I.7.3.8.5.2 Coding unit extension syntax

|  |  |
| --- | --- |
| cu\_extension( x0 , y0 , log2CbSize ) { | **Descriptor** |
| … |  |
| for( j = 0; j < nCbS; j = j + pbOffset ) |  |
| for( k = 0; k < nCbS; k = k + pbOffset ) |  |
| if( DmmFlag[ x0 + k ][ y0 + j ] | | sdc\_flag[ x0 ][ y0 ] ) { |  |
| if( CuPredMode[ x0 ][ y0 ] = = MODE\_INTRA ) { |  |
| **depth\_dc\_flag**[ x0 + k ][ y0 + j ] | ae(v) |
| dcNumSeg = DmmFlag[ x0 + k ][ y0 + j ] ? 2 : 1 |  |
| } else{ |  |
| **mul\_seg\_ flag** [ x0 + k ][ y0 + j ] | ae(v) |
| if(**mul\_seg\_ flag** [ x0 + k ][ y0 + j ]){ |  |
| **mul**\_**seg\_ num\_minus1**[ x0 + k ][ y0 + j ] | ae(v) |
| dcNumSeg = nSegNum[ x0 + k ][ y0 + j ] |  |
| }else |  |
| dcNumSeg = 1 |  |
| } |  |
| … |  |
| } |  |

I.7.4.9.5.2 Coding unit extension semantics

…

**depth\_dc\_flag**[ x0 ][ y0 ] equal to 1 specifies that depth\_dc\_abs[ x0 ][ y0 ][ i ] and depth\_dc\_sign\_flag[ x0 ][ y0 ][ i ] are present. depth\_dc\_flag[ x0 ][ y0 ] equal to 0 specifies that depth\_dc\_abs[ x0 ][ y0 ][ i ] and depth\_dc\_sign\_flag[ x0 ][ y0 ][ i ] are not present. When not present, depth\_dc\_flag[ x0 ][ y0 ] is inferred to be equal to 1.

**mul\_seg\_sdc\_flag**[ x0 ][ y0 ] equal to 1 specifies that multiple segmental prediction for inter segment-wise DC coding is applied. mul\_seg\_sdc\_flag[ x0 ][ y0 ] equal to 0 specifies that multiple segmental prediction for inter segment-wise DC coding is not applied. When not present, mul\_seg\_sdc\_flag[ x0 ][ y0 ] is inferred to be equal to 0.

**mul**\_**seg\_num\_minus1**[ x0 ][ y0 ] minus 1 specifies the number of segments when multiple segmental prediction for inter segment-wise DC coding is applied. When not present, mul\_seg\_num\_minus1 [ x0 ][ y0 ] is inferred to be equal to 0. A variable nSegNum[ x0 + k ][ y0 + j ] is derived as

nSegNum[ x0 ][ y0 ] = mul\_seg\_sdc\_flag[ x0 ][ y0 ] ? ( mul\_seg\_num\_minus1 [ x0 ][ y0 ] + 1 ) : 1.

I.8.5.3.3.5 Full sample interpolation process

…

* The prediction luma sample value predSampleLXL[ xL ][ yL ] is derived as specified in the following:
  1. predSampleLXL[ xL ][ yL ] = refPicLXL[ xIntL ][ yIntL ] (I‑191)

When sdc\_flag is equal to 1 and mul\_seg\_ flag[ xCb ][ yCb ] is equal to 1, the following applies in order:

1. A variable predAvg is derived as follows:

predAvg =,

where k = Log2( nPbW)\*(nPbH ).

