

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

AMAZON.COM, INC., AMAZON.COM SERVICES, LLC,
Petitioner

v.

NOKIA TECHNOLOGIES OY,
Patent Owner

Case: IPR2024-00691
U.S. Patent No. 8,050,321

DECLARATION OF DR. IAIN RICHARDSON

TABLE OF CONTENTS

I.	INTRODUCTION	1
II.	EDUCATIONAL BACKGROUND AND QUALIFICATIONS	2
III.	MATERIALS REVIEWED AND TIME OF INVENTION	5
IV.	LEGAL STANDARDS	5
1.	Overview	5
2.	Patent Claims.....	5
3.	Burden of Proof for Invalidity.....	7
4.	Claim Construction.....	7
5.	Person of Ordinary Skill in the Art	7
6.	Anticipation	8
7.	Obviousness.....	9
V.	PERSON OF ORDINARY SKILL IN THE ART (POSITA).....	12
VI.	CLAIM CONSTRUCTION.....	12
VII.	BACKGROUND	13
1.	The '321 Patent	13
2.	The Prior Art	19
a.	MPEG-1	19
b.	Kim.....	29
VIII.	The Grounds in the Petition do not anticipate or render obvious the Challenged claims.....	36
1.	Petitioner Has Not Established That The Challenged Claims Are Unpatentable In Light Of Ground 1 (MPEG-1).....	36
a.	Petitioner fails to demonstrate that limitations 8[a], 10[a], and 11[a] are obvious over MPEG-1	36
b.	Petitioner fails to demonstrate that limitations 8[b], 10[b], and [11[b] are obvious over MPEG-1	51
c.	Petitioner fails to demonstrate that limitations 8[d], 10[d], and 11[d] are obvious over MPEG-1 because MPEG-1	

	does not teach or suggest a method for decoding, a video decoder or a computer program product for decoding, resetting the identifier value as required.....	57
2.	Petitioner Has Not Established That The Challenged Claims Are Unpatentable In Light Of Ground 2—Kim.....	61
a.	Petitioner fails to demonstrate that limitations 8[a], 10[a], and 11[a] are obvious over Kim.....	62
b.	Petitioner fails to demonstrate that limitations 8[b], 10[b], and [11[b] are obvious over Kim.....	67
c.	Petitioner fails to demonstrate that limitations 8[c], 10[c], and 11[c] are obvious over Kim.....	70
d.	Petitioner fails to demonstrate that limitations 8[d], 10[d], and 11[d] are obvious over Kim because Kim does not teach or suggest a decoder resetting the identifier value as required	73
3.	Petitioner Has Not Established That The Challenged Claims Are Unpatentable In Light Of Ground 3.....	75
a.	Petitioner fails to demonstrate that Claim 9 is obvious over MPEG-1 in combination with Yagasaki.....	75
IX.	SIGNATURE.....	76

I, Dr. Iain Richardson, declare as follows:

I. INTRODUCTION

1. I have been retained on behalf of Nokia Technologies OY and asked to provide opinions in respect of Inter Partes Review (IPR) proceeding IPR2024-00691 which relates to U.S. Patent No. 8,050,321, the '321 Patent.

2. This Declaration summarises the independent opinions that I have formed to date which are based on the material currently available to me. I reserve the right to modify my opinions, if necessary, based on further review and analysis of information that I receive subsequent to the filing of this Declaration, including in response to any positions taken by Petitioners or their experts that I have not been provided yet. Further, I incorporate the opinions I set forth in my Declaration in support of Patent Owner's Preliminary Response (Ex. 2001) and have reproduced portions of those opinions in this Declaration for ease of reference.

3. My findings, as explained below, are based on my study, experience, knowledge and background in the fields discussed below. I am informed by my education and my extensive experience in the fields of video compression and multimedia coding and communications. I have also relied on my review and analysis of the prior art, information provided to me and information I have independently reviewed.

II. EDUCATIONAL BACKGROUND AND QUALIFICATIONS

4. Details of my professional qualifications and background are summarized below and are set out in detail in my Curriculum Vitae, attached to this declaration.

5. I am Director and co-founder of Vcodex BV.

6. I have over 30 years of experience related to video and image compression and coding. I am qualified by education and experience to testify as an expert with respect to subject matter in the fields of video, image, multimedia and general-purpose data encoding, decoding, compression, decompression, and transmission.

7. I received a Master of Engineering (M.Eng.) degree in Electronic and Electrical Engineering from Heriot-Watt University in Edinburgh, Scotland in 1990. I received a Doctor of Philosophy (Ph.D.) degree in Video Compression from Robert Gordon University in Aberdeen, Scotland in 1999. I worked as a Digital Signal Processing (“DSP”) Hardware Designer with GEC Avionics Ltd. from 1990 to 1993.

8. In 1993, I assumed a post as a Lecturer, then Reader and eventually a Full Professor in the field of image and video compression in the School of Engineering at Robert Gordon University. In 2009, I was awarded the position of Honorary Professor with Robert Gordon University.

9. At Robert Gordon University, I founded and ran an image communication technology research laboratory. I carried out original research in the field of data, image, and video compression, initiated and managed research projects, and supervised research students.

10. I am the author of five books and over 70 journal and conference papers on video and image coding and communication, including two widely cited books on the H.264 / MPEG-4 industry standards for video compression. In my 2002 book, *Video Codec Design*, I discussed in detail the efficient implementation of image and video compression on software and hardware platforms.

11. In 1995, during my employment at Robert Gordon University, I co-founded and became Technical Director, equivalent of Chief Technology Officer, at 4i2i Communications Limited. I held this position until 2000. In my role at 4i2i, I was responsible for and engaged in the design of software and hardware modules for compression and communication systems. The designs included video processing algorithms in software and hardware for implementation on processing devices and the design of compression algorithms such as block-based transforms, run-length coding, and variable-length coding in software and hardware. A modified version of one of my video compression hardware designs was successfully used by NASA to stream video from the International Space Station.

12. In 2007, I founded Vcodex Limited and currently serve as the CEO, through which I provide technical and strategic advice to technology companies particularly relating to video, image, and data compression for streaming.

13. In 2009, I founded Onecodec Limited and served as CEO, leading the company's development of innovative video, image, and multimedia data compression and storage software and systems. Onecodec's technology and business activities were merged with Vcodex Limited in 2015.

14. I have previously prepared expert reports and testified in a number of cases concerning video and image compression, graphics compression, general-purpose data compression, and multimedia streaming.

15. I am a named inventor on 11 patents or patent applications, including U.S. and non-US patents and applications.

16. Because of my background, training, and experience, I am qualified as an expert in the technical areas relevant to the '321 patent. My company Vcodex BV is being compensated for my independent analysis as an expert with respect to this inter partes review (IPR) proceeding, but neither my compensation nor Vcodex BV's compensation is contingent in any way on the content of my analysis or the outcome of this proceeding. I have no financial interest in Petitioner (Amazon.com, Inc., Amazon.com Services LLC) or Patent Owner (Nokia Technologies Oy) and I have no financial interest in the challenged patent.

III. MATERIALS REVIEWED AND TIME OF INVENTION

17. The opinions contained in this Declaration are based on the documents that I reviewed, the Petition (included all materials cited therein), my professional judgment, as well as my education, experience, and knowledge regarding technologies related to the '321 Patent.

IV. LEGAL STANDARDS

1. Overview

18. I am not an attorney or a legal expert, and I do not opine on matters of law. All the legal standards, which are listed in this section and which I applied in reaching the conclusions expressed throughout this Declaration, were provided to me and explained to me by counsel. In preparing this Declaration, I have endeavored to faithfully apply these legal standards. I do not intend to offer opinions on the law. Below, I describe the legal standards that I applied in analyzing the claims of the '321 Patent.

19. I have been informed by counsel that the challenger of a patent's validity bears the burden of proving invalidity of the patent.

2. Patent Claims

20. I have been informed by counsel that a patent's claims describe the invention made by the inventors and describe what the patent owner owns and what the owner may prevent others from doing.

21. I have been informed by counsel that claims may describe products, such as machines or chemical compounds, or processes for making or using a product. Claims are usually divided into parts or steps called limitations.

22. I have been informed by counsel that there are two types of claims: independent claims and dependent claims. An independent claim sets forth all the requirements that must be met in order to be covered by that claim. Thus, it is not necessary to look at any other claim to determine what an independent claim covers. A dependent claim does not itself recite all of the requirements of the claim, but refers to another claim for some of its requirements. In this way, a dependent claim depends on another claim. The law considers a dependent claim to incorporate all of the requirements of the claims to which it refers.

23. I have been informed by counsel that that a dependent claim then adds its own additional requirements. To determine what a dependent claim covers, it is necessary to look at both the dependent claim and any other claims to which it refers.

24. I have been informed by counsel that in a case where a dependent claim incorporates the limitations of an independent claim upon which it depends, if a reference fails to disclose any of the limitations of the independent claim, the claims which depend on that independent claim cannot be made invalid by such a reference. As a result, any of the references at issue in this proceeding that fail to

disclose all of the limitations of an independent claim necessarily also fail to disclose all of the limitations of claims which depend on that independent claim.

3. Burden of Proof for Invalidity

25. I have been informed by counsel and I understand that the party challenging the validity of an issued U.S. patent in an IPR bears the burden of proving invalidity through facts supported by a preponderance of the evidence. I have been informed that a “preponderance of the evidence” means evidence sufficient to prove that a fact is more likely true than it is not.

4. Claim Construction

26. I have been informed by counsel and I understand that the first step in determining validity is to properly construe the claims. I have further been informed by counsel that, in this proceeding, the Board must determine the scope of the claims by using the same claim construction standard that United States District Courts use. Under this standard, the words of a claim are given their “ordinary” and “customary” meaning as would have been understood by a person of ordinary skill in the art at the time of the invention. I have been informed that such a person looks to the claims, specification, and file history in understanding the meaning of the claim terms. I have applied this understanding in this declaration.

5. Person of Ordinary Skill in the Art

27. I have been informed by counsel and I understand that a person having

ordinary skill in the art may be defined by factors such as, when considered at the time of the alleged invention: (a) the levels of education and experience of the inventor and other persons actively involved in the field; (b) the types of problems encountered in the field; (c) prior art solutions to those problems; (d) rapidity with which innovations are made; and (e) the sophistication of the technology.

6. Anticipation

28. I have been informed by counsel and I understand that a patent claim can be found invalid as anticipated if each and every element of the claim, as properly construed, is found either explicitly or inherently in a single prior art reference. If the single prior art reference does not disclose explicitly or inherently any one claim limitation, then that claim is not anticipated. I have further been informed that the single prior art reference cannot merely disclose each element; rather, it must disclose all of the elements as arranged in the claim. I have been informed that for a prior art reference to “inherently” disclose something, the reference must necessarily function in accordance with, or include the claimed limitations, to anticipate. The missing descriptive matter cannot merely probably or possibly include the claimed limitations. I have further been informed by counsel that, with respect to an anticipation analysis, it is improper to combine separate, distinct embodiments of a single prior art reference.

29. I have been informed that for a reference to be anticipatory, the

reference must contain an enabling disclosure. I have been informed that merely naming or describing the subject matter is insufficient if it cannot be produced without undue experimentation.

7. Obviousness

30. I have been informed by counsel that a patent claim can be found invalid as obvious if the differences between the claim and the prior art references would have been obvious to a person of ordinary skill in the art at the time of the invention, taking into account (1) the scope and content of the prior art, (2) the differences between the prior art and the claims, (3) the level of ordinary skill in the art, and (4) any objective indicia of non-obviousness. I have further been informed by counsel that a claim can be found to be obvious in light of either a single prior art reference or multiple prior art references. I have been informed that references qualify as prior art for an obviousness determination only when they are analogous to the claimed invention. Two separate tests define the scope of analogous prior art: (1) whether the art is from the same field of endeavor, regardless of the problem addressed; and (2) if the reference is not within the field of the inventor's endeavor, whether the reference still is reasonably pertinent to the particular problem with which the inventor is involved.

31. I have been informed that to be obvious in light of a single prior art reference or multiple prior art references, there must be a reason either to modify the

single prior art reference or to combine the two or more prior art references to achieve the claimed invention. I have been informed by counsel that this reason may come from a teaching, suggestion, or motivation to combine which may come from the reference or references themselves, the knowledge or “common sense” of one skilled in the art, or from the nature of the problem to be solved, and may be explicit or implicit from the prior art as a whole.

32. I have been informed by counsel that the combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results.

33. I have further been informed by counsel that there is no suggestion to combine, however, if a reference teaches away from its combination with another source. A reference may be said to teach away when a person of ordinary skill, upon reading the reference, would be discouraged from following the path set out in the reference, or would be led in a direction divergent from the path that was taken by the applicant, or if it suggests that the line of development flowing from the reference’s disclosure is unlikely to be productive of the result sought by the applicant. If, when combined, the references would produce a seemingly inoperative device, then they teach away from their combination. I have been informed that it is improper to combine references where the references teach away from their combination.

