

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

AMAZON.COM, INC. and AMAZON.COM SERVICES LLC,

Petitioners

v.

INTERDIGITAL, INC.,

Patent Owner.

IPR2026-00193

U.S. Patent No. 11,936,876

**PETITION FOR *INTER PARTES* REVIEW
OF U.S. PATENT NO. 11,936,876**

Mail Stop PATENT BOARD
Patent Trial and Appeal Board
U.S. Patent & Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450

TABLE OF CONTENTS

I.		GROUND FOR STANDING.....	1
II.		IDENTIFICATION OF CHALLENGE	2
A.	Prior Art.....	2	
B.	Grounds for Challenge	2	
III.	'876 PATENT OVERVIEW	3	
A.	Technology Background	3	
B.	Purported Invention.....	7	
IV.	LEVEL OF ORDINARY SKILL	10	
V.	CLAIM CONSTRUCTION	11	
VI.	DETAILED EXPLANATION OF GROUNDS.....	12	
A.	Ground 1A: Xiong in view of VCEG-AJ21.....	12	
1.	Xiong.....	12	
2.	VCEG-AJ21	19	
3.	Motivation to Combine Xiong and VCEG-AJ21	20	
4.	Claim 1	22	
5.	Claim 2: The method of claim 1, further comprising decoding an intra prediction mode for the basic coding unit sized sub-block.	35	
6.	Claim 3: The method of claim 1, further comprising decoding a further binary split signaling syntax element specifying that at least one of the four equally sized sub-blocks greater than a basic coding unit size is further split into four smaller equally sized sub-blocks.	36	
7.	Claim 4: The method of claim 1, wherein the spatial intra partition type specifies a size of a partition of the basic coding unit sized sub-block.	36	
8.	Claim 5	37	

9.	Claim 6: <i>The apparatus of claim 5, wherein the one or more processors are configured to decode an intra prediction mode for the basic coding unit sized sub-block.</i>	40
10.	Claim 7: <i>The apparatus of claim 5, wherein the one or more processors are configured to decode a further binary split signaling syntax element specifying that at least one of the four equally sized sub-blocks greater than a basic coding unit size is further split into four smaller equally sized sub-blocks.....</i>	41
11.	Claim 8: <i>The apparatus of claim 5, wherein the spatial intra partition type specifies a size of a partition of the basic coding unit.</i>	41
12.	Claim 11	41
13.	Claim 12: <i>The method of claim 11, further comprising encoding an intra prediction mode for the basic coding unit sized sub-block.</i>	44
14.	Claim 13: <i>The method of claim 11, further comprising encoding a further binary split signaling syntax element specifying that at least one of the four equally sized sub-blocks greater than a basic coding unit size is further split into four smaller equally sized sub-blocks.</i>	45
15.	Claim 14: <i>The method of claim 11, wherein the spatial intra partition type specifies a size of a partition of the basic coding unit sized sub-block.</i>	45
16.	Claim 15	46
17.	Claim 16: <i>The apparatus of claim 15, wherein, the one or more processors are configured to encode an intra prediction mode for the basic coding unit sized sub-block.</i>	49
18.	Claim 17: <i>The apparatus of claim 15, wherein, the one or more processors are configured to encode a further binary split signaling syntax element specifying that at least one of the four equally sized</i>	

	<i>sub-blocks greater than a basic coding unit size is further split into four smaller equally sized sub-blocks.....</i>	49
19.	<i>Claim 18: The method of claim 15, wherein the spatial intra partition type specifies a size of a partition of the basic coding unit sized sub-block.</i>	49
B.	Ground 1B: Xiong in view of VCEG-AJ21 and H.264	50
1.	H.264.....	50
2.	Motivation to Combine Xiong, VCEG-AJ21, and H.264	53
3.	<i>Claim 9: The apparatus of claim 5, wherein at least one of a spatial intra partition type table and an intra prediction mode table is pre-stored and used by the one or more processors to decode picture data for the large block.</i>	55
4.	<i>Claim 10: The apparatus of claim 5, wherein at least one of a spatial intra partition type table and an intra prediction mode table is received by the one or more processors using one or more high level syntax elements and is used by the one or more processors to decode picture data for the large block.....</i>	57
VII.	CONCLUSION.....	60
VIII.	MANDATORY NOTICES	60
A.	Real Party in Interest.....	60
B.	Related Matters.....	60
C.	Notice of Counsel and Service Information.....	60
D.	Power of Attorney	61

EXHIBIT LIST

No.	Exhibit
1001	U.S. Patent No. 11,936,876 (“’876 patent”)
1002	File History of the ’876 patent
1003	Declaration of Dr. Scott T. Acton, Ph.D.
1004	Curriculum Vitae of Dr. Scott T. Acton, Ph.D.
1005	WO2008/083633 (“Xiong”)
1006	Certified English Translation of WO2008/083633
1007	European Patent No. EP2081388 (“Xiong EP”)
1008	Enlarging MB size for high fidelity video coding beyond HD (“VCEG-AJ21”), available at https://www.itu.int/wftp3/av-arch/video-site/0810_San/VCEG-AJ21.zip
1009	M. Faundez-Zanuy, F. Vallverdú_Bayes, F. Tarrés-Ruiz “Contributions to interframe coding.” Workshop on image analysis & synthesis in image coding. October 1994.
1010	ITU-T Recommendation H.264 (November 2007) (“H.264”)
1011	Editing State of Text Relating to ITU-T Rec. H.264 ISO/IEC 14496-10 Amendments 1 and 2 (“ITU-T H.264 Draft”)
1012	Declaration of Dr. Clifford Reader (“Reader”)
1013	Declaration of June Ann Munford (“Munford”)
1014	“The H.264/AVC Advanced Video Coding Standard: Overview and Introduction to the Fidelity Range Extensions,” Gary J. Sullivan, et al. (August 2004) (“Sullivan1”)
1015	ITU-T Recommendation H.264 (March 2005) (“H.264 (03/2005)”)

No.	Exhibit
1016	Recommendation H.264 (2005) Amendment 1 (06/06) (Munford, Appendix H26402) (“Amendment 1”)
1017	Recommendation H.264 (2005) Amendment 2 (04/07) (Munford, Appendix H26403) (“Amendment 2”)
1018	“The H.264/MPEG-4 Advanced Video Coding (AVC) Standard,” Gary J. Sullivan, Ph.D (July 2005) (“Sullivan2”), available at https://www.itu.int/ITU-T/worksem/vica/docs/presentations/S3_P1_Sullivan.pdf

LISTING OF CHALLENGED CLAIMS

Reference	Limitation
Claim 1	
1[pre]	A method of decoding picture data for at least a large block in a picture, the method comprising:
1[a]	determining that intra prediction is to be performed for the large block, wherein the large block has a block size equal to or larger than 32×32 , and the large block size is greater than a size of a basic coding unit;
1[b]	decoding a binary split signaling syntax element specifying that the large block is further split into four equally sized sub-blocks;
1[c]	decoding an intra prediction mode for the large block that is not further split into four equally sized sub-blocks;
1[d]	partitioning the large block into the four equally sized sub-blocks in response to the decoded binary split signaling syntax element specifying that the large block is further split into four equally sized sub-blocks; and
1[e]	decoding a second syntax element indicating a single spatial intra partition type for at least one basic coding unit sized sub-block, the single spatial intra partition type being determinable from among a plurality of spatial intra partition types.
Claim 2	
2	The method of claim 1, further comprising decoding an intra prediction mode for the basic coding unit sized sub-block.
Claim 3	
3	The method of claim 1, further comprising decoding a further binary split signaling syntax element specifying that at least one of the four equally sized sub-blocks greater than a basic coding unit size is further split into four smaller equally sized sub-blocks.

Reference	Limitation
Claim 4	
4	The method of claim 1, wherein the spatial intra partition type specifies a size of a partition of the basic coding unit sized sub-block.
Claim 5	
5[pre]	An apparatus, comprising:
5[a]	at least a memory and one or more processors, wherein the one or more processors are configured to:
5[b]	decode picture data for at least a large block in a picture by determining that intra prediction is to be performed for the large block, wherein the large block has a large block size equal to or larger than 32×32 , and the large block size is greater than a basic coding unit size, wherein the intra prediction is signaled for the large block and the one or more processors are configured to:
5[c]	decode a binary split signaling syntax element specifying that the large block is further split into four equally sized sub-blocks;
5[d]	decode an intra prediction mode for the large block that is not further split into four equally sized sub-blocks;
5[e]	partition the large block into the four equally sized sub-blocks in response to the decoded binary split signaling syntax element specifying that the large block is further split into four equally sized sub-blocks; and
5[f]	decode a second syntax element indicating a single spatial intra partition type for at least one basic coding unit sized sub-block, the single spatial intra partition type being determinable from among a plurality of spatial intra partition types.

Reference	Limitation
Claim 6	
6	The apparatus of claim 5, wherein the one or more processors are configured to decode an intra prediction mode for the basic coding unit sized sub-block.
Claim 7	
7	The apparatus of claim 5, wherein the one or more processors are configured to decode a further binary split signaling syntax element specifying that at least one of the four equally sized sub-blocks greater than a basic coding unit size is further split into four smaller equally sized sub-blocks.
Claim 8	
8	The apparatus of claim 5, wherein the spatial intra partition type specifies a size of a partition of the basic coding unit.
Claim 9	
9	The apparatus of claim 5, wherein at least one of a spatial intra partition type table and an intra prediction mode table is pre-stored and used by the one or more processors to decode picture data for the large block.
Claim 10	
10	The apparatus of claim 5, wherein at least one of a spatial intra partition type table and an intra prediction mode table is received by the one or more processors using one or more high level syntax elements and is used by the one or more processors to decode picture data for the large block.
Claim 11	
11[pre]	A method of encoding picture data for at least a large block in a picture, the method comprising:

Reference	Limitation
11[a]	determining that intra prediction is to be performed for the large block, wherein the large block has a block size equal to or larger than 32×32 , and the large block size is greater than a size of a basic coding unit;
11[b]	encoding a binary split signaling syntax element specifying that the large block is further split into four equally sized sub-blocks;
11[c]	encoding an intra prediction mode for the large block that is not further split into four equally sized sub-blocks;
11[d]	partitioning the large block into the four equally sized sub-blocks in response to the decoded binary split signaling syntax element specifying that the large block is further split into four equally sized sub-blocks; and
11[e]	encoding a second syntax element indicating a single spatial intra partition type for at least one basic coding unit sized sub-block, the single spatial intra partition type being determinable from among a plurality of spatial intra partition types.
Claim 12	
12	The method of claim 11, further comprising encoding an intra prediction mode for the basic coding unit sized sub-block.
Claim 13	
13	The method of claim 11, further comprising encoding a further binary split signaling syntax element specifying that at least one of the four equally sized sub-blocks greater than a basic coding unit size is further split into four smaller equally sized sub-blocks.
Claim 14	
14	The method of claim 11, wherein the spatial intra partition type specifies a size of a partition of the basic coding unit sized sub-block.

Reference	Limitation
Claim 15	
15[pre]	An apparatus comprising:
15[a]	at least a memory and one or more processors, wherein the one or more processors are configured to:
15[b]	encode picture data for at least a large block in a picture by determining that intra prediction is to be performed for the large block, wherein the large block has a large block size equal to or larger than 32×32 , and the large block size is greater than a basic coding unit size, wherein the intra prediction is signaled for the large block and the one or more processors are configured to:
15[c]	encode a binary split signaling syntax element specifying that the large block is further split into four equally sized sub-blocks;
15[d]	encode an intra prediction mode for the large block that is not further split into four equally sized sub-blocks;
15[e]	partition the large block into the four equally sized sub-blocks in response to the binary split signaling syntax element specifying that the large block is further split into four equally sized sub-blocks; and
15[f]	encode a second syntax element indicating a single spatial intra partition type for at least one basic coding unit sized sub-block, the single spatial intra partition type being determinable from among a plurality of spatial intra partition types.
Claim 16	
16	The apparatus of claim 15, wherein, the one or more processors are configured to encode an intra prediction mode for the basic coding unit sized sub-block.

Reference	Limitation
Claim 17	
17	The apparatus of claim 15, wherein, the one or more processors are configured to encode a further binary split signaling syntax element specifying that at least one of the four equally sized sub-blocks greater than a basic coding unit size is further split into four smaller equally sized sub-blocks.
Claim 18	
18	The method of claim 15, wherein the spatial intra partition type specifies a size of a partition of the basic coding unit sized sub-block.

U.S. Patent No. 11,836,876 (“the ’876 patent,” EX-1001) generally relates to video encoding technology that converts video signals into compressed data, building upon the H.264 standard. At a high level, the purported invention is directed at expanding the legacy 16x16 macroblock into a larger block that can be split hierarchically. But the idea of using blocks larger than 16x16 and a hierarchical structure for those blocks was well-known in the industry as far back as 1994. As explained in this Petition, prior art such as VCEG-AJ21 improved upon H.264 by using larger macroblocks, and prior art such as Xiong applied a hierarchical structure to those larger macroblocks that is nearly identical to what the ’876 patent purports to claim. Simply put, the inventor of the ’876 patent claimed someone else’s idea.

