#### 9.3.3.9 Binarization process for coeff\_abs\_level\_remaining

Input to this process is a request for a binarization for the syntax element coeff\_abs\_level\_remaining[ n ], the current sub-block scan index i, baseLevel, the colour component cIdx, the array statCoeff and the value of transform\_skip\_flag ~~and the luma location ( x0, y0 ) specifying the top-left sample of the current luma transform block relative to the top-left luma sample of the picture~~.

Output of this process is the binarization of the syntax element.

~~The variables cLastAbsLevel and cLastRiceParam are derived as follows:~~

* ~~If one or more of the following conditions are true, cLastAbsLevel and cLastRiceParam are set equal to 0:~~
* ~~fast\_rice\_adaptation\_enabled\_flag is equal to 0 and this process is invoked for the first time for the current sub-block scan index i,~~
* ~~this process is invoked for the first time for the current transform block.~~
* ~~Otherwise, cLastAbsLevel and cLastRiceParam are set equal to the values of cAbsLevel and cRiceParam, respectively, that have been derived during the last invocation of the binarization process for the syntax element coeff\_abs\_level\_remaining as specified in this subclause.~~

~~The variable cAbsLevel is set equal to baseLevel + coeff\_abs\_level\_remaining[ n ].~~

~~The variable cRiceParam is derived from cLastAbsLevel and cLastRiceParam as follows:~~

* ~~If fast\_rice\_adaptation\_enabled\_flag is equal to 0, the following applies:~~

~~cRiceParam = Min( cLastRiceParam + ( cLastAbsLevel > ( 3 \* ( 1  <<  cLastRiceParam ) ) ? 1 : 0 ), 4 ) (9‑13)~~

* ~~Otherwise (fast\_rice\_adaptation\_enabled\_flag is equal to 1), the following applies:~~
* ~~cRiceParam is initialized as follows:~~

~~cRiceParam = Min( cLastRiceParam + ( cLastAbsLevel  >>  ( 2  +  cLastRiceParam ) ), 7 ) (9‑13)~~

* ~~When this process is invoked for the first time for the current sub-block scan index i, cRiceParam is modified as follows:~~

~~cRiceParam = Max( 0, cRiceParam −  (9‑13)  
 ( transform\_skip\_flag[ x0 ][ y0 ][ cIdx ] | | cu\_transquant\_bypass\_flag ? 1 : 2 ) )~~

The variable sbType is derived as follows:

sbType =2\*( cIdx == 0 ? 1 : 0 ) + transform\_skip\_flag (9‑xx)

The initRiceValue is derived as follows:

initRiceValue = Min (statCoeff [sbType]/4, MaxRicePara)

The variables cLastAbsLevel and cLastRiceParam are derived as follows:

* If this process is invoked for the first time for the current sub-block scan index i, cLastAbsLevel is set equal to 0 and cLastRiceParam is set equal to initRiceValue.
* Otherwise (this process is not invoked for the first time for the current sub-block scan index i), cLastAbsLevel and cLastRiceParam are set equal to the values of cAbsLevel and cRiceParam, respectively, that have been derived during the last invocation of the binarization process for the syntax element coeff\_abs\_level\_remaining as specified in this subclause.

The variable cAbsLevel is set equal to baseLevel + coeff\_abs\_level\_remaining[ n ].

* If this process is invoked for the first time for the current sub-block scan index i, statCoeff[ sbType ] is set to
  + If coeff\_abs\_level\_remaining[ n ] >= 3\* (1<< (statCoeff[ sbType]/4))

statCoeff[ sbType]++

* + Otherwise, if 2\*coeff\_abs\_level\_remaining[ n ] < (1<< (statCoeff[ sbType]/4))

statCoeff[ sbType] −−

And a clipping is applied to statCoeff[ sbType]

statCoeff[ sbType] = Clip3(0, 4\* 7−1, statCoeff[ sbType])

The variable cRiceParam is derived from cLastAbsLevel and cLastRiceParam as:

cRiceParam = Min( cLastRiceParam + ( cLastAbsLevel > ( 3 \* ( 1  <<  cLastRiceParam ) ) ? 1 : 0 ),  7) (9‑13)

The variable cMax is derived from cRiceParam as:

cMax = 4  <<  cRiceParam (9‑14)

The binarization of the syntax element coeff\_abs\_level\_remaining[ n ] is a concatenation of a prefix bin string and (when present) a suffix bin string.