34. I have been informed by counsel that the degree of teaching away will depend on the particular facts; in general, a reference will teach away if it suggests that the line of development flowing from the reference's disclosure is unlikely to be productive of the result sought by the applicant. I have been informed that a reference teaches away, for example, if (1) the combination would produce a seemingly inoperative device, or (2) the references leave the impression that the product would not have the property sought by the applicant.

35. I have been informed by counsel that if the teachings of a prior art reference would lead one skilled in the art to make a modification that would render another prior art device inoperable, then such a modification would generally not be obvious. I have been informed that if a proposed modification would render the prior art device unsatisfactory for its intended purpose, then the proposed modification would generally not be obvious.

36. I have been informed by counsel that it is improper to use hindsight in making the obviousness determination.

37. I have been informed by counsel that some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness must be made. I have been informed that, for example, there may be no rational underpinning to support the legal conclusion of obviousness where systems are already complete

and the party challenging validity seeks to mix and match components, which each exist in the separate systems, from these complete systems.

V. PERSON OF ORDINARY SKILL IN THE ART (POSITA)

38. I understand that Petitioner has identified that the POSITA “would have had a (1) Bachelor’s degree in electrical engineering, computer engineer, computer science, or a comparable field of study such as physics, and (2) approximately two to three years of practical experience with video encoding/decoding . . . [and] [a]dditional experience can substitute for the level of education, and vice-versa.” For the purposes of my analysis for this Declaration, I have used the definition that the Petitioner has identified, as it is my opinion that any potential difference of opinion I may have with Petitioner on the level of ordinary skill in the art is not material to the issues with their arguments.

39. All of my opinions in this Declaration are from the perspective of a person of ordinary skill in the art, as I have defined it above and during the relevant time frame. During that time frame, I possessed at least the qualifications of a POSITA, as defined above.

VI. CLAIM CONSTRUCTION

40. I understand that claim terms are typically given their ordinary and customary meaning, as would have been understood by a person of ordinary skill in the art at the time of the alleged invention. In considering the meaning of the claims,

I also understand that one must consider the claim language, specification, prosecution history, and the knowledge of a person of ordinary skill in the art at the relevant time.

VII. BACKGROUND

1. The '321 Patent

41. The '321 Patent relates to an improvement in video decoding that enables a decoder to identify where to start a decoding process, such as when a user randomly selects a frame to start viewing video, for example when joining a stream part-way through or when rewinding or fast-forwarding. This is known as random access in video coding, which the '321 Patent is directed to improving. '321 Patent, 3:52-56, 3:62-4:4, 4:48-60. The claims of the '321 Patent (including those at issue here) are directed to specific problems and solutions disclosed in the specification.

42. Specifically, “[a] problem in the numbering of image frames arises when the user wishes to start browsing a video file in the middle of a video sequence. Such situations occur for example when the user wishes to browse a locally stored video file backward or forward or to browse a streaming file at a particular point; when the user initiates the playback of a streaming file from a random point, or when a video files that is to be reproduced is detected to contain an error that interrupts the playback or requires the playback to be resumed from a point following the error.” '321 Patent, 11:11-20. The Patent identifies this as a problem because “[w]hen the

browsing of a video file is resumed from a random point after previous browsing, discontinuity typically occurs in the image numbering. The decoder typically interprets this as unintentional loss of image frames and unnecessarily tries to reconstruct the image frames suspected as lost.” ’321 Patent, 11:20-25.

43. The Patent explains that “this [problem] can be avoided in the decoder by defining an initiation image in an independently decodable Group of Pictures GOP that is activated at the random point of the video file, and the number of the initiation image is set to zero. . . . Consequently, when activated at a random point, the **decoder preferably sets the identifier** of the first image frame, preferably an I-frame, of the independently decodable sub-sequence.” ’321 Patent, 11:26-40, emphasis added.

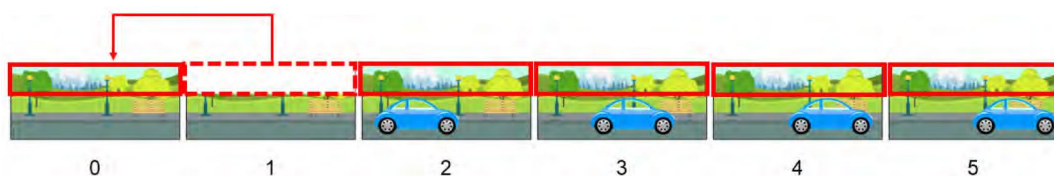
44. The ’321 Patent claims such an improved method “that enable[s] detection of image frames from which a decoder can start the decoding process.” ’321 Patent, 4:8-10. “The invention is based on the idea of encoding a video sequence comprising an independent sequence of image frames, wherein at least one reference picture is predictable from at least one previous image frame that is earlier than the previous reference image frame in decoding order.” ’321 Patent, 4:16-20.

45. I have provided illustrations below to further describe the random-access problems addressed by the ’321 Patent. As shown below, a video encoder

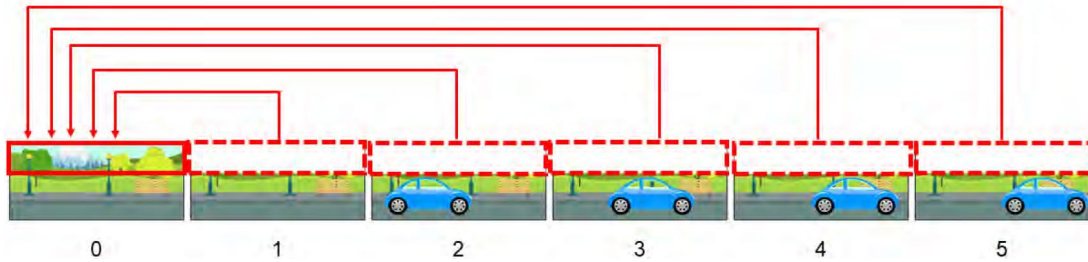
may exploit temporal redundancy, allowing redundant content in a series of video images (e.g., the red boxes below) to be predicted from earlier reference pictures that include the same content, improving compression by avoiding the need to encode redundant content in later pictures.



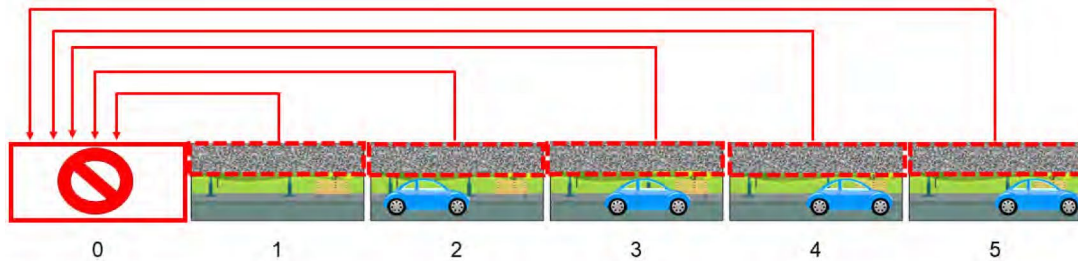
46. As shown below, the top of Picture 1 (dashed rectangle) is predicted from the top of Picture 0 (solid rectangle), exploiting the fact that Picture 0 contains the video content necessary for Picture 1.



47. In this example, the same information from Picture 0 can be used to improve the compression of a series of pictures. The top of Picture 0 (solid rectangle) is only transmitted once, and Pictures 1-5 predict that content (arrows) from Picture 0.

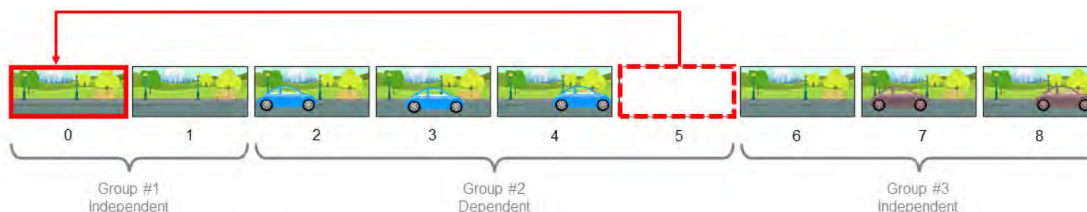


48. Temporal prediction can however lead to unwanted effects if errors occur during transmission. For example, decoding errors occur when the reference picture (Picture 0) is missing, and the predicted pictures (Pictures 1-5) cannot reference the video content.



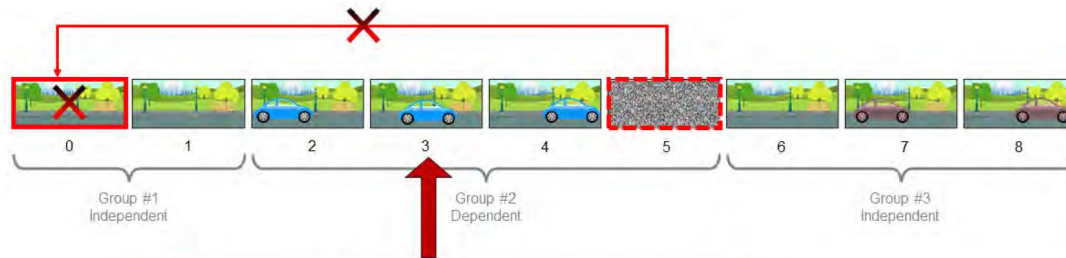
49. The '321 Patent discusses the concept of independent sequences or independently decodeable groups of pictures. In the illustrative example below, a video sequence (Pictures 0-8) includes two independent sequences: Group #1 (Pictures 0-1) and Group #3 (Pictures 6-8). It also includes a dependent sequence: Group #2 (Pictures 2-5). An independent sequence makes use of predictions only from pictures within the same independent sequence, whereas a dependent sequence includes at least one picture that refers to another picture outside of that dependent

sequence. Because Picture 5 in Group #2 is predicted from Picture 0 in Group #1, Group #2 is a dependent sequence.



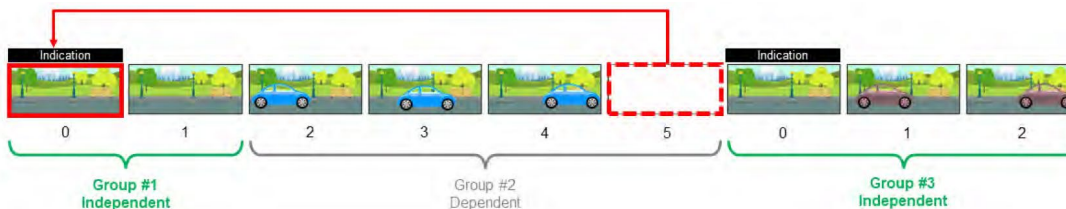
50. In the above example, when a user randomly accesses a specific picture within the middle of a video sequence, the decoder must determine where it can begin decoding so that the decoded pictures can be displayed without any prediction errors. As the '321 Patent explains, prior art methods of identifying sources for prediction (such as reference picture selection) can pose challenges for random access, since certain pictures may be predicted from previous pictures that are not available at the decoder. '321 Patent, 3:52-56, 3:62-4:4. As explained in the '321 Patent, these dependencies present challenges with random access because the decoder does not know from which image frames it can begin decoding. *Id.*

51. The example below illustrates this problem. The user selects Picture 3 as the random-access point. If the decoder starts decoding at Picture 3, an error occurs when decoding Picture 5 because Picture 0 was not previously decoded and so is not available at the decoder for predicting Picture 5.



52. The '321 Patent claims a specific solution to address these prior art random-access problems. '321 Patent, 4:48-60. Specifically, the identification of an initiation frame or picture and resetting the numbering of that frame to zero solves the issue so that the decoder does not inadvertently interpret that certain frames are missing. '321 Patent, 11:11-40

53. According to the '321 Patent claims, a decoder decodes an indication of the first image frame of an independent sequence. As an illustration, below are two independent sequences (Groups #1 and #3), each including an indication (black box) of a first image frame that informs the decoder where decoding can begin to avoid such problems with random access.



54. For example, if the user selects Picture 3 in Group #2 as the random-access point (red arrow), decoding begins at the first image of one of the independent sequences, either Picture 0 in Group #1 (left-hand green arrow) or Picture 0 in Group #3 (right-hand green arrow).



2. The Prior Art

a. MPEG-1

55. The MPEG-1 Part 2 Standard (ISO/IEC 11172-2, hereinafter MPEG-1) defines a bitstream and a decoding process. Encoders can produce compliant bitstreams that are decodable by MPEG-1's specified decoding process. An MPEG-1 decoder must produce identical results to the defined decoding process.

56. The MPEG-1 Standard specifies a decoding process that supports various possible techniques (i.e. encoding options) for encoding MPEG-1 compliant

bitstreams. An encoder can choose such options during encoding so long as the bitstream conforms to the standard as discussed above. Examples include picture type (an encoder can choose I, P or B picture types for coding each frame, within the constraints of the standard), macroblock type (an encoder can choose I, P or B macroblock types, within the constraints of the standard), motion vectors, high-level syntax element values, etc.