For this reason, Amazon.com, Inc. and Amazon.com Services LLC (“Amazon” or “Petitioner”) petition for *inter partes* review (“IPR”) of claims 1-18 (“Challenged Claims”) of the ’876 patent.

I. GROUNDS FOR STANDING

Petitioner certifies the ’876 patent is available for IPR, and Petitioner is not barred or estopped from requesting IPR on the grounds herein. Petitioner has not been served with any complaint by Patent Owner InterDigital, Inc. (hereafter, “PO”) asserting the ’876 patent.

II. IDENTIFICATION OF CHALLENGE

A. Prior Art¹

Petitioner respectfully requests cancellation of claims 1-18 (“the Challenged Claims”) in view of:

Prior Art
Xiong (WO2008/083633), published July 17, 2008; prior art at least under §102(a) ² (EX-1005, certified English translation EX-1006).
VCEG-AJ21; publicly available by October 8, 2008; prior art at least under §102(a) (EX-1008)
H.264, titled “Advanced video coding for generic audiovisual services”; publicly available no later than July 1, 2008; prior art under at least 35 U.S.C. §102(b) (EX-1010).

B. Grounds for Challenge

Ground		Claims	Prior Art
1A	§103	1-8, 11-18	Xiong, VCEG-AJ21
1B	§103	9-10	Xiong, VCEG-AJ21, H.264

¹ References herein are prior art based on the earliest priority date on the ’876 patent’s face: July 1, 2009.

² All references to 35 U.S.C. §§102, 103 are to the pre-AIA statutes.

III. '876 PATENT OVERVIEW

The '876 patent, titled “Methods and Apparatus for Signaling Intra Prediction for Large Blocks for Video Encoders and Decoders,” issued March 19, 2024, from application 17/351,649 (filed June 18, 2021). The '876 patent is a continuation of application 15/806,893 (filed November 8, 2017), which, in turn, is a continuation of application 13/381,538 (filed June 29, 2010). The '876 patent claims priority to Provisional Application No. 61/222,177, filed July 1, 2009. For purposes of this Petition only, Petitioner has used July 1, 2009 as the priority date for the '876 patent.

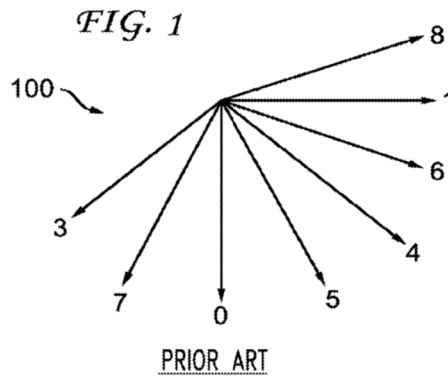
A. Technology Background

A video sequence displays consecutive images, or frames, at a rate that creates the illusion of motion. EX-1003 (“Acton”), ¶19. Each frame is divided into slices that are further partitioned into macroblocks (or “blocks” in newer standards), with each macroblock/block representing a small rectangular region of the frame that is individually processed. *Id.*

A video sequence generally is first processed using an encoder that generates encoded video data from the video sequence. Acton, ¶20. The encoded video data is then provided (e.g., by transmitting over a network) to a decoder, where the decoder reconstructs the original video sequence from the encoded video data. *Id.* Transmission of uncompressed video requires large amounts of data and bandwidth, which may stress a communication system and/or result in degradation or loss of

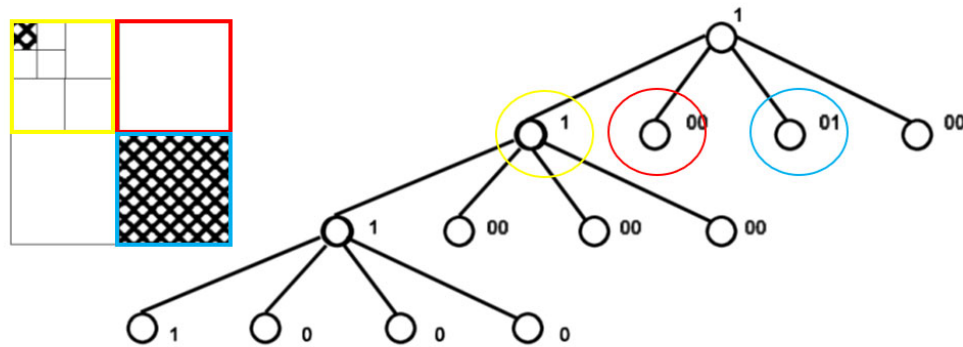
video quality. *Id.* Compression techniques have therefore been developed, and utilized by video encoders and video decoders, to reduce the amount of data required to transmit a video. *Id.* Such compression techniques include intra and inter prediction. *Id.* Intra prediction exploits spatial redundancy within a frame by predicting blocks in the frame based on other blocks in the frame, while inter prediction exploits temporal redundancy between frames by predicting blocks based on previously coded reference frames. *Id.*

Most modern video coding standards, including the MPEG-4 AVC Standard (“MPEG-4 AVC” or “H.264”), provide for intra prediction to exploit spatial correlations within a picture. EX-1001, 1:28-42; Acton, ¶21. Intra prediction works by predicting the contents of one block of pixels based on the contents of a nearby block of pixels. EX-1001, 1:28-42; Acton, ¶21. For example, intra prediction may indicate the direction of the set of pixels that are most similar to the pixels currently being coded, as shown in Figure 1 of the ’876 patent. EX-1001, Fig. 1, 1:48-2:12; Acton, ¶21.



In this figure, item 0 represents a “vertical prediction mode” (i.e., the value of the current block is predicted to be similar to the value of the block above it), item 1 represents a “horizontal prediction mode” (i.e., the value of the current block is predicted to be similar to the value of the block to its left), and the other numbers represent similar prediction directions (e.g., 4 represents “diagonal-down/right prediction mode”). EX-1001, 1:50-2:12. MPEG-4 AVC allows intra coding for block sizes of 4x4, 8x8, and 16x16 pixels. EX-1001, 1:28-42; Acton, ¶22.

As far back as 1994, there were proposals to use block sizes of up to 64x64 pixels. Acton, ¶23 (citing EX-1009, 4 (Table 2)). These large blocks would be coded using a hierarchical “quadtree structure.” *Id.* (citing EX-1009, 3 (Fig. 2) (“Dashed squares are intraframe coded zones.”)).



Id. (citing EX-1009, 3 (Fig. 2) (annotated)).

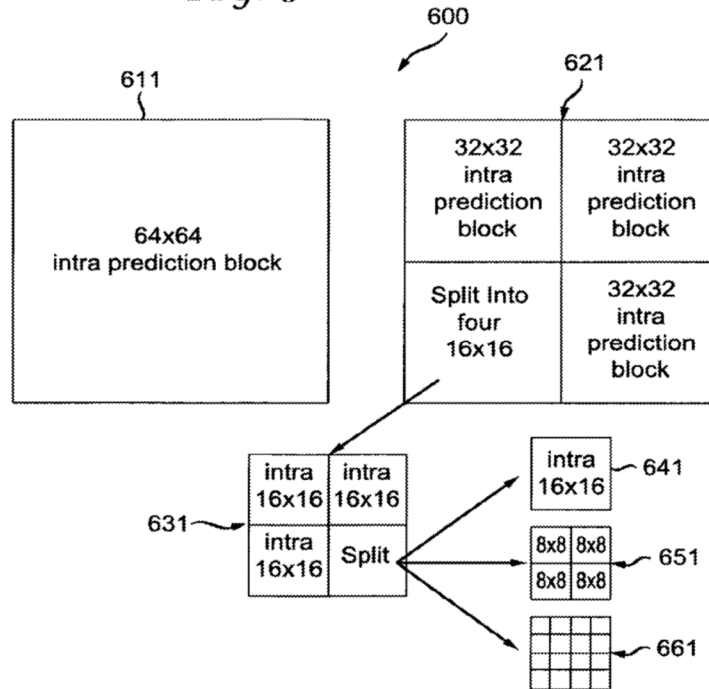
The quadtree allows encoding for any level of the hierarchy. EX-1009, 3; Acton ¶24. For example, as shown in the above figure, the single 1 bit at the top of the hierarchy indicates that the 64x64 block is not all encoded the same, but each of the four 32x32 sub-blocks must be encoded separately. EX-1009, 3; Acton, ¶24. A 0 bit at any level indicates that the entire block is encoded the same. EX-1009, 3; Acton, ¶24. For example, in the above figure “00” at the 32x32 block level indicates that the entire 32x32 block is encoded as interframe (binary code and associated 32x32 block highlighted in red), while “01” indicates that the entire 32x32 block is encoded intraframe (binary code and associated 32x32 block highlighted in blue). EX-1009, 3; Acton, ¶24. A 1 bit, on the other hand, indicates that the sub-blocks must be coded separately. So, as shown in the above figure, a “1” for a 32x32 block indicates that each 16x16 block within that particular 32x32 block is encoded separately (binary code and associated 32x32 block highlighted in yellow), because

some 16x16 blocks may be interframe, some may be intraframe, and some may need to be split further. Acton, ¶24. At the lowest level (e.g., 8x8 blocks), there is no need for the leading 0 bit, and the single bit indicates interframe or intraframe encoding for that leaf block. EX-1009, 3; Acton, ¶24. While this particular example uses a single bit to represent whether the block is interframe or intraframe encoded, the structure can be utilized with other block-based compression methods to reduce inter-block redundancy. EX-1009, 5; Acton, ¶24.

B. Purported Invention

According to the '876 patent, the purported invention is to signal intra prediction for a block that is larger than what it calls a “basic coding unit.” EX-1001, 4:20-34; Acton, ¶25. The '876 patent discloses signaling intra prediction by selecting a basic coding unit size and assigning a single spatial intra partition type for that basic coding unit size. EX-1001, 4:20-34; Acton, ¶25. Further, the intra prediction is hierarchical, allowing splitting from the large block to the basic coding unit size and merging from the basic coding unit size to the large block size. EX-1001, 4:20-34; Acton, ¶25. The hierarchical intra prediction uses “split signaling” to indicate whether each block is to be split into smaller sub-blocks, as shown in Figure 6. EX-1001, Fig. 6, 10:3-35; Acton, ¶25.

FIG. 6



The '876 patent includes an exemplary syntax table (Table 4) for the hierarchical coding. EX-1001, 11:10-48; Acton, ¶26.

TABLE 4

macroblock_layer) {	C	Descriptor
...		
intra64_flag	2	u(1)
if (intra64_flag==1) {		
intra64_DC_flag	2	u(1)
if (intra64_DC_flag==0) {		
intra_pred_mode_64	2	ue(v)/se(v)
if (intra_pred_mode_64==Multi-DIR) {		
intra64_multidir_index	2	ue(v)/se(v)
}		
}		
else {		
for (i32=0; i32<4; i32++) {		
intra32_flag[i32]	2	u(1)
if (intra32_flag[i32]==1) {		
intra32_DC_flag[i32]	2	u(1)
if (intra32_DC_flag[i32]==0) {		
intra_pred_mode_32[i32]	2	ue(v)/se(v)
if (intra_pred_mode_32[i32]==Multi-DIR) {		
intra32_multidir_index[i32]	2	ue(v)/se(v)
}		
}		
}		
}		
else {		
for (i16=0; i16<4; i16++) {		
sip_type[i16]	2	ue(v)/se(v)
if (sip_type[i16]==SIP16x16) {		
intra_pred_mode_16[i16]	2	ue(v)/se(v)
if (intra_pred_mode_16[i16]==Multi-DIR) {		
intra16_multidir_index[i16]	2	ue(v)/se(v)
}		
}		
else if (sip_type[i16]!=SIP16x16_DC) {		
mb_intra_prediction_syntax(); /*this is same as H.264*/		
}		
}		
...}		

As described in the '876 patent, a flag (“intra64_flag”) set to 1 indicates the entire 64x64 block uses one prediction mode (“intra_pred_mode_64”). EX-1001, 11:51-53; Acton, ¶27.³ If “intra64_flag” is set to 0, then for each 32x32 sub-block, a flag

³ The '876 patent is inconsistent with the capitalization of these variable names.

Table 4 does not capitalize them (e.g., “intra64_flag”), while the rest of the

(“intra32_flag”) is set to 1 to indicate that the entire 32x32 block uses one prediction mode (“intra_pred_mode_32”) or is set to 0 to indicate the 32x32 block is to be further split into 16x16 blocks. EX-1001, 11:51-12:6. The prediction mode for each of those 16x16 blocks is then set (“intra_pred_mode_16”). *Id.*, 12:49-51. For each of the 64x64, 32x32, and 16x16 levels, if the prediction mode is multi-directional (Multi-DIR), then the index of the angle is also coded for that block (“intra64_multidir_index,” “intra32_multidir_index,” “intra16_multidir_index”). *Id.*, 11:60-12:53. None of this, however, was new at the time of the patent; rather, as explained below, Patent Owner merely merged together ideas that were already well-known or would have been obvious in view of the existing prior art disclosures. Acton, ¶¶27-28.

