

# **Syntax on Parallel Processing Information Signaling (JCTVC-H0293)**

---

**Yungo Park  
Seungsoo Jeong  
JaeHyun Kim  
JeongHoon Park  
(Samsung)**

# Background

- ❖ HEVC has many tools to support parallel processing in multi-core environments
  - Tiles, WPP, Slice, Entropy Slice
- ❖ Multi-core system is currently being widely used in almost CE devices
  - Parallel processing for video, especially in high resolution, is getting more important
- ❖ Early knowing the parallel processing information in SPS level would be helpful for the decoder to prepare its multi-core decoding strategy
  - SPS is generally used in negotiation for playing capability checking
- ❖ In HEVC, the information for parallel processing is dispersed in SPS, PPS and Slice Header

	Slice	Entropy Slice	Tile	Wavefront
SPS	×	×	○	×
PPS	×	×	○	○

# Proposed

- ❖ Signal one bit flag to notify whether the bitstream has a possibility for parallel processing or not
  - Under the condition, all existing syntax (Tile, WPP) are put together
  - Under the condition, the syntax which have slice and entropy slice information are added together
- ❖ All parallel processing information (or picture partition) could come together and could be signaled at one time

	Slice	Entropy Slice	Tile	Wavefront
SPS	○	○	○	○
PPS	○	○	○	○

# Proposed Syntax

## Syntax for SPS

seq_parameter_set_rbsp( ) {	<b>Descriptor</b>
...	
<b>parallel_processing_param_enabled_sps_flag</b>	u(1)
if( parallel_processing_param_enabled_sps_flag )	
parallel_processing_param( )	
...	
}	

## Syntax for PPS

pic_parameter_set_rbsp( ) {	<b>Descriptor</b>
...	
if( parallel_processing_param_enabled_sps_flag ) {	
<b>parallel_processing_param_enabled_pps_flag</b>	u(1)
if( parallel_processing_param_enabled_pps_flag )	
parallel_processing_param( )	
}	
...	
}	

## Syntax for parallel\_processing\_param()

parallel_processing_param( ) {	<b>Descriptor</b>
<b>tile_enabled_flag</b>	u(1)
if( tile_enabled_flag ) {	
<b>num_tile_columns_minus1</b>	ue(v)
<b>num_tile_rows_minus1</b>	ue(v)
<b>tile_boundary_independence_flag</b>	u(1)
<b>uniform_spacing_flag</b>	u(1)
if( !uniform_spacing_flag ) {	
for( i = 0; i < num_tile_columns_minus1; i++ )	
<b>column_width[i]</b>	ue(v)
for( i = 0; i < num_tile_rows_minus1; i++ )	
<b>row_height[i]</b>	ue(v)
}	
}	
<b>entropy_coding_synchro</b>	u(v)
if( entropy_coding_synchro )	
<b>num_substreams_minus1</b>	ue(v)
<b>slice_enabled_flag</b>	u(1)
if( slice_enabled_flag )	
<b>entropy_slice_enabled_flag</b>	u(1)
}	

Newly Defined Syntax

Existing but Moved Syntax

# Conclusion

- ❖ It is proposed that **one flag signaling** which notifies whether the bitstream has parallel processing (or picture partition) information **in SPS or PPS** or not
  - Early notification for parallel processing information
- ❖ It is proposed that **slice and entropy slice information** are early **notified in SPS and PPS**
  - In AVC, before parsing each slice header, it was **impossible** for the decoder to know whether this picture consists of slice or not
  - Early SPS or PPS level picture slicing information could be helpful for multi-core decoder
- ❖ It is proposed that **parallel processing related syntax come together in SPS and PPS**
  - More readable