

H.264 Hardware Encoder in VHDL

Notes and usage information

The H.264 hardware encoder is designed as a modular system with small, efficient, low power components doing well defined tasks. The principle design aim was to make an scalable encoder for megapixel images suitable for use in camera heads and low power recorders.

The encoder is not designed to be all things to all people, but rather designed to efficiently implement a non-interlaced Base Profile with no limit to the number of streams or video resolution.

As such few generic parameters are provided, but components can be replaced as needed to customize the codec to a specific application. For example, only CAVLC encoding is performed, but this is performed by the h264cavlc and h264header modules only. If required, these can be replaced by custom modules to perform another form of encoding.

A tour of the Components

A diagram of the principle components is given on the next page.

Video comes in at the top, and is usually written to (external) RAM to buffer it temporarily. When needed it is read into the prediction components such as intra4x4.

Outputs from the prediction components pour through the transform loop: coretransform, quantise, dequantise, invtransform, reconstruct. For intra encoding, these reconstructed pixels are required immediately to predict the next block, in addition they are also written to (external) RAM for use for the next inter coded frame.

Because of the feedback, especially for intra encoding, this transform loop is timing-critical, since the latency is important as well as the throughput. Transform modules need all data in before they can output the first output pixel, but the delay between the last pixel in and the first pixel out is minimal.

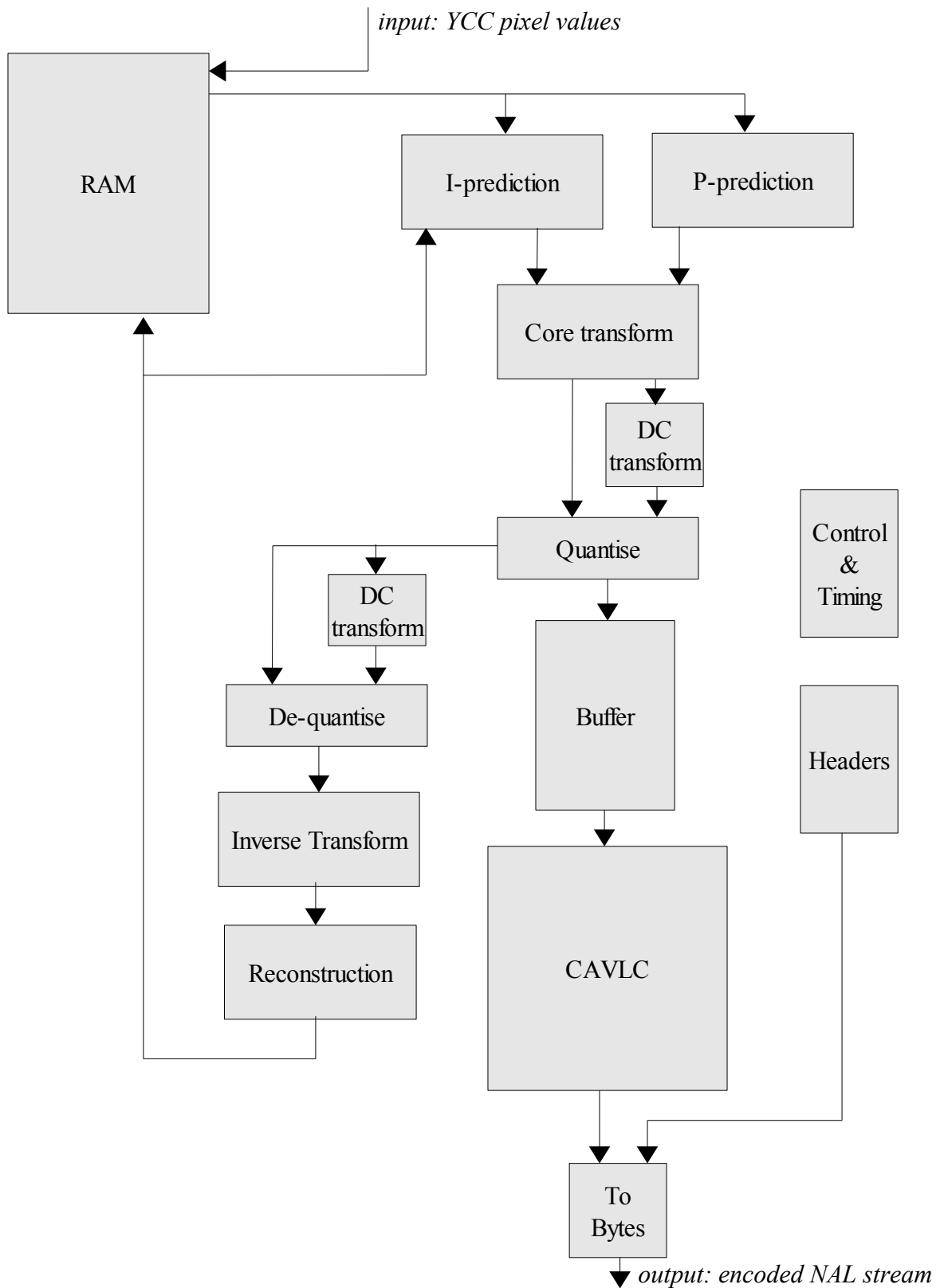
For blocks which use DC as well as AC components, a 2x2 DC transform (Hadamard transform) is also provided as part of the feedback loop. In order to speed the process, the intra8x8cc module (which encodes chroma for intra encoding) outputs the sums of each block as a separate DC data stream which is fed into the first dcttransform.

