# Draft Text Specification

The proposed text changes are based on the document JCTVC-M1005-v2.doc. The changes are marked in yellow.

**7.3.8.11 Residual coding syntax**

|  |  |
| --- | --- |
| residual\_coding( x0, y0, log2TrafoSize, cIdx ) { | **Descriptor** |
| … |  |
| for( n = 15; n >= 0; n− − ) { |  |
| xC = ( xS << 2 ) + ScanOrder[ 2 ][ scanIdx ][ n ][ 0 ] |  |
| yC = ( yS << 2 ) + ScanOrder[ 2 ][ scanIdx ][ n ][ 1 ] |  |
| if( sig\_coeff\_flag[ xC ][ yC ] ) { |  |
| if( numGreater1Flag < 8 ) { |  |
| **coeff\_abs\_level\_greater1\_flag**[ n ] | ae(v) |
| numGreater1Flag++ |  |
| if( coeff\_abs\_level\_greater1\_flag[ n ] && lastGreater1ScanPos = = −1 ) |  |
| lastGreater1ScanPos = n |  |
| } |  |
| if( lastSigScanPos = = −1 ) |  |
| lastSigScanPos = n |  |
| firstSigScanPos = n |  |
| } |  |
| } |  |
| signHidden = ( lastSigScanPos − firstSigScanPos > 3 && !cu\_transquant\_bypass\_flag ) |  |
| if ( cu\_transform\_skip\_flag && ( CuPredMode[ x0 ][ y0 ] == MODE\_INTRA ) && ( predModeIntra == INTRA\_DC ) |  |
| signHidden = 0 |  |
| if( lastGreater1ScanPos != −1 ) |  |
| … |  |

**8.4.4.2.1 General intra sample prediction**

Inputs to this process are:

– a sample location ( xTbCmp, yTbCmp ) specifying the top-left sample of the current transform block relative to the top‑left sample of the current picture,

– a variable predModeIntra specifying the intra prediction mode,

– a variable nTbS specifying the transform block size,

– a variable cIdx specifying the colour component of the current block.

– a variable cIdx specifying the colour component of the current block~~.~~,

* the residual sample array values r[ x ][ y ], with x, y = 0..nTbS−1.

**8.4.4.2.5 Specification of intra prediction mode INTRA\_DC**

Inputs to this process are:

– the neighbouring samples p[ x ][ y ], with x = −1, y = −1..nTbS \* 2 − 1 and x = 0..nTbS \* 2 − 1, y = −1,

– a variable nTbS specifying the transform block size,

– a variable cIdx specifying the colour component of the current block.

Outputs of this process are the predicted samples predSamples[ x ][ y ], with x, y = 0..nTbS − 1.

If cu\_transquant\_bypass\_flag is equal to 0 and cu\_transform\_skip\_flag is equal to 0, the ~~The~~ values of the prediction samples predSamples[ x ][ y ], with x, y = 0..nTbS − 1, are derived by the following ordered steps:

1. A variable dcVal is derived as follows:

dcVal =  (8‑41)

where k = Log2( nTbS ).

1. Depending on the value of the colour component index cIdx, the following applies:

* If cIdx is equal to 0 and nTbS is less than 32, the following applies:

predSamples[ 0 ][ 0 ] = ( p[ −1 ][ 0 ] + 2 \* dcVal + p[ 0 ][ −1 ] + 2 )  >>  2 (8‑42)

predSamples[ x ][ 0 ] = ( p[ x ][ −1 ] + 3 \* dcVal + 2 )  >>  2, with x = 1..nTbS − 1 (8‑43)

predSamples[ 0 ][ y ] = ( p[ −1 ][ y ] + 3 \* dcVal + 2 )  >>  2, with y = 1..nTbS − 1 (8‑44)

predSamples[ x ][ y ] = dcVal, with x, y = 1..nTbS − 1 (8‑45)

* Otherwise, the prediction samples predSamples[ x ][ y ] are derived as follows:

predSamples[ x ][ y ] = dcVal, with x, y = 0..nTbS − 1 (8‑46)

Otherwise (if cu\_transquant\_bypass\_flag is equal to 1 or cu\_transform\_skip\_flag is equal to 1) the values of the prediction samples predSamples[ x ][ y ], with x, y = 0..nTbS − 1, are derived as follows:

– predSamples[x][y] = .

The neighboring samples p[ x ][ y ], with x, y = 0..nTbS − 1, are reconstructed by

– p[x][y] = r[x][y] + predSamples[x][y].