100 MHz
Throughput	QCIF (176x144): @130 fps CIF (352x288): @37 fps
Latency	6,500 cycles/MB

Table 2. Performance results.

To measure the deblocking filter power consumption as well as that of the rest of the prototyping system, Xilinx XPower Analyzer tool has been used. Besides, to improve measurement accuracy, completely routed design and activity information of nodes obtained from the RTL level simulation is provided as input.

Routed design has been provided in NCD Xilinx proprietary format, while activity nodes information in VCD (Value Change Dump).

Table 3 specifies the power consumption obtained regarding the deblocking filter.

Block	Power consumption (mW)
Deblocking filter	18.69

Table 3. Deblocking filter power consumption.

IV. VALIDATION PLAN

This section describes the architecture of the validation platform designed using Xilinx Platform Studio tool.

This system establishes the validation platform for our IP block. The embedded processor generates input frames for the deblocking filter, receives the filtered samples and performs the conversion 4:2:0 - 4:4:4 and YUV - RGB to play the video on the TFT screen. Figure 2 shows the structure of the validation platform.

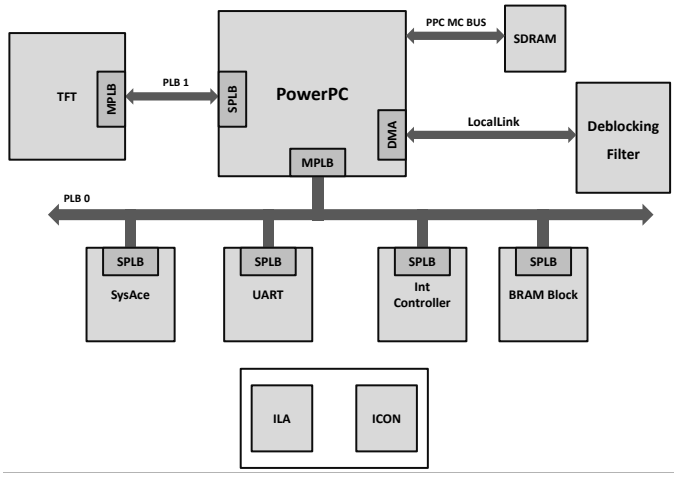

Figure 2. Validation platform block diagram.

V. CONCLUSIONS

In this work it has been presented a deblocking filter implementation for a H.264/SVC video decoder, following a proposed hardware design flow, which starts from the Open SVC Decoder solution. Its functionality is adapted to build a SystemC ESL description. From this point, a high-level synthesis flow has been followed to obtain its hardware architecture. A SoC based on a FPGA validation platform has been designed to verify the correct operation of the design.

VI. REFERENCES

[1] Nadeem, M.; Wong, S.; Kuzmanov, G.; Shabbir, A.; Nadeem, M.F.; Anjam, F., "Low-power, high-throughput deblocking filter for H.264/AVC," System on Chip (SoC), 2010 International Symposium on , vol., no., pp.93,98, 29-30 Sept. 2010.

[2] NamThang Ta; JinSeon Youn; HuiGon Kim; JunRim Choi; Seung-Soo Han, "Low-power High-throughput Deblocking Filter Architecture for H.264/AVC," Electronic Computer Technology, 2009 International Conference on , vol., no., pp.627,631, 20-22 Feb. 2009.

[3] Parlak, M.; Hamzaoglu, I., "A Low Power Implementation of H.264 Adaptive Deblocking Filter Algorithm," Adaptive Hardware and Systems, 2007. AHS 2007. Second NASA/ESA Conference on , vol., no., pp.127,133, 5-8 Aug. 2007.

[4] Conklin, G.J.; Hemami, S.S., "A comparison of temporal scalability techniques," Circuits and Systems for Video Technology, IEEE Transactions on , vol.9, no.6, pp.909,919, Sep 1999.

[5] Richardson, I. E., "H.264 prediction", in *The H.264 advanced video compression standard,* 2nd ed. Chichester, United Kingdom: Wiley, 2010, ch. 6, sec. 5, pp. 172-175.