57. The encoder makes decisions and chooses options whilst encoding a bitstream, as described above. The decoder processes the received MPEG-1 bitstream and decodes the bitstream according to the syntax element values in the bitstream. For example, the `temporal_reference` syntax element, part of MPEG-1's Picture Layer, is assigned in sequence to pictures in display order (syntax element `temporal_reference` at the Picture Layer, a 10-bit fixed length binary codeword, Ex 1005 at 38). An encoder chooses a value of Temporal Reference and signals it in the bitstream for an encoded Picture or Frame. The decoder receives the coded bitstream, parses this syntax element (i.e. reads the 10-bit `temporal_reference` syntax element value from the bitstream), and decodes it according to the standard (Ex 1005 at 44). Thus, an MPEG-1 decoder does not *choose* or *reset* a `temporal_reference` value; rather, an MPEG-1 decoder can only *decode* a `temporal_reference` value that was assigned by the encoder.

58. An MPEG-1 bitstream is organized into Groups of Pictures (GOPs),

each beginning with a syntax element `group_start_code` (Ex 1005 at 43) and including thereafter (in encoding order) an I-picture followed by zero or more P, B or I-pictures. A typical GOP structure consists of (in display order) an I-picture, optionally preceded by one or more B-pictures, followed by (a) P-pictures spaced at intervals and optionally (b) B-pictures between successive I- or P-pictures (Ex 1005 at 32).

59. An I-picture is intra coded, i.e. it can be decoded without reference to any other coded pictures. A P-picture is inter predicted from the previous I- or P-picture in the bitstream. A B-picture is inter predicted from either or both of the previous and following I- or P-pictures in display order.

60. A representative example is shown in Ex. 1005 at p71. Note that in coding (bitstream) order, B-pictures are placed after their respective previous and following I- or P-pictures. For example, B-pictures 1 and 2 in the following example are inter predicted from I-picture 0 and P-picture 3. In bitstream order, B-pictures 1 and 2 follow I-picture 0 and P-picture 3. This means that a decoder has to store only two previously coded pictures (0 and 3) prior to decoding and displaying each B-picture (1 and 2 in this example).

Note that because of the picture dependencies, the bitstream order, i.e. the order in which pictures are transmitted, stored, or retrieved, is not the display order, but rather the order which the decoder requires them to decode the bitstream. An example of a sequence of pictures, in display order, might be:

I	B	B	P	B	B	P	B	B	P	B	B	I	B	B	P	B	B	P
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18

Figure D.3 -- Typical sequence of pictures in display order

whereas the bitstream order would be as shown below:

I	P	B	B	P	B	B	P	B	B	I	B	B	P	B	B	P	B	B
0	3	1	2	6	4	5	9	7	8	12	10	11	15	13	14	18	16	17

Figure D.4 -- Typical sequence of pictures in bitstream order

61. A GOP may be encoded without the use of B-pictures, i.e. only using I- and P-pictures. An example of such a GOP is shown in Ex. 1005 Fig. D.17:

The original concept of a group of pictures was a set of pictures that could be coded and displayed independently of any other group. In the final version of this part of ISO/IEC 11172 this is not always true, and any B-pictures preceding (in display order) the first I-picture in a group may require the last picture in the previous group in order to be decoded. Nevertheless encoders can still construct groups of pictures which are independent of one another. One way to do this is to omit any B-pictures preceding the first I-picture. Another way is to allow such B-pictures, but to code them using only backward motion compensation.

Property 3. From a coding point of view, a concisely stated property is that a group of pictures begins with a group of pictures header, and either ends at the next group of pictures header or at the next sequence header or at the end of sequence, whichever comes first.

Some examples of groups of pictures are given below:

I																		
I	P	P																
I	B	P	B	P														
B	B	I	B	P	B	P												
B	B	I	B	B	P	B	B	P	B	B	P							
B	I	B	B	B	B	P	B	I	B	B	I	I						

Figure D.17 -- Examples of groups of pictures in display order

These examples illustrate what is possible, and do not constitute a suggestion for structures of groups of pictures.

Ex. 1005, 84-85.

62. This GOP structure begins with an I-picture, followed by a series of P-pictures. Each P-picture is predicted from the immediately preceding I- or P-picture, e.g. consider the following example:

I₀P₁P₂P₃P₄.....

63. In this example, P-picture P₁ is inter predicted from I-picture I₀. P-picture P₂ is inter predicted from P-picture P₁, and so on.

64. One important aspect of MPEG-1 related to Petitioner's arguments is MPEG-1's "closed_gop flag." MPEG-1's closed_gop flag is part of the "Group of Pictures layer," i.e. it is part of the syntax layer that applies to an entire GOP, MPEG-1 at 2.4.3.4 ("Group of pictures layer"), Ex 1005 at 43.

closed_gop -- This is a one-bit flag which may be set to "1" if the group of pictures has been encoded without motion vectors pointing to the previous group of pictures.

65. As specified in MPEG-1, when decoding an MPEG-1 video bitstream, the closed_gop flag is received in the group of pictures header. MPEG-1 defines "Closed GOPs" or "closed groups [of pictures]" as follows:

Closed GOP

A one bit flag follows the time code. It denotes whether the group of pictures is open or closed. Closed groups can be decoded without using decoded pictures of the previous group for motion compensation, whereas open groups require such pictures to be available.

A typical example of a closed group is shown in figure D.18a.

I	B	B	P	B	B	P	B	B	P	B	B	P
0	1	2	3	4	5	6	7	8	9	10	11	12

(a) closed group

B	B	I	B	B	P	B	B	P	B	B	P	B	B	P
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14

(b) open or closed group

Figure D.18 -- Example groups of pictures in display order

A less typical example of a closed group is shown in figure D.18b. In this example, the B-pictures which precede the first I-picture must use backward motion compensation only, i.e. any motion compensation must be based only on picture number 2 in the group.

If the closed_gop flag is set to 0 then the group is open. The first B-pictures that precede the first I-picture in the group may have been encoded using the last picture in the previous group for motion compensation.

MPEG-1 at D.5.2 ("Group of pictures"), Ex 1005 at 86.

66. The “one bit flag” referred to in the above extract is the `closed_gop` flag. As the `closed_gop` flag is a flag in the group of pictures header, it is a syntax element that applies to an entire GOP and not any one specific coded frame. The `closed_gop` flag is therefore not an indication of any specific image frame.

67. For example, in Figure D.18 above, the value of the `closed_gop` flag (0 or 1) defines a property of the entire GOP, such as the 13-picture GOP shown in Fig. D.18a or the 15-picture GOP shown in Fig. D.18b. More specifically, as the cited text above explains, the value of `closed_gop` defines certain properties of the coding of certain B-pictures in the GOP, namely whether or not B-pictures preceding the first I-picture may or may not use forward motion compensation (i.e. motion compensation from the preceding P- or I-picture, which is located in the preceding GOP).

68. In contrast, other syntax elements in an MPEG-1 video bitstream pertain to a specific coded picture, namely syntax elements found at the Picture Layer and lower syntax layers. MPEG-1 specifies that “The picture layer contains all the coded information for one picture”. Ex. 1005 at 87.

69. Thus, the `closed_gop` flag is not an indication of a particular image frame in the GOP and at most, indicates certain properties of certain B-pictures in the GOP.

70. Another aspect of MPEG-1 that is important with respect to Petitioner's argument is "Temporal Reference," encoded as a syntax element `temporal_reference` that is part of the Picture Layer syntax. Unlike the `closed_gop` flag, `temporal_reference` is a syntax element that is associated with a specific coded picture (Ex. 1005 at 38).

71. MPEG-1 explains that "Temporal Reference" can be used to define "the order in which the pictures must be displayed." For a given group of pictures (whether open or closed), the first picture in display order is assigned a Temporal Reference of zero:¹

¹ Temporal reference is also set to zero when there are more than 1024 pictures in a GOP. Ex. 1005, 87. This is a limitation of the number of bits used to express `temporal_reference` in the coded video bitstream.

D.5.3.2 Temporal reference

The Temporal Reference is a ten-bit number which can be used to define the order in which the pictures must be displayed. It may be useful since pictures are not transmitted in display order, but rather in the order which the decoder needs to decode them. The first picture, in display order, in each group must have Temporal Reference equal to zero. This is incremented by one for each picture in the group.

Some example groups of pictures with their Temporal Reference numbers are given below:

Example (a) in display order	I	B	P	B	P								
	0	1	2	3	4								
Example (a) in decoding order	I	P	B	P	B								
	0	2	1	4	3								
Example (b) in display order	B	B	I	B	B	P	B	B	P	B	B	P	
	0	1	2	3	4	5	6	7	8	9	10	11	
Example (b) in coded order	I	B	B	P	B	B	P	B	B	P	B	B	
	2	0	1	5	3	4	8	6	7	11	9	10	
Example (c) in display order	B	I	B	B	B	B	P	B	I	B	B	I	I
	0	1	2	3	4	5	6	7	8	9	10	11	12
Example (c) in coded order	I	B	P	B	B	B	B	I	B	I	B	B	I
	1	0	6	2	3	4	5	8	7	11	9	10	12

Figure D.19 -- Examples of groups of pictures and temporal references

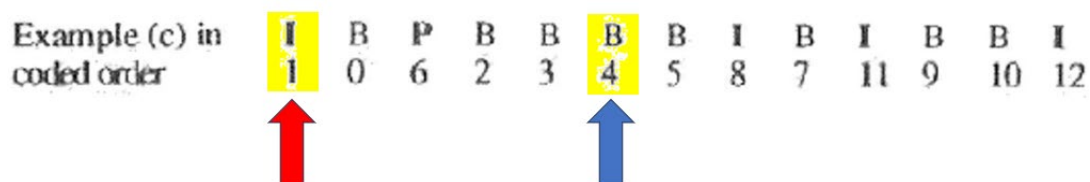
Ex. 1005, 87-88.

72. In an MPEG-1 coded video bitstream, the order in which pictures must be displayed (the display order) is in general different from the “coded order” of the pictures, which defines the order that the pictures are decoded. As illustrated in Figure D.19 and discussed earlier, the coded order for a picture can be and typically is different from the temporal order.

73. Further, MPEG-1 discloses prior art problems (not the claimed solution) because it does not fully enable random access at a user-specified playback point, and therefore does not meet the claim limitations that enable such random access. MPEG-1 describes that after a user specifies a random-access point, the decoder looks for the *next* sequence start code, group of pictures (GOP) start code,

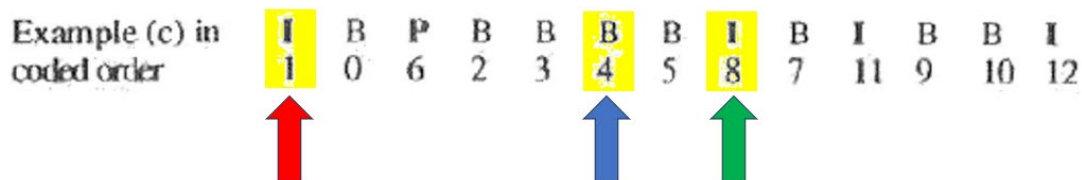
or picture start code which introduces an I-picture (collectively, “start codes”) to start decoding. Ex. 1005, 43, 87, 120. All frames prior to the next start code are discarded, and decoding and playback starts from that next start code (rather than from a specific identified frame), which is not the user-specified random-access point.

74. Applying the above-described teachings of MPEG-1 to Example (c) of Figure D.19 (which applies equally to all of MPEG-1) highlights the issues with MPEG-1 and confirms that MPEG-1 does not disclose the claimed invention of the ’321 Patent. As shown below, the first frame of the GOP in decoding order is I1 (red arrow) and the user selects, for example, frame B4 (blue arrow) as the playback point.

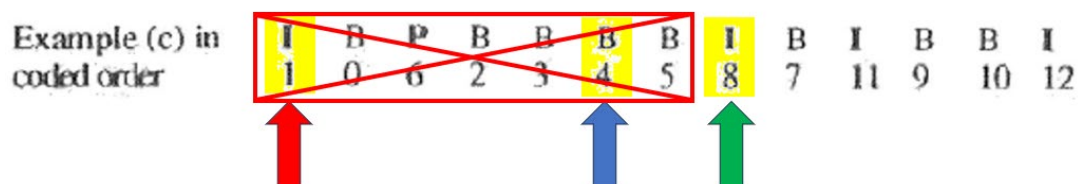


75. Thereafter, MPEG-1 starts decoding from frame I8 (green arrow, below), which is not the first frame I1 (red arrow) of the independent sequence. MPEG-1 starts decoding from I8 (green arrow) because MPEG-1 teaches that after a user specifies a random-access point, the decoder looks for the next sequence start code, group of pictures (GOP) start code, or picture start code which introduces an

I-picture (collectively, “start codes”) to start decoding. Ex. 1005, 43, 87, 120. In Example (c), the next start code is the picture start code which introduces an I-picture, I8. *See id.*



76. The decoder discards all frames until frame I8 and starts decoding and playback from I8 because MPEG-1 teaches that all prior frames are discarded and decoding and playback starts from that next start code (rather than from a specific identified frame), which is not the user-specified random-access point. Ex. 1005, 43, 87, 120.



