|  |  |
| --- | --- |
| **Joint Collaborative Team on Video Coding (JCT-VC)**  **of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11**  21st Meeting: Warsaw, PL, 19–26 June 2015 | Document: JCTVC-U0185 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **Comments on intra block copy with interpolation filters** | | |
| *Status:* | Input Document to JCT-VC | | |
| *Purpose:* | Information | | |
| *Author(s) or Contact(s):* | Minhua Zhou, Tim Hellman  16340 West Bernardo Drive San Diego, CA 92127 USA | Tel: Email: | +1-858-521-5838 minhua@broadcom.com |
| *Source:* | Broadcom Inc. |  |  |

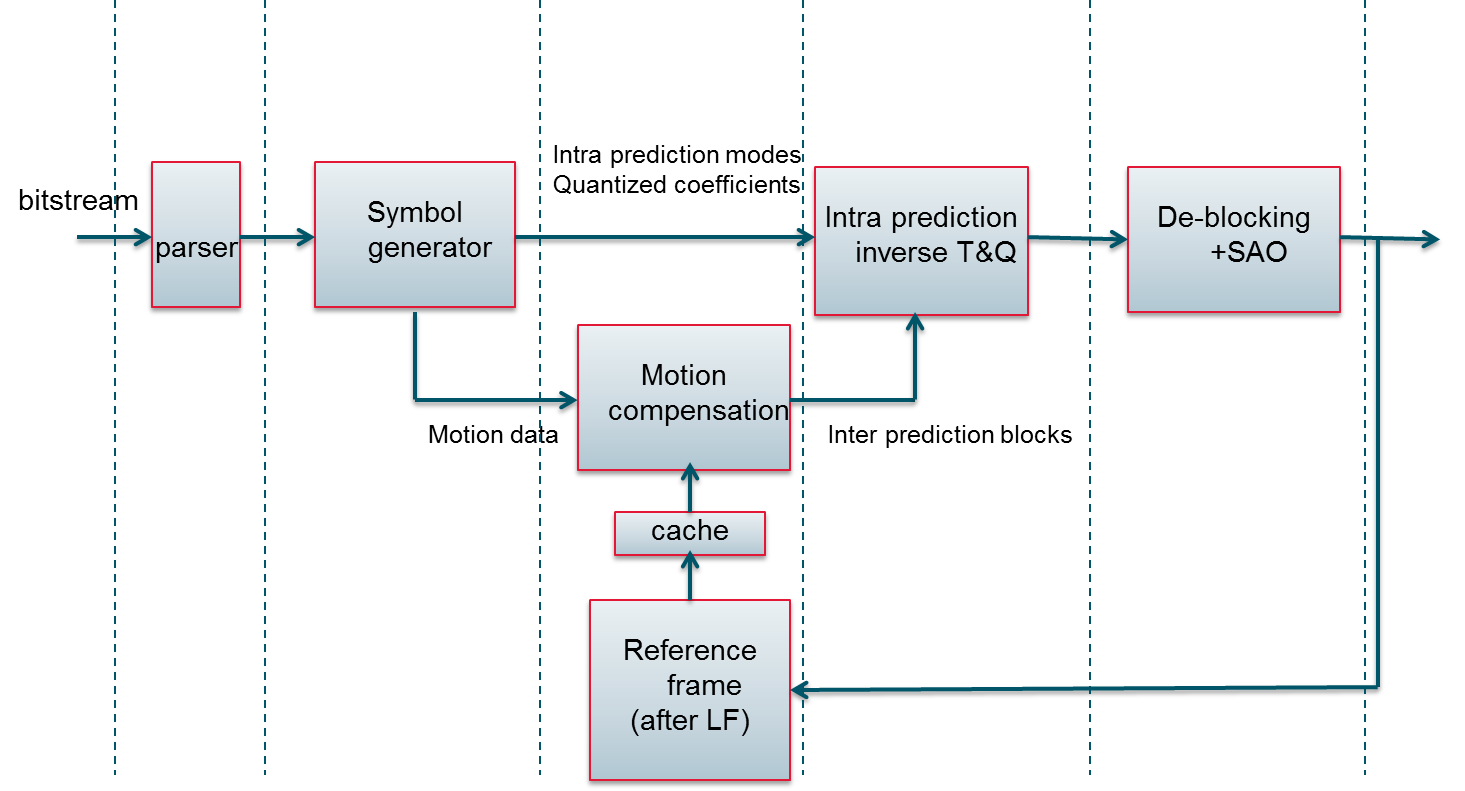
\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Abstract

