# Proposed Working Draft Text Modifications

**Working Draft Text Modifications for Method 1a**

### 8.3.1 Derivation process for luma intra prediction mode

Table 8‑2 specifies the number of luma intra prediction modes intraPredModeNum depending on log2PUSize.

Table 8‑2 – Specification of intraPredModeNum

|  |  |
| --- | --- |
| **log2PUSize** | **intraPredModeNum** |
| 2 | ~~18~~ 19 |
| 3 | 35 |
| 4 | 35 |
| 5 | 35 |
| 6 | 35 |

1. If candIntraPredModeA is equal to candIntraPredModeB, the candIntraPredModeA and candIntraPredModeC is derived as follows:

* If candIntraPredModeA is not equal to Intra\_Planar, candIntraPredModeA is set equal to Intra\_Planar
* Otherwise, candIntraPredModeA is set equal to Intra\_DC
* candIntraPredModeC is set as either Intra\_DC or Intra\_Vertical (mode 1) which is not used by candIntraPredModeA and candIntraPredModeB
* The candModeList[x] is derived as follows:
  + candModeList[0] = candIntraPredModeB
  + candModeList[1] = candIntraPredModeA
  + candModeList[2] = candIntraPredModeC

1. If candIntraPredModeA is not equal to candIntraPredModeB, the candIntraPredModeC is derived as follows:

* candIntraPredModeC is selected from Intra\_Planar, Intra\_DC or Intra\_Vertical (mode 1) which is not used by candIntraPredModeA and candIntraPredModeB
* The candModeList[x] is derived as follows:
  + candModeList[0] = Min( candIntraPredModeA, candIntraPredModeB )
  + candModeList[1] = Max( candIntraPredModeA, candIntraPredModeB )
  + candModeList[2] = candIntraPredModeC

1. IntraPredMode[ xB ][ yB ] is derived by applying the following procedure:

* If prev\_intra\_pred\_flag[ xB ][ yB ] is true, the IntraPredMode[ xB ][ yB ] is set equal to candModeList[ mpm\_flag ][ xB ][ yB ]]. The binarization process for mpm\_flag is specified as following table.

|  |  |
| --- | --- |
| mpm\_flag | bin\_string |
| 0 | 1 |
| 1 | 00 |
| 2 | 01 |

* Otherwise IntraPredMode[ xB ][ yB ] is derived by applying the following ordered steps:
  + IntraPredMode[ xB ][ yB ] = rem\_intra\_luma\_pred\_mode
  + Rank candModeList[x] such that candModeList[0] has the Min value and candModeList[2] has the Max value
  + When IntraPredMode[ xB ][ yB ] is equal or greater than candModeList[ 0 ], the value of IntraPredMode[ xB ][ yB ] is increased by one
  + When IntraPredMode[ xB ][ yB ] is equal or greater than candModeList[ 1 ], the value of IntraPredMode[ xB ][ yB ] is increased by one
  + When IntraPredMode[ xB ][ yB ] is equal or greater than candModeList[ 2 ], the value of IntraPredMode[ xB ][ yB ] is increased by one

**Binarization process for rem\_intra\_luma\_pred\_mode**

Table 9‑34 – Binarization for rem\_intra\_luma\_pred\_mode



**Working Draft Text Modifications for Method 1b**

### 8.3.1 Derivation process for luma intra prediction mode

Table 8‑2 specifies the number of luma intra prediction modes intraPredModeNum depending on log2PUSize.

Table 8‑2 – Specification of intraPredModeNum

|  |  |
| --- | --- |
| **log2PUSize** | **intraPredModeNum** |
| 2 | ~~18~~ 19 |
| 3 | 35 |
| 4 | 35 |
| 5 | 35 |
| 6 | 35 |

1. If candIntraPredModeA is equal to candIntraPredModeB, the candIntraPredModeB and candIntraPredModeC is derived as following two tables. The candModeList[x] is derived as follows:
   * candModeList[0] = candIntraPredModeA
   * candModeList[1] = candIntraPredModeB
   * candModeList[2] = candIntraPredModeC

**Table xx.** **candIntraPredModeB, candIntraPredModeC selection for 4x4 PU**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ModeA** | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** | **17** | **18** |
| ModeB | 3 | 12 | 16 | 0 | 11 | 11 | 13 | 14 | 15 | 17 | 18 | 4 | 5 | 1 | 6 | 4 | 8 | 2 | 9 |
| ModeC | 1 | 13 | 17 | 1 | 15 | 12 | 14 | 0 | 16 | 18 | 0 | 5 | 1 | 6 | 7 | 8 | 2 | 9 | 10 |

**Table xx. candIntraPredModeB, candIntraPredModeC selection for 8x8, 16x16, 32x32 PU**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **ModeA** | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** | **17** |
| ModeB | 3 | 22 | 30 | 0 | 19 | 20 | 24 | 26 | 28 | 32 | 34 | 19 | 21 | 23 | 25 | 27 | 29 | 31 |
| ModeC | 1 | 23 | 31 | 1 | 27 | 21 | 25 | 14 | 29 | 33 | 18 | 20 | 22 | 24 | 26 | 28 | 30 | 32 |
| **ModeA** | **18** | **19** | **20** | **21** | **22** | **23** | **24** | **25** | **26** | **27** | **28** | **29** | **30** | **31** | **32** | **33** | **34** |  |
| ModeB | 33 | 4 | 11 | 5 | 12 | 1 | 13 | 6 | 14 | 4 | 15 | 8 | 16 | 2 | 17 | 9 | 18 |  |
| ModeC | 34 | 11 | 5 | 12 | 1 | 13 | 6 | 14 | 7 | 15 | 8 | 16 | 2 | 17 | 9 | 18 | 10 |  |

