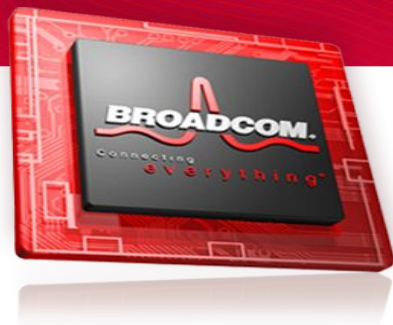


# AHG16/AHG17: Proposed Cleanup for Reshaper High Level Syntax (JVET-N0069)



Brian Heng, Minhua Zhou and Wade Wan

Broadcom Inc.


# Introduction

- In-loop reshaping adopted during the Marrakech meeting
- Asserted that two specific flag combinations result in undefined and/or not useful behavior
- This contribution proposes modifications in high level syntax to address these flag combinations including definition of a default model in the SPS
- The same concepts could be extended to use the APS instead of the SPS

## Current in-loop reshaper syntax

- **tile\_group\_reshaper\_model\_present\_flag** : This flag indicates whether or not a reshaping model is carried in this tile group.
- **tile\_group\_reshaper\_enable\_flag** : This flag indicates whether reshaping is enabled for this tile group or not.


```
if ( sps_reshaper_enabled_flag ) {
```



```
    tile_group_reshaper_model_present_flag
```

```
    if ( tile_group_reshaper_model_present_flag )
```

```
        tile_group_reshaper_model ( )
```



```
    tile_group_reshaper_enable_flag
```

```
    if ( tile_group_reshaper_enable_flag && !( qtbt_dual_tree_intra_flag && tile_group_type == I ) )
```

```
        tile_group_reshaper_chroma_residual_scale_flag
```

```
    }
```

## Combination #1: Reshaper Model Not Present

- Unclear what the expected behavior is when:
  - `tile_group_reshaper_model_present_flag=0`
  - `tile_group_reshaper_enable_flag=1`
- Issues
  - If this is signalled in the first picture, reshaper LUTs are invalid (note the VTM appears to fill with zeros)
  - If this is signalled at a random access point, use of a model signaled prior to the random access point breaks the concept of random access
  - In usage modes where a decoder decodes only a subset of pictures (e.g. scalability, trick modes), the concept of “previous picture” may not be available

## Combination #2: Reshaper Model Present But Not Enabled

- Unclear why the following combination is allowed:
  - `tile_group_reshaper_model_present_flag=1`
  - `tile_group_reshaper_enable_flag=0`
- Issue
  - Nothing broken but unclear why this combination would be useful

## Proposed Solution (1/2)

- Send a default reshaper model in the SPS (same syntax as currently defined in the tile group) which is used for Combination #1:
  - `tile_group_reshaper_model_present_flag=0`
  - `tile_group_reshaper_enable_flag=1`
- Allows reuse of a model across multiple frames and default can be overridden by sending another model
- Default model eliminates issues with propagating a “previous” model from frame to frame

<code>seq_parameter_set_rbsp() {</code>
<code>...</code>
<code><b>sps_reshaper_enabled_flag</b></code>
<code>if( sps_reshaper_enabled_flag ) {</code>
<code>    sps_default_reshaper_model ( )</code>
<code>}</code>
<code>...</code>
<code>}</code>

## Proposed Solution (2/2)

- Modify the tile group header syntax so “enable” flag is ahead of “present” flag
- This skips reshaper syntax if reshaper is not enabled for that tile group to avoid Combination #2
  - tile\_group\_reshaper\_model\_present\_flag=1
  - tile\_group\_reshaper\_enable\_flag=0

if( sps_reshaper_enabled_flag ) {
<b>tile_group_reshaper_enable_flag</b>
<b>if( tile_group_reshaper_enable_flag ) {</b>
<b>tile_group_reshaper_model_present_flag</b>
if(tile_group_reshaper_model_present_flag )
tile_group_reshaper_model ( )
if( !( qtbtt_dual_tree_intra_flag && tile_group_type == I ) )
<b>tile_group_reshaper_chroma_residual_scale_flag</b>
<b>}</b>
}

## Summary

- Recommend adopting the proposed modifications to reshaper HLS to address the discussed issues with the current syntax
- Alternatively, it is recognized that adaptation parameter sets (APS) could carry this information. The same concepts proposed in this contribution could be applied to APS signaling but were not explicitly suggested in this contribution.