77. Further, MPEG-1 does not disclose the decoder resetting any identifier value of the indicated first image frame. In the above example, regardless of whether I1 or I8 is the “indicated” first image frame, MPEG-1 never teaches a decoder resetting any identifier value of those frames.

78. Importantly, MPEG-1 does not identify decoding any indication of a first image frame of an independent sequence, starting the decoding from the indicated first image frame, and resetting an identifier value of the indicated first image frame. When the user selects a playback point, MPEG-1 teaches to discard all frames and start decoding at a start code (not an indicated first image frame). MPEG-1 suffers from the same random-access problems discussed above, and the '321 Patent acknowledged MPEG-1's GOP concept as a problem in random access and sought to address it. '321 Patent, 2:34-55, 3:52-56. Consequently, MPEG-1 fails to disclose several limitations of the asserted claims that enable random access from the user-specified point, as I explain further below.

b. Kim

79. Kim is directed to a Time Lapse recording/reproduction device and method. Specifically, Kim describes recording, compressing and reproducing intermittently recorded image data (i.e. single image frames spaced at intermittent time intervals), which is significantly different to the problems and solutions of the '321 Patent. Further, Kim is not related to video streaming. For example, Dr. Havlicek acknowledged that the device of Kim is “not your average consumer video,” “not 30-frames-a-second consumer video,” and the input video (camera video) “might run at 30fps, frames per second, but Kim's device isn't going to record that.” Ex. 2005, 39:14-40:22. Examples of where this type of technology may be

applied is in banks or other security-sensitive areas. In such implementations, single image frames may be recorded intermittently and then stored, for example on magnetic tapes. These individual images can be individually and incrementally reproduced when needed; but Kim does not enable random access of streaming video and therefore is not directed to the issues of the '321 patent. (Ex 1006 at 1:8-56).

80. More specifically, Kim describes a system that (a) encodes and stores individual video frames without prediction (I-frame) or with prediction (P-frame) from previously stored video frames (Ex 1006 at 3:33-4:45) and (b) identifying, decoding and displaying a selected frame of video data on a user display (Ex 1006 at Fig 4, 5:8-6:8).

81. According to Kim, considering Ex 1006 FIG. 5A, each “P-picture frame P1, P2, P3 has a current address (“CAD”) indicating the current address of the P-picture, and an I-picture reference address (“IAD”) indicating the address of the corresponding (or previous) I-frame.” Kim at 4:46-52 (shown below):

The bit stream generated and transmitted to the recording/
reproducing unit **140** as described above has a logical
structure as shown in, e.g., FIG. 5A. As shown therein, each
P-picture frame **P1, P2, P3** has a current address CAD
50 indicating the current address of the P-picture, and an
I-picture reference address IAD indicating the address of the
corresponding (or previous) I-frame. If image signals are

82. The logical address, CAD, is disclosed in Kim Figure 5A as

incrementing by one, while the IAD for an I-frame is always 0000 and the IAD for a P-frame references the logical address of the I-frame that it relates to.

83. More specifically, the IAD indicates the address of the corresponding (or previous) I-frame. For example, in Step S34, Kim's controller "detects the address of the corresponding I-picture frame (e.g. data stored in the area **(3)** of the bit stream B2 shown in Fig. 6) from the reproduced image data output from the de-MUX **131**." Ex. 1006, 5:41-44. Area **(3)** of the bit stream B2 shown in Fig. 6 is "the address of the corresponding I-picture frame," *id.* at 4:59-65, which Fig. 6 confirms is the physical address of the I-frame (not the logical address).

84. When Kim begins decoding, as stated above, Kim starts with the user-selected frame. And Dr. Havlicek acknowledged, and I agree, that recording and reproducing onto an optical disk, which is physical media, would be to utilize the physical address of what that information is stored. Ex. 2005, 55:16-56:4 (Q. Kim discloses recording onto an optical disk, which is a physical media, right? A. Correct. Q. And the way one would seek into a physical media would be, for example, going into the physical address of where that information is stored, correct? A. At the low level of the device, yes, I agree.").

85. So, when beginning decoding of the user-selected frame, Kim does not disclose using the logical address. Instead, if anything, the physical address is used.

86. For an I-frame, no other address is required (and Step 34 is never encountered). For a P-frame, Kim does disclose using an address for the I-frame, but the physical address. Not the logical address that Petitioner points to.

87. Figure 6 further illustrates why the CAD is an address that is not according to a numbering scheme. We know this because the CAD is dependent on an image size, which is variable and not predictable. For example, Figure 6 shows that the address for the second P picture is $100000 + N$, where N is the Byte length of the I frame (B1). Specifically, N is not determined according to a numbering scheme, but to the encoded size of the picture (which depends on the coding type and size of the picture data) that is necessarily variable and unpredictable.

88. Petitioner relies on FIG. 5A and Kim at 4:46-52 to show “an indication of at least one image frame, which is the first image frame, in decoding order, of an independent sequence, wherein all motion-compensated temporal prediction references of the independent sequence refer only to image frames within said independent sequence.” Petition at 46.

89. In my opinion, Petitioner is incorrect to suggest that FIG. 5A and 4:46-52 of Kim show the claimed “indication”.

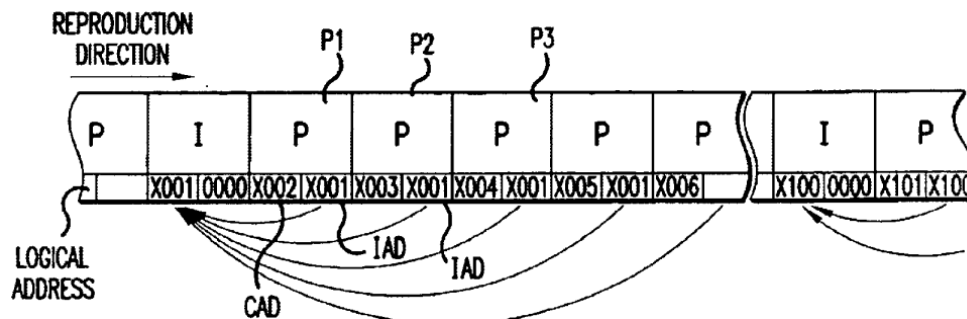


FIG.5A

90. Petitioner points to an IAD of 0000 as the claimed “indication” and argues FIG. 5A discloses an independent sequence because the P pictures in FIG. 5A refer to the first I frame. But the IAD=0000 value associated with the I frame is not the indication of the first image frame in decoding order because the user-selected frame (a P frame in Petitioner’s argument, not an I frame) is decoded first. Moreover, Kim’s IAD address is not reset by the decoder, as required by the claims. In Kim’s method, the IAD address for an I-picture is “zero or absent” and Kim discloses it as such referring to Fig. 6. Kim never teaches or hints that the decoder resets the IAD address. In fact, Kim never teaches any instance wherein the decoder alters the recorded bitstream.

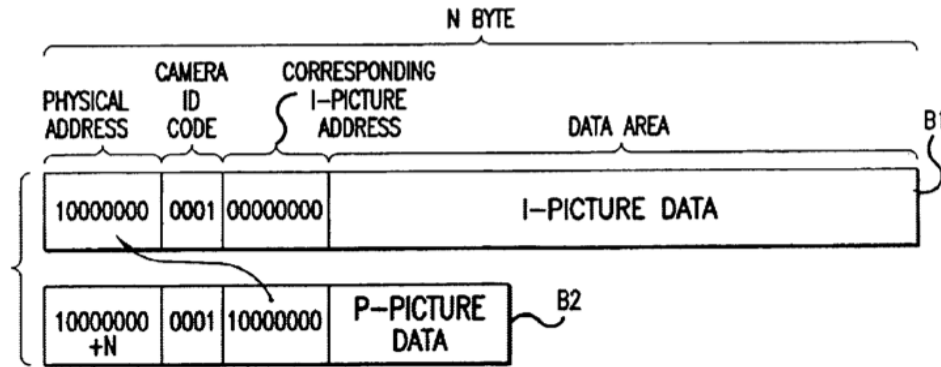


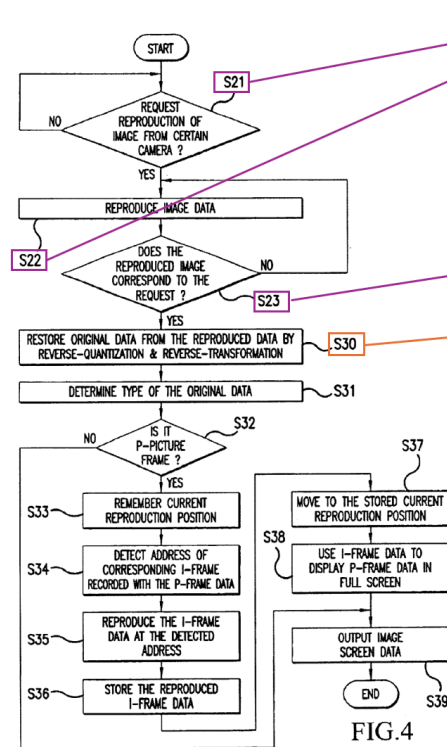
FIG.6

91. Since an I-picture in Kim does not require a reference to another I-picture, the IAD field in an I-picture can be “zero or absent” and a POSITA reading the disclosure would understand that the field could take any value. Kim, 4:66-5:1.

92. As the IAD field for an I-picture can be “zero or absent” in Kim, it does not have any functional use and therefore a POSITA would recognize that it is not disclosed as an indication that is “decoded from [a] video sequence”. I note that the only support for Petitioner’s statement that “A POSITA would have understood that decoding the bitstream in Figure 5A would include decoding... the IAD values” (Petition at 47) is Dr Havlicek’s statement that “Kim ... explains the process for decoding this bitstream”, referring to Kim at 5:8-6:8 and Fig. 4 (Ex. 1003 at 152-154). However, Kim Fig. 4 and 5:8-6:8 says nothing about decoding or even detecting the IAD field for an I-picture. Thus there is no independent support for Petitioner’s statement (Petition at 47).

93. Kim's decoder, shown in Fig 4, decodes and displays a single picture, based on a user request for one image, as shown by Fig. 4.

94. If the requested image is an I-picture, the corresponding coded image data is simply read from the optical disk, decoded and displayed. As shown in the annotated figure below (Figure 4 alongside column 5 of Kim), the first image data decoded is the image data for the frame the user selected (step S22), and Kim is unaware until a later point (step S31) whether the frame is an I-frame or a P-frame.



As shown in Steps S21 and S22 of FIG. 4, if the user requests reproduction of an image data from a particular camera or channel among the intermittent image data recorded on the optical disc 1, the controller 150 controls the recording/reproducing unit 140 to reproduce the corresponding image data from the optical disc 1. The image data reproduced from the optical disc 1 is multiplexed data, which is demultiplexed by the de-MUX 131. By demultiplexing, the camera (channel) ID code CHID is separated from the reproduced image data and output to the controller 150. Based on the camera ID code CHID, the controller 150 determines whether or not the reproduced image data corresponds to the selected camera or channel, in Step S23. This ensures that the correct image data is processed to comply with the user's request. If incorrect image data has been reproduced, the controller 150 controls the recording/reproduction unit 140 to reproduce the correct image data from the optical disc 1. In Step S30, the image data corresponding to the user's request is transmitted and decoded in the decoder 111. The decoding process of the decoder 111 involves reverse-quantization and reverse-transformation to restore the original image data prior to data compression carried out during recording. Then the property (type) of the currently restored image data is determined in Step S31 by the reproducing DSP 121, and the determination result is transmitted to the controller 150. More specifically, if the currently restored image data is detected to be P-picture frame data in Step S32, the controller 150 stores the current reproduction position in Step S33. Then in Step S34, the controller 150 detects the address of the corresponding I-picture frame (e.g., data stored in the area (3) of the bit stream B2 shown in FIG. 6) from the reproduced image data output from the de-MUX 131, and reproduces based on the detected address the corresponding (or previous) I-picture frame data recorded on the optical disc 1 by controlling the recording/reproduction unit 140.

In Step S36, the reproduced corresponding I-picture frame data is processed by the reproducing DSP 121, and decoded by the decoder 111 to be stored in the frame memory 101. The controller 150, after Step S36, controls the recording/reproducing unit 140 to move the video heads to the original recording position stored in Step S33.

Kim, Figure 4 and 5:1-41.

95. If the requested image is a P-picture, the P-picture data is first read and decoded, then the corresponding I-picture is located, read and decoded (using IAD

and CAD values), then the P-picture is reconstructed (by adding the decoded P-picture and I-picture data), then displayed. (Ex 1006 at Fig 4 and 5:8-6:8). Thus, the IAD=0000 value cannot be the claimed indication of the first image frame in decoding order, since the P-picture is the first image frame in decoding order.

96. In sum, Kim does not identify decoding any indication of a first image frame of an independent sequence, starting the decoding from the indicated first image frame, and resetting an identifier value of the indicated first image frame. When the user selects an image for output, Kim teaches to start decoding at the user-selected image (not an indicated first image frame of an independent sequence). Nor does Kim teach that the decoder resets the IAD value of the user-selected image. Kim's output of a single image frame does not solve the random-access problems discussed above. Thus, Kim fails to disclose several limitations of the asserted claims that meet the '321 Patent's aims of avoiding unnecessary decoding errors and enabling random access from a user-specified point.