IV. LEVEL OF ORDINARY SKILL

A person of ordinary skill in the art (“POSITA”) in the field of video encoding and decoding at the time of the purported invention of the ’876 patent would have had at least a bachelor’s degree in electrical engineering, computer science, computer engineering, or a related field, and approximately three years of experience in design, development, and/or testing of relevant multimedia processing, such as

specification refers to them as capitalized (e.g., “Intra64_flag”). For consistency, this Petition uses the non-capitalized versions from Table 4.

video, and/or image processing or coding. Acton, ¶29. Advanced graduate studies with coursework and/or research focusing on topics of video encoding and decoding could substitute for working experience in the field. *Id.*

V. CLAIM CONSTRUCTION

Claims should be construed “in accordance with the ordinary and customary meaning of such claim as understood by one of ordinary skill in the art and the prosecution history pertaining to the patent.” 37 C.F.R. §42.100(b). Petitioner is unaware of any “prior claim construction determination” related to the ’876 patent. Petitioner has not proposed any terms for construction because no constructions are necessary to resolve any disputes identified in this Petition. *Vivid Techs., Inc. v. Am. Sci. & Eng’g, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999).

Specifically regarding the term “basic coding unit,” Petitioner does not believe it requires construction to resolve this Petition, but for avoidance of doubt, the term should at least encompass a 16x16 block. Acton, ¶¶47-48. This is consistent with the ’876 patent disclosure of an embodiment in which the “basic coding unit” is “set” “to be 16x16.” EX-1001, 9:66-67.

VI. DETAILED EXPLANATION OF GROUNDS

A. Ground 1A: Xiong in view of VCEG-AJ21

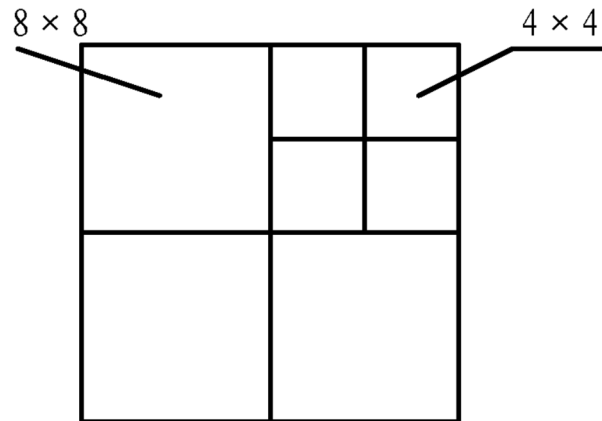
1. Xiong

Xiong (WO2008/083633) was published July 17, 2008 and qualifies as prior art under 35 U.S.C. §102(a).

Xiong discloses a method and device for intra-frame prediction encoding and decoding, which organizes blocks in hierarchical tree form. Xiong (Abstract).⁴ Xiong encodes “the basic encoding unit—a macroblock—or a sub-block obtained by dividing a macroblock.” Xiong, 6 (“basic encoding unit” is a macroblock, “or a sub-block obtained by dividing a macroblock”). After performing prediction encoding on each basic encoding unit, the final encoding result of each basic encoding unit is organized in a hierarchical tree structure. *Id.* For example, Figure 5 (below) shows a 16x16 macroblock, where the encoding result of the upper-left 8x8 sub-block is obtained using one prediction method (directional prediction method of H.264/MPEG-4 AVC); the encoding result of the lower-right 8x8 sub-block is obtained using a second prediction method (template matching prediction method); the encoding result of the upper-right 4x4 sub-block is obtained using a third

⁴ Citations to Xiong refer to the English translation of the WO2008/083633, EX-1006.

prediction method (displacement intra-frame prediction method); and the encoding results of the remaining three 4x4 sub-blocks of the upper right 8x8 block are obtained using the first prediction method (directional prediction method of H.264/MPEG-4 AVC). Xiong, 6, Fig. 5; Acton, ¶¶49-50.⁵



Xiong further explains that the macroblock can be sizes other than 16x16, including “a 32x32 macroblock.” Xiong, 6; Acton, ¶51.

Xiong discloses that the encoding result for the frame includes information indicating the block size and intra-frame encoding method for each block. Xiong, 7. If all sub-blocks within a block are encoded with the same prediction encoding

⁵ Figure 5 of Xiong appears to be missing hash marks that are included in the European counterpart. *Compare* EX-1005 (Original Xiong reference), Fig. 5 *with* EX-1007 (Xiong EP), Fig. 5.

method, information indicating the size of the block and the intra prediction encoding method used for that block is written into the encoding result for the frame.

Id. Similarly, if the block is a leaf node—meaning the system will not subdivide it further—information indicating the size of the block and the intra frame prediction encoding method used for that block is written into the encoding result for the frame.

Id. Otherwise (i.e., there are sub-blocks, and they use different intra prediction encoding methods), the subordinate branches are further processed. *Id.*; Acton, ¶52.

Xiong discloses a syntax table (Table 1) for encoding the multiple intra-frame prediction encoding methods. Xiong 7, 9 (Table 1); Acton, ¶53.

1	mb_pred(mb_type) {	C	Descriptor
2	ext_intra_pred_flag	2	u(1) ae(v)
3	if(ext_intra_pred_flag) {		
4	ext_16x16_pred_flag	2	u(1) ae(v)
5	if(ext_16x16_pred_flag)		
6	ext_16x16_pred_mode	2	ue(v) ae(v)
7	if(ext_16x16_pred_mode == DIP)		
8	for(compIdx = 0; compIdx < 2; compIdx++)		
9	dvd[0][0][compIdx]	2	ue(v) ae(v)
10	else {		
11	for(luma8x8BlkIdx=0; luma8x8BlkIdx<4; luma8x8BlkIdx++) {		
12	ext_8x8_pred_flag[luma8x8BlkIdx]	2	u(1) ae(v)
13	if(ext_8x8_pred_flag[luma8x8BlkIdx]) {		
14	ext_8x8_pred_mode[luma8x8BlkIdx]	2	ue(v) ae(v)
15	if(ext_8x8_pred_mode[luma8x8BlkIdx] == DIP)		
16	for(compIdx = 0; compIdx < 2; compIdx++)		
17	dvd[luma8x8BlkIdx][0][compIdx]	2	ue(v) ae(v)
18	} else {		
19	noSubMbPartSizeLessThan8x8Flag = 0		
20	for(luma4x4BlkIdx=0; luma4x4BlkIdx<4;		
21	luma4x4BlkIdx++) {		
22	ext_4x4_pred_mode[luma8x8BlkIdx][luma4x4BlkIdx]	2	ue(v) ae(v)
23	if(ext_4x4_pred_mode[luma8x8BlkIdx][luma4x4BlkIdx]		
24	==		
25	DIP)		
26	for(compIdx = 0; compIdx < 2; compIdx++)		
27	dvd[luma8x8BlkIdx][luma4x4BlkIdx][compIdx]	2	ue(v) ae(v)
28	}		
29	}		
	}		
	}		
	}		
	} else {		
	...		

Xiong, 9 (Table 1) (annotated with row numbers and colored rectangles).

The syntax element `ext_16x16_pred_flag` indicates whether the entire 16x16 block uses the same intra prediction encoding method or whether it is divided into 8x8 blocks that use different intra prediction encoding methods. *Id.*, 8 (“`ext_16x16_pred_flag` equal to 1 indicates that the macroblock for this intra prediction type is partitioned as 16x16; `ext_16x16_pred_flag` equal to 0 indicates that the macroblock is further divided into 8x8 blocks”). In Table 1, `ext_16x16_pred_flag` (shown in red rectangle) is coded as a syntax element in row

4. As seen in row 5, if `ext_16x16_pred_flag` is set to 1 (true), all of the statements in rows 6 through 9 (shown in blue rectangle) are processed. Acton, ¶54.

3	<code>if(ext_16x16_pred_flag)</code>		
4	<code>ext_16x16_pred_flag</code>	2	<code>u(1) ae(v)</code>
5	<code>if(ext_16x16_pred_flag)</code>		
6	<code>ext_16x16_pred_mode</code>	2	<code>ue(v) ae(v)</code>
7	<code>if(ext_16x16_pred_mode == DIP)</code>		
8	<code>for(compldx = 0; compldx < 2; compldx++)</code>		
9	<code>dvd[0][0][compldx]</code>	2	<code>ue(v) ae(v)</code>

Those rows code a syntax element for the prediction mode for the entire 16x16 block (row 6). The syntax element `ext_16x16_pred_mode` indicates the intra-frame prediction method for the 16x16 block. Xiong, 8 (“`ext_16x16_pred_mode` indicates the intra frame prediction encoding method for a macroblock partitioned as 16x16”). If the prediction mode is set to DIP (displacement intra-frame prediction)—where the differences between the current block and the candidate block are identified in a displacement vector—then `dvd` (the displacement vector) is coded (row 9). *Id.*, 2 (“[W]hen encoding the current block, a similar block can first be searched for within the already encoded region to serve as a candidate block. This candidate block is indicated using a displacement vector labeling method, which identifies the differences between the current block and the candidate block.”), 8 (“`dvd[ ][ ][ ]` represents the displacement vector used when the encoding method of displacement intra frame prediction (DIP) is applied”); Acton, ¶55.

If `ext_16x16_intra_pred_flag` is not set, then rows 11 through 24 (shown in green rectangle) are processed. Acton, ¶56.

11	for(luma8x8BlkIdx=0; luma8x8BlkIdx<4; luma8x8BlkIdx++) {		
12	ext_8x8_pred_flag[luma8x8BlkIdx]	2	u(1) ae(v)
13	if(ext_8x8_pred_flag[luma8x8BlkIdx]) {		
14	ext_8x8_pred_mode[luma8x8BlkIdx]	2	ue(v) ae(v)
15	if(ext_8x8_pred_mode[luma8x8BlkIdx] == DIP)		
16	for(compIdx = 0; compIdx < 2; compIdx++)		
17	dvd[luma8x8BlkIdx][0][compIdx]	2	ue(v) ae(v)
18	} else {		
19	noSubMbPartSizeLessThan8x8Flag = 0		
20	for(luma4x4BlkIdx=0; luma4x4BlkIdx<4; luma4x4BlkIdx++) {		
21	ext_4x4_pred_mode[luma8x8BlkIdx][luma4x4BlkIdx]	2	ue(v) ae(v)
22	if(ext_4x4_pred_mode[luma8x8BlkIdx][luma4x4BlkIdx] == DIP)		
23	for(compIdx = 0; compIdx < 2; compIdx++)		
24	dvd[luma8x8BlkIdx][luma4x4BlkIdx][compIdx]	2	ue(v) ae(v)

These instructions include a loop (starting at row 11), where each of the 8x8 sub-blocks of the 16x16 block are processed. For each 8x8 sub-block, the ext_8x8_pred_flag is set (row 12) if each of its 4x4 sub-blocks were encoded with the same intra prediction encoding method. Xiong 8, 9 (Table 1). The ext_8x8_pred_flag is checked at row 13. If it is set (i.e., all 4x4 sub-blocks encoded with the same intra prediction encoding method), then rows 14 through 17 are processed. Acton, ¶57.

14	ext_8x8_pred_mode[luma8x8BlkIdx]	2	ue(v) ae(v)
15	if(ext_8x8_pred_mode[luma8x8BlkIdx] == DIP)		
16	for(compIdx = 0; compIdx < 2; compIdx++)		
17	dvd[luma8x8BlkIdx][0][compIdx]	2	ue(v) ae(v)

Row 14 sets ext_8x8_pred_mode to the intra prediction encoding method used for that 8x8 block. Xiong, 8 (“ext_8x8_pred_flag[] equal to 1 indicates prediction with an 8x8 partition, and ext_8x8_pred_flag[] equal to 0 indicates that the sub-block is further divided into 4x4 sub-blocks”), 9 (Table 1). If the prediction

mode is set to DIP (displacement intra-frame prediction), then dvd (the displacement vector) is coded (row 17). Acton, ¶58.

If ext_8x8_intra_pred_flag is not set, then rows 19 through 24 are processed. Acton, ¶59.