1. If candIntraPredModeA is not equal to candIntraPredModeB, the candIntraPredModeC is derived as follows:

* candIntraPredModeC is selected from Intra\_Planar, Intra\_DC or Intra\_Vertical (mode 1) which is not used by candIntraPredModeA and candIntraPredModeB
* The candModeList[x] is derived as follows:
  + candModeList[0] = Min( candIntraPredModeA, candIntraPredModeB )
  + candModeList[1] = Max( candIntraPredModeA, candIntraPredModeB )
  + candModeList[2] = candIntraPredModeC

1. IntraPredMode[ xB ][ yB ] is derived by applying the following procedure:

* If prev\_intra\_pred\_flag[ xB ][ yB ] is true, the IntraPredMode[ xB ][ yB ] is set equal to candModeList[ mpm\_flag ][ xB ][ yB ]]. The binarization of mpm\_flag is specified as following table.

|  |  |
| --- | --- |
| mpm\_flag | bin\_string |
| 0 | 1 |
| 1 | 00 |
| 2 | 01 |

* Otherwise IntraPredMode[ xB ][ yB ] is derived by applying the following ordered steps:
  + IntraPredMode[ xB ][ yB ] = rem\_intra\_luma\_pred\_mode
  + Rank candModeList[x] such that candModeList[0] has the Min value and candModeList[2] has the Max value
  + When IntraPredMode[ xB ][ yB ] is equal or greater than candModeList[ 0 ], the value of IntraPredMode[ xB ][ yB ] is increased by one
  + When IntraPredMode[ xB ][ yB ] is equal or greater than candModeList[ 1 ], the value of IntraPredMode[ xB ][ yB ] is increased by one
  + When IntraPredMode[ xB ][ yB ] is equal or greater than candModeList[ 2 ], the value of IntraPredMode[ xB ][ yB ] is increased by one

.

**Binarization process for rem\_intra\_luma\_pred\_mode**

Table 9‑34 – Binarization for rem\_intra\_luma\_pred\_mode

|  |  |  |
| --- | --- | --- |
| **rem\_intra\_luma\_pred\_mode** | **log2PUSize** | **Bin string** |
| ~~less than 32~~ | 2 | FL, cMax = 4 |
| 3...6 | FL, cMax = 5 |
| ~~32~~ | ~~2...6~~ | ~~111110~~ |
| ~~33~~ | ~~2...6~~ | ~~111111~~ |

**Working Draft Text Modifications for Method 2**

### 8.3.1 Derivation process for luma intra prediction mode

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 log2PUSize specifying the size of the current prediction unit,

– variable arrays IntraPredMode (If available) that are previously (in decoding order) derived for adjacent coding units.

Output of this process is the variable IntraPredMode[ xB ][ yB ].

Table 8‑2 specifies the number of luma intra prediction modes intraPredModeNum depending on log2PUSize.

Table 8‑2 – Specification of intraPredModeNum

|  |  |
| --- | --- |
| **log2PUSize** | **intraPredModeNum** |
| 2 | 18 |
| 3 | 34 |
| 4 | 34 |
| 5 | 34 |
| 6 | 34 |

Table 8-7 specifies the mapping table between intraPredMode and the rearranged intra prediction order intraPredOrder.

Table 8‑7 – Specification of intraPredOrder

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **intraPredMode** | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** | **17** |
| **intraPredOrder** | - | - | - | - | 1 | 5 | 13 | 17 | 21 | 29 | 33 | 3 | 7 | 11 | 15 | 19 | 23 | 27 |
| **intraPredMode** | **18** | **19** | **20** | **21** | **22** | **23** | **24** | **25** | **26** | **27** | **28** | **29** | **30** | **31** | **32** | **33** | ~~34~~ |  |
| **intraPredOrder** | 31 | 2 | 4 | 6 | 8 | 10 | 12 | 14 | 16 | 18 | 20 | 22 | 24 | 26 | 28 | 30 | ~~32~~ |  |

**Binarization process for rem\_intra\_luma\_pred\_mode**

Input to this process is a request for the syntax element rem\_intra\_luma\_pred\_mode and cNumBins.

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

The binarization for rem\_intra\_luma\_pred\_mode is given by Table 9-33.

Table 9‑34 – Binarization for rem\_intra\_luma\_pred\_mode

|  |  |  |
| --- | --- | --- |
| **rem\_intra\_luma\_pred\_mode** | **log2PUSize** | **Bin string** |
| ~~less than 32~~ | 2 | FL, cMax = 4 |
| 3...6 | FL, cMax = 5 |
| 32 | ~~2...6~~ | ~~111110~~ |
| ~~33~~ | ~~2...6~~ | ~~111111~~ |