VIII. THE GROUNDS IN THE PETITION DO NOT ANTICIPATE OR RENDER OBVIOUS THE CHALLENGED CLAIMS

- 1. Petitioner Has Not Established That The Challenged Claims Are Unpatentable In Light Of Ground 1 (MPEG-1).**
 - a. Petitioner fails to demonstrate that limitations 8[a], 10[a], and 11[a] are obvious over MPEG-1**

97. Each of limitations 8[a], 10[a], and 11[a] are written with similar

language and require “[decoding/decode] from the video sequence an indication of at least one image frame, which is the first image frame, in decoding order, of an independent sequence, wherein all motion-compensated temporal prediction references of the independent sequence refer only to image frames within said independent sequence.” In my opinion, Petitioner² has failed to show that MPEG-1 teaches or suggests these limitations.

98. I understand Petitioner asserts two distinct arguments as to how the “[decoding/decode] from the video sequence an indication of at least one image frame, which is the first image frame, in decoding order, of an independent sequence limitations” are taught or suggested—pointing to MPEG-1’s closed GOP (closed_gop flag) or, alternatively, to MPEG-1’s Temporal Reference (temporal_reference syntax element).

i. MPEG-1’s Closed GOP Flag Is Not the Claimed “Indication”

99. *First*, Petitioner first asserts that MPEG-1’s closed_gop flag teaches

² Petitioner put forth an expert declaration that in my view is substantively identical to the limitation-by-limitation analysis. Therefore, references to “Petitioner” as used in these claim sections refers to both Petitioner’s argument and the substantively identical expert declaration substance.

or suggests “[decoding/decode] from the video sequence an indication of at least one image frame, which is the first image frame, in decoding order, of an independent sequence.” In my opinion, Petitioner is incorrect for at least two reasons: (1) the closed_gop flag is a single-bit flag that only indicates whether the *entire* GOP is closed or open; it does not identify any image frame and therefore not an indication according to the claims; and (2) the closed_gop flag does not indicate *which frame is the first*. In my view it is not sufficient that every closed GOP necessarily has a first image frame because the claims require indicating “which is the first image frame.” I explain this further below.

100. As an initial matter, MPEG-1’s closed_gop flag is only an indication of whether or not the current group of pictures (i.e. the GOP that immediately follows the closed_gop flag) is open or closed; it is not an indication of *any* image frame (whether or not the image frame is first in decoding order). Ex. 1005, 37; The closed_gop flag is present in a structural level of the coded bitstream data (the GOP Layer) that relates to an entire group of coded pictures and not to any particular frame:

0.5 Structure of the coded video bitstream

This part of ISO/IEC 11172 specifies a syntax for a coded video bitstream. This syntax contains six layers, each of which either supports a signal processing or a system function:

Layers of the syntax	Function
Sequence layer	Random access unit: context
Group of pictures layer	Random access unit: video
Picture layer	Primary coding unit
Slice layer	Resynchronization unit
Macroblock layer	Motion compensation unit
Block layer	DCT unit

Ex. 1005, 16.

101. Syntax elements in the GOP layer apply to an entire GOP, i.e. a collection of coded pictures, whereas syntax elements in the Picture layer and below apply to a specific coded picture. This means that when a decoder processes the bitstream and decodes the `closed_gop` flag, it does not decode an indication about any particular coded frame, but instead receives information that pertains to an entire GOP. There is no further information provided about a particular frame, or identifying a particular frame, when the decoder processes the `closed_gop` flag. For this reason alone, the `closed_gop` flag cannot indicate any specific coded frame in an independent sequence, let alone the first one.

102. Further, in my opinion, a POSITA would understand “indication” in the claims to refer to information in an encoded bitstream that necessarily informs decoders of something. Here, the indication is “of at least one image frame, which is the first image frame, in decoding order, of an independent sequence.” A POSITA would understand that this “indication” cannot merely be another element (in this

case a particular image frame) that may coincidentally appear alongside certain information in the bitstream (in this case, the `closed_gop` flag). That is, in video coding, an indication in a bitstream must inform a decoder of something with certainty.

103. Logically, an indication takes the following form in a coded bitstream such as an MPEG-1 bitstream:

If *indication* is present, then *thing indicated* is confirmed necessarily to the decoder.

104. However, Petitioners' MPEG-1 Ground relies on a `closed_gop` flag in MPEG-1 that does not indicate what Petitioners allege to be the first image frame. That is, in MPEG-1 the fact that a `closed_gop` flag has a value of "0" or "1" does not inform a decoder that any type of image frame must necessarily follow. Rather, an MPEG-1 decoder that receives a `closed_gop` flag with a value of "0" or "1" has no certainty about what type of frame will follow, since this is determined by what is actually received in the decoded bitstream, which may for example be affected by transmission errors.

105. The concept of an "indication" that can be decoded from a coded bitstream applies across video coding systems. I will provide an example of an indication from the H.264 Standard.

106. The H.264 Standard defines a “NAL unit” as “A syntax structure containing an indication of the type of data to follow and bytes containing that data in the form of an RBSP interspersed as necessary with emulation prevention bytes.” Ex. 2006, ITU-T Rec. H.264 (03/2005) at 8 (as paginated within the exhibit). In H.264, the NAL unit syntax includes a parameter called `nal_unit_type` that serves as an indication of the type of data contained in a NAL unit. Table 7-1 (excerpted below) provides possible values of `nal_unit_type`. Ex. 2006, ITU-T Rec. H.264 (03/2005) at 57 (as paginated within the exhibit).

Table 7-1 – NAL unit type codes

nal_unit_type	Content of NAL unit and RBSP syntax structure	C
0	Unspecified	
1	Coded slice of a non-IDR picture <code>slice_layer_without_partitioning_rbsp()</code>	2, 3, 4
2	Coded slice data partition A <code>slice_data_partition_a_layer_rbsp()</code>	2
3	Coded slice data partition B <code>slice_data_partition_b_layer_rbsp()</code>	3
4	Coded slice data partition C <code>slice_data_partition_c_layer_rbsp()</code>	4
5	Coded slice of an IDR picture <code>slice_layer_without_partitioning_rbsp()</code>	2, 3
6	Supplemental enhancement information (SEI) <code>sei_rbsp()</code>	5
7	Sequence parameter set <code>seq_parameter_set_rbsp()</code>	0
8	Picture parameter set <code>pic_parameter_set_rbsp()</code>	1

107. A POSITA would understand that, for example, `nal_unit_type = 7` serves as an indication of a sequence parameter set. In other words, if `nal_unit_type = 7`, then the NAL unit is necessarily a sequence parameter set. If (hypothetically) this were not the case, then a decoder would not know with certainty based on `nal_unit_type` what type of data was contained in the NAL unit and `nal_unit_type` would not serve as an indication of the type of data contained in a NAL unit. In such a hypothetical scenario, the decoder would need other information from the bitstream to resolve what type of data was contained in the NAL unit (if the decoder were ever to resolve the uncertainty).

108. Nor can the `closed_GOP` flag be used inherently to determine an initiation picture, or any specific coded frame or picture, in an independently decodable group of pictures. MPEG-1 specifies that a “group of pictures shall contain one or more I-Pictures.” Ex. 1005, 85. That is, MPEG-1 does not constrain the number of I-pictures within a group of pictures and therefore there is no upper bound to that number.

109. The Figures and examples in MPEG-1 illustrate that a wide variation in GOP sizes and structures is possible, such that for example there may be a varying number of I pictures in GOPs that conform to MPEG-1:

Some examples of groups of pictures are given below:

```

I
I  P  P
I  B  P  B  P
B  B  I  B  P  B  P
B  B  I  B  B  P  B  B  P  B  B  P
B  I  B  B  B  B  P  B  I  B  B  I  I

```

Figure D.17 -- Examples of groups of pictures in display order

Ex. 1005, 85.

110. Thus, while MPEG-1's closed_gop flag can indicate whether or not the GOP is closed, it cannot and does not identify or indicate an I-picture in the GOP that is the first image, in decoding order, of that group of pictures.

111. This distinction supports my view that MPEG-1's closed_gop flag fails to meet the requirement of the claim. The “idea of the [’321 Patent’s] invention” “is to determine an initiation picture in an independently decodable group of pictures, whereby in the decoding phase any picture prior to said initiation picture is defined as unusable for reference.” ’321 Patent, 4:31-38. Conversely, if such a picture is not identified or determined, the scalability and random access benefits described by the ’321 Patent would not be achieved.

112. I understand that the Board granted institution and, in doing so, the Board relied on Petitioner’s representation that the following five factors relating to MPEG-1 were all true:

- (1) a GOP header precedes the first image frame in the GOP in decoding order;
- (2) the GOP header contains the closed_gop flag specifying that the GOP is closed;
- (3) the closed_gop flag in the GOP header is decoded before the first image frame in the closed GOP is decoded;
- (4) the closed_gop flag indicates that the closed GOP is an “independent sequence”; and
- (5) the closed_gop flag also indicates that “the next image frame decoded is the first image frame in decoding order” of the “independent sequence,” i.e., of the closed GOP.

113. Based on these five factors, the Board determined that the “indication of at least one image frame, which is the first image frame, in decoding order, of an independent sequence” is met by the closed_gop flag. Paper No. 10, at 26-27. In my view, at least two of these five factors are premised on incorrect interpretations.

114. Considering factor (1), “a GOP header precedes the first image frame in the GOP in decoding order,” this is not always the case in an MPEG-1 bitstream. As Dr. Havlicek agreed, if playback begins from a random point in the bitstream, an MPEG-1 decoder will “discard all bits until it finds a sequence start code, a group of pictures start code, or a picture start code which introduces an I-picture.” Ex. 2005, 101:17-22; Ex. 1003, ¶100.

115. MPEG-1 defines a particular structure for a coded video sequence. A sequence begins with a sequence start code, the first syntax element in a sequence

header, which is followed by one or more GOPs (Ex 1005 at 32).

116. A GOP is a series of one or more coded pictures (Ex 1005 at 32). Each GOP starts with a group_start_code (*Id. at 37*) and contains one or more picture() syntax structures.

117. A picture starts with a picture_start_code (p38) and contains one or more slices.

118. The syntax element closed_gop flag (in the GOP header) indicates whether or not the GOP has been encoded without motion vectors pointing to the previous GOP. Ex. 1005 at p43. It does not indicate the first frame in decoding order.

119. In my view, Petitioner's argument regarding closed_gop reads out of the claim, the requirement to indicate **one image frame** of an independent sequence. Under petitioner's argument, any indication of an **independent sequence** would of necessity indicate **one image frame** of an independent sequence, because any sequence of coded frames necessarily begins with a first frame in decoding order. Petitioner's closed_gop argument rewrites and broadens the claim, from 'decoding an indication of ... the first image frame in decoding order of an independent sequence' to 'decoding an indication of an independent sequence'.

120. In his deposition, Dr. Havlicek admitted that if random access

playback begins after the first I-picture in a closed GOP, an MPEG-1 decoder would not read a GOP header and therefore would not decode the alleged claimed indication (the closed GOP flag). Ex. 2005, 101:17-103:8. Thus, the closed_gop flag of MPEG-1 does not disclose the claimed indication of the first image frame in decoding order because the flag only signals to the decoder whether the *entire* GOP is open or closed and does not indicate which frame is the first image frame in decoding order, particularly for example when a user selects for playback a frame in the middle of a video sequence and the closed_gop flag is not decoded.

121. Considering factor (3), “the closed_gop flag in the GOP header is decoded before the first image frame in the closed GOP is decoded,” the Board and Petitioner appear to assume that the closed_gop flag is always decoded before the first image frame in a closed GOP is decoded. Paper No. 10, at 27. This assumption is incorrect. The closed_gop flag in the GOP header is not always decoded before a first image frame is decoded. For example, consider a typical (and indeed frequently occurring) random access scenario. When a user selects a frame in the middle of a video sequence, MPEG-1 teaches that all bits are discarded until a start code is reached, as discussed above. In Example (c) of Figure D.19 of MPEG-1 (Ex. 1005, 87), when the user selects, for example, frame B4, the first frame that is decoded is frame I8. In this instance, neither the closed_gop flag nor any syntax elements of the GOP header are decoded or read in the bitstream; however, frame I8 is decoded first.

Thus, factor (3) is not true for every MPEG-1 decoding scenario. Although factor (3) may be true in some circumstances, MPEG-1 teaches that it is not true for others. For at least this reason, MPEG-1 does not teach a decoder that decodes an indication of an image frame, which is the first image frame, in decoding order, of an independent sequence.

