H.8.4.3 Depth value reconstruction process

Inputs to this process are:

* a luma location ( xB, yB ) specifying the top-left luma sample of the current block relative to the top-left luma sample of the current picture,
* a variable nT specifying the prediction size
* predicted samples predSamples[ x ][ y ], with x, y =0..nT−1
* a variable intraPredMode specifying the prediction mode of the current prediction block

Output of this process is:

* reconstructed depth value samples resSamples[ x ][ y ], with x, y = −1..2\*nT−1.

The values of the prediction samples predSamples[ x ][ y ], are derived as specified by the following ordered steps.

* 1. Depending on intraPredMode the array wedgePattern[ x ][ y ] with x, y =0..nT−1 specifying the binary segmentation pattern is derived as follows.
     + If intraPredMode is equal to Intra\_DepthPartition( 35 ), the following applies.
       - 1. wedgePattern = WedgePatternTable[ Log2( nT) ][ wedge\_full\_tab\_idx[ xB ][ yB ] ]
     + Otherwise ( intraPredMode is not equal to Intra\_DepthPartition( 35 ) ), the following applies.
       - For x, y = 0..nT−1 wedgePattern[ x ][ y ] is set equal to 0.
  2. The variable log2SubSample is set equal to ( nT < 32 ) ? 0 : 1.
  3. For p in the range of 0 to 1, inclusive, the variable dcPred[ p ] is derived as specified in the following:

sumPred = 0  
 numPred = 0.   
 for( x = 0; x < ( nT >> log2SubSample ); x++ ) {   
 xS  = x << log2SubSample    
 for ( y = 0; y < ( nT >> log2SubSample ) ;y++ ) {   
 yS = y << log2SubSample .  
 if ( p = = wedgePattern[ xS ][ yS ] ) {  
 sumPred += predSamples[ xS ][ yS ]  
 numPred += 1  
 }  
 }  
 dcPred[ p ] = ( numPred > 0 ) ? ( sumPred / numPred ) : 0

[ Ed. (GT): Is this averaging necessary for DMM or DC mode? ]

* 1. For x, y = 0..nT−1, the reconstructed depth value samples resSamples[ x ][ y ] are derived as specified in the following.
     + If dlt\_flag[ nuh\_layer\_id ] is equal to 0, the following applies:
       - 1. ~~resSamples[ x ][ y ] = dcPred[ wedgePattern[ x ][ y ] ] +   
             SdcResidual[ xB ][ yB ][~~~~wedgePattern[ x ][ y ] ]~~
         2. resSamples[ x ][ y ] = predSamples[ x ][ y ] + SdcResidual[ xB ][ yB ][wedgePattern[ x ][ y ] ]
     + Otherwise ( dlt\_flag[ nuh\_layer\_id ] is equal to 1 ), the following applies.
       - 1. dltIdxPred = DepthValue2Idx[ dcPred[ wedgePattern[ x ][ y ] ] ] (H‑65)  
            dltIdxResi = SdcResidual[ xB ][ yB ][wedgePattern[ x ][ y ] ] (H‑66)  
            ~~resSamples[ x ][ y ] = Idx2DepthValue[ dltIdxPred + dltIdxResi ] (‑67)~~
         2. resSamples[ x ][ y ] = predSamples[ x ][ y ] + (Idx2DepthValue[ dltIdxPred + dltIdxResi ] - dcPred[ wedgePattern[ x ][ y ] ]) (‑67)