###### Derivation process of ctxIdxInc for the syntax element significant\_coeff\_flag

Inputs to this process are the color component index cIdx, the current coefficient scan position ( xC , yC ) and the transform block size log2TrafoSize, and scanIdx.

Output of this process is ctxIdxInc.

The variable sigCtx depends on the current position ( xC, yC ), the transform block size and previsously decoded bins of the syntax element significant\_coeff\_flag. For the derivation of sigCtx, the following applies.

* If log2TrafoSize is less than or equal to 3, sigCtx is derived as follows.

shift = log2TrafoSize = = 3 ? 1 : 0  
sigCtx = ( shift \* 15 ) + ( ( yC >> shift ) << 2 ) + ( xC >> shift ) (9‑55)

* Otherwise if xC + yC is less than 2, sigCtx is derived as follows.

sigCtx = 31 + ( yC << 1 ) + xC

When the scanIdx is neither 1 nor 2

* If xC + yC is less than 5, sigCtx is derived as follows.

temp = significant\_coeff\_flag[ xC + 1 ][ yC ] + significant\_coeff\_flag[ xC + 2 ][ yC ] +   
 significant\_coeff\_flag[ xC ][ yC + 1 ] + significant\_coeff\_flag[ xC + 1 ][ yC + 1 ] +   
 significant\_coeff\_flag[ xC ][ yC + 2 ]

* Otherwise if (xC + yC is greater than 4), sigCtx is derived using previously decoded bins of the syntax element significant\_coeff\_flag as follows.

The variable sigCtx is initialized as follows.

sigCtx = 39

* When xC is less than ( 1 << log2TrafoSize ) − 1, the following applies.

sigCtx = sigCtx + significant\_coeff\_flag[ xC + 1 ][ yC ] (9‑59)

* When xC and yC are less than ( 1 << log2TrafoSize ) − 1, the following applies.

sigCtx = sigCtx + significant\_coeff\_flag[ xC + 1 ][ yC + 1 ] (9‑60)

* When xC is less than ( 1 << log2TrafoSize ) − 2, the following applies.

sigCtx = sigCtx + significant\_coeff\_flag[ xC + 2 ][ yC ] (9‑61)

* When yC is less than ( 1 << log2TrafoSize ) − 1, the following applies.

sigCtx = sigCtx + significant\_coeff\_flag[ xC ][ yC + 1 ] (9‑62)

* When yC is less than ( 1 << log2TrafoSize ) − 2 and sigCtx is less than 43, the following applies.

sigCtx = sigCtx + significant\_coeff\_flag[ xC ][ yC + 2 ]

- When the scanIdx is 1 or 2

(9‑56)

* The variable sigCtxA, sigCtxB, sigCtxC, sigCtxD and sigCtxE are initialized as follows.
* sigCtxA, sigCtxB, sigCtxC, sigCtxD, and sigCtxE are set to 0
* When the scanIdx is 1 and the position is available in the transform block
  + sigCtxA = significant\_coeff\_flag[ xC ][ yC + 1]
  + sigCtxB = significant\_coeff\_flag[ xC - 1 ][ yC + 1 ]
  + sigCtxC = significant\_coeff\_flag[ xC + 2 ][ yC ]
  + sigCtxD = significant\_coeff\_flag[ xC ][ yC + 2]
  + sigCtxE = significant\_coeff\_flag[ xC + 1][ yC + 1]
* When the scanIdx is 2 and the position is available in the transform block
  + sigCtxA = significant\_coeff\_flag[ xC + 1 ][ yC ]
  + sigCtxB = significant\_coeff\_flag[ xC + 1 ][ yC - 1 ]
  + sigCtxC = significant\_coeff\_flag[ xC ][ yC + 2]
  + sigCtxD = significant\_coeff\_flag[ xC +2][ yC]
  + sigCtxE = significant\_coeff\_flag[ xC + 1 ][ yC + 1]
  + swap xC and yC coordinates
* If xC + yC is less than 5, sigCtx is derived as follows.
  + When the scanIdx is 1 or 2

temp = sigCtxA + sigCtxC + sigCtxD + sigCtxE

When xC is not 0,

temp = temp + sigCtxB

(9‑57)

sigCtx = 34 + Min( 4, temp )

* Otherwise (xC + yC is greater than 4), sigCtx is derived using previously decoded bins of the syntax element significant\_coeff\_flag as follows.
* The variable sigCtx is initialized as follows.

sigCtx = 39 (9‑58) (9‑63)

* When the scanIdx is 1 or 2
* When xC and yC are less than ( 1 << log2TrafoSize ) − 1, the following applies.

sigCtx = sigCtx + sigCtxE (9‑60)

* When xC is less than ( 1 << log2TrafoSize ) − 2, the following applies.

sigCtx = sigCtx + sigCtxC (9‑61)

* When yC is less than ( 1 << log2TrafoSize ) − 1, the following applies.

sigCtx = sigCtx + sigCtxA (9‑62)

* When yC is less than ( 1 << log2TrafoSize ) – 1 and xC is greater than 0, the following applies.
  + sigCtx = sigCtx + sigCtxB (9‑62)
* When yC is less than ( 1 << log2TrafoSize ) − 2 and sigCtx is less than 43, the following applies.

sigCtx = sigCtx + sigCtxD

The context index increment ctxIdxInc is derived using the color component index cIdx and sigCtx as follows.

* If cIdx is equal to 0, ctxIdxInc is derived as follows.

ctxIdxInc = sigCtx (9‑64)

* Otherwise (cIdx is greater than 0), ctxIdxInc is derived as follows.

ctxIdxInc = 44 + sigCtx (9‑65)

[Ed. (BB): The context derivation assumes maximum transform sizes less than or equal to 32x32 for luma and 16x16 for chroma and minimum transform sizes greater than or equal to 4x4.]