19	noSubMbPartSizeLessThan8x8Flag = 0		
20	for(luma4x4BlkIdx=0; luma4x4BlkIdx<4; luma4x4BlkIdx++) {		
21	ext_4x4_pred_mode[luma8x8BlkIdx][luma4x4BlkIdx]	2	ue(v) ae(v)
22	if(ext_4x4_pred_mode[luma8x8BlkIdx][luma4x4BlkIdx] == DIP)		
23	for(compIdx = 0; compIdx < 2; compIdx++)		
24	dvd[luma8x8BlkIdx][luma4x4BlkIdx][compIdx]	2	ue(v) ae(v)

In those rows, for each of the 4x4 sub-blocks, the ext_4x4_pred_mode is set (row 21) to the intra prediction encoding method used for that 4x4 block. Xiong, 8, 9 (Table 1). If the prediction mode is set to DIP (displacement intra-frame prediction), then dvd (the displacement vector) is coded (row 24). Acton, ¶60.

So at each level of the hierarchy, a flag (ext_[NxN]_pred_flag) is set. If set to 1, the prediction mode for that entire block is coded (ext_[NxN]_pred_mode). Acton, ¶61. If the flag is not set, there is a loop that processes each of the sub-blocks. Further, when the intra-prediction encoding method is set for a particular level of the hierarchy, and that intra-prediction encoding method is DIP (displacement intra prediction), then the displacement vector is also coded for that block. Xiong, 5, 9 (Table 1); Acton, ¶61.

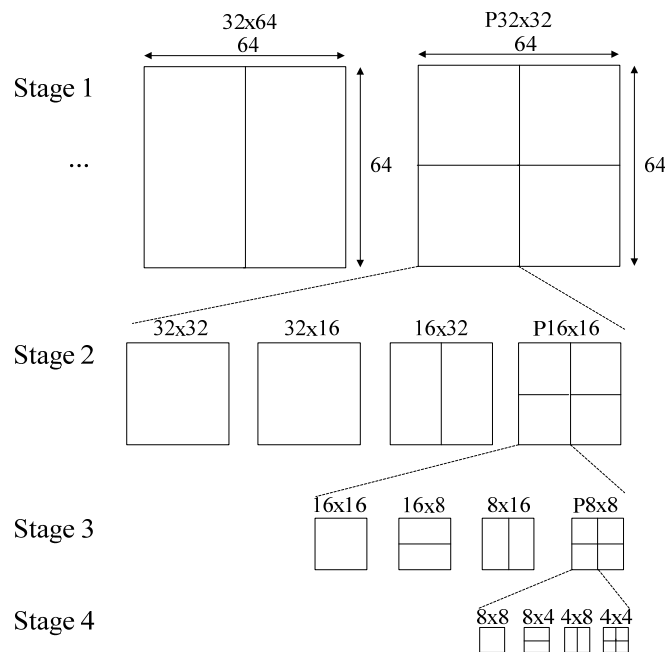
Notably, the syntax table of Xiong (Table 1) is analogous to the syntax table of the '876 patent (Table 4). Each table has a flag at each hierarchical level indicating whether a prediction mode is applied for the entire block (intra[N]_flag for the '876 patent, and ext_[NxN]_pred_flag for Xiong). Each codes the mode for the entire block if the flag is set to 1 (intra_pred_mode_[N] for the '876 patent, and ext_[NxN]_pred_mode for Xiong). If the flag is set to 0, each has a loop that processes each subblock. And each codes direction information for the block based on the prediction mode (intra[N]_multidir_index for the '876 patent, and dvd for Xiong). Acton, ¶62.

2. VCEG-AJ21

VCEG-AJ21 is a printed publication that was publicly available by October 8, 2008, making it prior art to the '876 patent under 35 U.S.C. §102(a). Reader, ¶¶31-48. For example, Dr. Reader explains that VCEG-AJ21 would have been uploaded and made available for download by the public by the beginning of the VCEG October 2008 meeting, where it was scheduled to be discussed. *Id.*, ¶44.

VCEG-AJ21 extended the architecture of the H.264 codec with enlarged macroblock sizes. VCEG-AJ21, 1 (Abstract). In particular, VCEG-AJ21 discloses enlarging the H.264 macroblock size of 16x16 to “NxN,” where N is increased according to powers of 2, and gives examples of 32x32, 64x64, and 128x128. VCEG-AJ21, 3 (“N indicates the maximum pixel size for the enlarged MB and can

be increased with power of 2 such as 32, 64 or 128.”). VCEG-AJ21 also discloses using a hierarchical structure for large blocks. Figure 3 shows such a hierarchy starting with a 64x64 block. Acton, ¶¶63-64.



VCEG-AJ21, 3 (Fig. 3).

3. Motivation to Combine Xiong and VCEG-AJ21

A POSITA would have been motivated to combine Xiong and VCEG-AJ21 to extend Xiong’s hierarchical block structure to larger blocks, including 64x64 and 128x128 and would have had a reasonable expectation of success in doing so. Acton, ¶65.

Both references build upon H.264. Xiong (Background Art); VCEG-AJ21, 1 (Abstract). In particular, both references describe macroblocks larger than the 16x16 macroblocks used in H.264. Xiong, 6 (“Of course, the macroblock here is not

necessarily limited to a 16×16 macroblock; it can also be of other sizes, such as a 32×32 macroblock.”); VCEG-AJ21, 3 (“Table 1 lists the H.264 elements that must be modified by enlarging the MB sizes. As aforementioned, this is not the exhaustive list. In Table 1, N indicates the maximum pixel size for the enlarged MB and can be increased with power of 2 such as 32, 64 or 128.”); Acton, ¶66.

A POSITA would have used VCEG-AJ21’s disclosure of 64x64 and 128x128 macroblocks as examples of macroblocks “of other sizes,” referred to in Xiong. Acton, ¶67. A POSITA would have had a reasonable expectation of success using 64x64 and 128x128 blocks with Xiong because the hierarchical structure and related syntax is easily expanded using routine engineering principles. *Id.* In particular, a POSITA would have known to reuse or replicate the portion of software/firmware in Xiong’s system for encoding and decoding the syntax elements described in Table 1 (which addresses 16x16 and 8x8 blocks) to work on a larger, 32x32 block (e.g., using a recursive function call to address each level of the hierarchy). *Id.* The result—in accordance with Xiong’s disclosure of `ext_16x16_pred_flag` and 32x32 macroblocks—would include an `ext_32x32_pred_flag` for 32x32 macroblocks. *Id.* Further, a POSITA would have understood that it would be a simple extension (another reuse or replication of code) to add the 64x64 macroblocks disclosed by VCEG-AJ21, providing `ext_64x64_pred_flag` and `ext_64x64_pred_mode`. *Id.* In fact, a POSITA would have understood generally that VCEG-AJ21’s disclosure of

an enlarged macroblock size combined with Xiong would result in `ext_[NxN]_pred_flag` and `ext_[NxN]_pred_mode`, where “*N* indicates the maximum pixel size for the enlarged MB and can be increased with power of 2 such as 32, 64 or 128.” VCEG-AJ21, 3; Acton, ¶67. And as explained in VCEG-AJ21, the benefit of extending the system to use larger block sizes is “enhanced coding efficiency for high resolution video coding. This extension is definitely beneficial to the long-term living of H.264 for high resolution video coding beyond HD resolutions.” VCEG-AJ21, 6; Acton, ¶¶67-69.

4. Claim 1

a. 1[pre]: *A method of decoding picture data for at least a large block in a picture, the method comprising:*

If limiting, Xiong discloses or suggests 1[pre]. Xiong discloses a “method...for intra frame prediction...decoding.” Xiong (Abstract). Xiong discloses using macroblocks that can be 32x32 pixels. *Id.*, 6 (“Of course, the macroblock here is not necessarily limited to a 16×16 macroblock; it can also be of other sizes, such as a 32×32 macroblock.”). Xiong further discloses decoding picture data from macroblocks. *Id.*, 10 (“Based on the information indicating block size and the intra frame prediction encoding method used for each block, decoding is performed.”); Acton, ¶70.

Thus, Xiong discloses or suggests *a method of decoding picture data* for a macroblock (*a large block* (such as a 32x32 block)) *in a picture*.⁶ Acton, ¶70.

VCEG-AJ21 also discloses using macroblocks of any size NxN, where “N indicates the maximum pixel size for the enlarged MB and can be increased with power of 2 such as 32, 64 or 128.” VCEG-AJ21, 3. Thus, the combined teachings of Xiong and VCEG-AJ21 use large blocks that are 32x32, 64x64, or even 128x128. Acton, ¶71.

- b. ***1[a]: determining that intra prediction is to be performed for the large block, wherein the large block has a block size equal to or larger than 32×32, and the large block size is greater than a size of a basic coding unit;***

Xiong, alone or in combination with VCEG-AJ21, discloses or suggests limitation 1[a]. Xiong discloses intra-frame prediction, as well as inter-frame prediction. Xiong, (Background Art). Xiong’s Background Art section discusses H.264/MPEG-4 AVC and how its intra-prediction encoding method can improved. *Id.*; Acton, ¶72. To that end, Xiong discloses that if the ext_intra_pred_flag is set to 1, then multiple intra-prediction encoding methods are used within the frame,

⁶ Italicized text is claim text unless otherwise noted.

meaning intra-prediction encoding (and therefore decoding)⁷ is handled on a block-by-block basis (*determining that intra prediction is to be performed for the large block*). Xiong, 8, 9 (Table 1); Acton, ¶72.

Xiong discloses blocks larger than 16x16. Xiong, 6 (“Of course, the macroblock here is not necessarily limited to a 16×16 macroblock; it can also be of other sizes, such as a 32×32 macroblock.”). While Xiong expressly recites a 32x32 macroblock as one example of a macroblock “of other sizes,” a POSITA would have understood that the reference to macroblocks “of other sizes” would include a 64x64 macroblock, as macroblocks are generally sized based on powers of 2, and therefore a 64x64 macroblock is the next size up from a 32x32 macroblock ($2^5=32$, $2^6=64$). Acton, ¶73. VCEG-AJ21 also discloses using macroblocks of any size NxN, where “N indicates the maximum pixel size for the enlarged MB and can be increased with power of 2 such as 32, 64 or 128.” VCEG-AJ21, 3. Thus, the combined teachings of Xiong and VCEG-AJ21 use macroblocks that are 32x32, 64x64, or even 128x128 (*wherein the large block has a block size equal to or larger than 32×32*). Acton, ¶73. And a POSITA would have been motivated to make this combination for the reasons discussed above in §VI.A.3.

⁷ Xiong discloses an “encoding/decoding method” based on the encoding structure of Table 1. Xiong, 10.

Xiong discloses 16x16 blocks for intra prediction coding that, as discussed above (*supra* §V), corresponds to a *basic coding unit*. Xiong, 8 (“ext_16x16_pred_flag equal to 1 indicates that the macroblock for this intra prediction type is partitioned as 16x16.”). Therefore, Xiong’s 32x32 block (*the large block*) is both *equal to or larger than 32×32*, and has a size that is *greater than a size of a basic coding unit* (i.e., larger than a 16x16 block used for intra-prediction coding). Acton, ¶74. Similarly, VCEG-AJ21’s 64x64 and 128x128 blocks are also *equal to or larger than 32×32* and *greater than a size of a basic coding unit* (larger than a 16x16 block used for intra prediction coding), thereby qualifying as a “large block” within the scope of this claim, and meeting this limitation. *Id.*

c. 1[b]: decoding a binary split signaling syntax element specifying that the large block is further split into four equally sized sub-blocks;

Xiong in combination with VCEG-AJ21 discloses or suggests limitation 1[b]. In particular, the combination discloses or suggests flags that indicate whether a 64x64 block and a 32x32 block is split into four equally sized sub-blocks (32x32 and 16x16 respectively). Acton, ¶75. Those flags are *binary split signaling syntax element[s] specifying that the large block is further split into four equally sized sub-blocks*. *Id.*

Xiong discloses flags that indicate whether the 16x16 block and the 8x8 block are further split into four equally sized sub-blocks. Xiong, 8, 9 (Table 1); Acton, ¶76.

For example, “ext_16x16_pred_flag equal to 0 indicates that the macroblock is further divided into 8x8 blocks,” “and ext_8x8_pred_flag[] equal to 0 indicates that the sub-block is further divided into 4x4 sub-blocks.” Xiong, 8. Xiong also discloses that the macroblock may be sizes other than 16x16, “such as a 32x32 macroblock.” *Id.*, 6. As discussed above describing Xiong, a POSITA would have understood from Xiong’s teachings of using ext_16x16_pred_flag to indicate splitting of a 16x16 block into 8x8 subblocks, that when using a 32x32 macroblock, there would also be an ext_32x32_pred_flag that, when set to 0, indicates the 32x32 macroblock is divided into four 16x16 blocks. *See* §§VI.A.1, VI.A.3; Acton, ¶76.