Output from quantise is fed to the buffer which delays and reorders the blocks for output to the cavlc module which encodes the data. Header data is mixed in after cavlc, and the stream is turned into a byte stream by tobytes, which also stuffs 03 bytes to prevent startcode emulation. The output from tobytes is a NAL, and a done signal is asserted at the end.

It is up to higher level code to add Annex B startcodes between frames (00 00 00 01), or else count and buffer the bytes output and add a header in mp4 or rtp format, or other format as required.

Multiple streams may be simultaneously encoded by the H.264 encoder; these may be of different resolutions. There is no design limit to the video resolution.

The components are independently upgradeable if required.



Principal components in H.264 encoder
 Note that RAM is usually implemented off-chip

Target hardware

The design has been compiled for the following devices:

- Xilinx Spartan 3 family - (3174 Slices)
- Altera Cyclone III family – (26,754 LEs)

It is likely to compile successfully for most other FPGA and ASIC technologies.

This encoder is being built into a commercial application which uses a Spartan 3A 1400 which is about 4 times the size of the requirement quoted above.

Note that the Cyclone III is rather larger than the Spartan 3 due to the use of small or unlatched RAM elements in the design which the Spartan can map to distributed RAM but the Cyclone needs to implement in discrete logic. A modification to intra4x4 and intra8x8cc components to permit TOPI to have a two-clock latency rather than one would permit latched RAM in this situation.

Parameters: PP and SP

It is necessary to include Picture Parameters (PP) and Stream Parameters (SP) to specify the details of the encoder for the decoder to use. These are usually encoded as separate NAL units which can be transmitted immediately before the first NAL unit of image stream.

Some recommended Stream Parameters (SP) are:

profile_idc	01000010	(66)	
constrained_set0_flag	0	(0)	
constrained_set1_flag	0	(0)	
constrained_set2_flag	0	(0)	
constrained_set3_flag	0	(0)	
reserved_zero_4bits	0000	(0)	
level_idc	00101000	(40)	
seq_parameter_set_id	1	(0)	
log2_max_frame_num_minus4	1	(0)	
pic_order_cnt_type	011	(2)	
num_ref_frames	010	(1)	
gaps_in_frame_num_value_allowed_flag	0	(0)	
pic_width_in_mbs_minus1	000010110	(21)	**
pic_height_in_map_units_minus1	000010010	(17)	**
frame_mbs_only_flag	1	(1)	
direct_8x8_inference_flag	1	(1)	
frame_cropping_flag	0	(0)	
vui_parameters_present_flag	0	(0)	

** Replace the pic_width and pic_height with appropriate values, these are in 16-pixel-units and thus the parameters here encode an image of 352x288.