This document explains how IBC (intra block copy) would be implemented on the top of existing HEVC v1 decoder architecture, and why adding IBC interpolation filters is costly as compared to ones in MC. It is asserted that adding interpolation filters to IBC imposes a significant burden on the HEVC SCC decoder design, and recommends the architecture implication of IBC sub-pel vector support be carefully investigated in addition to its memory bandwidth impact.

# Introduction

For UHD video applications typical market requirement is to support 10-bit 4:2:0 video at 4K@60. The cycle budget is around 0.67 and 1.33 cycle/color pixel for 330 MHz and 660 MHz clock rate, respectively. Such a tight cycle budget poses challenges to real-time HEVC decoder implementations; pipelined hardware architecture is often used to drive performance, lower power consumption and area cost.



**Figure 1. High-level block partitions of a HEVC v1 decoder**

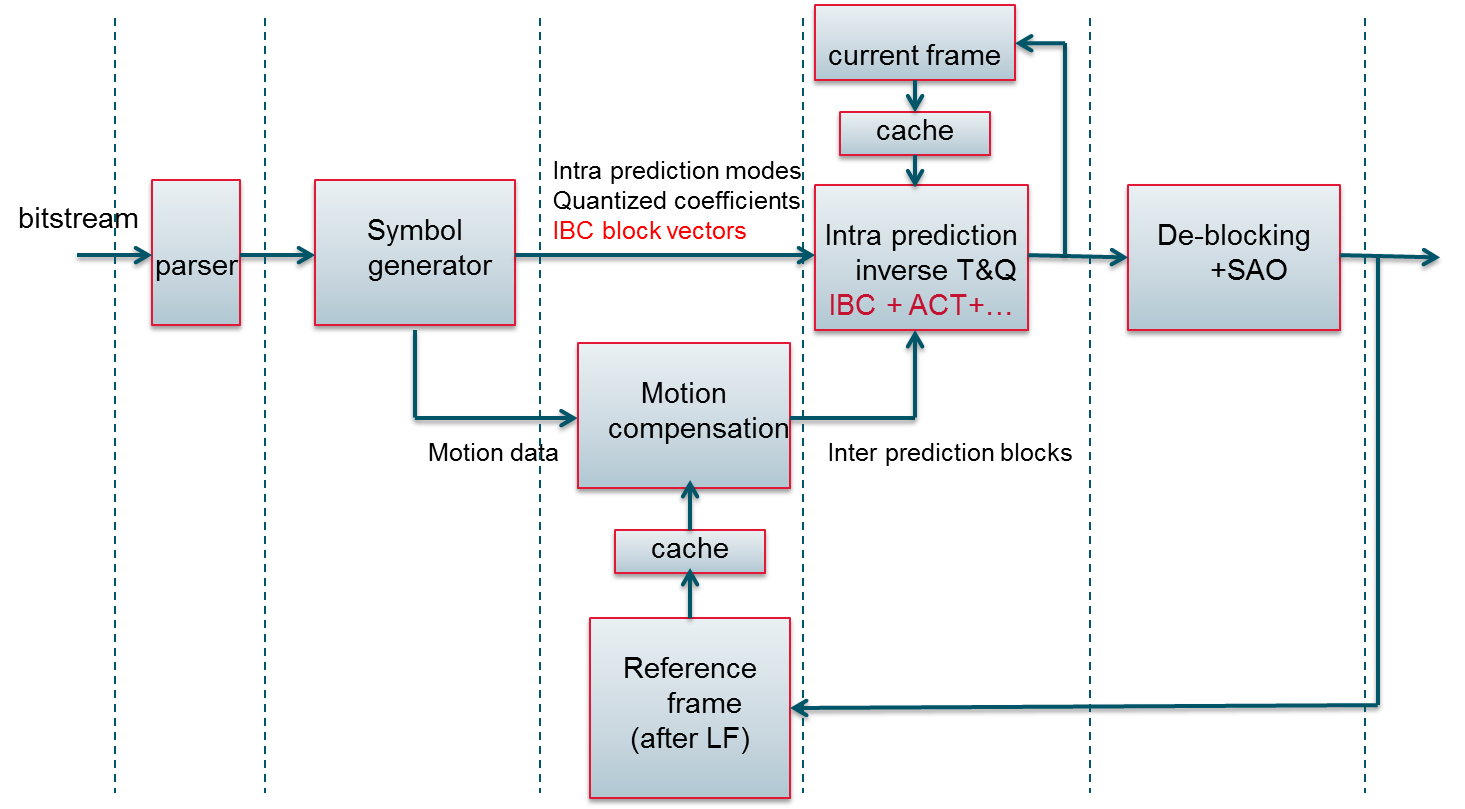
Figure 1 depicts high-level partitions of a typical HEVC decoder, in which CBABC parsing, symbol generation, motion compensation (MC), inverse intra prediction + inverse quantization + inverse transform (ITQ), in-loop filters (de-blocking + SAO) are built in different hardware blocks and placed on different pipeline stages.