Further, a POSITA would have understood that the combination of Xiong and VCEG-AJ21 taught macroblocks of any size with width and height of 2^N , where N is an integer (e.g., $2^5=32x32$, $2^6=64x64$, $2^7=128x128$). VCEG-AJ21, 3; Acton, ¶77. For each block size, there would be a flag indicating whether that block is split into four equally sized sub-blocks, as Xiong explicitly disclosed for 16x16, 8x8 blocks. Xiong, 7 (“Specifically, if multiple blocks in the frame are encoded using at least two different intra frame prediction encoding methods, with each method applied to one or more blocks in the frame, information indicating that multiple intra frame prediction encoding methods were used for prediction encoding is written into the encoding result of the frame.”), 10; Acton, ¶77. So, for a 64x64 block, there would be a flag (e.g., ext_64x64_pred_flag) that indicates that block should be partitioned

into four 32x32 sub-blocks. Acton, ¶77. And a POSITA would have been motivated to make this combination for the reasons discussed above in §VI.A.3.

The flags indicating whether a block is to be subdivided are set to either 0 (block is to be split) or 1 (block uses a single prediction mode, and is not to be split), and are therefore *binary*. Xiong, 8, 9 (Table 1) (each flag is encoded as “u(1)” indicating a 1 bit unsigned integer). Xiong discloses decoding these flags. *Id.*, 10 (“If ext_16x16_pred_flag equals 1, this indicates that the macroblock is partitioned as 16x16. The ext_16x16_pred_mode is then parsed to obtain the encoding method of the current block.”); Acton, ¶78.

Therefore, Xiong in combination with VCEG-AJ21 discloses a flag (*a binary split signaling element*) specifying that a 32x32 block or a 64x64 block (each a *large block*) is split into 16x16 sub-blocks (for a 32x32 block) or 32x32 blocks (for a 64x64 block) (*four equally sized sub-blocks*). Acton, ¶79.

d. 1[c]: decoding an intra prediction mode for the large block that is not further split into four equally sized sub-blocks;

Xiong in combination with VCEG-AJ21 discloses or suggests limitation 1[c]. As discussed for limitation 1[b], Xiong discloses a flag that indicates whether a block is further split into four equally sized sub-blocks, such as ext_16x16_pred_flag and ext_8x8_pred_flag. *See* §VI.A.4.c. When ext_16x16_pred_flag is set to 1, that “indicates that the macroblock for this intra prediction type is partitioned as 16×16.”

Xiong, 8. Similarly, when `ext_8x8_pred_flag` is set to 1, that “indicates prediction with an 8×8 partition.” Xiong, 8; Acton, ¶80.

As also discussed for limitation 1[b], Xiong discloses a 32x32 macroblock, and Xiong in combination with VCEG-AJ21 discloses or suggests a 64x64 macroblock, and a POSITA would have understood, based on Xiong’s teachings, there would also be an `ext_32x32_pred_flag` and an `ext_64x64_pred_flag` for those blocks. See §VI.A.4.c; Acton, ¶81. Just like `ext_16x16_pred_flag` and `ext_8x8_pred_flag`, when `ext_32x32_pred_flag` or `ext_64x64_pred_flag` is set to 1, it indicates that the intra prediction mode for that block (the *large block*) applies to the entire block; it *is not further split into four equally sized sub-blocks*. Xiong, 8; Acton, ¶81.

As shown in the syntax of Table 1, Xiong discloses that if `ext_16x16_pred_flag` is set (i.e., is 1), `ext_16x16_pred_mode`—“the intra frame prediction encoding method for a macroblock partitioned as 16x16”—is encoded. Xiong, 8, 9 (Table 1). Similarly, if `ext_8x8_pred_flag` is set, `ext_8x8_pred_mode`—“the intra frame prediction encoding method for each sub-block partitioned as 8×8”—is encoded. Xiong, 8, 9 (Table 1). Xiong discloses decoding `ext_16x16_pred_mode` and `ext_8x8_pred_mode`. Xiong, 10 (“If `ext_16x16_pred_flag` equals 1, this indicates that the macroblock is partitioned as 16×16. The `ext_16x16_pred_mode` is then parsed to obtain the encoding method of

the current block. The decoding method corresponding to this encoding method is used to decode the 16×16 macroblock.”), *id.* (“For each 8×8 block, based on its `ext_8x8_pred_flag`, if `ext_8x8_pred_flag` equals 1, the intra frame prediction encoding method `ext_8x8_pred_mode` of the 8×8 block is parsed from the code stream.”); Acton, ¶82.

A POSITA would have understood that just like when Xiong’s `ext_16x16_pred_flag` is set to 1, `ext_16x16_pred_mode` is encoded to indicate the intra-frame encoding for the entire 16×16 block, for Xiong’s 32×32 macroblock, if `ext_32x32_pred_flag` was set to 1, the intra-frame encoding method for that entire 32×32 block would be encoded. Xiong, 8 (“`ext_16x16_pred_mode` indicates the intra frame prediction encoding method for a macroblock partitioned as 16×16 ”), 9 (Table 1) (“if(`ext_16x16_pred_flag`)” encode `ext_16x16_pred_mode`); Acton, ¶83. A POSITA would have also understood that `ext_32x32_pred_mode` would be decoded in the same way as `ext_16x16_pred_mode` and `ext_8x8_pred_mode`, thereby decoding it would provide an intra prediction mode for the large block that is not further split into four equally sized sub-blocks, meeting this limitation. Acton, ¶83.

Further, a POSITA would have understood, based on Xiong’s teachings, that for a 64×64 macroblock, as disclosed by VCEG-AJ21, if `ext_64x64_pred_flag` was set to 1, the intra-frame encoding method for that entire 64×64 block would be

encoded (i.e., there would be `ext_64x64_pred_mode` similar to `ext_16x16_pred_mode` and `ext_8x8_pred_mode`). Acton, ¶84. A POSITA would have also understood that `ext_64x64_pred_mode` would be decoded in the same way as `ext_16x16_pred_mode` and `ext_8x8_pred_mode`, thereby decoding it would provide an intra prediction mode for the large block that is not further split into four equally sized sub-blocks, meeting this limitation. *Id.*

Therefore, this limitation is met both by decoding the prediction mode for a 32x32 block (`ext_32x32_pred_mode`) (*decoding an intra prediction mode for the large block*) that is not split into 16x16 subblocks (`ext_32x32_pred_flag` set to 1) (*that is not further split into four equally sized sub-blocks*), and separately by decoding the prediction mode for a 64x64 block (`ext_64x64_pred_mode`) (*decoding an intra prediction mode for the large block*) that is not split into 16x16 subblocks (`ext_64x64_pred_flag` set to 1) (*that is not further split into four equally sized sub-blocks*). Acton, ¶85.

- e. **1[d]: *partitioning the large block into the four equally sized sub-blocks in response to the decoded binary split signaling syntax element specifying that the large block is further split into four equally sized sub-blocks; and***

Xiong in combination with VCEG-AJ21 discloses or suggests limitation 1[d]. As discussed for limitation 1[b], Xiong discloses a flag that indicates whether a block

is further split into four equally sized sub-blocks, such as `ext_16x16_pred_flag` and `ext_8x8_pred_flag`. *See* §VI.A.4.c; Acton, ¶86.

As also discussed for limitation 1[b], Xiong and VCEG-AJ21 each disclose a 32x32 macroblock, and a POSITA would have understood, based on Xiong's teachings, there would also be an `ext_32x32_pred_flag`, and `ext_32x32_pred_flag` set to 0 indicates the 32x32 macroblock is divided into four 16x16 blocks. *See* §VI.A.4.c; Acton, ¶87.

As also discussed for limitation 1[b], VCEG-AJ21 discloses or suggests a 64x64 macroblock, and a POSITA would have understood, based on Xiong's teachings, there would also be an `ext_64x64_pred_flag`, and `ext_64x64_pred_flag` set to 0 indicates the 64x64 macroblock is divided into four 32x32 blocks. *See* §VI.A.4.c; Acton, ¶88.

Xiong discloses "If `ext_16x16_pred_flag` equals 0, this indicates that the 16x16 macroblock is divided into 8x8 blocks, and further parsing of each 8x8 block of the macroblock is required," and "[i]f `ext_8x8_pred_flag` equals 0, this indicates that the current block is further divided into 4x4 sub-blocks, and the intra frame prediction encoding method `ext_4x4_pred_mode` of each 4x4 block is parsed." Xiong, 10; Acton, ¶89. A POSITA would have understood the decoder to process `ext_32x32_pred_flag` and `ext_64x64_pred_flag` in a similar manner—i.e., if `ext_32x32_pred_flag` is set to 0, the current block is divided into four 16x16 sub-

blocks, and if ext_64x64_pred_flag is set to 0, the current block is divided into four 32x32 sub-blocks. Acton, ¶89. As such, Xiong discloses or suggests dividing the 32x32 macroblock (*partitioning the large block*) into four 16x16 sub-blocks (*into the four equally sized sub-blocks*) when ext_32x32_pred_flag is set to 0 (*in response to the decoded binary split signaling syntax element specifying that the large block is further split into four equally sized sub-blocks*). *Id.* In addition, Xiong also discloses or suggests dividing the 64x64 macroblock (*partitioning the large block*) into four 32x32 sub-blocks (*into the four equally sized sub-blocks*) when ext_64x64_pred_flag is set to 0 (*in response to the decoded binary split signaling syntax element specifying that the large block is further split into four equally sized sub-blocks*). *Id.*

- f. **1[e]: decoding a second syntax element indicating a single spatial intra partition type for at least one basic coding unit sized sub-block, the single spatial intra partition type being determinable from among a plurality of spatial intra partition types.**

Xiong in combination with VCEG-AJ21 discloses or suggests limitation 1[e]. As discussed for limitation 1[b], Xiong discloses a flag, ext_16x16_pred_flag, that indicates whether a 16x16 block is split into four 8x8 sub-blocks. *See* §VI.A.4.c; Acton, ¶90.

As `ext_16x16_pred_flag` sets the intra partition type for a 16x16 block—corresponding to a *basic coding unit* (*supra* §V)—it is *indicating a single spatial intra partition type for at least one basic coding unit sized sub-block*. Acton, ¶91.

Xiong thus discloses `ext_16x16_pred_flag` (*a second syntax element*) that when set to 1 indicates an intra partition type of 16x16 (*indicating a single spatial intra partition type for at least one basic coding unit sized sub-block*), where the intra partition type is selected from, at least 16x16, 8x8, and 4x4 (*the single spatial intra partition type being determinable from among a plurality of spatial intra partition types*). Acton, ¶92.

To the extent PO argues `ext_16x16_pred_flag` does not indicate a *single spatial intra partition type*, Xiong further discloses encoding (and therefore decoding)⁸ “the size of each block,” which is a syntax element indicating a single spatial intra partition type being determinable from among a plurality of spatial intra partition types. Xiong, 4 (“Information indicating the size of each block can be written into the encoding result, so that the blocks within the same frame are not limited to a fixed partitioning scheme, which is beneficial for using different intra

⁸ Xiong discloses an “encoding/decoding method” based on the encoding structure of Table 1. Xiong, 10.

frame prediction encoding methods to achieve improved encoding performance.”); Acton, ¶93.

To the extent PO argues that “ext_16x16_pred_flag” does not alone meet this limitation, this limitation would have been obvious to a POSITA, based on the teachings of Xiong. The combination of ext_16x16_pred_flag and ext_8x8_pred_flag indicate the partition size, as shown in the table below. Acton, ¶94.

ext_16x16_pred_flag	ext_8x8_pred_flag (for each 8x8 sub-block)	Partition Size
1	N/A	16x16
0	1	8x8
0	0	4x4

A POSITA would have been motivated by a desire for efficiency and simplicity to encode both ext_16x16_pred_flag and ext_8x8_pred_flag as a single syntax element. Acton, ¶95. Doing so would require fewer bits to be transmitted, thereby saving bandwidth. *Id.* And decoding and processing the single syntax element would be a simpler process and would be possible given the limited number of partition types (8x8, 4x4) that are smaller than the basic coding unit. *Id.*

5. Claim 2: *The method of claim 1, further comprising decoding an intra prediction mode for the basic coding unit sized sub-block.*

Xiong discloses or suggests the additional limitation of claim 2. Acton, ¶96.

Xiong discloses decoding an intra prediction mode for a 16x16 macroblock—which is a *basic coding unit sized sub-block*, see §V. Xiong, 8 (“ext_16x16_pred_mode indicates the intra frame prediction encoding method for a macroblock partitioned as 16×16”). As discussed for limitations 1[a] and 1[e], Xiong discloses doing intra prediction on a 16x16 block. See §§VI.A.4.b and VI.A.4.e; Acton, ¶97.

Xiong discloses decoding the intra prediction mode for a 16x16 block by parsing ext_16x16_pred_mode. Xiong, 8 (“ext_16x16_pred_mode indicates the intra frame prediction encoding method for a macroblock partitioned as 16x16”). As discussed for limitation 1[a], Xiong discloses using a 32x32 block, of which a 16x16 block would be a sub-block. See §VI.A.4.b; Acton, ¶98. Xiong thus discloses decoding ext_16x16_pred_mode (*decoding an intra prediction mode*) for a 16x16 block that is contained within a 32x32 block (*basic coding unit sized sub-block*). Acton, ¶98.