As a NAL, these can be encoded as (hex bytes):

```
67 42 00 28 DA 05 82 59
```

Some recommended Picture Parameters (PP) are:

pic_parameter_set_id	1 (0)
seq_parameter_set_id	1 (0)
entropy_coding_mode_flag	0 (0)
pic_order_present_flag	0 (0)
num_slice_groups_minus1	1 (0)
num_ref_idx_l0_active_minus1	1 (0)
num_ref_idx_l1_active_minus1	1 (0)
weighted_pred_flag	0 (0)
weighted_bipred_idc	00 (0)
pic_init_qp_minus26	1 (0)
pic_init_qs_minus26	1 (0)
chroma_qp_index_offset	1 (0)
deblocking_filter_control_present_flag	0 (0)
constrained_intra_pred_flag	0 (0)
redundant_pic_cnt_present_flag	0 (0)

As a NAL these can be encoded as (hex bytes):

68 CE 38 80

So encoding both using Annex B format (ie, with startcode of 00 00 00 01), gives:

00 00 00 01 67 42 00 28 DA 05 82 59
00 00 00 01 68 CE 38 80

These bytes can be put at the start of the stream (and repeated if needed before any IDR frame).

Clocks

There are two clocks in use by the modules, CLK which is nominally the pixel clock rate, with a design frequency of 60MHz or below, and CLK2 which is a double rate clock and should run at exactly twice CLK rate. CLK is used in the h264cavlc module and also by the back end which emits the byte stream. CLK2 is used by the prediction and transform logic which works at higher data rates than the pixel rate. As a result, 512 double rate clocks are available for each macroblock (256 pixels) for the prediction and transform logic feedback loop.

Top level

A skeleton top level is provided, to allow a real one to be written for your application. A simple top level is provided for simulation which reads and writes files and can dump intermediate data and an annotated output bit stream if required.

RAM

At minimum, an entire uncompressed reference image must be buffered in RAM to allow inter prediction[*]. Usually the incoming image is buffered as well as the reference image, so two copies might be needed. If encoding multiple streams, images from all streams need to be buffered. Depending on resolution, this might be a lot of memory, and thus it is anticipated that this is implemented off-chip.

[*] of course, you could use intra prediction only and thus avoid this overhead, but compression is then usually around 10:1, rather than the 50:1 quoted for inter compressed streams. The actual compression will vary with the contents of the picture stream.

Prediction components

The intra prediction currently available (intra4x4 and intra8x8cc) only considers a subset of possible modes, and uses a simple SAD (sum of absolute differences) comparison. This makes the intra frames a little larger than they otherwise might be, but tests against reference software (which can choose a wider range of modes and other better comparison computations) shows only a few percent improvement.

Also, only a simple inter prediction (p-frames) component is available at present. Zexia has others available but they cannot be released at present under an open source license; they will probably be released once commercial agreements expire. If you want to work on this, please drop me an email, since a pool of good inter prediction components will enhance this codec and I'd be pleased to help.

Since the target is Base Profile, no attempt has been made to encode B-frames, however the same cavlc and transform loop can be used so it's just a case of modifying the front end prediction and header generation.

Patents

H.264 is covered by patents, you will need a license from MPEG-LA. According to the MPEG-LA web site (<http://www.mpegla.com>), there is no royalty payable on less than 100,000 units.

The author knows of no other patents which cover this encoder, but that doesn't mean to say there are none (see the disclaimer below).

Contact

To contact the author and maintainer of this encoder, Andy Henson, email h264@zexia.co.uk

Copyright (BSD-style license)

Written by Andy Henson
Copyright (c) 2008 Zexia Access Ltd
All rights reserved.

--

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

- * Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.

Zexia H.264 Hardware Encoder in VHDL

- * Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- * Neither the name of the Zexia Access Ltd nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

--

THIS SOFTWARE IS PROVIDED BY ZEXIA ACCESS LTD ``AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL ZEXIA ACCESS LTD OR ANDY HENSON BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

End