For the derivation of the prefix bin string, the following applies:

* The prefix value of cu\_qp\_delta\_abs, prefixVal, is derived as follows:

prefixVal = Min( cMax, coeff\_abs\_level\_remaining[ n ] ) (9‑15)

* The prefix bin string is specified by invoking the TR binarization process as specified in subclause 9.3.3.2 for prefixVal with the variables cMax and cRiceParam as inputs.

When the prefix bin string is equal to the bit string of length 4 with all bits equal to 1, the suffix bin string is present and it is derived as follows:

* The suffix value of cu\_qp\_delta\_abs, suffixVal, is derived as follows:

suffixVal = coeff\_abs\_level\_remaining[ n ] − cMax (9‑16)

* The suffix bin string is specified by invoking the EGk binarization process as specified in subclause 9.3.3.3 for suffixVal with the Exp-Golomb order k set equal to cRiceParam + 1.

## *9.3 CABAC parsing process for slice segment data*

**9.3.1 General**

**…**

The storage process for context variables is applied as follows:

* + When ending the parsing of the coding tree unit syntax in subclause 7.3.8.2, entropy\_coding\_sync\_enabled\_flag is equal to 1 and CtbAddrInRs % PicWidthInCtbsY is equal to 1, the storage process for context variables as specified in subclause 9.3.2.3 is invoked with TableStateIdxWpp and TableMpsValWpp as outputs. Similarly, the entries of the Rice parameter initialization counter statCoeff[k], k in the range 0 to 3, inclusive, are stored in TableStatCoeffWpp.
  + When ending the parsing of the general slice segment data syntax in subclause 7.3.8.1, dependent\_slice\_segments\_enabled\_flag is equal to 1 and end\_of\_slice\_segment\_flag is equal to 1, the storage process for context variables as specified in subclause 9.3.2.3 is invoked with TableStateIdxDs and TableMpsValDs as outputs. Similarly, the entries Rice parameter initialization counter statCoeff[k], k in the range 0 to 3, inclusive, are stored in TableStatCoeffDs.

….

### 9.3.2 Initialization process

#### 9.3.2.1 General

Outputs of this process are initialized CABAC internal variables and Rice parameter initialization counter statCoeff.

The context variables of the arithmetic decoding engine are initialized as follows:

* + If the coding tree unit is the first coding tree unit in a tile, the initialization process for context variables is invoked as specified in subclause 9.3.2.2 and statCoeff[k] is set to 0 for k in the range 0 to 3, inclusive.
  + Otherwise, if entropy\_coding\_sync\_enabled\_flag is equal to 1 and CtbAddrInRs % PicWidthInCtbsY is equal to 0, the following applies:
  + The location ( xNbT, yNbT ) of the top-left luma sample of the spatial neighbouring block T (Figure 9‑2) is derived using the location ( x0, y0 ) of the top-left luma sample of the current coding tree block as follows:

( xNbT, yNbT ) = ( x0 + CtbSizeY, y0 − CtbSizeY ) (9‑3)

* + The availability derivation process for a block in z-scan order as specified in subclause 6.4.1 is invoked with the location ( xCurr, yCurr ) set equal to ( x0, y0 ) and the neighbouring location ( xNbY, yNbY ) set equal to ( xNbT, yNbT ) as inputs, and the output is assigned to availableFlagT.
  + The synchronization process for context variables is invoked as follows:
  + If availableFlagT is equal to 1, the synchronization process for context variables as specified in subclause 9.3.2.4 is invoked with TableStateIdxWpp and TableMpsValWpp as inputs. The entries in the Rice parameter initialization counter statCoeff[k], k in the range 0 to 3, inclusive, are set to those from the table TableStatCoeffWpp.
  + Otherwise, the initialization process for context variables is invoked as specified in subclause 9.3.2.2.
  + Otherwise, if CtbAddrInRs is equal to slice\_segment\_address and dependent\_slice\_segment\_flag is equal to 1, the synchronization process for context variables as specified in subclause 9.3.2.4 is invoked with TableStateIdxDs and TableMpsValDs as inputs. The entries in the Rice parameter initialization counter statCoeff[k], k in the range 0 to 3, inclusive, are set to those from the table TableStatCoeffDs.
  + Otherwise, the initialization process for context variables is invoked as specified in subclause 9.3.2.2 statCoeff[k] is set to 0 for k in the range 0 to 3, inclusive.

...