6. **Claim 3:** *The method of claim 1, further comprising decoding a further binary split signaling syntax element specifying that at least one of the four equally sized sub-blocks greater than a basic coding unit size is further split into four smaller equally sized sub-blocks.*

Xiong in combination with VCEG-AJ21 discloses or suggests the additional limitation of claim 3. Acton, ¶99.

In addition to teaching larger blocks and prediction flags for the blocks, Xiong also discloses the `ext_16x16_pred_flag`, which, when set to 0, “indicates that the [16x16] macroblock is further divided into 8x8 blocks.” Xiong, 8; Acton, ¶100.

As discussed for limitation 1[b], Xiong in combination with VCEG-AJ21 discloses a flag for a 64x64 block that is *a binary split signaling syntax element specifying that the large block is further split into four equally sized sub-blocks*. See §VI.A.4.c; Acton, ¶101. And the flag for a 32x32 block (e.g., `ext_32x32_pred_flag`) is *a further binary split signaling syntax element specifying that at least one of the four equally sized sub-blocks (32x32 blocks)—which are greater than a basic coding unit size (e.g., a 16x16 block, see §V)—is further split into four smaller equally sized sub-blocks*. Acton, ¶101.

7. **Claim 4:** *The method of claim 1, wherein the spatial intra partition type specifies a size of a partition of the basic coding unit sized sub-block.*

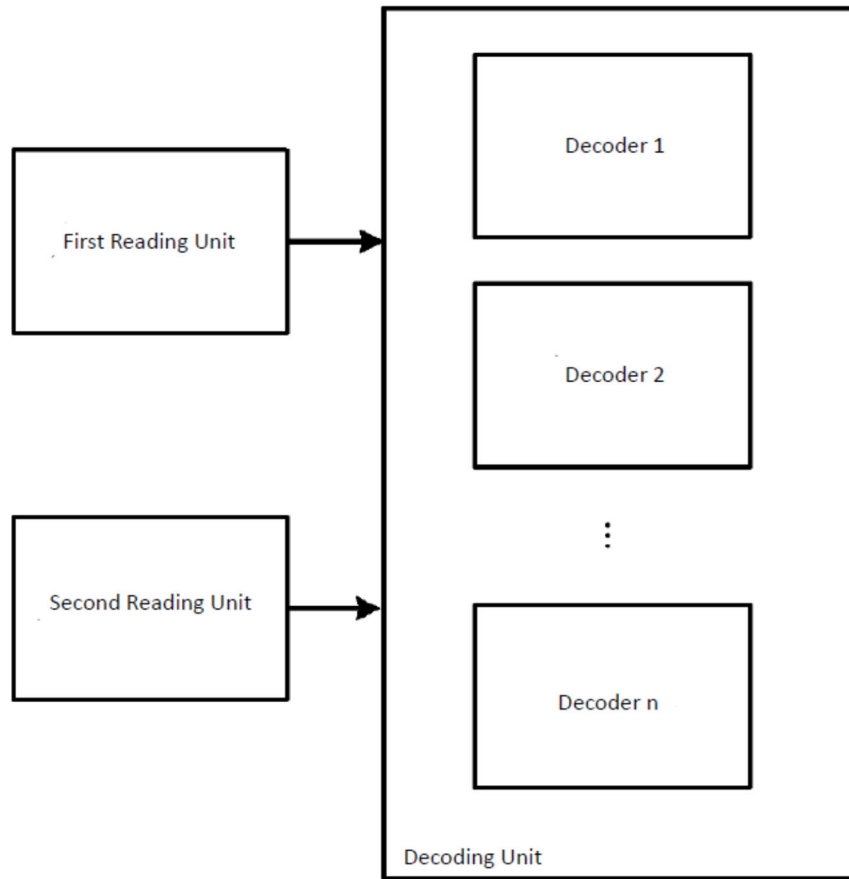
Xiong in combination with VCEG-AJ21 discloses or suggests the additional limitation of claim 4. Acton, ¶102.

As discussed for limitation 1[e], when `ext_16x16_pred_flag` is set to 1, that indicates a *spatial intra partition type* of a 16x16 block. See §VI.A.4.f; Acton, ¶103. Thus, the flag specifies the size (16x16, which is the size of a basic coding unit, see §V) of a *partition of the basic coding unit sized sub-block*. Acton, ¶103.

8. Claim 5

a. 5[pre]: *An apparatus, comprising:*

If limiting, Xiong discloses or suggests 5[pre]. Xiong discloses a “device for intra frame prediction...decoding.” Xiong (Abstract). Xiong’s Figure 9 (below) shows a schematic of a decoding apparatus. Acton, ¶104.



Xiong, Fig. 9, 5 (“Figure 9 is a structural diagram of the intra frame prediction decoding device according to the second embodiment of the present invention.”).

b. 5[a]: at least a memory and one or more processors, wherein the one or more processors are configured to:

Xiong discloses or suggests 5[a]. A POSITA would have understood that Xiong’s “intra frame prediction decoding device” that “includes at least two decoders” and two “reading unit[s]” would include a *memory* and at least one *processor*. Xiong, 12, Fig. 9; Acton, ¶105. Decoders and reading units are computing devices or components that include memory to store instructions for decoding as

well as the video data being decoded. Acton, ¶105. Such computing devices and components process instructions through a microprocessor. *Id.*

- c. **5[b]: *decode picture data for at least a large block in a picture by determining that intra prediction is to be performed for the large block, wherein the large block has a large block size equal to or larger than 32×32 , and the large block size is greater than a basic coding unit size, wherein the intra prediction is signaled for the large block and the one or more processors are configured to:***

Xiong in combination with VCEG-AJ21 discloses or suggests 5[b]. Xiong discloses *decode picture data for at least a large block in a picture* for substantially the same reasons as 1[pre]. See §VI.A.4.a; Acton, ¶106. Xiong in combination with VCEG-AJ21 discloses or suggests *determining that intra prediction is to be performed for the large block, wherein the large block has a large block size equal to or larger than 32×32 , and the large block size is greater than a basic coding unit size* for substantially the same reasons as 1[a]. See §VI.A.4.b; Acton, ¶106.

Xiong further discloses *intra prediction is signaled for the large block*. Xiong, 10 (“Based on the information indicating block size and the intra frame prediction encoding method used for each block, decoding is performed.”); Acton, ¶107.

- d. **5[c]: *decode a binary split signaling syntax element specifying that the large block is further split into four equally sized sub-blocks;***

Xiong in combination with VCEG-AJ21 discloses or suggests 5[c] for substantially the same reasons as 1[b]. See §VI.A.4.c; Acton, ¶108.

- e. **5[d]: *decode an intra prediction mode for the large block that is not further split into four equally sized sub-blocks;***

Xiong in combination with VCEG-AJ21 discloses or suggests 5[d] for substantially the same reasons as 1[c]. See §VI.A.4.d; Acton, ¶109.

- f. **5[e]: *partition the large block into the four equally sized sub-blocks in response to the decoded binary split signaling syntax element specifying that the large block is further split into four equally sized sub-blocks; and***

Xiong in combination with VCEG-AJ21 discloses or suggests 5[e] for substantially the same reasons as 1[d]. See §VI.A.4.e; Acton, ¶110.

- g. **5[f]: *decode a second syntax element indicating a single spatial intra partition type for at least one basic coding unit sized sub-block, the single spatial intra partition type being determinable from among a plurality of spatial intra partition types.***

Xiong in combination with VCEG-AJ21 discloses or suggests 5[f] for substantially the same reasons as 1[e]. See §VI.A.4.f; Acton, ¶111.

- 9. **Claim 6: *The apparatus of claim 5, wherein the one or more processors are configured to decode an intra prediction mode for the basic coding unit sized sub-block.***

Xiong in combination with VCEG-AJ21 discloses or suggests claim 6 for substantially the same reasons as claim 2. See §VI.A.5; Acton, ¶112.

- 10. Claim 7: *The apparatus of claim 5, wherein the one or more processors are configured to decode a further binary split signaling syntax element specifying that at least one of the four equally sized sub-blocks greater than a basic coding unit size is further split into four smaller equally sized sub-blocks.***

Xiong in combination with VCEG-AJ21 discloses or suggests claim 7 for substantially the same reasons as claim 3. *See* §VI.A.6; Acton, ¶113.

- 11. Claim 8: *The apparatus of claim 5, wherein the spatial intra partition type specifies a size of a partition of the basic coding unit.***

Xiong in combination with VCEG-AJ21 discloses or suggests claim 8 for substantially the same reasons as claim 4. *See* §VI.A.7; Acton, ¶114.

12. Claim 11

- a. 11[pre]: *A method of encoding picture data for at least a large block in a picture, the method comprising:***

If limiting, Xiong discloses or suggests 11[pre]. Xiong discloses a “*method...for intra frame prediction encoding.*” Xiong (Abstract). Xiong discloses using macroblocks that can be 32x32 pixels. *Id.*, 6. Xiong further discloses encoding picture data from macroblocks. *Id.*, 5 (“From these candidate encoding results, the one with the best performance indicator is selected as the final encoding result for that block.”). Therefore, Xiong discloses *encoding picture data* for 32x32 blocks (*a large block in a picture*). Acton, ¶115. VCEG-AJ21 also discloses using macroblocks of any size NxN, where “N indicates the maximum pixel size for the enlarged MB and can be increased with power of 2 such as 32, 64 or 128.” VCEG-

AJ21, 3. Thus, the combined teachings of Xiong and VCEG-AJ21 use large blocks that are 32x32, 64x64, or even 128x128. Acton, ¶115.

- b. 11[a]: *determining that intra prediction is to be performed for the large block, wherein the large block has a block size equal to or larger than 32×32, and the large block size is greater than a size of a basic coding unit;***

Xiong in combination with VCEG-AJ21 discloses or suggests 11[a] for substantially the same reasons as 1[a]. *See* §VI.A.4.b; Acton, ¶116.

- c. 11[b]: *encoding a binary split signaling syntax element specifying that the large block is further split into four equally sized sub-blocks;***

Xiong in combination with VCEG-AJ21 discloses or suggests 11[b]. As discussed for limitation 1[b], Xiong teaches *decoding a binary split signaling syntax element specifying that the large block is further split into four equally sized sub-blocks*. *See* §VI.A.4.c; Acton, ¶117. Xiong discloses that the decoding discussed for limitation 1[b] is an example of “decoding” in Xiong’s “in the intra frame prediction encoding/decoding method.” Xiong, 5. As such, a POSITA would have understood Xiong to disclose an encoding method that results in the data that is decoded as described for limitation 1[b]. Acton, ¶117. Therefore, Xiong in combination with VCEG-AJ21 further discloses or suggests encoding the same binary split signaling syntax elements (ext_32x32_pred_flag and ext_64x64_pred_flag) discussed in as being decoded in limitation 1[b]. *See* §VI.A.4.c; Xiong, 8, 9 (Table 1); Acton, ¶117.

d. 11[c]: *encoding an intra prediction mode for the large block that is not further split into four equally sized sub-blocks;*

Xiong in combination with VCEG-AJ21 discloses or suggests 11[c]. As discussed for limitation 1[c], Xiong discloses *decoding an intra prediction mode for the large block that is not further split into four equally sized sub-blocks*. See §VI.A.4.d; Acton, ¶118. Xiong discloses that the decoding discussed for limitation 1[c] is an example of “decoding” in Xiong’s “in the intra frame prediction encoding/decoding method.” Xiong, 5. As such, a POSITA would have understood Xiong to disclose an encoding method that results in the data that is decoded as described for limitation 1[c]. Acton, ¶118. Therefore, Xiong alone or in combination with VCEG-AJ21 further discloses or suggests *encoding an intra prediction mode for the large block that is not further split into equally sized sub-blocks* (ext_32x32_pred_mode and ext_64x64_pred_mode) discussed in as being decoded in limitation 1[c]. See §VI.A.4.d; Xiong, 8, 9 (Table 1); Acton, ¶118.

e. 11[d]: *partitioning the large block into the four equally sized sub-blocks in response to the decoded binary split signaling syntax element specifying that the large block is further split into four equally sized sub-blocks; and*

Xiong in combination with VCEG-AJ21 discloses or suggests 11[d] for substantially the same reasons as 1[d]. See §VI.A.4.e; Acton, ¶119.

- f. **11[e]: *encoding a second syntax element indicating a single spatial intra partition type for at least one basic coding unit sized sub-block, the single spatial intra partition type being determinable from among a plurality of spatial intra partition types.***

Xiong in combination with VCEG-AJ21 discloses or suggests 11[e]. As discussed for limitation 1[e], Xiong teaches *decoding a second syntax element indicating a single spatial intra partition type for at least one basic coding unit sized sub-block, the single spatial intra partition type being determinable from among a plurality of spatial intra partition types*. See §VI.A.4.f; Acton, ¶120. Xiong further teaches *encoding a second syntax element indicating a single spatial intra partition type for at least one basic coding unit sized sub-block, the single spatial intra partition type being determinable from among a plurality of spatial intra partition types* (ext_16x16_pred_flag) discussed in as being decoded in limitation 1[b]. See §VI.A.4.c; Xiong, 8, 9 (Table 1); Acton, ¶120.

