

Inclusion Of Parallel Processing Schemes In The Main Profile

Stewart Worrall

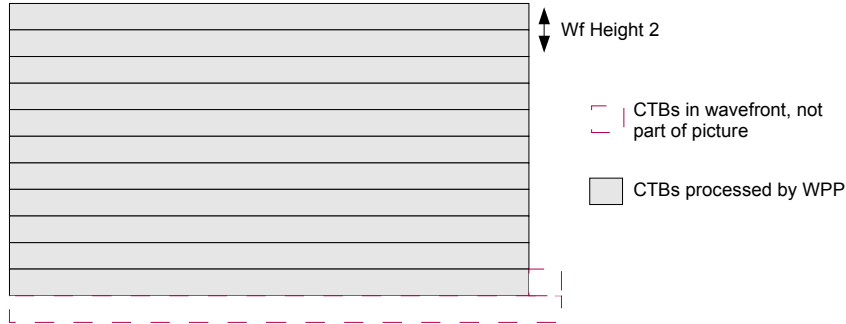
Aspex Semiconductor

August 2011

- Will examine a number of issues relating to the parallel processing tools (tiles and wavefronts)
 - Requirements
 - Parallel processing efficiency
 - Compression efficiency issues
 - Motion Estimation bandwidth
 - Delay
 - Profile specifications
- Conclusions drawn based on these issues
 - Recommend specifying either wavefronts or no parallel tools in the Main Profile

Requirements For Parallel Processing

- Parallel encoding already used in AVC
 - FMO and slices for baseline profile
 - Slices or WPP may be used for other profiles
- There are two key benefits for including parallel processing in HEVC
 - Facilitating parallel decoding
 - Will be useful for low delay applications and for decoding UHD content
 - Parallel CABAC encoding
 - WPP implementations in AVC result in bin throughput spikes, and require buffering of encoded symbols



- WPP processing inefficiency
 - If picture height is not integer multiple of wavefront height
 - Top left and bottom right corners
 - Efficiency: number of CTBs in picture/number that could be processed in parallel by wavefronts
- Tiles inefficiency
 - Different size tiles
 - Different size tile groups when not processing all tiles in parallel
 - Efficiency: number of CTBs in smallest tile group/number of CTBs in largest tile group

Parallel Processing Efficiency

- Examined parallel processing for 2-8 cores
- In general, differences are not too significant
- Class E results are similar for levels of parallelism most likely to be used

Class	CTB Size	Average Efficiency		
		WPP	Tiles: LB	Tiles: CLB
A	64	91%	93%	93%
	32	95%	92%	92%
B	64	87%	83%	81%
	32	94%	79%	78%
E	64	88%	72%	72%
	32	92%	69%	69%

- Tiles constrain prediction meaning loss of efficiency with increasing numbers of tiles
- Both tiles and WPP require additional bits to support entry point indication
- Tile boundary artefacts: minor artefacts visible in some sequences
 - May be possible to mitigate using non-normative encoding algorithms
- Compression efficiency comparison using HM-6.1
 - Tile settings (see F335): one entry point per tile
 - Wavefronts settings: one wavefront substream for each line, one entry point
- Approx. 1% difference between tiles and WPP (in favour of WPP)
 - WPP incurs cost over anchors (AI: 0.2%, RA: 0.9%, LB: 2.2%)
- Conclusion: both tiles and WPP incur coding efficiency penalty, WPP incurs 1% less efficiency penalty than tiles