122. Considering factor (5), “the closed_gop flag also indicates that ‘the next image frame decoded is the first image frame in decoding order’ of the ‘independent sequence,’ i.e., of the closed GOP,” the Board and Petitioner again assume that the closed_gop flag is always decoded before the first image frame in a closed GOP is decoded. Paper No. 10, at 27. This is not the case for MPEG-1. As I discussed above, the first image frame decoded is not necessarily the image frame following the closed_gop flag. Instead, MPEG-1 teaches that the first decoded image frame is the frame following one of the aforementioned start codes and therefore the closed_gop flag is not always decoded and does not indicate the first image frame in decoding order, as discussed above. Thus, for similar reasons discussed for factors (1) and (3), factor (5) is also not taught by MPEG-1.

**ii. MPEG-1’s Temporal Reference Is Not the Claimed
“Indication”**

123. *Second*, Petitioner points to MPEG-1’s Temporal Reference of zero as an alternative teaching of “decoding from the video sequence an indication of at

least one image frame, which is the first image frame, in decoding order, of [the] independent sequence.” Pet. 23-24. The Board and Petitioner are in my view incorrect that MPEG-1’s Temporal Reference of zero teaches this claim element. MPEG-1 does not teach that a Temporal Reference of zero is an indication of the first image frame in *decoding order* of *an independent sequence*. Rather, MPEG-1 teaches that “[t]he first picture, in *display order*, in each group must have Temporal Reference equal to zero.” Ex. 1005, 87, emphasis added.

124. MPEG-1 further teaches that “[t]he Temporal Reference is a ten-bit number which can be used to define the order in which the pictures must be displayed.” *Id.* The Temporal Reference is not used only for *closed GOPs (independent sequences)*, but also *open GOPs (dependent sequences)*. Thus, MPEG-1’s Temporal Reference (signalled by the temporal_reference syntax element) cannot indicate the first image frame in *decoding* order of an *independent* sequence; it is merely the first picture in *display* order as taught by MPEG-1.

D.5.3.2 Temporal reference

The Temporal Reference is a ten-bit number which can be used to define the order in which the pictures must be displayed. It may be useful since pictures are not transmitted in display order, but rather in the order which the decoder needs to decode them. The first picture, in display order, in each group must have Temporal Reference equal to zero. This is incremented by one for each picture in the group.

Some example groups of pictures with their Temporal Reference numbers are given below:

Example (a) in display order	I	B	P	B	P								
	0	1	2	3	4								
Example (a) in decoding order	I	P	B	P	B								
	0	2	1	4	3								
Example (b) in display order	B	B	I	B	B	P	B	B	P	B	B	P	
	0	1	2	3	4	5	6	7	8	9	10	11	
Example (b) in coded order	I	B	B	P	B	B	P	B	B	P	B	B	
	2	0	1	5	3	4	8	6	7	11	9	10	
Example (c) in display order	B	I	B	B	B	B	P	B	I	B	B	I	I
	0	1	2	3	4	5	6	7	8	9	10	11	12
Example (c) in coded order	I	B	P	B	B	B	B	I	B	I	B	B	I
	1	0	6	2	3	4	5	8	7	11	9	10	12

Figure D.19 -- Examples of groups of pictures and temporal references

Ex. 1005, 87.

125. During deposition, Dr. Havlicek confirmed that he did not consider a decoder for MPEG-1 for everything it teaches by way of technology, but instead based his analysis on a modified, limited MPEG-1 decoder that was designed only for closed GOPs, with no B-frames in the beginning, and containing no more than 2^{10} (1024) frames in a GOP. Ex. 2005, 97:21-98:11.

126. Considering the entirety of MPEG-1's teaching, it is not the case that a closed GOP will always have a first picture in decoding order that is an "I-frame with a Temporal Reference of zero" as the Petition suggests. Neither the closed_gop flag nor the temporal_reference syntax element indicate any such constraint. For example, Figure D.19 shows multiple instances where an I-picture is not the first

image in display order and an I-picture does not have a Temporal Reference of zero, in contrast to the claims' requirement that the indication be of the first image frame in *decoding order* (not display order):

D.5.3.2 Temporal reference

The Temporal Reference is a ten-bit number which can be used to define the order in which the pictures must be displayed. It may be useful since pictures are not transmitted in display order, but rather in the order which the decoder needs to decode them. The first picture, in display order, in each group must have Temporal Reference equal to zero. This is incremented by one for each picture in the group.

Some example groups of pictures with their Temporal Reference numbers are given below:

Example (a) in display order	I	B	P	B	P								
	0	1	2	3	4								
Example (a) in decoding order	I	P	B	P	B								
	0	2	1	4	3								
Example (b) in display order	B	B	I	B	B	P	B	B	P	B	B	P	
	0	1	2	3	4	5	6	7	8	9	10	11	
Example (b) in coded order	I	B	B	P	B	B	P	B	B	P	B	B	
	2	0	1	5	3	4	8	6	7	11	9	10	
Example (c) in display order	B	I	B	B	B	B	P	B	I	B	B	I	I
	0	1	2	3	4	5	6	7	8	9	10	11	12
Example (c) in coded order	I	B	P	B	B	B	B	I	B	I	B	B	I
	1	0	6	2	3	4	5	8	7	11	9	10	12

Figure D.19 -- Examples of groups of pictures and temporal references

Ex. 1005, 87.

127. Dr. Havlicek acknowledges that under Patent Owner's understanding of the full scope of an MPEG-1 decoder (which must be designed to handle all required decoding scenarios, including those ones Dr. Havlicek has excluded in his analysis, such as open GOPs or B-frames preceding I-frames in closed GOPs), the temporal_reference syntax element would not be an indication of a first image frame.

Ex. 2005. 98:20-99:22.

128. Thus, it is my opinion that the Petitioner has failed to show that an MPEG-1 decoder discloses an indication of an image frame, which is the first image frame, in decoding order, of an independent sequence as required by the [a] limitations.

b. Petitioner fails to demonstrate that limitations 8[b], 10[b], and 11[b] are obvious over MPEG-1

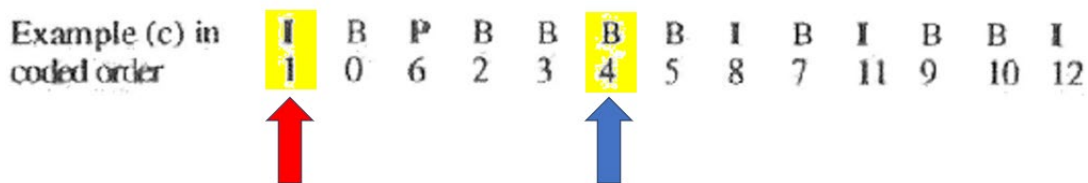
129. Each of limitations 8[b], 10[b], and 11[b] are written with similar language and require an “starting the decoding of the video sequence from said first image frame of the independent sequence, whereby the video sequence is decoded without prediction from any image frame decoded prior to said first image frame” where “from said first image frame of the independent sequence” is the first image frame identified from the “indication of at least one image frame, which is the first image frame, in decoding order, of an independent sequence” from 8[a], 10[a], and 11[a].

130. As I discussed above, MPEG-1 does not teach or suggest the “indication of at least one image frame, which is the first image frame, in decoding order, of an independent sequence” as the claim requires. Thus, MPEG-1 does not teach or suggest starting the decoding of the video sequence from the “first image frame” of the independent sequence under either of the two theories Petitioner and Dr. Havlicek assert for the “indication of at least one image frame, which is the first

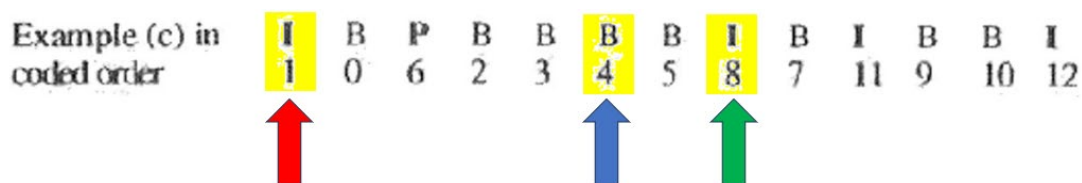
image frame, in decoding order, of an independent sequence.”

131. Further, Dr. Havlicek acknowledged that when considering the full teaching of MPEG-1, an MPEG-1 decoder starts decoding during random access at either a sequence start code, a group of pictures start code, or a picture start code of an I-frame, and discards all preceding bits. Ex. 1003, ¶100; Ex. 2005, 101:17-104:8. That is because MPEG-1 discloses that after a user specifies a random-access point, the decoder looks for the next sequence start code, group of pictures (GOP) start code, or picture start code which introduces an I-picture (collectively, “start codes”) and begins decoding at that point. Ex. 1005, 43, 87, 120. All prior frames are discarded, and decoding and playback starts from that next start code, not from the alleged first image frame of an independent sequence identified by Petitioner.

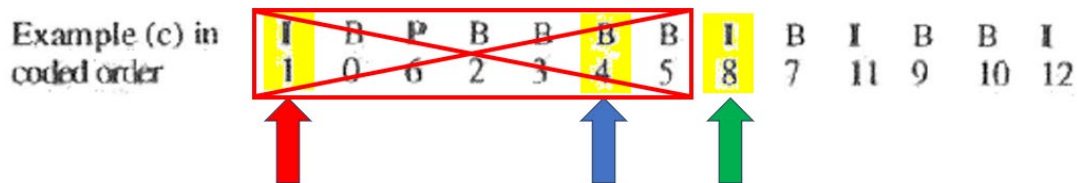
132. To illustrate, I show Example (c) of Figure D.19 below which applies to any MPEG-1 decoder. As shown below, the first frame of the GOP in decoding order is I1 (red arrow). Consider that the user selects, for example, frame B4 (blue arrow) as the playback point.



133. Thereafter, the MPEG-1 decoder starts decoding from frame I8 (green arrow), which is not the first frame of an independent sequence, such as frame I1 (red arrow) which Petitioner identifies as the first frame of an independent sequence. MPEG-1 starts decoding from I8 (green arrow) because MPEG-1 teaches that after a user specifies a random-access point, the decoder scans forward in the coded bitstream for the next sequence start code, group of pictures (GOP) start code or picture start code which introduces an I-picture (collectively, “start codes”) to start decoding. Ex. 1005, 43, 87, 120. In Example (c), the next start code is the picture start code which introduces an I-picture, I8. *See id.*



134. Thus the decoder discards all frames until frame I8 and starts decoding and playback from I8 because MPEG-1 teaches that all prior frames are discarded and decoding and playback starts from that next start code and not from the first image frame of an independent sequence, as claimed in the '321 Patent. Ex. 1005, 43, 87, 120.



135. Further, none of the start codes disclosed in MPEG-1 are themselves first image frames of independent sequences. Instead, the start codes identify “some of the layers in the coding syntax.” Ex. 1005, 26, 31, 41-44. Moreover, additional information follows the start codes before and is parsed and decoded prior to decoding the following image frame. *Id.*, 36-38.

2.4.2.3 Sequence header

Syntax	No.of bits	Mnemonic
sequence_header() {		
sequence_header_code	32	bslbf
horizontal_size	12	uimsbf
vertical_size	12	uimsbf
pel_aspect_ratio	4	uimsbf
picture_rate	4	uimsbf
bit_rate	18	uimsbf
marker_bit	1	"1"
vhv_buffer_size	10	uimsbf
constrained_parameters_flag	1	
load_intra_quantizer_matrix	1	
if (load_intra_quantizer_matrix)		
intra_quantizer_matrix []	8*64	uimsbf
load_non_intra_quantizer_matrix	1	
if (load_non_intra_quantizer_matrix)		
non_intra_quantizer_matrix []	8*64	uimsbf
next_start_code()		
if (nextbits() == extension_start_code) {		
extension_start_code	32	bslbf
while (nextbits() != '0000 0000 0000 0000 0000 0001') {		
sequence_extension_data	8	
}		
next_start_code()		
}		
if (nextbits() == user_data_start_code) {		
user_data_start_code	32	bslbf
while (nextbits() != '0000 0000 0000 0000 0000 0001') {		
user_data	8	
}		
next_start_code()		
}		
}		

Ex. 1005, 36.

2.4.2.4 Group of pictures layer

Syntax	No. of bits	Mnemonic
group_of_pictures() {		
group_start_code	32	bslbf
time_code	25	
closed_gop	1	
broken_link	1	
next_start_code()		
if (nextbits() == extension_start_code) {		
extension_start_code	32	bslbf
while (nextbits() != '0000 0000 0000 0000 0000 0001') {		
group_extension_data	8	
}		
next_start_code()		
}		
if (nextbits() == user_data_start_code) {		
user_data_start_code	32	bslbf
while (nextbits() != '0000 0000 0000 0000 0000 0001') {		
user_data	8	
}		
next_start_code()		
}		
do {		
picture()		
} while (nextbits() == picture_start_code)		
}		

Ex. 1005, 37.