- 13. Claim 12: *The method of claim 11, further comprising encoding an intra prediction mode for the basic coding unit sized sub-block.***

Xiong in combination with VCEG-AJ21 discloses or suggests the additional limitation of claim 12. Acton, ¶121.

As discussed for claim 2, Xiong discloses or suggests *decoding an intra prediction mode for the basic coding unit sized sub-block*. See §VI.A.5; Acton, ¶122. Xiong further discloses or suggests *encoding* the same intra prediction mode for the

basic coding unit sized sub-block (ext_16x16_pred_mode) discussed as being decoded in claim 2. *See* §VI.A.5; Xiong, 8, 9 (Table 1); Acton, ¶122.

- 14. Claim 13: *The method of claim 11, further comprising encoding a further binary split signaling syntax element specifying that at least one of the four equally sized sub-blocks greater than a basic coding unit size is further split into four smaller equally sized sub-blocks.***

Xiong in combination with VCEG-AJ21 discloses or suggests the additional limitation of claim 13. Acton, ¶123.

As discussed for claim 3, Xiong in combination with VCEG-AJ21 discloses or suggests *decoding a further binary split signaling syntax element specifying that at least one of the four equally sized sub-blocks greater than a basic coding unit size is further split into four smaller equally sized sub-blocks. See* §VI.A.6; Acton, ¶124. Xiong in combination with VCEG-AJ21 further discloses or suggests encoding the same *further binary split signaling syntax element specifying that at least one of the four equally sized sub-blocks greater than a basic coding unit size is further split into four smaller equally sized sub-blocks (ext_32x32_pred_flag)* discussed as being decoded in claim 3. *See* §VI.A.6; Xiong, 8, 9 (Table 1); Acton, ¶124.

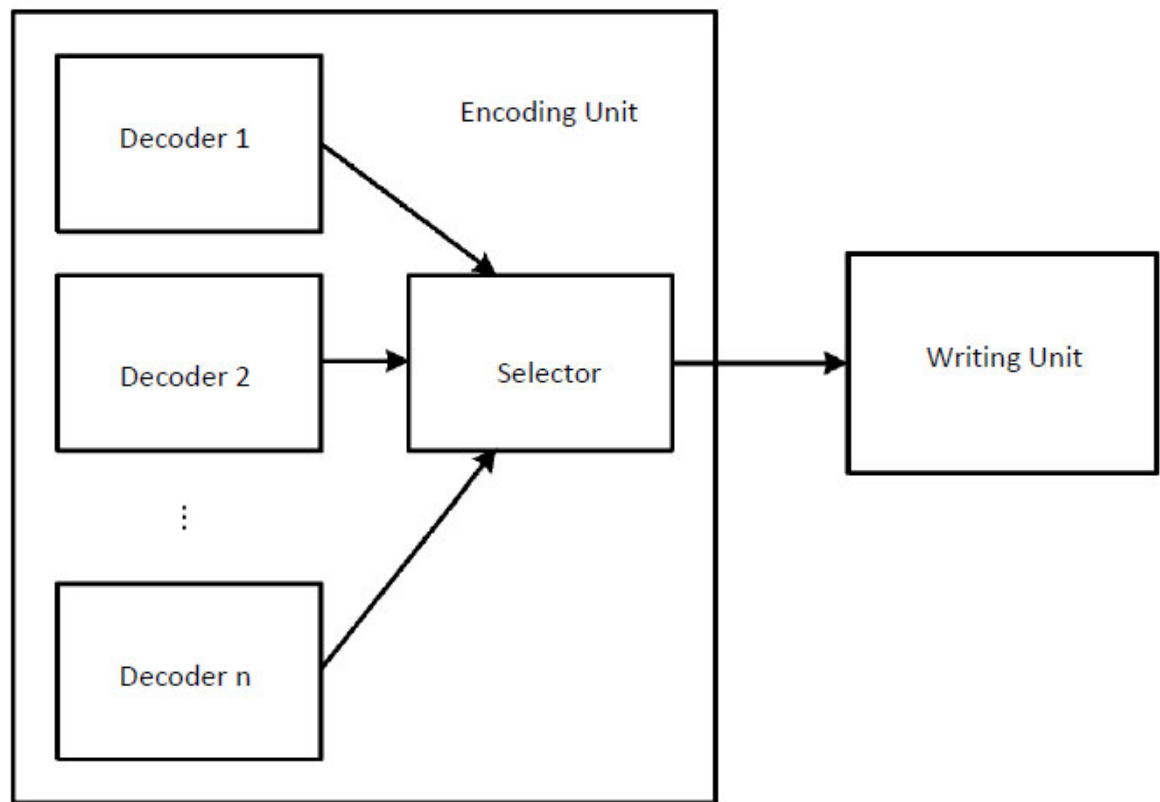
- 15. Claim 14: *The method of claim 11, wherein the spatial intra partition type specifies a size of a partition of the basic coding unit sized sub-block.***

Xiong in combination with VCEG-AJ21 discloses or suggests claim 14 for substantially the same reasons as claim 4. *See* §VI.A.7; Acton, ¶125.

16. Claim 15

a. 15[pre]: *An apparatus comprising:*

If limiting, Xiong discloses or suggests 15[pre]. Xiong discloses a “device for intra frame prediction encoding.” Xiong (Abstract). Xiong’s Figure 8 shows a schematic of an encoding apparatus. Acton, ¶126.



Xiong, Fig. 8, 5 (“Figure 8 is a structural diagram of the intra frame prediction encoding device according to the second embodiment of the present invention;”).

b. 15[a]: *at least a memory and one or more processors, wherein the one or more processors are configured to:*

Xiong discloses or suggests 15[a]. A POSITA would have understood that Xiong discloses a memory and processor for encoding video. Xiong (Abstract), 11, Fig. 8; Acton, ¶127. In particular, a POSITA would have understood the disclosure of an “intra frame prediction encoding device” that comprises “at least two encoders,” a “selector,” and a “writing unit,” would include a memory and at least one processor. Xiong, 11, Fig. 8; Acton, ¶127. Encoders and writing units are computing devices or components that include memory to store instructions for encoding as well as the video data being encoded. Acton, ¶127. Such computing devices and components process instructions through a microprocessor. *Id.*

c. 15[b]: *encode picture data for at least a large block in a picture by determining that intra prediction is to be performed for the large block, wherein the large block has a large block size equal to or larger than 32×32, and the large block size is greater than a basic coding unit size, wherein the intra prediction is signaled for the large block and the one or more processors are configured to:*

Xiong in combination with VCEG-AJ21 discloses or suggests 15[b]. Xiong teaches *encode picture data for at least a large block in a picture*, for substantially the same reasons as 11[pre]. See §VI.A.12.a; Acton, ¶128. Xiong in combination with VCEG-AJ21 discloses or suggests *determining that intra prediction is to be performed for the large block, wherein the large block has a large block size equal*

to or larger than 32×32 , and the large block size is greater than a basic coding unit size, for substantially the same reasons as 11[a]. See §VI.A.12.b; Acton, ¶128.

Xiong further discloses or suggests *intra prediction is signaled for the large block*. Xiong, 7 (“the encoding result of the frame includes information...indicating the intra frame prediction encoding method used for each block.”); Acton, ¶129.

- d. 15[c]: *encode a binary split signaling syntax element specifying that the large block is further split into four equally sized sub-blocks;***

Xiong in combination with VCEG-AJ21 discloses or suggests 15[c] for substantially the same reasons as 11[b]. See §VI.A.12.c; Acton, ¶130.

- e. 15[d]: *encode an intra prediction mode for the large block that is not further split into four equally sized sub-blocks;***

Xiong in combination with VCEG-AJ21 discloses or suggests 15[d] for substantially the same reasons as 11[c]. See §VI.A.12.d; Acton, ¶131.

- f. 15[e]: *partition the large block into the four equally sized sub-blocks in response to the binary split signaling syntax element specifying that the large block is further split into four equally sized sub-blocks; and***

Xiong in combination with VCEG-AJ21 discloses or suggests 15[e] for substantially the same reasons as 1[d] and 11[d]. See §§VI.A.4.e, VI.A.12.e; Acton, ¶132.

- g. 15[f]: *encode a second syntax element indicating a single spatial intra partition type for at least one basic coding unit sized sub-block, the single spatial intra partition type being determinable from among a plurality of spatial intra partition types.***

Xiong in combination with VCEG-AJ21 discloses or suggests 15[f] for substantially the same reasons as 11[e]. See §VI.A.12.f; Acton, ¶133.

- 17. Claim 16: *The apparatus of claim 15, wherein, the one or more processors are configured to encode an intra prediction mode for the basic coding unit sized sub-block.***

Xiong in combination with VCEG-AJ21 discloses or suggests claim 16 for substantially the same reasons as claim 12. See §VI.A.13; Acton, ¶134.

- 18. Claim 17: *The apparatus of claim 15, wherein, the one or more processors are configured to encode a further binary split signaling syntax element specifying that at least one of the four equally sized sub-blocks greater than a basic coding unit size is further split into four smaller equally sized sub-blocks.***

Xiong in combination with VCEG-AJ21 discloses or suggests claim 17 for substantially the same reasons as claim 13. See §VI.A.13; Acton, ¶135.

- 19. Claim 18: *The method of claim 15, wherein the spatial intra partition type specifies a size of a partition of the basic coding unit sized sub-block.***

Xiong in combination with VCEG-AJ21 discloses or suggests claim 18 for substantially the same reasons as claims 4 and 14. See §§VI.A.7, VI.A.15; Acton, ¶136.

B. Ground 1B: Xiong in view of VCEG-AJ21 and H.264

1. H.264

H.264 is a standard for “higher compression of moving pictures for various applications such as videoconferencing, digital storage media, television broadcasting, Internet streaming, and communication.” H.264, i. The revision of H.264 referenced herein (EX-1010) was approved by the ITU-T Study Group 16 in November 2007 and was publicly available by July 1, 2008, making it prior art under §102(b). H.264, i; Munford, ¶¶7-36.⁹ For example, Ms. Munford explained that “the final, approved edition of ITU-T H.264 2007 was posted to the ITU website and made available for purchase no later than May 29, 2008,” “and the substance of that document [through amendments to March 2005 version] was publicly available no later than September 2007.” Munford, ¶¶35-36.

⁹ To the extent PO argues that EX-1010 was not publicly available by July 1, 2008, the substance of that reference was available in the form of Amendments 1 and 2 to the 2005 version of H.264, which were publicly available no later than September 2007. EX-1015 (“H.264 (03/2005)”); Munford, ¶36, Appendices H26402 (EX-1016 (“Amendment 1”)), H26403 (EX-1017 (“Amendment 2”)). Alternative citations to H.264 (03/2005), Amendment 1, and/or Amendment 2 are provided as footnotes throughout this Petition.

The '876 patent refers to H.264 as an example of a “modern video coding standard[]” that “employ[s] various coding modes to efficiently reduce the correlations in the spatial and temporal domains.” EX-1001, 1:28-42; Acton, ¶¶137-138.

H.264 includes Table 7-11 (below), titled Macroblock types for I slices. It indicates the allowed macroblocks for intra predicted frames. H.264, 90-91¹⁰; Acton, ¶139.

¹⁰ H.264 (03/2005), 86.

Table 7-11 – Macroblock types for I slices

mb_type	Name of mb_type	transform_size_8x8_flag	MbPartPredMode (mb_type, 0)	Intra16x16 Pred Mode	CodedBlock Pattern Chroma	CodedBlock Pattern Luma
0	I_NxN	0	Intra_4x4	na	Equation 7-34	Equation 7-34
0	I_NxN	1	Intra_8x8	na	Equation 7-34	Equation 7-34
1	I_16x16_0_0_0	na	Intra_16x16	0	0	0
2	I_16x16_1_0_0	na	Intra_16x16	1	0	0
3	I_16x16_2_0_0	na	Intra_16x16	2	0	0
4	I_16x16_3_0_0	na	Intra_16x16	3	0	0
5	I_16x16_0_1_0	na	Intra_16x16	0	1	0
6	I_16x16_1_1_0	na	Intra_16x16	1	1	0
7	I_16x16_2_1_0	na	Intra_16x16	2	1	0
8	I_16x16_3_1_0	na	Intra_16x16	3	1	0
9	I_16x16_0_2_0	na	Intra_16x16	0	2	0
10	I_16x16_1_2_0	na	Intra_16x16	1	2	0
11	I_16x16_2_2_0	na	Intra_16x16	2	2	0
12	I_16x16_3_2_0	na	Intra_16x16	3	2	0
13	I_16x16_0_0_1	na	Intra_16x16	0	0	15
14	I_16x16_1_0_1	na	Intra_16x16	1	0	15
15	I_16x16_2_0_1	na	Intra_16x16	2	0	15
16	I_16x16_3_0_1	na	Intra_16x16	3	0	15
17	I_16x16_0_1_1	na	Intra_16x16	0	1	15
18	I_16x16_1_1_1	na	Intra_16x16	1	1	15
19	I_16x16_2_1_1	na	Intra_16x16	2	1	15
20	I_16x16_3_1_1	na	Intra_16x16	3	1	15
21	I_16x16_0_2_1	na	Intra_16x16	0	2	15
22	I_16x16_1_2_1	na	Intra_16x16	1	2	15
23	I_16x16_2_2_1	na	Intra_16x16	2	2	15
24	I_16x16_3_2_1	na	Intra_16x16	3	2	15
25	I_PCM	na	na	na	na	na

H.264, 91 (Table 7-11).¹¹ Table 7-11 of H.264 is substantively identical to Table 1 in the '876 patent. EX-1001, 2:50-3:18. According to the '876 patent, this table “shows the detailed signaling for macroblock types for Intra coded slices (I slices).” *Id.*, 2:31-35.