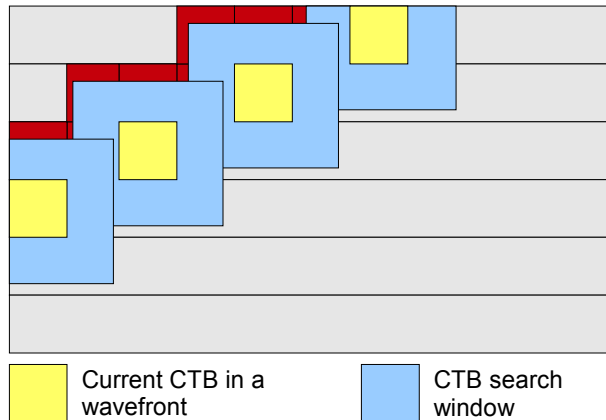
Compression Efficiency

■ Tiles vs. WPP

	Random Access Main			Random Access HE10		
	Y	U	V	Y	U	V
Class A	2.3%	2.1%	2.3%	2.1%	4.0%	4.2%
Class B	1.1%	1.0%	0.9%	1.0%	1.5%	0.8%
Class C	0.3%	0.5%	0.6%	0.3%	0.4%	0.6%
Class D	-0.9%	-0.7%	-0.7%	-0.9%	-0.7%	-0.9%
Class E						
Overall	0.7%	0.7%	0.8%	0.7%	1.3%	1.2%
	0.7%	0.7%	0.8%	0.7%	1.3%	1.2%
Overall (no Class D)	1.3%	1.2%	1.3%	1.1%	2.0%	1.9%
Class F	1.6%	1.7%	1.8%	1.5%	1.8%	1.4%
Enc Time[%]	-			-		
Dec Time[%]	-			-		

	All Intra Main			All Intra HE10		
	Y	U	V	Y	U	V
Class A	0.9%	-0.1%	1.0%	0.1%	-0.4%	0.1%
Class B	1.0%	0.6%	0.7%	0.5%	-1.5%	-2.0%
Class C	0.7%	0.6%	0.6%	0.3%	-1.8%	-2.5%
Class D	-0.2%	-0.1%	-0.1%	-0.1%	0.0%	-0.1%
Class E	1.9%	0.3%	0.6%	1.3%	-6.2%	-7.8%
Overall	0.8%	0.3%	0.6%	0.4%	-1.8%	-2.2%
	0.8%	0.3%	0.5%	0.4%	-1.7%	-2.1%
Overall (no Class D)	1.1%	0.3%	0.7%	0.6%	-2.5%	-3.0%
Class F	1.2%	0.9%	1.0%	1.1%	-0.8%	-0.7%
Enc Time[%]	-			-		
Dec Time[%]	-			-		

	Low delay B Main			Low delay B HE10		
	Y	U	V	Y	U	V
Class A						
Class B	0.9%	0.9%	0.6%	-0.1%	-2.1%	-2.9%
Class C	0.0%	0.0%	0.3%	-0.4%	-1.8%	-1.5%
Class D	-1.0%	-1.1%	-0.6%	-1.0%	-1.3%	-1.3%
Class E	0.5%	0.0%	-0.3%	0.3%	-5.5%	-5.5%
Overall	0.1%	0.0%	0.1%	-0.3%	-2.4%	-2.6%
	0.1%	-0.1%	0.1%	-0.3%	-2.3%	-2.5%
Overall (no Class D)	0.5%	0.3%	0.2%	-0.1%	-3.1%	-3.3%
Class F	-0.8%	-0.7%	-0.4%	1.1%	0.0%	0.3%
Enc Time[%]	-			-		
Dec Time[%]	-			-		

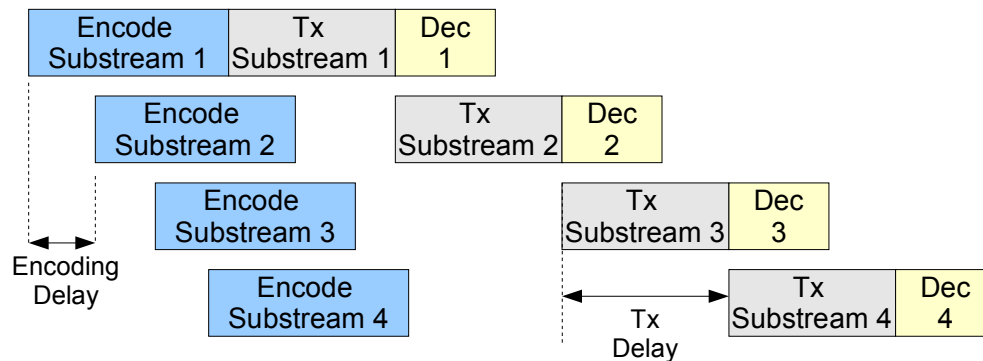


Parallelism	WPP (% of non-parallel)	
	128Kb	256Kb
1	100%	100%
2	54%	62%
3	37%	46%
4	36%	42%
5	36%	36%
6	38%	29%

