

JVET-O0306

AHG17: Entry point offsets without start code emulation

K. Abe, T. Toma, V. Drugeon

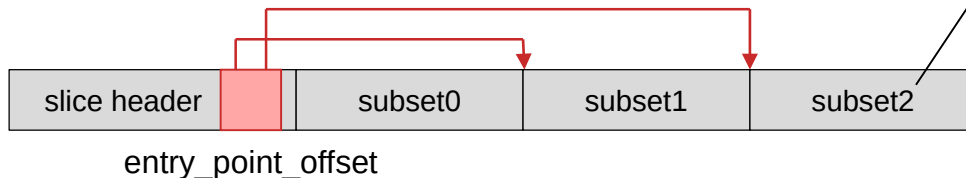
Panasonic Corporation

■ Current syntax of entry point offset

	Descriptor
slice_header() {	
....	
if (entropy_coding_sync_enabled_flag)	
num_entry_point_offsets	ue(v)
if(NumEntryPoints > 0) {	
offset_len_minus1	ue(v)
for(i = 0; i < NumEntryPoints; i++)	
entry_point_offset_minus1[i]	u(v)
}	
byte_alignment()	
}	

entry point offsets are used for:

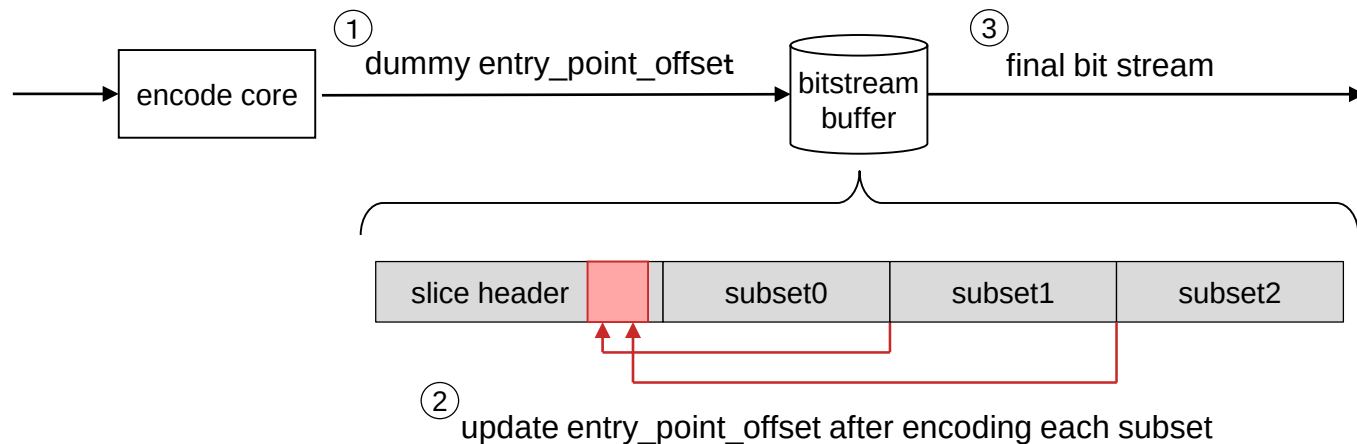
- Bricks (new in VVC)
- WPP (same as in HEVC)



An encoder needs to know the number of bits in each subset to encode slice header.

Problem

■ An example of real time encoder implementation



■ Problems

(Problem1) Need to update by bit unit operation since entry_point_offset field is not byte aligned.

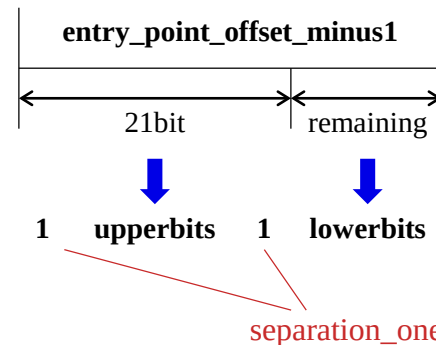
(Problem2) 0x03 (emulation prevention byte) may be inserted in the field and extend the field length, which would overlap and overwrite the data area of subset0.

■ Proposed entry point offset syntax

slice_header() {	Descriptor
....	
if (entropy_coding_sync_enabled_flag)	
num_entry_point_offsets	ue(v)
if(NumEntryPoints > 0) {	
offset_len_minus1	ue(v)
while(!byte_aligned())	
entry_point_alignment_zero_bit	f(1)
for(i = 0; i < NumEntryPoints; i++) {	
entry_point_separation_one_bit	f(1)
entry_point_offset_minus1_upperbits[i]	u(v)
if(offset_len_minus1 > 20) {	
entry_point_separation_one_bit	f(1)
entry_point_offset_minus1_lowerbits[i]	u(v)
}	
}	
byte_alignment()	
}	

(Item1) byte align before entry point offset field for easier updating during encoding.

(Item2) separate entry points by one bit and divide into upper and lower bits to avoid start code emulation without prevention byte:



Conclusion

2 items are proposed to solve entry point offset issues in real time encoders.

(Item1) byte align before entry point offset field.

(Item2) separate entry points by one bit and divide into upper and lower bits.

We propose to adopt the above two items into VVC.