According to H.264, some of the items in Table 7-11 are signaled, including “mb_type.” H.264, 90.¹² Other items in Table 7-11 are not signaled, but are determined from the signaled elements, including “MbPartPredMode,” “Intra16x16PredMode,” “CodedBlockPatternChroma,” and “CodedBlockPatternLuma.” *Id.*, 92 (“To each Intra_16x16 prediction macroblock, an Intra16x16PredMode is assigned, which specifies the Intra_16x16 prediction mode, and values of CodedBlockPatternLuma and CodedBlockPatternChroma are assigned as specified in Table 7-11.”)¹³; Acton, ¶140.

2. Motivation to Combine Xiong, VCEG-AJ21, and H.264

A POSITA would have been motivated to combine H.264 with the combination of Xiong and VCEG-AJ21 discussed above (*see* §VI.A.3) to extend H.264’s macroblock structure to larger macroblocks using Xiong’s hierarchical

¹¹ H.264 (03/2005), 87.

¹² H.264 (03/2005), 86.

¹³ H.264 (03/2005), 88.

block structure and would have had a reasonable expectation of success in doing so. Acton, ¶141.

Each of Xiong and VCEG-AJ21 expressly build upon H.264. Xiong, (Background Art); VCEG-AJ21, 1 (Abstract). In particular, each reference describes extending the size of the 16x16 macroblocks used in H.264. Acton, ¶142.

Xiong discloses that “the macroblock here is not necessarily limited to a 16×16 macroblock; it can also be of other sizes, such as a 32×32 macroblock.” Xiong, 6. Xiong goes on to disclose that the “specific prediction encoding methods” may refer to the methods of H.264/AVC. Xiong, (Background Art), 6; Acton, ¶143.

VCEG-AJ21 proposes “extend[ing] the architecture of H.264 codec with enlarged MB sizes.” VCEG-AJ21, 1. VCEG-AJ21 recognizes the limitation of H.264’s 16x16 macroblock and the benefits of larger macroblocks, especially for High Definition (“HD”) video. *Id.* (“It may not be efficient for the HD or beyond HD since the MB size of 16x16 pixels is relatively small for high resolution video contents.”), 2 (“It implies that relatively large block modes are more advantageous to the high resolution videos in terms of the Rate-distortion optimization.”). In particular, VCEG-AJ21 discloses expanding macroblocks to any pixel size that is a power of 2, such as 32, 64, or 128. *Id.*, 3 (“Table 1 lists the H.264 elements that must be modified by enlarging the MB sizes. As aforementioned, this is not the exhaustive

list. In Table 1, N indicates the maximum pixel size for the enlarged MB and can be increased with power of 2 such as 32, 64 or 128.”); Acton, ¶144.

A POSITA would have understood the teachings of Xiong and VCEG-AJ21 to build on the work of H.264 by adding enlarged block sizes using routine engineering. Acton, ¶¶145-146.

3. Claim 9: *The apparatus of claim 5, wherein at least one of a spatial intra partition type table and an intra prediction mode table is pre-stored and used by the one or more processors to decode picture data for the large block.*

Xiong in view of VCEG-AJ21 and H.264 discloses or suggests the additional limitation of claim 9. Acton, ¶147.

H.264’s Table 7-11 (shown in part below, and annotated) includes values for an intra prediction mode (“Intra16x16PredMode,” column outlined in red) associated with the coded “mb_type” (outlined in blue) H.264, 91 (Table 7-11)¹⁴; Acton, ¶148.

¹⁴ H.264 (03/2005), 87.

mb_type	Name of mb_type	transform_size_8x8_flag	MbPartPredMode (mb_type, 0)	Intra16x16 Pred Mode	CodedBlock PatternChroma	CodedBlock PatternLuma
0	I_NxN	0	Intra_4x4	na	Equation 7-34	Equation 7-34
0	I_NxN	1	Intra_8x8	na	Equation 7-34	Equation 7-34
1	I_16x16_0_0_0	na	Intra_16x16	0	0	0
2	I_16x16_1_0_0	na	Intra_16x16	1	0	0
3	I_16x16_2_0_0	na	Intra_16x16	2	0	0
4	I_16x16_3_0_0	na	Intra_16x16	3	0	0

When decoding in H.264, the decoder takes the encoded mb_type from the macroblock layer bitstream (*see* H.264 at Table 9-11) and decodes into the integer mb_type according to the binary encoding. H.264, 220-21 (Table 9-11), 248 (Table 9-36).¹⁵ The indexed row allows the processor of the decoder to know the Intra16x16PredMode, which is an *intra prediction mode*. H.264, 90 (“Each table presents the value of mb_type, the name of mb_type, the number of macroblock partitions used (given by the NumMbPart(mb_type) function), the prediction mode of the macroblock (when it is not partitioned) or the first partition (given by the

¹⁵ H.264 (03/2005), 208-209, 223 (Table 9-27).

MbPartPredMode(mb_type, 0) function) and the prediction mode of the second partition (given by the MbPartPredMode(mb_type, 1) function”)¹⁶; Acton, ¶149.

A POSITA would have understood this table to be implemented in software or firmware (*intra prediction mode table*) that, when executed by the system processor (*the one or more processors*), assigns the correct value for the intra prediction mode of the incoming block. Acton, ¶150. That software or firmware would be stored on the system prior to execution (*pre-stored and used...to decode picture data*) on a block (*the large block*). *Id.*, ¶¶150-151.

A POSITA would have further understood that such a pre-stored intra prediction mode table would be used in the combination of Xiong and H.264 to provide the intra prediction modes for large blocks. Acton, ¶152. In particular, Table 7-11 would be supplemented with additional rows to include the additional partition types disclosed in Xiong and VCEG-AJ21 (32x32 and 64x64), along with the intra partition modes associated with those partition types. *Id.*

4. **Claim 10: *The apparatus of claim 5, wherein at least one of a spatial intra partition type table and an intra prediction mode table is received by the one or more processors using one or more high level syntax elements and is used by the one or more processors to decode picture data for the large block.***

H.264 discloses or suggests the additional limitation of claim 10. Acton, ¶153.

¹⁶ H.264 (03/2005), 86.

H.264 discloses high-level syntax elements for transmitting scaling matrices from the encoder to the decoder. H.264, 40, 42, 68, 76.¹⁷ In particular, flags “seq_scaling_matrix_present_flag” and “pic_scaling_matrix_present_flag” are set to 1 to indicate that a scaling matrix is being transmitted. *Id.*; Acton, ¶154. The actual matrix is transmitted through a set of scaling lists, encoded using the “scaling_list” function. H.264, 40-41;¹⁸ Acton, ¶154.

These matrices allow user customized scaling factors to be sent from the encoder to the decoder for perceptual-based scaling. Acton, ¶¶155-157 (citing documents authored by Gary Sullivan, introducing “Fidelity Range Extensions,” a set of amendments to H.264, incorporated into the November 2007 version of H.264 (EX-1010)).

H.264 thus provides seq_scaling_matrix_present_flag and scaling_list (*high level syntax elements*) transmitted from the encoder to the decoder (*received by the one or more processors*) that represent matrices of scaling factors used to decode blocks (*an intra prediction mode table...used by the one or more processors to decode picture data for the large block*). Acton, ¶158.

¹⁷ H.264 (03/2005), 39, 41, 65, 72; Amendment 2, 17.

¹⁸ H.264 (03/2005), 40.

To the extent PO argues that a custom scaling matrix is not *an intra prediction mode table*, it would have been obvious to a POSITA to transmit matrices disclosed in H.264 to send a table of intra prediction modes. Acton, ¶159. Using the approach outlined in H.264—indicating a matrix is to follow using a flag, and then sending the matrix if the flag is set to 1—for a table of intra prediction modes would require only routine engineering to replicate or reuse code for transmittal of scaling matrices for intra prediction mode tables. *Id.* Doing so would have provided the benefit of customization of intra prediction modes, creating both additional efficiency and enhanced resolution. *Id.*

To the extent PO argues that *received by the one or more processors* includes the processor reading instructions from memory, this limitation is alternatively met by the processor loading Table 7-11 from memory. Acton, ¶160. As discussed for claim 9, Table 7-11 of H.264 is an intra prediction mode table. *See* §VI.B.3. This table is received by the decoding processor when the software code implementing that table is read from memory. H.264, 293 (“Hypothetical Reference Decoder” disclosed in Annex C).¹⁹ The hypothetical reference decoder is able to take the table index `mb_type` and determine the prediction mode from the `mb_type`. Acton, ¶160.

¹⁹ H.264 (03/2005), 264.

VII. CONCLUSION

Inter Partes Review of the Challenged Claims is respectfully requested.

VIII. MANDATORY NOTICES

A. Real Party in Interest

Petitioners identify themselves as real parties in interest.

B. Related Matters

To the best of Petitioners' knowledge, the '876 patent has not been involved in any other matters.

C. Notice of Counsel and Service Information

LEAD COUNSEL
Jessica Kaiser (Reg. No. 58,937) kaiser-ptab@perkinscoie.com PERKINS COIE LLP 1900 Sixteenth Street, Suite 1400 Denver, CO 80202-5255 Telephone: (303) 291-2300
BACK-UP COUNSEL
Robin Brewer, Reg. No. 60,951 PERKINS COIE LLP 505 Howard Street Suite 1000 San Francisco, CA 94105 Tel: (415) 344-7000 Brewer-ptab@perkinscoie.com Jon Carter, Reg. No. 75,145

PERKINS COIE LLP

1155 Avenue of the Americas 22nd Floor.

New York, NY 10036-2711

Tel: (212) 262-6900

Carter-ptab@perkinscoie.com

David Pekarek Krohn, Reg. No. 69,443

PERKINS COIE LLP

33 East Main Street, Suite 201

Madison, WI 53703-3095

Telephone: (608) 663-7460

pekarekkrohn-ptab@perkinscoie.com

Petitioners consent to electronic service. All services and communications to the attorneys listed above may be sent to:

Amazon-InterDigital-IPR@perkinscoie.com

D. Power of Attorney

A power of attorney is filed herewith according to 37 C.F.R. §42.10(b).

Respectfully submitted,

/ Jessica Kaiser /

Jessica Kaiser

Reg. No. 58,937

Attorney for Petitioner

PERKINS COIE LLP

1900 Sixteenth Street, Suite 1400

Denver, CO 80202-5255

Date: January 9, 2026

CERTIFICATE OF WORD COUNT UNDER 37 CFR §42.24(d)

Pursuant to 37 C.F.R. §42.24(a), Petitioners hereby certify that portions of the above-captioned Petition for *inter partes* review of U.S. Patent No. 11,936,876, in accordance with and reliance on the word count provided by the word-processing system used to prepare this Petition, that the number of words in this paper is 10,773. Pursuant to 37 C.F.R. §42.24(a), this word count is in compliance and excludes the table of contents, table of authorities, mandatory notices under §42.8, certificate of service, certificate of word count, appendix of exhibits, and any claim listing. This word count was prepared using Microsoft Word.

Respectfully submitted,

/ Jessica Kaiser /

Jessica Kaiser
Reg. No. 58,937
Attorney for Petitioner

Date: January 9, 2026

PERKINS COIE LLP
1900 Sixteenth Street, Suite 1400
Denver, CO 80202-5255

CERTIFICATE OF SERVICE

The undersigned hereby certifies that true copies of the Petition for *inter partes* review of U.S. Patent No. 11,936,876 and supporting materials (Exhibits and Power of Attorney) were served via overnight delivery on the Patent Owner at the correspondence address of record as listed on Patent Center:

Patent Docketing
200 Bellevue Parkway
Suite 300
Wilmington, DE 19809

Respectfully submitted,

/ Jessica Kaiser /

Jessica Kaiser
Reg. No. 58,937
Attorney for Petitioner

Date: January 9, 2026