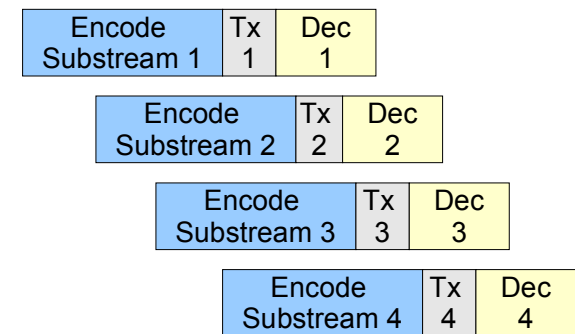
- Benefits for ME for tiles stated in JCTVC-F335
- WPP can also reduce ME bandwidth
 - Single cache or multiple caches can be used
- Cache simulations performed to demonstrate scenario for WPP
- Benefits for tiles also confirmed by simulations
- Avoided direct comparison as optimal tile and WPP ME approaches appear to be different
 - Implementation specific

Delay Issues

- Processing delay
 - Tiles: zero
 - WPP: $2(S_{wpp} - 1)P_{CTB}$
- Transmission delay
 - End-to-end time to send from encoder to decoder
- Delay may be transmission bound or processing delay bound



a) Transmission delay bounded scenario



b) Encode delay bounded scenario

Delay Bounds

- Tiles delay is always transmission delay bound
- WPP is transmission delay bound (e.g. no impact of wavefront processing delay) if
 - $T_{\text{line}} > 2P_{\text{CTB}}$
- WPP delay is only an issue if a line of CTBs can be transmitted in less time than it takes to encode/decode two CTBs

Class	Frame Rate	Parallel Streams	Parallel Proc For Low Delay		Parallel Proc for More Cycles	
			2P_CTB (ms)	WPP Proc. Delay (ms)	2P_CTB (ms)	WPP Proc. Delay (ms)
A	30	4	0.07	0.10	0.27	0.40
	60	4	0.03	0.05	0.13	0.20
B	30	4	0.13	0.20	0.52	0.78
	60	4	0.07	0.10	0.26	0.39
E	30	4	0.28	0.42	1.11	1.67
	60	4	0.14	0.21	0.56	0.83

■ Tiles

- Current CD text does not permit implementation of level conforming decoders
 - A two core decoder designed to decode 1080p cannot decode a 1080p bitstream encoded without tiles
 - Potential issues with mapping cores to tiles if tile sizes are not tightly constrained
- Solution part 1: choose a single value for tiles_or_entropy_coding_sync_idc
- Solution part 2: specify number of tiles, and shapes, for particular levels
 - Low number of tiles: 2, 4, 8 tied to particular levels
 - Good compression efficiency, poor flexibility
 - Flexibility issues do not make this attractive, as it will be less flexible for parallel encoders than AVC
 - Large number of tiles: as in JCTVC-F335
 - Poor compression efficiency
 - Reasonable flexibility
- Deciding on a solution for specifying tile sizes in the Main Profile is non-trivial

- Simple specifications
 - $\text{tiles_or_entropy_coding_sync_idc} = 2$
 - $\text{num_substreams_minus1} = \text{PicHeightInCtbs} - 1$
- High encoder and decoder flexibility for small compression efficiency loss

Conclusions and Recommendations

- For many issues there is little to choose between tiles and WPP
- There seem to be clear advantages for WPP in the areas of
 - Profile simplicity
 - Compression efficiency
- Based on technical merits, we rank the parallel processing schemes in the following order
 - tiles_or_entropy_coding_sync_idc = 2
 - tiles_or_entropy_coding_sync_idc = 0
 - tiles_or_entropy_coding_sync_idc = 1
- Tiles will incur either significant compression efficiency losses or decrease in parallel processing flexibility
 - More parallel processing flexibility allows more freedom to optimize for particular application requirements
 - We believe that it is more important to have flexibility for the encoder, than to have parallel decoding
 - Main Profile without parallel processing options is better than tiles, particularly for building scalable solutions
 - If tiles are disabled, then WPP can be implemented as currently done with AVC
 - Existing loose tiles specification makes building decoders more complex