



MEDIATEK

JVET-Q0173

AHG9: On signalling the DPB parameters with delta values

Authors: Chen-Yen Lai, Chia-Ming Tsai, Tzu-Der Chuang, Olena Chubach, Ching-Yeh Chen, Chih-Wei Hsu, Yu-Wen Huang, Shaw-Min Lei

Presenter: Chen-Yen Lai

Overall Summary

- DPB parameters are signalled from the lowest temporal sublayer to the highest one directly.
 - *max_dec_pic_buffering_minus1[]* and *max_num_reorder_pics[]*
 - Bitstream conformance requirement in the spec:
the values of these two syntax elements in the higher temporal sublayer shall be greater than or equal to those in the lower temporal sublayer.
- Propose to signal parameter delta values between successive temporal sublayers instead of signalling absolute values of all sublayers.
 - The bitstream conformance requirements can be removed.

DPB Syntax Structure in VVC Draft 7

dpb_parameters(dpbSizeOnlyFlag, maxSubLayersMinus1, subLayerInfoFlag) {	Descriptor
for(i = (subLayerInfoFlag ? 0 : maxSubLayersMinus1); i <= maxSubLayersMinus1; i++) {	
max_dec_pic_buffering_minus1[i] → Need a bitstream conformance requirement	
if(!dpbSizeOnlyFlag) {	
max_num_reorder_pics[i] → Need a bitstream conformance requirement (v)	
max_latency_increase_plus1[i]	ue(v)
}	
}	
}	

- Bitstream conformance requirements:
 - When i is greater than 0, *max_dec_pic_buffering_minus1[i]* shall be greater than or equal to *max_dec_pic_buffering_minus1[i - 1]*.
 - When i is greater than 0, *max_num_reorder_pics[i]* shall be greater than or equal to *max_num_reorder_pics[i - 1]*.

Signalled *max_dec_pic_buffering_minus1* and *max_num_reorder_pics* by Delta Values

	Descriptor
dpb_parameters(dpbSizeOnlyFlag, maxSubLayersMinus1, subLayerInfoFlag) {	
for(i = (subLayerInfoFlag ? 0 : maxSubLayersMinus1); i <= maxSubLayersMinus1; i++) {	
max_dec_pic_buffering_delta [i] 	Do not need bitstream conformance requirement
if(!dpbSizeOnlyFlag) {	
max_delta_num_reorder_pics [i] 	Do not need bitstream conformance requirement
max_latency_increase_plus1 [i]	ue(v)
}	
}	
}	

- Signal difference between two successive temporal sublayers instead of absolute value in each temporal sublayer
- **Remove** bitstream conformance requirements and reduce the number of coded bits



Thank You

Reconstruction of maxReqDpbSize[]

`max_dec_pic_buffering_minus1_delta[i]` specifies, for each CLVS of the CVS, the delta value between two successive sublayers to derive the maximum required size of the DPB in units of picture storage buffers, `maxReqDpbSize[i]`, when `Htid` is equal to `i`, as follows:

if (`i == 0`),

`maxReqDpbSize[0] = max_dec_pic_buffering_delta[0]`

otherwise,

`maxReqDpbSize[i] = ((subLayerInfoFlag) ? maxReqDpbSize[i-1] : 0) + max_dec_pic_buffering_delta[i]`

The value of `max_dec_pic_buffering_minus1_delta[i]` shall be in the range of 0 to `MaxDpbSize`, inclusive, where `MaxDpbSize` is as specified in clause A.4.2. When `i` is greater than 0, `max_dec_pic_buffering_minus1[i]` shall be greater than or equal to `max_dec_pic_buffering_minus1[i-1]`. When `max_dec_pic_buffering_minus1_delta[i]` is not present for `i` in the range of 0 to `maxSubLayersMinus1 - 1`, inclusive, due to `subLayerInfoFlag` being equal to 0, it is inferred to be equal to `0` `max_dec_pic_buffering_minus1[maxSubLayersMinus1]`. And `maxReqDpbSize[i]` for `i` in the range of 0 to `maxSubLayersMinus1 - 1`, inclusive, is inferred to be equal to `maxReqDpbSize[maxSubLayersMinus1]`

Reconstruction of maxReorderPics[]

max_delta_num_reorder_pics[i] specifies, for each CLVS of the CVS, the delta value between two successive sublayers used to derive the maximum allowed number of pictures of the CLVS that can precede any picture in the CLVS in decoding order and follow that picture in output order when Htid is equal to i as follows:

if (i == 0),

maxReorderPics [0] = **max_delta_num_reorder_pics** [0]

otherwise,

maxReorderPics [i] = ((subLayerInfoFlag) ? **maxReorderPics** [i-1]:0) + **max_delta_num_reorder_pics** [i]

The value of **max_delta_num_reorder_pics**[i] shall be in the range of 0 to **maxReqDpbSize**[i] - 1 **max_dec_pic_buffering_minus1**[i], inclusive. ~~When i is greater than 0, **max_num_reorder_pics**[i] shall be greater than or equal to **max_num_reorder_pics**[i - 1].~~ When **max_num_reorder_pics**[i] is not present for i in the range of 0 to **maxSubLayersMinus1** - 1, inclusive, due to subLayerInfoFlag being equal to 0, it is inferred to be equal to ~~0.~~ **max_num_reorder_pics**[**maxSubLayersMinus1**].—And **maxReorderPics** [i] for i in the range of 0 to **maxSubLayersMinus1** - 1, inclusive, is inferred to be equal to **maxReorderPics** [**maxSubLayersMinus1**]