1. A variable predMin is set equal to the minimum value for all predSampleLXL [ x ][ y ] with x = 0,… nPbW-1, y = 0, … nPbH-1. A variable predMax is set equal to the maximum value for all predSampleLXL[ x ][ y ] with x = 0,… nPbW-1, y = 0, … nPbH-1.
2. If nSegNum[ xCb ][ yCb ] is equal to 1, a variable segPred[ 0 ] is set equalt to predAvg. A variable segIdx[ x ][ y ] is set equal to 0 for all x = 0,… nPbW-1, y = 0, … nPbH-1.
3. Otherwise, if nSegNum[ xCb ][ yCb ] is equal to 2, the following applies for all x = 0,… nPbW-1, y = 0, … nPbH-1:
   * If predSampleLXL[ x ][ y ] < predAvg, segIdx[ x ][ y ] is set equal to 0;
   * Otherwise (predSampleLXL[ x ][ y ] >= predAvg), segIdx[ x ][ y ] is set equal to 1.
4. Otherwise (nSegNum[ xCb ][ yCb ] is equal to 3), a variable thres0 is set equal to (predAvg + predMin + 1 ) >> 1, and a variable thres1 is set equal to (predAvg + predMax + 1 ) >> 1. Then the following applies for all x = 0,… nPbW-1, y = 0, … nPbH-1:
   * If predSampleLXL[ x ][ y ] < thres0, segIdx[ x ][ y ] is set equal to 0;
   * Othewise, if predSampleLXL[ x ][ y ] > thres1, segIdx[ x ][ y ] is set equal to 2;
   * Otherwise, segIdx[ x ][ y ] is set equal to 1.
5. When nSegNum[ xCb ][ yCb ] is greater than 1, the following applies for j from 0 to nSegNum[ xCb ][ yCb ] - 1:
   * A variable segMin[ j ] is set equal to the minimum value for all predSampleLXL [ x ][ y ] with x = 0,… nPbW-1, y = 0, … nPbH-1satisfying segIdx[ x ][ y ] is equal to j. A variable segMax[ j ] is set equal to the maximum value for all predSampleLXL[ x ][ y ] with x = 0,… nPbW-1, y = 0, … nPbH-1 satisfying segIdx[ x ][ y ] is equal to j.
   * segPred[ j ] is set equal to (segMin[ j ] + segMax[ j ]+ 1)>>1.
6. If DltFlag[ nuh\_layer\_id ] is equal to 0, for all x = 0,… nPbW-1, y = 0, … nPbH-1, predSampleLXL[ x ][ y ] is set equal to Clip3( 0, (1 << BitDepthY )- 1, segPred[segIdx[ x ][ y ]] + DcOffset[ xCb ][ yCb ][ segIdx[ x ][ y ] ]. Otherwise, (DltFlag[ nuh\_layer\_id ] is equal to 1), for all x = 0,… nPbW-1, y = 0, … nPbH-1, predSampleLXL[ x ][ y ] is set equal to Idx2DepthValue[ DepthValue2Idx( segPred[segIdx[ x ][ y ]]) + DcOffset[ xCb ][ yCb ][ segIdx[ x ][ y ] ]].

I.8.5.4.1 General

…

Otherwise (sdc\_flag is equal to 1), if mul\_seg\_ flag[ xCb ][ yCb ] is equal to 0, for x in the range of 0 to nCbSL − 1 and y in the range of 0 to nCbSL − 1, resSamplesL[ x ][ y ] is set equal to DcOffset[ xCb ][ yCb ][ 0 ]. Otherwise ( mul\_seg\_ flag[ xCb ][ yCb ] is equal to 1 ), for x in the range of 0 to nCbSL − 1 and y in the range of 0 to nCbSL − 1, resSamplesL[ x ][ y ] is set equal to 0.

Table I‑ – Association of ctxIdx and syntax elements for each initializationType in the initialization process

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Syntax structure** | **Syntax element** | **ctxTable** | **initType** | | |
| **0** | **1** | **2** |
| ... |  |  |  |  |
| dim\_not\_present\_flag |  | 0 | 1 | 2 |
| mul\_seg\_sdc\_flag | Table I-21’ |  | 0 | 1 |
| mul\_seg\_num\_minus1 | Table I-22’ |  | 0, 1 | 2,3 |

Table ‑21’ – Values of initValue for mul\_seg\_sdc\_flag and mul\_seg\_type ctxIdx

|  |  |  |
| --- | --- | --- |
| **Initialization variable** | **ctxIdx of** mul\_seg\_sdc\_flag | |
| **0** | **1** |
| **initValue** | 154 | 154 |

Table ‑22’ – Values of initValue for mul\_seg\_type ctxIdx

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Initialization variable** | **ctxIdx of** mul\_seg\_type | | | |
| **0** | **1** | **2** | **3** |
| **initValue** | 154 | 154 | 154 | 154 |

Table I‑ – Syntax elements and associated binarizations

| **Syntax structure** | **Syntax element** | **Binarization** | |
| --- | --- | --- | --- |
| **Process** | **Input parameters** |
| cu\_extension( ) | … | … | … |
| sdc\_flag | FL | cMax = 1 |
| mul\_seg\_sdc\_flag | FL | cMax = 1 |
| mul\_seg\_num\_minus1 | TR | cMax = 3, cRiceParam = 0 |

Table I‑ –Assignment of ctxInc to syntax elements with context coded bins

| **Syntax element** | **binIdx** | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| **0** | **1** | **2** | **3** | **4** | **>=5** |
| ... | .. | .. | … | … | … | … |
| sdc\_flag | 0 | na | na | na | na | na |
| mul\_seg\_sdc\_flag | 0 | na | na | na | na | na |
| mul\_seg\_num\_minus1 | 0 | 1 | na | na | na | na |
| dim\_not\_present\_flag | 0 | na | na | na | na | na |