WHAT IS CLAIMED IS:

1. A method of video encoding, comprising:

determining whether a current block included in a current picture is to be coded in an intra block copy (IBC) non-merge mode;

when the current block is to be coded in the IBC non-merge mode, determining whether a maximum number of IBC merge candidates MaxNumIBCMergeCand is greater than 1, wherein MaxNumIBCMergeCand is to be encoded in a coded video bitstream;

when the current block is to be coded in the IBC non-merge mode and the MaxNumIBCMergeCand is greater than 1, encoding, in the coded video bitstream corresponding to the current block, a motion vector predictor index mvp_l0_flag for predicting a block vector associated with the current block in the IBC non-merge mode;

when the current block is to be coded in the IBC non-merge mode and the MaxNumIBCMergeCand is not greater than 1, the motion vector predictor index mvp_l0_flag is not encoded in the coded video bitstream; and

generating the coded video bitstream including at least the current block and the MaxNumIBCMergeCand.

2. The method according to claim 1, wherein when the MaxNumIBCMergeCand is not greater than 1, the motion vector predictor index mvp_l0_flag is implied as 0.

3. The method according to claim 1, wherein the current block is to be coded in an IBC Advanced Motion Vector Prediction (AMVP) mode.

4. The method according to claim 1, wherein signaling of the motion vector predictor index mvp_l0_flag for the IBC non-merge mode in the generated coded video bitstream is not dependent on a maximum number of candidates MaxNumMergeCand.

5. The method according to claim 4, wherein the motion vector predictor index mvp_l0_flag specifies the motion vector predictor index of list 0.

6. The method according to claim 1, further comprising:

determining the maximum number of IBC merge candidates MaxNumIBCMergeCand by processing circuitry of a video encoder; and

indicating the maximum number of IBC merge candidates MaxNumIBCMergeCand by including a syntax element in the generated coded video bitstream.

7. A method of video decoding, comprising:

receiving a coded video bitstream including a current picture;

determining whether a current block included in the current picture is encoded in an intra block copy (IBC) non-merge mode;

when the current block is encoded in the IBC non-merge mode, determining whether a maximum number of IBC merge candidates MaxNumIBCMergeCand is greater than 1, wherein the MaxNumIBCMergeCand is to be decoded from the coded video bitstream;

when the current block is encoded in the IBC non-merge mode and the MaxNumIBCMergeCand is greater than 1, decoding, from the coded video bitstream, a motion vector predictor index mvp_l0_flag for predicting a block vector associated with the current block in the IBC non-merge mode;

when the current block is encoded in the IBC non-merge mode and the MaxNumIBCMergeCand is not greater than 1, the motion vector predictor index mvp_l0_flag is not in the coded video bitstream; and

reconstructing the current block based on the MaxNumIBCMergeCand.

8. The method according to claim 7, wherein when the MaxNumIBCMergeCand is not greater than 1, the motion vector predictor index mvp_l0_flag is inferred as 0.

9. The method according to claim 7, wherein the current block is encoded in an IBC Advanced Motion Vector Prediction (AMVP) mode.

10. The method according to claim 7, wherein signaling of the motion vector predictor index mvp_l0_flag for the IBC non-merge mode in the coded video bitstream is not dependent on a maximum number of candidates MaxNumMergeCand.

11. The method according to claim 10, wherein the motion vector predictor index mvp_l0_flag specifies the motion vector predictor index of list 0.

12. The method according to claim 7, further comprising:

determining the maximum number of IBC merge candidates MaxNumIBCMergeCand based on a syntax element in the coded video bitstream.

13. An apparatus for performing video encoding, comprising:

processing circuitry configured to:

determine whether a current block included in a current picture is to be coded in an intra block copy (IBC) non-merge mode;

when the current block is to be coded in the IBC non-merge mode, determine whether a maximum number of IBC merge candidates MaxNumIBCMergeCand is greater than 1, wherein MaxNumIBCMergeCand is to be encoded in a coded video bitstream;

when the current block is to be coded in the IBC non-merge mode and the MaxNumIBCMergeCand is greater than 1, encoding, in the coded video bitstream corresponding to the current block, a motion vector predictor index mvp_l0_flag for predicting a block vector associated with the current block in the IBC non-merge mode;

when the current block is to be coded in the IBC non-merge mode and the MaxNumIBCMergeCand is not greater than 1, the motion vector predictor index mvp_l0_flag is not encoded in the coded video bitstream; and

generate the coded video bitstream including at least the current block and the MaxNumIBCMergeCand.

14. An apparatus for performing video decoding, comprising:

processing circuitry configured to:

receive a coded video bitstream including a current picture;

determine whether a current block included in the current picture is encoded in an intra block copy (IBC) non-merge mode;

when the current block is encoded in the IBC non-merge mode, determine whether a maximum number of IBC merge candidates MaxNumIBCMergeCand is greater than 1, wherein the MaxNumIBCMergeCand is to be decoded from the coded video bitstream;

when the current block is encoded in the IBC non-merge mode and the MaxNumIBCMergeCand is greater than 1, decode, from the coded video bitstream, a motion vector predictor index mvp_l0_flag for predicting a block vector associated with the current block in the IBC non-merge mode;

when the current block is encoded in the IBC non-merge mode and the MaxNumIBCMergeCand is not greater than 1, the motion vector predictor index mvp_l0_flag is not in the coded video bitstream; and

reconstruct the current block based on the MaxNumIBCMergeCand.

15. An apparatus for performing video encoding, comprising:

a memory, configured to store computer-executable instructions; and

a processor, when executing the computer-executable instructions stored in the memory, configured to cause the apparatus to implement the method of any one of claims 1 to 6.

16. An apparatus for performing video decoding, comprising:

a memory, configured to store computer-executable instructions; and

a processor, when executing the computer-executable instructions stored in the memory, configured to cause the apparatus to implement the method of any one of claims 7 to 12.

17. A computer readable medium storing a bitstream that can be encoded by a method of video encoding, the method comprising:

determining whether a current block included in a current picture is to be coded in an intra block copy (IBC) non-merge mode;

when the current block is to be coded in the IBC non-merge mode, determining whether a maximum number of IBC merge candidates MaxNumIBCMergeCand is greater than 1, wherein MaxNumIBCMergeCand is to be encoded in a coded video bitstream;

when the current block is to be coded in the IBC non-merge mode and the MaxNumIBCMergeCand is greater than 1, encoding, in the coded video bitstream

corresponding to the current block, a motion vector predictor index mvp_l0_flag for predicting a block vector associated with the current block in the IBC non-merge mode;

when the current block is to be coded in the IBC non-merge mode and the MaxNumIBCMergeCand is not greater than 1, the motion vector predictor index mvp_l0_flag is not encoded in the coded video bitstream; and

generating the coded video bitstream including at least the current block and the MaxNumIBCMergeCand.

18. A non-transitory computer-readable medium having instructions stored therein, which when executed by a processor in a video encoder causes the processor to execute the method of any one of claims 1 to 6.

19. A non-transitory computer-readable medium having instructions stored therein, which when executed by a processor in a video decoder causes the processor to execute the method of any one of claims 7 to 12.