#### **7.3.4.2 Sample adaptive offset CABAC syntax**

|  |  |
| --- | --- |
| sao\_offset\_cabac( rx, ry, cIdx ) { | Descriptor |
| **sao\_type\_idx**[ cIdx ][ rx ][ ry ] | ae(v) |
| if( sao\_type\_idx[ cIdx ][ rx ][ ry ] = =5 ) { |  |
| **sao\_band\_position**[ cIdx ][ rx ][ ry ] | ae(v) |
| for( i = 0; i < 4; i++ ) |  |
| **sao\_offset\_sign** [ cIdx ][ rx ][ ry ][ i ] | ae(v) |
| } |  |
| if( sao\_type\_idx[ cIdx ][ rx ][ ry ] != 0 ) |  |
| for( i = 0; i < 4; i++ ) |  |
| **sao\_****offset**[ cIdx ][ rx][ ry ][ i ] | ae(v) |
| } |  |

#### **Sample adaptive offset unit CABAC semantics**

**sao\_offset\_sign**[ cIdx ][ rx ][ ry ][ i ]specifies sign value of sao\_offset when sao\_type\_index equal to 5.

The variable offsetSign is derived as follows.

* If sao\_type\_idx[ cIdx ][ rx ][ ry ] is smaller than 5 and i is larger than 1,offsetSign is set to −1.
* Otherwise if sao\_type\_idx[ cIdx ][ rx ][ ry ] is smaller than 5 and i is smaller than 2,offsetSign is set to 1.
* Otherwise (sao\_type\_idx[ cIdx ][ rx ][ ry ] equals 5), offsetSign is set to sao\_offset\_sign[ cIdx ][ rx ][ ry ][ i ].

The array SaoOffsetVal is derived as follows.

SaoOffsetVal[ cIdx ][ rx ][ ry ][ 0 ] = 0

SaoOffsetVal[ cIdx ][ rx ][ ry ][ i + 1 ] =   
 offsetSign\*sao\_offset[ cIdx ][ rx ][ ry ][ i ] << ( bitDepth – Min( bitDepth, 10 ) )

Table ‑ – Values of variable initValue for sao\_offset ctxIdx

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Initialization variable** | **sao\_offset ctxIdx** | | | | | |
| **0** | **1** | **2** | **3** | **4** | **5** |
| **initValue** | 143 | 140 | 185 | 140 | 200 | 140 |

| Table 9‑32 – Syntax elements and associated types of binarization, maxBinIdxCtx, ctxIdxTable, and ctxIdxOffset | | | | | |
| --- | --- | --- | --- | --- | --- |
| **Syntax element** | **initialisationType** | **Type of binarization** | **maxBinIdxCtx** | **ctxIdxTable** | **ctxIdxOffset** |
| sao\_merge\_left\_flag | 0 | FL, cMax = 1 | 0 |  | 0 |
| 1 | 0 |  | 3 |
| 2 | 0 |  | 6 |
| sao\_merge\_up\_flag | 0 | FL, cMax = 1 | 0 |  | 0 |
| 1 | 0 |  | 1 |
| 2 | 0 |  | 2 |
| sao\_type\_idx | 0 | U | 1 |  | 0 |
| 1 | 1 |  | 2 |
| 2 | 1 |  | 4 |
| sao\_band\_position | 0 | FL, cMax = 5 | na | na | na, (uses Decode Bypass) |
| 1 | na | na | na, (uses Decode Bypass) |
| 2 | na | na | na, (uses Decode Bypass) |
| sao\_offset\_sign | 0 | FL, cMax = 1 | na | na | na, (uses Decode Bypass) |
| 1 | na | na | na, (uses Decode Bypass) |
| 2 | na | na | na, (uses Decode Bypass) |
| sao\_offset | 0 | U | 1 |  | 0 |
| 1 | 1 |  | 2 |
| 2 | 1 |  | 4 |

| Table 9‑37 – Assignment of ctxIdxInc to binIdx for all ctxIdxTable and ctxIdxOffset values | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Syntax element** | **ctxIdxTable,  ctxIdxOffset** | | **binIdx** | | | | |
| **0** | **1** | **2** | **3** | **>=4** |
| sao\_merge\_left\_flag |  | 0 | cIdx | na | na | na | na |
| 3 | cIdx | na | na | na | na |
| 6 | cIdx | na | na | na | na |
| sao\_merge\_up\_flag |  | 0 | 0 | na | na | na | na |
| 1 | 0 | na | na | na | na |
| 2 | 0 | na | na | na | na |
| sao\_type\_idx |  | 0 | 0 | 1 | 1 | 1 | 1 |
| 2 | 0 | 1 | 1 | 1 | 1 |
| 4 | 0 | 1 | 1 | na | na |
| sao\_offset |  | 0 | 0 | 1 | 1 | 1 | 1 |
| 2 | 0 | 1 | 1 | 1 | 1 |
| 4 | 0 | 1 | 1 | 1 | 1 |