For CTU based decoder pipelining, MC depends on the CABAC decoding to obtain motion data, but is independent of the rest of processing blocks. MC operates ahead of ITQ and produces all the inter-prediction blocks for the CTU. ITQ takes the inter-prediction blocks produced by MC as one of input data, performs inverse intra prediction, inverse quantization and inverse transform and generates the reconstructed CTU at the end.

An alternative architecture would be combining the MC and ITQ into one block, and run MC and ITQ parallel at block level, but this requires more overhead for control and sync-up. The alternative architecture is more expensive than the architecture of separate MC and ITQ that is widely used in hardware decoder solutions.

Note that the inverse intra prediction of the current PU depends on the reference sample availability of neighboring PUs. The neighboring PUs need to be fully reconstructed before the inverse intra prediction can be carried out. It is therefore combined with inverse quantization and inverse transform in a same hardware block. ITQ is the second challenging block (after CABAC) for a HEVC decoder design due to the block by block interdependency. In particular, supporting 4x4 intra predictions makes it challenging to meet cycle budget in this block.

# IBC implementation on the top of HEVC v1 decoder



**Figure 2. High-level block partitions of a HEVC SCC decoder**

Figure 2 depicts the high-level partitions of a typical HEVC decoder after IBC added. Like intra prediction, the reconstruction of an IBC-coded PU depends on the reference sample availability of neighboring PUs of the current frame. For CTU based decoder pipelining, it can neither be combined with MC nor be placed in separate block on its own. It must be combined with ITQ. Therefore, IBC is not sharing the hardware logic with MC in hardware design. Anything new added to IBC, such as interpolation filter, will stress ITQ to meet budget and make ITQ design more difficult and costly.

Any pipelined logic has cycle overhead associated for pipeline up and pipeline down (e.g. 8 cycles for an interpolation logic processing 4 pixels per cycle). In the MC, this overhead is at CTU level, while in ITQ the same overhead will be at block-level since block type can change from bock to block. Since the minimum PU size for IBC is 4x8 and 8x4, this pipeline overhead makes interpolation filter implementation in ITQ way more expensive than the one in MC.

Another side effect of adding IBC with interpolation filter is that the existing design of intra prediction, inverse transform and quantization may need to be re-architected and re-designed due to reduced cycle budget caused by adding more complex IBC.

For a HEVC SCC decoder design, the ITQ will be very challenging because not only the IBC, but also other SCC and RExt tools such as ACT, RDPCM, CCP and rotation will have to be all combined into this hardware block. Particularly, ACT is an additional processing step.

# Conclusion

We see that adding interpolation filter to IBC imposes a significant burden on the ITQ design for a SCC decoder, and recommend the architecture implication of IBC sub-pel vector support be carefully investigated in addition to its memory bandwidth impact.

# References

1. ISO/IEC 230082:2014 HEVC Second Edition
2. JCTVC-T1005, “High Efficiency Video Coding (HEVC) Screen Content Coding: Draft 3,” 20th Meeting: Geneva, CH, 10 February – 17 February 2015.

# Patent rights declaration(s)

**Broadcom does not have current or pending patent rights relating to the technology described in this contribution.**