2.4.2.5 Picture layer

Syntax	No. of bits	Mnemonic
picture() {		
picture_start_code	32	bslbf
temporal_reference	10	uimsbf
picture_coding_type	3	uimsbf
vbv_delay	16	uimsbf
if ((picture_coding_type == 2) (picture_coding_type == 3)) {		
full_pel_forward_vector	1	
forward_f_code	3	uimsbf
}		
if (picture_coding_type == 3) {		
full_pel_backward_vector	1	
backward_f_code	3	uimsbf
}		
while (nextbits() == '1') {		
extra_bit_picture	1	"1"
extra_information_picture	8	
}		
extra_bit_picture	1	"0"
next_start_code()		
if (nextbits() == extension_start_code) {		
extension_start_code	32	bslbf
while (nextbits() != '0000 0000 0000 0000 0000 0001') {		
picture_extension_data	8	
}		
next_start_code()		
}		
if (nextbits() == user_data_start_code) {		
user_data_start_code	32	bslbf
while (nextbits() != '0000 0000 0000 0000 0000 0001') {		
user_data	8	
}		
next_start_code()		
}		
do {		
slice()		
} while (nextbits() == slice_start_code)		
}		

Ex. 1005, 38.

136.

137. Thus, Petitioner has failed to show that an MPEG-1 decoder discloses starting the decoding of the video sequence from the first image frame of the independent sequence.

- c. **Petitioner fails to demonstrate that limitations 8[d], 10[d], and 11[d] are obvious over MPEG-1 because MPEG-1 does not teach or suggest a method for decoding, a video decoder or a computer program product for decoding, resetting the identifier value as required**

138. Each of limitations 8[d], 10[d], and 11[d] are written with similar language and require “[resetting/reset] the identifier value for the indicated first image frame of the independent sequence.” Petitioner has failed to show that MPEG-1 teaches or suggests these [d] limitations.

139. Petitioner alleges that the Temporal Reference (temporal_reference syntax element) being set to zero in an MPEG-1 bitstream discloses the claimed “resetting the identifier value for the indicated first image frame of the independent sequence.” Petitioner does not provide a separate argument for its ‘closed_gop flag’ or ‘temporal_reference syntax element’ arguments discussed above with respect to earlier claim limitations. In my opinion, Petitioner is wrong.

140. First, the challenged claims require the **decoder** (not the encoder) to reset the identifier value for the indicated first image frame of the independent sequence. The ‘321 patent is clear that this is an important aspect of the invention: “Consequently, when activated at a random point, the **decoder preferably sets the identifier** of the first image frame, preferably an I-frame, of the independently decodable sub-sequence.” ’321 Patent, 11:26-40, emphasis added. Dr. Havlicek

acknowledged this. Ex. 2005, 38:19-39:1 (“They [the challenged claims] are all saying a decoding method and implementations thereof.”). The Board, in the Institution Decision, acknowledged that Temporal Reference is “assigned (in sequence) to the . . . pictures in display order,” beginning with 0 for the first picture in display order. Paper No. 10, at 35 (citing Ex. 1005, 44, 87). That is, the Temporal Reference is assigned by the encoder by means of the `temporal_reference` syntax element. Thus, MPEG-1’s temporal reference can only be set or reset by the **encoder** for the first image frame in display order. Ex. 1005, 44, 87. Thus, MPEG-1 fails to teach this claim element.

141. The Temporal Reference (`temporal_reference` syntax element) is a syntax element whose value is determined by the encoder and which informs the decoder of the display order of decoded pictures “since pictures are not transmitted in display order, but rather in the order which the decoder needs to decode them.” Ex. 1005, 87.

142. But MPEG-1 does not teach that the **decoder resets** the Temporal Reference to 0. At most, it teaches that the **encoder** assigns the Temporal Reference of a first picture in display order to zero in certain situations and that the decoder reads that syntax element from the bitstream, which may be zero-valued. Ex. 1005, 87. Petitioner argues that after decoding an exemplary closed GOP, “when an MPEG-1 decoder decodes the next exemplary closed GOP, the Temporal Reference

for the first picture in decoding order of that exemplary closed GOP **would be reset to zero**, as that picture would be the first picture in display order for that next GOP.” Pet. 28, emphasis added.

143. The Board, in the Institution Decision, cited the same passage. Paper No. 10, at 35-36 (citing to Ex. 1005, 44; *see* Ex. 1003, ¶¶ 109-110). In my view, this argument fails for two reasons. First, the encoder, not the decoder assigns the Temporal Reference value and therefore there is no system or method for decoding in MPEG-1 that resets the Temporal Reference value. Ex. 1005, 87. Petitioner appears to acknowledge this through the choice of language above, acknowledging that the Temporal Reference ‘would be reset to zero’ but not that an MPEG-1 decoder itself performs ‘resetting’ this value.

144. Second, the claim language requires “resetting the identifier value for *the* first image frame of *the* independent sequence.” *See* ’321 Patent, Claim 8 (and similar for other challenged claims), emphasis added. That is, the resetting of the identifier value is reset for the independent sequence of the prior limitations, which I understand to be the antecedent basis for this limitation.

145. However, Petitioner’s argument relies on the Temporal Reference being set to zero for the *subsequent first image frame of the subsequent* independent sequence, not the first image frame of the actual, claimed independent

sequence being decoded: “when an MPEG-1 decoder decodes *the next* exemplary closed GOP, the Temporal Reference for the first picture in decoding order of *that exemplary closed GOP* would be reset to zero, as that picture would be the first picture in display order for *that next GOP*.” Pet. 28 (emphasis added). Thus, even accepting Petitioner’s argument concerning resetting (which I do not accept, as Petitioner cannot show the decoder resetting any identifier value), Petitioner has pointed to a resetting of “the next independent sequence” not “the independent sequence” referred to earlier in the same claim.

146. Third, the claims require the decoder to reset the identifier value for the claimed first image frame of the independent sequence, which, as discussed above for Element [a], has been identified by Petitioner as the first image frame in *decoding order*. In contrast, MPEG-1’s Temporal Reference is only reset by the encoder for the first image frame in *display order*. Ex. 1005, 44, 87. Thus, MPEG-1 fails to teach this claim element.

147. Further, for Petitioner to make its argument, it has to show that the Temporal Reference must be reset (set to zero) for the indicated first image frame of the independent sequence. As I have shown above (III.B.1), MPEG-1 does not require that an I-frame (which is what Petitioner asserts to be the indicated first image frame in decoding order) always have a temporal reference of zero, and as a result there is no “resetting” of an identifier value pertaining to an I-frame, whether

or not it is the first I-frame of a GOP. This is confirmed by Fig. D.18(b), which shows B-frames with Temporal Reference values of 0 and 1, while the first I-frame has a Temporal Reference of 2. Ex 1005 at 86.

148. Thus, Petitioner has failed to show that an MPEG-1 decoder discloses the claimed resetting the identifier value for the indicated first image frame of the independent sequence.

2. Petitioner Has Not Established That The Challenged Claims Are Unpatentable In Light Of Ground 2—Kim.

149. Here, I will summarise what I understand to be Petitioner's arguments concerning the decoding process of Kim, as a whole, so as to provide context for the following discussion.

150. Petitioner suggests that Kim's grouping of an I-picture followed by P-pictures that refer to it, teaches the claimed 'independent sequence' (Petition at 44), that the I-picture teaches the claimed 'first image frame of the independent sequence' and that decoding an I-picture address IAD=0000 teaches the claimed 'indication of at least one image frame.

151. Petitioner identifies a 'first path' and a 'second path'. The 'first path' comprises a user request for a picture that is stored as an I-picture frame (petition at 50). The 'second path' comprises a user request for a picture that is stored as a P-

picture frame (petition at 51).

152. As I explain below, when Kim's decoder processes the so-called 'first path', which is the path annotated in red in Petitioner's Figure 4 (on Pet. at 42) which refers to the user-selected frame being an I-frame, it does not decode, detect or use an IAD value. Specifically, step S34 of Fig. 4 is not carried out because step S32 identifies that the requested frame is not a P-picture frame. That is, the IAD is never decoded, interpreted or otherwise used when the user-selected frame is an I-frame. Thus at least claim 8a is not satisfied in this instance.

153. When Kim's decoder processes the so-called 'second path', which is the path annotated in blue in Petitioner's Figure 4 (on Pet. at 42) which refers to the user-selected frame being a P-frame, it does not start decoding of the video sequence from the alleged 'first frame of the independent sequence', because the first frame that is decoded is the requested P-picture frame (Kim Fig. 4, step S30) and only later in the process illustrated in Figure 4 is the I-picture frame decoded (Kim Fig. 4, step S35). Thus at least claim 8b is not satisfied in this instance.

**a. Petitioner fails to demonstrate that limitations 8[a],
10[a], and 11[a] are obvious over Kim**

154. Each of limitations 8[a], 10[a], and 11[a] (the [a] limitations) are written with similar language and require "[decoding/decode] from the video sequence an indication of at least one image frame, which is the first image frame,

in decoding order, of an independent sequence, wherein all motion-compensated temporal prediction references of the independent sequence refer only to image frames within said independent sequence.” Petitioner has failed to show that Kim teaches or suggests these [a] limitations.

155. Petitioner identified the 0000 IAD value of one or more I frames as allegedly satisfying an indication of the first image frame in decoding order, when a user selects a P-picture frame for playback. However, the user-selected P-picture frame (not the corresponding I-picture frame of what Petitioner alleges is a sequence) is decoded first and thus the 0000 IAD value cannot indicate the first image frame in decoding order. Ex. 1006, 5:12-42, Fig. 4.

156. Petitioner points to Figure 5A of Kim as “show[ing] the recited “indication of at least one image frame, which is the first image frame, in decoding order, of an independent sequence.” Pet. 44. Specifically, Petitioner states that “Kim also teaches the recited ‘indication’” because “in Kim’s Figure 5A, an IAD of 0000 or zero is ‘an indication of at least one image frame, which is the first image frame, in decoding order, of an independent sequence.’” However, an IAD of 0000 in Kim is merely a property of Kim’s stored I-frame.

157. During deposition, Dr. Havlicek confirmed that decoding starts at Kim’s step S23, which determines the user-selected frame. Ex. 2005, 28:6-20.

Specifically, “under the assumption that a P-picture was requested, then some elements of the data associated with the P-picture will be decoded again in diamond S23.” Ex. 2005, 27:3-28:5, referring to Kim (Ex 1006) Fig. 4 and 5:23-29. Thus, even if Kim decodes an I-frame’s IAD in the process of decoding a user-selected P-frame, the decoder nevertheless first decodes that user-selected P-frame (at step S30, Kim Fig.4), not the I-frame that has an associated IAD of 0000 (at step S35, Kim Fig.4).

158. Thus, even accepting all of Petitioner’s arguments about an I-frame’s IAD of 0000 being an indication of an image frame, an IAD value of 0000 is not an indication of the first image frame in decoding order of the independent sequence because Dr. Havlicek confirmed that Kim starts decoding at the user selected frame regardless of whether it is an I-frame or a P-frame. Ex. 2005, 28:6-20.

159. Further, Petitioner fails to consider that Kim confirms that these IAD values (IAD of 0000 associated with an I-frame) are not decoded by the decoder at all, and are at most used by the controller after the decoder steps occur. Ex. 1006, 5:42-43. Petitioner’s argument relies on a POSITA combining the bitstream of Figure 5B with the decoding pathways of Figure 4. Pet. 41. Thus under Petitioner’s reasoning, Figure 4 provides the only source of potential evidence for Petitioner relating to decoding. But Kim makes it clear in its description of Figure 4 that the controller of Kim does not consider IAD information until after it decodes the

requested frame, as follows.

160. Kim explains in column 5 that the decoder first acts to reproduce the correct, requested image data (lines 11-25, also shown as steps S21, S22, and S23 in Figure 4). Only after decoding that data (Step 30, 5:29-31) does the Kim decoder actually look to determine what “type” the original data is. Fig. 4 (step S31/S32); 5:38-45. If it started decoding at the corresponding I-picture, which Petitioner identifies as the claimed first image frame in decoding order, it would not need to determine what type (I- or P-) the original data is to begin decoding.

161. The Board acknowledged that “Kim discloses that ‘the image data corresponding to the user’s request is transmitted and decoded’ by a decoder” and then “[a]fter decoding the image data, Kim’s controller uses the IAD in the image data, e.g. to ‘detect[] the address of the corresponding I-picture frame’ if the image data is a P-picture.” Paper No. 10, at 58. This confirms that an I-frame’s IAD of 0000 in Kim is not an indication of a first image frame in decoding order of an independent sequence because it does not indicate at which image frame the decoder should begin decoding, since when the user selected frame is a P-frame, Kim begins decoding at the P-frame.

162. During deposition, Dr. Havlicek suggested that the I-frame’s IAD or CAD may be read before step 23 or outside of Figure 4. Ex. 2005, 51:1-16. To the

extent Petitioner attempts to rely on this suggestion, such an argument is not disclosed in the Petition or in Dr. Havlicek's declaration and therefore would in my view be an improper new argument. Further, decoding IAD and CAD values during the process of finding the user-selected frame does not disclose decoding an identification of a first image frame in decoding order, because (as Dr. Havlicek admits) the first image frame to be decoded is the one the user selected, whether it is an I-frame or a P-frame and regardless of any associated IAD or CAD values. Moreover, the IAD and CAD values that Petitioner pointed to are the logical addresses shown in Kim Figure 5A . Dr Havlicek confirmed that when searching on optical medium, Kim would utilize the physical address:

Q. Kim discloses recording onto an optical disk, which is a physical media, right?

A. Correct.

Q. And the way one would seek into a physical media would be, for example, going into the physical address of where that information is stored, correct?

