# JCTVC-I0178: Simplified alpha bit-depth restriction in chroma LM mode

Xingyu Zhang, Oscar C. Au,
Feng Zou, Jingjing Dai, Chao Pang

香港科技大學
HONG KONG UNIVERSITY OF SCIENCE AND TECHNOLOGY

# Background

➢ LM mode: Using a linear model to predict Chroma from Luma.

$$Pred_C[x, y] = \alpha \cdot Rec_L{}'[x, y] + \beta$$

In order to use integer operations, the $\alpha$ is up-scaled by $2^{13}$ and clipped to the range $[-2^{15}, 2^{15}-1]$, named $\alpha'$.
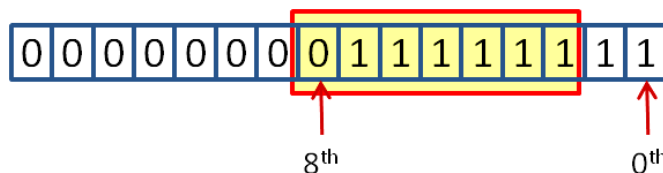
After the clipping operation, the precision of $\alpha'$ is reduced to 7 bits with a normalized factor *iShift*.

# Background

After clipping in the range $[-2^{15}, 2^{15}-1]$

$$\alpha' = 2^8 - 1$$
$$iShift = 13$$

| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

8th        0th

After bit-depth restriction

$$\alpha' = 2^6 - 1$$
$$iShift = 13 - 2 = 11$$

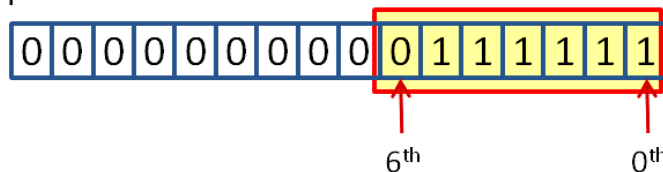| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |

6th        0th

Fig. 1 Example of alpha bit-depth restriction.

Thus, the actual prediction process using a lower precision $\alpha'$ is as

$$pred_C[x, y] = (\alpha' \cdot Rec_L'[x, y] >> iShift) + \beta'$$

# Proposed Simplification –Scheme 1

*If-else* conditional judgment can be removed if the last B bits of $\alpha'$ is set to 0, with B>=5.

```
iShift=13;
......
a3 = Clip3(-( 1 << 15 ), ( 1 << 15 ) - 1, a3);
if (a3 <= (2^6 -1) &&  a3 >= -2^6 )
 {
    // do nothing
 }
else
 {
    Short n = CountLeadingZerosOnes(a3);
    a3 = a3 >> (9-n);
    iShift - = (9-n);
 }
```

Bit-depth restriction process in HM6.0

```
iShift=13;
......
a3 = Clip3(-( 1 << 15 ), (1 << 15) - 1, a3);
a3= a3 & (~(2^6-1));
Short n = CountLeadingZerosOnes(a3);
a3 = a3 >> (9-n);
iShift - = (9-n);
```

Proposed Simplification with B=6

# Proposed Simplification –Scheme 2

*If-else* conditional judgment can be removed by:

1. $\alpha$ value is up-scaled by $2^{13-B}$ and clipped to the range $[-2^{15-B}, 2^{15-B}-1]$ to get $\alpha'$
2. After the clipping operation, $\alpha'$ is left shifted by B bits.

```
iShift=13;
……
a3 = Clip3(-( 1 << 15 ), ( 1 << 15 ) - 1, a3);
if (a3 <= (2^6 -1) &&  a3 >= -2^6 )
 {
   // do nothing
 }
else
  {
   Short n = CountLeadingZerosOnes(a3);
    a3 = a3 >> (9-n);
    iShift - = (9-n);
  }
```

```
iShift=7;
……
a3 = Clip3(-( 1 << 9 ), (1 << 9) - 1, a3);
a3=  a3<<6;
Short n = CountLeadingZerosOnes(a3);
a3 = a3 >> (9-n);
iShift - = (9-n);
```

Proposed Simplification with B=6

Bit-depth restriction process in HM6.0

# Simulation Results on HM6.0

Scheme 1 and Scheme 2 have the same results. (Thank Microsoft (JCTVC-I0412) and Intel (JCTVC-I0476) for crosschecking)

| Test (Class A-E) | Intra-Main + LMChroma=1 | | | Intra-Main + LMChroma=1 | | |
|---|---|---|---|---|---|---|
| | Y BD-Rate (%) | U BD-Rate (%) | V BD-Rate (%) | Y BD-Rate (%) | U BD-Rate (%) | V BD-Rate (%) |
| B=5 | 0.00% | -0.21% | -0.18% | 0.00% | -0.24% | -0.24% |
| B=6 | 0.00% | -0.46% | -0.39% | 0.00% | -0.58% | -0.60% |
| B=7 | 0.00% | -0.88% | -0.76% | 0.00% | -0.98% | -0.96% |
| B=8 | 0.00% | -1.18% | -1.02% | 0.03% | -1.33% | -1.19% |

| Test (Class A-E) (10bit sequences excluded) | Intra-HE-10bit | | | Intra-HE-10bit | | |
|---|---|---|---|---|---|---|
| | Y BD-Rate (%) | U BD-Rate (%) | V BD-Rate (%) | Y BD-Rate (%) | U BD-Rate (%) | V BD-Rate (%) |
| B=5 | 0.00% | -0.05% | -0.01% | 0.00% | -0.06% | 0.00% |
| B=6 | 0.00% | -0.07% | 0.00% | 0.00% | -0.12% | 0.00% |
| B=7 | 0.00% | -0.15% | 0.02% | 0.00% | -0.18% | -0.01% |
| B=8 | 0.00% | -0.15% | 0.12% | 0.02% | -0.19% | 0.11% |

# Thank you