A. At the low level of the device, yes, I agree.

Ex. 2005, 55:16-56:4.

163. Kim does not describe the physical address of the requested frame or the corresponding I-frame as the IAD or CAD, but treats those as separate values.

See Ex. 1006, 4:59-5:7 (describing the physical addresses). Moreover, in my view the physical addresses of coded image frames could not be the disclosed IAD or CAD values because each value would need to be unique as a physical address when stored on the optical medium, whereas Kim discloses that each I-frame picture IAD value ‘is zero or absent’ (Ex. 1006, 4:67-5:1), i.e. each I-frame IAD value may be the same.

164. Kim explains that the IAD or CAD values are logical addresses (as disclosed in Ex. 1006, 4:46-58). These do not need to be unique because the IAD points to the “corresponding or previous I-frame” and is a relative pointer to the requested user frame, not an absolute or physical address. *See* Ex. 1006, 4:46-58.

165. Thus, Petitioner has failed to show that Kim discloses an indication of an image frame, which is the first image frame, in decoding order, of an independent sequence as required by the [a] limitations.

**b. Petitioner fails to demonstrate that limitations 8[b],
10[b], and [11[b] are obvious over Kim**

166. Each of limitations 8[b], 10[b], and 11[b] are written with similar language and require “starting the decoding of the video sequence from said first image frame of the independent sequence.” It is my opinion that Kim fails to disclose these limitations because Kim teaches starting decoding from the user-selected frame, not from the first image frame of the alleged independent sequence.

Ex. 1006, 5:12-38, Fig. 4. Kim's decoding begins at step S30 of Figure 4, where the user-selected image is decoded. Here, when Petitioners point to the indication, they refer to the IAD of 0000 in the I-frame, but this detecting occurs in Step S34, which only occurs in the P-frame path, in which the I-frame is decoded second, after decoding the P-frame data.

167. Petitioner argues that the indicated first image frame in decoding order is the I-frame of Figure 5A that precedes the P-frames that refer to it through the IAD. Pet. 47. As stated above, however, instead of starting the decoding of the video sequence from said first image frame (the one for which the indication identified), Dr. Havlicek admitted that Kim teaches starting the decoding of the video sequence from the user selected frame, which can be a P-picture frame, rather than starting the decoding of the video sequence from "said first image frame", which Petitioner identifies as the corresponding I-picture frame:

"Q. And where will that decoding process start in either of those two cases [whether the user requests an I-picture or a P-picture]?"

A. As you said correctly, in the diamond S23."

Ex. 2005, 28:18-20

"Q. And so the first step that Kim takes in [servicing the user request] is it's going to start the decoding of whatever frame or image that the user requested by whichever manner that that happens, like you just talked about; is that right?"

A. I agree.

Ex. 2005, 48:8-13.

168. Thus, as Petitioner's expert admits, Kim does not start decoding at the alleged first image frame of the independent sequence identified by Petitioner (the I-frame).

169. I understand that Petitioner may argue that before the user selected frame is displayed as an output, Kim discloses decoding the I-frame. *See, e.g.*, Pet. 51-52 ("A POSITA would have understood this second path when applied to Figure 5A's independent sequence to disclose 'starting the decoding of the video sequence from said first image frame of the independent sequence' because the system decodes the I-picture (the first image frame of the independent sequence) before it can produce the P-picture that depends on it."). But the claim language requires "starting the decoding of the video sequence." In this apparent argument, Petitioner adds an additional requirement that "starting the decoding" must mean producing the image data for display. This is not supported by Kim, which clearly describes decoding the user-requested data (which may be P-picture data), step S30, before decoding the corresponding I-picture data, step S36 (Kim 5:29-52).

170. Further, Petitioner argues that this limitation would be disclosed by considering only the I-frame path in Kim (i.e. if Kim did not have any P-pictures). *See* Pet. 51. However, Kim's disclosure must be understood for all of its teachings, as a time lapse device for recording and reproducing intermittent data using P-

pictures, i.e. for recording and reproducing compressed image data such as I-picture frame data and P-picture frame data. *See* Paper No. 10, at 43-44 (citing Ex. 1006, 3:60-61, 4:19, 4:28-33, 6:4-8, 6:55-58 (claim 3)).”

171. In my view, a POSITA would not reasonably view Kim’s disclosure of processing I-frames as teaching a decoder that starts decoding an independent sequence at said first image frame. Further, the said first image frame is what Petitioner has identified is the frame that corresponds to an I-frame’s IAD of 0000. If Kim were to be re-designed as Petitioner appears to suggest, to only decode I-frames, an I-frame IAD that is always set to zero or absent (as disclosed in Kim, 4:65-5:1) would not identify any particular image frame (since e.g. every I-frame disclosed in Kim’s Fig. 5A and 5B has the same IAD of 0000) and would be redundant since all frames would be I-frames. Thus Petitioner’s argument does not appear to make technical sense.

172. Thus, in my view Petitioner has failed to show that Kim discloses “starting the decoding of the video sequence from said first image frame of the independent sequence.”

**c. Petitioner fails to demonstrate that limitations 8[c],
10[c], and 11[c] are obvious over Kim**

173. Each of limitations 8[b], 10[b], and 11[b] are written with similar language and require “decoding identifier values for image frames according to a

numbering scheme.” It is my opinion that Petitioner has failed to show this limitation is disclosed or taught by Kim.

174. Petitioner points to, and the Board adopts Petitioner’s position, that the logical addresses in Figure 5A as the identifier values for image frames according to a numbering scheme that are decoded. But this is incorrect. Kim does not disclose decoding these logical values at all.

175. When Kim begins decoding, as stated above, Kim starts with the user-selected frame. And Dr. Havlicek acknowledged, and I agree, that recording and reproducing onto an optical disk, which is physical media, would be to utilize the physical address of what that information is stored. Ex. 2005, 55:16-56:4 (Q. Kim discloses recording onto an optical disk, which is a physical media, right? A. Correct. Q. And the way one would seek into a physical media would be, for example, going into the physical address of where that information is stored, correct? A. At the low level of the device, yes, I agree.”).

176. So, when beginning decoding of the user-selected frame, Kim does not disclose using the logical address. Instead, if anything, the physical address is used. For an I-frame, no other address is required (and Step 34 is never encountered). For a P-frame, Kim does disclose using an address for the I-frame, but the physical address. Not the logical address that Petitioner points to.

177. More specifically, the IAD indicates the address of the corresponding (or previous) I-frame. For example, in Step S34, Kim's controller "detects the address of the corresponding I-picture frame (e.g. data stored in the area (3) of the bit stream B2 shown in Fig. 6) from the reproduced image data output from the de-MUX 131." Ex. 1006, 5:41-44. Area (3) of the bit stream B2 shown in Fig. 6 is "the address of the corresponding I-picture frame," *id.* at 4:59-65, which Fig. 6 confirms is the physical address of the I-frame (not the logical address). For an I-frame, no other address is required (and Step 34 is never encountered). For a P-frame, Kim does disclose using an address for the I-frame, but the physical address. Not the logical address that Petitioner points to.

178. Thus, only physical addresses are decoded. But these physical addresses are not according to a numbering scheme. Figure 6 further illustrates why the CAD is an address that is not according to a numbering scheme. We know this because the CAD is dependent on an image size, which is variable and not predictable. For example, Figure 6 shows that the address for the second P picture is $100000 + N$, where N is the Byte length of the I frame (B1). Specifically, N is not determined according to a numbering scheme, but to the encoded size of the picture (which depends on the coding type and size of the picture data) that is necessarily variable and unpredictable.

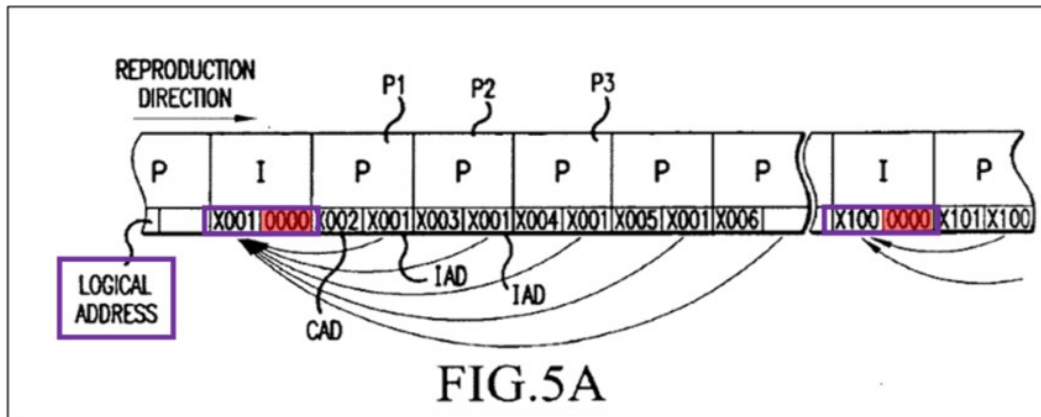
179. Thus, the logical addresses are not decoded by Kim and thus can't

meet this limitation. Likewise, the physical addresses are not according to a numbering scheme, and Petitioner has not shown to the contrary. Thus, this limitation is not met.

d. Petitioner fails to demonstrate that limitations 8[d], 10[d], and 11[d] are obvious over Kim because Kim does not teach or suggest a decoder resetting the identifier value as required

180. Each of limitations 8[d], 10[d], and 11[d] (the [d] limitations) are written with similar language and require “[resetting/reset] the identifier value for the indicated first image frame of the independent sequence.” Petitioner has failed to show that Kim teaches or suggests these limitations because the claims require that the *decoder* reset the identifier value, but Kim teaches at most that the *encoder* sets the identifier value. Kim never teaches a decoder resetting any identifier values.

181. Contrary to Petitioner’s argument, Kim does not teach that its reproducing method resets the identifier value, IAD. Petitioner argues, at Pet. 55, that Figure 5A “shows that the IAD component of each logical address is reset to zero for each I-picture” and provides an annotated Figure 5 (reproduced below) where the I-picture logical address is in red and the “whole logical address” is outlined in purple:



Pet. 56.

182. Likewise, the Board, in its Institution Decision pointed solely to Figure 5A as the disclosure of the indicated value being reset. Paper No. 10, at 63. I disagree with Petitioner's argument.

183. First, Kim describes that Figure 5A "illustrate[s] examples of logical data structure of a digital data stream *compressed and recorded on the optical disk.*" Ex. 1006, 2:48-51 (emphasis added). Of course, compressing and recording of such a data stream occurs during encoding of the data, not during decoding or reproducing of the data. Kim discusses the act of reproducing (as shown in Figure 4) as the decoding of the image data (Kim 5:8-6:8). Dr Havlicek appears to agree with this understanding, with the caveat that he believes there may be some part of decoding that occurs which is shown in Figure 4, concerning the IAD of an I-frame. Ex. 2005, 37:3-38:15.

184. Petitioner points to Figure 4 (which Kim describes as a reproducing method) as allegedly disclosing two paths for decoding an image in response to a

user requesting for image reproduction. *See* Pet. 48. Thus, an IAD value that is recorded in the logical structure of Figure 5A is generated during the encoding process and not the decoding process. Therefore, Kim does not and cannot disclose resetting the alleged identifier value during decoding (i.e. as part of a method for decoding or a video decoder).

185. Kim does not teach or suggest resetting any IAD value when decoding or reproducing the requested image data. Thus, Petitioner has failed to show that Kim teaches a method for decoding or a video decoder “[resetting/reset] the identifier value for the indicated first image frame of the independent sequence” and therefore has failed to show that the [d] limitations are satisfied.

3. Petitioner Has Not Established That The Challenged Claims Are Unpatentable In Light Of Ground 3.

a. Petitioner fails to demonstrate that Claim 9 is obvious over MPEG-1 in combination with Yagasaki

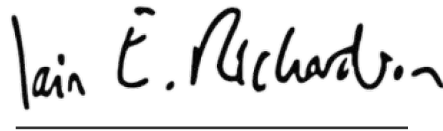
186. It is my opinion that Petitioner cannot show it has met its burden for Ground 3. Claim 9 depends on Claim 8 and Petitioner relies on its Ground 1 to show that independent claim 8 is taught or suggested. As I have explained above, MPEG-1 fails to teach or suggest at least limitations 8[a], 8[b] and 8[d]. For at least that reason, Ground 3 fails.

IX. SIGNATURE

This Declaration represents the opinions that I have formed up to date based on the material currently available to me. I reserve the right to revise, supplement, and/or amend my opinions stated in this Declaration based on new information that becomes available to me and on my continuing analysis of the materials available to me.

I declare that all statements made herein of my knowledge are true, and that all statements made on information and belief are believed to be true, and that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine, imprisonment, or both, under Section 1001 of Title 18 of the United States Code.

Dated: December 18, 2024

A handwritten signature in black ink that reads "Iain E. Richardson". The signature is written in a cursive, slightly slanted style. Below the signature is a horizontal line.

Dr. Iain Richardson